

컴퓨터를 이용한 실제에 준하는 FMS 구축[†]

김성식* · 배경한*

Building a Real-like FMS Using a Computer Network

Sung-Shick Kim* and Kyoung-Han Bae*

Abstract

This study proposes a half-simulation-half-real system that is used as an FMS building tool. In this hybrid system, softwares related to the operation of the FMS and the computer network on which the softwares run are real, while the physical movements of the devices in the system are simulated. The study shows the structure of the proposed system as well as the building procedure for the system. Advantages and usages of the system are also stated in detail.

1. 서 론

자동화 공장의 개념은 단체기계의 자동화에서 시작하며, 공장 전체에 걸쳐 통신수단으로 연결된 기계들의 작동은 물론 물자의 흐름과 재고 등의 공장 전체의 운용 자동화로 진전되고 있다. FMS나 소위 CIM 개념 하의 공장들을 이러한 진보된 자동화의 결과로 볼 수 있다.

FMS 구축과정은 시스템 도입고려 및 목표설정 단계, 개념설계 단계, 상세설계 단계, 장비구축 및 관련 소프트웨어 개발단계, 그리고 조정단계[5]를 거치게 된다. 이러한 과정은 FMS를 새로 구축하는 경우나 기존의 공장을 자동화시키는 경우 등에 따

라서 다소 차이는 있으나 공통적으로 활용되는 절차이다.

공장을 구축할 때 통상적으로 수행하는 절차를 보기로 하자. 먼저 기존의 공장을 자동화 또는 FMS화할 때를 보면 기계 또는 기계군의 자동화를 계속하여 완전히 자동화된 공장을 꾸미게 된다. 의사결정자가 기계의 자동화 우선순위를 결정하면, 먼저 가장 우선순위가 높은 기계들을 자동화한다. 이들을 가동하다가 기계군으로 묶이고 이들을 다시 연결하여 자동화 공장으로 진보한다. 즉, 자동화 섬(island of automation)[2,7]들을 구축하고 이들을 연결하여 자동화 공장을 취하는 과정을 거치게 된다. 따라서, 관련 소프트웨어들도 기계를 직접

[†] 본 연구는 학술진흥재단의 지원에 의하여 이루어졌음

* 고려대학교 산업공학과

움직이는 하부통제용 소프트웨어에서 Cell 통제용과 공장 전체의 운용 소프트웨어로 확장 진보하게 된다. 즉, Bottom-up 형태의 접근방법을 사용하게 된다.

이때 야기되는 문제는 비록 마스터플랜에 따른 세부설계가 이루어졌다고 하여도 오랜 시간에 걸쳐 필요한 부분별로 장비들을 자동화하기 때문에 이들을 LAN을 이용하여 통합할 때 컨트롤러간에 호환성 문제가 발생하며, 기계 또는 Cell별로 작성된 소프트웨어들이 전체의 관점에서 최선인지의 여부에 관한 문제, 전체 운용프로그램에 연결하는 문제, 이런 문제에 따른 공장 운용프로그램의 유연성 상실, 각각의 프로그램의 공통성 부족으로 인한 데이터베이스 구축의 어려움 등의 문제가 발생하기 쉬워서 통합시 상당한 어려움을 겪게 된다.

공장을 처음부터 꾸미는 경우를 보자. 이 경우에 상설설계는 일반적으로 시뮬레이션을 이용하여 검증 및 완성되며, 이 설계에 따라 장비를 발주하게 된다. 이때의 문제는 비록 상설설계에 따라 기계를 도입하고 이를 위한 소프트웨어가 개발된다고 하더라도 소프트웨어의 테스트는 장비의 구축이 완료된 다음에만 가능하다는 점이다. 또한 실제 상황에서 빈번히 일어나는 설계 또는 장비의 변경에 따라서 이들을 운용하는 소프트웨어도 변하게 되어 프로그램을 바꾸거나 때에 따라서 전체 구조까지 바꾸는 일이 발생하기도 한다.

만일 우리가 건설하고자 하는 FMS가 이미 존재한다면 앞에서 제시한 두가지 경우의 문제들은 발생하지 않거나 최소화될 것이다. 즉, 이미 원하는 시스템이 존재하므로 새로운 공장은 단순히 기존의 공장을 복제하므로써 건설하면 되기 때문이다.

문제는 그러한 공장이 존재하지 않는다는 것이며, 본 연구는 건설 예정인 FMS를 컴퓨터와 네트워크만으로 완성하여 복제의 원형을 만드는 데 초점을 맞추고 있다.

본 연구에서 제시하는 시스템은 컴퓨터 네트워크, 운용 소프트웨어와 데이터베이스는 공장에서

가동될 시스템과 같고, 단지 물리적 장비만은 그 장비를 통제하는 해당 컴퓨터에서 시뮬레이션을 하게 된다. 즉, 장비를 제외하고는 실제 공장과 동일한 공장을 컴퓨터 네트워크상에 구축하는 반 실제 반 시뮬레이션(half-real-half-simulation) 시스템이다.

이때 제시되어지는 의문은 시스템을 현대의 컴퓨터에서 모두 시뮬레이션으로 처리가 가능하지 않는가 하는 점이다. 그러나, 본문에서 자세히 보아도 이겠지만 네트워크상의 통신문제와 각 해당기기 또는 Cell 통제컴퓨터의 프로그램 구조 및 형태는 주어진 장치와 시스템 특성에 따라 변하기 때문에 이것들까지 시뮬레이션으로 실제와 같이 처리하려면 오히려 실제기기를 사용하는 것보다 더욱 복잡하고 반면에 단순화시키면 현실성이 결여되어 실용성이 저하된다.

본 연구에서 제시하는 시스템은 생산기계보다 훨씬 값은 싸지만 실질적으로 공장을 통제하는 핵심요소인 컴퓨터와 네트워크, 그리고 시스템 운용을 담당하는 운용 소프트웨어는 현실과 같이 처리하고, 이들 요소보다 상대적으로 값이 비싸고 단지 명령에 따라 움직이는 부분만은 시뮬레이션으로 처리하게 한다.

이러한 시스템은 완성될 공장을 실험하는 도구이며, 구축되어지는 FMS의 골격으로 모두 사용하게 된다. 새로운 공장건설시에는 이 시스템을 이용하여 FMS 운영시스템을 완성한 후 이에 알맞은 실제장비를 도입하여 물리적 FMS를 구축할 수 있고, 기존의 재래식 생산공장에는 이러한 컴퓨터 네트워크로 이루어진 시스템을 먼저 도입하고 컴퓨터 디스플레이와 입력장치를 이용하여 작업자에게 작업명령을 주고 또한 보고를 받으며 운영을 통제컴퓨터가 하계한 후 향후 투자계획에 따라서 점진적으로 장비를 자동화시키고 통제컴퓨터에 연결시켜서 인간의 개입을 줄이고 FMS를 구축하여 나갈 수 있다.

본 연구는 이와같이 실제에 준하는 FMS 구축

모형을 이용하여 FMS를 구축하는 절차와 FMS 구축모형의 구축절차를 소개하고, FMS 구축모형을 구축하는 도구로서 실제 사건과 시뮬레이션 사건의 동기화(synchronization) 방법, 각 컴퓨터의 프로그램 구조와 역할을 보이게 된다.

2. FMS 구축 절차

일반적으로 FMS 구축단계는 시스템 도입고려 및 목표설정 단계, 개념설계 단계, 상세설계 단계, 장비구축 및 관련 소프트웨어 개발단계, 그리고 조정단계로 구분한다. 본 연구에서 제안하는 FMS 구축방법은 일반적인 FMS 구축단계에 맞추고 있으며 구축방법은 <그림 1>에 요약되어 있다.

시스템 도입고려 및 목표설정 단계에서는 FMS 도입시의 경제성, 성능, 그리고 회사의 능력 등을 판단하며, 개념설계 단계에서는 이들을 종합하여 개괄적인 시스템을 설계한다. 기존에 구축된 시스템이 있으면서 이를 FMS로 개조하고자 할 때는 기존의 시스템을 FMS로 전환할 수 있는지의 여부를 면밀하게 분석한다.

경제성 평가가 끝나고 현재 시스템 평가분석(기존 시스템 개조시), 개략 설계가 완성되면 이에 따라 상세설계가 이루어지게 된다. 상세설계에서는 개략설계에서 제시된 공장의 형태, 가공제품 종류, 이에 따른 기계의 성능 형태 등을 기초로 목표한 제품의 질과 양이 제대로 생산되도록 기계의 사양, 제품의 상세한 흐름, 공정관리 방법 등을 정하고 여기에 적합한 데이터베이스, 통신방법 등을 정하게 된다. 개념설계, 시스템 분석, 상세설계 단계는 앞의 단계에서 정한 제약조건하에서 최선의 방안을 찾는 일이기 때문에 여의치 않으면 앞의 단계로 정보를 feedback 시키는 일도 발생하게 된다.

상세설계로 기계의 사양, 성능, 제품의 흐름방법 등이 결정되어 설계도가 완성되면 이들을 효율적으로 운용하는 방안이 강구된다. 즉, 스케줄링 방법, 작업차레 정하는 방법(sequencing), 공구(tool)

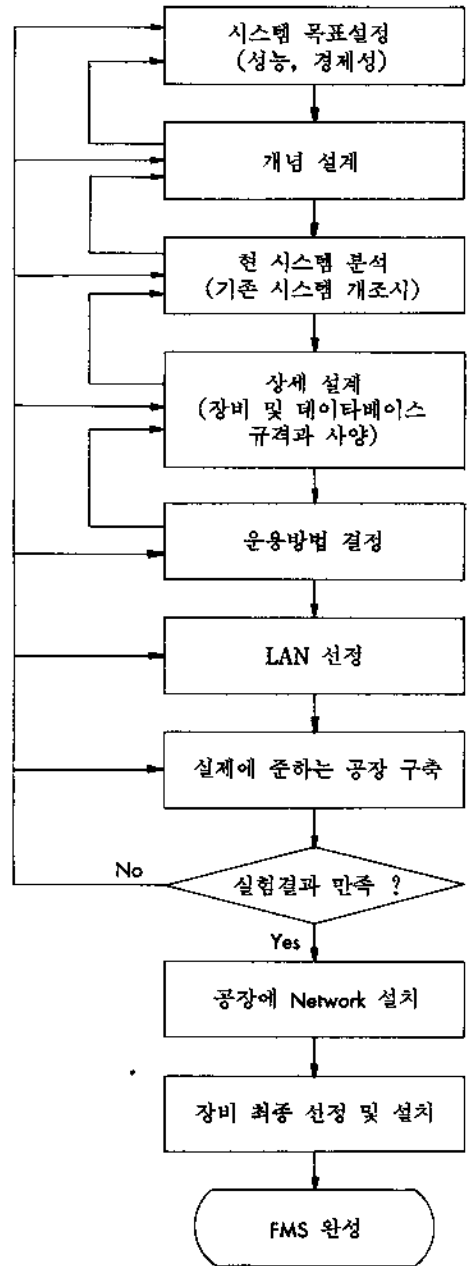


그림 1. FMS 구축절차

와 NC 프로그램의 구체적인 처리방법, 데이터베이스 설계 등이 여기서 결정되고 이러한 정보를 처리할 수 있는 구체적인 LAN의 도입이 다음 단

계로 이루어진다.

운용방안이 결정되면 본 연구에서 제시한 LAN과 컴퓨터를 이용한 공장모형의 구축단계이다. 여기서는 앞 단계에서 정하여진 운용방법, 데이터베이스 처리, LAN상의 통신 등에 관한 프로그램이 완성되며 장비의 동작은 정하여진 사양을 참조하여 시뮬레이션으로 처리된다.

다음 단계는 컴퓨터로 구성된 공장을 실제로 운영하는 것이다. 즉, 상세설계의 내용을 실험하는 단계이며, 실험결과가 만족스럽지 않을 때는 해당되는 앞 단계로 돌아가서 그 부분을 고치게 된다. 이러한 실험과 교정의 반복을 통하여 컴퓨터로 구성된 공장에서 만족할만한 공장 운영이 이루어지면 공장에 실제 네트워크를 구축하고 서론에서 언급한 것처럼 자동화 공장의 실제화(realization)를 이루게 된다.

3. 실제에 준하는 FMS 구축절차

본 연구에서 제안하는 FMS 모형은 상세설계를 완성하고, 운용방법과 LAN을 결정한 후에 구축해야 한다.

운용방법을 결정할 때는 스케줄링, 재고정책 결정, 작업물 투입순서 결정(sequencing) 등의 FMS 운용계획(FMS planning)[3]과 NC 프로그램 처리방법, 공구 처리방법, 팔렛 교환 및 처리방법, 데이터베이스 운영방법 등의 구체적인 운용방법을 결정하여야 한다.

운용방법이 결정되면 상세설계를 바탕으로 FMS에 도입될 기계장비의 동작과 성능을 분석하여 FMS 모형을 구축할 때 시뮬레이션용 자료로 사용한다.

장비의 동작분석이 완료되면 상세설계를 이용하여 공장의 컴퓨터 배치를 완성하고 각 컴퓨터의 역할을 결정한다. 컴퓨터의 역할은 기계장비의 임무와 구성요소의 규모에 따라서 통제단위를 설정하고, 각 통제단위에 컴퓨터를 배정하므로써 결정

된다. 기계장비의 임무는 자재운반, 저장, 기계가공, 부품조립, 검사 등으로 나눌 수 있으며, 구체적인 임무는 부품의 흐름, 가공종류 등의 고려사항에 따라서 나뉘어진다. 그리고, 기계장비 구성요소의 규모는 throughput, 부품흐름, 작업조건 등의 고려 요소에 의하여 결정된다. 이러한 과정을 거쳐 통제단위가 결정되면 각 통제단위에게 컴퓨터의 임무를 할당한다. 그리고, 통제단위에 임무가 부여된 컴퓨터들의 작업을 통제하고 상위수준의 컴퓨터의 명령을 처리하는 공장 통제용 컴퓨터를 따로 둔다.

컴퓨터의 결정과 역할 할당작업이 완료되면 LAN을 이용하여 컴퓨터들을 결합한다. 그리고, 다음 작업으로 데이터베이스 시스템을 선정하고 구축한다. 데이터베이스를 프로그램보다 먼저 구축하는 이유는 프로그램이 데이터베이스 시스템에 종속되며, 자료의 종류와 형태가 정립되어야만 프로그램을 구축할 수가 있기 때문이다. 데이터베이스를 구축하기 위해서는 시스템에서 생성되는 자료의 형태와 자료의 흐름을 분석해야 한다. 자료의 흐름은 컴퓨터 내부에서의 자료 흐름과 LAN 통신을 이용한 통신자료 흐름으로 구분되며 이들은 각 컴퓨터의 데이터베이스 설계에 매우 중요한 작업이다. 자료의 흐름을 분석하는 방법에는 Ross와 Schoman이 개발한 SADT(Structured Analysis and Design Technique)[7]과 De Marco가 개발한 자료흐름도(Data Flow Diagram)[1] 등의 방법을 사용한다.

마지막 단계에서 FMS 운용 소프트웨어를 구축한다. FMS 운용소프트웨어는 공장 통제용(Shop Floor Control) 프로그램과 장비통제용 프로그램으로 나뉘며, 장비통제용 프로그램에서는 장비의 동작을 이산형 시뮬레이션으로 표현한다. 운용소프트웨어의 구성은 시간 동기화처리 프로그램과 기계장비 동작 시뮬레이션 프로그램을 제외하면 실제 시스템의 운용프로그램과 동일하다. FMS 운용소프트웨어는 실시간(real time) 작업을 하므로 프

로그래밍은 실시간 처리가 가능하도록 구성되어야 한다. FMS 운용소프트웨어의 구성은 다음 장에서 설명하겠지만 FMS 운용 프로그램, 자료처리 프로그램, 생산관리 프로그램, LAN 통신 프로그램, 기계장비 통신 프로그램 등의 실제로 FMS에서 사용할 프로그램들과 FMS 모형의 운영에 필요한 시간 동기화처리 프로그램과 기계장비 동작 시뮬레이션 프로그램으로 구성된다.

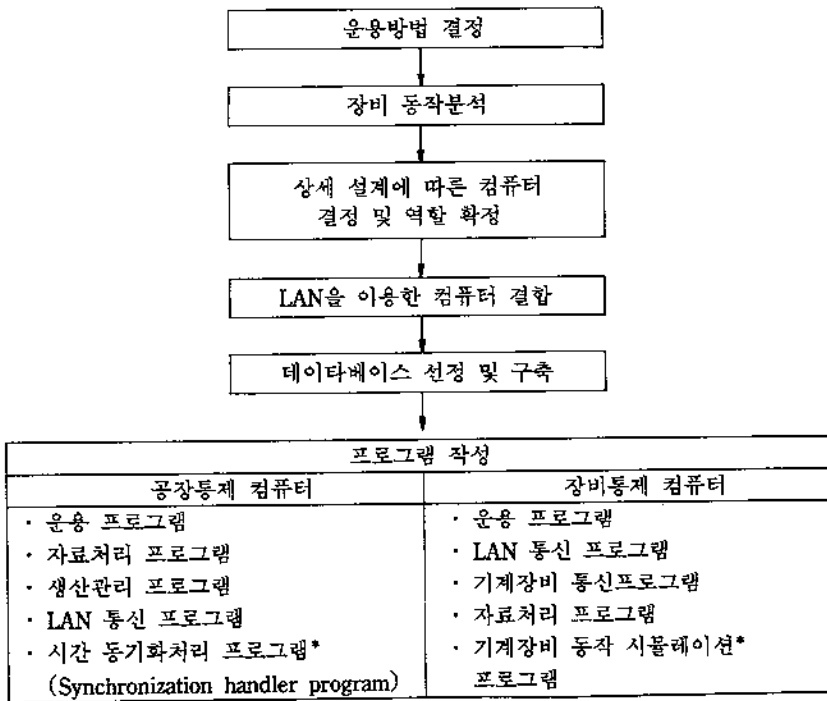
지금까지 설명한 시스템 구축 절차를 요약하면 <그림 2>와 같다.

4. 시간 동기화 방법

본 FMS 모형은 여러 컴퓨터가 LAN에 연결되어진 분산형 시뮬레이션 모형이고, 각 컴퓨터에는 운용프로그램을 처리하는 실제시간(real time)과

이산형 시뮬레이션으로 처리되는 물리적 사건의 시뮬레이션 시간이 혼합되어 있다. 즉, 컴퓨터의 운용 소프트웨어가 작업을 하는 실제 활동은 실제 시계에 따라 시간을 소비하면서 진행되며, 실제로 물리적 시간을 소비하며 움직이는 장비의 동작시간은 시뮬레이션의 사건발생 방법에 의하여 짧은 시간에 발생(generation)되므로, 시간의 흐름이 실제 시계로 볼때 압축되어 있는 상태가 된다. 따라서, 시간발생 순서는 실제와 다르게 되는데 실제 순서를 지키지 않으면 시스템이 운영되지 않는다.

이러한 이유로 FMS 모형의 시간을 동기화시키는 도구가 필요하다. 본 연구에서 개발하여 사용하는 도구는 시스템 전체의 시간을 관장하는 독립된 시간 동기화 처리 프로세스(Synchronization handler process)와 모든 컴퓨터가 이 프로세스에게서 받은 시간을 이용하여 자신의 시간을 맞추는 시간 동기화



* 표시 부분은 FMS 완성시 제거되는 부분

그림 2. 실제에 준하는 FMS 구축 절차

모들이다. 시스템에 속한 통제 컴퓨터들이 시간정보를 전달하면 시간 동기화처리 프로세스는 전달된 시간정보를 이용하여 표준화된 현재 시간을 발생시켜 이 시간을 전달한다. 그러면 각 통제 컴퓨터는 전달된 현재 시간에 자신의 시간을 맞추는 것이다.

시간 동기화처리 프로세스의 동기화 구조는 다음과 같다.

① 현재 전체 컴퓨터내에 실제시간을 소비하는 프로그램이 작동되고 있지 않으면 시물레이션의 다음 사건 시간으로 시계는 옮겨간다.

② 컴퓨터 중에서 하나 이상 실제 시간을 소비하는 작업을 하고 있을 경우에 시계는 실제시간과 같이 전진하며, 주어진 시간간격마다 시물레이션에 의하여 발생한 다음 사건 시간이 그 간격내에 있는지를 판단해서 있으면 그 사건이 발생한 것으로 한다.

③ 시물레이션상의 다음 사건으로 옮겨갈 수 있는 경우라도 하나 이상의 컴퓨터가 현재 사건발생 중이면 시간의 전진을 멈추고 시물레이션 시간이 도착할 때까지 기다린다. 도착한 시물레이션 시간이 기존의 시물레이션 시간들 보다 전에 발생한 사건들이면 그 시간으로 시계를 전진시키고, 아니면 기존의 시물레이션 시간중 가장 빠른 시간으로 시계를 전진시킨다.

5. 운용 소프트웨어의 구조

실제에 준하는 FMS에서는 각 컴퓨터에서 작동되는 운용 소프트웨어는 단일한 프로그램이 아닌 여러개의 프로그램이 필요하며 동시에 여러 프로세스가 동시에 움직이게 된다. 프로그램의 구성은 다음과 같다.

① 시간 동기화처리 프로그램

물리적 사건의 시물레이션 시간과 실제 컴퓨터의 운용 프로그램의 처리시간(real time)을 같은 시간 스케일에서 동기화시켜 전진시키는 프로그램으로 공장통제 컴퓨터에서 작동한다.

② 운용 프로그램(Main Program)

공장통제 컴퓨터와 장비통제용 컴퓨터의 운영을 관리하는 프로그램이다. 공장통제 컴퓨터의 운용 프로그램은 주문(order) 처리와 자재흐름을 통제하는 프로그램으로 상위 컴퓨터와 장비통제용 컴퓨터의 입력을 받아서 작동된다. 장비통제용 컴퓨터의 운용 프로그램은 공장통제 컴퓨터의 명령 또는 다른 컴퓨터의 정보를 받아 자신의 장비 또는 작업장(Cell)을 통제한다.

③ 통신 프로그램

통신은 LAN을 이용한 컴퓨터간의 통신과 컴퓨터와 기계장비간의 통신이 있다. 컴퓨터와 기계장비간의 통신은 장비통제용 컴퓨터에서 발생한다. 그러므로, 공장통제 컴퓨터는 LAN 통신용 프로그램을 구축해야 하고, 장비통제용 컴퓨터에서는 LAN 통신용 프로그램과 기계장비 통신용 프로그램을 함께 구축해야 한다.

④ 데이터베이스 처리 프로그램

운용 프로그램 또는 다른 컴퓨터의 통신입력에 의하여 이틀로부터 오는 데이터를 받아서 수정하거나 요구에 따라서 데이터를 제공하는 프로그램이다.

⑤ 데이터베이스 시스템

데이터베이스를 구축하고 데이터베이스 처리 프로그램의 요구를 수행하기 위해 상세설계 단계에서 결정한 데이터베이스 시스템을 각 컴퓨터에 구축한다.

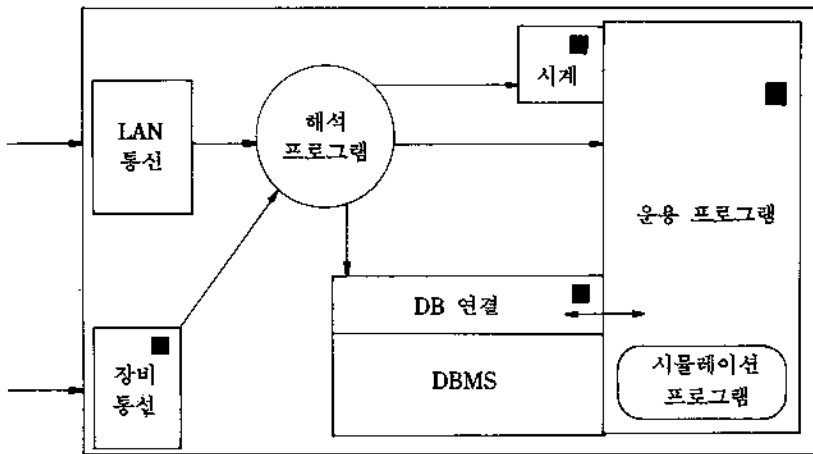
⑥ 해석 프로그램

외부에서 통신프로그램을 통하여 입력된 정보를 해석하여 해당 프로그램에 해야할 일을 전해주는 프로그램이다.

⑦ 장비동작 모의 프로그램

장비통제용 컴퓨터에 존재하며 장비의 동작을 묘사한다. 이 프로그램은 독립적으로 존재하는 것이 아니라 운용 프로그램에 포함되어 있으며 실제로 장비를 갖추게 되면 필요가 없어진다.

앞에서 언급한 프로그램의 관계는 <그림 3>에



■ : 자체보유 정보를 외부로 보내는 통신 프로세스 표시

그림 3. 프로그램 관계도

표현되어 있다. LAN을 통하여 들어온 통신자료는 해석 프로그램으로 보내어진다. 해석 프로그램은 통신자료를 해석하여 운영 프로그램이나 데이터베이스 처리 프로그램 또는 공장 통제컴퓨터의 경우 시간에 해당하는 정보를 시간 동기화 프로그램에게 자료를 전달한다. 각 해당 프로그램은 명령에 따른 프로그램을 수행하고 자체의 통신시스템을 이용하여 외부의 컴퓨터나 장비에게 명령을 보내거나 결과를 통보한다. 장비통제용 컴퓨터의 경우 장비가 작업을 완료하거나 자료를 요청할 경우 기계장비 통신 포트를 통하여 통신자료를 보내며, 이 통신자료는 기계장비 통신용 프로그램을 거쳐 해석 프로그램으로 전달된다.

이들 다수의 프로그램은 동시에 독립된 프로세스로 구동이 되어야 하며, 각 프로세스간에는 대화(Inter-Process Communication)가 가능해야 한다. 이러한 문제는 멀티타스킹과 내부통신이 가능한 O.S를 사용하는 것이 해결하기 쉬운데 이에 알맞는 대표적인 O.S는 UNIX 시스템[8]이다.

현재 많이 쓰이는 방법은 인터럽트를 사용하여 시간을 나누어서 프로그램을 순차적으로 작동시키

는 방법이지만, 이는 간단한 시스템인 경우에만 가능하고 내부통신 기능을 수행하기에는 매우 복잡하므로 범용성있는 소프트웨어는 이러한 방법으로 작성하기에는 현실적으로 불가능하다. 따라서, FMS의 기능을 제대로 발휘하려면 멀티타스킹과 내부통신이 가능한 O.S가 준비되어야 한다.

본 연구가 제안하는 FMS 모형의 컴퓨터와 장비의 구성과 이들의 배치는 구축하고자 하는 실제 시스템과 같다. 예를 들어서 <그림 4>를 보자. <그림 4>는 일반적인 FMS의 계층적 구성을 나타내고 있다. 이 그림의 FMS 구성을 본 연구의 FMS 모형으로 전환하면 <그림 5>와 같다. 그리고, <그림 5>에서 보듯이 시간 동기화 프로그램과 최하부 단위의 장비의 운영을 표현한 시뮬레이션 모형을 제외한 나머지는 실제 시스템으로 구성되며, FMS 모형의 컴퓨터와 장비의 구성과 배치는 구축하고자 하는 실제 시스템과 같다.

FMS는 컴퓨터를 이용하여 공장을 통제하는 시스템이므로 컴퓨터 프로그램을 본 연구에서 제안하는 모형의 형태로 구축할 수 있으며, 따라서 모든 형태의 구축하고자 하는 FMS는 이 FMS 모형을

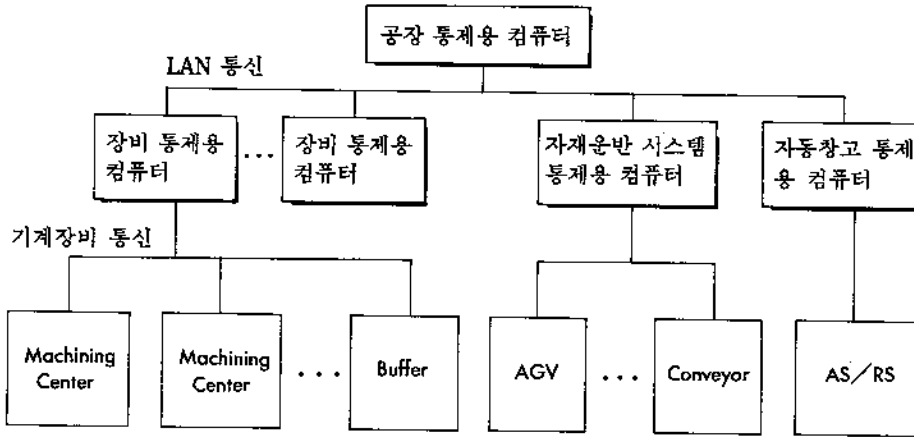
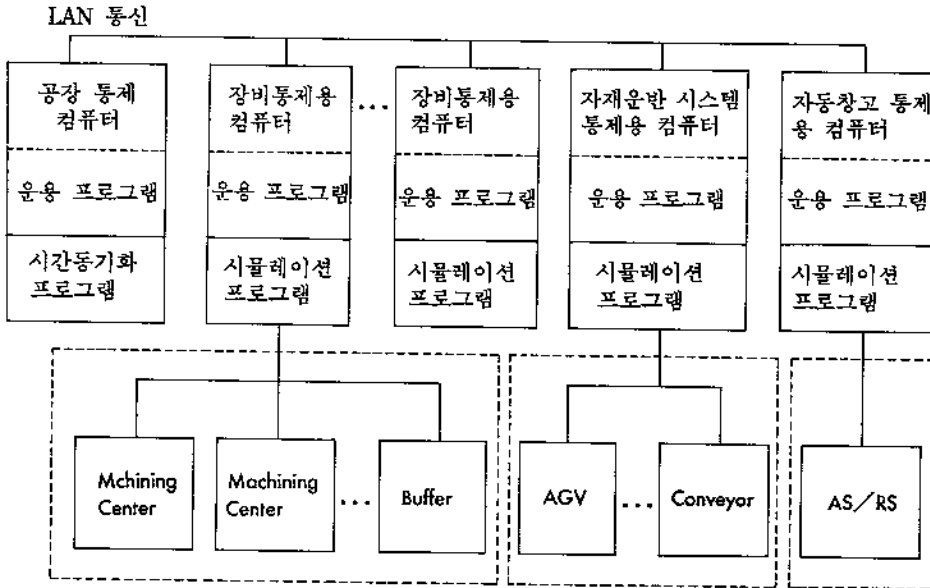


그림 4. FMS의 컴퓨터 계층도



※ 점선 내부는 시뮬레이션의 대상이 되는 장비를 표시

그림 5. 제안된 시스템의 모형 구성도

이용하여 표현이 가능하다.

6. 결 론

본 연구에서 제시하는 FMS를 구축하는 방법과

여기에 사용되는 도구로서 실제와 시뮬레이션이 혼합된 시스템의 유용성은 상식적으로 이해가 되는 방법이다. 필자들이 속한 연구팀에서는 현재 이 방법을 사용하여 FMS를 구축하고 있다. 1년 가량의 시간이 지난 현재 네트워크상에서 운용프로그램이

거의 완성되고 기계 도입도 일부 이루어졌다. 이는 본 연구가 제안한 방법을 사용하지 않고는 단시간 내에 이같은 성과를 거두기는 힘들었을 것으로 생각된다. 즉, 필자들은 실제 시스템 구축을 통하여 본 연구의 제안을 수행하고 있고 상당한 효과를 거두고 있는 중이다. 본 연구가 제안하는 방법으로 시스템을 구성할 때 각 소프트웨어들은 FMS의 형태가 변하여도 모듈 몇가지만 바꾸면 바로 사용할 수 있도록 필히 소프트웨어를 모듈화하여 범용성을 갖도록 하는 문제는 중요한 것으로 생각된다. 앞으로 표준 소프트웨어 모듈이 개발되어 FMS의 형태에 따라 모듈조합만 바꾸면 즉시 쓸 수 있는 시스템이 개발되어야 할 것으로 생각된다.

참고문헌

[1] De Marco, T., *Structured Analysis and System Specification*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1979.
 [2] Groover, M.P., *Automation, Production Systems, and Computer Integrated Manufacturing*,

Prentice-Hall, Inc., Englewood Cliffs, N.J., 1987.

[3] Kalkunte, M.V., Sarin, S.C. and Wilhelm, W.E., "Flexible Manufacturing Systems: A Review of Modeling Approaches for Design, Justification and Operation", *Flexible Manufacturing Systems: Methods and Studies, Studies in Management Science and Systems*, Vol. 12, 1986.

[4] Law, A.M. and Kelton, W.D., *Simulation Modelling and Analysis*, McGraw-Hill, Inc., 1982.

[5] Nigel, R.G., *Implementing Flexible Manufacturing Systems*, Macmillan Education LTD, 1988.

[6] Pressman, R.S., *Software Engineering: a Practitioner's Approach*, McGraw-Hill, Inc., 1982.

[7] Stevens, W.R., *UNIX Network Programming*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1990.

[8] Vail, P.S., *Computer Integrated Manufacturing*, PWS-KENT Publishing Co., 1988.