

Windows CE 기반의 CNC 선반 개발에 관한 연구

임 태완*, 이 철 수**

*씨에스캠, twim@cscam.co.kr

**전남대학교, cslee@cadcam.chonnam.ac.kr

Development Opened CNC Software Based on MS Winsows CE

Taewan Im *, Chulsoo Lee**

*CSCAM Co, Ltd.

**Chonnam University

ABSTRACT

This paper researched about technique of Opened CNC that were able to adapt themselves to rapid development of software and hardware. It is basic research what develop a scheme whereby technic make property.

This paper theorized about to realize Opened CNC Software which is developing CNC Software flow from building Windows CE operating system's image that is possible realtime acting and multitasking. And Opened CNC Software's component designed independent classified modules. Classify Opened CNC Software's component which was consisted of basic OS Kernel, NC Code parser, Servo Motor Control, Software PLC, MMI(Man-Machine Interface). And show there's functional example.

Key Words : CNC, WindowsCE

1. 서 론

근래 PC의 발달로 인하여 전용 하드웨어에 의해 수행되던 작업을 하나의 프로세서에서 소프트웨어를 이용하여 처리하는 경향을 볼 수 있다. 예를 들어, CNC 공작기계의 제어기가 PC로 대체되면서 하나의 PC가 서보 모터의 제어, 가공 지령의 해석 및 경로의 생성, 사용자 인터페이스 등을 모두 하나의 프로세서에서 수행하게 되었다. 하지만 CNC 소프트웨어는 실시간으로 동작하여야 하므로 하드웨어적인 특징에 의해 모든 프로그램이 DOS에서 이루어져 왔다. 이는 개방형 CNC

프로그램에서 이루어져야 할 프로그램의 모듈별 설계에 크게 제약을 가져오므로 이를 위해 실시간 동작과 multi tasking을 지원해주는 OS를 요구한다. 또한 개발 기간을 단축하기 위해 기존의 개발환경과 유사한 환경을 요구하고 있다.

이를 위해 본 논문은 실시간 지원 OS인 WindowsCE를 기반으로 함으로서 저렴한 라이선스 비용과 개발 기간의 단축 개발 환경의 친숙함을 얻도록 하였으며 개방형 CNC 소프트웨어가 갖추어야 할 구성을 제시하였다.

2. 개방형 CNC 소프트웨어

2.1 개방형 CNC machine 의 구조

CNC machine 은 크게 CNC 유닛, 서보드라이버와 Spindle 유닛, 릴레이 제어부로 구분할 수 있다. CNC 유닛은 공구 경로를 해석하고, 생성하며 각종 프로그램을 메모리에 담고 있으며 나머지의 모든 유닛들과 통신을 제어하는 기능을 한다. 릴레이 제어 유닛은 이미 결정되어진 제어 로직을 바탕으로 각종 센서를 통해 기계의 상태를 파악하고 릴레이 등을 통하여 기계의 상태를 변화시키는 역할을 하며 Spindle 유닛과 서보드라이버는 각 축을 구동하여 직접적으로 가공을 하도록 한다.

2.2 개방형 CNC 소프트웨어의 구성

CNC 소프트웨어는 CNC 유닛에 실행파일의 형태로 있으며 실시간 OS를 기반으로 크게 6가지로 각 모듈별로 분류된다. 사용자에게 CNC의 정보를 나타내고 파라미터 등을 입력받는 MMI(Man Machine Interface)와 외부기기에서 직접적으로 가공데이터를 넘겨 받을 수 있는 통신 모듈, 가공 데이터를 해석하여 넘겨 주는 해석 모듈과 공구 경로 생성 모듈, 해석되어진 데이터를 기계에 직접 지령하는 모션 제어(Motion Control) 모듈 마지막으로 스위치와 센서 등을 입력으로 하여 미리 정의되어진 로직으로 Machine의 접점들을 제어하는 접점 제어 모듈로 구성되어진다. 이들 모든 모듈은 PC에서 처리하여야 하므로 PC의 OS가 갖추어야 할 요건으로 Multi tasking의 지원과 실시간으로 동작할 수 있어야 한다.

2.3 개방 제안한 소프트웨어 구성

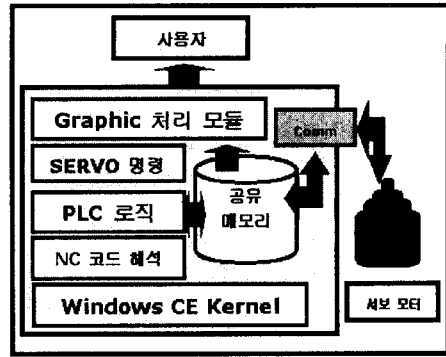


Fig. 1 Proposed consist of software

본 논문에서 제안하는 소프트웨어의 구성을 살펴 보면 실시간 실행과 Multi tasking이 가능한 Windows CE를 기반으로 하여 각각의 모듈들이 독립적으로 동작하고 공유자원에 대한 경합을 방지하기 위해 공유메모리를 기준으로 데이터를 교환한다. Graphic 처리 모듈은 공유 메모리 상에 데이터를 액세스 하여 화면에 표시하는 부분으로 실시간으로 동작할 필요가 없지만 많은 메모리와 시스템의 사용을 요구하므로 다른 모듈들의 처리가 완료되어질 때 까지 대기한 후 시스템에 여유가 생길 경우에 처리되어 지도록 한다.

NC 코드 해석 모듈의 경우 가공하려는 데이터를 해석하여 machine에 적합한 명령단위로 변환하여 준다. 이 경우 지령하는 명령보다 해석하는데 시간이 더 많이 소요되는 경우를 미리 방지하고자 공유메모리 상에 해석한 데이터가 일부분 미리 가지고 있을 수 있도록 버퍼를 둔다.

Servo 명령 모듈은 NC machine에 해석되어진 경로와 가공에 필요한 접점을 지령하는 부분으로 항상 가공 시에는 실시간으로 구동되어야 한다. Servo 명령 모듈이 구동되어질 경우 다른 모듈들은 정지하고 NC Machine의 상태를 계속 확인하면서 명령을 내리도록 한다. Servo 명령 모듈이 모든 일을 완료하였을 경우에도 다른 모듈이 동작 할 수 있도록 Servo 명령 모듈은 시스템의 사용을 중단한다.

PLC 로직 모듈은 하드웨어적인 접점의 제어 로직을 소프트웨어적으로 구동하도록 되어 있으며 미리 정의

되어진 로직에 따라 입력에 관한 출력의 상태를 변화 하도록 한다.

Communication 모듈은 Servo의 지령을 NC machine에 지령하고 NC machine의 상태를 CNC 소프트웨어에 알려 주는 역할을 한다.

2.4 CNC 소프트웨어가 구동 되어질 OS

개방형 CNC 소프트웨어가 구동 되기 위해서는 기본적으로 각 모듈들이 독립적으로 동작할 수 있도록 Multi tasking이 지원되어야 한다. 또한 각 모듈간에 우선 순위를 갖고 시스템의 사용에 관한 스케줄링이 가능하도록 하여야 한다. 이러한 요건을 만족하기 위해서 기존의 PCNC 형태의 CNC 소프트웨어들은 Microsoft사의 WindowsNT에 Realtime OS를 추가하여 실시간을 지원하고 Multi Tasking이 가능하도록 하였다. 하지만 이를 구성하기 위해서는 WindowsNT의 라이선스 비용과 Realtime OS의 라이선스 비용을 부담하여야 하므로 많은 라이선스 비용을 지불하여야 했다. 하지만 Microsoft사의 Realtime OS인 WindowsCE를 사용하므로 저렴한 라이선스 비용과 실시간에 대한 지원을 이룰 수 있었으며 또한 기존의 개발환경과 거의 유사한 개발환경을 구축할 수 있으므로 개발기간을 단축할 수 있으며 별도의 교육 없이 개발이 가능하도록 할 수 있다. 여기에 모든 이벤트들에 대한 처리는 OS에서 관리하므로 개발자가 따로 이벤트들에 대한 명확한 정의 없이 OS의 이벤트만으로도 구축이 가능하므로 편리하게 소프트웨어를 설계할 수 있다.

3. 개방형 CNC 소프트웨어의 설계와 구현

3.1 OS의 구현

Microsoft사의 WindowsCE는 휴대용 장치인 PDA나 별도의 OS Setting이 필요 없이 바로 사용이 가능한 Embedded System에 탑재되기 위하여 개발되었다. 이런 특성 때문에 하드웨어적인 호환성은 떨어지지만 OS의 크기는 약 10mega-byte 정도만으로 구동이 되므로 경제적으로 구축이 가능하다. 또한 OS는 하나의 이미지 파일로 생성이 되어 적재되기 때문에 사용자가 OS를

수정하여서 발생하는 OS의 손상을 방지할 수 있으며 OS의 구동에도 기존의 Windows 시스템에 비해 아주 짧은 시간에 이루어진다. 또한 라이선스 비용도 아주 저렴하고 개발자의 환경을 기존의 Windows 시스템과 동일하게 하기 위한 라이브러리까지 제공된다. 이러한 라이브러리는 Win32와 동일한 구조를 갖는다.

먼저 WindowsCE는 Microsoft사의 Platform Builder3.0을 이용하여 해당하는 하드웨어에 사용하는 CPU를 선택하고 CPU에 맞는 필요한 드라이버들을 선택한다. 그 후 각 파일들을 컴파일 하여 하나의 Binary 파일을 생성한다. 이 파일은 나중에 OS Loader를 통하여 메모리에 적재되어 구동이 된다. 이렇게 생성된 OS에 맞는 Application을 개발할 수 있도록 Platform Builder는 라이브러리를 생성하여 준다. 생성된 라이브러리를 이용하여 OS에 맞는 Application을 개발할 수 있다.

3.2 Graphic 처리 모듈의 구현

사용자와 CNC Machine 간에 명령과 정보의 표현은 Graphic 처리 모듈을 통하여 이루어진다. 이는 사용자에게는 CNC Machine의 상태를 보여주고 CNC Machine을 동작시키는데 필요한 각종 파라미터와 가공 시 필요한 환경설정 값들을 입력할 수 있도록 한다. 이러한 Graphic 모듈은 많은 화면을 필요로 하기 때문에 그에 따른 메모리의 필요량도 많아진다. 이를 해결하기 위해서는 각 종류별로 처리되어지는 그래픽을 사용자의 선택에 따라서 표현되기 직전에 필요한 메모리를 할당 받고 다른 화면의 출력이 필요로 할 경우에 해당 화면은 메모리를 해지 하여 반환하도록 하여 메모리의 사용량을 줄이도록 한다.

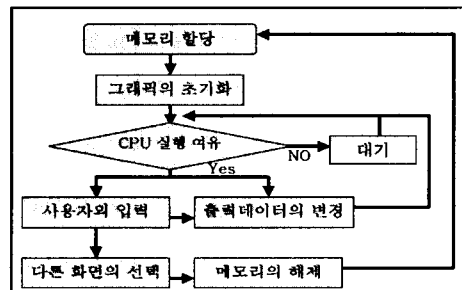


Fig. 2 Proposed consist of software

3.3 NC Code 해석 모듈의 구현

CNC 를 이용한 가공에는 NC Code 에 형식을 갖춘 데이터 파일이 이용된다. 하지만 이 NC 데이터 파일을 직접적으로 Machine 에 지령하기에는 부적합하기 때문에 이를 해석하고 각각에 필요한 요소로 구분한 뒤 해당 되는 부분으로 지령을 내린다.

Fig. 3 에서 볼 수 있듯이 명령은 크게 두 가지로 분류한다. 첫째로는 Tool Position Command 로 직접으로 각 축의 위치를 지령하는 부분이다. 다음으로 부가적인 Command 로 Machine 을 제어하는데 있어서 필요한 Relay 를 제어하는 부분이다.

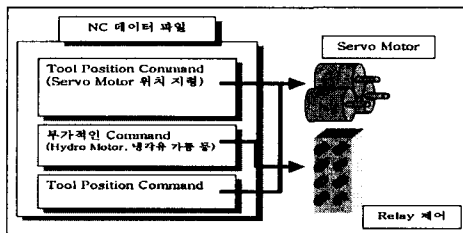


Fig. 3 Assortment of NC Data

Fig. 3 에서 볼 수 있듯이 명령은 크게 두 가지로 분류한다. 첫째로는 Tool Position Command 로 직접으로 각 축의 위치를 지령하는 부분이다. 다음으로 부가적인 Command 로 Machine 을 제어하는데 있어서 필요한 Relay 를 제어하는 부분이다.

3.3.1 Tool Position Command

Tool 의 위치를 지령하는 Tool Position Command 는 기존의 상용 시스템에서 사용되는 NC 데이터의 명령들 경우에 미리 정의되어진 규칙으로 복잡한 경로를 단순하게 표현한 경우가 많다.

복잡한 경로로 정의되어진 Code 들은 먼저 명령을 Cycle 단위로 나누고 각 Cycle 단위의 명령들을 보다 단순한 Segment 단위로 분리한다. 분리된 Segment 들은 직선과 원호로 구분되어지고 원호로 구분되어지는 부분은 다시 직선 Segment 로 변환한 데이터를 생성시켜 준다. 최종적으로 생성되어진 직선 Segment 데이터들을 시스템의 환경에 맞추어 직선 세그먼트 데이터로

생성한다.

3.3.2 부가적인 Command

가공 중에 Machine 의 부가적인 상태를 제어하는 명령이다. 이는 Machine 이 지원하는 Option 에 따라 기능 차이가 있고 Machine 의 제작 과정에서 지원하는 명령들을 미리 정의하여야 한다. 예를 들어 공구를 교환하는 ATC(Auto Tool Changer)에 경우에 서보 모터에의 제어 되기 보다는 Machine 에서 단순히 Relay 의 제어만으로 가능 하기 때문이다. 하지만 ATC 가 없는 Machine 의 경우에는 이러한 명령들을 지원할 수 없기 때문이다.

NC 데이터의 해석은 Fig. 4 와 같이 두가지 방식에 의해 데이터를 해석하도록 한다.

메모리데이터의 해석은 외부에서 CNC Machine 외부에서 데이터를 공급할 경우에 필요하고 파일데이터의 해석은 내부 저장공간에 저장되어져 있는 데이터를 해석할 경우에 사용한다. 또한 데이터가 서보 명령으로 전달할 때 자원으로의 부족으로 인하여 데이터의 해석 시간이 부족할 경우를 대비하여 공유메모리상의 버퍼에 미리 일정량을 해석하여 보관하도록 한다. 이렇게 해석할 수 있는 양이 많은 면 좋지만 메모리의 양이 한정되어 있으므로 일정량만큼을 보관하도록 한다.

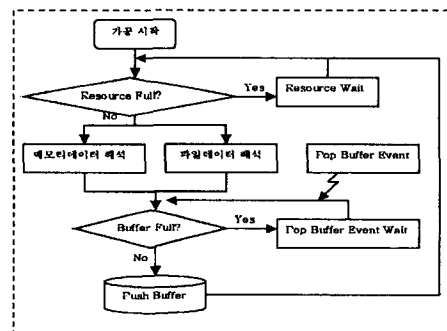


Fig. 4 Flow of NC Parsing

3.4 점점 제어 모듈의 구현

PLC 점점 제어 모듈은 CNC 프로그램 내부에서 출력하는 점점들을 하드웨어 점점으로 미리 설계된 출력 Logic 으로 출력을 하는 것이다. 이는 Servo 제어 모듈

에서 Machine 과의 Communication 모듈의 중간에 위치하여 Servo 제어 모듈에서 Machine 으로 출력하는 접점을 제어한다. PLC 접점 제어 모듈에서는 미리 정의하여둔 Logic 을 Local Storage로부터 읽어 들인다. Logic 을 구성하기 위해서 몇가지 논리 연산자를 지원한다. 먼저 LOAD 명령으로 접점의 상태를 읽고, AND, OR, NOT 의 명령으로 논리 연산을 수행하고 OUT 으로 정의되어진 접점으로 출력한다. 또한 CNC 소프트웨어 시스템의 실시간 OS로부터의 클럭을 이용하여 Timer 를 구현한다. 입력이 들어 오고 일정시간이 지난 후에 반응하는 TON, 입력이 끊어지고 일정시간이 지난 후에 출력을 중지하는 TOFF 등의 Timer 명령을 지원한다.

PLC 접점 제어 모듈의 경우에는 내부의 독립모듈로 동작하지만 다른 모듈에 비해 실시간을 요구하지 않으므로 다른 모듈의 리소스 사용을 위해 자신의 모듈은 리소스의 여유가 있을 때 까지 대기하도록 한다.

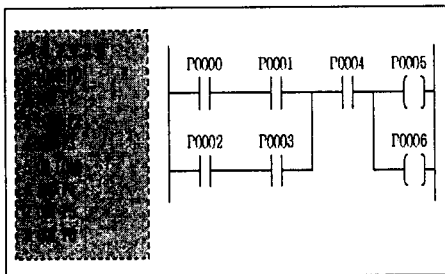


Fig. 5 Example of PLC Logic

3.5 Communication 모듈의 구현

Communication 모듈의 경우 NC 장비와 통신을 하기 위해 H/W 적으로 구현을 한다. H/W 적으로 특정한 메모리 어드레스 공간에 일정영역을 확보하고 이 공간을 OS Kernel Level 에서 NC 통신 장비의 메모리에 기록을 하도록 한다. 이렇게 하여 NC 통신 장비는 이 메모리 공간에 있는 데이터를 NC Machine 으로 보내어지게 된다. NC Communication 장비의 공유 메모리를 제어하기 위해서는 하드웨어적으로 접근이 이루어져야 하므로 특별히 신경을 써야 한다. 독립적인 모듈에서 동시에 한 물리적인 주소를 접근하여 액세스하려는 것

을 방지한다. 그렇기 위해서는 Communication 모듈과의 통신은 오로지 한 영역에서만 이루어지도록 한다. 다른 모듈에서 직접적으로 Communication 모듈과의 데이터 교환은 되도록 자제하도록 하며, 모든 Communication 모듈과의 데이터 교환은 공유메모리를 이용하여 교환하도록 한다.

3.6 서보 명령 모듈의 구현

서보 명령에 있어서 지령하는 방법은 원점 복귀의 상태와 수동조작의 상태, 자동 가공 및 원격 가공, 데이터 입력 및 수정의 4 가지 형태로 구분하여 지령한다.

원점 복귀와 수동 가공의 상태에서는 서보에 지령하는 방법은 작업자의 입력만을 기다렸다가 작업자의 입력이 있을 경우에만 동작을 하도록 한다. 자동 가공 및 원격 가공의 경우에 NC 해석 모듈등에서 해석되어진 데이터를 파라미터등과 조합하여 명령을 내린다. 데이터의 입력 및 수정의 경우에는 Machine 이 동작하지 않도록 고정하는 역할을 한다.

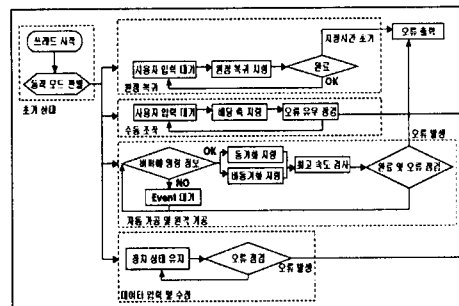


Fig. 6 Flow of Servo Command Module

3.7 공유메모리의 구현

모든 모듈간의 데이터 이동은 공유메모리를 통하여 이동하고 모듈간에 직접적으로 데이터의 이동은 못하도록 한다. 공유메모리는 용도에 따라 크게 두 가지로 분류하여 사용한다. 그 중 하나인 선형적인 메모리는 파라미터나 Machine 의 상태 등을 갖고 있는 경우에 사용하고 나머지 하나인 링 버퍼의 경우에는 서보 지령 명령을 저장하는 버퍼의 경우에 사용한다.

4. 구현 사례

하드웨어적인 구성을 살펴 보면 작업자에게 상태를 보여주는 CRT와 CNC 소프트웨어를 구동하는 PC인 One Board PC, 작업자의 입력을 받는 OP Panel, 각 축을 움직이는 Servo Motor, Servo Motor를 제어하는 Servo Driver와 Relay 등을 제어하는 Relay 제어부로 이루어 구성하였다.

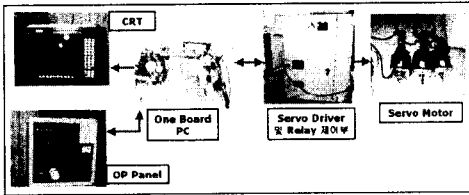


Fig. 7 Component of CNC Machine

구현 되어진 CNC Machine은 2개의 X,Z축과 하나의 Spindle을 갖는 선반이며 Storage로는 DiskOnChip2000-16M와 Ethernet Lan을 통하여 네트워크가 가능하도록 하였다.

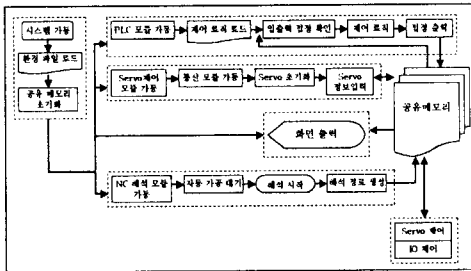


Fig. 8 Flow of Full Software

CNC 소프트웨어의 전체적인 흐름은 Fig. 8과 같다.

5. 결론

본 논문은 MS-WindowsCE 기반의 개방형 CNC 소프트웨어에 관한 연구를 하였다. 개방형 CNC 소프트웨어의 구동을 위해 MS-WindowsCE를 기본 OS로 채택하고 MS-WindowsCE를 시스템에 특성에 맞게 생성하였으며 이를 토대를 최적화된 라이브러리의 생성방법과 CNC 소프트웨어의 제작에 관한 흐름을 설명하였다. 개방형 CNC 소프트웨어가 갖추어야 할 NC Code의 해

석과 해석되어진 데이터를 근거로 Servo의 컨트롤, 사용자의 Machine 제어를 위한 PLC 등을 기술하였으며 이들의 독립적인 동작을 위한 프로세스의 생성과 관리, 시스템 자원의 효율적인 사용과 모듈간의 통신을 위한 공유메모리의 관리에 대하여 기술하였다. CNC Machine의 개발에 있어서 CNC 제어 소프트웨어에 종속되는 것을 막고자 범용적인 개방형 CNC 소프트웨어의 개발 방법을 정의하였다. WindowsCE를 기본 OS로 채택함으로써 시스템의 필요 사양을 낮추었으며 모듈간의 독립적인 동작을 위해 WindowsCE의 멀티 태스킹을 이용하였으며 모듈간의 우선순위를 정의하였다. 제안된 각 모듈의 알고리즘과 방법론은 C언어를 이용하여 구현하였으며 WindowsCE OS를 제작하기 위해 Microsoft사의 Platform Builder 3.0을 사용하였으며 Application 개발을 위해 Microsoft사의 Platform Builder로부터 라이브러리를 생성하여 Microsoft사의 Embedded Visual C++3.0을 이용하여 컴파일 하였다.

참고문헌

- 이제필, "개방형 CNC 컨트롤러의 고속, 고정도 가공을 위한 실시간 선독보간 및 NURBS 보간 알고리즘", 박사 학위 논문, 전남대학교, 2001
- Microsoft Press, "Microsoft Windows CE Developer's Kit"
- James Y. Wilson, Aspi Havewala, "Building Powerful Platforms with Windows CE", ADDISON-WESLEY, 3, 2001
- 이철수, 이제필, "실시간 제어에 의한 개방형 CNC 소프트웨어 모듈의 설계 및 구현", 한국공작기계학회지, 제 8 권 제 5 호, pp.54-62, 1999
- 한기상, 권용찬, 김주한, "모듈구조를 갖는 개방형 CNC의 구현", 한국정밀공학회지, 제 17 권, 제 5 호, pp.34-40, 5 월, 2000
- 최중률, 개방형 CNC 기술과 발전 방향, 제 1 회 첨단 CNC 공작기계 국제기술 세미나, pp.59-80, 10 월, 2000