

# 선반용 대화형프로그램의 개발 (A Development of Conversational Program for Lathe)

신동수\*, 권영두 (화천기공(주)), 정성중 (한양대 공대)

D.S.Shin, Y.D. Kwon(Hwacheon machine tools co.,ltd), S.C. Chung(Hanyang Univ.)

**Key Words** : Part Program (가공프로그램), Numerical Control (수치제어), CAM (Computer Aided Manufacturing), Manual Programming (수동프로그래밍), SFP (Shop Floor Programming; 작업장프로그래밍), Tool Path (공구궤적), Machining Process (가공공정), Tool Database (공구데이터베이스), Graphic-Based Programming (그래픽기반형 프로그래밍), Conversational Program (대화형 프로그램)

## 초록

This paper describes a development of conversational program for lathe. The proposed program is a NC programming system specialized for use in machine tool controls. Its easy to use graphic interface, built-in database capabilities make it an ideal way to program part on the shop floor.

In order to manage effectively machining process, it proposed the algorithm of tree structures for design processing. Also using interactive structure, it proposed the methods of conversational programming to generate automatically NC program.

## 1. 서론

요사이 다중 소량생산의 발전에 힘입어 대화형 프로그램에 대한 많은 관심을 가지게 되었으며, 숙련작업자의 감소와 경험이 적은 작업자가 간단하게 작업할 수 있는 요구 등의 사용자들의 환경변화로 인하여 관심이 높아지고 있다.<sup>(1)</sup> CAM

(Computer Aided Manufacturing) 은 컴퓨터에 의한 생산기술시스템이며 주로 가공을 위한 정보를 처리한다. 다시 말하면, 컴퓨터를 이용함으로써 제조공정의 공기단축과 정도향상, 숙련기능자의 부족 및 복잡한 형상의 제작 등의 문제점을 해소하는 역할을 하는 것이다.<sup>(2)</sup> 따라서, 본래는 공정설계부터 시작하여 그 범위는 상당히 넓다.

CAM 의 발전은 사실상, NC (Numerical Control) 공작기계의 발전과 함께 되어왔다. 그때까지 마스터 모델 (목형) 을 이용하여 그 형상을 탐침 (Probe) 에 의해 같은 모양으로 공구를 움직여서 가공하는 모방 공작기계가 가공의 주류였다. 1952 년 MIT 에서 최초로 NC 제어장치가 프라이스 선반에 이용되었다. 그후 여러 가지 NC 공작기계가 개발되고, NC 프로그램 자동작성 장치의 개발도 이루어졌다. 1955 년 APT (Automatical Programmed tools), 1957 년 APT-II, 1961 년 APT-III, 1965 년에 EXAPT (Extended APT)-I, II 와 1970 년에 EXAPT-III 등이 개발되어 실용화 되었다.<sup>(3)</sup>

자동 프로그래밍 시스템은 초기에 대형 컴퓨터 상에 개발되었지만 일본에서는 1970년대가 되자 마이크로 컴퓨터에 자동 프로그래밍 시스템을 자동시키게 되었다. 그러나, 수치제어 공작기계의 작업명령 언어인 파트 프로그램 (Part Program) 을 작성하기 위하여 전산기를 이용하는 CAM 시스템은 현재에도 그 시장을 넓혀가고 있으나 프로그램을 습득하는 데 많은 시간이 소모되며, 비교적 단순한 형상의 공작물 특히 선삭 가공물을 위한 CAM 시스템의 채용은 가격면에서 타당성이 낮은 편이다.<sup>(4)</sup> 작업장에서 작업자가 직접 가공프로그램을 작성하고 초기 시험 절삭시의 작업자 보정치가 반영되어 최적화된 가공프로그램의 생성을 도모하기 위하여 각 CNC 시스템 제조회사에서는 자사의 수치제어 장치에서만 사용 가능한 특별한 프로그래밍 도구를 개발하여 기본 혹은 선택사양으로 제공하고 있다. Simens 사의 Blue-print/ Support 프로그래밍과 WOP 시스템, MAZAK 사의 대화형프로그램, MORISEIKI 사의 MORIC-3Y, Enshu 의 TOMATO, YASNAC 사의 Compact 프로그래밍, OKUMA 사의 RakuRaku (らくらく) 프로그래밍 및 FANUC 사의 Super-Cap, Symbolic-Cap 등이

있다. 이러한 프로그램은 자사의 수치제어장치만이 이해하도록 설정되어 있다. 그러나, 현재 MMI (Man Machine Interface) 의 개방요구를 통하여 기존의 전용 OS (Operating System) 환경이 아닌 Windows 환경하에서의 개발을 추구하고 있다.

따라서 가공 Know-How 가 포함된 확실한 가공프로그램의 작성, 단기간의 프로그램 완성, 풍부한 가공메뉴를 통한 가공능력의 강화, 가공시간 단축을 위한 대화형 및 테스트 가공 모드 (Mode) 를 위한 대화형 프로그램의 개발이 필요하며, 또한 다종의 가공사이클을 개발하여 작업현장에서 손쉽게 운영할 수 있는 대화형 프로그램의 개발이 필요하다.

본 논문에서는 오픈 아키텍처 (Open Architecture) 개념을 통한 Windows 환경하의 선반용 다기능 대화형 프로그램 개발에 대하여 서술하였다. 다목적의 도형데이터 정의, 가공공정을 위한 공구데이터 정의 및 공구와 공작물의 간섭과 충돌회피를 위한 테스트 가공 알고리즘을 통하여 가공프로그램 단시간의 편집을 할 수 있으며, 풍부한 데이터베이스를 통한 가공능력의 향상과 가공 Know-How 가 포함된 가공사이클의 개발함을 목적으로 한다. 그래픽 기반형 프로그래밍 기법을 통하여 초보자가 이해하기 쉽고 사용이 용이한 프로그래밍 방법론을 제시하고, 작업장 프로그래밍과 CAM 시스템의 혼함을 통하여 보다 친숙한 대화형 프로그램을 보여준다. 또한, 공구 데이터베이스를 통한 가공공정 관리방법론을 제시하고 다양한 가공프로그램을 생성하기 위하여 Preprocessor 방법으로 관리화일을 차별화하며 도형의 정의와 대화형 프로그램을 위한 대화형 데이터 구축을 제시한다. 따라서, 자동 가공프로그램의 생성을 보다 간편하고 그래픽을 통한 검증으로 신뢰성 확보를 보여줌으로써 프로그램의 유용성을 입증한다.

## 2. 가공 프로그래밍 시스템의 분류

새로운 세대의 NC 프로그래밍 시스템이 도래하고 있으며, 그것은 Control - Resident Shopfloor Programming 시스템과 오프라인 (Off-Line) CAM 시스템을 결합시킨 형태인 Companion 프로그래밍 시스템의 개념이다.<sup>(5)</sup> 이러한 시스템은 G 코드 작업의 필요성과 공학분야 및 제조분야에서의 중복성을 제거할 수 있으며 작업자의 오류를 감소시킬 수 있다. 가공프로그램은 온라인 (On-Line) 혹은 오프라인 (Off-Line) 으로 생성할 수 있다. 온라인의 의미는 공작기계 제어기 상에서 직접적으로 프로그래밍할 수 있는 것을 말하며, 일찍이 제어기를 통하여 가공코드를 입력 혹은 MDI (Manual Data Input) 로 입력하는 방법이라 할 수 있다. 오프라인의 의미는 공작기계가 아닌 다른 곳에서 파트프로그램을 생성하는 것을 의

미하고 수동 혹은 프로그래밍 시스템의 도움을 통하여 가공프로그램을 생성하는 것을 말한다.

각각의 방법론에서는 장점과 단점을 가지고 있으며, 실제로 어디에서 프로그래밍을 할 것인가 보다는 어떻게 프로그램을 생성할 것인가가 더 중요한 문제로 대두된다. 근본적으로 수동 프로그래밍 (Manual Programming), 작업장 프로그래밍 (Shopfloor Programming) 및 CAM 으로 나눌 수 있다.

### 2.1 수동 프로그래밍

Fig. 1 은 수동 프로그래밍을 위한 흐름도를 나타내는 것으로 Line by Line 을 통하여 프로그램을 작성하게 된다. 각 제조회사에 따른 다양한 가공코드 형식, 수치계산 및 기하학적인 지식을 요구하며, 프로그래머는 절삭에 따른 부품상의 특별한 사양을 계산하고 편집기 상에서 프로그램을 써 나가야 한다.

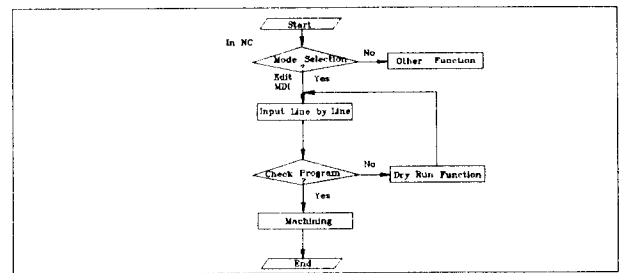


Fig. 1 Flowchart for Manual Programming.

숙련된 작업자는 수동프로그래밍을 통하여 탭핑, 드릴링, 보링 및 단순한 페이스 컷과 같은 단순한 작업을 수행하기 위한 효율적인 방법을 알고 있지만, 복잡한 부품 (Complex Parts) 에 대해서는 많은 노력과 시간이 필요하다. 심지어 더 복잡한 부품은 프로그램하기가 어렵거나 불가능하게 된다. 수치계산, 프로그램 편집 및 입력에 대한 오류가 빈번히 발생하며 수동프로그래밍 기법으로 작성된 프로그램의 검증은 Dry Run 방법만이 가능하다.

### 2.2 작업장 프로그래밍

Fig. 2 는 작업장 프로그래밍 방법의 흐름도를 나타내며, NC 공작기계가 발달됨에 따라 제어기 제조회사에서는 더 쉽게 사용할 수 있는 방법에 대하여 연구하게 되었으며, 이러한 영향은 SFP (ShopFloor Programming) 제어의 개발 연구를 유도하였다.

제 1 세대 SFP 제어는 빈칸을 메우는 (Fill - in - the - Blank) 접근법을 통하여 프로그래밍하는 대화형이었다. 최근에는 대화형 그래픽 SFP 제어가 등장하고 있으며, 이것은 부품 기하학 생성, 공구궤적 생성 및 검증을 문자가 아닌 그래픽을 통하여 할 수가 있다. 일반적으로 언어기반형 및 그래픽 기반형 시스템은 문자기반형 시스템 보다 더 쉽게 이해 및 사용할 수 있다.

종래의 명령라인 인터페이스 (Command Line

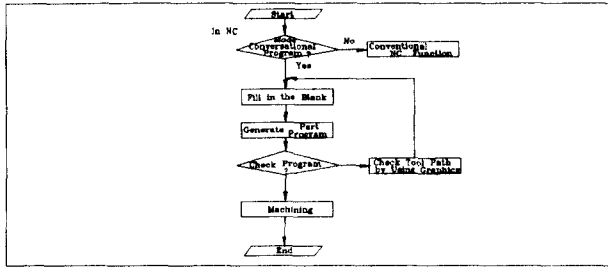


Fig. 2 Flowchart for Shopfloor Programming.

Interface) 보다 그래픽 인터페이스 (Graphic Interface) 의 경우가 사용자에게 높은 생산성의 성과를 보여주고 있다.<sup>(6)</sup> SFP 제어는 몇가지 장점을 제공한다. 그 하나는 SFP 제어를 사용하여 부품을 프로그래밍하는 것은 수동 프로그래밍 방법보다 편리하다. 단순한 CAM 시스템과 같은 함수를 가지고 있으며, 가공 프로그램의 생성과정 중 작업자의 오류를 줄일 수 있다. 또한, 프로그램상에서 수치계산을 수행할 수 있고 좋은 SFP 시스템은 프로그램의 수정 및 편집을 빠르고 간단히 수행할 수 있다. 가공 프로그램의 검증에 위한 그래픽 피드백 (Graphic Feedback) 을 포함하고 있다. 많은 SFP 제어는 가공중에 프로그램을 작성할 수 있는 백그라운드 프로그래밍 (Background Programming) 을 할 수 있다. SFP 는 협소한 작업장에서 동일한 작업자가 부품을 프로그래밍하거나 일을 셋업하거나 작업할 경우 유리하다. 또한 작업의 효율성 판단을 통하여 직접적으로 프로그래밍하거나 대화로써 작업을 하는 것을 선택할 수 있다.

그러나, 열악한 작업환경에서는 편안하게 작업을 할 수 없으며, 더 복잡한 형상을 작업하는 경우에 어느정도 제약을 지니고 있다. 이러한 이유 때문에 상대적으로 단순한 부품에 사용되며 일대일로써 작업을 수행하기 때문에 다른 공작기계에서 어떻게 프로그래밍할 것인가의 문제를 해결하기가 어렵다.

### 2.3 CAM 시스템

Fig. 3 은 CAM 에 대한 흐름도를 나타내며 오프라인 CAM 시스템은 수동프로그램의 단점을 해결하기 위하여 사용되어 왔다. 단순하거나 복잡한 부품을 정확하고 빠르게, 또한 쉽게 프로그래밍하기 위하여 설계되었다. 대부분의 CAM 시스템은 다음 3 가지 형태로써 부품을 프로그래밍하고 간단한 방법으로 작업을 수행한다. 부품의 기하학을 정의하는 Modeller 기능, 공구와 커터패치를 생성하고 기계가 이해하는 형식으로 변환하는 Postprocessor 기능을 수행한다. CAM 은 프로그램 과정을 다양한 각도에서 화면상에 보여주는 방법을 통하여 작업자의 오류를 방지한다. 컴퓨터상의 모의가공을 통하여 가공프로그램을 검증하며

유용한 가공시간을 계산한다. 보다 정확한 가공 프로그램을 생성하기 위한 Postprocessor 기능을 가지고 있고 다양한 부품과 많은 기계의 가공프로그램을 단일 시스템상에서 프로그래밍 할 수 있다.

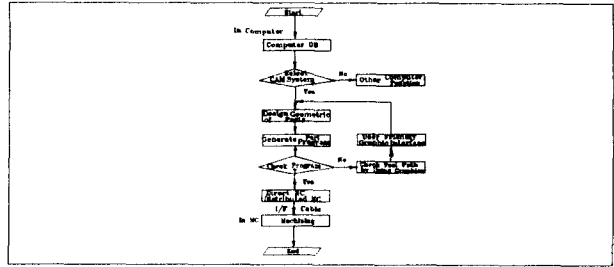


Fig. 3 Flowchart for CAM System.

Fig. 4 는 숙련도, 축수 및 대화의 요구도에 따른 대응 시스템을 나타내고 있다.<sup>(5,7)</sup> 여기서, 기초 프로그래머와 작업자의 경우 대화의 요구도는 높아지고 있으며 숙련가일수록 대화의 요구도는 낮아진다. 따라서 대화성이 높고 다축을 수용하며 초보자용의 프로그래밍 환경이면 좋은 환경을 형성할 수 있다.

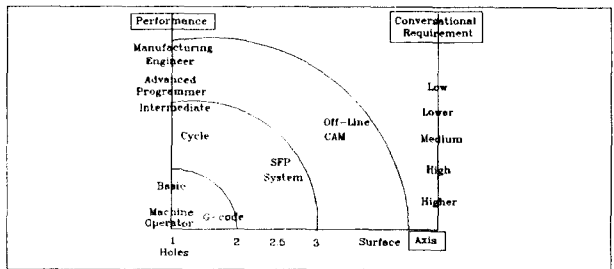


Fig. 4 Classification of Various Programming Systems for Conversational Requirement vs Axis vs Performance.

### 3. 개발환경 및 구성모듈 설계

Fig. 5 는 Windows 환경하의 Borland 사에서 제공하는 Delphi 프로그램 Tool 을 이용하여 개발된 프로그래밍 시스템의 구조를 나타내고 있으며, 그림과 같이 6 가지의 구성모듈을 설계하였다.

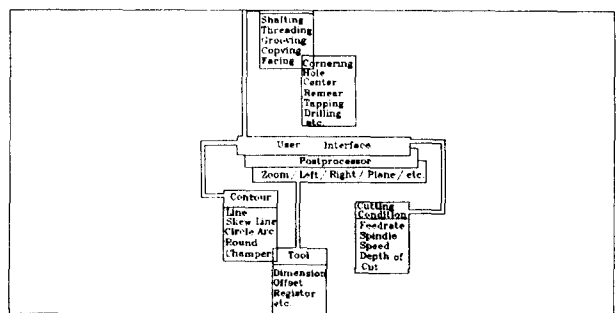


Fig. 5 Structure of the Developed System.

가공물 가공물 재질, 가공물 치수 및 형태 등을 고

려하여 가공물 형상을 정의한다.

가공물 윤곽 황/ 중/ 정삭을 위한 가공물의 윤곽을 그래픽 기반 대화형을 통하여 최종 형성되는 가공물의 윤곽을 정의하는 기본적인 요소로 구성된다. 선, 사선, 원, 원호, 라운드, 모따기 등으로 구성된다.

가공물 사이클 다양한 가공동작을 1 개의 블록(Block) 으로 지령하여 효율적으로 프로그램을 작성할 수 있도록 하는 가공공정 모듈이다. 봉제가공의 6 가지, 나사가공의 2 가지, 코너가공의 6 가지, 모방가공의 6 가지, 단면가공의 6 가지, 홈가공의 6 가지, 구멍가공의 5 가지로 총 37 가지의 사이클을 개발할 예정이다.

공구 공구재질, 공구형상, 터렛번호, 공구치수 및 공구종류 등을 선택하여 가공에 쓰일 공구를 정의한다. 공구 데이터는 절삭조건과 연계하여 데이터베이스에서 관리한다.

절삭조건 절삭가공에 관련된 중요한 정보인 이송속도, 스핀들속도, 절삭깊이 및 표면조도 등을 정의한다.

포스트프로세서 이 부분은 현 가공물의 확대/축소 등과 같이 보는 각도를 자유자재로 할 수 있도록 하는 기능등이 있고 또한 가공코드의 생성 모듈이 있어 제어기 종류에 따른 가공 프로그램을 생성한다.

#### 4. 프로그램 구조 및 기능

##### 4.1 도형 정의 알고리즘

도형정의를 보다 간편하게 하기 위한 모델을 Fig. 6 에 나타내었으며, 그림과 같이 각각의 최종 공작물의 형상을 Table 2. 의 선, 사선, 라운드, 모따기 등으로 좌표값에 따라 모델을 생성한 경우를 보여준다. 먼저, 출발 원점을 정의하며 정의가 없는 경우는 Default 값으로 그림과 같은 위치에 원점이 정의된다.

Table 2 는 선, 사선, 라운드 및 모따기에 대한 데이터베이스 형식을 보여주며, 여러 형식의 경우는

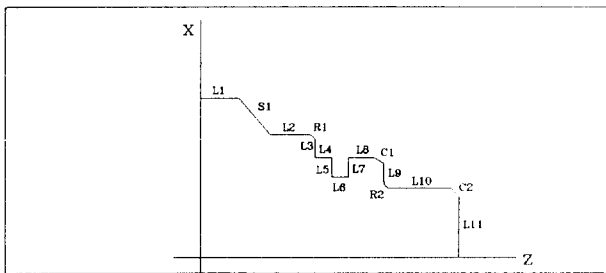


Fig. 6 Components of Part.

사용자의 편리한 형식대로 사용할 수 있다. Fig. 6 과 같은 체계의 도형정의를 모델형태의 파악 및 공정입력을 편리하게 하기 위한 Tree 구조 방법론으로 활용한 것이며, 모델의 History 등의 활용에 편리성이 있기 때문에 사용하였다.<sup>(8)</sup> 그림에서와 같이 원점은 x 축 상단에 설정한 경우를 나타내며 그에 따른 모델성분을 보여준다.

Table 2. Defined Components of Part

Definition	Coordinate System				Remarks
L1	x1	y1	x2	y2	
S1	x2	y2	d1	x1	y1
L2	x1	y1	x2	y2	
R1	x1	y2	r1	x1	y1
L3	x1	y1	x2	y2	
L4	x1	y1	x2	y2	
L5	x1	y1	x2	y2	
L6	x1	y1	x2	y2	
L7	x1	y1	x2	y2	
L8	x1	y1	x2	y2	
C1	x2	y2	c1	x1	y1
L9	x1	y1	x2	y2	
R2	x2	y2	r2	x1	y1
L10	x1	y1	x2	y2	
C2	x2	y2	c2	x1	y1
L11	x1	y1	x2	y2	

Fig. 7 은 모델의 형태파악이 용이한 특징에 대하여 보여주며, Solid 모델링의 데이터베이스와 근접화시켜 후에 유용하게 데이터를 활용할 수 있도록 하였다. Fig. 7 은 원점을 x 축에 설정한 경우를 보여준다. 현재방향의 같은 방향의 가치는 같은 형태의 도형정의를 의미하여 현재방향의 반대방향의 가치는 다른 형태의 도형정의를 나타낸다. 모델요소의 차별화에 편리성을 가지고 있으며, 공정입력에 용이한 데이터베이스를 형성할 수 있다. Fig. 8 은 원점을 z 축에 설정한 경우를 보여준다. 처음과 끝부분 Tree 구조의 형태가 교체되었음을 알 수 있다.

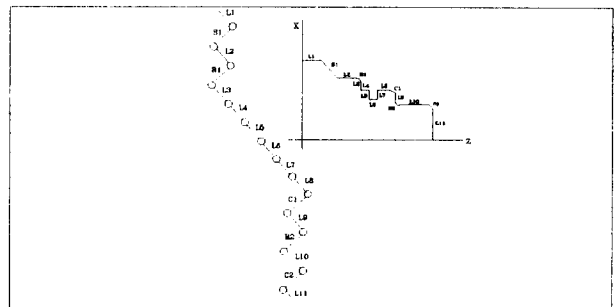


Fig. 7 Tree Structures for Design Processing in Zero Setting of X Axis.

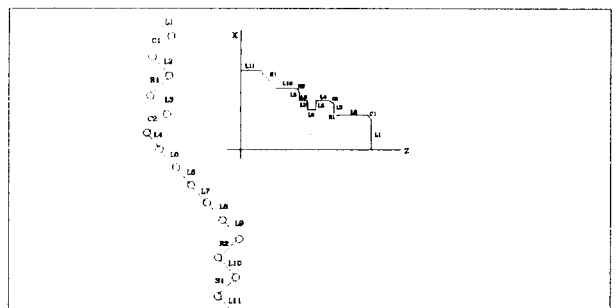


Fig. 8 Tree Structures for Design Processing in Zero Setting of Z Axis.

Fig. 9 는 가공물의 윤곽을 정의하기 위한 흐름도를 나타내며, 윤곽정의는 도형의 기본요소<sup>(9)</sup>인 선, 사선, 라운드, 모따기 등의 좌표값을 통하여 정의된다. 그림과 같이 먼저 가공물의 원점을 설정하고 설정된 원점으로부터 각 도형요소의 좌표값이 정의된다.

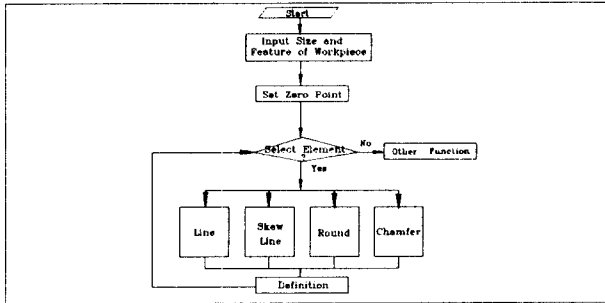


Fig. 9 Flowchart of Contour Definition.

#### 4.2 공정입력<sup>(2,10)</sup> 및 작업설계

설계와 생산을 결부시키는 생산 소프트웨어에 대하여 중요한 위치를 차지하고 있지만, 취급 정보량이 많고 그의 결정기구에 경험적인 부분이 포함되는 등의 개발상 문제를 내포하고 있다. 여기서는 기계가공에 한정하여 공정입력의 역할, 처리에 대하여 서술한다. 공정설계에 대한 취급정보는 Fig. 10 에 나타내고 있으며, 처리의 흐름은 Fig. 11 에서 보여준다.

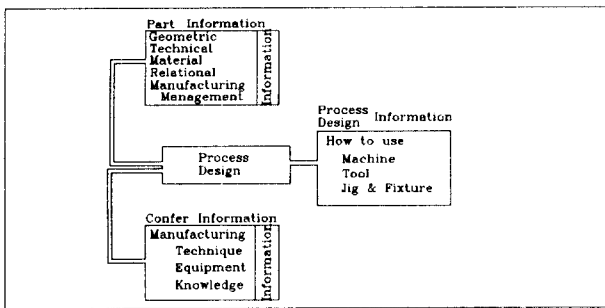


Fig. 10 Information of Process Design.

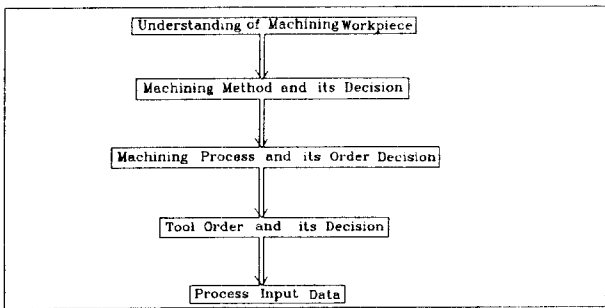


Fig. 11 Flowchart of Process Input.

Fig. 10과 같이 세부적인 항목은 다음과 같다. 부품 및 제품정보 부품의 완성형상을 표시하는 기하형상정보가 있고, 기술정보는 마무리정도, 표면

조도, 촌법 등을 표시하며 재질, 소재의 처리와 같은 소재의 가공법을 표시하는 소재정보, 조립한상대를 추정할 수 있는 부품접속정보 그리고 납기, 생산요구량, 생산 Lot 량 등으로 알려진 생산관리 정보가 있다.

참조정보 홈, 구멍과 같은 가공형상특징과 가공방법과의 대응관계 그리고 가공조건과 공구의 관계 등의 생산기술정보와 소유하고 있는 공작기계, 설비 공구 등의 능력, 제원에 관계된 생산설비정보 그리고 설비 등을 효율적으로 사용하기 위한 Know-How 와 부품 및 제품 정도향상을 위한 Know-How 등의 생산지식정보가 있다.

공정입력정보 직절한 공작기계와 그의 순서, 적절한 공구와 그의 사용순서 그리고 적절한 치공구의 사용순서가 있다. 본 연구에서는 공정입력정보에 대한 사항만을 개발 프로그래밍에 도입할 예정이다.

Fig. 11 의 세부적인 내용은 다음과 같다.

가공대상과 그의 이해 Modeller를 이용한 부품구성표가 기재되어 있는 정보와 이해에서 도형정의와 구성 및 다른 도형과의 접속관계의 정보를 이해하고 형상특징을 인식하는 것이다.

가공방법과 그의 순서결정 Fig. 12 는 황삭가공의 방법에 대한 결정을 위하여 형상을 분할한 예이다.<sup>(9)</sup> 공구의 조건과 절삭조건의 고려를 통한 결정이다. (a) 경우보다는 (b) 와 (c) 가 효율적이다.

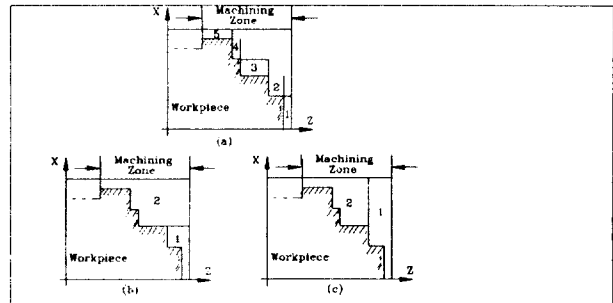


Fig. 12 Decision of Method for Rough Machining.

공구후보의 결정 가공면, 가공형상특징은 절삭운동과 이송운동에 대하여 공구의 Nose 형상과의 조합을 통하여 생성되며, 이에 따른 상세한 절삭조건 결정과 공구의 특정한 작업설계에 대하여 사용가능한 공구의 후보가 선택되어진다.

Fig. 13 은 가공되어질 도형정의의를 위한 메뉴를 나타내고 있으며 그림과 같이 6 가지 기본요소로 구성되어 있다.

Fig. 14 는 공정입력모델 중 축가공의 경우에 대한 6 가지 입력모델을 나타낸다. 그때의 공구는 일반적인 공구모델을 선택하는 과정을 보여준다.

Fig. 15 는 공정입력모델 중 나사가공의 4 가지 입력모델과 외경, 내경에 대한 입력모델을 나타내고 있다. 이때의 공구는 2 가지 형태이며, 1 줄 나사의 경우를 나타내고 있다.

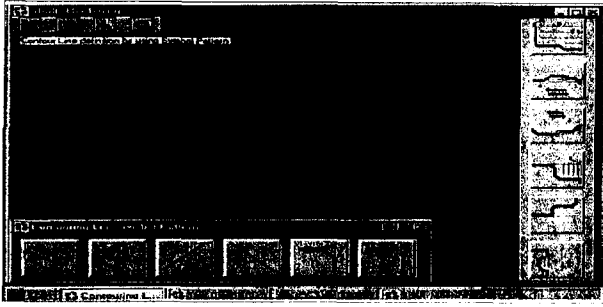


Fig. 13 Fundamental Elements for Feature Definition.

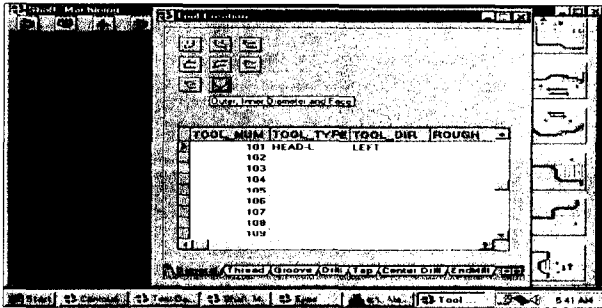


Fig. 14 Process Input of Shaft Machining.

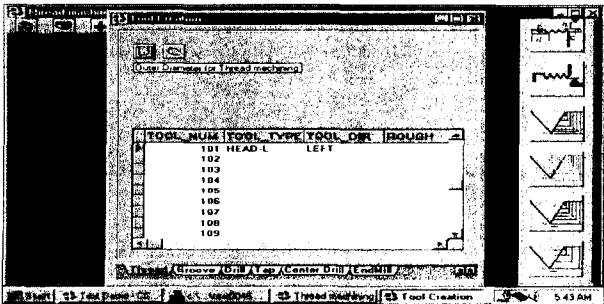


Fig. 15 Process Input of Thread Cutting.

Fig. 16 은 공정입력모델 중 코너가공에 대한 입력모델이다.

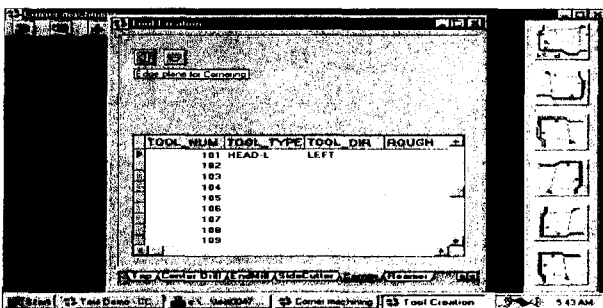


Fig. 16 Process Input of Corner Machining.

Fig. 17 은 홀가공의 공정입력모델을 나타내고 있다. 센터링, 리머와의 그밖의 입력모델을 오른쪽 버튼 (Button) 에 나타내었다.

## 5. 공구계획 및 검증

### 5.1 공구계획

Fig. 7, 8 과 같은 가공물의 형상을 Fig. 9 의

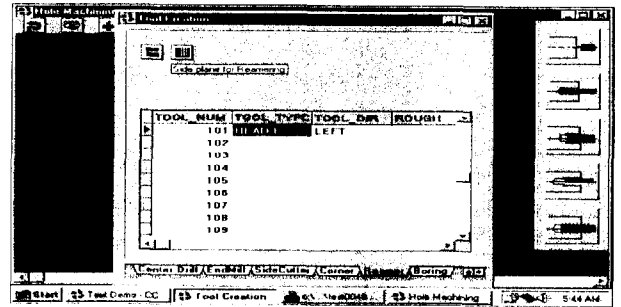


Fig. 17 Process Input of Hole Machining.

도형정의 흐름도를 통하여 최종 가공물의 형상을 정의한 후, Fig. 11 의 공정입력을 통하여 공구계획 (Tool Path) 을 위한 전체적인 알고리즘은 다음 Fig. 18 과 같다.

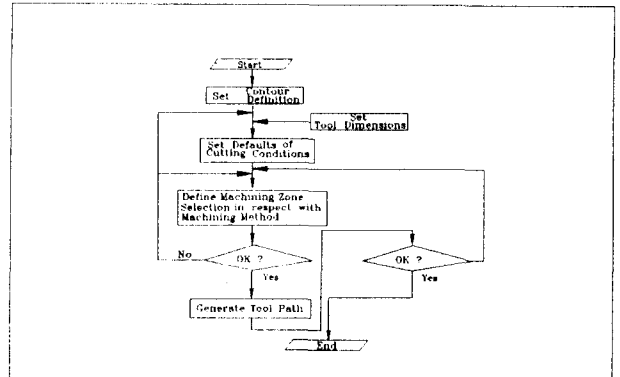


Fig. 18 Algorithm for Tool Path.

Fig. 18 과 같이 공구계획의 만족스럽지 못하면 가공 영역의 조건을 수정할 수 있도록 유연성을 부여하였다.

실제가공을 위한 공구계획은 MMI (Man Machine Interface) 의 API (Application Programmable Interface) 라이브러리의 도움을 얻어 Fig. 19 와 같이 사용할 공구의 끝선단을 가공물의 기준면 즉, x 축에 평행한 면에 MPG (Manual Pulse Generator) 를 이용하여 공구와 가공물과의 좌표계 관계를 셋팅한다. 따라서 그림과 같은 순서에 의하여 공구와 가공물의 충돌을 방지하기 위한 수단이며, 또한 자동 프로그래밍을 효율적으로 사용하기 위한 방향으로 가공 작업설계의 정확성 높이는 작업을 삽입하여야 한다. 이것은 Shopfloor 의 Know-How 가 수반된 자동 프로그래밍 방법을 제안한 것이다.

### 5.2 가공 프로그램 검증

본 논문에서는 Wireframe 형태<sup>(11)</sup>로 수행된 것이며, 3 각법에 의한 보는 평면을 조절하고 Fig. 18,

19 및 20 과 같은 방법을 통하여 수행될 것이다.

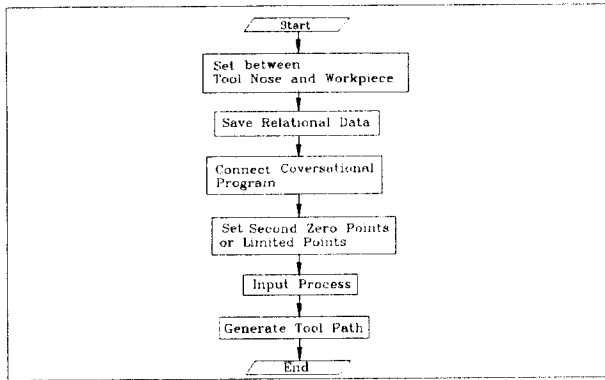


Fig. 19 Practical Flowchart for Tool Path.

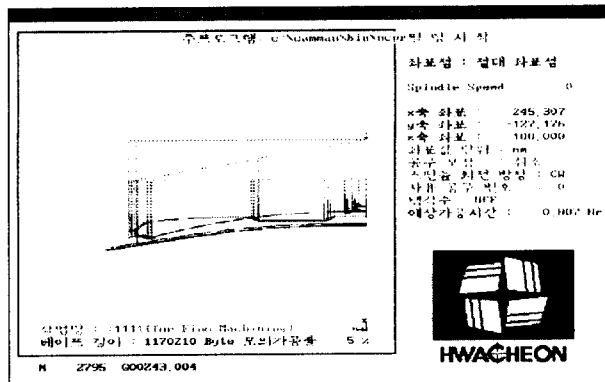


Fig. 20 Graphic Simulation by Wireframe<sup>(11)</sup>.

## 6. 결론

선반용 대화형 프로그램 개발을 통하여 다음과 같은 결론을 얻었다.

1. CAD / CAM 의 역사와 관계된 연구를 Review 하였다.
2. Tree 구조 방법론의 도형정의를 통하여 모델의 History 및 공정입력 등의 응용성을 높였다.
3. 작업설계를 통하여 작업장에서 효율적으로 사용할 수 있는 대화형 프로그램 알고리즘을 제시하였다.
4. 공구레직 및 경로검증을 위하여 알고리즘을 설계 하였다.
5. 그래픽 기반형 프로그램을 통하여 친숙한 유지 인터페이스를 제안하였다.

## 후기

이 논문은 1995년도 공업기반기술사업 중기저점 과제중 가공 Programming 환경 및 Monitoring 기술

(공고번호 951-60-4) 의 연구비 지원으로 수행되었으며 이에 관계자 여러분께 감사드립니다.

## 참고문헌

1. 강성균, 이지석, 최종률, 1995, "대화형의 그래픽을 이용한 선삭용 고기능 작업장 프로그래밍 시스템", 한국정밀공학회, 추계학술대회 논문집, pp.707 - 712.
2. 精密共學會 編, 1991, 生産ソフトウェアシステム, オーム, pp. 89 - 106.
3. 최병규, 1989, Cam 시스템과 CNC 절삭가공, 청문각.
4. Yoram Koren, 1983, Computer Control of Manufacturing Systems, McGraw-Hill International editions.
5. Kristin Kelley, 1994, "Getting Beyond G-Code", Modern Machine Shop Magazine, Gardner Publications, Inc.
6. Chris McMahon, Jimmie Browne, 1993, CAD CAM from principles to practice, Addison Wesley Publishing.
7. Masao Yamamoto, Hirokazu Isobe, Kazuya Yamazaki, 1987, "マシンセンタ 用 對話形 MC とまと", 機械と工具, Vol. 12, pp. 123 - 127.
8. Korea Graphics Technologies, 1996, Exploring I-DEAS Design, Korea Graphics Technologies Co., pp. 1 - 26.
9. FANUC , 199, FANUC Super Cap Version 1, FANUC LTD., pp. 2-34 - 2-80.
10. Roger S. Pressman, John E. Williams, 1977, Numerical Control & Computer Aided Manufacturing, John Wiley & Sons, Inc.
11. 신동수, 1995, CAMMAN User's Manual, 화천기공, pp. 21 - 24.