

# NC용 Graphical OS 설계

°최 성락, 정 광조  
한국 기계연구원 자동제어실

## Design of a Graphical Operating System for the NC Controller

°Choy Sung Lark, Chung Gwang Jo  
Korea Institute of Machinery and Metals, Automatic Control Lab.

### Abstract

In this paper, an operating system is designed for NC Controllers. Host computer is the IBM-PC, and the controller panel is designed in graphics mode to look easily for users. Pull-down menu is selected, which is useful to minimize a mistake made by users, so users can use NC controller more effectively. Also, many functions are proposed, such as NC editor for programming the NC program directly, simulation function for testing edited NC program, and new key-pad that appropriates for the functions of OS.

기본 구조는 그림 1과 같으며, 주 메뉴에서 각각의 부 메뉴로 들어갈 때에는 pull-down menu 방식으로 이루어지며, 사용자는 메뉴를 선택할 때 원하는 곳으로 화살표 키를 사용해서 이동한 후 CR (Carriage Return) 키로써 선택을 하면 된다. 따라서, 이 방식을 사용하게 되면 사용자에 의한 오입력(誤入力)이 최소로 줄어들게 된다. 그리고 이 방식은 최근 PC에서 많이 이용되는 Windows나 OS/2, Turbo C 등 유명한 S/W들에 사용되는 방식이다. 이 menu 방식은 주 메뉴나 부 메뉴 등 각 메뉴들을 새로이 추가하거나 삭제할 때 매우 간편함을 제공하며, 사용자가 직접 새로이 추가하거나 삭제할 수도 있다. OS의 초기 화면은 그림 2와 같다.

### 1. 서론

기존 NC (Numeric Control) controller에 사용되는 OS (Operating System)은 주로 문자 위주의 system이 대부분으로써, NC 전용 terminal과 전용 controller를 사용한다. 이는 전용 기기를 사용하기 때문에 사용자가 효율적으로 NC를 사용할 수 있는 장점이 있는 반면에, 개발자의 입장에서 본다면, 주변기기 모두를 새로 디자인 하여야 하기 때문에 개발기간이 그만큼 길어지며, NC장치의 가격도 또한 비싸진다는 단점이 있다. 사용자 입장에서는, 사용법을 모두 알아야만 작동을 시킬 수 있어서, 사용법을 배우기가 그리 쉽지 않다. 따라서, 본 연구에서는 가격이 저렴한 IBM-PC를 H/W로 하고, 그에 맞는 interface와 controller를 개발하며, PC상에서 NC machine을 사용하기 쉽도록 사용환경, 즉 OS (Operating System)를 개발하는데 목표를 둔다. OS를 design하는 것이기 때문에 직접 H/W를 구동하는 부분은 사용자가 직접 작성하여 사용할 수 있도록 한다. 또한 OS에 사용되는 각 menu도 사용자가 정의하여 사용할 수 있도록 유연성을 둔다.

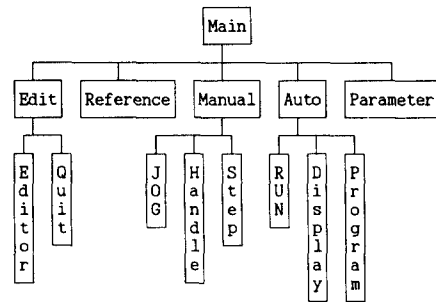


그림 1. OS의 구조

Fig. 1. The Structure of OS

### 2. NC OS의 구조

NC용 OS는 PC상에서 graphic card를 VGA로 사용하며, 화면상의 모든 display는 graphic mode에서 이루어진다.

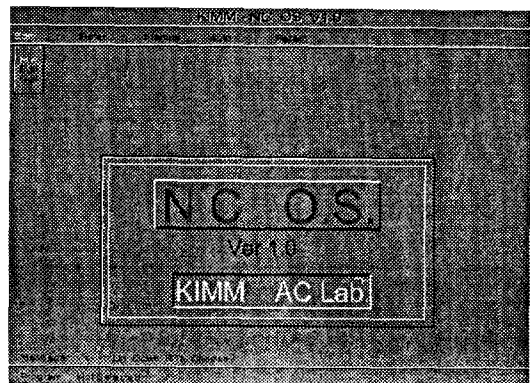


그림 2. OS의 초기 화면

Fig. 2. Initial Screen of OS

- ① EDITOR : Edit mode로 들어간다. 이 mode에는 editor로 가던가 program 실행을 마치게 되는 2가지 부메뉴가 있다.
- ② REFERENCE : Reference return mode로써 MZPR (Machine Zero Point Return)기능과 WZPR(Work Zero Point Return)기능등 2가지 기능이 있다.
- ③ MANUAL : Manual drive mode이다. 이에는 Jog mode, Handle mode, Step mode등이 있다.
- ④ AUTO : Automatic drive mode이다. Program select, Display, RUN등의 부 메뉴가 있고 이중 RUN에는 다시 RUN과 Simulation기능으로 나뉜다.
- ⑤ PARAM : Parameter set mode이다. OS 전체에서 쓰이고, 사용자가 정의해서 사용할 수 있는 매개변수를 정의하고 그 값을 할당할 수 있다.

### 3. NC 전용 Key-Pad

Host를 IBM-PC로 사용하였으나, PC에서 사용하는 keyboard는 NC전용으로 사용하기에는 적합치가 않다. 그 이유는 우선 각 key가 너무 작고, key사이사이로 먼지등 이물질이 들어가기 쉬워서 내구성의 문제가 있다. 또한 NC의 기능이 한 눈에 들어오지 않고 각 기능에 할당된 key를 사용자가 외워서 사용해야 한다는 불편함이 있다. 따라서, NC에 필요한 모든 기능을 정의할 수 있도록 하였고, 기능의 지정은 S/W로써 수정, 변경할 수가 있어서 여러 기종의 NC에 적용시킬 수 있는 key-pad를 새로이 설계하였다. 그에 대한 그림은 그림 3과 같다. 제안된 key-pad는 기존의 IBM-PC의 keyboard간의 연결 mechanism을 그대로 사용하였고 각 key의 값은 기존의 값과 동일하게 할당 하였으며 각 key의 기능지정은 C언어로 작성된 main control program에서 처리한다. key의 배치는 기존의 상용화된 수종의 NC 장치의 model을 참조하여 Cursor, Edit, Run, Address, Data의 5개의 그룹으로 할당하였다.

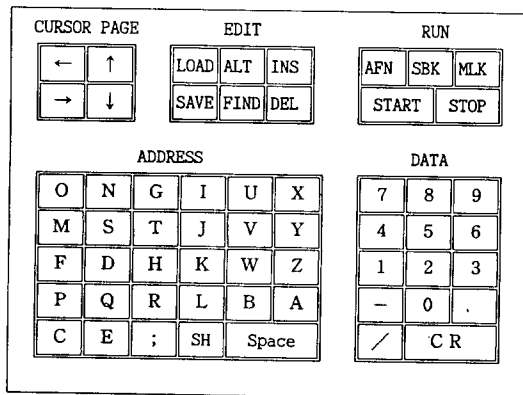


그림 3. OS에 사용되는 Key-pad

Fig. 3. The Key-pad used in OS

## 4. 각 기능의 요약

### (1) Editor

IBM호환기종의 PC에는 이미 여러 종류의 editor가 보급되어 있으나 NC program자체가 갖는 고유의 편집기능을 보강하기 위해서는 NC전용의 편집기능이 필수적이다. 본 연구에서는 앞절의 NC전용 key-pad를 전제로 하여 전용 editor를 실현하였다. 가능한 한개의 키에 한가지의 기능만을 할당하여 사용자의 편의를 높이는 데 초점을 맞추고 사용자의 오류가 빈번할 것으로 예상되거나 여러 기능이 복합된 경우, 2개 이상의 key조합으로 처리하였다. Editor의 main screen은 그림 4와 같다.

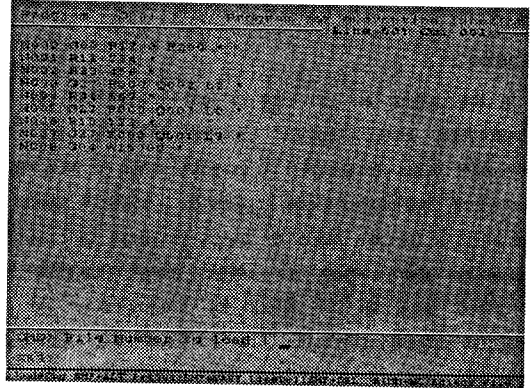


그림 4. Editor의 주 화면

Fig. 4. Main Screen of the Editor

### (2) Editor의 주 기능과 Key 조합

Load : disk에 저장되어 있는 program을 읽는다.

(LOAD Key)

Save : 현재 작성한 program을 disk에 저장한다.

(SAVE Key)

Alternative : 커서위치에 있는 글자를 다른 글자로 바꾼다.

(ALT Key)

Find : 원하는 번호의 라인에 cursor를 위치한다.

(FIND Key)

Insert : Insert/Overwrite mode를 toggle한다.

(INS Key)

Delete : 현 cursor위치의 글자를 지운다.

(DEL Key)

Line Delete : 현 cursor위치의 라인을 지운다.

(SH+DEL Key)

Mode Change : 입력mode와 명령mode사이를 전환

(SH+ALT Key)

New : 현 메모리를 지우고 새로 edit한다.

(Load명령으로)

Rename : 현재 program을 다른 program 번호로 바꾼다.

(Save명령으로)

Print : 현재 program을 printer로 출력  
(SH+Save Key)

(4) Editor의 Buffer

Edit할 buffer는 linked list로 구현하여 memory를 절약할 수 있도록 하였으며, 그에 관한 source는 다음과 같다.

```

struct Edit__Buffer (
  struct Edit__Buffer *H;      /* Points to Head */
  struct Edit__Buffer *T;      /* Points to Tail */
  char buf[80];                /* Edit Buffer */
);

typedef struct Edit__Buffer  BUF;

BUF  *_buffer;  /* 현재 Buffer를 가리킨다. */
BUF  *start;    /* 처음 Buffer를 가리킨다. */
BUF  *last;     /* 맨끝 Buffer를 가리킨다. */

```

Editor의 여러 가지 기능중 가장 중요한 기능은 return key를 눌렀을 때의 buffer처리 기능이다. Return 기능의 flow chart는 그림 5와 같다.

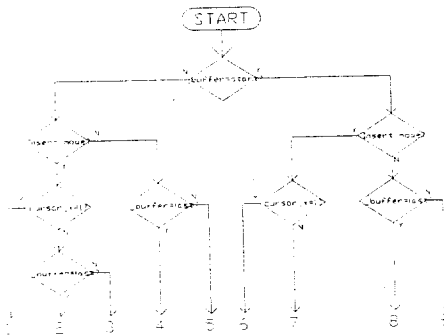


그림 5. Return기능의 Flow Chart  
Fig. 5. The Flow Chart of Return Function

그림 5에서 맨 밑의 번호는 각 case를 나타내며, 이에따라 처리하는 routine이 각각 다르며 또한 각 case마다 scroll을 처리해야하는지 판단해야 한다.

(2) Reference Return기능

MZPR(Machine Zero Point Return)은 기계 시스템의 절대 원점을 RLS(Reference Limit Switch)에 의해 설정하는 기능으로써, 시스템의 초기화 과정에서 수행된다. WZPR(Work Zero Point Return)은 parameter로 지정한 절대 좌표계상의 임의의 점을 새로운 좌표계로 찾아 지정하는 기능으로, 마찬가지로 RLS에 의해 절대 원점을 찾은후 새로운 원점으로 이동하고 그 지점을 원점으로 하는 위치값을 설정한다. 특히, 이 기능은 G기능(G90)의 형태로 running중에도 program으로 수행가능하다.

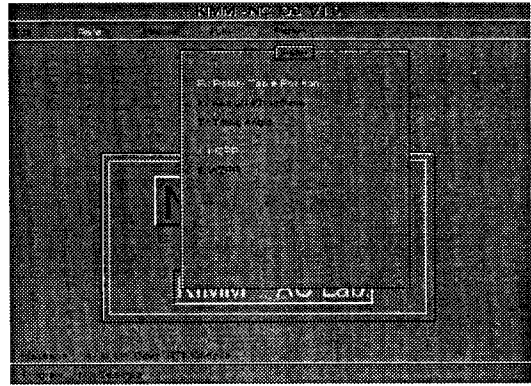


그림 6. Reference Return Mode의 화면  
Fig. 6. Reference Return Mode

(3) Manual Axes Drive 기능

수동축 이동기능은 table이나 tool의 H/W적 위치를 operator가 임의로 설정할 수 있도록 하는 기능으로 빠른 시간 내에 정확한 위치로의 이동을 위해 3가지 기능이 준비되어 있다. Jog Mode는 사용자가 임의의 축을 가장 빠른 시간내에 jog key를 사용하여 원하는 위치에 이동시키는 rapid traverse 기능으로 이후의 정확한 위치에서의 setting은 다음의 handle과 step기능으로 수행한다. 우선 축을 지정하고 FDR(Feed Rate)과 회전방향(CW, CCW)을 지정한다. 이동하는 동안 position은 화면상에 display되며 이 값은 NC control board에 있는 position controller(KM 3702)의 command register값을 PC가 주기적으로 읽어 표시하는 것이다. Position data값은 0.001 단위의 이송값이 된다. 모든 지정 명령은 key-pad로 한다.

Handle Mode는 Jog와 비슷하나 움직이는 command를 MPG(Manual Pulse Generator)를 이용한다. 그리고 1 step당 움직일 수 있는 거리를 0.001, 0.002, 0.004 (mm or deg/step) 등으로 조정해 줄 수 있어서, 사용자가 원하는 것을 선택한 후 MPG를 사용해서 이송시키는 mode이다.

Step Mode는 jog mode나 handle mode와 같이 동작원리나 화면표시가 비슷하다. Key-pad로 부터 특정입력을 받아 지정된 step만큼 이송을 행하는 것으로 축을 선택하고 한 key를 누를 때마다 이송될 거리양을 선택하여 이송을 행한다.

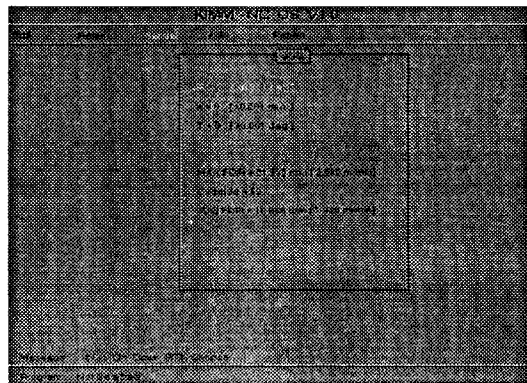


그림 7. Jog Mode

Fig. 7. Jog Mode

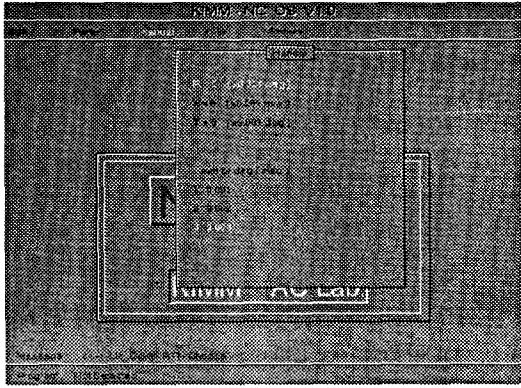


그림 8. Handle Mode

Fig. 8. Handle Mode

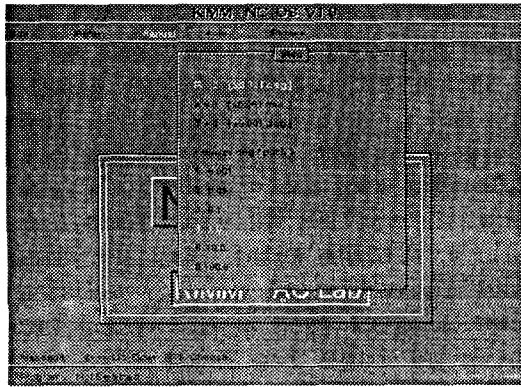


그림 9. Step Mode

Fig. 9. Step Mode

(4) Automatic Axes Drive 기능

① Program Select Mode

이 mode를 선택하면 화면상에 지금까지 저장되어 있는 program들의 번호와 comment들이 나타난다. 사용자는 메뉴선택시와 마찬가지로 화살표 key와 CR key를 사용해서 실행시키길 원하는 program번호를 선택하면 된다. 이때 화살표 키를 움직임에 따라 해당 program은 반전되어 나타난다. 선택시에는 edit시 작성한 comment를 참조하여 선택할 수 있다. 이 program select mode는 RUN mode로 들어가기 전에 반드시 실행해서 실행시킬 program을 선택해야 하며, 이를 무시했을 때에는 RUN mode에서 error message를 display한다. Program의 수가 무척 많아 한 화면에 display되지 못할 경우에는 화살표키를 움직임에 따라 자동으로 프로그램이 scroll되어 화면에 표시된다. 이 mode에 관한 화면은 그림 10과 같다.

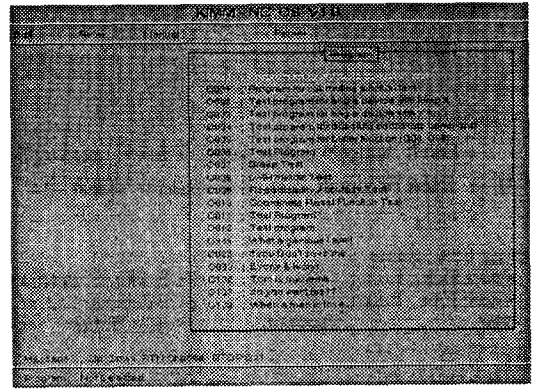


그림 10. 프로그램 선택 mode

Fig. 10. Program Select Mode

② RUN Mode

Run Mode에는 다시 2가지 부메뉴가 있는데, 이는 RUN과 Simulation이다. 두가지 mode의 차이점은 단지 실행을 화면상에서 할 것이냐, 아니면 직접 기계를 구동시키며 할 것이냐이다.

이 mode를 사용하려면 우선 Program select mode에서 program을 선택한 후에 행해져야 한다. RUN mode에서는 Display mode에서 선택한 option에 따라 LIST를 선택했다면 화면상에서는 선택한 program의 list들이 나오고, 한 line을 수행할 때마다 그 과정이 표시되어 진다. 즉, 현재 수행하고 있는 라인에 반전되어 display된다.

Simulation기능은 editor등을 이용해서 작성한 NC language를 직접 NC를 구동해 보지 않고도 눈으로 작업을 확인할 수 있는 기능으로써, 일반적으로 NC는 3차원의 가공물을 다루므로, 3차원의 image를 표현할 수 있어야 한다. 또한 실제 눈으로 보는 것처럼 3차원으로 표현된 물체에 대한 view point를 바꾸어 가며 볼 수 있으면 더욱 편리한 기능이 될 것이다. 본 연구에서 구현한 simulator는 UNIX의 lex, yacc등의 utility로 문법적인 분석을 마친후, 그것을 각 명령에 따라 그림 - 라인을 그리거나 점을 찍는다 - 을 3차원으로 그려준다. 3차원 parameter는 사용자가 parameter set라는 menu에서 초기 설정할 수도 있다. 이 mode에 관한 화면은 그림 11과 같다.

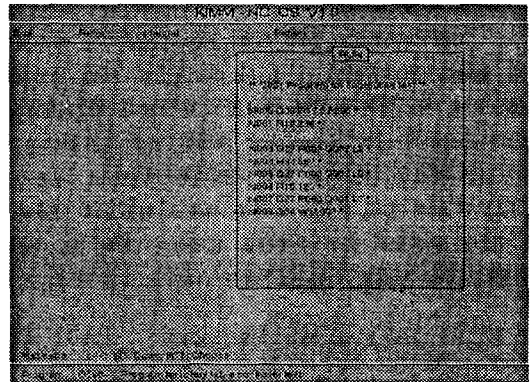


그림 11. RUN Mode

Fig. 11. Run Mode

③ Display Mode

Display mode에서는 RUN mode에서 화면상에 무엇을 display할 것인가를 결정한다. 수행되는 program의 list를 보고 싶다면 그에 해당하는 list가 나오고 현재 수행중인 line은 사용자가 알아보기 쉽도록 반진되어 나타난다. Program list가 한 화면에 모두 display하지 못한다면 자동으로 scroll되어 계속 display를 할 수 있다.

또한 현재 tool의 위치를 display하고자 한다면 각 축에 대한 위치가 보통 글씨보다 크게 확대되어 쉽게 위치를 알아볼 수 있도록 display된다.

(5) Parameter Set 기능

Parameter set mode에서는 NC System에 필요한 각종 parameter의 설정과 변경을 수행하는 mode이다. Parameter에 대한 내용은 parameter를 설명하는 comment내용과 그 값이다. 화살표키로 원하는 위치에 가서 CR키를 누르면 먼저 comment를 입력후 해당하는 값을 입력할 수 있게 되어있다. 이 mode에서는 유일하게 화살표키와 리턴키 이외의 키를 사용할 수 있다. 이 mode에 관한 화면은 그림 12과 같다.

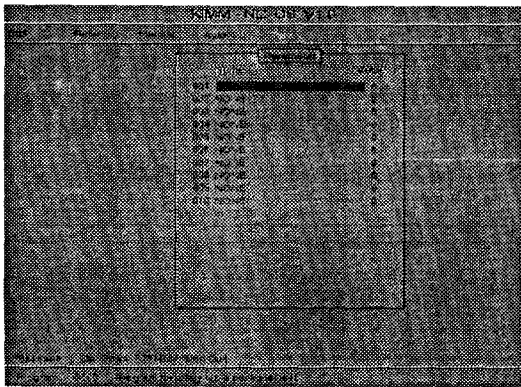


그림 12. Parameter Set Mode

Fig. 12. Parameter Set Mode

5. Hardware의 Prototype

지금까지 설계된 NC OS의 개념을 좀더 구체화 시켜 운용가능한 H/W의 기본 구조를 그림 14가 구성하고 이에 따른 실제적인 prototype을 제작하였다. NC장치는 진동, 소음 및 기타 기계, 전자적 noise요인이 상존 하고 있는 생산현장에서 사용될 가능성이 많기 때문에 NC장치의 설계에 있어서 시스템의 신뢰도 및 안정도를 높이기 위한 구조가 필수적이어서, 다음과 같은 내용을 고려하였다.

① Key-pad를 비틀출식 mask형으로 설계, 제작하여 이물질이 끼어들지 않도록 배려하였고 가능한한 controller를 외부로부터 밀폐시키도록 하였다.

② NC의 machine operating panel기능 (Jog SW, speed override선택 SW, cycle start/stop key등)을 가능하면 별도로 설치하지않고 기존의 key-pad와 monitor를 활용할 수 있도록 S/W로 처리하였다.

③ M, S, T기능등 I/O신호들은 주 제어장치와 별도의 전원을 사용하도록 하였고 signal interface는 최소 2단계 이상의 전기적 절연이 되도록 설계하였다.

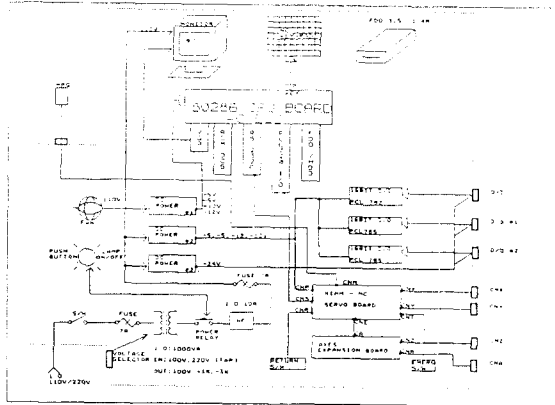


그림 13. NC장치 H/W의 구성

Fig. 13. The H/W of the NC Controller

6. S/W와 H/W간의 Protocol

OS에서 축 이동을 수동 또는 자동으로 수행하는 sequence는 최종의 기능선택화면에서 선택한 키값에 따라 지정된 수행 subroutine을 호출, 수행함으로써 진행되는데, 이때 축 이동 관련 subroutine은 공히 그림 14의 H/W구성도의 KIMM-NC servo board와 IBM-PC간의 RS-232C 연결라인을 통해 필요한 축 제어기능을 위해 정해진 ASCII protocol로 명령 및 응답을 주고 받는다. 즉, KIMM-NC servo board에서는 NC에 필요한 각종 축제어 관련 G기능들이 내장되어 있으며 이를 이용한 각종 응용 Program은 IBM-PC내부의 S/W에서 조합된 기능으로 제작할 수 있다. 또한 축 이동과 관련이 없는 M, S, T등의 기능은 PC BUS상의 별도의 I/O로 각각 8 bit의 BCD code와 stroboe 신호로 NC LF에 전달 된다. 상세한 H/W의 설계내용 설명은 생략한다.

7. 결론 및 향후 연구방향

본 연구에서 개발한 NC장치의 OS는 최대한 사용자가 편리하게 사용할 수 있도록, 그리고 저가격으로 NC장치를 구성할 수 있도록하는것을 목표로 하여 설계되었다. 전용 key-pad를 사용, NC장치를 사용하면서 PC의 장점도 이용할 수 있도록 설계하였고, 사용자는 단지 3개의 키(Enter, F1, F2)만으로 메뉴를 선택할 수 있게하여 실수를 최대한 줄일 수 있도록 하였다. 또한 key-pad와 S/W만으로 구성할 수 있는 것, H/W적인 스위치등은 제거하여 S/W로 구성하였고, 주 제어기는 IBM-PC로 하여 저가격을 실현하였다.

향후 연구방향은 NC장치의 기능도 많아지고 지능적인 기능도 요구되는바, 한번에 여러가지 작업을 수행할 수 있는 환경을 만들어 주는 상용 multi-tasking OS하에서 좀더 강력한 기능을 구현하고자 한다.

## 8. 참고 문헌

1. P.Silberschatz, "Operating System Concepts", 2nd Ed., Addison Wesley, 1985
2. 稻葉清石衛門, "NC 讀本", 5回改訂, 성안당, 1991
3. "Microsoft C/C++ V7.0 Run-Time Library Reference Manual", Microsoft Co., 1992
4. G.Sutty, S.Blair, "Advanced Programmer's guide to the EGA/VGA", Brady, 1988
5. '91년 수치제어(NC) 기술사업, 상공부 보고서, 1991
6. PC용용형 NC제어장치 개발, 과학기술처 보고서, 1989