G-코드 해석기를 내장한 NC 제어기 개발

Development of an NC System with a G-code Interpreter

*#김병섭, 노승국, 박종권

*#B. S. Kim(bkim@kimm.re.kr), S. K. Ro, J. K. Park

한국기계연구원 지능형생산시스템연구본부

Key words : CNC, G-code, Digital Controller

1. 서론

공작기계를 운전하는데 있어서 G-코드는 기계에 명령을 내리기 위한 일반적인 언어이다. G-코드의 기본 집합체는 RS-274로 1960년대 초반 미국 전기 산업 협회(Electronic Industry Association) 의하여 수치제어로 작동하는 기계의 표준 코드로 제정되었으며 1980년에 개정되어 사용되고 있다. 산업계에서는 G-코드의 기본집합에 대하여 표준화하여 사용하고 있지만 여러 업체마다 업체 특성에 맞게 변형된 G-코드 또한 사용되고 있는 실정이다. 이 논문에서는 탁상용 소형 밀링머신의 제어를 위해 제작된, 기본적인 G-코드 명령을 해석하고 수행하는 NC 시스템의 개발에 대하여 기술한다. 개발된 NC 시스템은 PC 기반 시스템으로 윈도우 환경의 사용자 인터페이스 부분에서 1차적인 G-코드의 선처리를 하고 그 결과를 원형 버퍼를 통해 DSP 프로그램 쪽에 넘겨주면, 2000 Hz의 타이머 인터럽트에 의해 구동되는 DSP 보드에서는 1차 처리된 G-코드를 실시간으로 해석하여 각축의 지령 좌표를 생성하고 서보 제어루프를 구동해 나가는 구조로 되어있다. 사용된 DSP 보드는 Spectrum사의 Daytona 보드로 TI사의 TMS320C6701칩을 탑재하고 있으며 현재 구현된 G-코드와 일부 M-코드는 다음과 같다: G00 (Rapid Motion), G01 (Linear Motion), G02 (CW Circular Arc), G03 (CCW Circular Arc), G04 (Dwell), G17 (X-Y Plane Selection), G18 (Z-X Plane Selection), G19 (Y-Z Plane Selection), M21 (Contouring On), M22 (Contouring Off), M30 (Program End & Reset).

2. NC 제어기의 구성 - 사용자 인터페이스

개발된 NC 시스템은 크게 두 개의 프로그램에 의해 구성되어 있다. 첫 번째는 윈도우 환경에서 실행되어 사용자의 입력을 받는 PC쪽의 사용자 인터페이스 프로그램이고 두 번째는 1초당 수 천번씩 타이머 인터럽트를 받아서 실시간에 기계의 각 구동축에 대한 명령을 해석하고 서보 제어 루프를 구동하는 DSP 프로그램이다. 두 프로그램은 쌍방향 공유 메모리(Dual Port RAM)를 통하여 1차 처리된 G-코드나 조그 명령 등을 DSP쪽으로 보내거나 DSP 프로그램의 상태를 사용자 인터페이스 쪽에 알리면서 서로 간의 통신을 유지한다.

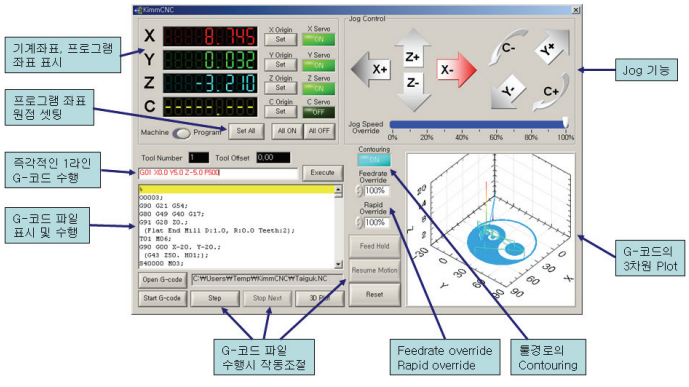

Fig. 1 Graphical user interface of an NC system

Fig. 1은 사용자 인터페이스의 외관과 제공되는 기능에 대한 간략한 설명을 보이고 있다. 특징적으로 사용자가 G-코드 파일을 읽어 들이면 우측 하단에 툴경로를 3차원 그래프로 표시하는 기능이 있어서 공작기계 구동 전에 전체 툴경로를 파악할 수 있으며 또한 현재의 툴위치가 붉은점으로 3차원 그래프에 표시되어 G-코드 상의 어느 위치를 가공 또는 이송하고 있는지 이해하기 쉽도록 되어 있다. 또한 컨투어링(Contouring) 기능을 설정할 수 있도록 하여 G-코드에 의해 생성되는 연속되는 툴경로가 공간상에서 10° 보다 작은 각도로 붙어서 연속되는 경우 감속 없이 바로 다음 경로로 진입할 수 있도록 하여 감속-정지-가속을 행함으로써 오는 이송시간의 손실을 막을 수 있도록 하였다.

사용자가 G-코드 파일을 'Open G-code' 버튼을 눌러서 열면 전체 파일 내용을 읽어서 버퍼에 저장하고 좌측 하단의 리스트 박스에 그 내용을 라인별로 표시한다. 'Start G-code' 버튼을 누르면 사용자 프로그램은 G-코드를 순차적으로 한 라인씩 버퍼에서 뽑아서 문법 체크와 함께 의미있는 어절인 토큰(token)을 분리한다. 예를 들어서 'G01 X100 Y50 Z0 F500' 이라는 G-코드 라인을 토큰으로 분리하면 수행되어야 할 작업이 직선 운동(G01)이고 최종적으로 움직이고자 하는 좌표가 (100, 50, 0)에 요구되는 벡터상의 피드레이트(feedrate)가 500 mm/min 임을 내부변수에 기록한다.

그 다음에는 G-코드의 전처리를 수행하는데 우선 운동 특성을 파악하여 그것이 X-, Y-, Z- 특정축에 국한하여 일어나는 것인지 특정 평면, 또는 XYZ 공간에서 일어나는 것인지를 알아낸다. 예를 들어, 위의 G01 명령의 수행을 시작할 때 시작 위치 좌표가 (35, 50, 0)라면 G01은 X-축만 35에서 100으로 이송하면 되는 것이고 (35, 40, 0) 위치라면 X-축은 35에서 100으로 Y-축은 40에서 50으로 이송하되 Z-축은 현재 위치에 그대로 있으면 되는 것이다 (즉, XY 평면 이송). 사용자 인터페이스 프로그램에서는 한 개의 축만 움직이면 myline() 함수로, 두 개의 축이 움직이면 myplane() 함수로 세 개의 축이 동시에 움직이면 myspace() 함수로 처리하도록 하였는데 두 개 이상의 축이 동시에 움직일 경우라도 어느 한축을 운전축(driving axis)으로 설정하여 그 운전축의 속도, 가속도, 감속 시작 위치를 사전에 계산한다. 이때 운전축의 속도는 피드레이트 뿐만 아니라 각축의 최대 제한 속도를 고려하여 한계를 벗어나지 않도록 3차원 벡터량을 각축에 투영하여 계산하도록 하였다.

전처리가 끝난후 원형 버퍼를 통해 DSP 쪽에 제공되는 정보는 G01의 경우, 라인번호, 운동모드 (직선운동이 특정축이나 2차원 평면에 국한되는지 XYZ 3차원 공간에서 일어나는지 구별하여 정보 지정), 시작점 (x, y, z) 좌표값, 종점 (x, y, z) 좌표값, 운전축의 좌표가 종점을 향해 증가하는지 감소하는지에 대한 플래그, 운전축의 속도, 운전축의 가속도, 운전축의 감속 시작 위치, 컨투어링 기능 사용시 사잇각을 계산하기 위한 방향여현(directional cosine) 값들이다. G02와 G03의 원형 운동의 경우 추가적으로 중심점 (x, y, z) 좌표값, 원형 운동의 수직축 방향, 호의 시작각과 끝각이 계산되며 속도, 가속도는 각속도, 각가속도 단위로 계산되어 원형버퍼를 통해 DSP에 넘어간다.

사용자 인터페이스 프로그램과 DSP 프로그램 사이의 원형 버퍼는 크기가 제한되어 있어서 현재 4개 라인의 G-코드 전처리 정보만 수용 가능하다. 따라서 사용자 인터페이스 프로그램은 1초에 수회씩 원형 버퍼를 체크하여 DSP 프로그램이 수행을 완료하여 비워진 버퍼 영역에 전처리를 수행한 G-코드를 넣어주도록 되어있다.

3. DSP 프로그램

DSP 프로그램은 G-코드 명령을 실시간으로 인터폴레이션(interpolation) 하여 각 축의 추종되어야 할 위치 명령값을 만들고 PID 따위의 서보 제어루프를 구동하여 각 축을 원하는 위치로 제어하는 것이 주 기능이다. 일반적으로 제어 대상물의 주파수 대역폭(bandwidth)에 10배 정도를 곱한 값을 샘플링 레이트(Sampling rate)로 설정하여 초당 수천 번씩 계산 및 제어를 수행하는데 개발된 NC 프로그램에서는 2000 Hz의 샘플링 레이트를 사용하도록 하였다.

DSP 프로그램에서는 한 개의 G-코드 라인을 원형버퍼에서 가져온 후 다음 라인의 G-코드와 툴 경로의 인접 각도를 계산한다. 그 각이 10° 보다 작고 컨투어링 기능이 선택되어 있으면 현재 G-코드의 목표 좌표에 이르러서도 감속하지 않고 속도를 유지한 채 다음 라인의 G-코드를 수행하도록 내부 플래그를 설정한다. 일정한 시간 간격을 가지고 타이머 인터럽트가 걸릴 때마다 DSP 프로그램은 이전의 속도와 가속도를 가지고 현재의 속도와 위치를 계산하여 서보 제어루프가 추종해야 할 명령값을 생성한다. 이때 계산된 속도값은 사용자 인터페이스에서 계산된 속도 보다 클 수 없으며, 계산된 위치가 감속위치에 도달했고 컨투어링을 사용하지 않는 경우 종점 좌표값에서 속도가 0이 되도록 감속하여 계산한다. 직선운동인 경우, 계산은 이미 설정된 운전축을 대상으로 하며 나머지 축들은 시작점 좌표값과 종점 좌표값에 의한 직선식을 이용하여 계산한다. 원호 운동의 경우 각속도와 각가속도를 이용하여 직선운동과 같은 방법으로 계산한 후 최종적으로 사인과 코사인을 이용하여 직교좌표계에서 각축의 명령값을 생성한다.

G-코드의 인터폴레이션으로 각축의 실시간 명령값을 생성한 후 DSP 프로그램은 서보 제어 루프를 구동한다. 피드백 되어 읽은 각축의 실제 위치값과 계산된 명령값과의 차이를 계산하여 그 오차를 바탕으로 PID 제어기나 다른 제어이론에 의해 설계된 제어기를 이용하여 제어신호를 계산한다. 계산된 제어신호는 각 축의 모터 드라이버에 보내어 명령된 위치에 각 축이 이송되도록 한다.

실제 NC 제어기는 위에서 기술한 G-코드 해석 및 수행 이외에도 고려해야 할 다른 기능들이 많다. 예를 들어 G-코드의 수행 중에 잠시 멈추었다 다시 계속하기, 조그 모드로 수동 운전하기 등은 기본적으로 갖추어야 할 기능 중의 하나이며 사용자 인터페이스 프로그램과 DSP 프로그램 사이에서 어떻게 명령과 필요한 데이터를 주고 받을지 잘 설계되어야 한다.

Fig. 2 Experiment of the NC system with a 3-axis desktop milling machine

개발된 NC 시스템은 Fig. 2에서 보이는 것과 같이 탁상형 3축 밀링머신에 장착하여 NC로서의 기능을 테스트 하였다. 상용 CAM 프로그램을 이용하여 마이크로 펌프 금형을 가공하기 위한 G-코드를 생성한 후 가공을 수행하였는데 가공 결과로 Fig. 3에 몇가지 예를 보이고 있다. 사용된 3축 밀링머신은 X-, Y-축에 보이스 코일 모터와 LM 가이드를 가지고 있으며 Z-축은 리니어 모터와 공기 베어링을 장착하였고 약 25x25x25 mm^3의 스트로크를 가지고 있다.

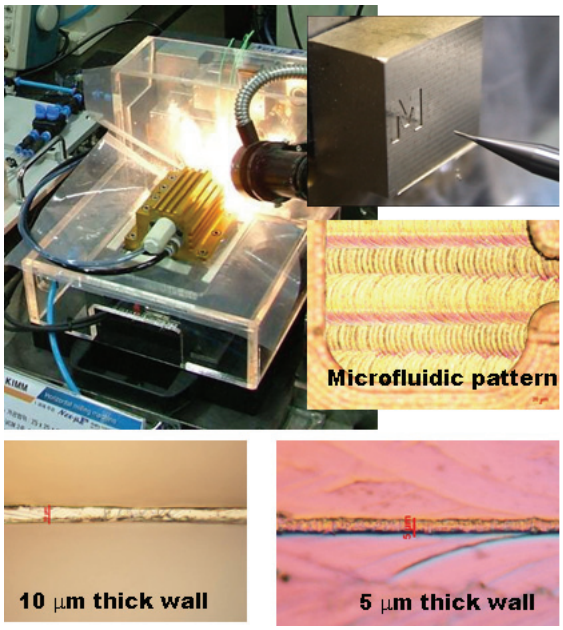

Fig. 3 Experimental results from the NC system and a 3-axis desktop milling machine

4. 결론

PC를 기반으로 하는 NC 시스템을 개발하였다. 윈도우 환경에서 구동되는 사용자 인터페이스 프로그램과 DSP에서 구동되는 DSP 프로그램으로 구성되어 있으며 G-코드 인터프리터 기능을 구현하여 상용 CAM 소프트웨어에서 생성된 G-코드를 이용한 가공이 가능하다. 개발된 NC 프로그램의 다른 특징으로 G-코드를 3차원 그래프로 그려주는 기능과 전체 G-코드의 수행시간을 줄여주는 컨투어링 기능이 있다. 개발된 NC 시스템을 탁상형 3축 밀링머신에 적용하여 마이크로 펌프 금형 가공 등 기본적인 NC 기능이 현장에 적응 가능함을 확인하였다.

참고문헌

1. Smid P., "CNC Programming Handbook, Third Edition" Industrial Press, Inc., 2007.
2. Gibbs D. and Crandell T., "An Introduction to CNC Machining and Programming " Industrial Press, Inc., 1991.