UNIX SYSTEM을 이용한 종합제어 장치

박 성식, 신 용길, 노 태석, 조 규복

Application of UNIX System to Automatic Control

Sungsik Park, Youngkil Shin, Taesuk Rho, Kyubok Cho
Hyosung Industries Co., LTD.

ABSTRACT

This paper deals with the topics when Unix based computer is applied to automatic manufacturing system. The total control of material flow in the automatic storage and retrieval system is taken as an example.
And some technical issues are proposed for the wider application of UNIX to automatic control.

1. 서 론

computer를 이용하여 제조 공정에서 하위 기기를 종합제어 하는 system을 구현할 때 multi-processing을 염두에 두어야 하는 것은 말할 나위도 없다.
즉 어느 한 순간에 computer는 동등 또는 상위 level의 다른 computer와 data를 주고 받아야 하며 화면과 keyboard를 통해서 사용자와 대화해야만 하며 연결된 하위 기기를 끊임없이 제어, 감시해야만 한다. 이러한 일련의 사건과 현상은 정해진 순서대로 일어나는 것이 아니라 서로 독립적으로 무작위하게 일어나며 computer는 이 모든 것을 인간이 의식하지 못할정도의 빠르기로 동시에 처리하는 것 처럼 보여야 한다.
UNIX는 이와 같은 용도의 처리에 적합하면서도 업가의 microcomputer및 workstation이 계속 UNIX를 채택하고 있어 현재나 향후 가장 널리 보급되는 O/S이다. 본고는 자동창고내 물류 기기를 제어하는 computer의 O/S로 UNIX를 채택한 경우 그 내부 구성을 예로 들어 다른 유사한 분야의 단서를 제공하고자 한다.

2. multiprocessing O/S의 선택

복잡하고 다양한 system을 제어할 때 각각의 역할에 맞는 process들과 각 process들을 유기적으로 통합 조정하는 process를 상정하는 것은 지극히 자연스런운 발상이다.
정해진 제약 조건에서 보다 탈피하여 자연스럽게 설계되어진 system이 성능면에서나 향후 보전면에서 더 뛰어난 system임은 두말 할 나위도 없다.multiprocessing system을 구현할 때 각 process끼리 서로 동기를 맞추어 data교환을 할 수만 있다면 각 process내부 설계는 다른 process를 의식하지 않고 완전히 독립적으로 행해질 수 있다. 그러나 이와 같은 구상이 MS-DOS와 같은 single task O/S를 선택하면 640K이내의 적은 memory만 사용할 수 밖에 없어 큰 규모의 프로그램을 작성할 수 없다는 사실은 간과하더라도 개발자의 작위적인 프로그래밍으로만 구현될 수 밖에 없음으로 그 개발 과정및 test가 복잡해지고 길어진다. 그러나 UNIX를 채택하면 O/S상에서 그 지원이 가능함으로 한결 부담을 덜 수 있다. 이 경우 개발자는 각 process간의 data교환 및 형식만을 정의하면 된다.
특히 system v 계열의 UNIX는 semaphore (1)및 공유 기억장치등 process간 통신 기능이 탁월하다.(2)
UNIX이외에 microcomputer용으로 O/S-2와 MS-WINDOW등도 multiprocessing기능이 지원되고 특히 MS-WINDOW의 경우 향후 MS-DOS의 뒤를 이어 PC O/S의 주류를 이룰 전망이지만 그 역사가 짧은 탓으로 아직 UNIX만큼 호응을 얻지 못하고 있다. UNIX system의 경우 역사적으로 잘 정비된 documentation등으로 프로그래머에게 충분한 정보를 제공하고 있다.그러나 개발자 개인이나 집단이 어떤 종류의 O/S를 선택하느냐 여부는 O/S의 기술적 요소이외에 그 개인이나 집단이 과거

부터 얼마나 어떤 O/S에 친숙해 있느냐 하는 특수한 조건에 좌우되는 경우가 많다. O/S-2의 경우 최근 한글화가 마무리 되어있고 무엇보다도 DOS의 풍부한 S/W 자산을 계승할 수있다는 장점이 있다.

3.C 언어의 선택 (3)

UNIX상에서 제어 system을 구현하고자 할 때는 특히 C 언어로 개발되는 것이 바람직하다. 그 이유는 첫째로 원래 UNIX의 kernel이 대부분 C 언어로 작성되었으며 process간의 통신및 외부 기기와의 통신등은 kernel의 기능을 이용해야 한다. C 언어로 작성된 프로그램은 kernel의 기능을 system call의 형태로 자연스럽게 이용할 수 있다.

두 번째로 제어 system의 구현은 일반 data processing과 달리 file data의 흐름을 처리하는 것이 아니라 명확히 규정된 상태의 천이를 처리한다. 이 점에 있어서 C 언어는 그 특성상 COBOL과 같은 다른 언어에 비해 설계에 있어 유리하다. 사용하는 언어가 설계에 큰 영향을 미치기 때문이다. 그외에 C 언어의 구조화된 특성으로 말마암아 설계및 test가 용이하다는 장점도 있다.

4. 구성예

그림 1.은 상위로는 재고관리 Computer와 연결 되어 있고 사용자를 위한 terminal및 keyboard가 부착되어 있으며 하위로는 6대의 자동창고 적재용 stacker crane, 1대의 입고 conveyor 제어를 위한 PLC, 다른 1대의 출고 conveyor 제어를 위한 PLC가 연결되어 자동창고 내의 물류 흐름을 제어하는 system이다. 자동창고의 규모는 10,000 cell이 조금 넘는 규모이다.(4)

사용자는 도움 그림, 도움말 등을 이용하여 pull-down menu로 화면을 선택할 수 있으며 모든 기기의 상태및 작업 현황을 알 수 있다. 그리고 자체적으로 작업지령을 내릴 수 있다.

real-time성이 보장되지 않는 일반 UNIX system v 계열의 32 bit microcomputer에서 6개의 관련 process가 동시에 진행되지만 상위 재고 관리 computer와의 작업지시 및 작업결과 data 수수, 하위 기기의 동작 지령및 상태의 화면 표시등이 변화가 일어면 바로 즉시 이루어 지고 있다. 이는 응답 시간이 critical한 요소로 문제가 되지 않는 한 일반 범용 UNIX라도 공장 자동화 분야의 종합제어 system에 응용 가능하다는 것을 보여주고 있다.

그림 2.는 이와 같은 system의 내부 프로그램 구성을 설명한 것이다.

- 상위 재고관리 computer와의 통신을 위한 process
- 화면 처리를 위한 process
- 입고 conveyor 제어를 위한 process
- 출고 conveyor 제어를 위한 process
- 하위 기기 제어를 위한 process를 종합제어 하는 process

등이 동시에 진행된다. 각각의 process들 끼리의 통신은 semaphore및 공유 memory를 이용하여 행해진다. 향후에 하위 기기의 error이력을 분석하여 설비의 진단및 예측을 행하는 process도 추가될 예정이다. UNIX같은 multiprocessing을 지원하는 O/S는 그 용량이 허용하는 범위 내에서 새로운 process의 부가가 용이하여 기존의 프로그램 변경없이 향후 새로운 기능의 추가등 여러가지 가능성을 확신할 수 있게 한다.

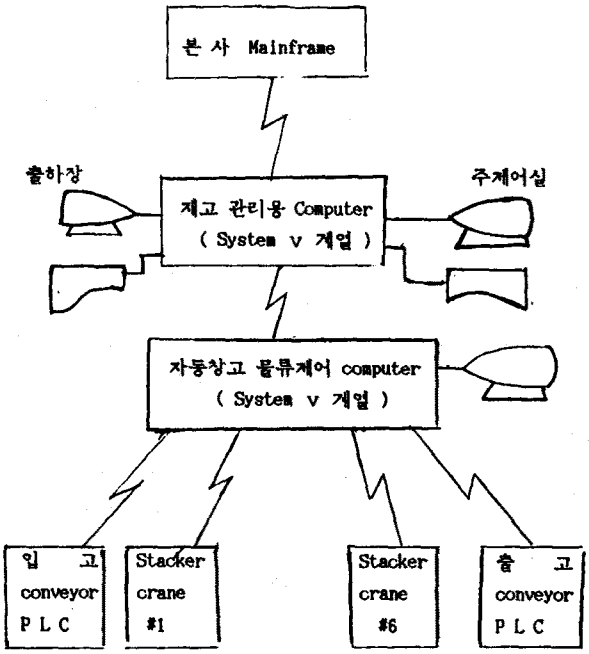
5. 결론

공장 자동화 분야의 제어 system에 UNIX를 보다 광범위하게 적용하고자 한다면 다음과 같은 기술적 요소가 고려 되어야 한다.

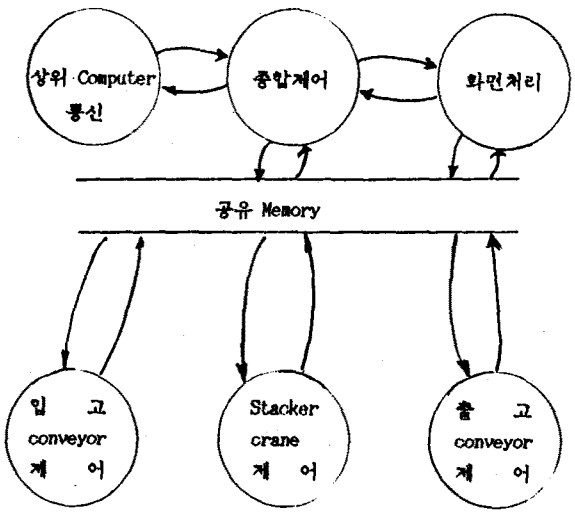
1) real-time UNIX의 채택
   고속의 data 수집및 복잡한 계산을 정해진 시간이내에 해야만 할 때는 일반 UNIX에 real-time성을 보강한 O/S가 선택되어야 한다
2) 각종 device driver개발
   현재 일반 범용 computer는 asynchronous 통신board, 프린터 , LAN ,disk controller board 등이 표준으로 장착되어 있으나 공장 자동화 분야에 널리 쓰이는 PIO board, image processing board, analog data 수집 board등 여러 기능 board에 대한 device driver 기술이 개발되어 보급되어야 한다.

6. 참고 문헌

1) DiJkstra,E.W. "Cooperating Sequential Processes" in Programming Languages, ed.F.Genuys,Academic Press,New York, NY, 1968.
2) Bell Laboratories "UNIX Programmers Manual" Volume 2.
3) Kernighan,B.W. and Ritchie,D.M. "The C Programming Language" Prentice-Hall, 1978.
4) 효성중공업(주) "자동화 창고 System" 운영설명서(PP)", 1990.


본 사 Mainframe
출하장
재고 관리용 Computer
(System v 계열)
주제어실
자동창고 물류제어 computer
(System v 계열)
입 고
conveyor
P L C
Stacker
crane
#1
Stacker
crane
#6
출 고
conveyor
P L C

그 림 1.


상위 Computer
통신
종합제어
화면처리
공유 Memory
입 고
conveyor
제 어
Stacker
crane
제 어
출 고
conveyor
제 어

그 림 2.