

선반용 STEP-NC 파트 프로그래밍 및 XML의 채용 Part programming with STEP-NC turning data model and XML adaptation

정대혁, 이병언, 서석환

포항공과대학교 산업공학과 STEP-NC/e-Manufacturing 연구실

Abstract

분산 제조 시스템간의 협업을 가능케 하는 e-Manufacturing paradigm 의 성공을 위해서는 CAD-CAM-CNC chain 의 제조 분야뿐만 아니라 SCM, ERP, CRM 등의 상위 시스템과의 제품 및 가공 정보의 원활한 흐름이 필수적이다. STEP 과 STEP-NC 는 이를 해결할 표준으로 자리매김 할 것으로 기대되고 있다. 본 논문은 STEP-NC 의 국내 promotion 차원에서 참고문헌 [1]의 밀링용 파트프로그래밍 소개에 이어: 1) 선반용 데이터 모델, 2) 선반용 파트 프로그램의 작성 방법, 3) XML 을 이용한 파트 프로그램의 표현 방법들을 소개하고, 4) STEP-NC 기반의 생산 환경 구축을 위한 방안을 기술하였다.

Key Words: STEP-NC, e-Manufacturing, Turning, XML, STEP-compliant CNC

1. 서론

IT 기술의 발전은 분산된 제조 시스템에서 협업을 가능케 하는 e-Manufacturing 패러다임을 가능케 하였다. e-Manufacturing 패러다임의 핵심은 제조 분야 (CAD-CAM-CNC chain)와 SCM, CRM, ERP 등의 상위 시스템들과의 정보 흐름을 원활히 하는 것으로서, STEP 과 STEP-NC 는 제품정보와 가공정보의 표준으로서 자리매김 할 것이다.

STEP 은 CAD, CAPP, CAM 간의 제품 정보 교환을 위한 국제 표준이며, STEP-NC 는 CAM 과 CNC 간의 인터페이스를 위해 새롭게 제정된 국제 표준이다. STEP-NC 의 경우 밀링, 선반,

EDM 등에 대해서 국제 표준으로의 제정이 완료되었거나 혹은 상당 부분 진행된 상태이다. 현재 국제적으로 STEP-NC 를 채용한 CAD, CAM, CNC, toolkit 등이 속속 개발되고 있으며, 국내에서는 본 연구실을 포함하여 STEP-NC 에 관한 기초 연구 및 개발연구를 상당한 수준으로 수행한 바 있으며, 이제는 이 기술을 국내에 보급하고 산업체 기술로 이관하는 일이 중요한 과제로 남아있다.

이에 본 논문에서는 STEP-NC의 이해와 국내 보급을 위해 선반 공정을 중심으로 (참고문헌 [1]에서는 밀링가공을 다루고 있음): 1) STEP-NC 데이터 모델을 소개하고, 2) 파트 프로그램 작성방식을 자세히 설명하고, 아울러 3) 새롭게 대두되고 있는 XML을 이용한 파트 프로그래밍 방법을 소개하였으며, 4) STEP-NC기반의 생산 환경 구축을 위한 방안을 기술하였다.

2. STEP-NC 선반용 데이터 모델 소개

ISO 14649 로 규격화된 STEP-NC 데이터 모델에 관한 전체적인 구조 및 밀링 부분은 문헌[1]에 자세히 기술되어 있다. 본 논문에서는 선반 관련 데이터 모델인 ISO 14649 Part 12 와 Part 121 에 대해서 먼저 데이터 모델을 설명한다.

ISO 14649 Part 12 [2] x-z 축의 기계 동작을 요하는 주로 회전체 형상의 가공을 대상으로 선반 기계 (밀링가공을 포함하는 복합기에 대해서도 확장되고 있는 추세 임)에서 수행되는 공정 및 가공정보를 정의하고 있다. 선반용 데이터 모델 역시 밀링 [1]과 동일한 구조를 가지며, 정보 내용 (information contents)에서 선반

고유의 회전 형상들을 특징 형상으로 정의하고 있으며, facing, threading, contouring 등의 선반을 통해 수행할 수 있는 공정과 이에 필요한 정보들을 정의하고 있다.

ISO 14649 Part 121 은 선반 가공에 사용되는 공구 규격, 즉 내외경 바이트, thread 바이트, groove 바이트, knurl 공구를 정의할 수 있는 데이터 모델을 정의한다. 이미 공구에 관해 다른 국제 표준 (ISO 1832 등)들이 존재하지만, Part 121 은 공구의 모든 요소를 정의하기 보다는 (즉 catalogue model 이 아님) 공구 경로 생성등과 같이 실질적인 사용에 필요한 정보 (requirement model)를 정의하고 있다.

3. 파트 프로그램의 작성

본 장에서는 선반용 STEP-NC 파트프로그램 작성방법을 Figure 1 의 형상을 예로 들어 설명한다. Figure 1 에서 빗금친 부분은 델타 볼륨으로서 제거되어야 할 볼륨이며, 이를 제거한 부분이 최종 형상이다. 델타 볼륨의 분해는 작업자 (혹은 파트프로그래머, 알고리즘)에 따라 다양하게 이루어질 수 있으며, Figure 1 은 그 일례로서 두 개의 revolved_flat, 1 개의 general_revolution, 1 개의 outer_diameter, 1 개의 round_hole, 2 개의 groove 형상들로 예제 형상을 구성하였다.

선반용 파트 프로그램도 기본적으로는 참고문헌[1]에서 제시하고 있는 밀링용 파트프로그램과 동일한 구조를 가지며, 다만 정보내용이 다를 뿐이다. 이하에서는 선반가공을 대상으로: 1) 특징 형상 정보, 2) 공정 정보, 3) 공구 정보, 4) 공정의 실행 순서, 5) 소재 정보를 기술하는 방식을 기술한다.

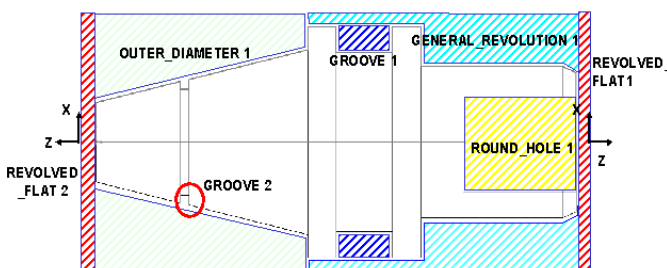


Figure 1. 디자인 파트와 델타 볼륨 분해 결과.

3.1 특징 형상 정보의 작성

STEP-NC 에서 특징 형상은 작업 단위인 workingstep 으로 제거해야 할 볼륨을 의미한다. STEP-NC 파트 프로그램에서의 특징 형상 정보는 G-Code 파트 프로그램과는 달리 가공하고자 하는 파트의 형상을 미리 알 수 있도록 해주며, operation 할당과 공구 경로 생성의 기본 정보로서의 역할을 수행한다.

선반 STEP-NC 데이터 모델에서는 선반 특징 형상으로 X-Z 축의 공구 움직임을 통해 제거 가능한 회전체 형상을 정의하고 있으며, revolved_flat, revolved_round, groove, knurl, outer_diameter, outer_diameter_to_shoulder, general_revolution 등을 정의하고 있다. 이들 각각의 특징 형상들은 형상을 정의하기 위한 정보들을 포함한다.

```
#10=REVOLVED_FLAT('REVOLVED FLAT 1',#1,(#22,#23),#172,
#176,21.0,#178);
#11=REVOLVED_FLAT('REVOLVED FLAT 2',#1,(#31,#32),
#183#187,12.0,#189);
#12=GENERAL_REVOLUTION('GENERAL_REVOLUTION 1',
#1,(#20,#21),#194,#198,21.0,#200);
#13=ROUND_HOLE('HOLE1 FLAT BOTTOM ',#1,
(#26,#27,#28),#207,#215,#216,$,#217);
#14=OUTER_DIAMETER('OUTER_DIAMETER 1',#1,(#29,#30),
#218,#222,#223,#224);
#15=GROOVE('GROOVE 1',#1,(#24,#25), #226,#230,35.0,#232);
#16=GROOVE('CUT_IN',#1,(#33),#236,#240,18.4,#242);
```

Figure 2. 선반 특징 형상 정보 작성 예.

```
#200=GENERAL_PROFILE($,#201);
#201=POLYLINE(",(#202,#203,#204,#205,#206));
#202=CARTESIAN_POINT(",(21.000,0.000, 0.000));
#203=CARTESIAN_POINT(",(23.000,0.000, 5.000));
#204=CARTESIAN_POINT(",(23.000,0.000, 55.000));
#205=CARTESIAN_POINT(",(35.000,0.000, 55.000));
#206=CARTESIAN_POINT(",(35.000,0.000, 95.000));
```

Figure 3. general_revolution 의 프로파일 정보 작성 예.

Figure 2 는 Figure 1 을 구성하는 특징 형상 정보를 파트 프로그램으로 작성한 것이다. 특히 Figure 3 은 이 중에서 general_revolution 정보를 구체적으로 정의한 것으로서, 파트 프로그램에서는 general_revolution 을 4 개의 직선 (polyline)으로 정의하고 있다.

3.2 공정 정보의 작성

operation은 주어진 특징 형상을 가공하기 위한 방법을 정의한 것으로 여기에는 가공 종류, 가공 전략, 접근/후퇴 전략, 가공 조건, 기계 동작, 필요 공구 등 가공 방법에 대한 모든 것을 정의한다. 선반용 STEP-NC 데이터 모델에서는 선반 공정들로 *contouring*, *facing*, *cutting_in*, *grooving*, *threading*, *knurling* 등과 이에 대한 황삭, 정삭등을 정의하고 있으며, 각 공정마다 필요한 정보를 정의하고 있다.

Figure 4는 Figure 2의 *general_revolution* (#21)을 가공하기 위한 operation으로서 *contouring_rough* entity(#20)를 이용하여 가공하는 것을 작성한 것이다. *contouring_rough* 공정에 포함되는 정보는 공구 후퇴 높이(30mm), 가공 시작 위치, 사용 공구, 가공 조건, 기계 동작, 공구 접근 전략, 공구 퇴출 전략, 가공 전략, 정삭 여유량(0.5mm) 정보들이다.

```
#20=CONTOURING_ROUGH($,$,'ROUGH GENERALREVOLUTION1',
30.000,$,#280,#61,#60,#130,#130,#131,0.5);
```

```
#21=CONTOURING_FINISH($,$,'FINISH GENERAL REVOLUTION1',
30.000,$,#280,#61,#60,#130,#130,#132,0.0);
```

Figure 4. 공정 정보의 작성 예.

3.2.1 접근/퇴각 및 가공 전략 정보의 작성

한편 각 operation에서 어떤 방법으로 가공할지에 대한 가공 전략은 *turning_machining_strategy*에 의해 정의가 된다. 선반 가공에서는 가공 전략으로 *unidirectional_turning*, *bidirectional_turning*, *contour_turning*, *groove_strategy*, *thread_strategy*등을 정의할 수 있다. Figure 5에서 가공 전략으로 *unidirectional_turing*(#131)을 사용하도록 정의하고 있으며, *overcut_length*, 절삭 깊이(3mm), 피드 변화량, *lift_height*(2mm), 피드 방향, *back path* 방향, *stepover*방향과 각 방향에 대한 피드 정보들을 정의할 수 있다. 만일 파트 프로그램에서 위에서 기술한 정보들이 정의되지 않은 경우에는 CNC에서 자체적으로 결정하게 된다. Figure 6은 *unidirectional_turning*에 의한 가공 전략을 설명한 그림이다.

또한 접근 경로와 후퇴 경로에 대해서도 접근/후퇴 전략이 필요하며, 이는 *approach_retract_strategy*에 의해 정의가 된다. 선반 가공을 위한 접근 및 퇴출 전략은 별도로 정의되어

있지 않으며, 밀링 공정과 동일하게 사용된다. Figure 5에서 접근/후퇴 전략으로 *plunge_ramp* (#130) entity를 정의하였다.

```
#130=PLUNGE_RAMP($,45.000);
#131=UNIDIRECTIONAL_TURNING($,$,(3.000),$,$,$,$,2.000,$,$);
#132=UNIDIRECTIONAL_TURNING($,$,(0.500),$,$,$,$,$,$);
#133=PLUNGE_RAMP($,30.000);
#134=PLUNGE_RAMP($,40.000);
#131=UNIDIRECTIONAL_TURNING($,.,(3.000),$,$,$,$,2.000,$,$);
```

Figure 5. 가공 전략 정보의 작성 예.

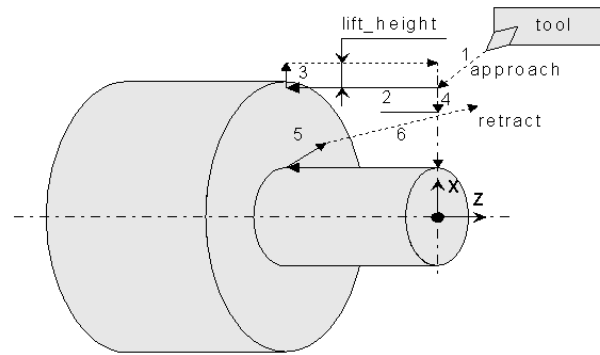


Figure 6. unidirectional_turning.

3.2.2 가공 조건 정보의 작성

한편 이송량(feed), spindle 회전수 등과 같은 가공 조건 정보는 *turning_technology* entity를 이용해 정의한다. 이송량은 *feedrate*나 *federate_per_revolution* 중의 하나로써 정의하며, 주속 일정 제어(*const_spindle_speed*)방식 혹은 절삭속도 일정 제어(*const_cutting_speed*)방식 중 하나로써 spindle 회전수를 정의한다. 그리고 spindle 과 축의 동기화, spindle 및 feed의 override 등에 대한 추가적인 정보도 정의된다.

Figure 7에서는 tool center point를 기준으로 회전당 0.3mm의 feed를 설정하였으며, 스핀들 스피드(#65)는 주속 일정 제어 방식으로 500RPM을 설정하였다. 그 외의 추가적인 정보는 주지 않았다.

그리고 가공 도중 수행되는 절삭유의 사용 여부, 사용 절삭유의 종류, 절삭유의 압력, 칩 제거 기능, 스핀들의 초기 방향, tail stock 과 방진구의 사용등과 같은 기계 동작은 *turning_machine_function*을 통해 정의한다. Figure 7에서는 절삭유를 사용하는 것으로 정의하였다.

```
#60=TURNING_MACHINE_FUNCTIONS(T,.,$,0,.,F,.,$,0,$,$);
#61=TURNING_TECHNOLOGY($,.,TCP,.,#62,0.300,.,F,.,F,.$);
#62=CONST_SPINDLE_SPEED(500);
```

Figure 7. 가공 조건 정보의 작성 예..

3.3 선반 공구 정보의 작성

G-Code 파트 프로그램에서는 공구 정보로서 turret상에서의 인덱스만을 알 수 있는데 반해, STEP-NC 파트 프로그램에서는 2장에서 설명한 바와 같이 공구 경로 생성을 위해 필요한 공구의 종류 및 형상에 관한 정보들을 제공한다. STEP-NC 데이터 모델에서는 선반 공구로서, general_turning_tool, grooving_tool, threading_tool, knurling_tool들을 정의하고 있으며, 각 공구 정보는 cutting edge angle, rake angle, clearance angle 등과 같은 인서트의 기하학 정보와 공구 홀더의 길이와 폭에 관한 정보들을 포함하고 있다.

Figure 8은 Figure 4의 contouring_rough(#20) 공정에서 공구 정보를 정의한 부분으로서, side cutting edge angle 이 25도, end cutting edge angle 이 5도, side clearance angle 이 7도, end clearance angle 이 3도, nose radius 가 0.5mm 인 인서트를 사용하며, 높이가 40mm, 폭이 60mm 이며, 인서트와 조립되었을 때 총 길이가 120mm 인 공구 홀더를 사용하는 것을 알 수 있다.

```
#280=TURNING_MACHINE_TOOL('',#281,(#283),120,40,$)
#281=GENERAL_TURNING_TOOL(#282,,LEFT,,40,60,,CW,);
#282=TOOL_DIMENSION($,$,$,$,25,5,7,3,5,0.5,$);
```

Figure 8. 선반 공구 정보의 작성 예.

3.4 workingstep 과 가공 순서 정보의 작성

STEP-NC 파트프로그램은 workingstep 이라는 작업 단위로 가공작업을 할당 한다. workingstep 의 정보요소는 그 작업단위에서 제거해야 할 볼륨 (feature), 이에 사용되는 작업 타입 (operation), 사용되는 공구 (tool)등 모든 정보요소를 다 포함한다. 가공 순서는 program_structure 라는 entity 를 이용하여 지정하는데, 순차적인 직선형 공정계획 (workplan) 뿐 만 아니라 무순서 실행(non_sequential)과 대안 실행(selective), 조건부 실행(if_statement), 반복 실행(while_statement)등의 비선형 공정계획도 지정할 수 있다

Figure 9는 Figure 1의 예제형상을 가공하기 위한 파트 프로그램에서 순서를 지정한 부분을 보이는데, 파트 프로그램은 project(#34)라

는 entity 로 시작된다. 파트 프로그램의 가공 순서는 project 정보 요소 중 workplan(#35)을 통해 정의된다. 여기서 선반 가공에서 발생하는 두 가지 셋업을 위해 두 개의 workplan(#36, #37)을 정의하였으며, #36의 경우 8개의 workingstep 을 순차적으로 수행하도록 정의 하였다. #36은 Figure 1의 우측 부분의 가공을 지령하는 것으로서 가공 순서를 살펴보면, Figure 1의 우측 끝부분에 대해서 facing 가공(#38, #39)을 수행하고, 외곽 부분에 대해 contouring 가공(#40, #41)을 수행하고, groove 부분에 대해 grooving 가공(#42, #43)을 수행하고, 구멍부분에 대해 drill 가공(#44)과 reaming 가공(#45)을 수행하도록 순서를 주고 있다.

```
#34=PROJECT('TURNING EXAMPLE 1',#35,(#1),$,$,$);
#35=WORKPLAN('MAIN WORKPLAN',(#36,#37),$,$2,$);
#36=WORKPLAN('WORK PLAN FOR SETUP1',(#38,#39,#40,#41,#42,#43,#44,#45),$,$,$);
#37=WORKPLAN('WORK PLAN FOR SETUP2',(#47,#48,#49,#50,#51),$,$4,$);
#38=MACHINING_WORKINGSTEP('WS ROUGH CIRCULAR_FACE 2',#56,#11,#22,$);
#39=MACHINING_WORKINGSTEP('WS FINISH CIRCULAR_FACE 2',#56,#11,#23,$);
#40=MACHINING_WORKINGSTEP('WS ROUGH GENERAL_REVOLUTION 1',#56,#12,#20,$);
#41=MACHINING_WORKINGSTEP('WS FINISH GENERAL_REVOLUTION 1',#56,#12,#21,$);
#42=MACHINING_WORKINGSTEP('WS ROUGH GROOVE 1',#56,#15,#24,$);
#43=MACHINING_WORKINGSTEP('WS FINISH GROOVE 1',#56,#15,#25,$);
```

Figure 9. 실행 순서 정보의 작성 예.

3.5 가공 소재 정보의 작성

STEP-NC 파트 프로그램에서 소재 정보는 workpiece 라는 entity 에 의해서 정의 된다. 기존의 G-code 에서는 소재 정보를 직접 표현하지 못하고, 단지 소재의 형상과 재질을 감안하여 가공 조건을 설정하지만, STEP-NC 파트 프로그램에서는 소재의 초기 형상과 가공 후 최종 형상, 재질 그리고 소재의 체결 위치 등 소재에 관련된 상세한 정보를 정의할 수 있다. Figure 10은 Figure 1의 예제 형상에 사용될 소재정보를 작성한 것이다. 여기서 소재의 재질을 'ST-50'이란 steel 이 사용되고, 초기 형상은 실린더 형태로서 길이는 175mm, 직경은 70mm 로 정의하였다.

```

#1=WORKPIECE('SIMPLE WORKPIECE',#2,0.01,$,$,$,());
#2=MATERIAL('ST-50','STEEL',(3));
#3=PROPERTY_PARAMETER('E=200000N/M2');
#4=RIGHT_CIRCULAR_CYLINDER('WORKPIECE PIECE', #5,
175.0, 35.0);
#5=AXIS1_PLACEMENT('WORKPIECE PIECE PLACEMENT',#6,#7);
#6=CARTESIAN_POINT('WORKPIECE PIECE: LOCATION ',
(0.000,0.000,0.000));
#7=DIRECTION(' AXIS ',(0.000,0.000,1.000));

```

Figure 10. 소재 정보 작성 예.

4. 컴퓨터 지원 파트 프로그래밍 시스템

3 장의 파트프로그래밍 과정은 마치 수작업으로 작성하는 것처럼 설명하였으나, 실제 STEP-NC 파트프로그래밍을 수작업으로 하기는 거의 불가능하다. 따라서 컴퓨터지원 시스템에 의한 방식이 필수적이다. 밀링을 지원하는 STEP-NC 프로그래밍 시스템은 미국[5], 유럽의 것[6]과 한국의 것 (Korea STEP-NC[7])등이 존재하나, 선반의 경우 본 연구팀에서 2003년에 기본형을 개발한 바 있는 TurnSTEP이 최초의 것이다[8]. TurnSTEP의 CGS 모듈(파트 프로그램 생성 모듈)은 수작업으로 수행한 아래의 과정을 알고리즘화 하여 자동화 시킨 것이다.

(1) 특징 형상 인식 단계: 입력 받은 디자인 정보로부터 델타 볼륨을 분해하고, 각 델타 볼륨에 해당하는 선반 특징 형상을 할당한다.

(2) 공정 할당 단계: 인식된 특징 형상을 생성하기 위해 필요한 공정 및 황삭, 정삭등을 할당한다.

(3) 각 공정에 공구 할당 단계: 각 공정 수행에 있어 적절한 공구를 할당한다.

(4) 각 공정에 가공 조건 설정 단계: 가공 조건 정보를 설정하고, 선택된 공구등을 고려하여 가공 조건을 설정한다.

(5) workingstep 생성 및 실행 순서 결정: 위 다섯 단계를 거쳐 생성된 공정들과 특징형상을 통해 machining_workingstep을 정의하고, 이들의 실행 순서를 결정하여, 공정 계획 정보를 생성한다.

(6) 파트 프로그램 생성 단계: 생성된 공정 계획 정보를 파트 프로그램으로 변환한다.

Figure 11에서 13은 Figure 1의 예제형상에 대하여 TurnSTEP이 파트프로그램을 생성하는 과정을 보인 것이다. Figure 11은 AP203 형상 파일을 입력으로 받아 델타 볼륨 분해를 수행한 결과로서, Figure 1에서 보이는 델타 볼륨 분해 및 특징 형상 인식과 동일하게 이루어졌으며, 파트 프로그램상에서의 특징 형상 정보

도 3.1 절의 Figure 2와 동일하게 출력되었음을 알 수 있다.

Figure 12는 인식된 선반 특징 형상의 가공순서가 자동으로 부여된 결과이다. 전통적으로 사용되어 온 직선형 공정계획, 즉, 한 작업이 끝나고 다른 작업을 부여하는 직선형 공정계획 방식은 프로그램의 편의를 위한 것이지만 실제로 그 순서가 반드시 지켜지지 않아도, 혹은 현장 사정 상 그 보다는 다른 순서를 따르는 것이 더 적합한 경우가 많이 있다. 예컨대, round_hole feature와 general_revolution feature는 딱히 어느 것을 먼저 하도록 지정할 필요가 없다. 뿐만 아니라, 선반공작기계가 복합화 (2개 이상의 주축, 2개 이상의 터렛), 다기능화 (C축 가공 가능)됨에 따라 동시가공, 병렬가공 등이 가능해지며, 이 경우 순서를 임의로 지정해주는 종래의 직선형 공정계획 방식은 비합리적이다. STEP-NC는 이를 처리할 수 있는 비선형 공정계획 (nonlinear process planning) 기능을 허용하며, TurnSTEP 시스템은 이 기능을 구현한 세계최초의 시스템이다. 즉, Figure 11의 좌측 창에서는 workingstep의 순서가 AND, Parallel 등의 노드 타입의 조합으로 작성된 것을 보이고 있으며, 3장에서 설명한 파트 프로그램과 동일한 공정이 생성되었음을 알 수 있다.

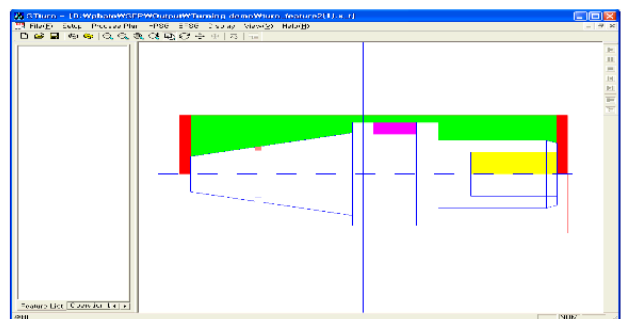


Figure 11. 델타 볼륨 분해 결과.

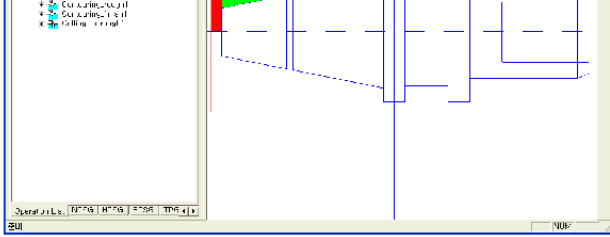


Figure 12. 공정 계획 정보 생성 결과

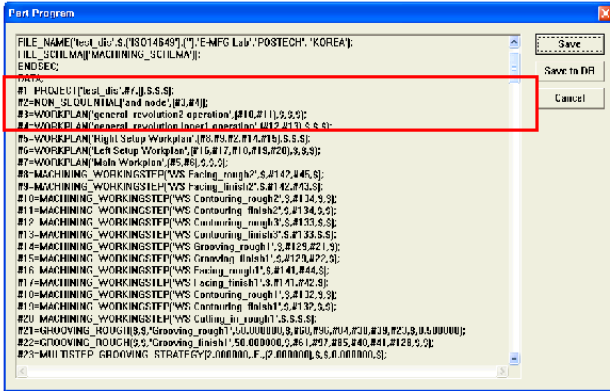


Figure 13. 파트 프로그램 생성 결과.

이들 각 공정들에 대해서 TurnSTEP 내부의 자동화 알고리즘과 사용자 지정 정보에 의거 3.2 절, 3.3 절, 3.4 절에서 설명한 가공 전략, 가공 조건, 공구, 가공 순서등의 정보들이 자동으로 설정 된다. 이들 정보를 바탕으로 하여 Figure 13 과 같은 물리 화일 형태의 파트프로그램이 자동으로 생성되었다. 주기할 것은 Figure 13 의 박스안에 공정계획 부분은 non_sequential 을 이용한 무순서 실행으로 출력된 것이다.

5. XML 을 채용하는 파트프로그램

5.1 ISO 10303 Part 28 에 의한 XML 방식

3 장에서는 파트프로그램을 STEP 에서 전통적으로 채용하고 있는 물리 화일 (ISO 10303 Part 21 규격의 physical file)의 형태로 작성하였다. Part 21 은 EXPRESS 언어를 표현하기 위한 수단으로서 하나의 분야에서 필요한 정보를 모델링하고 표현하는 데는 좋은 방법론이나 인터넷 상에서 서로 정보를 교환하는 데는 적합하지 않은 단점이 있다.

STEP 과 STEP-NC 가 e-Manufacturing 환경에서 사용되기 위해서는 인터넷을 통해 관련 정보가 자유롭게 흘러가야 하며, 이러한 목적에 비추어 part 21 은 한계를 갖고 있다. 이러한 part 21 의 한계를 극복하기 위해 현재 인터넷 상에서 정보 교환의 표준으로 각광을 받고 있는 XML(eXtensible Markup Language)을 이용하여 파트 프로그램을 표현하려는 움직임이 있으

며, 이의 일환으로 ISO 10303 Part 28 (XML representations of EXPRESS schemas and data)이 새로운 국제 표준으로 제정 중에 있다[9].

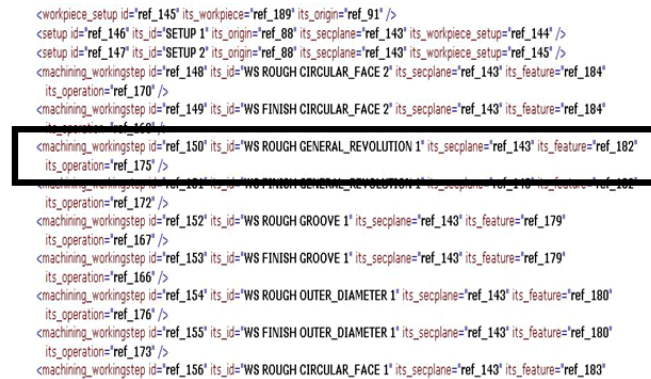


Figure 14. XML 형식의 파트 프로그램 예.

현재 제정되고 있는 ISO 10303 Part 28 에서는 크게 3 가지 방법으로 EXPRESS data 와 XML data 의 변환을 정의하고 있다. 첫 번째 방법은 Late-Bound markup declarations 방법으로 XML 의 instance 에 해당하는 EXPRESS 정보를 모두 표현하는 것이다. 두 번째 방법과 세 번째 방법은 각각 EXPRESS-typed Early Binding (ETEB) 방식과 Object Serialization Early Binding (OSEB) 방식으로 불리는데 앞서의 Late-bound 방식과 달리 EXPRESS Schema 에 해당하는 XML schema 를 생성하고 이에 기초하여 XML instance 를 생성하는 방식이다. 두 방법에서의 차이는 ETEB 방식에서는 EXPRESS 의 entity 와 attribute 모두 XML 의 element 로 표현하는데 반해 OSEB 방식에서는 EXPRESS 의 entity 는 XML 의 element 로, 각 entity 의 attribute 는 XML 의 attribute 로 표현하는 차이점이 있다. Figure 14 는 STEP part 21 physical file 로 표현된 파트 프로그램을 OSEB 방식을 이용한 XML instance 로 변환한 것을 보여주는 것으로, 3.4 절의 Figure 9 에서 설명한 machining_workingstep 들을 정의한 부분을 XML 형태로 변화한 결과이다. 특히 Figure 14 의 박스안에 표시된 element 는 Figure 9 의 #40 의 machining_workingstep 을 변환시킨 것으로, 표현 방식에 상관없이 동일한 속성값의 정보를 갖는다.

5.2 CEB에 의한 XML 방식 (XDMSTEP)

STEP-NC 데이터 모델은 ARM (Application

Reference Model, ISO 14649)과 AIM (Application Interpreted Model, ISO 10303 AP 238)의 두 개가 존재한다. 최근 들어 이들 두 개의 모델을 병행하여 사용함으로써 정보의 내용과 구성상 효율성을 기하고자 하는 시도로서 새로운 XML 방식이 대두되고 있다.

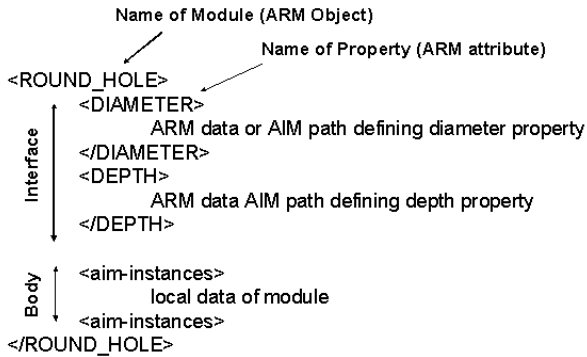


Figure 15. STEP-NC XML 파일 구조 예 [11]

이 방식은 2004년 3월 초에 미국 워싱턴에서 개최된 ISO TC184/SC1 전문가 회의에서 미국의 STEPTools 사에 의해 제안되었으며, 세계적인 공감대가 형성되었다. XDMSTEP (XML Data Modules for STEP[11])으로 명명된 이 방식에 따르면 하나의 XML 파일은 Figure 15에서와 같이 interface와 body 부분으로 나뉘어진다. Interface 부분은 module의 name과 property를 각각 ARM의 object와 attribute로 정의한다. 그리고 body에서는 ARM 정보에 해당하는 AIM instance들을 정의한다. 각각의 XML 파일은 interface를 통해 다른 XML 파일들을 reference할 수 있으며, 실제 데이터는 body에 저장된다. 한편 제안된 방법의 또 다른 특징은 파트프로그램의 모듈화이다. 현재의 파트프로그램은 하나의 파트 프로그램에 모든 정보를 담고 있다. 하지만 제안된 방법은 하나의 파트 프로그램을 machining_workingstep, machining_feature, tool, machining_operation 등의 여러 모듈로 나뉘어서 각각을 XML 파일로 저장한다.

5.3 XDMSTEP에 기반한 파트 프로그래밍

5.2절에서 설명한 방식을 채용하게 되면 파트프로그램을 인터넷 환경에서 제조 공정 정보를 분산시켜 저장하거나 혹은 전문 사이트로부터 받아 오므로서 다양한 시나리오를 통해 파트 프로그램을 생성할 수 있다. Figure 16은

STEPTools사에서 제안한 새로운 XML 방식에 의한 파트 프로그램 생성 시나리오이다.

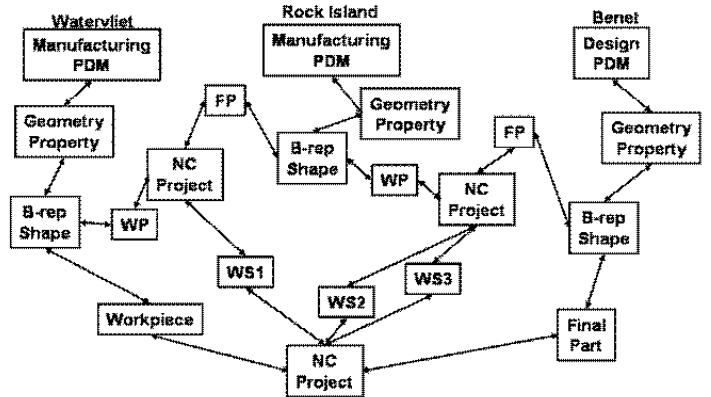


Figure 16. XML 방식의 파트 프로그램 생성 시나리오 [11]

지금까지의 파트 프로그램 생성 방식은 shop floor에서 파트 프로그램 생성에 필요한 모든 정보, 예를 들면 공구 정보나 가공 조건 정보 등을 직접 가지고 있어야 했다. 그러나 제안된 방법에 의하면 공구 정보는 공구 메이커 회사에, 가공 조건 정보는 가공 조건 전문 회사에 제안한 방식을 따른 XML 형식으로 저장되어 있고, 파트 프로그램 생성자는 파트 프로그램 생성 시 각 회사의 DB에 접속해 원하는 정보를 검색하고, 이에 대한 link만 첨가하면 되는 것이다. 뿐만 아니라 다른 회사에서 생성한 기존의 파트 프로그램의 machining_workingstep 모듈들을 이용해 새로운 파트 프로그램을 생성할 수도 있다.

기존에는 CAD 파일을 해석하고 가공 조건을 설정하고 공구 경로를 생성하는 것이 중요한 연구 테마였다. 그러나 이와 같은 방식에서는 가공에 대한 직접적인 방법 보다는, XML을 이용해 원하는 정보를 효율적으로 검색하고, 정보를 주고 받는 infra의 구축 등에 대한 연구가 주요 테마가 된다. 그리고 공구 회사, 가공 전문 회사, 공정 계획 전문 회사 등 분산 환경에서 여러 생산 주체들 간의 협업을 통해 파트 프로그램을 생성할 수 있다. 또한 XML DB의 생성과 정보를 주고 받을 때의 보안 등에 관한 것도 중요한 문제가 된다.

6. 맺음말

본 논문에서는 선반 가공을 중심으로 STEP-NC 데이터 모델을 소개하고, 이를 이용한 파트 프로그램의 작성에 대해 자세히 살펴 보았다. 또한 현재 파트 프로그램 작성의 새로운 방법론으로 제시되고 있는 XML 을 이용한 파트 프로그램의 생성에 관해 살펴보았다.

이미 위에서 언급한 바와 같이 STEP-NC 는 단순한 CNC 의 인터페이스뿐만 아니라 e-manufacturing 환경 구축을 위한 필수 요소로서 기대되고 있다. 그러므로 STEP-NC 에 대한 이해를 바탕으로 이를 생산 환경에 적용하기 위해서는 각 분야에서 다음과 같은 노력이 경주 되어야 한다.

1) CNC 메이커: STEP-NC 를 이해하고, 스스로 STEP-NC 정보를 통해 기계 구조에 맞는 공구 경로를 생성하고 이를 집행하며, 더 나아가 STEP-NC 가 담고 있는 정보를 바탕으로 지적 자율 제어를 수행할 수 있는 제어기 개발.

2) CA 시스템 개발 업체: STEP-NC 는 방대한 정보로 인하여, 파트 프로그램을 작성하거나 혹은 이를 사용하기 위해서는 컴퓨터 지원 시스템이 필요하다. CA 시스템 개발 업체에서는 특징 형상 인식 기술, 가공 조건 할당 기술, 공구 경로 생성 기술, 비선형 공정 계획 정보 생성 기술등과 같은 CA 및 제조 분야의 다양한 기술들을 접목하여, STEP-NC 를 지원할 수 있는 형태로 컴퓨터 지원 시스템 개발.

3) 사용자 입장: STEP 및 STEP-NC 정보가 상위 시스템에서 생산 현장의 제어기까지 자유롭게 유통 될 수 있는 생산환경의 구축

4) MTB 입장: STEP-NC 의 인터페이스와 지적 자율 제어 기능을 수행할 수 있는 공작 기계 개발.

5) IT solution 업체: DB, Web service 등과 같이 STEP 과 STEP-NC 정보들이 자유롭게 전송되고, 관리될 수 있는 IT infra 구축.

7. 참고 문헌

- [1] 이병언, 정대혁, 서석환, "STEP-NC milling data model and part programming," 2004 춘계 IE/MS 학술대회 프로시딩, May 21, 2004.
- [2] International Standards Organization, ISO 14649, Part 12: Process Data for Turning. 2003.

- [3] International Standards Organization, ISO 14649, Part 12: Tools for Turning, 2003.
- [4] International Standards Organization, ISO 14649, Part 11: Process Data for Milling, 2003.
- [5] M. Hardwick, "US STEP-NC implementation," white paper presented in ISO TC184/SC4 Meeting, San Francisco, USA, June 2001.
- [6] F. Glantschnig, "STEP-NC is reality," white paper presented on ISO TC184/SC4 Meeting, Charleston, USA, October 2000.
- [7] S. H. Suh, "Korea STEP-NC," white paper presented on ISO TC184/SC4 Meeting, San Francisco, USA, June, 2001.
- [8] S. H. Suh, "STEP-NC Turning System under Development by POSTECH, Korea," white paper presented on ISO TC184/SC4 Meeting, Stuttgart, Germany, June, 2003.
- [9] International Standards Organization, ISO 10303 Part 28: Implementation methods: XML representations of EXPRESS schemas and data, 2002.
- [10] ISO TC 184/SC4 N535, Guidelines for the development and approval of STEP application protocols, 1998
- [11] Martin Hardwick, "XDMSTEP: XML Data Modules for STEP", ISO TC184/SC1/WG7 meeting, March 2004