

사용자를 위한 설치, 설정 그리고 운영 방법

번역: 박 상규

연락처: derek.sangkyu.park@gmail.com

Mach3 라이선스 구매관련: derek.sangkyu.park@gmail.com  
또는 cnc@aacnc.com

## Mach 3 의 사용 방법



번역상의 오류가 발견되어 수정을 원하신다면  
derek.sangkyu.park@gmail.com 으로 이메일 주세요.  
확인 후 수정하여 다시 올리겠습니다.

기타 Mach 3 프로그램에 관한 문의는 영문작성 시  
support@artofcnc.ca 로 연락 주시하시거나 국내의 여  
러 카페 동호회를 이용해 주세요.

이 번역물에 대한 소유권은 번역자에게 있으며 타인에  
게 영리의 목적으로 판매하는 것은 금지합니다. 다만  
번역물에 한해서 비 영리의 목적으로 타인에게 배포 할  
수 있습니다.

번역을 마무리하며 . . . . .

벌써 뿌려야 할 씨앗을 이제 뿌렸다. 타고난 농부가 아닌터에 씨앗 뿌리는 것에 서툴 수 밖에 없지만 어쨌든 뿌려졌다. 바꿔 말하면 원래 전문적으로 번역 하는 사람도 아니고 기계나 전자를 전공한 사람도 아니라 번역물을 읽으면서 “ 이게 무슨 말이지??” 라고 생각 하시는 분도 있지 않을까 싶다. 그런 경우 번역 내용을 비판 하기 보다는 번역자에게 꼭 수정할 내용을 건의해 주셔서 다른 후학에게 도움을 주었으면 하는 바램이다. 또 내용을 읽다 보면 실제 작업자의 장비와 관련이 없는 사항도 있을 것이고 평소 잘 모르고 지났던 내용이 있을 것이라 본다.

번역자에 대해 간단히 소개 하면 2009년 1월 현재 미국 Oregon 주 Portland 소재의 반도체 장비 제조회사에서 엔지니어로 재직하고 있다.

처음 집에서 CNC 라우터를 설계 하며 첫 걸음을 하던 중, 대중적인 머시닝 소프트웨어인 Mach3가 몇 년이 지나도 한글 매뉴얼이 없어, 답답해 하시는 분들을 지켜보며, 분명 나보다 훨 낫으신 분들이 있을텐데 아무도 나서지 않아, 부족 하지만 이 글쓰기가 시작을 하게 됐다.

한국의 정서상, 정품 소프트웨어 사용에 대한 인식이 약한 이유도 있고, 로컬 딜러가 상대적으로 가격이 싼 소프트웨어 몇 개 팔아서 번역작업에 비용을 들이긴 수치타산이 안 맞기 때문에 섣뚱 하려고 하는 사람이 지금까지 없던 것은 충분히 이해할 수 있는 대목이다. 이렇게 작업 하는 나 자신조차 몇 개월에 걸쳐 조금씩 시간을 쪼개 가면서 했지만 이것을 통한 직접적 보상은 없기 때문에 사실 동기 부여가 상대적으로 약한 것도 사실 이었다.

Mach 1,2,3 를 만든 Artsoft 사의 매뉴얼 및 포럼을 보고 느끼는 것은, 타국어로 된 매뉴얼이 하나씩 올라 오고 있지만 한국어는 아직까지 없었다. 그래서 시작을 한 이유도 있었고, Artsoft 사에 모르는 부분을 직접 배우려고 연락 하며 지내던 중 우연히 소프트웨어 판매에 대한 딜러쉽을 갖게 되어 내가 뭔가 도움이 될 수 있을 거 같아 시작을 한 이유도 있다. (말한대로 번역 해서 올렸다고 금전적인 보상이 있는 것은 아니다.)

나 자신도 이 분야 첫 걸음을 들이면서 도움을 많이 받았다. 이젠 조금이나마 도움 이 됐으면 하는 것이 일차적 바램이다.

Mach2나 Mach3는 라이선스 파일을 공유 할 수 있다. 그러므로 무료로 업그레이드가 가능하다. 그리고 이 매뉴얼은 Mach2와 대부분 공유 할 수 있다. Mach2나 3에는 마법사 기능이 있는데 사각형, 원 포켓팅, 드릴링 등등 비교적 단순한 파트를 만드는 작업에, 캠 소프트웨어의 사용을 거치지 않고 직접 작업 할 수 있는, 생각 보다 훨씬 강력하고 유용한 기능이 있어 사용을 추천해 주고 싶다. 한가지, 마법사의 모든 기능을 활용 하려면 추가의 라이선스를 구매 해야 한다는 것이 흠이긴 하다.

다시 말씀 드리지만, 내용물의 잘못된 부분의 수정을 원하신다면 [dererk.sangkyu.park@gmail.com](mailto:dererk.sangkyu.park@gmail.com) 으로 연락 주시기 바라며 정품 구매를 원하신다면 위의 같은 이메일 주소로 연락 주셔도 되고 한국의 로컬 파트너인 [cnc@aacnc.com](http://cnc@aacnc.com) 으로 연락 주시면 되겠다. 추후 사항은 상호 연락을 통해서 이루어 질 것이다.

회색으로 된 10장의 일부는 번역자의 이해 부족으로(전반적으로 그렇지만) 혹은 Mach3 소프트웨어를 운영함에 있어 반드시 알아야 하는 내용이 아니기에 번역을 생략 했다. 이점 양해 바라며 필요하다고 느끼시면 연락 바란다.

부디 이 매뉴얼을 읽으시는 모든 분들에게 조금이나마 도움이 되기를 진심으로 바라며 이 글을 마친다.

2009년 번역자 박상규 올림.

# 목차

1. 서론 .....	12
2. CNC 머시닝시스템 소개 .....	14
2.1 머시닝 시스템의 각 부분 .....	14
2.2 Mach 3의 적용 .....	15
3. Mach 3 장비 컨트롤 소프트웨어 개요 .....	16
3.1 설치.....	16
3.1.1 다운로드 .....	16
3.1.2 설치 .....	16
3.1.3 윈도우스 재시작 .....	17
3.1.4 바탕화면 아이콘.....	17
3.1.5 설치한 프로그램 테스트 하기.....	18
3.1.6 Mach3 프로그램 충돌 시 드라이브 테스트 하기.....	19
3.1.7 드라이버의 수동설치와 제거 .....	19
3.2 표시 화면 .....	19
3.2.1 각 화면의 내용별 항목 .....	20
3.2.2 단축키와 버튼의 사용법 .....	20
3.2.3 DRO에 데이터 입력 하기.....	21
3.3 조그 사용법 .....	21
3.4 매뉴얼 데이터 입력(MDI) 과 티칭 .....	22
3.4.1 MDI .....	22
3.4.2 티칭.....	22
3.5 마법사 기능 - CAM 소프트웨어 없이 할 수 있는 여러 작업 .....	23
3.6 G 코드 프로그램 실행하기 .....	25
3.7 툴 패스(Tool Path) 표시화면 .....	26
3.7.1 툴 패스 보기 .....	26
3.7.2 툴 패스 화면 확대, 축소 그리고 끌여 당기기 .....	26
3.8 기타 그 밖의 화면 .....	26
4 하드웨어의 문제 및 장비와의 연결 .....	28
4.1 중요 안전사항 .....	28
4.2 Mach 3로 할 수 있는 작업.....	28
4.3 EStop 컨트롤 .....	29
4.4 PC의 페러렐 포트(프린터 포트, 병렬 포트) .....	30
4.4.1 페러렐 포트의 역사 .....	30
4.4.2 논리 신호 .....	30
4.4.3 전기적 노이즈 와 보드의 과열로 인한 손상 .....	31
4.5 축 드라이브 옵션(모터의 선택) .....	32
4.5.1 스텝모터와 서보모터 .....	32
4.5.2 축의 움직임에 필요한 요소 계산하기 .....	33
4.5.3 Step 과 Dir 신호의 작동 원리 .....	34
4.6 리밋 스위치, 홈 스위치 .....	35

4.6.1	설정을 위한 방법	35
4.6.2	스위치	35
4.6.3	스위치 의 설치 위치	36
4.6.4	Mach3 상에서 공유된 스위치의 사용방법	37
4.6.5	기준점 정하기	37
4.6.6	기타 홈, 리밋 스위치옵션과 힌트	38
4.7	스핀들 컨트롤	38
4.8	냉매제(coolant)	40
4.9	나이프(knife) 방향 컨트롤	40
4.10	탐침 스케너(digitize probe)	40
4.11	선형 글레스 스케일 인코더	40
4.12	스핀들 인덱스 펄스	41
4.13	펌프 충전 - 펄스 모니터	42
4.14	기타 다른 기능	42
5.	Mach 3을 장비와 드라이버에 맞게 설정 하기	43
5.1	설정에 관한 전략	43
5.2	초기 설정	43
5.2.1	포트의 주소 설정	43
5.2.2	엔진 주파수 정의	44
5.2.3	기타 특수 기능 정의	44
5.3	사용하게 될 입출력 신호의 정의	44
5.3.1	축 그리고 스핀들 출력 신호 설정	44
5.3.2	입력 신호 설정	45
5.3.3	에물레이션 입력 신호	46
5.3.4	출력 신호	47
5.3.5	인코더 입력 설정	47
5.3.5.1	인코더	47
5.3.5.2	MPG	48
5.3.6	스핀들의 설정	48
5.3.6.1	냉매제의 제어	48
5.3.6.2	스핀들 릴레이 제어	48
5.3.6.3	모터 제어	48
5.3.6.4	Modbus 스핀들 제어	49
5.3.6.5	일반 파라미터	49
5.3.6.6	풀리(Pully)의 비율	49
5.3.6.7	특수 기능	49
5.3.7	Mill Options 탭	50
5.3.8	테스트 하기	51
5.4	단위 설정	51
5.5	모터의 튜닝	52
5.5.1	단위당 스텝 계산	52
5.5.1.1	기계적 움직임의 계산	52
5.5.1.2	회전 대비 모터 스텝 계산	53
5.5.1.3	모터 회전 대비 Mach 3 스텝 계산하기	53

5.5.1.4 단위당 Mach3 스텝 .....	53
5.5.2 최대 모터 속도 설정 .....	54
5.5.2.1 모터 속도의 실질적 효용 .....	54
5.5.2.2 모터의 최대 속도 계산하기 .....	55
5.5.2.3 단위당 스텝 자동으로 계산 하기 .....	55
5.5.3 모터의 가속도 정하기 .....	56
5.5.3.1 관성과 힘 .....	56
5.5.3.2 각기 다른 가속도 값의 실험.....	56
5.5.3.3 서보 모터의 에러를 피해야 하는 이유 .....	56
5.5.3.4 가속도 값의 바른 선택 .....	56
5.5.4 축의 테스트 및 저장 .....	56
5.5.5 같은 방법으로 다른 축 설정하기 .....	57
5.5.6 스펀들 모터 셋업 .....	58
5.5.6.1 모터 속도, 스펀들 속도 그리고 풀리(pully) .....	58
5.5.6.2 PWM 스펀들 컨트롤러 .....	59
5.5.6.3 스텝과 방향에 의한 스펀들 컨트롤러 .....	59
5.5.6.4 스펀들 드라이브 테스트 하기 .....	60
<b>5.6 기타 설정 사항 .....</b>	<b>60</b>
5.6.1 홈과 소프트 리미트 설정.....	60
5.6.1.1 스텝 과 방향 기준 정하기 .....	60
5.6.1.2 홈 스위치 의 위치 .....	60
5.6.1.3 소프트 리미트 설정 .....	60
5.6.1.4 G28 홈 위치 .....	61
5.6.2 시스템 핫키 설정 .....	61
5.6.3 백래쉬 설정 .....	61
5.6.4 슬레이브 설정 .....	62
5.6.5 툴 패스(tool path) 설정 .....	62
5.6.6 초기 상태 설정 .....	63
5.6.7 다른 논리 항목 설정 .....	65
<b>5.7 프로파일 정보의 저장 방법 .....</b>	<b>66</b>
<b>6. Mach3 컨트롤과 파트 프로그램 운영방법 .....</b>	<b>68</b>
6.1 소개 .....	68
6.2 제어에 관한 설명 .....	68
6.2.1 화면 바꾸기 .....	68
6.2.1.1 리셋 .....	68
6.2.1.2 라벨 .....	69
6.2.1.3 화면선택 버튼 .....	69
6.2.2 축 제어 패밀리(family) .....	69
6.2.2.1 좌표 값 DRO .....	69
6.2.2.2 기준 정하기 .....	69
6.2.2.3 장비 좌표 .....	70
6.2.2.4 스케일(Scale) .....	70
6.2.2.5 소프트리미트 .....	70
6.2.2.6 조회하기 .....	70
6.2.2.7 반경/직경 수정 .....	70

6.2.3 “ Move to” 제어 .....	70
6.2.4 MDI 와 티치 컨트롤 패밀리(family) .....	70
6.2.5 조깅 컨트롤 패밀리( Jogging control family) .....	71
6.2.5.1 핫키 조그하기(hotkey jogging) .....	71
6.2.5.2 병렬(프린터) 포트 혹은 Modbus MPG 조깅 .....	72
6.2.5.3 스핀들 스피들 제어 패밀리 .....	72
6.2.6 피드 제어 패밀리(feed control family) .....	72
6.2.6.1 피드 속도(unit per minute) .....	72
6.2.6.2 피드 회전 속도 .....	73
6.2.6.3 피드 화면표시 .....	73
6.2.6.4 피드반복 하기.....	73
6.2.7 프로그램 동작 제어 패밀리 .....	73
6.2.7.1 작업 시작(Cycle Start) .....	73
6.2.7.2 작업 일시 중지( FeedHold) .....	73
6.2.7.3 작업 중단(Stop) .....	74
6.2.7.4 되돌리기(Rewind) .....	74
6.2.7.5 싱글 블록(Single BLK) .....	74
6.2.7.6 뒤로 가기 (Reverse run) .....	74
6.2.7.7 라인 번호(Line number) .....	74
6.2.7.8 여기서 시작(Run from Here) .....	74
6.2.7.9 다음 라인 설정(Set next line) .....	74
6.2.7.10 블록 삭제(Block Delete) .....	74
6.2.7.11 선택 할 수 있는 정지(.Optional Stop) .....	75
6.2.8 파일 제어 패밀리 .....	75
6.2.9 툴(엔드밀) 세부사항.....	75
6.2.10 G 코드와 툴패스 제어 패밀리 .....	75
6.2.11 작업 옵셋 과 툴 테이블 제어 패밀리 .....	76
6.2.11.1 작업옵셋 .....	76
6.2.11.2 툴 .....	77
6.2.11.3 옵셋 테이블 직접 제어 .....	77
6.2.12 회전 직경 제어 패밀리 .....	77
6.2.13 접선 제어 패밀리 .....	78
6.2.14 리밋 그리고 기타 제어 패밀리 .....	78
6.2.14.1 입력 활성화 4 .....	78
6.2.14.2 설정된 리밋 건너뛰기(Override limits).....	78
6.2.15 시스템 셋팅 제어 패밀리 .....	78
6.2.15.1 단위 .....	79
6.2.15.2 Z축 제한(safe Z) .....	79
6.2.15.3 CV 모드/각도 리밋(제한) .....	79
6.2.15.4 오프라인 .....	79
6.2.16 인코더 제어 패밀리 .....	79
6.2.17 자동 Z축 제어 .....	79
6.2.18 레이저 점화 출력 패밀리(Laser trigger output famil).....	80
6.2.19 맞춤형사용자 제어 패밀리(custom control families) .....	80
<b>6.3 마법사 사용 하기 .....</b>	<b>81</b>

6.4	G 코드 파트 프로그램 실행하기 .....	82
6.5	파트 프로그램 편집하기 .....	83
6.6	매뉴얼 작업으로 준비하고 파트 프로그램 실행하기 .....	83
6.6.1	직접 작성한 프로그램 입력하기 .....	83
6.6.2	파트 프로그램을 실행하기 전에 .....	83
6.6.3	프로그램 실행 .....	84
6.7	다른 파일을 불러들여 G 코드 만들기 .....	84
<b>7.</b>	<b>좌표계, 툴 테이블 고정자 .....</b>	<b>85</b>
7.1	장비 좌표 계 .....	85
7.2	작업 옵션 .....	86
7.2.1	작업 원점을 주어진 좌표에 설정하기 .....	87
7.2.2	실제 장비에서의 홈 .....	88
7.3	엔드밀(tool)의 서로 다른 길이에 관하여 .....	88
7.3.1	사전 설정 가능한 엔드밀(Tool) .....	89
7.3.2	사전 설정 불 가능한 엔드밀(tool) .....	89
7.4	옵션 값이 저장 되는 방법 .....	89
7.5	같은 작업 여러 번 반복하기 - 고정자 사용 .....	90
7.6	터칭의 실용성 .....	91
7.6.1	엔드 밀(end mills) .....	91
7.6.2	접점 찾기(edge finding) .....	91
7.7	G72&G92 옵션 .....	91
7.7.1	G52 사용하기 .....	92
7.7.2	G92 사용하기 .....	93
7.7.3	G52 와 G92 사용상의 주의 .....	93
7.8	엔드밀(tool)의 직경 .....	93
<b>8.</b>	<b>DXF, HPGL 그리고 이미지 파일 불러오기 .....</b>	<b>95</b>
8.1	소개 .....	95
8.2	DXF 불러오기 .....	95
8.2.1	파일 로딩 .....	96
8.2.2	각각의 레이어를 위한 작업 지정 .....	96
8.2.3	전환 옵션 .....	97
8.2.4	G 코드 생성 .....	97
8.3	HPGL 불러오기 .....	98
8.3.1	HPGL 에 관하여 .....	98
8.3.2	불러올 파일 선택 하기 .....	98
8.3.3	파라미터 불러오기 .....	99
8.3.4	G 코드 파일 작성하기 .....	99
8.4	BMP&JPEG 파일 불러오기 .....	100
8.4.1	불러올 파일 선택하기 .....	100
8.4.2	렌더링 타입 선택하기 .....	100
8.4.3	레스터(Raster) 와 나선형 렌더링 .....	101
8.4.4	점 발산(dot diffusion) 렌더링 .....	101
8.4.5	G 코드 파일 작성하기 .....	101



<b>9. 커터 보정(cutter compensation)</b> .....	<b>103</b>
<b>9-1 보정에 관한 소개</b> .....	<b>103</b>
<b>9-2 두 종류의 윤곽선</b> .....	<b>104</b>
9.2.1 재료의 경계 윤곽선 .....	104
9.2.2 툴 패스(tool path) 윤곽선 .....	104
9.2.3 진입 동작 프로그래밍 .....	105
<b>10. Mach 2 G &amp; M 코드 언어 참고</b> .....	<b>106</b>
<b>10.1 정의 내리기</b> .....	<b>106</b>
10.1.1 선형 축 .....	106
10.1.2 회전 축 .....	106
10.1.3 스케일(비율) 입력 .....	106
10.1.4 제어 점 .....	106
10.1.5 좌표 계 상의 선형 운동 .....	107
10.1.6 피드 비율( feed rate) .....	107
10.1.7 호(arc motion) .....	107
10.1.8 냉매제(coolant) .....	107
10.1.9 머무르기(dwell) .....	108
10.1.10 단위 .....	108
10.1.11 현재 위치 .....	108
10.1.12 선택된 좌표면 .....	108
10.1.13 툴 테이블 .....	108
10.1.14 툴 교체 .....	108
10.1.15 팔렛 셔틀(pallet shuttle) .....	108
10.1.16 패스 제어 모드(Path control mode) .....	108
<b>10.2 제어를 통한 해석 프로그램의 상호 작용</b> .....	<b>109</b>
10.2.1 피드와 속도를 반복하기(feed and speed override controls) .....	109
10.2.2 블록 삭제(Block Delete) 제어 .....	109
10.2.3 옵션 프로그램 정지(Optional program stop) 제어 .....	109
<b>10.3 툴 파일(Tool file)</b> .....	<b>109</b>
<b>10.4 파트 프로그램의 언어</b> .....	<b>109</b>
10.4.1 개요 .....	109
10.4.2 파라미터 .....	110
10.4.3 좌표계 .....	111
<b>10.5 라인 포맷</b> .....	<b>110</b>
10.5.1 라인 번호 .....	111
10.5.2 서브루틴 라벨(subroutine label) .....	112
10.5.3 문자 .....	112
10.5.3.1 숫자 .....	112
10.5.3.2 파라미터 값 .....	113
10.5.3.3 표식 과 이진 연산(expression and binary operations) .....	113
10.5.3.4 일진 연산 값(unary operation value) .....	113
10.5.4 파라미터 설정 .....	113
10.5.5 주석과 메시지 .....	113
10.5.6 반복 아이템 .....	113
1.1. 10.5.7 아이템 순서 .....	113

10.5.8 명령어와 장비 모드 .....	113
<b>10.6 형식상의 그룹 .....</b>	<b>113</b>
<b>10.7 G 코드 .....</b>	<b>115</b>
10.7.1 고속 선형 운동 G0 .....	116
10.7.2 피드 비율에 따른 선형 운동 G1 .....	117
10.7.3 피드 비율에 따른 호(arc) G2, G3 .....	117
10.7.3.1 반경 형태의 호 .....	117
10.7.3.2 중심점 형태의 호 .....	118
10.7.4 머무르기(dwell) G4 .....	119
10.7.5 좌표 계 데이터 톨 과 작업 옵셋 테이블 설정 G10 .....	119
10.7.6 시계/반 시계 방향 회전 포켓 G12, G13 .....	119
10.7.7 극(polar) 모드 들어가기, 나가기 G15, G16.....	119
10.7.8 좌표면 선택 G17, G18, G19 .....	120
10.7.9 길이 단위 G20, G21 .....	120
10.7.10 홈으로 돌아가기 G28, G30 .....	120
10.7.11 축 기준점 G28.1 .....	120
10.7.12 수직 프로브(Probe) .....	121
10.7.12.1 수직 프로브 명령 .....	121
10.7.12.2 수직 프로브 명령 사용하기 .....	121
10.7.12.3 예제 코드 .....	121
10.7.13 커터 반경 보정 G40, G41, G42 .....	122
10.7.14 톨 길이 옵셋 G43, G44, G49 .....	123
10.7.15 배율 계수(scale factor) G50, G51 .....	123
10.7.16 임시 좌표 계 옵셋 G52 .....	123
10.7.17 절대 좌표 계에서의 동작 G53 .....	124
10.7.18 작업 옵셋 좌표 계 선택 G54~G59 & G59P~ .....	124
10.7.19 패스(path) 제어 모드 설정 G61, G64 .....	124
10.7.20 좌표 계 회전 G68, G69 .....	124
10.7.21 길이 단위 G70, G71 .....	125
10.7.22 고정 사이클(canned cycle) 고속 팩 드릴링 G73 .....	125
10.7.23 형식 모션 취소하기 G80 .....	125
10.7.24 고정 사이클(canned cycle) G81~G89 .....	125
10.7.24.1 예비 동작과 중간 동작 .....	126
10.7.24.2 G81 사이클 .....	127
10.7.24.3 G82 사이클 .....	127
10.7.24.4 G83 사이클 .....	128
10.7.24.5 G84 사이클 .....	128
10.7.24.6 G85 사이클 .....	128
10.7.24.7 G86 사이클 .....	128
10.7.24.8 G87 사이클 .....	129
10.7.24.9 G88 사이클 .....	130
10.7.24.10G89사이클 .....	130
10.7.25 거리 모드 설정 하기 - G90, G91 .....	130
10.7.26 IJ 모드 설정 G90.1, G91.1.....	130
10.7.27 G92 옵셋 G92, G92.1, G92.2, G92.3 .....	131
10.7.28 피드 비율 모드 설정 G93, G94, G95 .....	131

10.7.29 고정 싸이클 회귀 높이(return level) G98, G99 .....	132
<b>10.8 내장된 M 코드 .....</b>	<b>132</b>
10.8.1 프로그램 시작하기와 끝내기 M0, M1, M2, M30 .....	132
10.8.2 스핀들 제어 M3, M4, M5 .....	133
10.8.3 톨 교체 M6 .....	133
10.8.4 냉 매제 제어 M7, M8, M9 .....	133
10.8.5 첫 라인에서 다시 시작 하기 M47 .....	133
10.8.6 제어 반복하기 M48, M49 .....	133
10.8.7 서브루틴 호출 M98 .....	133
10.8.8 서브루틴에서의 회귀 .....	133
<b>10.9 매크로 M 코드 .....</b>	<b>134</b>
10.9.1 매크로 개요.....	134
<b>10.10 기타 입력 코드 .....</b>	<b>134</b>
10.10.1 피드 비율 -F 설정.....	134
10.10.2 스핀들 속도 - S 설정 .....	134
10.10.3 톨 선택 - T .....	134
<b>10.11 에러 처리하기 .....</b>	<b>134</b>
<b>10.12 실행 순서 .....</b>	<b>134</b>
<b>11. 부록 1 Mach 3 스크린 샷 .....</b>	<b>136</b>
<b>12. 부록 2 예제 개략도 .....</b>	<b>139</b>
12.1 릴레이를 사용한 Estop 그리고 리밋 .....	139
<b>13. 부록 3 - 설정의 저장 .....</b>	<b>141</b>

## 1 서론



어떤 장비든 위험성은 내제되어 있다. 특히나 컴퓨터에 의해 동작되는 기계는 손으로 작동 시키는 기계보다 훨씬 위험율이 높을 수 있는데 그것은, 예를 들면, 컴퓨터가 8인치 두께의 선반 척(chuck)을 비대칭으로 3000rpm 의 속도로 회전을 하게 한다거나 오퍼레이션 착오로 작업 테이블 자체를 밀링 한다고 했을 때를 어떻게 될지 상상 할 수있지 않은가.

이 매뉴얼은 여러분의 안전에 대한 사전예방 및 기술을 설명 하고 있지만 여러분 각 장비에 대한 세부 사항은 알 수 없는 관계로 장비의 효율이나 어떠한 파손 및 상해에 대한 책임을 질 수 없다는 것을 명시 하는 바이다. 그리고 사용자의 해당 국가나 지역의 법률에 따라 여러분이 각자 설계 하고 만드는 것에 대한 책임이 있다는 것을 잘 염두 해 주기 바란다.

혹시라도 의심 가는 부분이 있다면 반드시 전문가에게 가서 상의 하라고 권고 하는 바이다. 그것이 본인이나 타인에게 위험으로 인한 피해를 주는 것 보다 백 번 낫지 않겠는가?!

이 매뉴얼은 Mach3Mill 소프트웨어와 각자 소유한 장비와의 상호 관계를 충분히 설명 할 수 있도록 했으며, 각각의 축을 어떻게 설정 하는가와 프로그래밍을 위한 입력 언어 및 포맷의 지원으로 6개의 축까지 가능한 강력한 CNC 시스템을 구축 할 수 있도록 도와줄 것이다. 일반적으로 이러한 장비들은 대개 밀링, 라우터, 그리고 플라즈마 커팅 테이블등이 될 것이다.

Mach3Mill이 프로파일을 회전시키는 2축 선반등을 제어 할 수도 있지만, Mach3Turn과 같이 완전히 분리된 프로그램 또한 선반과 같은 장비가 완전히 작동하도록 개발되어 있다.

이와 아울러 온라인 상의 위키 문서 형식의 Customizing Mach3 설명서는 스크린 배치 변환 방법이나 스크린, 마법사를 디자인 하는 법 그리고 특수한 장치를 연결하는 방법 등을 자세하게 설명해 놓았다.

또한 Mach3 의 Yahoo에서 주최하는 온라인 포럼에 참여 하기를 권고 하는 바이다. [www.machsupport.com](http://www.machsupport.com) 에 링크가 있으니 참고 하길 바란다.(한국의 경우 동호회나 클럽을 자주 이용하는 것이 바람직 하며 영어에 제한을 두지 않는다면 국외의 여러 웹사이트들로 많은 정보를 얻을 수 있을 것이다. - 번역자 소견) 만약 필요한 어플리케이션이 일정 수준을 요구 한다면 OEM이나 지역 대리점에서 그에 맞는 시스템을 구매해야 할 것이다.

일부 이 매뉴얼에는 회색으로 쓰여진 글이 있는데 그것은 장비의 컨트롤러에 기능은 있을 수 있으나 Mach3 에서는 현재 적용되지 않으며 향후에 반드시 실행될 것이란 공약이 될 수 없음을 알기 바란다.

National Institute for Standards and Testing(NIST) EMC 프로젝트에 참가한 팀과 또한 Mach3 사용자들로, 경험, 자료 그리고 도움이 될 코멘트 없이는 만들어 지지 못했을 이 문서이기 때문에 그들 모두에게 감사를 드린다.  
차후로 Artsoft 주식회사는 증진적인 제품개발을 할 것이며 개발에 도움이 되는 모든 것들을 겸허히 받아 들일 것이다.

아트 페널티와 존 프렌티스는 이 작업의 저작권을 주장하는 바이다. 이 매뉴얼의 복제는 전적으로 Mach3의 데모카피 나 정품의 평가를 위해서만 사용 할 수 있다. 그러므로 타 회사들의 이 매뉴얼에 대한 판매는 허가 되지 않는다.

비록 매뉴얼에 대한 정확도나 완성도를 위해 수 많은 노력을 했다 하여도 보증이나 적용성에 대해서는 법적인 책임이 없다는 것을 알기 바라며 이 매뉴얼에 명시한 정보들은 “ 있는 그대로” 를 기술 했고 저작자나 출판인은 이 정보로 인한 어떤 상해나 피해에도 책임과 의무가 없음을 알리는 바이다. 이 매뉴얼의 사용은 계약 조건에 포함되어 있으며 Mach3 소프트웨어 인스톨을 할 때 반드시 동의해야 하는 내용이다.

Windows XP 와 Windows 2000 은 Microsoft 의 등록상표이며 만약 다른 상표가 이 매뉴얼에서 발견 된다면 다음 출간의 정정에 참고가 되도록 ArtSoft에 알려 주기 바란다.

## 2. CNC 머시닝 시스템 소개

### 2.1 머시닝 시스템의 각 부분

이번 장에서는 이 매뉴얼에서 사용하게 될 용어를 소개하며 수동으로 조작되는 밀링 머신의 각각의 부분의 목적에 대한 이해를 전달 하게 될 것이다.

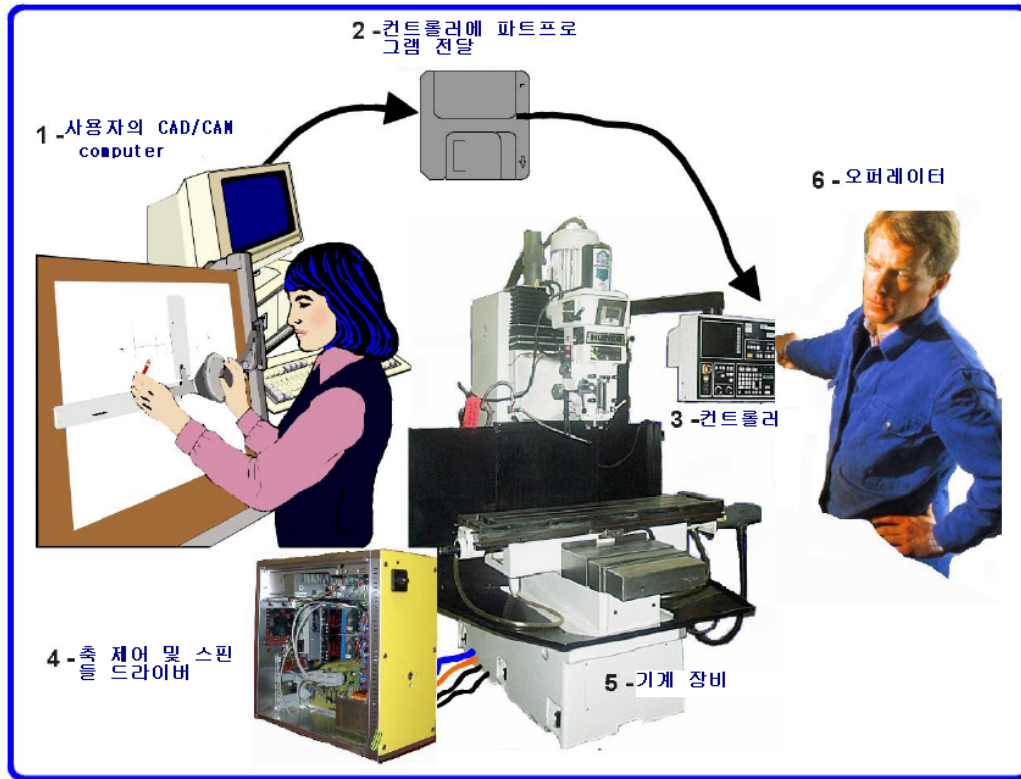


그림 1.1 - 전형적인 NC 머시닝 시스템

보통 제품을 만들 때 초기에 디자이너는 그림의 1번과 같이 CAD/CAM(Computer Aided Design/ Computer Aided Manufacturing) 프로그램을 이용한다. 이 프로그램들은 최종적으로 “G-코드” 형태의 파트 프로그램을 생성하며 그림의 2번과 같이 다시 장비 컨트롤러로 전송하고(네트워크 및 플로피 등등을 사용) 그림 3번과 같이 장비의 컨트롤러가 이것을 해석해서 대상물을 절삭하는 것이다. 그림의 5번 기계장비의 각 축들은 그림 4번의 컨트롤러에 의해 신호가 서보나 스텝 모터에 전달 되고 맞물린 리드스크류, 랙기어, 벨트등을 회전 시켜 장비 각각의 축들을 움직이게 한다.

위의 그림은 밀링 머신을 위주로 표시 되었으나 플라즈마, 레이저 커터 및 라우터에도 적용이 된다. 별도의 매뉴얼은 Mach3의 선반이나 수직 프레스에 관하여 소개 한다.

장비의 컨트롤러는 스피드 모터의 시동, 정지 및 속도를 제어할 수 있으며, 또한, 냉매제를 작동 시킬 수도 있고 파트 프로그램 운영이나 오퍼레이터가 작업 범위를 넘기는 오류를 하지 않는가 등등을 체크 할 수도 있다.

**메모 [DP1]:** 간단히 말해 CAD와 CAM 으로 작업하고 CAM의 포스트프로세스를 통해 G코드를 생성하고 그것을 Mach3 와 같은 머시닝 소프트웨어를 통해 실제 절삭을 하게 된다.

장비 컨트롤러는 버튼, 키보드, MPG 또는 조이스틱 과 같은 장치를 제어 할 수 있어 수동으로 파트 프로그램을 시작, 정지 할 수 있게 해주며 또한 모니터로 작업자에게 그 상황을 보여 주기도 한다.

G-code 의 명령어 들은 장비의 각 축에 아주 복잡한 좌표운동을 시킬 때가 있어 장비 컨트롤러는 리얼타임으로 계산 할 수 있는 능력이 있어야 하는데 예를 들면, 나선형 절삭 시 무수한 삼각측량 계산이 요구 되는 이유로 우수한 성능의 수행능력을 필요로 한다. 과거 이런 이유로 장비들이 가격이 비쌀 수 밖에 없었다.

## 2.2 Mach3 의 적용

Mach3는 PC에서 사용되는 소프트웨어 패키지이며 그림 1-1의 3번 같은 장비 컨트롤러를 대치 할 수 있는 경제적이고 강력한 툴이다.

Mach3를 실행하기 위해서는 WindowsXP 나 혹은 Windows2000, 1GHz 프로세서와 해상도 1024 X 768의 환경을 요구한다. 데스크 탑 환경이 노트북 컴퓨터 보다 가격 면이나 성능 면에서 좋을 것이라 본다. 물론 이때 사용 되는 컴퓨터는 Mach3가 사용되지 않을 때, 그림 1-1 에서의 1번과 같이 CAD/CAM 과 같은 소프트웨어도 사용 할 수 있다.

Mach3는 병렬 포트(프린터 포트)를 이용하여 통신하며 필요에 의해서라면 시리얼(COM) 포트로도 가능 하다. 장비 축에 사용 하는 모터 드라이버는 반드시 스텝 신호와 방향 신호를 인식 할 수 있어야 한다. 대부분의 인코더가 있는 AC, DC 서보 모터처럼 스텝 모터 드라이버는 위의 기능에 의해 작동 된다.

만약 리졸버(resolver)를 사용 하는 서보가 내장된 옛날 NC 장비를 교체 한다면 각 축의 드라이버 또한 거기에 대응 해야 한다는 것을 명심 하기 바란다.

**메모 [DP2]:** 공간적으로 직각인 2쌍의 권선(捲線)을 갖는 회전자(回轉子)와 고정자(固定子)로 되어 있으며, 철심(鐵心)과 코일로 구성되어 있다.

### 3. Mach 2 장비 컨트롤 소프트웨어 개요

지금 여러분이 이것을 읽고 있는 이유는 Mach3가 여러분의 작업에 중요한 부분이기 때문 이라고 본다. 한번 데모버전 소프트웨어를 다운로드 받아서 시도 해 보길 바라며 당장 사용할 장비가 없어도 사용 가능하다. 사실 장비에 연결 하지 않는 것이 더 바람직 할 것이다. 패키지를 구매 하였다면 아마 아래의 일부 혹은 전체 스텝 은 이미 완성 됐다고 본다.

#### 3.1 설치

Mach3는 ArtSoft 주식회사에 의해 인터넷으로 판매 된다. 다운로드 파일은 하나의 실행 파일로 8MB 용량이며 무한 기간 사용 가능 하지만 속도 나 특정 작업을 할 수 없도록 제약 되어 있으며 라이선스를 구매 하면 제약을 해제 하여 모든 기능을 사용 할 수 있다. 옵션이나 가격 등 자세한 세부 사항은 아래의 ArtSoft 주식 회사 웹 페이지를 참고 하기 바란다.

[www.artofcnc.ca](http://www.artofcnc.ca) 혹은 [www.machsupport.com](http://www.machsupport.com)

##### 3.1.1 다운로드

[www.artofcnc.ca](http://www.artofcnc.ca) 사이트에서 download - Mach and LazyCAM 으로 가서 자동 설치 되는 파일을 다운 받아 실행 한다.

##### 3.1.2 설치

설치 시 장비와 연결이 되어 있지 않아도 상관없다. 만약 이것을 처음 배우는 것이라면 오히려 장비가 연결 되어 있지 않는 것이 더 낫을 것이다. 장비의 케이블과 커넥터가 PC의 어디에 연결이 되는지 잘 살펴 보아라. 연결이 되어 있다면 컴퓨터와 장비의 전원을 끄고 드라이버와 PC 간에 연결된 25핀 병렬포트 케이블을 뽑은 후 다시 PC 전원을 켜다. 설치 파일을 실행 시키면 계약 조건, 폴더 지정 등 한 단계씩 실행이 되며 설치가 끝나면 반드시 PC를 재 시동 한다.

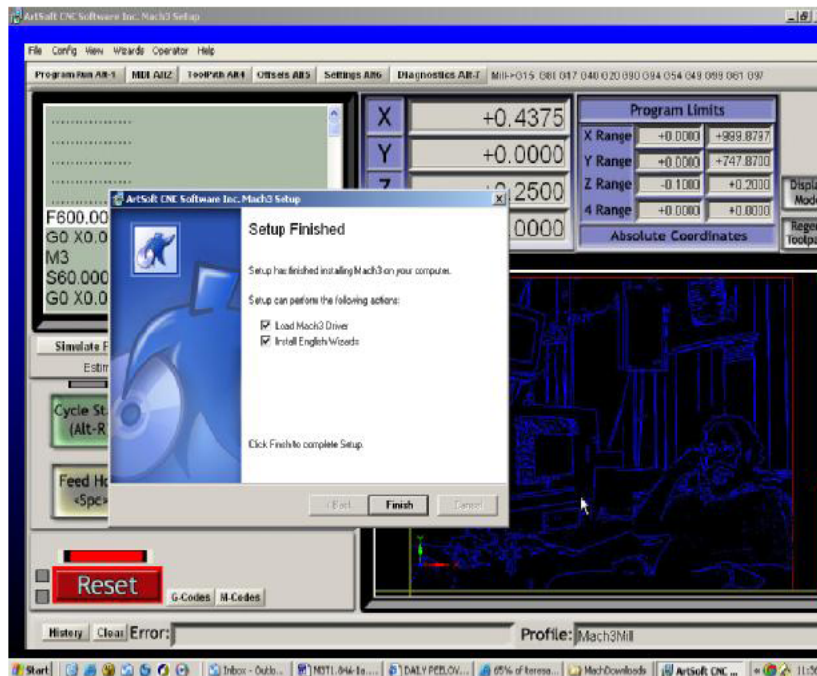


그림 3.1 설치 화면



백그라운드 상의 이미지는 Mach3Mill 스크린 이지만 동시에 Mach3Turn 또한 설치 될 것이다. 설치 중에 반드시 *Load Mach3 Driver* 와 *Install English Wizards* 항목을 반드시 체크 하여 설치 하기 바란다.

Mach3의 설치의 끝 부분에서 “ *Initialize System* ” 을 반드시 체크 하고 finish 버튼을 누른다. 그리고 컴퓨터를 리부팅 하여 다시 시작 한다.

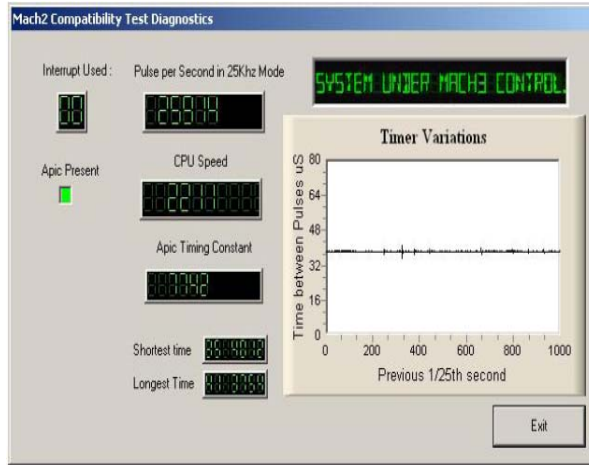


그림 3.2 DriverTest 실행화면

### 3.1.3 Windows 재 시작

프로그램을 사용 하기 전에 컴퓨터를 리부팅 하는 것은 **필수**다. 리부팅을 실행 하지 않을 경우 차후에 제어판에서 프로그램을 인위적으로 지울 수 있는 상황에 직면할 수 있다는 것을 알기 바란다. 이에 대한 자세한 내용은 아래 내용을 참고 하기 바란다. 제 시작을 했다면 생략하고 다음으로 넘어 가도 좋다.

Mach3에서 “ driver ” 는 독창적인 부분이다. Mach3는 매우 정확한 타이밍에 신호를 장비의 각 축으로 보내어 제어한다. Windows는 일반적으로 사용자의 프로그램들을 자신의 책임 하에 진행시키려 한다. 하지만 Mach3 는 그 일반적인 사용자 프로그램이 되어서는 안 된다. 왜냐하면 프로그램이 Windows 의 최 하부 구조로 스스로 내려가 인터럽트를 직접 관할해야 하기 때문이다. 각 축을 초당 45000번의 회수로 처리 할 수도 있는 일들을 실행하기 위해서는 드라이버가 자신만의 코드를 가지고 있어야 하는데 문제는 Windows 는 이런 프로세스를 바이러스라고 간주 하기 때문에 그 작업을 가능 하게 하려면 특별한 인가를 요청해야 한다. 바로 그것이 Windows를 재 시작 해야 하는 이유이다. 만약 재 시작 하지 않을 경우 Windows가 블루 스크린을 띄울 수 있으며 아울러 그 안의 드라이버 까지 파손 시킬 수 있다. 이런 일이 발생했다면 수동으로 Mach3 프로그램을 제거 하고 다시 설치 하는 방법 밖에는 없다.

무시무시한 경고지만 사실 첫 설치 후 windows를 재 시작 하라는 말 밖엔 뭐라 할 말이 없다. 그러나 Mach 소프트웨어의 새로운 버전을 업데이트 한다면 비록 설치 중 재 시작 하라는 메시지가 나올 지라도 그럴 필요는 없다.

### 3.1.4 바탕화면 아이콘

자 이제 컴퓨터를 재 시작 했다고 가정하자. 바탕화면에 몇 개의 아이콘이 보일 것이다. 그 중 Mach3.exe 가 사용자 인터페이스 코드가 된다. 이것을 실행하면 Mach3Mill, Mach3Turn 등등 어떤 프로파일을 운영하고 싶은지 묻는 창이 뜬다. 이때 Mach3Mill 이나 Mach3Turn은 단축 아이콘 안에 있는 “ /p ” 라는 독립변수로 정의된 프로파일을 실행 할 것이다. 필요에 따라 각각의 것들을 수행 하면 된다.

Mach3 폴더 안에 있는 DriverTest.exe를 단축 아이콘을 만들어 바탕 화면에 생성해 본다. 이밖에 스크린 디자인 프로그램이나 스크린샷 조작 프로그램 또한 Mach3 다운로드 사이트를 통해 실행 가능하다.

### 3.1.5 설치한 프로그램 테스트 하기.

설치한 프로그램은 반드시 시험해 보기 바란다. Mach3는 간단한 프로그램이 아니다. 작업을 수행하기 위해서 Windows 와는 거의 별개로 작동한다. 이것은 여러 가지의 이유가 있으며, 모든 시스템에서 작동 한다는 것을 뜻하지 않는다. 예를 들자면, QuickTime 의 시스템 모니터 qtask.exe 가 돌고 있다면 Mach 프로그램과 충돌 할 수도 있으며 실제 다른 알려지지 않은 프로그램들도 같은 경우를 만들 수 있다. Windows 는 백그라운드에 많은 프로세스를 운영 할 수 있다. 어떤 것들은 Systray 에 아이콘으로 표시 되기도 하고 어떤 것들은 전혀 알 수 없는 것들로 표시 되기도 한다. 다른 가능한 돌출적 문제 요소로는 네트워크 설정 인데, 10Mbps 나 100Mbps 속도 를 자동으로 선택 하게 하지 말고 수동으로 하나를 택하라고 권장 한다. 마지막으로 장비가 인터넷을 사용 할 수 있도록 되어 있다면 하나 이상의 로봇 형태의 호스트가 생성 될 수 있고 이것은 사용 자가 무엇을 하는지 살필 수 있는데 이런 네트워크 상의 트래픽(traffic)은 Mach3 의 운영에 영향을 줄 수 있다. 검색 엔진을 사용 하여 spybot 같은 소프트웨어를 찾아 내고 장비를 안전하게 사용 하길 바란다.

이러한 이유들로, 비록 필수적인 것은 아니더라도, 설치한 프로그램에 이상이 있는 가를 확인하기 위해 시험 해 보라는 것이다.

Mach3 폴더 안에 있는 DriverTest 아이콘을 더블 클릭 해보라. 위의 그림 3-2과 같을 것이다.

그림에서 Pulse Frequency를 제외한 모든 것은 무시해도 좋다. Pulse Frequency 의 값은 25,00 Hz에서 일정 범위로 움직인다 하지만 여러분의 컴퓨터에서는 다를 수 있다. 왜냐하면 Mach3는 Windows 의 클럭을 이용하여 펄스 타이머를 교정하는데, 아주 잠깐 동안 Windows의 클럭이 다른 프로세스에 의해 영향을 받아 영향을 줄 수 있기 때문이다. 이렇게 불안정한 Windows의 클럭을 사용 하는 Mach3의 타이머에 대해 잘못된 인식을 갖을 수 있을 것이다.

만약 여러분의 컴퓨터에서 위 그림과 비슷한 상태라면 이상이 없다고 봐야 하니 DriverTest 프로그램을 종료 하고 다음으로 넘어간다.

Windows 전문가라면 아마 다른 몇 가지에 관심이 생길 것이다. 하얀 화면은 일종의 타이밍 분석 툴이다. 그림의 파형은 인터럽트의 변환에서 오는 타이밍이다. 17인치 모니터 상에서 1/4인치 이상의 라인은 볼 수 없을 것이다. 설명 위의 그림과 다르더라도 범주 내에서 운영 할 수 있기 때문에 장비가 연결이 되어 있다면 조그 하여 움직이거나 G0/G1 의 움직임이 부드럽다는 것을 볼 수 있을 것이다.

위 화면을 실행 하려 할 때 아래 둘 중 하나의 증상이 생긴 다면 문제가 발생한 것이니 다음을 잘 읽어 보기 바란다.

1. “ Driver not found or installed, contact Art” 이것은 어떤 이유에서 인지 Windows 상에 드라이버가 로드 되어 있지 않았다는 것을 말하며 Windows XP의 드라이버 데이터베이스가 깨졌을 경우인데 이때는 다시 컴퓨터를 리붓 하는 것이 도움이 될 것이다. Windows 2000 에서는 일종의 드라이버 로딩 버그 현상으로 나타나는데 수동으로 드라이버를 로딩 해 주어야 한다.

2. 메시지 창이 뜨고...3..2..1 그리고 자동 리붓을 한다. 이때는 둘 중의 하나다. 프로그램 설치 후 리붓을 하지 않았던지 드라이버가 깨지거나 사용 불가능한 경우다. 이럴 때는 드라이버를 수동으로 제거 하고 다시 설치 하라. 그래도 반복 될 때는 [www.artofcnc.ca](http://www.artofcnc.ca) 의 링크 이메일 주소로 연락 주면 가이드라인을 제시해 줄 것이다.

소수의 컴퓨터 마더보드 중에는 BIOS가 사용 하지 않는 APIC 타이머를 가지고 있기도 한데 이것이 Mach3의 설치에 방해를 준다. 이때는 Mach3 설치 폴더 안에 있는 “ specialdriver.bat” batch 파일을 찾아서 실행해 준다. 이것은 Mach3 가 옛 i8529 인터럽터 컨트롤러를 사용 할 수 있게 해 준다. Mach3 업그레이드 버전을 다운 받아 실행 한다면 항상 이런 과정을 반복 해야 할 것이다. OriginalDriver.bat 파일은 이런 변화를 돌려 놓는다.

### 3.1.6 Mach3 프로그램 충돌 시 드라이버 테스트 하기.

Mach3를 사용 할 때 어떤 이유에서든 충돌을 경험 할 수도 있다. 이때는 간헐적인 하드웨어 문제이거나 소프트웨어 버그일 수 있으니 충돌 직후 바로 DriveTest.exe 를 실행 한다. 만약 충돌 직후 2분을 넘기면 Windows 블루스크린 을 보게 될 수도 있다. DriveTest.exe 는 Mach3 프로그램이 사라지더라도 드라이버를 재가동 시켜 안정된 상태를 만든다.

아마 충돌 후 처음 이 것을 실행 하려 할 때 드라이버를 찾는 것에 실패 할 수 있지만 이런 경우 단지 다시 실행 하면 문제를 해결 할 수 있게 될 것이다.

### 3.1.7 드라이버의 수동 설치와 제거.

이 번 항목은 오직 DriverTest를 실패 했을 경우에 한해서 참고 하길 바란다.

Mach3.sys 파일은 Windows 제어판을 통해서 설치 및 삭제가 가능하다. Windows2000과 XP는 화면상의 차이가 있지만 실행하는 프로세스의 단계는 같다.

설치 시

- 제어판 열기 , 더블 클릭 시스템.
- 하드웨어 탭 선택 그리고 새 하드웨어 추가 마법사 열기(전에 언급 한 것과 같이 Mach3는 Windows의 최 하부 구조에서 작동한다.) Windows 가 자동으로 찾게 하면 아무것도 못 찾게 된다.
- 새 하드웨어 장치 추가하기를 하고
- “ 목록에서 직접 선택한 하드웨어 설치(고급)” - Mach1/2 pulsing engine 선택 - 디스크 있음 선택
- 폴더 C:WMach3 에서 Mach3.inf 파일 선택
- 드라이버 설치 끝

제거

- 제어판 - 시스템 - 하드웨어 탭에서 장치 관리자. 여기에서 Mach1 pulsing engine 을 선택하고 오른쪽 마우스 버튼을 누른 후 “ 삭제” 를 선택 하면 제거 된다. 이때 Mach3.sys가 Windows 폴더로부터 제거가 된다.

마지막 한가지 알아 둘 것. Windows는 Mach3의 설정에 관한 모든 것을 Profile 파일에 저장해 두며 이 정보는 Mach3 드라이버를 재 설치 하던 삭제 하던 남아 있다.

만약 완전히 지우고 싶다면 .XML 프로파일 파일들을 찾아서 지워야 할 것이다.

## 3.2 표시 화면

자 이제 장비 없는 Mach3를 실행 할 준비가 되어 있다. 충분한 연습을 통해서 설정 하는 것을 배우고 난 후에는 실제 장비를 보다 쉽게 설정 할 수 있을 것이다. CNC 장비가 없이도 많은 것을 배울 수 있으며 혹시 장비가 있더라도 연결 하지 말고 시작해 보라.

Mach3는 작업자의 편의를 위해서 표시 화면을 사용자가 만들 수 있다. 아마도 부록 1에 나와 있는 표시 화면이 여러분의 것과 같지 않을 수도 있다. 만약 여러분이 보는 화면이 완전히 다르다면 그것은 아마도 공급자가 여러분의 환경에 맞게 개정을 한 것 이라 본다.

Mach3Mill 아이콘을 더블클릭 한다. Mill Program Run 화면이 눈 앞에 나타날 것이며 부록 1의 그림과 비슷한 것이다. (DRO(Digital Read Out) 설정 등은 차이가 있을 것이다.)

빨간색 Reset 버튼을 보라. 깜빡이는 빨간/초록 LED(여기서의 LED는 스크린 상의 가상 LED 이다.) 와 조그마한 노란색 네모 박스를 볼 수 있을 것이다.

Reset 버튼을 누르면 깜박임이 멈추고 녹색이 되며 노란색 LED는 멈춘다. 자 이제 시작해 보자!!

만약 Reset이 위와 같이 되지 않으면 장착된 병렬 포트에 문제가 있거나 혹은 PC에서 그전에 설치된 Mach3가 Emergency Stop(Estop)에 해당하는 포트 핀이 잘못 설정되어 작동 하지 않을 경우이다. Offline 버튼을 누르면 리셋을 할 수 있다.

이번 장의 대부분 설명은 Reset이 제대로 되지 않은 상태에서는 진행할 수 없으니 참고 하기 바란다.

### 3.2.1 각 화면의 내용별 항목

화면에는 다음과 같은 대상 부류를 볼 수 있을 것이다.

- 버튼 (예 : .Reset, Stop Alt-S, 등등)
- DRO (Digital Read Out) 숫자가 나오는 모든 것은 DRO 라고 보면 된다.  
주요 부분은 물론 X, Y, Z, A, B 그리고 C 축이다.
- LEDs ( 다양한 모양과 크기로 보인다)
- 스크롤 바가 옆에 있는 G-code 표시 화면 창
- 툴 패스(tool path) 화면 창(스크린 상에 보이는 텅빈 창)
- 조깅 제어부분.

Program Run 화면에 보이지 않는 중요한 제어 부분이 하나 더 있다:

- MDI (Manual Data Input) 라인

버튼과 MDI 는 여러분이 Mach3에 사용할 입력 창이 될 것이다.

DRO는 Mach3에 의해 표시 될 수도 있고 직접 입력에 의해서 표시 될 수도 있으며 창의 배경색은 입력 시 다른 색으로 변한다.

G-코드 화면과 툴 패스 화면은 Mach3가 보여주는 정보 화면이다. 그와 동시에 G-코드 화면의 스크롤바를 움직인다 단지 툴패스 창에서 화면 축소, 확대, 회전 그리고 움직임을 조작 할 수 있다.

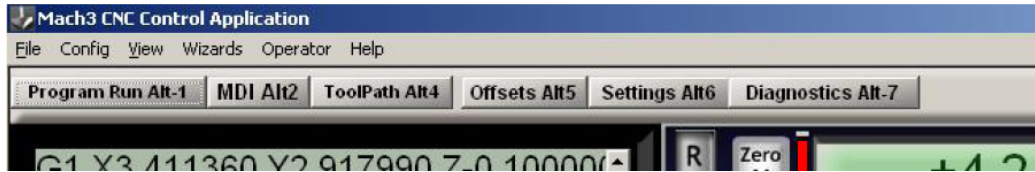


그림 3.3 화면 선택 버튼

### 3.2.2 단축키와 버튼의 사용법

화면상에 있는 버튼 들은 대부분 키보드 핫키가 있다. 이것은 대부분 버튼 이름 다음에 위치하거나 버튼 근처의 라벨에 있다. 해당 키보드를 누르는 것은 마우스로 버튼을 누르는 것과 똑 같다. 스펀들의 전원 제어, 냉매제의 흐름 제어, MDI 화면으로의 전환 등, 단지 키보드 하나와 마우스 버튼으로 할 수 있다면 흥미로울 것이다. 한가지, 어떤 단축키는 Ctrl 키나 Alt 키와 복합적으로 사용 된다. 그때 문자가 대문자라고 해서 Shift 키를 함께 사용해서는 안 된다.

콘트롤 패널의 스위치로 Mach3를 제어 할 수 있는데 그것은 키보드의 에뮬레이터를 사용 함으로 가능 하다.(예 : Ultimarc IPAC). 이것은 키보드와 순차적으로 연결이 되어 Mach3에 단축키로 버튼을 활성화 시키는 가상 키놀림을 전송한다.

만약 버튼이 현재 스크린에 나타나지 않는다면 키보드의 단축키가 활성화 된 것이 아니다. 범용 스크린에 적용되는 어떤 특정한 키보드 단축키도 있다. 이것은 5장에서 설명될 것이다.

### 3.2.3 DRO에 데이터 입력

DRO창에 데이터 값을 직접 입력 할 수 있는데, 창을 클릭하여 입력 하거나 설정된 핫키를 눌러 입력 하거나 혹은 DRO를 선택하기 위한 범용 핫키를 누른 후 원하는 값만큼 키보드의 화살표 키로 입력 할 수 있다. 피드 비율 45.6을 Program Run 화면에 한번 입력해 보라. 새로운 값을 입력 완료하기 위해서는 **Enter 키를 반드시 눌러야** 하며 취소를 하기 위해서는 Esc 키를 누른다. Backspace 와 Delete 키는 DRO 창에 입력 시 사용되지 않는다.

주의 : DRO 창에 직접 데이터 값을 입력 하는 것이 항상 좋은 일은 아니다. 한 예로, 스피들의 실제 속도는 Mach3에 의해서 계산 되어지는 값 인데 그것을 임의로 입력 하면 그 값을 덮어 쓰게 된다. 각 축의 DRO 창에 값을 입력 할 수 있지만 7장을 자세히 읽기 전에는 시도 하지 않기를 바란다. 이것은 장비를 사용 하는 좋은 방법이 아니다.

### 3.3 조그 사용법

조그를 사용하여 수동으로 툴을 작업 범위의 어떤 좌표 위치로도 움직일 수 있다. 물론 어떤 장비에서는 작업 테이블이 움직이는 경우도 있다. 우리는 여기서 “ 툴을 움직인다” 라는 표현을 편의상 사용 할 것이다.

조깅(jogging) 컨트롤은 일종의 특별한 팝업(pop up) 창이다. 이것은 키보드의 **Tab** 키로 on, off 할 수 있다. Tab 키를 누르면 그림 3.4 와 같은 화면이 나타날 것이다.

조그(Jog)는 키보드를 이용 하거나 마우스로 직접 화살표 부분을 눌러 움직이게 한다. 키보드의 화살표 키를 눌러 X, Y 를 움직이게 하고 PgUp/PgDn 키를 눌러 Z 축을 움직인다. 사용자의 편의를 위해 키를 제 설정 가능하며 5장을 참고 하기 바란다. 조그 화면은 다른 모든 화면에서도 사용 가능하다.

그림 3.4 에서 Step LED 에 노란 불이 들어 와 있는 것을 볼 수 있다. 여기에서 조그 모드 버튼을 누를 때 마다 Continuous, Step 그리고 MPG 모드로 넘어 간다.

Continuous 모드에서는 키를 누르고 있는 동안 선택된 축은 계속 움직이게 된다. 이때 속도는 “ Slow jog rate” 의 % DRO 창으로 조절 할 수 있다. 비율은 0.1 에서 100% 까지 어떤 값을 입력 해도 된다. DRO 창 양 옆 “ Up”, “ Dn” 버튼은 5%씩 조절 한다. Shift 키를 누른 상태에서 조깅을 하면 어떤 값이던 관계 없이 100%로 동작 한다. 이것은 가고자 하는 위치에 빠르고 정확하게 움직일 수 있도록 도와 준다.

Step 모드에서는 한번씩 키를 누를 때 마다 Step DRO 창에 표시된 값 만큼 움직인다. 어떤 값으로도 쓸 수 있으며 움직이는 현재의 피드 비율과 같을 것이다. 또한 Cycle Jog Step 버튼 옆의 미리 설정된 스텝 만큼 반복 할 수도 있다.



그림 3.4 조그 제어(tab 키 사용)

로터리 인코더는 병렬 포트 입력 핀을 통해 Mach3에 Manual Pulse Generator(MPG) 기능으로 연결 될 수 있다. MPG 모드에서는 노브(knob)의 회전에 의해 조깅이 된다. Alt A, Alt B, Alt C 키는 사용 가능한 축을 따라 순차적으로 선택 할 수 있게 해 주고 LED는 현재 선택된 축을 조깅 할 수 있도록 보여준다.

조깅을 할 수 있는 다른 방법 중 하나는, PC의 게임 포트나 혹은 USB 포트에 연결된 조이스틱을 사용 하는 것이다. Mach3는 Windows와 호환 가능한 아날로그 조이스틱에 작동 한다. (X 축을 페라리 자동차 핸들처럼 움직일 수 있을 정도다) 하지만 조이스틱 장치를 위해서는 적절한 Windows 드라이버가 필요 할 것이다. 조절은 조이스틱 버튼으로 가능 하지만 반드시 안전상의 이유로 스틱이 중간에 놓여 있기를 바란다.

만약 스로틀(throttle) 제어 기능이 있는 조이스틱 이라면 그것으로 속도 조절이나 피드 비율 조절을 할 수 있다(5장 참조)  
 이런 조이스틱으로 장비를 쉬운 방법으로 자유롭게 수동 제어 할 수 있다. 추가로, 다중 조이스틱 역시 사용 할 수 있으며 Mach 와 호환되는 소프트웨어가 있어야 한다.

자 이제 조깅 옵션을 선택해 보라. 그러나 그에 상응 하는 키보드 단축키가 있다는 것도 잊지 말아라. 하다 보면 작업 하기 편한 방법을 찾게 될 것이다.

### 3.4 매뉴얼 데이터 입력(MDI) 과 티칭(teaching)

#### 3.4.1 MDI

마우스나 키보드의 단축키를 통해서 MDI 화면으로 이동 해 보자.  
 데이터 입력을 위한 커다란 입력 라인을 볼 수 있다. 한 줄로 되어 있으며 창에 가능한 값을 입력 하고 Enter 키를 누른다. Esc 키를 누르면 취소 되고 Backspace키는 타이핑을 정정 할 때 사용 할 수 있다.

G-code 명령어를 안다면 직접 라인 안에 타이핑 해 보라. 모른다면 다음과 같이 해 보자.

G0 X1.6 Y2.3(0 = zero)  
 이것은 X=1.6 Y=2.3 단위(mm 혹은 inch) 만큼 움직일 것이다. DRO 창에서

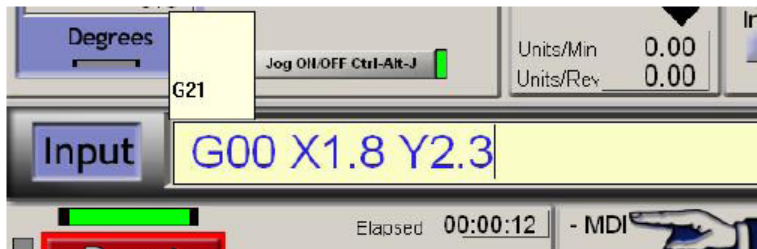


그림 3.4 MDI 에 명령어가 입력 되는 과정

변한 위치 만큼 읽혀진 것을 보라.

다른 여러 가지 명령어를 시도해 보라. MDI 창이 활성화 되어 있을 때 위, 아래 화살표 키를 누르면 Mach3는 과거 사용 했던 명령어들을 보여 주며 선택 할 수 있게 해 준다. 이것은 다시 입력 하는 것 보다 쉬운 방법이다.

MDI 라인에는 몇 개의 명령어들을 연이어 쓸 수 있으며 실행 방법은 10장을 참고 하라. - 항상 명령이 순차적으로 진행되는 것은 아니다. 예를 들어 피드 비율을 F2.5 라고 설정했다면 이 명령이 앞에 있던 뒤에 있던 어떤 다른 속도 제어에 관한 명령에 앞서 이것을 먼저 수행하게 될 것이다.

#### 3.4.2 티칭(Teaching)

Mach3는 MDI창에서 작성한 명령어들을 순차적으로 기억 할 수도 있고 파일로 저장 할 수 도 있다. 이것은 G-코드 프로그래밍 되어 반복적인 실행을 가능 하게 한다.

MDI 화면상에서 Start Teach 버튼을 클릭해 보라. 녹색 LED가 점등되어 티칭모드에 있다는 것을 알린다. MDI에 명령어를 계속 입력 하고 Enter 키를 누르면 Mach3는 곧 바로 실행에 옮기며 이것을 정해 놓은 티치(Teach) 파일에 저장을 한다. 끝 났다면 Stop Teach 버튼을 누른다.

직접 해 보거나 아래를 실행해 보아라.

G21  
F100  
G1 x10 y0  
G1 x10 y5  
X0  
Y0

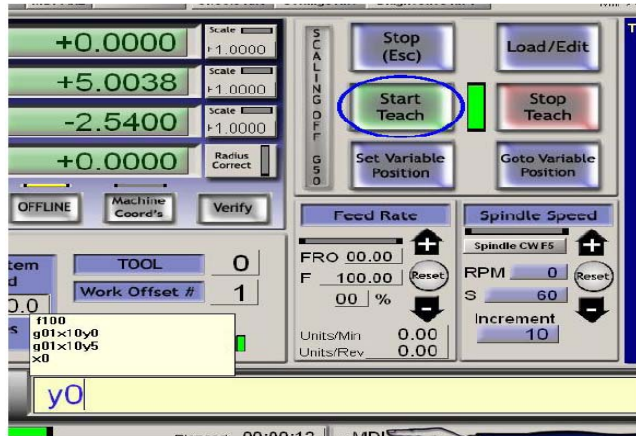


그림 3.5 사각형 모양을 티칭 하는 중

여기서 0은 숫자 영이다. 다음으로 Load/Edit를 클릭한 후 Program Run 화면으로 전환한다. 그러면 입력했던 명령어 등이 그림 3.6과 같이 G-code 창에 표시된다. 이때 Cycle Start를 누르면 Mach3는 프로그램을 실행한다. 편집기를 사용 하면 언제든지 정정이 가능하며 선택한 파일로 저장이 가능 하다.

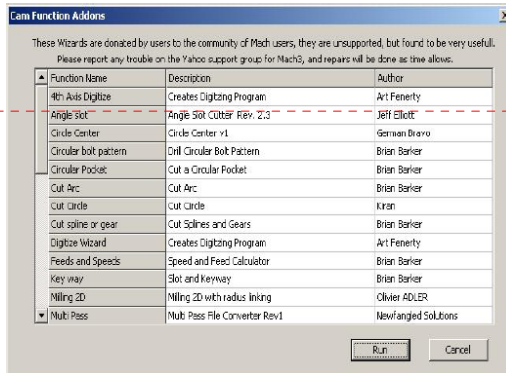
메모 [DP3]: 여기에서는 Windows 의 보조프로그램 중 메모장 프로그램도 가능



그림 3.6 티칭된 프로그램 실행

### 3.5 마법사(Wizards) 기능 - CAM 소프트웨어 없이 할 수 있는 여러 작업

Mach3에는 add on 기능이 있어 사용자로 하여금 관련 정보와 함께, 복잡한 업무를 자동으로 수행하게 할 수 있게 해준다. 이런 것들은 다른 Windows나 타 프로그램 에서 볼 수 있는 마법사 기능과 비슷하다. 전통적인 Windows 마법사 프로그램들은 데이터 베이스나 스프레드쉬트 등에서 파일 등을 끌어 들어와 진행한다. Mach3 마법사는 예를 들어, 원형 포켓팅, 격자 구멍 내기, 모델의 표면을 디지털링 등을 포함 하고 있다.



메모 [DP4]: 이 기능의 완전한 사용을 위해서는 추가적인 라이선스 구매가 요구된다. [www.machsupport.com](http://www.machsupport.com) 참고.

그림 3.7 마법사 테이블 메뉴

선택하는 것은 어렵지 않다. Mach3의 Program Run 화면에서 Load Wizard 를 클릭 한다. 그러면 그림 3.7 과 같이 설치된 마법사가 나타날 것이다. 예를 들어 Circular pocket Run 을 클릭해 본다. 그러면 아래 3-8과 같은 화면을 볼 수 있을 것이다. 자세히 보면 사용할 단위와 포켓의 중심위치와 어떻게 대상물에 접근해 나가야 하는지 그리고 그 이외의 기타 정보를 입력 하게 한다.

모든 입력 아이템이 여러분의 장비에 똑 같이 적용 되는 것은 아닐 것이다. 예를 들어, 스피들의 속도를 수동으로 설정 할 수도 있을 것이고 그런 경우에는 마법사 화면에 있는 제어기능들을 무시 해도 된다.

설정이 완성 됐다면 Post Code 버튼을 눌러 보라. 이것은 G-code 파트 프로그램을 작성하여 Mach3로 옮겨 놓는다.

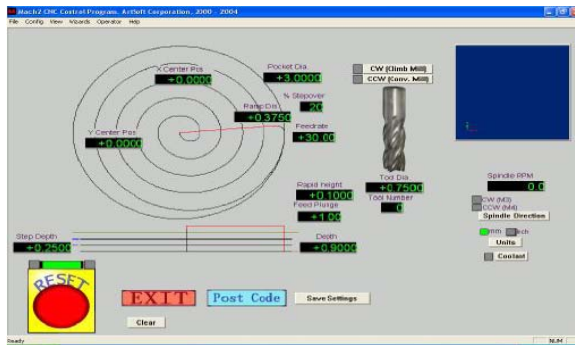


그림 3.8 원형 포켓 기본 화면

먼저 배웠던 티칭의 예제가 메뉴얼 작업이었다면 이것은 몇 개의 데이터 값 입력으로 자동화 할 수 있는 작업이다.

툴패스 화면은 절삭할 모양을 보여 준다. 파라미터 값을 조절하여 다시 Post 코드를 실행 할 수도 있다. 필요에 의해서 현재의 값을 마법사에 저장 했다가 다음에 다시 실행할 때 초기 값으로 현재의 구성을 재연 할 수도 있다. EXIT를 누르면 Mach3의 메인 화면으로 돌아가서 마법사가 만든 프로그램을 실행 시킬 수 있다.

이런 마법사의 실행이 때때로 이곳에서 기술 하는 것 보다 빠르고 쉬울 수 있을 것이다.

**메모 [DP5]:** 단순한 포켓팅이나 원, 호, 사각 모양 드릴링 등등을 자동으로 실행할 때 아주 유용한 툴이라 사용을 적극 권장 한다.

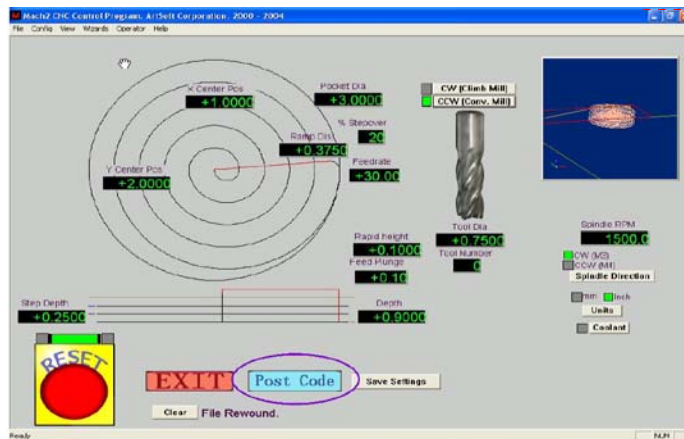


그림 3.9 값을 넣은 원형 포켓 화면



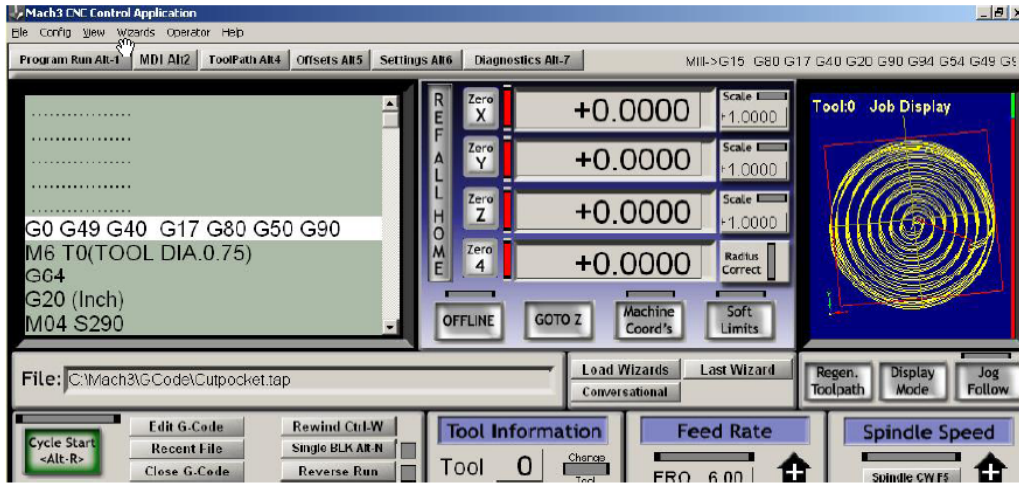


그림 3.10 - 원형 포켓팅 을 위한 프로그램

### 3.6 G 코드 프로그램 실행하기

자, 이제 파트 프로그램을 입력하고 편집해 보자. Mach3에서도 편집을 할 수 있지만 손쉽게 사용 할 수 있는 Windows의 보조프로그램 중에 하나인 메모장을 이용하여 아래의 내용을 텍스트 파일로 저장을 한다. (예 spiral.txt).

```
G20 f100
G0 x1 y0 z0
G3 x1 y0 z-0.2 i-1 j0
G3 x1 y0 z-0.4 i-1 j0
G3 x1 y0 z-0.6 i-1 j0
G3 x1 y0 z-0.8 i-1 j0
G3 x1 y0 z-1.0 i-1 j0
G3 x1 y0 z-1.2 i-1 j0
M00
```

다시 이야기지만 0은 숫자 영을 표시 한다. M0 후에 Enter 키 누르는 것을 잊지 말아라. Mach3 메뉴 바에서

File>Load G-code를 선택하여 이 프로그램을 실행한다. G-code 창에 위의 내용이 보일 것이다. Program Run 화면 상에서 Start Cycle, Pause, Stop 그리고 Rewind 버튼과 단축키를 눌러 어떻게 진행 되는지 보라.

프로그램이 진행됨에 따라 하이라이트 된 선이 움직이는 것을 볼 수 있을 것이다.

Mach3는 톨패스가 느리게 동작하는 것을 막기 위해 움직임을 실행하기 전에 먼저 읽어 드린다.

Pause를 누르면 이런 후 과정이 반영이 되어 나타난다.

G-code라인 중 어디로든 움직여서 Run from here 버튼을 누르면 바로 그 라인에서 진행 하게 된다.

**주의 :** 이 프로그램 파일들은 컴퓨터의 하드 디스크에서 돌아 가야 한다 플로피나 USB 드라이브는 속도가 상대적으로 느려 Mach3가 고속으로 접속 할 수 없다. 그리고 파일은 읽기 전용 파일이어서도 안 된다.

### 3.7 툴패스(tool path) 표시 화면

#### 3.7.1 툴 패스 보기

Mach3가 처음 실행 될 때 Program Run 화면에 텅 빈 화면이 보일 것이다. 예제 중 나선 프로그램을 실행하면 그 화면 안에 원이 나타나는 것을 볼 수 있을 것이다. 이것은 윗면을 보는 것이 된다.(즉, 위에서 아래로 XY 면을 보는 것이다)

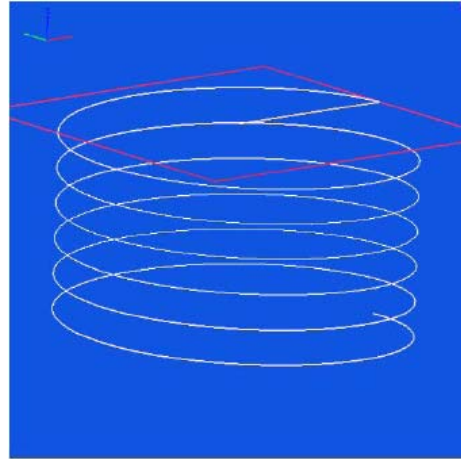


그림 3.11 툴 패스 예제 (spiral.txt)

보이는 화면은 선으로 표현이 되고 이 선은 툴이 구의 공간 내부를 따라 움직이게 될 진행 경로가 된다.

마우스를 화면에 놓고 구형체를 끌어서 돌려 가며 여러 각도에서 물체를 볼 수도 있다. 상단 좌측의 축들은 X, Y Z 중 어떤 축이 움직이고 있나를 보여주며 마우스를 중앙에서 위로 끌어 당기면 구형체는 사실 나선형을 그리며 절삭 되어 가는 것을 알 수 있다. 각각의 G3 라인은 나선형 프로그램에서 원을 그리며 동시에 Z가 0.2 만큼 아래로 내려 가는 것을 보여 준다. 초기의 G00는 직선움직임이 된다. 몇 번 그렇게 시도 하다 보면 무엇을 만들고 있나를 손쉽게 알 수 있을 것이다. 그림 3.11에 나타난 화면의 색과 장비에 셋업된

프로그램과 차이가 있을 수 있는데 그것도 설정 가능 하며 5장을 참고 하기 바란다.

#### 3.7.2 툴 패스 화면 확대, 축소 그리고 끌어 당기기

툴패스 화면은 shift 키를 누름과 동시에 커서를 화면에서 끌어 내리거나 올리면서 축소 확대 시킬 수 있다. 그리고 오른쪽 마우스 버튼을 누른 상태에서 Windows 화면을 끌어 당기면 화면의 그림이 전후 좌우로 움직일 것이다.

툴패스 화면을 더블 클릭 하면 처음 수직 아래 면으로 돌아 가며 축소나 확대 했던 이미지도 원 상태로 돌아 간다.

**주의 :** 장비가 운영 중 일 때는 당기거나, 확대 및 축소를 할 수 없다.

### 3.8 그 밖의 항목

마지막으로 마법사 다른 기능들을 둘러 보고 아래 유용한 기능들을 보면서 직접 기능을 익히기를 바란다.

- o 장비가 실행하는 파트 프로그램을 끝내는 데 걸리는 시간을 측정 하기 위한 버튼.
- o 선택한 파트 프로그램에서 피드 비율을 반복 할 수 있는 기능
- o 파트 프로그램의 모든 축으로 툴의 움직임을 확장 할 수 있게 하는 DRO

- 축을 나타내는 DRO 좌표계를 보여주는 버튼들. 차후 7장에서 설명을 한다.
- X, Y축 위치나 Z축을 안전하게 쓸 수 있게 정보를 설정 할 수 있도록 보여주는 화면.
- Mach3 의 모든 입출력을 로직 레벨(0과1)을 모니터 할 수 있도록 보여주는 화면

## 4. 하드웨어의 문제 및 장비와의 연결

이번 장은 하드웨어 연결을 설명 한다. 5장에서는 Mach3와 장비의 연결에 관한 자세한 내용을 설명 할 것이다. Mach3를 소프트웨어로 운영되는 장비를 구매 했다면 이번 장은 읽지 않아도 된다. 왜냐하면 공급자가 이미 시스템을 어떻게 구성 해야 하는지 설명서를 주기 때문이다.

이 장에서는 Mach3가 어떻게 제어하는지 그리고 마이크로 스위치나 스텝퍼 모터를 어떻게 연결 하는 지를 설명할 것이다. 단, 여러분이 간단한 회로도 는 이해를 할 수 있다고 간주 할 것이다. 그렇지 않다면 주변으로부터 도움을 요청 하기 바란다.

### 4.1 중요 안전사항



어떤 장비던 위험성은 내제되어 있다. 이 매뉴얼은 여러분의 안전에 대한 사전예방 및 기술을 설명 하고 있지만 여러분 각 장비에 대한 세부 사항은 알 수 없는 관계로 장비의 효율이나 어떠한 파손 및 상해에 대한 책임을 질 수 없다는 것을 명시 하는 바이다. 그리고 사용자의 해당 국가나 지역의 법률에 따라 여러분이 각자가 설계 하고 만드는 것에 대한 책임이 있다는 것을 잘 염두 해 주기 바란다.

혹시라도 의심 가는 부분이 있다면 반드시 전문가에게 가서 상의 하라고 권고 하는 바이다. 그것이 본인이나 다른 사람에게 위험으로 인한 피해를 주는 것 보다는 백 번 낡지 않겠는가?!

### 4.2 Mach3가 할 수 있는 일

Mach3는 상당히 유동성이 있는 프로그램으로 밀링머신과 터닝 머신 같은 장비들을 운영 할 수 있다. Mach3 에 의해 운영되는 장비들의 특징으로는

- 사용자가 제어 할 수 있게 긴급 정지 버튼(Estop)은 반드시 준비되어야 한다.
- 2개나 3개축이 서로 90도의 각으로 보고 있다(여기서는 X, Y, Z 축 상호관계).
- 작업 대상물에서 상대적으로 움직일 수 있고 작업 물에 따라 원점이 설정된다. 상대적 움직임이란 물론 톨이 움직이는 것을 말하거나. (예 : 밀링머신의 퀴(quill) Z축을 따라 움직일 때) 테이블이 움직이는 것을 말하기도 한다.

추가적으로

- 홈 포지션을 알려주는 스위치가 있거나.
- 장비의 상대적 움직임 범위를 알려 주는 리미트 스위치가 있거나.

- 스피들 제어. 스피들이 밀링 장비에서 툴을 움직이거나 아니면 작업 물이 터닝 장비에서 돌아 간다.
- 3개의 추가적인 축을 사용 한다. 이런 것들은 로터리방식이나 리니어 방식으로 정의 된다. 추가적인 하나의 축은 ,Y 혹은 Z 축에 슬레이브로 사용 될 수 있고 이때 두 개가 하나의 축으로 함께 움직이지만 각각 설정을 따로 잡는다. ( 슬레이브 축 설정 하기 를 참고)
- 기계의 보호 장치를 위한 인터락 스위치
- 냉매제(coolant) 운영 방식 제어
- 장비에 놓여 있는 물체를 디지털이징 하는 프로브 사용
- 장비에서 툴의 위치를 알려 주는 선형 인코더(linear glass scale) 같은 인코더
- 특수 기능 포함.

Mach3와 Mach3 가 운영되는 컴퓨터 는 프린터 포트에 연결된다. 단순한 장비라면 오직 하나의 포트만 필요 할 것이다. 하지만 복잡한 장비라면 2개를 필요로 할 수도 있다.

LCD 디스플레이나 툴 체인저, 그리고 컨베이어 같은 특수 기능들은 Modbus 장치를 통해서 사용 할 수 있다.

버튼은 입력신호에 대응 하여 가상 키 누름을 할 수 있는 키보드 에뮬레이터와 연계될 수 있다.

Mach3는 총 6개의 축을 동시에 제어 할 수 있는데, X, Y, Z축을 제외한 2개의 축 만 회전 운동 시킬 수 있다. 나머지는 모두 선형 운동이 되어야 한다. 쉽게 말해 Mach 소프트웨어는 X,Y,Z를 제외한 2개 이상의 의 축을 초과하는 원형 운동 즉 로봇 팔이나 6축 보행 로봇 과 같은 운동역학 계산이 필요한 기기들은 지원 하지 못한다. 이런 동작 중의 피드 비율은 파트 프로그램에서 정한 값으로 유지 되며 이때 축에 가해지는 가속도와 최대속도는 제한 되어 있다. 또한 다양한 조깅 제어로 수동으로 축을 움직일 수 도있다.

Mach3가 할 수 있는 일은, 스피들을 on/off 또는 회전 방향을 제어 할 수 있으며 RPM을 제어 할 수 있고 나사 절삭과 같은 각도 위치를 모니터링 할 수 도있다.

Mach3는 두가지 타입의 냉매제를 켜다 끌 수 있다.

Mach3는 Estop, 인터락, 기준점 스위치등의 작동을 운영자에게 보고 해 준다.

Mach3는 256가지의 각각 다른 툴(엔드밀)에 대한 속성을 저장 할 수 있다. 하지만 장비가 자동 툴 체인저를 장착하고 있다면 그것은 운영자가 별도로 제어를 해 주어야만 한다.

### 4.3 EStop(Emergency Stop) 컨트롤

모든 장비에는 하나 이상의 EStop 버튼이 있어야 한다. 이것은 보통 빨간색으로 버섯 머리 모양을 하고 있으며 어느 곳에서도 쉽게 누를 수 있는 위치에 있어야 한다.

각 EStop 버튼은 눌러짐과 동시에 모든 것이 즉각 정지 되어야한다 : 스피들은 회전을 멈추고 각 축의 동작도 멈춘다. 이때, 소프트웨어는 장비를 제어할 수 없다.

전원을 차단 하는 것으로 Estop을 대체 하는 것은 충분치 않다. 왜냐하면 회로의 콘덴서에 남아 있는 전기가 버튼을 누른 후에도 모터를 천천히 움직이게 할 수 있기 때문이다.

장비는 리셋 버튼이 눌러지기 전까지 시작되어서는 안 된다. 만약 Estop 버튼이 눌러 졌다면 버튼의 머리를 회전시켜 작동 상태로 놓여지기 전까지 장비가 작동 되어서는 안 된다.

이때 작업중인 파트는 일반적으로 계속 해서 완성 하긴 거의 불가능 하지만 적어도 안전은 지킬 수 있다.

#### 4.4 PC의 페러렐 포트(프린터 포트, 병렬포트).

##### 4.4.1 페러렐 포트의 역사

처음 IBM이 PC를 만들었을 때 25핀 짜리 연결 장치를 제공 하였다. 이것이 지금까지 우리가 PC에 사용하고 있는 병렬포트의 시조가 된 것이다. 데이터 전송이 용이하여 프린터를 제외한 다른 여러 장치도 이것을 이용 하고 있다. 예를 들면 PC 끼리 파일을 전송 한다든지 스캐너나 zip 드라이버와 같은 주변장치를 연결 할 때 그리고 장비의 제어를 위해서도 사용 한다.

현재 USB가 점차적으로 대체 하고 있어 Mach3도 USB 포트를 사용 할 수 있게 하려 한다.

PC는 25핀 암컷 D 컨넥터다. 옆 그림 4.1 과 같다. 화살표들은 정보 전달 방향을 나타낸다. 한 예로 15번 핀 은 PC로 들어가는 입력을 가리킨다.

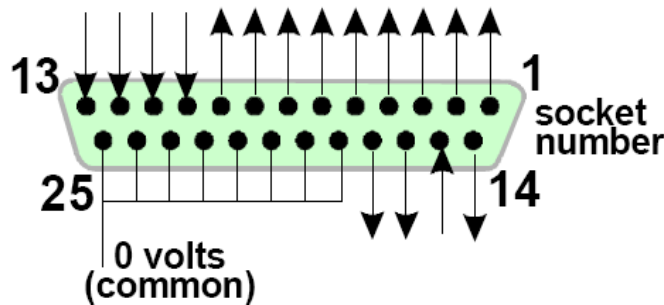


그림 4.1 - PC 뒷부분의 암컷 병렬포트

##### 4.4.2 논리 신호

이번 섹션은 읽으면서 지루함을 느낄 수 있지만 원론적인 인터페이스 회로를 접하면서 다시 여기로 돌아 올 수도 있을 것이다. 스텝모터드라이버 보드의 설명서와 이것을 함께 읽는 것이 때로는 유용 할 것이다.

Mach3의 모든 입출력 신호는 이진수다. 이런 신호들은 병렬 포트의 입력핀 이나 출력핀에 제공되는 전압에 의한 것으로서 컴퓨터의 0 볼트 선에서 상대적으로 측정되는 값이며 이때, 프린터 포트의 18번 에서 25 핀 까지 연결된 선에 해당 된다.

처음 성공적인 직접회로(74xx 시리즈)는 TTL(transistor-transistor logic)을 사용 하였다. TTL 회로는 0V ~ 0.8V 사이의 전압을 lo(로) 라고 부르고 2.4 V ~ 5V 사이의 전압을 hi(하이) 라고 부른다. 5볼트 이상 전압을 흘리거나 극을 바꿔 연결 하면 회로에서 연기가 날 것이다. 병렬 포트는 원래 TTL방식으로 설계 되었으며 전압은 lo 와 hi 신호로 정의 된다. 알아 두어야 할 것은 두 신호 사이의 최소 값은 1.6 볼트라는 것이다.

물론 lo 가 0 이나 1을 상정 한다고 하는 것은 독단적인 생각일 것이다. 하지만 아래의 내용과 같이 lo = 1 이라고 하는 것이 대부분 인터페이스 회로에서 옳다고 봐야 한다.

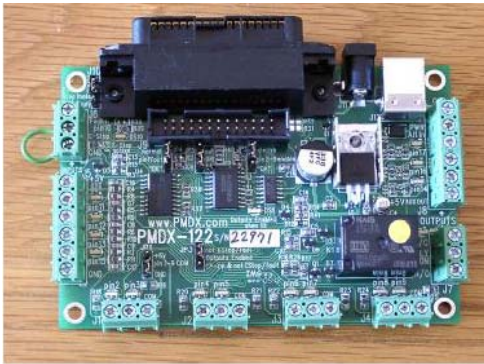
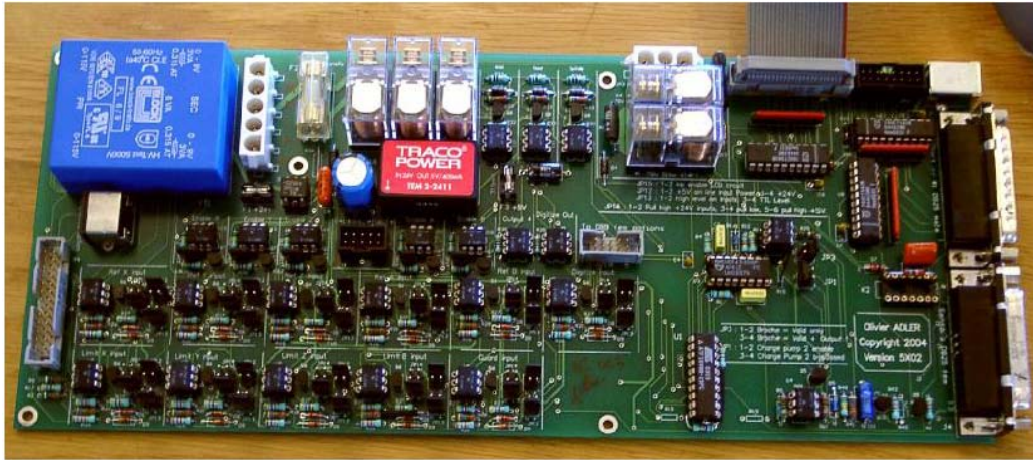
출력신호가 어떤 일을 하기 위해서는 연결된 회로에 이 신호가 흘러야만 한다. “ hi” 일 때는 컴퓨터에서 전류가 흘러 나간다. “ lo” 일때는 컴퓨터로 전류가 흘러 들어 온다. 전류가 더 많이 흘러 들어 올 수록 0 볼트 근처로 유지 하기가 힘들어 한계치 인 0.8V 까지 근접 하게 될 것이다.비슷한 이유로, 흘러 나가는 hi는 최소 한계인 2.4볼트 근처까지 가게 만들 수 있다. 그러므로 과도한 전류는 lo와 hi를 최소 요구값인 1.6 볼트 경계선을 넘어 회로를 불안정 하게 만들게 된다. 마지막으로, 흘러 나가는 hi 보다 대략 20배 더 높은 lo를 인가 하는 것이 더

무모한 일이다. 결국 이것이 의미 하는 것은 1을 lo 신호로 배정하는 것이 최선이란 것이다. 이것을 활성 lo 로직 이라고 부른다. 하지만 이것의 단점은 포트에 연결된 장치가 5 볼트 전압을 공급해 줘야 한다는 것이다. 이것은 가끔 PC게임 포트 소켓 이나 장치에 연결된 전원 공급 장치에서 추출 된다.

입력 신호로 돌아가, 컴퓨터는 hi 입력을 위해 40uA(micro Amps)보다 낮은 전류를 공급 받아야 하고 lo입력을 위해 0.4mA 보다 낮은 전류를 공급 해 줘야 한다.

왜냐하면 현재의 컴퓨터 마더보드들은 병렬 포트를 포함한 여러 복합 기능이 hi, lo 룰에 따르는 한 칩에 내제 되어 있기 때문이다. 지금 장비가 옛날 컴퓨터에 작동 한다면 컴퓨터를 업그레이드 할 때 장비가 번덕스럽게 작동 하는 것을 볼 수 있을 것이다. 2번에서 9번 핀은 같은 속성을 가지고 있다.(프린트 할 때 데이터 핀으로 사용 된다) 1번 핀도 프린트 할 때 필수 적이지만 다른 핀들은 사실 최적화된 디자인이라면 적게 사용된다. 잘 만들어진 브레이크 아웃 보드(다음 그림 참조) 는 전기적 호환성의 문제를 최소화 할 것 이다.

**4.4.3 전기적 노이즈와 보드의 과열**



**그림 4.2- 시중 판매 하는 브레이크 아웃 보드들의 예.**

이전 섹션은 지나 칠 수 있지만 이번 섹션은 읽어 두는 것이 바람직 할 것이다. 그림 4.1에서와 같이18번에서 25번 까지의 핀은 컴퓨터의 전원 공급 장치 0볼트쪽에 연결 되어 있다. PC 안팎의 모든 신호는 이것과 연관이

있다. 만약 높은 전류가 흐르는 모터에 다수의 긴 배선을 연결했다면 그 선들로부터 흐르는 전류로 인하여 전압차가 발생 할 수 있고 이것은 노이즈와 같은 현상을 일으켜 문제를 발생시킬 수도 있을 것이다. 이것이 심하다면 컴퓨터에도 무리를 줄 수 있을 것이다.

병렬 포트에 연결된 Mach3를 사용하여 축과 스피들을 제어 한다면 30~240V로 작동이 될 것이고 많은 전류를 사용 할 것이다. 적절히 연결된 상태라면 컴퓨터에 손상을 주는 일이 없지만 잘못 접지 했을 때는 컴퓨터의 마더보드나 CD-ROM 그리고 하드디스크까지 파손 시킬 수 있다.

이러한 이유로 여러분에게 격리된 브레이크아웃 보드(Isolating breakout board)를 사용 할 것을 권유한다. 이것은 쉬운 연결을 위한 터미널을 제공 하며 드라이브 홈, 리밋 스위치를 위한 0V 가 따로 제공 된다. 또한 포트에 흐르는 과도한 전류의 입출력을 차단 한다. 전원 공급 장치와 드라이버 보드 와 같은 단일 보드는 주위의 라디오나 TV등의 시그날에 영향을 주지 않기 위해 금속 상자에 장착하는 것이 좋다.

## 4.5 축 드라이버의 선택 항목

### 4.5.1 스텝모터와 서보 모터

축을 움직이는 모터는 크게 두 가지 타입이 있다.

- 스텝 모터
- 서보 모터(AC 나 혹은 DC)

둘 중 어느 것이든 리드(혹은 볼)스크류, 벨트, 체인 혹은 렉&피니언을 통해 축을 이송 할 수 있게한다. 기계적 운송 방식은 속도와 필요한 토크를 결정하며 그에 따라 모터와 장비 사이의 필요한 기어링을 결정한다.

스텝모터 드라이버(Stepper motor)의 속성은

1. 저 예산
2. 단순한 4선, 6선 혹은 8선 결선
3. 적은 유지보수
4. 모터의 속도는 대략 1000rpm 그리고 토크는 3000 oz.in(21Nm) 정도로 까지로 제약이 있다. 최대 속도는 모터의 회전과 모터를 운영하는 회로에 공급 되는 최대 허용 볼트에 좌우 된다. 최대 토크는 모터의 최대 허용 전류에 좌우 된다.
5. 장비는 마이크로 스텝 컨트롤러를 사용할 수 있어 어떤 속도에서든 실용적이고 부드러운 운영 효과를 볼 수 있다.
6. 오픈 루프 방식 이기 때문에 가해지는 힘이 많을 때 탈조가 발생 할 수 있으며 이때 장비의 운영자는 즉각적인 반응을 살필 수 없을 수도 있다.



**메모 [DP6]:** 장비를 설계함에 있어, 필요한 토크, 전압, 전류를 잘 생각해 보기 바란다. 탈조를 막기 위한 중요 조건중 하나가 된다.

**메모 [DP7]:** 인코더를 장착 하면 closed loop 이 되어 서보와 같이 위치 추적, 제어가 가능 하다.

서보모터는(servo motor)

1. 상대적으로 고가이다(특히 AC 모터 일 경우)
2. 모터와 인코더 양쪽 모두 배선작업이 필요 하다.
3. DC 모터에서는 브러시의 유지보수가 필요 하다.



4. 모터는 4000rpm 이상의 속도를 낼 수 있고 실질적으로 무제한의 토크를 가질 수 있다(경제력이 허락한다면)

5. Closed loop 방식이라 드라이브 된 위치는 언제나 정정 가능 하다.(인코더로부터 위치에 대한 피드백이 전해 진다)

일반적으로 스텝 모터는 초고속 초정밀도를 요구 하지 않는 이상 충분히 만족할 만한 결과물을 만들 것이다.

여기서, 두 가지를 말 하고 넘어갈 것이 있다. 첫째로 옛날 장비의 서보 시스템은 디지털 방식이 아닐 것이라 본다. 말하자면, 일련의 스텝 펄스와 방향신호로 제어되는 것이 아니다. Mach3로 이것을 제어 하려면 위치를 알려주는 리저버는 무시해야 할 것이며 사체배(수직위상) 인코더(quadrature encoder)를 장착하고 또한 모든 전자제어 부분을 교체 해야 할 것이다.

둘째로 제조자의 데이터가 불 분명한 중고 스텝 모터의 사용을 조심하라. 그것들은 5상 모터 일 수도 있고, 요즘 나오는 모터와 호환이 안 될 수도 있으며, 요즘 나오는 모터 보다 훨씬 낮은 토크를 가질 수도 있다. 실제로 테스트 해 보지 않는 이상, 사고로 손상이 된 것이어서 못 쓰는 것일 수도 있을 것이다. 만약 전문적인 기술과 경험이 있지 않는 이상 모터 같은 것은 새것으로 구매 하기 바란다. 그리고 그렇게 한번 제대로 된 것을 구매 했다면 다시 또 사지 않아도 될 테니 말이다.

#### 4.5.2 축의 움직임에 필요한 요소 계산하기

축을 움직일 수 있는 총괄적 계산은 매우 복잡할 것이며 아마도 필요한 모든 데이터를 갖고 있지 않을 수도 있다.(예: 절삭 하려고 할 때 필요한 최대한의 힘은 얼마인가?) 하지만 성공을 위해서는 반드시 필요하다.

이번이 매뉴얼을 다시 읽는 것이라면 다음 섹션으로 넘어가도 좋다.

예제 : 밀링 테이블 Y축 이송

움직일 수 있는 최소 거리부터 점검해 보자. 이것은 장비에서 이루어 질 수 있는 작업의 정밀도에 대한 절대 한계를 나타낸다. 그리고 난 후 속도와 토크를 측정할 것이다.

예제는 여러분이 Y축 이송장치를 디자인 한다고 가정 한다. 이제 여러분은 0.1 인치(대략 0.2mm) 피치의 스크류를 사용 하며 최소 거리를 0.0001 인치(대략 0.0002mm)를 움직이는 것으로 만들고 싶다면 스크류가 모터 샤프트에 1:1 비율로 물려 있을 경우 이때 모터는1/1000 바퀴 회전이 되어야 할 것이다.

스텝모터를 이용하여 이송 하기

스텝 모터의 최소 스텝은 어떻게 제어 되는 가에 달려 있다. 보통 한 바퀴 회전은 200 스텝(step)으로 구성 되어 있다. 때문에 위의 예제를 위해서는 마이크로 스텝 컨트롤러가 필요 할 것이며 많은 보드가 200 스텝인 풀 스텝(Full step) 대비 10 마이크로 스텝을 지원한다. 이것을 적용하면 1/2000 회전을 가능 하게 해 준다.

메모 [DP8]: 보통 한 스텝에 1.8도 움직인다.

다음으로, 가능한 피드 속도를 보자. 최대 모터 속도가 500rpm 이라고 가정해 보자. 이것은 분당 50인치(대략 120cm) 혹은 15초의 속도로 축의 양 끝을 운행 한다고 가정한다. 엄청난 정도는 아니더라도 쓸 만 하다고 본다. 이 속도라면 마이크로 스텝 모터 드라이브는 초당 16,666(500X200X10/60) 펄스가 필요로 할 것이다. 1 GHz 의 PC에서 Mach3는 초당 35,000 펄스를 만들어 낼 수 있으며 총 6개의 축까지 동시에 펄스를 생성 할 수 있다. 그러므로 여기엔 문제가 없다.

자 그럼 여기에서 장비가 필요로 하는 모터의 토크를 알아야 한다. 이것을 측정하는 하나의 방법은 장비에 절삭 하기 가장 힘든 물체를 놓고 12인치 정도 되는 지렛대의 한 쪽끝을 이송 스크류의 손잡이 중심 부분에 고정 하고 지렛대의 다른 쪽 끝에 저울을 걸어 돌리면서 각일 때 읽히는 무게를 측정 하여 계산 한다. 절삭 할 때 필요한 토크는 저울의 눈금값X12인치가 된다.

다른 방법으로는 기성품에 달려 있는 모터 싸이즈를 보고 따라 하는 것이다.

적절한 출력 속도를 잘 고려 해야 하며 2:1 기어를 사용하여 속도를 줄이고 토크는 2배로 증가 시킬 수도 있다.

서보 모터를 이용한 이송 장치

우리는 한 스텝 사이즈가 얼마 정도 되는지 보았다. 서보 모터는 인코더가 장착되어 있어 위치 파악을 할 수 있다. 인코더는 구멍이 난 디스크로 되어 있고 한 구멍을 지날 때 마다 사체배(수직 위상) 펄스(quadrature pulse)를 만들어 낸다. 그러므로 300개의 구멍이 있는 디스크라면 한 바퀴 회전 할 때 300번의 펄스를 생성할 것이다.(CPR: Cycle Per Revolution) 이것은 상업용 인코더에 있어서는 상당히 낮은 숫자이다. 이때 인코더 회로는 모터의 한 회전당1200번의 수직 위상 출력을 한다. (QCPR : Quadrature Count Per Revolution) 서보의 드라이브 회로는 모터를 입력 스텝 펄스(input step pulse) 당 한번의 수직 위상 출력으로 회전 시킨다. 몇몇 고가의 서보 회로는 스텝 펄스를 상수로 공급하거나 나눌 수도 있다. 이것을 종종 전기적 기어링 이라고 부른다.

서보 모터의 최고 속도가 보통 4000rpm 이 됨에 따라 기계적인 감속이 필요 한데 여기서 대략 5:1이 적절해 보인다. 이것은 스텝당 0.0000167 인치의 거리를 움직이게 한다.

최대 속도를 얼마큼 만들 수 있을까? 초당 35,000 스텝 펄스로 우리는 리드스크류를 초당 5.83 회의 회전을 얻었다(35000/(1200X5)) 이것은 9초 동안 5인치를 이송 하는 속도다. 알아야 할 것은, 여기에서 속도라는 것은 Mach3에 의한 펄스 비율이지 모터의 속도가 아니다. 때문에 이것은 이 예제에서 대략 1750rpm이 된다. 인코더가 회전 당 더 많은 펄스를 준다면 더 많은 제약을 받을 것이다. 만약 고감도 인코더를 사용 한다면 서보 회로에 전기적 기어링을 사용해서 이런 제한을 극복 할 수 있다.

마지막으로 가용한 토크를 생각해 보아야 한다. 이 문제에 있어서 서보의 경우 스텝 모터 보다 더 넓은 환경에 있다. 왜냐하면 서보는 탈조의 대한 걱정이 없기 때문이다. 만약 장비로부터 능력 이상의 토크를 모터에 가한다면 모터는 과열이 되거나 드라이버 보드는 과전류에 의해 누전이 발생 할 것이다.

#### 4.5.3 스텝(Step) 과 방향(Dir) 신호의 작동 원리.

Mach3는 한 펄스(로직 1)를 한 스텝으로 하며 이것은 모터가 움직이는 한 스텝과 같다.

Dir 출력은 스텝펄스가 생성 되기 이전에 형성될 것이다. 로직 파형은 그림 4.4와 같다.

펄스 사이의 간격은 스텝의 속도가 높을수록 좁아 진다.

드라이버 회로는 보통 스텝(Step)과

방향(Dir) 을 위해 Active Lo 로 설정을 한다.

이때의 설정을 따르기 위해 Mach3또한

Active Lo 로 설정해야 한다. 그렇지 않으면 스텝(step) 신호가 그림 4.5 와 같이

펄스 사이의 간격이 바뀌게 되어 모터가

제대로 동작을 하지 않을 것이다.

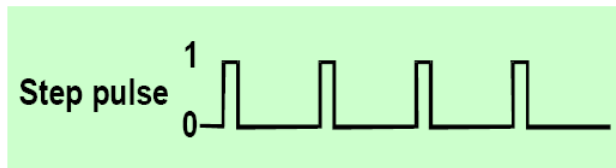


그림 4.4 스텝 펄스 파형

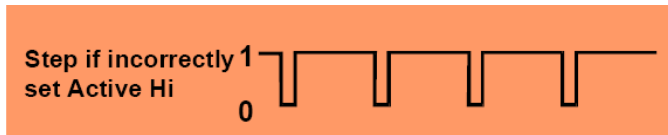


그림 4.5 잘못 설정된 스텝 펄스 파형

## 4.6 리밋 스위치, 홈(영점) 스위치

### 4.6.1 설정을 위한 방법

리밋 스위치는 선형 운동하는 축을 정해진 한계 이상으로 가 장비가 손상되는 것을 막기 위해 쓰인다. 이 스위치 없이도 장비를 사용할 수 있지만 작은 실수로 비싼 장비를 손상시킬 수 있다. 축에는 홈 스위치를 장착하는 것이 좋으며 Mach3는 모든 축을 홈으로 옮길 수 있다. 이것은 장비에 전원이 on 일때 마다 실행 될 필요가 있으며 이로써 현재의 위치가 어디 인가를 알려 준다. 만약 홈 스위치가 없다면 눈으로 시작이 될 만한 위치를 잡아야 한다. 축의 홈 스위치는 어느 좌표점이던 가능 하다. 그러므로 홈 스위치가 반드시 Machine Zero 좌표가 될 이유는 없다.

각 축에는 3개의 스위치를 쓸 수 있다.(예 : 양 끝의 리밋 스위치와 홈 스위치) 그러므로 3축 밀링 머신은 이 스위치들을 위한 총9개의 병렬 포트 입력핀을 필요로 할 것이다.

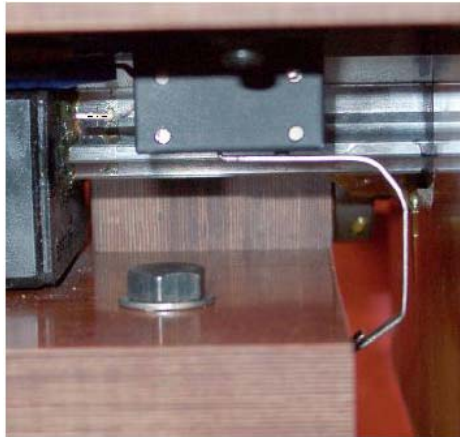


그림 4.6 리밋

그러므로 오직 5개의 입력핀을 가지고 있는 병렬 포트로서는 쉽지 않은 일이다. 그래서 이런 문제는 아래의 3가지 방법으로 해결 할 수 있다.

1. 리밋 스위치가 외부 로직회로(또 다른 드라이버 보드 회로)에 연결이 되고 이 로직은 리밋 스위치가 작동되면 회로를 차단 시킨다. 각기 분리된 스위치는 Mach3에 입력 신호를 준다.
2. 하나의 핀이 한 축의 모든 입력을 공유 하고 Mach3는 리밋의 제어와 홈위치 감지를 동시에 담당 한다.
3. 스위치들은 키보드 에뮬레이터로 연결 할 수 있다.

여기에서 처음 방법이 가장 좋은 방법이며 단지 소프트웨어의 설정으로 기계적 손상을 막을 수 없다고 생각하는 고가의 장비나 고속의 장비라면 반드시 사용 해야 할 방법이다. 드라이버 회로에 연결된 리밋 스위치는 눌렀을 때 동작을 정지시킬 수 있다. 이것은 최대 거리를 설정 하지 않는 것 보다 안전하며 사용자가 한계점으로부터 장비를 안쪽으로 조그 할 수 있으나 지원하는 드라이브가 있느냐에 따른다.

작은 장비라면 2번째 방법을 써도 된다. 이때 Mach2 는 3개의 축을 운영하는데 3개의 입력이 필요 하고 하나의 리밋과 홈 스위치를 공유함 으로 2개의 스위치가 필요로 한다.

키보드 에뮬레이터 는 병렬 포트 보다 훨씬 느린 반응을 보이지만 저속의 출력을 내는 장비의 리밋 스위치를 위해서 라면 충분하다. 자세한 아키텍처를 원한다면 Mach2 customization manual 을 참고하기 바란다.

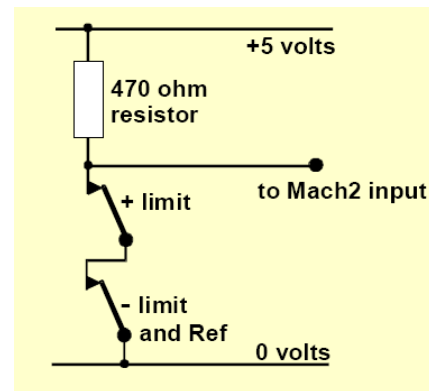


그림 4.7 두개의 스위치를 이용한 OR 로직

### 4.6.2 스위치

스위치를 선택 할 때는 몇 가지 고려 사항이 있다. 만약 두 개의 스위치로 입력을 공유 하려 한다면 두 개가 서로

연결이 되어서 둘 중 하나라도 작동이 될 때 로직 신호 1이 되어야 한다. 이것은 어렵지 않다. 만약 그림 4.7과 같이 연결이 되어 한 개라도 작동이 된다면 활성(Active) Hi 신호를 보낸다. 알아 두어야 할 것은 안정된 동작을 위해 병렬 포트의 입력을 순간 정지 해야 한다. 스위치에서 현저한 전류를 동반함에 따라 470R 값 정도 나타나는데 이는 대략 10mA 정도의 전류를 주게 된다. 스위치의 배선이 길어짐에 따라 노이즈를 탈 염려가 있으니 보호막이 있는 전선 케이블로 그라운드 터미널을 연결 하는 것도 고려해 볼 만 하다.(장비의 프레임에 연결하는 것은 좋지 않다)



4.8 광 센서 스위치 예제

만약 LED와 포토 트랜지스터를 이용한 스위치를 사용한다면 일종의 OR 게이트를 사용 해야 할 것이다.

광 센서 스위치는 금속 절삭 작업 시 냉매제(coolant)와 떨어져 있을 때 제대로 작동 할 수 있고 나무를 밀링 할 때는 먼지 때문에 고장을 일으킬 수 있다.

장비에 자석 스위치는 사용 해서는 안 된다. 금속을 절삭 할 때 발생하는 파편들이 자석에 붙을 테니 말이다.

스위치에 대한 작동 재연성은 스위치의 질적 우수성, 열만큼 튼튼하게 장착되어 있는가, 그리고 스위치 작동 높이에 따라 결정 된다. 그림 4.6의 예는 정밀도가 떨어지는 경우다. 영점을 잡을 때는 스위치의 재연성이 아주 중요하다.

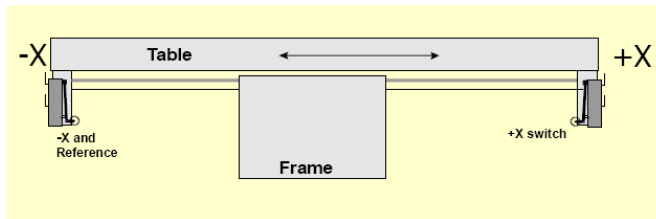


그림 4.9 기계적 정지로 과잉 이송을 방지 한다.

과잉 이송이란 스위치가 작동된 후의 움직임을 말하는데 이것은 축의 움직임에 대한 관성 때문에 생길 수 있다. 그림 4.7의 광 스위치 같은 경우 스위치를 작동시키는 긴 깃발이 있을 경우라면 이런 문제는 없을 것이다.

#### 4.6.3 스위치의 설치 위치

스위치의 설치 위치에 대한 선택은 절삭 시의 파편이나 먼지로부터 떨어진 곳에 설치 하고 잘 휘어지는 배선을 이용해서 설치 한다.

예를 들어 그림 4.6 과 4.8과 같이 스위치는 움직이는 케이블이 있어야 함에도 불구하고 둘 다 테이블의 아래에 위치하여 잘 보호되고 있다.

자 이때 하나의 케이블 안에 여러 개의 축과 연결된 배선을 넣는 것이 간편 하다는 것을 볼 수 있다. 모터 선과 스위치 선을 복선 케이블로 공유 하는 것은 삼가 하라.

두 개의 분리된 케이블을 함께 놓고 사용 하고 싶을 수 있으나 이것은 양쪽 모두 보호막으로 싸여져 있을 때 문제가 발생하지



그림 4.10 X=0 Y=0 위치

않을 것이며 보호막은 드라이브 회로에  
그라운드 되어야 한다.

#### 4.6.4 Mach3 상에서 공유된 스위치의 사용 방법

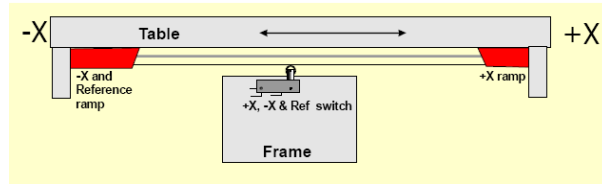


그림 4.11

일단 두 개의 리미트 스위치를 하나의 입력 핀에 연결한다. 그리고 기준 스위치가 있는 방향으로 축을 움직일 수 있도록 Mach3에서 설정해 준다. 축의 끝에 있는 리미트 스위치는 홈 스위치의 기능을 동시에 담당한다.

정상적 이라면, mach3 는 스위치가 작동될 때 축은 움직임을 멈추고 리미트 스위치가 작동 됐다는 메시지를 화면에 나타낼 것이다. 다시축을 움직이게 하려면 다음을 참고 하라.

1. Setting 화면(탭)에 있는 Auto limit override 버튼이 on 되어 있을 때. 이런 경우 Reset 버튼을 누르고 스위치로부터 약간 뒤로 물러 나도록 조그 한다. 그리고 난 후 기준점을 설정 한다.
2. Override limits 버튼을 클릭한다. 화면상에 빨간색 LED 가 깜빡이는 것이 보일 것이다. 이때 다시 Reset 버튼을 누르고 조그 하여 스위치로부터 떨어 지게 하면 자동적으로 깜빡임이 사라지게 된다. 다시 장비의 기준점을 설정해야 한다. 한번의 입력으로 리미트 스위치를 덮어쓰기(override) 하라고 정의 할 수 있다.

주의: Mach3 에서 조깅 하는 속도가 제한 되어 있지만 주의 하지 않으면 위의 두 경우 모두 축이 물리적 정지 장치에 충돌 할 수 있기 때문에 조심 하기 바란다.

#### 4.6.5 기준점 정하기

G 코드를 실행 하거나 버튼을 눌러서 기준점으로 축을 움직인다면, 주어진 속도로 홈 스위치 방향으로 이송 할 것이며 결국 스위치를 누르게 된다. 그리고 스위치가 작동 되면 이송하던 축이 반대방향으로 살짝 후진하여 스위치로 부터 떨어진다. 기준점을 설정 할 때는 리미트에 대한 개념이 적용되지 않는다.

축의 기준점이 설정되고 0 이나 혹은 Config-State 대화상자 저장된 어떤 값이 축 DRO 창에 장비의 절대 좌표 값으로 나타나게 된다. 만약 0으로 설정한다면 장비의 절대좌표 역시 0이 된다. 만약 기준점이 마이너스 방향으로 간다면, 마이너스 값이 DRO 창에 나타날 것이다. 이것이 의미 하는 것은 마이너스 값만큼 리미트로 부터 떨어져 있다는 것이다. 이로서 축의 이송 거리를 낭비 할 수도 있지만 홈으로 조그 할 때 실수로 리미트 스위치를 작동 시키지는 않을 것이다. 이 문제를 해결 하는 또 다른 방법으로 소프트 리미트를 생각해 보기 바란다.

만약 홈 스위치를 작동되게 한 후 살짝 뒤로 후진하게 (Jog off)하지 않으면 원하는 방향의 반대로 이송하게 될 것이다. 이때 별개의 홈 스위치가 있거나 축의 기준점 끝에 있다면 상관없다. 하지만 축의 반대편에 있고 Mach3가 리미트 스위치의 공유된 사실을 모른다면 홈 스위치를 넘어 충돌 할 때까지 이송하게 될 것이다. 때문에 기준점을 설정하기 위해서는 반드시 스위치에 닿아 작동 된 후 주의하여 뒤로 후진하라. 이런 문제로 걱정이 된다면 Mach3를 자동으로 후진 할 수 없게 설정 할 수도 있다.

#### 4.6.6 기타 홈 스위치, 리밋 스위치 옵션과 힌트

##### 리밋 스위치 근방에 있지 않은 홈 스위치

이송의 한계점에 홈 스위치를 두는 것이 편리 하지 않을 수 있다. 커다란 버티컬 밀링 머신이나 플레너 밀링 머신을 예를 들어 보자. Z 축 칼럼(기둥)의 이송이 8피드 정도가 되고 천천히 움직여 장비 전체의 절삭 작업에 무리를 주지 않는다고 한다. 하지만 홈 위치가 칼럼의 맨 위에 있으면 느린 동작으로 기준점을 찾기 위해 16피드 정도를 느리게 움직여야 할 것이다. 만약 기준점이 중간 정도에 있으면 이송 시간은 절반으로 줄 것이다. 그런 장비는 대개 Z 축에 별도의 홈 스위치가 있을 것이고, 기준점 설정 이 후, mach3의 축 DRO에 어떤 값을 넣어서 머신 0 점을 만들어 칼럼의 최고 높이를 설정할 것이다.

##### 분리된 고 정밀도의 홈 스위치

X, Y 축에 정밀도를 자랑하는 장비를 사용 한다면 그에 상응 하는 정밀도의 홈 스위치를 사용 해야 할 것이다.

##### 여러 축의 리밋 스위치들을 같이 연결 할 때

Mach3는 어떤 축에 어떤 스위치가 작동되어 있는지 구분 하지 못하기 때문에 모든 리밋 스위치들은 OR로직으로 연결되고 하나의 리밋 입력핀으로 전달 된다. 각 축은 이때 입력 핀에 연결된 각각의 기준 스위치를 가질 수 있다. 3축을 가진 장비는 그래서 4개의 입력만 있어도 되는 것이다.(홈 3개 리밋 1개)

##### 여러 축에 홈 스위치들을 같이 연결 할 때

Mach3 의 입력단자가 정말 모자란다면 모든 홈 스위치를 OR 로직으로 만들고 모든 홈 입력신호를 갈게 한다. 이런 경우 단지 한번에 한 축씩 기준점을 정 할 수 있다. - 이때 REF 버튼은 스크린 상에서 제거 한다. 그리고 홈 스위치들은 그것들의 축상에서 이송의 끝 위치에 있어야 한다.

##### 슬레이브(Salving)

겐트리 타입의 밀링 장비에서 한 개의 축에 모터를 두 개 사용 한다면, 예를 들어 Y 축으로 움직일 때 양쪽에서 모터를 움직여 이송 한다면 모터를 정의 할 때 하나는 Y 또 다른 하나는 A로 정의 하여 A를 Y에 슬레이브로 설정해 놓고 작동 시키면 된다. 5장 Mach3 설정 참조. 홈 스위치와 리밋 스위치를 양쪽 모두에 설치 한다. 일반적으로 Mach3에 의하여 Y 와 A 는 모두 동일한 스텝과 방향으로 이송 하게 된다. 기준점 설정을 할 때 같은 방향으로 움직여 두 스위치에 동시에 닿은 후 뒤로 살짝 물러나는데 이때 전원이 off 된 후의 축들의 직각을 교정 하는 데 도움이 될 수 있다.

## 4.7 스피들 컨트롤

Mach3 가 스피들을 제어 하는 방법은 아래와 같은 3 가지가 방법이 있다. 또 다른 방법으로는 이 제어 법들을 무시하고 그냥 매뉴얼로 조작 할 수 도 있다.

1. 모터의 On(시계 방향 혹은 반 시계 방향) 과 Off 를 릴레이/접지 제어 한다.
2. 모터가 스텝과 방향 펄스로 제어 된다. (스피들 모터가 서보일 때)
3. 모터가 PWM(pulse width modulation) 신호에 의해 제어 된다.

### 1 .On/Off 모터 제어

M3 와 스크린 버튼은 스피들의 회전 방향이 시계 방향이 되도록 한다. M4는 반 시계 방향이 되게 한다. M5는 스피들이 멈추도록 한다. M3, M4는 프린터 포트이 출력 핀과 연계 하여 외부 출력 신호를 작동 시키도록 설정 할 수 있다. 그리고 난 후 릴레이 등을 통하여 모터를 장비에 연결 하여 사용 할 수 있다.

이것이 비록 쉽게 보이겠지만 실제로는 상당한 주의를 요구한다. 실제 스피들을 반대방향으로 사용 할 일이 없다면 M3와 M4를 같은 방향으로 만들거나 M4를 연결 하지 않도록 한다.

분명한 것은 시계방향과 반 시계 방향을 같이 동작 하게 만들어 에러가 발생하는 상황이 생길 수 있는데 이때 모터의 접촉기로 인해 접지 사고로 파손 될 수 있으니 조심 하기 바란다. 또 다른 어려움 중의 하나는, G코드는 M3가 스피들이 시계 방향으로 회전 하는 중에 M4명령을 내릴 수 있게 되어 있다는 것이다.(혹은 반대의 경우도 가능). 만약 스피들이 AC 모터고 위와 같이 에러가 발생하는 상황이라면 AC 퓨즈가 터지거나 혹은 서큐잇 브레이크(circuit breaker)를 작동 시킬 것이다. 안전을 위해서 컨택터에 시간차를 두고 작동 할 수 있게 하거나 혹은 인버터(inverter)를 장착하여 모터의 회전 방향을 바꿀 수 있도록 한다.

## 2. 모터의 스텝과 방향 제어.

만약 스피들 모터가 서보 모터이며 스텝과 방향을 가진다면 두개의 출력 신호로 속도와 방향을 설정 할 수 있다. Mach3는 스피들과 모터 사이의 기어 박스나 풀리에 대한 계산을 하게 될 것이다. 자세한 사항은 5장 모터 회전에서 기술 한다.

## 3. PWM 모터 제어

스텝과 방향 제어를 대체하는 것으로, Mach3는 요구되는 최고 속도의 퍼센트로 듀티 사이클(duty cycle)을 정하는 PWM 신호를 출력 할 수 있고 (예를 들어 PWM 신호가 0% 일때 0V, 50% 일 때 5V,

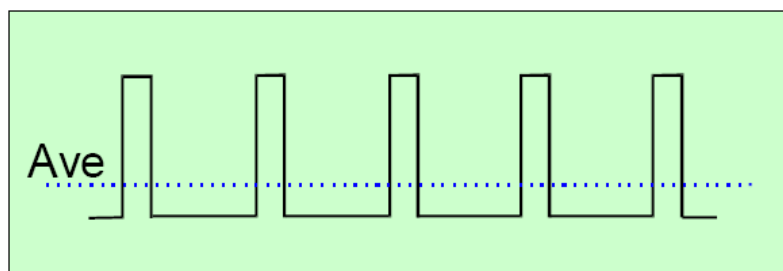


그림 4.12 대략 20% 일 때 PWM(pulse width modulation) 신호

100%일 때 10V) 이것을 다양한 주파수를 사용 하는 인버터와 함께 인덕션 모터(induction motor)에서 사용 할 수 있다. 또, PWM신호는 간단한 DC 속도 제어에서 정류소자(triac)를 작동 시키도록 유도 할 수 있다.

그림 4.12와 4.13은 진폭(pulse width)이 대략 주기의 20% 일 때와 50% 일 때다. PWM 스피들 속도 신호를 DC 전류로 바꾸기 위해서는 펄스 신호가 변환 되어야 한다. 그러므로 PWM 신호의 평균을 찾는 회로가 사용 될 것이다. 회로는 단순한 콘덴서(capacitor)와 저항기로 구성 할 수도 있고 혹은 1.열만만큼 진폭과 최종 전압과의 관계를 선형으로 만들고 싶은가 2. 진폭(pulse width)을 변형시키는 반응 속도에 따라 회로를 복잡하게 구성 할 수도 있다.

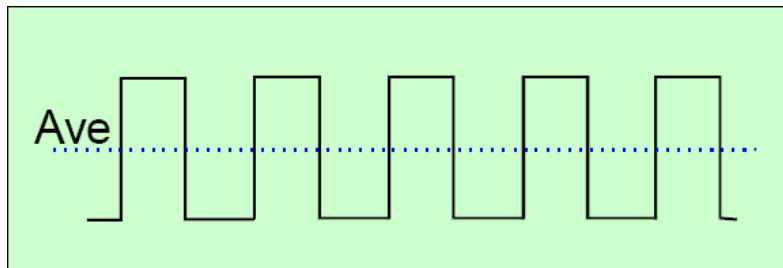


그림 4.13 대략 50% 일 때 진폭 변조 신호(pulse width modulation)

값싼 PWM 속도 제어회로는

보통 메인으로부터 보호되어 있지 않기 때문에 주의를 요한다.

자세한 사항은 Mach2DN 사이트에서 찾을 수 있으며 또는 PWM converter 나 PWM Digispeed 라는 단어를 검색어로 하고 google 과 같은 검색 엔진을 사용하여 원하는 자료를 찾아 보기 바란다.

PWM 신호는 스피들 스텝 궤 상의 출력이다. 모터의 시계 방향, 반 시계 방향을 사용 하는 모터를 저속으로 사용 할 때와 전원을 끌 때 세심한 주의를 기울여야 할 것이다.

**주의:** 많은 사용자들이 PWM 혹은 변속 스피들 드라이버를 사용 할 때 장비 축 움직임에 방해가 되거나 리미트 스위치 센싱을 방해하는 전기적 노이즈가 있음을 발견 하게 된다. 이런 경우 우리는 옵티컬 보호장치가 있는 브레이크아웃 보드를 사용 하길 강력히 권고 하는 바이며 케이블을 커버에 잘 입히고 컨트롤 케이블로부터 파워 케이블을 좀 떨어진 곳에서 운영 하라고 권한다.

#### 4.8 냉매제(coolant)

대상물의 절삭 시 생기는 과열을 막기 위한 쿨러는 밸브나 펌프는 출력 신호를 사용하여 제어 할 수 있는데 스크린의 버튼이나 혹은 M7,M8, M9으로 제어 가능하다.

#### 4.9 나이프(knife) 방향 컨트롤

로터리 축 A는 회전 하여 나이프와 같은 틀이 X와 Y 의 G1움직임 방향으로 근 접선을 형성할 수 있도록 한다. 이렇게 함으로써 비닐이나 천도 재단 할 수 있게 된다.

**주의:** 현재의 버전에서는 이 기능이 호(arc ; G2/G3 동작)를 만들지 않는다. 연이은 G1동작으로 커브를 프로그램하는 것은 여러분의 몫이다.

#### 4.10 디지털 프로브(digitize probe)

Mach3에 디지털 프로브를 연결하여 모델 디지털 타이징 시스템을 만들 수 있다. 프로브(탐침)가 대상물에 접촉 하면 입력 신호를 보내 오고 레이저와 같은 비 접촉 프로브에 리딩(reading)하는 것을 요청하는 출력 신호를 준비 한다.

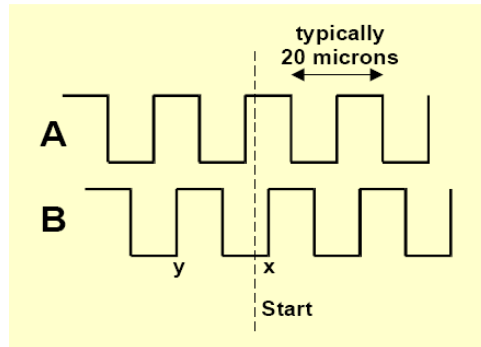
프로브의 끝은 정확한 구형 이어야 하며.(혹은 구의 일부 모양이라도 갖추어야 한다) 스피들 위치의 정 중앙에 장착 되고 Z 방향으로 일정한 거리에 고정 되어 있어야 한다. 비 금속 재료를 스캔 하기 위해서는 팁이 모든 방향으로 약간씩 휘 수 있는 상태가 되어야 하며 동시에 브레이크 스위치 같은 것이 장착되어야 한다. 프로브가 자동 톨 체인저를 이용한 톨에 사용 된다면 프로브 장치는 무선이 되어야 할 것이다.

이러한 요구 사항들은 개인이 직접 프로브를 제작 하고자 하는 이들에게 상당히 어려운 과제가 될 것이며 상업용 프로브는 상당히 고가이다. 개발자의 기능이 향상이 되어 레이저 프로브를 사용 할 수 있도록 되어있다.

#### 4.11 선형(glass scale) 인코더

Mach3는, 수직위상 신호 출력이 가능한 인코더가 연결되어 있다면, 4쌍의 신호를 입력 받는다(그림 4.15 와 같은 인코더). 그리고 이것은 인코더 전용 DRO 창에 위치를 표시해 준다. 이 값들은 주축 DRO 들로부터 값을 불러 오거나 저장 할 수 있다.

그림 4.14 수직 위상 신호





인코더의 내부 구조는 줄이 그어져 있는 유리격자부분과(10 마이크로 간격) 그 사이사이 같은 사이즈의 분리된 투명한 부분이 있다. 격자를 통해 포토 트랜지스터에 빛이 쏘여지면 그림 4.15와 같이 신호가 생성된다. 한 주기는 20 마이크로 길이 가 된다.

첫 번째 신호로부터 5 마이크로 떨어진 포토 트랜지스터는 A로부터  $\frac{1}{4}$  주기인 B 신호를 생성한다.

전체적인 이해에 관한 설명은 상당히 길다. 그러나 신호의 변화는 5마이크론 단위로 변하니 최소 분해도는 5 마이크로 이란 것을 알 것이다. 그리고 변화의 순서로 어느 방향으로 움직이고 있다는 것도 알 것이다. 예를 들어 A가 hi 일때 B가 hi 에서 lo 로 간다면 시작으로부터 왼쪽으로 간다는 것을 알 수 있을 것이다.



그림 4.15 Glass scale encoder

Mach3 는 논리 신호를 원한다. 어떤 인코더는(일부 Heidenhain Model) 아날로그 싸인파형(sinewave)을 만든다. 이때 쓸만한 전자 회로는 5마이크로 보다 더 높은 분해도로 더 많은 격자를 형성 시킨다. 이렇게 하고 싶다면 증폭기나 비교측정기 같은 장비를 이용하여 파형을 네모로 쪼개야 한다. TTL 출력 인코더는 직접 병렬 포트의 입력 핀에 연결이 되지만 노이즈로 인해 오류를 범하게 할 수 있으므로 Schmitt trigger chip 같은 장치를 경유하여 연결시키는 것이 바람직하다. 글래스 스케일은 내부에 드라이브 칩과 광원을 위해서 직류전압(보통 5볼트)을 공급받는다.

주의 :

- a. 아주 작은 백래쉬 나 기계적 드라이브의 용수철은 서보를 불 안전 하게 만듦에 따라 리니어 스케일을 서보 드라이브의 피드백 인코더로 사용 하기 쉽지 않다.
- b. 서보 모터의 회전 인코더를 인코더 DRO에 연결 하는 것은 쉽지 않다. 이것은 사실 위치 판독기로 수동 조작 하는 것에는 생각해 볼만 하다. 문제는 서보 모터에 쓰이는 드라이브 안의 0 볼트는 PC나 브레이크 아웃 보드의 0 볼트와 같지 않다는 것이다. 둘을 같이 연결 하면 문제를 발생 시킬 수 있다.
- c. 선형 축에 선형 인코더를 사용 하는 최대 잇점은 계측의 정밀도나 스크류, 벨트 등등에 연계하지 않는다는 것이다.

## 4.12 스피들 인덱스 펄스

Mach3 는 스피들의 회전에 대한 펄스 입력 신호를 받는다.이로써, 스피들의 실제 속도를 표시 할 수 있고, 톨의 움직임 작료를 나타내며, 스레드(thread) 절삭을 할 때 사용 할 수 있다. 이것은 분당 출력 보다는 회전당 출력을 제어 하는데 사용 할 수 있다.

### 4.13 펌프 충전 - 펄스 모니터

Mach3는 주파수대가 12.5kHz인 펄스를 정상적으로 작동할 때 한쪽이나 양쪽 병렬 포트에 계속 출력한다. 이 신호는 Mach3가 로딩 되지 않았거나 EStop 모드이거나 혹은 펄스 제너레이터가 어떤 고장 일 때는 발생하지 않는다. 그리고 이 신호는 축과 스펀들 드라이브 등을 작동 시키는 다이오드 펌프를 통해 콘덴서를 충전 할 때 사용 한다. 이 기능은 상업용 브레이크 아웃 보드에 종종 적용이 된다.

### 4.14 기타 다른 기능

Mach3는 15가지의 OEM Trigger 입력 신호를 받을 수 있는데 이것은 여러분 각자의 사용을 위해 지정 할 수 있다. 예를 들어 사용자가 만든 매크로를 이용 하여 테스트 할 수 있거나, 버튼 클릭을 시뮬레이션 할 때 사용 할 수 있다.

추가로, 매크로 사용을 인가하는 4개의 사용자 입력을 이용 할 수 있다.

Input #1 은 파트 프로그램의 운영을 정지 시키는데 사용 될 수 있다. 장비의 가드에 연결이 되어야 할 것이다.

입력 에뮬레이션 구조의 자세한 사항은 Mach3 Customisation wiki 에 나와 있다. 셋업 창은 섹션 5에 정의 되어 있다. 릴레이 작동 출력은 스펀들에 사용 하지 않으며 쿨런트(coolant) 는 여러분에 의해 작동 되고 작성된 매크로에 의해 제어 된다.

**마지막 당부** 여기에 나온 모든 입 출력에 관한 것은 설령 두 개의 병렬 포트를 사용 할 지라도 한 꺼번에 모든 것을 제어 하기는 곤란할 것이다. 만약 원하는 모든 입출력을 극대화 시키고 싶다면 ModBus 장치 사용을 해야 할 것이다.

## 5.Mach 3를 장비와 드라이브에 맞게 설정 하기.

만약 장비와 그것을 운영 하는 Mach3 컴퓨터를 구매 했다면 이번 장은 읽지 않아도 된다.

장비 공급자는 Mach3를 설치 및 설정을 끝냈을 것이고 어떻게 사용 하라는 것만 알려 줄 것이니 말이다.

Mach3 의 설치 및 설정 방법을 알리는 지침서를 페이퍼 카피 해서 가지고 있기를 권장 한다.

만약 어떤 경우에 의하여 Mach3를 처음부터 새로 재 설치 하는 것에 대비하여 기존의 설정 데이터를 잃어 버리지 않도록 반드시 백업해 두기를 권장한다. 장비의 각 프로파일은 .xml 파일에 있다. (예, Mach3mill.xml)

### 5.1 설정에 관한 전략

이번 장에서는 설정에 관한 세부 사항을 기술 한다. 동시에 하나씩 진행하는 테스트를 따라 가다 보면 이것이 얼마나 간단 명료 하지 않게 될 것이다. 이번 장은 한 번 읽어 보고 다시 컴퓨터와 장비를 이용해서 해 보는 것이 바람직 하다. 기술한 내용은 Mach3가 전혀 설정 되지 않은 상태 혹은 장비 없이 진행한 다는 가정 하에 3장에서 했던 것과 같이 진행한다. 이번 장의 내용은 상위 메뉴바의 Config 메뉴 대화 창 안에 모두 있다.

### 5.2 초기 설정

첫 대화 창으로 Config > Ports and Pin 을 실행 보자. 이 창은 여러 탭이 있지만 모양은 아래 그림 5.1 과 같다.

#### 5.2.1 포트의 주소 설정

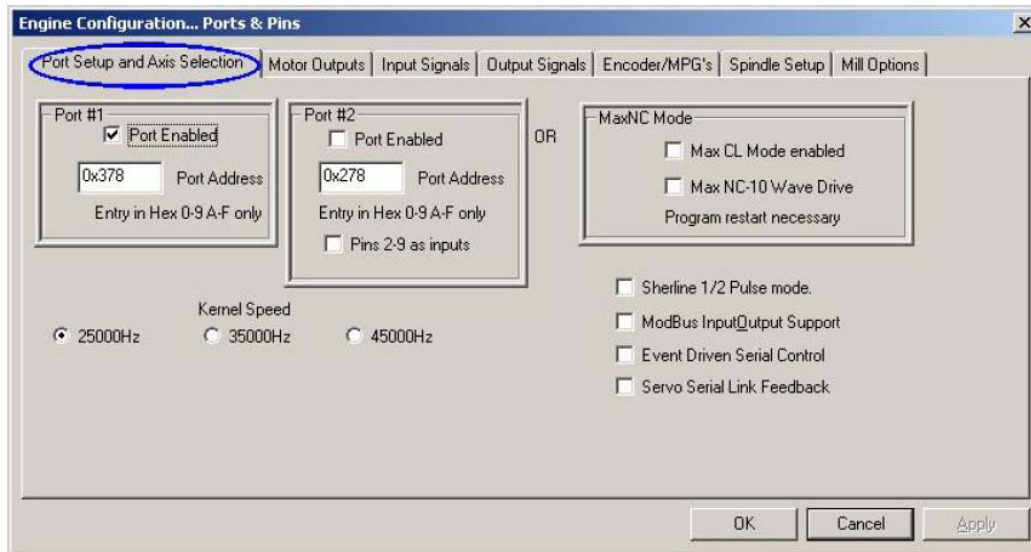


그림 5.1 Ports Setup and Axis Selection 탭

만약 컴퓨터가 하나의 병렬 포트만 있다면(대부분 그렇지만) 0X378 (16진수378) 포트 1의 기본 어드레스는 거의 정확할 것이다. 그렇지만 모터에 전원이 공급 되는 데도 만약 제어가 안 된다면 아래의 내용을 참고 하고

**메모 [DP9]:** 시작 - 제어판 - 시스템 - 하드웨어 탭 - 장치 관리자 - 포트 - 해당 포트 더블 클릭 - 리소스 탭의 설정 부분에서 확인 가능

확인해 보기 바란다.

하나이상의 PCI 카드를 장착해서 사용 한다면 어느 주소를 사용 하는지 알아야 한다. 다음의 단계에서 찾을 수 있다. Windows의 제어판을 실행해 보라. 그리고 시스템을 열어 하드웨어 탭을 클릭하고 장치 관리자 버튼을 누른다. 포트(COM&LPT) 아이템을 확장한다. LPT나 ECP 포트를 더블 클릭 한다.

새로운 팝업 창이 나타나면 리소스 탭을 연다. 첫 줄 IO 범위의 네 개의 숫자가 사용할 주소다. 값을 따로 적어 두고 창을 닫아라.

**주의 :** 다른 PCI 카드의 설치 및 제거가 프린터 포트를 건드리지 않아도 주소를 바꿀 수 있으니 주의 하기 바란다.. 만약 두 번째 프린터 포트를 장착하여 사용 하고 싶다면 위의 내용을 다시 반복 한다. 장치 관리자 및 제어판 모두를 닫는다.

그림 5.1 그림 과 같이 창에 포트 주소(Port Address)를 입력 한다 그리고 0x 는 쓰지 않는다. 필요 하다면 포트2 도 값을 입력 하고 Enable 박스를 체크 하면 된다.

**Apply 버튼을 눌러 값을 저장 한다. 이것은 중요하다. Apply 버튼을 누르지 않은 상태에서, 어떤 값을 어떤 탭에 넣더라도 이것은 자동으로 저장 되지 않는다.**

### 5.2.2 엔진 주파수 정의

Mach3 드라이버는 25,000Hz, 35,000Hz, 45,000Hz. 에서 작동 할 수 있는데 이것은 Mach3를 운영할 때 프로세서의 속도에 따라 그리고 다른 프로그램으로 인한 부하의 정도에 따라 설정 할 수 있다. 필요한 주파수는 어떤 축이던 최고 속도로 드라이브 할 때의 최대 주파수 비율에 달려 있다. 25,000Hz 가 스테퍼 모터 시스템에서는 적절하다. 10 마이크로 스텝을 지원 하는 Gecko 201 같은 경우 1.8도의 표준 스텝 모터를 750RPM 으로 회전 시킬 수 있을 것이다. 고 분해도 샤프트 인코더를 가진 서보 모터의 경우 고 주파 율을 필요로 한다. 자세한 사항은 모터 튜닝에 기록되어 있다.

1GHz의 클럭 스피드를 가진 컴퓨터는 분명 35,000Hz에서 훌륭히 돌아간다. 그러나 이 속도 보다 낮은 컴퓨터를 갖고 있지 않는 이상 이것을 선택 해서 높은 스텝 비율로 사용 할 수 있다. Mach3의 데모 버전에서는 25000Hz 에서만 작동 될 것이다. 그리고 Mach3 프로그램을 강제로 종료 했다면 다시 실행 할 때 25,000Hz로 복귀 된다. 실제 사용중인 시스템의 주파수는 Diagnostics 스크린에서 확인 할 수 있다.

**절대로 Apply 버튼을 누르는 것을 잊어서는 않된다.**

### 5.2.3 기타 특수 기능 정의

특별한 설정을 위해 체크 박스에 표시 할 일이 있을 것이다. 대부분 그와 연관된 하드웨어를 장착 할 것이며 사용 설명이 같이 있을 것이다.

## 5.3 사용하게 될 입 출력 신호의 설정.

이제 기초적인 설정은 성립 하였으므로 어떤 입력과 출력 신호를 사용 할 지 정의를 해보고자 한다. 어떤 프린터 포트를 사용 할지, 어떤 핀 번호를 지정 할 지 등등을 정의 내려보자. Mach3를 사용 할 수 있는 브레이크 아웃 보드 라면 어떻게 설정할 것인가에 대한 가이드 라인을 제시해 주거나 이미 연결에 대한 정의가 내려진 설정 프로파일(.XML) 파일을 제공할 것이다.

### 5.3.1 축 그리고 스텝들 출력 신호 설정

그림 5.2 와 같이 Output signal 탭을 본다. X,Y 그리고 Z 축이 어디 포트에 방향과 스텝 핀이 각각 연결 되어 있는지 입력해 준다. 그리고 각 축의 Enable 란을 클릭 한다. 만약 로터리 회전 축이나 슬레이브 축이 있어도

**메모 [DP10]:** 대체적으로 Mach3 폴더 안에 있으며 mach3mill.xml, mach3turn.xml, mach3router.xml 등등 각각의 프로 파일로 구성하며 이곳에 설정 값들이 저장되어 있다.

**메모 [DP11]:** 만약 여러 개의 전선이 하나로 뭉뚱되어 있어 구분 하기 힘든경우 스텝핀 넘버와 방향 핀 넘버를 Mach3가 자동으로 찾을 수 있도록 도와주는 기능이 있으니 참고 하기 바란다.

설정해 주어야 한다. 스피들이 매뉴얼로 작동 되는 것이면 설정 하지 않아도 된다.

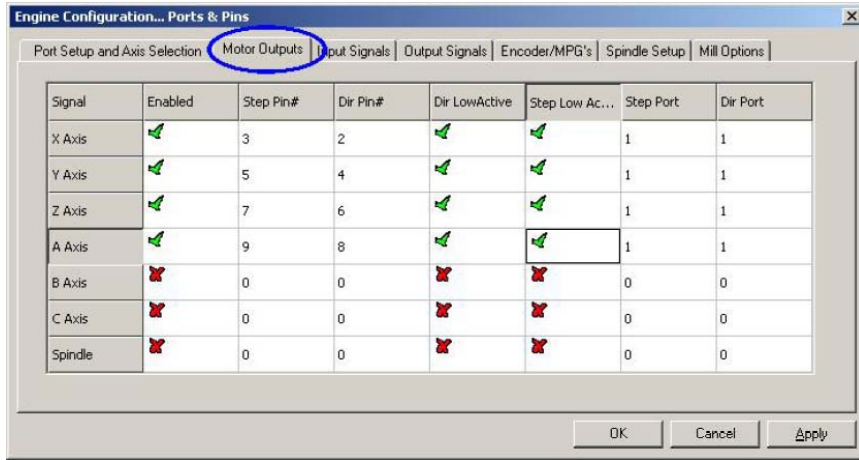


그림 5.4 축과 스피들의 모터 출력 핀 정의

스핀들 속도가 Mach3에 의해 제어 된다면 spindle을 enable 시켜 주고 스텝 핀과 포트를 설정해 준다.

반드시 Apply 버튼을 누르는 것을 잊지 않는다.

### 5.3.2 입력신호 설정.

자 Input Signal 탭을 선택해 보자. 그림 5.5와 같을 것이다.

우리는 4.6 장의 계획 중 하나를 선택했다는 가정하에 다음을 기술한다. 1번의 전략과 같이 리미트 스위치가 같이 연결이 되어 있고 Estop 을 작동시키거나 전자 회로에 의해 축을 멈추게 한다면 리미트 입력란에 어느 것도 체크 하지 않는다.

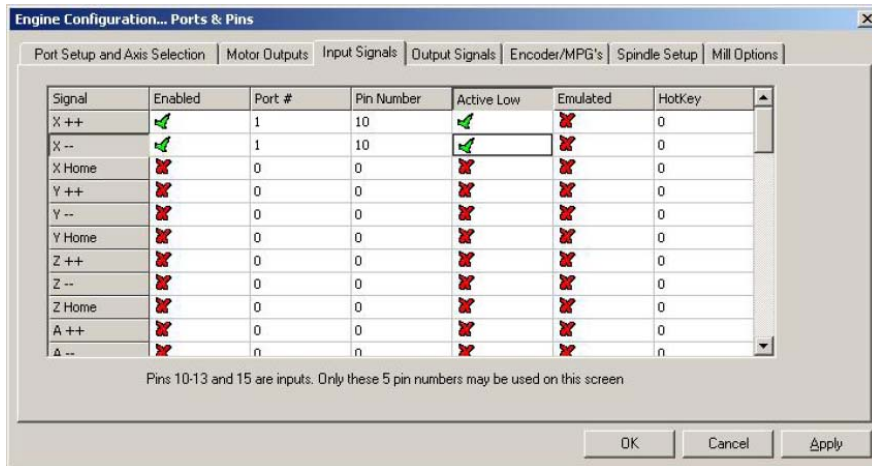


그림 5.5 입력 신호(input signals) 탭

2번 전락이라면 아마 X, Y 그리고 Z 축에 홈 스위치가 있을 것이다. “ Home” 박스에 체크를 하고 포트 핀을 정의해 준다. 리밋과 홈 스위치를 겹쳐서 쓴다면 각 축에 Limit ++, Limit -- 박스를 체크 하면 된다.

**Input 1**은 특별한 항목으로 안전 가드가 없을 때 파트프로그램의 진행을 차단 할 수 있다. 나머지 3개는, 그리고 Input 1이 가드 인터록에 사용 하지 않을 경우, 사용자의 선택에 따라 사용 할 수 있으며 매크로 코드로 테스트 될 수 있다. 나중에 각자 한번 시험해 보도록 한다 .

**Input 4:** 외부로 부터 누름버튼 스위치를 연결 하여 Single Step 기능을 만들 수 있다.

**Index Pulse :** 장비 스피들의 회전으로부터 오는 펄스신호 센서가 있다면 체크 한다.

**Limit Override :** Mach3가 리밋 스위치를 제어 하고 외부로 부터 연결된 버튼이 있어 리밋으로 부터 조그한 후 스위치를 off 할 수 있는 환경에 있다면 이것을 체크 한다. 외부 버튼이 없다면 그냥 스크린 버튼을 이용 해서 동작 시킬 수있다.

**Emergency Stop** 상자는 반드시 체크되어야 한다.

**OEM Trigger :** 화면에 버튼이 보이지 않는 상태에서 OEM 버튼기능을 가능하게 하려 할 때 체크한다.

**Timing :** 하나 이상의 슬롯 이나 마크가 있는 스피들 센서가 있을 때 체크한다.

**Digitize :** 만약 프로브(탐침)을 연결 하고 싶다면 를 체크한다.

**Torch Height Control :** 플라즈마 봉으로 높이 조절이 필요 할 때 체크 한다.

Encoder #1 - #3: 글래스 스케일 인코더가 장착 되어 있어 축의 위치를 측정 할 때 각 축에 연관된 인코더를 체크한다. 단, 서보 시스템 방식의 서보 모터 인코더 일 때는 이것을 체크 하지 않는다.

만약 하나의 프린터 포트가 있다면 5개까지 입력이 가능하다. 포트가 2개 있다면 10개가 될 것이다. 위에서 설명한 바와 같이, 리밋과 홈 스위치는 서로 공유 될 수 있다. 알아 둘 것은, 인코더는 축당 두개의 입력이 요구 된다. 입력신호부의 부족은 흔한 일이기 때문에 물리적 Limit Override 스위치 같은 장치는 사용 하지 않는 것이 도움이 될 것이다. 또한 몇몇 입력 신호를 위해서 키보드 에뮬레이터를 고려해 보는 것도 괜찮을 것이다.

**Apply 버튼을 눌러 설정한 데이터를 저장 하는 것을 잊지 말라.**

### 5.3.3 에뮬레이션 입력 신호

에뮬레이션을 입력 신호로 사용하려 한다면 Port/Pin 번호나 Active Hi, Lo 신호는 생략해도 좋지만 Hotkey 칼럼은 잊어서는 않된다. Hotkey 값에 대응하는 코드가 Key-down 메시지에 전달되면 신호 처리는 “ active “ (작동) 상태가 될것이며 반대로 Key-up 메시지가 전달되면 “ inactive” (처리불가) 상태가 될 것이다.

일반적으로 Key-up, Key-down 신호는

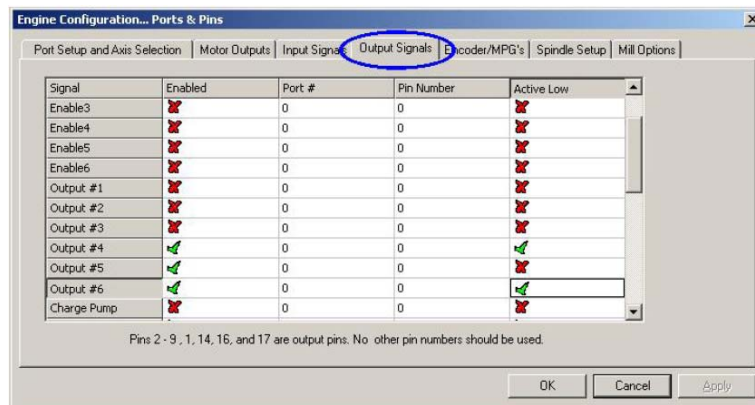


그림 5.6 - Output Signals

입력부에 연결된 스위치에 의해 작동 하는 키보드 에뮬레이터(Ultimarc IPAC 혹은 Hagstrom)에서 받는다.이로서, 프린터 포트에 있는 여유분의 핀 수 보다 더 많은 스위치의 센싱을 가능 하게 하지만 스위치가 작동 할 때 상당한 시간이 지체 되고, 때로는 Windows 에 의해 Key-up, Key-down 메시지가 나타 나지 않을 수도 있다. 에뮬레이션 신호는 Index 나 Timing 을 위해 사용 될 수 없으며 Estop 으로 사용 되어서는 않된다.

### 5.3.4 Output Signals(출력 신호)

그림 5.6 과 같이 필요한 출력 신호를 설정한다.

모든 축 드라이브가 Enable output 에 연결될 수 있음에 따라 아마도 한개의 Enable 출력을 사용하게 될 것이다. 사실, Charge pump/pulse 모니터 기능을 사용하고 있다면 해당 output 으로 부터 축 드라이브를 작동 시킬 수 있다.

Output# 신호는 스피들, 쿨런트 펌프나 혹은 밸브의 시작 정지 제어를 위해 사용 하거나 개인 설정의 Mach3 버튼 이나 혹은 매크로의 제어를 위해 사용한다.

만약 브레이크 아웃 보드가 Mach3의 정상적 운영을 지속적으로 확인 하기 위한 펄스 신호 입력을 받는다면 Charge pulm 라인은 활성화 되어 정의 내려져야 한다. Charge Pump 2는 두번째 포트에 연결된 또 다른 브레이크아웃 보드가 있을 경우나 두번째 포트 자체를 점검 할 때 사용 한다.

### 5.3.5 인코더 입력 설정

Encoder/MPGs 탭은 선형 인코더의 분해능과, 연결에 대한 설정 이나 혹은 축을 조그하여 움직이는 MPG(Manual Pulse Generator)에 대한 설정을 위해 사용 한다.

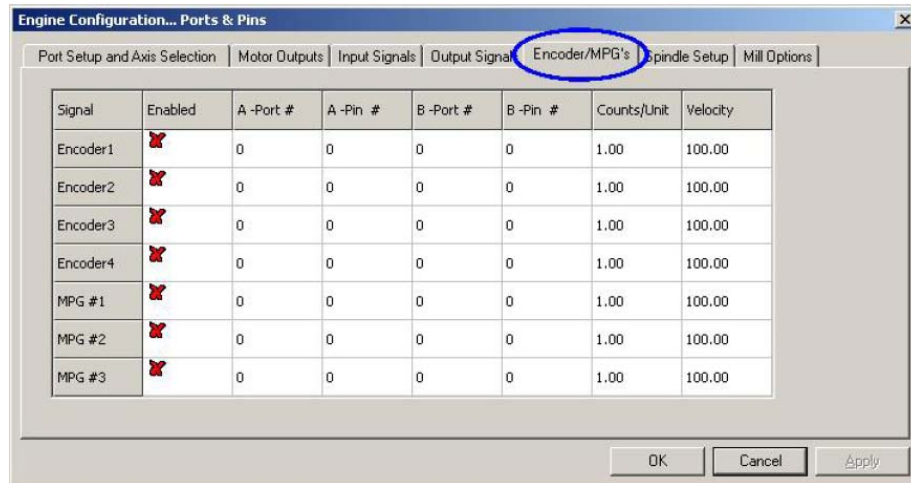


그림 5.7 Encoder Inputs.

Config>Ports&Pins 의 구체적 설명을 위해 여기서 다루어 질 것이다. 이 창에서는 active Lo, Hi 가 필요하지 않는데 그것은 인코더의 카운트가 반대 방향으로 간다면 단지 A와 B에 할당된 핀을 바꾸면 된다.

#### 5.3.5.1 인코더

Counter per unit 값은 인코더의 분해능에 대응하여 설정되어야 한다. 그러므로 20마이크론 스케일을 가지는 선형 인코더는 매 5마이크로 마다 카운트 할 수 있는데(수직 위상 신호를 이용하여), 그것은 단위(mm)당 200 번 카운트 할 수 있는 것이다. Inch(인치)단위로 환산하면 인치당 200X 25.4=5080 카운트가 될 것이다. Velocity 는

사용 되지 않는다.

### 5.3.5.2 MPGs

Couter per unit 값은 수직위상 신호의 계산된 숫자로 정의 내려 지는데, Mach3에서 MPG의 움직임이 보여지기 위해 생성되는 값이다. 100CPR(cycle per rotation) 인코더는 값이 2 정도가 적절할 것이다. 높은 분해능을 원한다면, 기계적 감도를 갖추기 위해 값을 더 올려야 할 것이다. 1024 CPR 인코더에서 값이 100 정도에서도 잘 운영 되는 것을 볼 수 있다.

### 5.3.6 스피들의 설정

다음으로 아래 그림과 같이 Spindle Setup 탭을 보자. 여기에서는 어떻게 스피들과 쿨런트(냉매제)가 제어 되는지 정의 한다. 스피들을 on/off 하거나 PWM 신호를 이용 하여 속도제어를 하거나 혹은 스텝과 방향 신호를 제어 하는 것을 Mach3가 할 수 있게 하거나, 아니면 직접 제어 할 것인지 선택할 수 있다. 대화 창은 아래 그림과 같다.

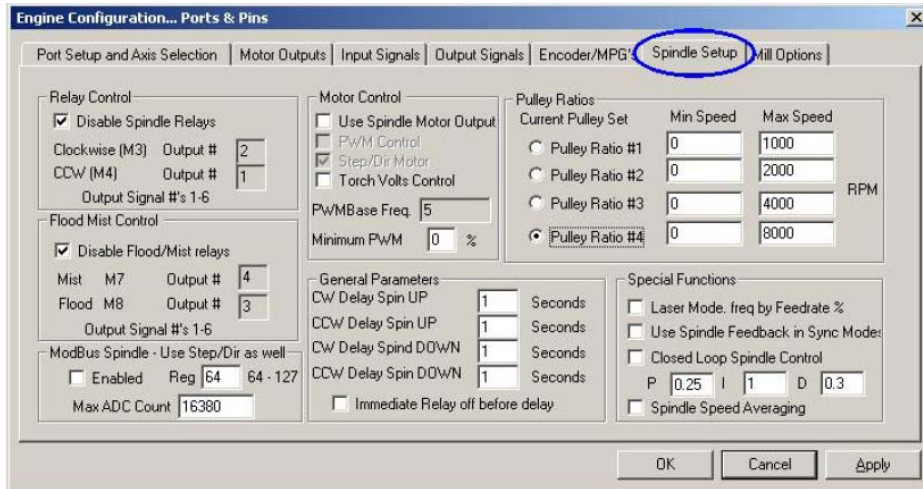


그림 5.8 스피들 셋업

#### 5.3.6.1 Flood Mist Control – 냉매제 제어

코드 M7은 액체 냉매제를(flood coolant) on 시킬수 있고 M9는 분무기형 냉매제(mist coolant)를 on 할 수 도, 또 모든 쿨런트를 off 할 수도 있다. 위의 그림에서 Flood Mist control 섹션을 보면 어떤 출력 핀을 사용하여 기능을 활성화 시킬지 예를 볼 수 있다. 출력을 위한 Port/Pins 는 이미 Output signals 탭에 정의내려져 있다. 이 기능을 사용하고 싶지 않다면 단지 Disable Flood/Mist relays 란을 체크 하면 된다.

#### 5.3.6.2 Relay Control – 스피들 릴레이 제어

만약 스피들 속도를 손으로(매뉴얼로) 제어 하거나 PWM 신호에 의해 제어 된다면 Mach3 는 방향과, 언제 시작, 정지 할지를 두개의 출력을 사용해서 정의 할 수 있다. 이 두개의 출력을 위한 포트/핀 은 이미 Output Signals 탭에 설정되어 있다.

여러분이 스텝과 방향으로 스피들을 제어 한다면 위의 제어들은 필요가 없다. M3, M4, M5 는 자동으로 생성되는 일련의 펄스를 제어 하게 된다. 만약 사용을 원하지 않으면 “ Disable Spindle Relays” 를 체크 한다.

#### 5.3.6.3 Motor Control – 모터 제어

만약 스피들을 PWM 이나 스텝과 방향에 의한 제어가 되게 하려면 Use Motor Control 을 체크한다. 그리고 PWM



control 이나 Step/Dir Motor 중 하나를 선택 한다.

### **PWM Control**

PWM 신호는 디지털 신호다, 사각파형의 폭은 시간의 퍼센트가 되고 최고속 회전 할 때의 퍼센트로 나타난다. 예를 들어 모터와 PWM 드라이버가 있고 모터는 최고 속력 3000rpm 이라고 할 때 그림 4.12를 참고로 하면  $3000 \times 0.2(20\%) = 600\text{rpm}$  이 된다. 같은 이유로 그림 4.13을 참고 하면 1500rpm 이 될것이다.

Mach3는 펄스 폭 범위와 사각파형 주파수 높이 범위 사이에서 타협점을 찾아야 한다. 주파수가 5Hz 라고 한다면 25,000 Hz 커널 스피드에서 운영 되는 Mach3는 5,000 가지의 속도를 출력 할 수있다. 10Hz 로 간다면 2500 가지로 줄어들지만 그래도 1 혹은 2 RPM 분해능에 이르게 된다.

사각파형의 저주파수가 시간을 증가시켜 모터 드라이브에 속도 변화가 요구 된다는 것을 알리게 된다. 5와 10Hz 사이에서 타협점을 찾게 되고 이때 선택된 주파수는 PWMBase Freq 에 입력이 된다.

드라이브와 모터들에는 최저 속도가 있다. 일반적으로 쿨링 팬이 저속에서는 효율적이지 못하기 때문에 높은 토크와 높은 전류가 그와 반대로 요구 되는 것이다. Minimum PWM% 상자는 Mach3가 PWM 신호를 출력하는 것을 멈추게 할 정도의 최고 속도에 대한 퍼센트를 설정 할 수 있도록 한다.

알아두어야 할 것은 PWM 전자회로 드라이버는 최저 속도에 대한 설정이 있을 수 있으며 Mach3의 풀리(pulley) 설정은 그에 대한 최저 속도를 셋업 할 수 있게 한다. 일반적으로 풀리의 리미트를 Minimum PWM % 값 이나 하드웨어 리미트 보다 살짝 높게 설정하여 속도를 제어 하거나 혹은 단지 멈춰 버리는 것을 방지하고 에러 메시지를 보낼 수 있다.

### **Step/Dir Motor**

이것은 서보 드라이버나 스텝 펄스에 의해 제어 되는 변속 드라이버가 될 것이다.

Mach3 풀리(pulley) 설정(.5.5.6.1 설정 참조)을 사용하여 모터나 혹은 회로에 필요한 최저 속도를 정의한다.

#### **5.3.6.4 Modbus Spindle – Use Step/Dir as well Modbus : 스펀들 제어**

Modbus 장치(예: Harmann ModIO) 상의 아날로그 포트를 설정하게 해준다. 자세한 사항은 여러 분이 가지고 있는 Modbus 장치 설명서를 참고 한다.

#### **5.3.6.5 General Parameters – 일반적인 파라미터**

Mach3는 명령어를 계속 진행 하기 전에 스펀들의 시작 혹은 정지 후에 지연 시간을 제어 한다. 이로서, 절삭이 이루어지기 전에 스펀들에 가속 시간을 주고, 소프트웨어적 방어를 제공하여 시계방향에서 반 시계 방향으로 급격한 전환을 방지 한다. 머무르는 시간은 초 단위로 입력한다.

Immediate Relay off before delay 를 체크 하면 M5 가 실행 되자마자 스펀들의 릴레이 스위치를 off 할 것이다. 그리고 체크 되지 않는다면 회전이 서서히 지체 될 때 까지 기다리게 된다.

#### **5.3.6.6 Pully ratios – 풀리의 비율**

Mach3 스펀들 모터의 속도를 제어 한다. 스펀들 속도는 S 문자를 통해 프로그램 할 수 있다. Mach3 풀리 시스템은 기어박스 셋팅이나 혹은 4개의 각기 다른 풀리의 연관성에 대한 정의를 내리도록 한다. 스펀들 모터를 튜닝 한 후야 어떻게 작동 하는지 이해하기가 쉬울 것이다. 자세한 것은 아래 5.5.6.1 에서 참고 하기 바란다.

#### **5.3.6.7 Special function – 특수 기능**

Laser mode 는 피드비율에 의한 레이저의 파워를 제어 하지 않는다면 항상 off 해 둔다. Use Spindle feedback in sync mode 는 체크 하지 않는다.

Closed Loop Spindle Control 을 체크 하게 되면 S 문자로 명령되어진 인덱스/타이밍 센서에 의한 실제 스피들 속도와 소프트웨어가 읽는 값을 매칭 시키려 할 것이다. 정확한 스피들 속도는 Mach3Turn 을 사용하는 CNC 선반장비에서 중요한 것이 아니니 체크 하지 않아도 된다. 만약 사용 해야 한다면 P, I 그리고 D 변수가 0과 1 사이에서 설정이 된다. P는 루프(loop)의 게인(gain)을 제어 한다. 그리고 과도한 값은 입력하면 속도는 그 근처의 값에서 불 안정한 속도 변화를 보이게 될 것이다. D 변수는 제동 시키는 것에 적용이 되어 속도의 변화율을 이용하여 진동 하는 것을 안정화 시킨다. I 변수는 실속도와 요청된 속도 사이의 차이를 감지하여 일정한 상태의 정밀도를 유지 시킨다. 이런 값들을 튜닝 시키는 것은 Operator > Calibrate spindle 에서 열리는 창을 참고 하면 된다.

Spindle Speed Average 가 체크 되면 Mach3는 실제의 스피들 속도를 유도 할 때 몇번의 회전에 걸쳐 발생하는 인덱스/타이밍 펄스 사이의 시간을 평균화 시킨다. 초기 제어 속도가 느린 스피들 드라이브를 사용 하거나 단기적 속도 변화를 주는 제어에 유용하게 사용 할 수 있을 것이다.

### 5.3.7 Mill Options 탭

Config>Ports&Pins 의 마지막 탭인 Mill Options 에 대해 알아 본다.

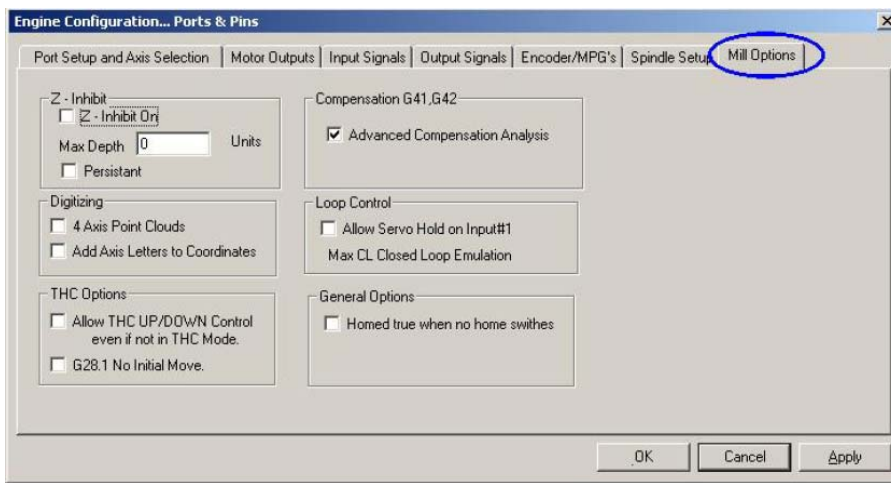


그림 5.9 Mill Options 탭

**Z-inhibit** : Max Depth 값은 Z가 아래로 움직 일 수 있는 최저 값을 나타낸다. Persistent 체크 박스는 Mach3 가 동작 할 때 마다 체크 하게 된다.

**Digitizing** : 4 Axis Point Clouds 체크 박스가 활성화 되면 A, X, Y, Z 축의 상태가 기록 된다. Add Axis Letters to Coordinates 포인트 클라우드 파일에 축의 이름과 함께 데이터 접두어를 표시 한다.

**THC Options** : Allow THC UP/Down Control... 은 THC 모드가 아니어도 작동 하게 한다. G28.1 No Initial Move 는 초기 움직임이 없음을 지시 한다.

**Compensation G41 , G42** : Advenced Compensation Analysis 가 활성화 되면 좀 더 세밀한 분석을 하게 되어 복잡한 형태를 자를 때 (G41, G42를 사용 하여) 커터 직경 보정으로 인해 대상물에 흠집을 내는 위험성을 줄인다.

**Homed true when no Home swithes** : 글 내용과 같이 시스템을 이미 기준점이 잡혀 있었던 것처럼 만든다. 이것은 Ports & Pins Inputs 탭에 홈 스위치가 정의 내려져 있지 않을 때만 사용 한다.

### 5.3.8 테스트 하기

자 이제 장비를 가지고 간단한 테스트를 할 만큼 Mach3 설정은 끝 마쳤다.

이제 Mach3Mill 프로파일을 시작 하고 Diagnostics 스크린 탭으로 가서 화면을 본다. 이곳에서 입 출력에 관한 많은 LED(화면상의)를 보게 될 것이다. 확인해야 할 것은 외부의 EMO 신호가 비 활성화 되어 있는지 확인한다. (빨간색 LED가 깜빡 이지 않는다.)

쿨런트나 혹은 스피들 회전과 관계된 출력을 사용하고 있다면 Diagnostics 스크린 상에서 연관된 버튼을 사용하여 스위치를 on/off 할 수 있다. 이때 장비에서 반응을 보이거나 혹은 멀티미터기로 전압을 모니터 할 수 있어야 한다.

다음으로 홈과 리밋 스위치를 작동해 본다. 신호가 활성화 되면 화면상에서 대응 하는 노란색 LED 가 점등 되는 것을 볼 수 있을 것이다.

이런 일련의 테스트는 여러분의 프린터 포트에 연결된 선이 바르게 연결 되어 있는지 확인 시켜 주고 입출력이 혼선이 되어 있지 않은지 가르킨다.

두개의 포트를 사용 하고 모든 테스트 신호가 그중 하나에만 나타 난다면 임시로 설정한 스위치의 사용을 고려해야 할것이다. 그래서, 홈이나 혹은 리밋 스위치들 중하나가 이 임시 스위치를 경유 하게 하여 작동의 유무를 판단 할 수 있다. 이런 테스트를 한다면 절대 Apply 버튼을 누르는 것을 잊어 서는 않된다. 모든 작동이 확인 됐다면 다시 복구 하도록 한다. 만약 문제가 발견되면 어떤 문제인지 추려내야 하는데 이렇게 함으로서 작동을 시작한 후의 오류를 막을 수 있을 것이다. 멀티미터가 없다면 로직 프로브 나 혹은 실제 LED가 있는 D25 어댑터 를 구해서 핀의 상태를 확인 할 수 있다. 컴퓨터로 들어 오고 나가는 신호가 정확 하지 않은지, 즉 Mach3 를 작동 할 때 오동작 이 있지 않은지, 혹은 D25 에 신호가 가지 않은지, 즉 장비나 브레이크 아웃 보드의 배선이 제대로 되어 있는지 확인 해야 할 것이다. 잘 모르겠다면 전문가에게 문의 하라고 말하고 싶다. 이런 것에 익숙해 지면 장비를 운영할 때 무엇이 문제 인지 스스로 잘 해결 할 수 있을 것이다.

### 5.4 단위 설정.

기초적 기능이 작동된다는 가정하에 축 드라이브에 대한 설정을 해 본다.

처음 결정해야 할 것은 단위를 mm로 할 것인가 inch 로 할 것인가를 정한다. 어떤 것을 설정 하던 파트 프로그램은 그에 따라 운영 할 수 있다. 설정 단위가 시스템 장비의 단위와 같다면 더 사용 하기 쉬울 것이다. 예를 들면 볼 스크류가 만들어질 때 사용된 단위등등. 그러므로, 0.2인치의 스레드(thread)를 가진(5tpi) 스크류는 인치로 설정 하는 것이 유리 하고 반대로 mm 단위로 만들어진 리드(볼)스크류는 mm 단위가 유리 하게 된다. 물론 단위를 환산해서 적용 시킬 수도 있지만 이것 역시 고려를 잘 해야 하는 문제가 될 것이다.

그렇다고 반드시 작업 하는 단위가 위에 따라 그렇게 되어 하는 것은 아니다.



그림 5.10 단위 설정

설정 방법은 Config > Setup Unit 에서 설정한다. 한번 설정한 것은 다음 섹션에 나오는 모든 것을 다시 설정 하기 전에는 절대로 바꾸면 안 되며 함부로 바꿀 시에는 모든 것이 엉망이 될 것이다.

## 5.5 모터의 튜닝

자 | 이제 장비를 움직일 만한 모든 것은 갖추어 졌다. 이번 섹션에서는 축의 움직임을 셋팅 하는 것과 Mach3에 의해 스프린들이 제어 된다면 그것을 셋팅 하는 것을 알아 본다.

각각의 축에 대한 전반적인 전략은 a) 톨이나 혹은 테이블의 단위당 움직임을 위해 얼마만큼의 스텝 펄스를 계산 하는 가 와 b) 모터의 최 고속 설정을 정하는 것 그리고 c) 이때 모터에 필요한 감/가속 을을 설정 하는 것이다.

한 번에 한축씩 해 **보자**.

주의 할 것은 전원을 인가 하기 전에 브레이크 아웃 보드, 컴퓨터로 가는 배선이 잘 되었는지 다시 잘 확인 해 보기 **바란다**.

### 5.5.1 단위 대비 스텝 계산 하기

Mach3는 각 축이 스텝당 움직인 거리(mm 혹은 inch)를 **자동**으로 계산 하게 할 수 있지만 미세조정(fine tuning)을 위해서 남겨 놓는 것이 좋을 듯 하니 이번에는 전반적인 이론에 대해 논의해 본다.

Mach 3가 단위당 움직임(step/inch, step/mm) 에 필요한 스텝 수는 물리적 움직임, 예를 들면 볼 스크류의 피치나 스크류와 기어 사이의 기어링, 스텝 모터의 속성에, 마이크로 스텝 핑이나 서보 모터의 인코더 혹은 전자회로의 전기적 기어링에 따라 달라진다. 이제 하나씩 살펴보고 다시 이들에 관해 조합해 본다.

#### 5.5.1.1 기계적 움직임의 계산

모터 샤프트의 회전을(모터 회전 당 inch 혹은 mm) 통해 한 단위(mm, inch)만큼 움직이는 거리를 계산 하려한다. 아마도 인치단위에서는 회전이 더 많고 mm 단위에서는 더 적을 것이지만 계산하는 것에는 차이가 없다.

스크류와 너트 방식에서는 스크류 피치간의 길이와 스크류에 감긴 나사선이 몇 개인가를 알아야 할 것이다. 인치 스크류는 thread per inch (tpi) 로 나타낼 것이며 이때 피치는 1/tpi 다. (예 : 8tpi 의 피치는 1/8 = 0.125" )

만약 여러줄 나사 스크류라면 실효 피치 값을 구하기 위해 나선의 수 곱하기 피치를 하면 된다. 실효 스크류 피치(effective screw pitch)란 스크류가 한 번 회전 할 때 축이 움직이는 거리를 가리킨다.

이제 단위당 스크류 회전을 계산 할 수 있다. **단위당 스크류 회전을 = 1 / 실효 스크류 피치**

만약 스크류가 모터에 직접 물려 있다면 이때는 단위 당 모터 회전을 이 된다. 그리고 모터에 벨트나 체인으로 연결되어 있다면 Nm 을 모터 기어의 톱니수로 하고 Ns 를 스크류 기어 톱니수로 한다고 할 때

**단위당 모터 회전을 = 단위당 스크류 회전 X Ns / Nm** 이 된다.

예를 들면, 8tpi 스크류가 48개의 톱니를 가진 기어에 물려 있고 16개의 톱니를 가진 기어가 모터에 물려 있다면 이때 모터 샤프트 피치는 8 X 48 / 16 = 24 가 된다.

메트릭 단위의 예로, 두 줄 나사선 스크류가 있고 나사산 거리는 5mm 라고 할 때( 실효 피치는 10mm 가 된다) 이것이 24개의 톱니를 가진 기어에 연결 되어 있고 48개의 톱니를 가진 기어가 스크류에 물려 있다면 (단위당 스크류 회전을은 =0.1 이고 단위당 모터의 회전은은 0.1 X 48 / 24 = 0.2 가 될 것이다.

랙 피니언 이나 체인 벨트 방식도 계산은 비슷 하다.

벨트는 메트릭(metric)과 영국식 단위 피치가 보편적이며, 5, 8 mm 나 0.375" (3/8" ) 가 일반적으로 쓰인다. 랙의 경우 톱니의 피치(tooth pitch)를 알아야 한다. 가장 좋은 방법은 50에서 100개의 톱니 폭을 측정하는 것이다. 알아 두어야 할 것은 표준 기어는 다이어미터 피치(diameter pitch) 로 되어 있기 때문에  $\pi$  (pi = 3.14152...) 가 들어가서 길이가 일반적 유리수로 나타나지는 않을 것이다.

**메모 [DP12]:** 만약 X, Y 축에 사용되는 리드(볼) 스크류의 스텝이 같다면 한 축을 정의 하고 다른 한축은 값을 복사하여 넣으면 될 것이다.

**메모 [DP13]:** 초보자 라면 한번 짚은 경험 할 수 있는 문제다. 본인 역시 드라이버 보드와 파워를 태워 본적이 있어서 남 이야기 처럼 느껴 지지 않는다.

**메모 [DP14]:** 차후에 설명하겠지만 초보자가 설정하기 편한 방법이다.

앞으로 이것을 **톱니 피치(tooth pitch)** 라고 한다.

피니언, 스포라켓, 풀리 등의 톱니 수를Ns로 놓고 이것이 렉, 벨트, 체인등을 견인 한다면

**단위당 샤프트 회전율 = 1/(tooth pitch X Ns)** 가 된다.

예를 들면, 3/8" 체인에 13개의 톱니를 가진 기어가 모터에 샤프트에 물려 있다면 단위당 모터 회전율은  $1/(0.375 \times 13) = 0.2051282$  가 된다. 수치상으로 보면 상당한 속도를 낼 수 있으므로 추가적인 기어를 사용하여 원하는 토크를 낼 수 있게 해 준다. 이 예제의 경우 단위당 모터 회전율을 기어 박스의 감소율로 곱하기 해야 한다. 예를 들어 10:1 비율의 기어 박스라면 2.051282가 된다.

**단위당 모터 회전율 = 단위당 샤프트 회전율 X Ns / Nm**

로터리 축(예: 4축용 로터리 테이블)에서 단위는 “도(degree)” 가 될 것이다. 이때 웜기어(worm) 비율로 계산 해야 하며 종종 90 : 1 비율이 많다. 그러나 모터에 직접 물린 웜기어의 한번 회전은 4도 가 되므로 단위당 모터 회전은 0.25 가 될 것이다. 2 : 1비율에서는 단위당 0.5 가 될 것이다.

#### 5.5.1.2 회전 대비 모터 스텝 계산 하기

일반적으로 스텝 모터는 한 회전 시 200 스텝의 분해도(resolution) 를 갖는다. 즉 한 스텝은 1.8도가 된다. 그리고 서보 모터의 경우는 샤프트에 장착되어 있는 로터리 인코더에 의존해서 분해도를 생성한다. 이때 인코더의 분해도는 CPR(Cycle Per Resolution) 이라고 표현한다. 두개의 수직위상 신호를 출력하기 때문에 실제 분해능은 거기에 4배의 값을 적용한다. 125에서 2000 회의 CPR을 갖는다면 이에 대응하여 500에서 8000 스텝당 회전율을 갖출 수 있게 되는 것이다.

**메모 [DP15]:** 보통 2상 모터의 경우를 이야기 하며 0.9도의 분해능을 갖는 모터도 있다.

#### 5.5.1.3 모터 회전 대비 Mach3 스텝 계산 하기.

스텝 모터 드라이버를 사용 할 때 마이크로 스텝핑이 지원되는 모터 드라이버를 사용 하라고 강력히 권고 하는 바이다. 그렇지 않고 풀 스텝(full step) 이나 1/2 스텝(half step)만 사용 하는 드라이버를 된다면 커다란 모터를 사용 하는 편이 바람 직 할 것이며 일정 속도이상의 성능에서는 공명(resonance) 현상으로 작업을 성공적으로 끝마치는 데 문제가 생길 수 있다.

어떤 마이크로 스텝 드라이버는 고정된 마이크로 스텝(일반적으로 10)으로 운영 하며 어떤 것들은 설정에 따라 다르게 움직이기도 한다. 10 마이크로 스텝 일 경우 Mach 3 는 한 회전당 2000 펄스를 축 드라이브에 전송 한다.

어떤 서보 드라이브는 모터 인코더로부터 수직위상(quadrature)신호 당 한 펄스가 되기를 요구한다.(만약 300 CPR 인코더라면 한 회전당 1200 스텝이 된다.) 또 다른 것들은 전자적 기어로 입력스텝에 상수로 곱하거나 나눠 값을 구하기도 한다.

스텝 수에 곱하기를 하여 값을 구하는 것은 Mach 2에 상당히 유용 한데 그것은 고 분해능 인코더가 달린 작은 규모의 모터 속도가 Mac3가 만들어 내는 최대 펄스비율 에 의해 제한 될 수 있기 때문이다.

#### 5.5.1.4 단위 당 Mach3 스텝

결론 적으로, **Mach 3 의 단위당 스텝 = Mach2 의 회전당 스텝 X 단위당 모터의 회전수**

아래 그림 5.11 에서는 Config > Motor tuning 다이얼 로그 박스를 보여 준다. 설정 하려는 축을 선택 하기 위해 버튼을 누르고 계산된 Mach 3 steps per unit 값을 그림처럼 상자에 넣는다. 넣을 숫자가 정수가 될 필요는 없으므로 가능한 정확한 기입 한다.

**절대로 Save axis Settings 버튼을 클릭 하는 것을 잊지 말아라.**

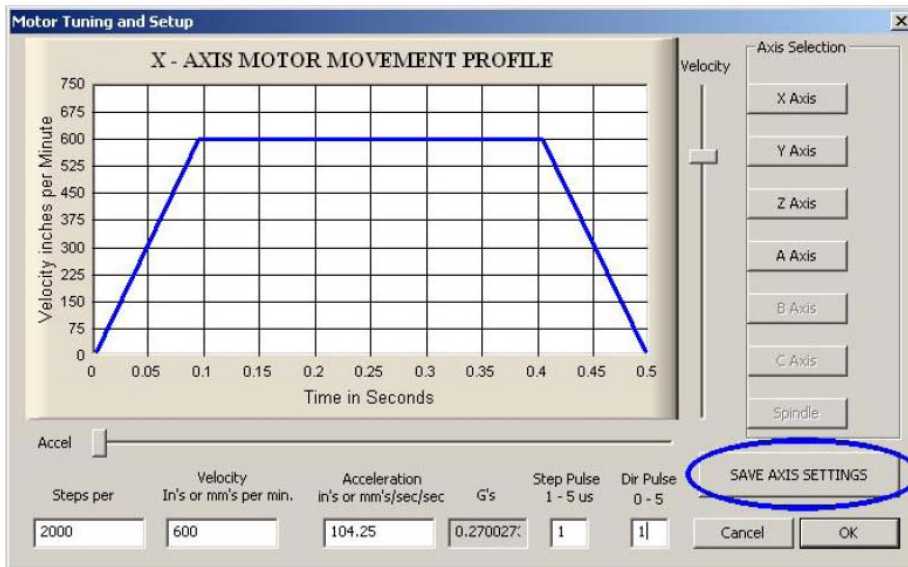


그림 5.11 모터 튜닝 대화 상자

### 5.5.2 최대 모터 속도 설정

Config > Motor 윈도우에서 Velocity 슬라이더 바를 움직임에 따라 시간 대비 velocity(속도)를 짐작 할 수 있는 그림이 나온다. 축은 가속 했다가 그 다음 감속 한다. 일단 Velocity 를 최대로 놓는다. Acceleration(가속) 슬라이드 바를 움직여 감 가속도를 조정해 본다.

각 슬라이더 바를 움직임에 따라 상자 안의 대응 값 또한 변하는 것을 볼 것이다. Velocity는 단위당 초로 표시 되고 Accel 은 단위당 초의 제곱으로 표시된다. 가속도의 값은 Gs로도 나타 나는데 이것은 테이블이나 혹은 작업 대상물에 가해지는 힘을 나타 낸다. 최대 속도는 Mach3 가 제공하는 최대 펄스율 에 의해 제한 된다. 예를 들어 25,000Hz 와 2000 단위 당 스텝으로 설정 되어 있다면, 가능한 최대 속도는 750 단위 당 초가 될 것이다.

이때의 최대치란 단지 Mach3가 최대로 가동 할 수 있는 값이며 실제로 사용 할 때는 직접 시도를 해 가면서 개인이 판단 하면 된다.

#### 5.5.2.1 모터 속도의 실질적 효용

단위당 스텝 을 설정 한후 각 축에 대한 설정을 끝냈다. 이제 대화창에서 저장후 빠져나와 Reset 버튼을 누르면 LED 의 깜빡임이 사라진다.

다시 Config>Motor Tuning 으로 가서 축을 선택한다. 슬라이드 바를 조절하여 최대 가용 속도의 20% 정도로 설정 한다. 그리고 모터를 움직여 보고 만약 빠르다고 생각 된다면 다시 창으로 돌아가 모터의 속도를 감속해 본다. 반대의 경우도 마찬가지다. 커서 Up 키와 Down 키로 속도를 조절 할 수도 있다.

움직임의 방향이 원하는 방향과 반대로 간다면 a) Config>Ports&Pins>Motor Output 에서 Dir 핀 에설정된 Low Active 설정을 바꾸거나 b) Config > Motor Reversals 로 가서 혹시 방향을 반대로 하지 않았는지 확인해 본다. 다른 방법으로는 A, B 쌍으로 된 모터의 배선을 바꿀 수도 있다.

만약 스텝퍼 모터에서 높은피치의 이상한 소리를 내거나 덜덜거린다면 결선이 잘못 이루어 진 것이거나 혹은 속도를 너무 빠르게 설정해 놓은 것이다. 스텝모터의 결선 라벨링이 가끔 혼선을 초래 하기도 한다. 모터를 제공하는 회사의 매뉴얼을 꼭 참고 하기 바란다.

만약 서보 모터가 최대 속력으로 날라 다니며 드라이버에 오류 데이터를 보낸다면 전기자(armature) 혹은 인코더 연결을 반대로 해야 한다.(그 회사에서 제공되는 매뉴얼을 확인해 본다.) 한번 구매 하여 한번에 성공 하길 바란다.

대부분의 드라이버는 1마이크로 최소 펄스 폭에서 잘 작동한다. 처음 작동에서 위의 설명과 같은 증상이 있다면 스텝 펄스가 반대로 설정 되어 있지 않은지 확인해 보고 펄스 폭을 대략 5 마이크로 초(microsecond) 로 설정해 본다. 스텝과 방향 인터페이스는 매우 단순하지만 그래도 이것역시 일종의 작업이라 할 수 있기 때문에 한 번 잘못 설정 되면 무엇이 문제 인지 쉽게 찾지 못할 수 있다.

### 5.5.2.2 모터의 최대 속도 계산 하기

최대 모터 속도를 계산 하고자 한다면 이번 섹션의 아래 내용을 읽어 보기 바란다.

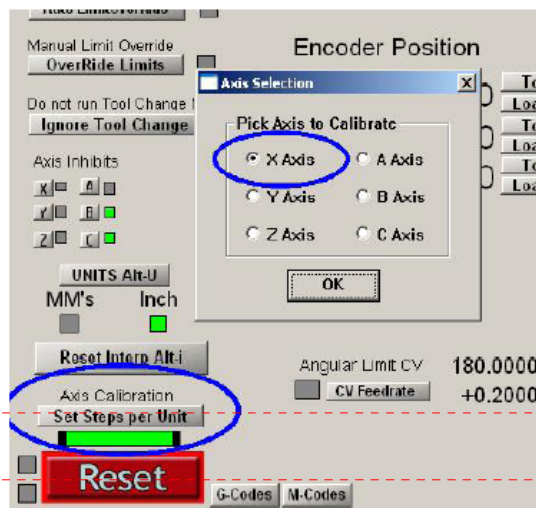
축의 최대 속도에 대한 정의는 여러 가지로 기술 할 수 있다

- 모터에 가할 수 있는 최대 속도(아마도 서보용으로 4000rpm정도나 혹은 스텝모터 용으로 1000rpm 일때)
- 볼 스크류에 가할 수 있는 최대 속도(길이, 직경, 끝 부분이 어떻게 물려 있는가 등의 영향)
- 감속 기어 박스 혹은 벨트 드라이브의 최대 속도.
- 결함없이 신호 처리 할 수 있는 보드의 최대 속도
- 장비 이송부위의 원활 성을 유지 할 수 있는 최대 속도

위의 처음 두 항목은 여러분이 공감 할 것이다. 제조자의 기술적 사양을 참고 해야 하며 스크류와 모터에 가 할 수 있는 속도를 계산 하고 축의 움직임에 초당 단위(인치, 밀리) 거리를 적용 시킨다. 그리고 이것을 Motor Tuning > Velocity 상자에 값을 넣는다.

### 5.5.2.3 단위당 스텝, 자동으로 설정하기

축드라이버에 사용하는 기어박스의 비율을 계산 할 수 없거나 혹은 스크류의 정확한 피치를 잘 모를 경우가 있을 수 있다. 이럴 경우 만약 다이얼 게이지나 혹은 캘리퍼 같은 것을 이용하여 움직이는 축에 장착하여 실제 얼마나 움직였는지 볼 수있게 하여 그 움직인 값을 그림 5.12와 같이 진행하여 실제 얻은 값을 Mach3 에 저장 하면 된다. 옆의 그림과 같이 Set Steps per Unit 을 누르고 원하는 축을 선택한 후 OK 를 누르면 다음 창이 나오고 거기에 움직이고 싶은 값을 넣는다. 그러면 축이 움직이고 여러분은 축에 장착한 DRO 혹은 캘리퍼, 다이얼 게이지 등등, 실제 움직인 값을 읽을 것이다. 그러면 그 다음에 나타날 창에 실제 값을 입력 시키면 여러분은 설정이 끝난 것이다. 주의 할 것은 EMO 버튼을 누를 준비를 한다. 왜냐하면 초기 얼마만큼 움직이는지 알 수 없기 때문에 충돌의 염려가 있기 때문이다.



**메모 [DP16]:** 이것을 두번 이상 진행해 보고 재연성을 확인해 보기 바란다.

**메모 [DP17]:** 처음 아주 작은 값부터 시작하여 차차 큰 값을 넣으면 충돌을 미리 방지 할 수 있다.

그림 5.12 자동 설정

### 5.5.3 모터의 가속도 설정하기

#### 5.5.3.1 관성과 힘

어떠한 모터도 장비 안에서 즉석으로 속도 변환을 할 수 없다. 회전체에는 회전 운동량을 주기 위해 토크(torque)라는 것이 필요 하다. 이 힘으로 변환된 토크는 장비의 움직이는 부분을 가속 하게 된다. 이 힘의 일부가 물론 마찰력을 이기고 물체를 절삭 하는 역할을 한다.

Mach3는 주어진 비율로 모터를 감속 또는 가속 시킨다. 모터에 절삭, 마찰 그리고 관성력 보다 많은 토크를 가할 수 있다면 아무 문제 없을 것이다. 반대로 그 보다 적은 토크를 가 하게 된다면 스텝모터 같은 경우 정지 하거나 서보 모터 같은 경우 서보의 위치 에러가 증가 할 것이다. 서보의 에러가 아주 클 경우 드라이버는 결함 상태를 나타 내거나 그렇지 않더라도 절삭의 정밀도에 심각한 영향을 줄 것이다.

#### 5.5.3.2 각기 다른 가속도 값의 실험

모터 튜닝 창에서 각기 다른 가속도 값으로 장비를 움직이고 멈추고 해 본다. 낮은 가속도 에서는 어쩌면 모터의 감/가속에서 오는 소리를 들을 수 있을 것이다.

#### 5.5.3.3 서보 모터의 에러를 피해야 하는 이유

파트 프로그램에 의한 장비 축의 움직임은 두 개나 그 이상의 좌표 값의 조합으로 이루어 진다. 그러므로 X=0, Y=0 에서 X=2, Y=1 로 움직인다고 할 때, Mach3 는 Y 보다 X를 두배의 속도로 움직 이게 될 것이다. 이것은 단지 등속운동으로만 조합 할 뿐만 아니라 감 가속 할 때 필요한 운동 관계를 적용 시킨다. 그러나 가속 되는 모든 움직임은 가장 느린 축을 기준으로 이루어 진다.

만약 어떤 축에 지나친 가속을 설정 한다면 비록 Mach3 가 그것을 받아 들이더라도 현실 적으로 설정을 따라 가지 못하여 궤도를 따라 절삭 할 때 정밀도가 떨어질 것이다.

#### 5.5.3.4 가속도 값의 바른 선택

절삭 대상의 질량, 스크류와 모터의 관성 모멘트, 마찰력과 모터로 부터의 토크 등에 대한 값을 알 때 얼마 만큼의 가속 값이 필요 한지 계산 할 수 있다. 볼 스크류와 선형 레일 제조자 들은 카탈로그에 일반적으로 필요한 데이터 값의 계산 방법을 제시해 준다.

장비가 끌어 낼 수 있는 최고의 성능을 아주 세밀하게 분석 하는 수고가 힘들다면 적당히 값을 설정하면서 기계에 무리가 가지 않는 방향으로 하라고 권고한다. 물론 아주 과학적이지는 않지만 어쩌면 이것이 이상적인 사용이 아닌가 싶다.

### 5.5.4 축의 테스트 및 저장

마지막으로 계속 진행 하기 전에 Save Axis Settings 버튼을 눌러 저장 하는 것을 잊지 말아야 한다.

MDI 창에 정의된 G0를 실행하여 움직인 계산값을 확인해 본다. 대략 적인 측정으로 자 같은 것을 이용하면 된다. 더 정확한 것을 원한다면 Dial Test Indicator(DTI) 와 슬립 게이지 블록을 이용한다.

예를 들어 X 축을 테스트 하기 위해 4 인치 게이지 블록을 사용 한다고 가정 한다.

MDI 화면을 이용하여 인치 단위와 절대 좌표(absolute coordinates) 를 선택 한다. (G20 G90) 테이블의 블록을 셋업하고 DTI 프로브에 닿도록 축을 조그 하여 이동 한다. X 가 마이너스 방향으로 움직이는 것으로 마무리 된다는 것을 명심한다.

베즐(Bezel: 홈) 을 돌려서 값이 영이 되게 한다. 그림 5.13 와 같다.



이제 Mach3 의 MDI 화면에서 G92X0 버튼을 눌러 옵셋을 설정하고 X 축 DRO 가 0이 되게 한다.

다시 G0 X4.5 를 입력하여 X = 4.5 지점으로 움직이게 한다. 간격이 대략 0.5 인치 가량이 될 것이다. 만약 이와 같지 않다면 단위당 스텝 계산에 문제가 있다고 볼 수 있다. 다시 한번 검사해 보라.



그림 5.13 영점 위치 조절을 위한 과정

게이지 블록을 삽입하고 G0 X4를 MDI 창에 입력하여 X = 4.0 좌표로 움직이게 한다. 이것은 X 축 마이너스 방향으로 움직이며 기계의 백 레쉬가 발생하지 않게 할 것이다. DTI 에 읽힌 값은 위치설정 에러를 나타낼 것이며 대략 1000 정도가 된다. 그림 5.14는 게이지의 위치를 나타낸다.

게이지를 제거 하고 G0 X0 로를 입력하여 0이 되는 가를 확인 한다. 위의 순서를 반복하여 얼마만큼의 재연성이 있는가를 확인해 본다. 만약 결과치가 많이 다를수록 장비에 뭔가 문제가 있다는 것을 의미 하며 만약 같은 수치로 에러가 난다면 단위당 스텝 설정값의 문제이니 세밀한 조정으로 단위당 스텝 값을 얻어 최대한 정밀도를 산출해 내도록 한다.

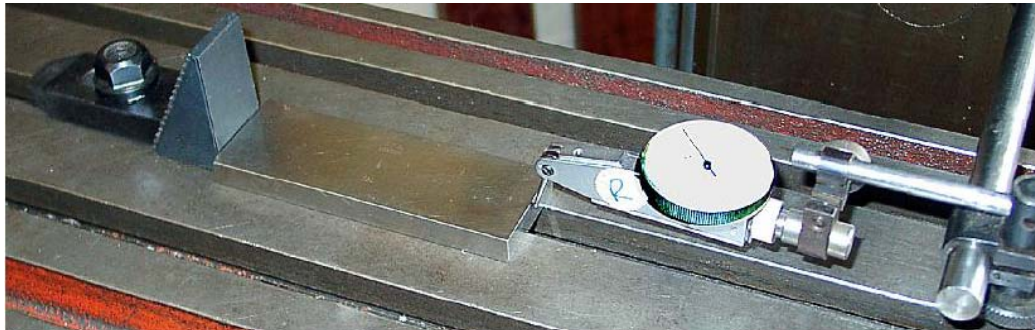


그림 5.14 게이지 블러의 위치

다음으로 축이 반복적인 일정 속도에서 탈조 하는 스텝이 없는지 확인해야 한다. 게이지를 제거하고 MDI에서 G0 X0로 하여 DTI 가 영이 되는지 확인한다. 편집기를 사용하여 다음 프로그램을 입력 한다.

```
F1000
G20 G90
M98 P1234 L50
M30
O1234
G1 X4
G1 X0
M99
```

Cycle Start 를 눌러 실행하여 움직임이 부드러운지 살펴 본다.

이과정이 끝나면 DTI 에 읽히는 값은 0이 된다.

### 5.5.5 다른 축의 제 설정

첫 번째 축에 대한 설정을 잘 마무리 했으면 다음 축 역시 어렵지 않게 제 설정 할 수 있을 것이다.

### 5.5.6 스피들 모터 셋업

스핀들 모터의 속도가 일정하거나 혹은 수동으로 조절이 된다면 이번 섹션은 그냥 넘어 가도 좋다. Mach3에 의해서 모터가 on off 되고 양쪽 방향으로 제어 된다면 릴레이의 출력에 의한 셋업이 이루어 져야 할 것이다.

Mach3가 방향, 스텝펄스를 제어하는 서보모터나 PWM 모터 컨트롤러에 의해 제어 된다면 이번 섹션은 읽어 두기 바란다.

#### 5.5.6.1 모터 속도, 스피들 속도 그리고 풀리(pulley)

스텝과 방향 그리고 PWM, 두 가지 모두 모터의 속도를 제어 할 수 있게 한다. 장비를 운영할 때 운영자와 파트프로그램이 고려하는 것은 스피들의 속도다. 모터와 스피들 속도는 기어와 풀리(pulley)에 관계된다.



그림 5.15 풀리(pully) 스피들 드라이브

만약 모터 속도 제어를 제어 할 수 없다면 Pulley 4 를 선택해서 대략 10,000 rpm 정도의 최대 속도로 만든다. 그러면 대략 6000 rpm을 요구하는 S 문자로 시작하는 프로그램 에서 Mach3는 오류를 범하지 않을 것이다.

Mach3 는 풀리(pulley)의 비율이 얼마인지 스스로 알지 못하기 때문에 선택에 대해서는 각자가 책임을 져야 한다. 사실 이에 관한 것은 두 단계로 나뉘어 진다. 시스템이 설정될 때, 4개 까지 풀리(pulley)의 조합을 사용 할 수 있다. 이것을 물리적 크기에 따라 설정하고 그 다음 파트 프로그램이 진행 되면 1-4 까지 어떤 풀리(pulley)가 사용 되는지 정의 내린다. PWM 에서 최대 속도는 100% 펄스 폭 일 때 얻어 지며 Motor Tuning 의 스텝과 방향을 위한 스피들 축에 Vel 값이 있다.

예를 들어, 풀리 1 이 모터로부터 스피들에 5:1 비율로 감속 시키고 최대 모터의 속도를 3600rpm 이라고 가정 하자. Config >Logic 에 있는 풀리 1 의 최대속도는 720rpm(3600/5) 이다. 풀리 4은 4:1 비율로 가속 시킨다. 같은 속도의 모터라면 최대 속도는 14,400rpm (3600X4) 가 될 것이다. 다른 풀리들은 중간 정도의 비율이 된다. 풀리가 속도 증감을 위해 정의 내려질 필요는 없다고 하여도 논리적인 차원에서 제어되는 수치는 있어야 한다.

최저 속도(Minimum Speed) 값은 모든 풀리에 대등하게 적용되며 최대 속도의 퍼센트로 표현된다. 또한 PWM 신호율의 최소 퍼센트로도 표현이 된다. 요구 되는 속도 치수 보다(S 문자나 기타에 의해) 적다면 Mach3는 저속 범위내에서 진행하며 풀리 비율을 바꿔야 한다는 요청을 할 것이다. 예를들면, 10,000rpm 의 최대 속도로 풀리4가 있다고 할 때 최소 퍼센트 5%라고 한다면 S499는 다른 풀리를 요청 할 것이다. 이것은 모터나 컨트롤러가 최소 요구 조건 보다 낮게 운영되는 것을 막기 위함이다.

일단 Config > Logic안의 다른 제어 사항은 나중에 이야기 하기로 한다.

Mach3의 풀리 비율에 관한 사항은 아래와 같다.

-파트 프로그램이 S문자를 실행 하거나 혹은 속도 설정 DRO 창에 값이 입력 됐다면 그 값은 현재 선택된 풀리의 최대 속도를 위한 값과 비교가 된다. 만약 요구되는 속도가 최대 값 보다 더 나온다면 에러가 발생 할 것이다.

- 그렇지 않다면 원하는 풀리의 최대 퍼센트 비율이 PWM 폭 설정에 사용 되거나 Motor Tuning 의 스피들 축(Spindle Axis)에 설정 되어 있는 것과 같이 스텝펄스가 생성되어 모터 속도의 최대 퍼센트를 만들어 낸다.

예를 들어, 풀리 #1의 최대 스피들 속도가 1000rpm 이라고 한다. S1100은 에러를 나타낼 것이다. S600은

60%의 펄스 폭을 나타낼 것이며 만약 최대 스텝 방향 속도가 3600rpm 이라면 모터는 2160rpm 으로 스텝핑 할 것이다.(3600 X 0.6)

### 5.5.6.2 PWM 스펀들 컨트롤러

PWM 컨트롤에 의한 스펀들 모터를 설정 하려면 Spindle Axis 가 Enable 되어 있는지 확인 하고 PWM Control 상자가 체크 되어 있나를 살펴 보라( Port and Pins > Printer port > Axis selection : 그림 5.1)

Apply 버튼을 누르는 것을 잊지마라. Spindle Step을 위해 Output Signal Selection 탭에 있는 Output 핀을 정의 내린다.(그림 5.4) 핀은 반드시 PWM 모터 컨트롤 회로에 연결이 되어 있어야 한다. Spindle Direction 은 필요 없으니 0 으로 설정한 후 Apply 버튼을 누른다.

Ports and Pins 에 있는 External Activation 을 정의 내리고 Configure > Output Device 에서 PWM 컨트롤러 스위치의 on/off를 선택한다. 필요에 따라서 회전 방향을 선택 한다.

Configure > Logic 창으로 이동하여 PWMBase Freq 상자를 찾는다. 이 안의 값은 펄스폭 변조된(Pulse Width Modulated) 사각 파형의 진동수 이다. 바로 이 신호가 Spindle Step 핀에 나타난다. 여기에 더 높은 진동수를 선택하면 더 빠르게 속도변화에 대응하지만 선택한 속도의 분해능은 낮아질 것이다.

각각 다른 속도는 Engine pulse frequency / PWMBase freq 가 된다. 그러므로, 예를 들면 35,000 Hz 의 속도로 운영하고 PWMBase 를 50 Hz로 한다면 700개의 불연속 속도가 가능할 것이다. 모터의 최대 속도가 3600rpm 일 때 6rpm 이하의 스텝으로 제어 하는 것이 실제 운영시스템에서 충분할 것이다.

### 5.5.6.3 스텝과 방향에 의한 스펀들 컨트롤러

스핀들 모터를 스텝 방향 제어를 위해 설정하려면 Spindle Axis 가 체크 되어 있는지 확인 한다. ( Port and Pins > Printer Port and Axis selection 탭 (그림 5.1). PWM Control를 체크 하지 않는다. 그리고 Apply 버튼을 누르는 것을 잊지 말라. 출력핀을 Output Signals Selection 탭에서 정의 내려 Spindle Step 과 Spindle Direction 을 . (그림 5.6) 설정한다. 이 핀들은 모터드라이브 회로에 연결 되어야 한다. Apply를 눌러 변화된 것을 저장한다.

External Activation 신호를 Ports and Pins .에서 정의 내리고 스펀들 모터 컨트롤러를 on/off 정의를 내려 M5가 스펀들을 정지 시킬 때 스펀들 모터 전원을 끄려고 한다면 Configure > Output Devices 에서 스펀들 모터 컨트롤러의 전원 스위치를 on/off를 정의 내린다. Mach3가 스텝 펄스를 보내지 않음에 따른 모터의 회전은 없겠으나 드라이버를 어떻게 디자인 했는가에 따라 전원이 서서히 소멸 될 수도 있을 것이다.

Configure > Motor Tuning으로 가서 Spindle Axis를 살펴본다. 이것의 단위는 회전으로 표시 된다. 그러므로 단위당 스텝은 한 회전당 펄스의 횟수가 될 것이다.

Vel 대화 상자에는 최고 속도일 때 초당 회전 수로 설정되어 져야 한다. 그래서 3600rpm 모터는 60 이 되는 것이다. 이것은 무수히 카운트하는 인코더에서는 불가능 한데 그것은 Mach3의 최대 펄스 비율 때문이다.(예 : 100 라인 인코더는 35,000Hz 시스템에서 초당 87.5 회전이 가능 하다) 스펀들은 일반적으로 이러한 제한된 상황을 극복 할 수 있는 전기적 기어링을 할 수 있는 강한 모터가 적합하다.

Accel 대화 상자는 스펀들의 자연스러운 시작과 정지를 위해서 실험적인 값을 정하는 것이 좋다.

주의 : 작은 값을 넣고 싶으면 타이핑 하는 것이 Accel 슬라이더 바를 움직이는 것 보다 낫다.

#### 5.5.6.4 스펀들 드라이브 테스트하기

회전속도계나 섬광 촬영장치를 가지고 있다면 스펀들의 회전 속도를 측정 할 수 있을 것이다. 그렇지 않다면, 대부분이 그럴겠지만, 경험에 의해서 눈대중으로 판단 할 수 밖에 없다.

Mach3 Setting 화면에서 900rpm이 가능한 풀리를 선택 한다. 장비의 벨트나 기어박스를 대응하는 위치에 장착한다. Program Run 화면에서 스펀들 스피드를 900rpm으로 하고 회전 시키고 속도를 측정해 본다. 잘못 되었다고 판단하면 다시 계산하고 시작 한다.

같은 방법으로 모든 풀리에서 적절한 상태의 속도를 측정한다.

### 5.6 기타 설정 사항

#### 5.6.1 원점 조정

##### 5.6.1.1 스텝과 방향 참조하기

Config>Home/Softlimit 대화창은 G28.1 이나 화면 버튼 과 같은 기준점 운영이 어떻게 정의 내려지는지 보여 준다. 그림 5.16 을 참고 하라. Speed %는 기준점을 찾기위해 전속으로 달려 스위치기에 다가 갈때 정시 하는 순간 총돌을 방지 하기 위한 속도를 설정 한다. 설정을 하려 할 때, Mach3는 축의 위치를 알지 못한다. 그리고 움직이는 방향은 Home Neg 에 체크가 되어



그림 5.16 기준점 설정 화면

있느냐에 달려 있다. 관련축에 체크 되어 있다면 축은 마이너스 방향으로 Home input이 작동 될때 까지 움직일 것이다. 그리고 Home input 이 이미 작동 됐다면 플러스 방향으로 움직일 것이다. 유사한 경우로, 상자에 체크되어 있지 않다면 축은 플러스 방향으로 입력이 작동 될때까지 움직일 것이고 만약 작동 됐다면 마이너스 방향으로 움직일 것이다.

##### 5.6.1.2 홈 스위치의 위치

만약 Auto Zero DRO 가 원점에 간 상태에서 체크 되어 있다면 축 DRO는 실제의 0점보다는 Home off 에 정의된 Reference/Home 스위치 위치 값으로 설정이 된다. 이것은 테이블이 크거나 축의 속도가 느린 장비에서 기준점을 잡는 시간을 줄일 수 있다.

##### 5.6.1.3 소프트 리미트 설정

전에 설명된 바와 같이 리미트 스위치의 작동은 장비의 안전을 위한 타협점을 찾기 위함이며 실수로 이것을 건드렸다면 작업자에 의해 조작이 필요하고 장비의 리셋이 요구 될 수 도 있다.

소프트 리미트는 이러한 불편한 사고를 방지 하기 위한 보호막 역할을 할 수 있다.

X, Y, Z 축에 소프트웨어적으로 설정된 범위 밖의 움직임을 차단하며 이때 범위는 -999999 에서 + 999999(inch 혹은 mm) 까지 가능 하다. 조그해서 움직여 리미트 근처 까지 가면 테이블 상에서 정해진 슬로우 존(slow zone)에서 속도가 감소한다. 만약 슬로우 존이 너무 크다면 장비의 효율적인 작업 범위를 작게 만들 것이다. 반대로 너무 작게 만든다면 하드웨어의 리미트에 충돌 하게 될 것이다. 정의된 리미트는 Software Limits 토글 버튼을 사용해서 on 되어 있을 때 적용될 수 있으며 자세한 사항은 리미트 과 기타 사항 패밀리를 살펴 본다.

### 5.6.1.4 G28 홈 스위치

G28 좌표는 G28이 실행 될 때 절대 좌표에서 움직인 위치를 가리킨다. 좌표값들은 현재 단위(G20/G21)로 해석되며 단위 가 바뀔 때 자동으로 전환되지 않는다.

### 5.6.2 시스템 핫키 설정

Mach3는 글로벌 핫키를 설정할 수 있어 조깅이나 MDI 창에 값을 넣을 일 등등을 할 수 있다.

이런 키들은 그림 5.17 과 같이 System Hotkey Setup 에서 설정 될 수 있다.

요구되는 기능을 위해 버튼을 누르고 핫키로 사용될 키를 누른다. 대화 상자에 값이 나타날 것이다.

코드의 사용이 중복 되는 것을 막아 심각한 혼란이 발생하는 것을 방지 하기 바란다.

이 대화 상자는 OEM trigger 로서 사용될 외부 버튼을 위한 코드를 사용 가능 하게 한다.

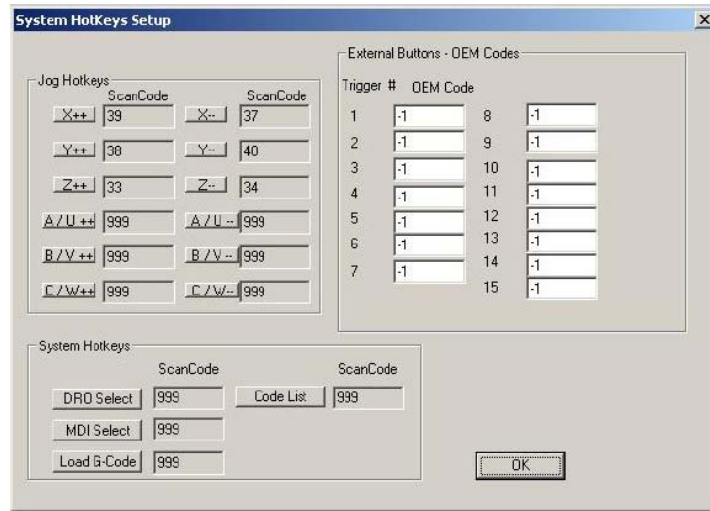


그림 5.17 - 핫키 와 OEM trigger 설정

### 5.6.3 백레쉬 설정

Mach3는 축에 생기는 백레쉬(backlash)를 각각의 축에 필요한 좌표값을 보강 할 수 있다. 비록 드릴링이나 보어링 할때 유용 하게 응용해서 사용 할 있지만 , 장비가 안고 있는 문제이기 때문에 한번에 연이어 질삭 못하고 각 축의 보정을 위해 다른 축은 잠시 멈추게 된다.

Config > Backlash 대화창에서 축의 보강해야 할 거리를 측정 해서 그 값을 입력 할 수 있다. 이런 모션 상에서의 설정되어야 할 속도 도 열거 된다. 그림 5.18을 참고 하라.

주의 : a) Config > Initial State 대화창에서 Backlash Enabled 설정이 on 되어 있을때 만 사용 할 수 있다.

b) 백레쉬 보강은 장비 디자인 자체에서 개선이 불가능한 경우 마지막 보루이다.

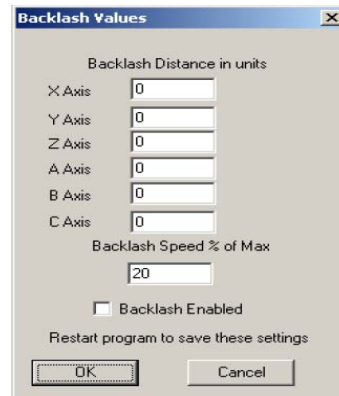


그림 5.18 Backlash 설정

c) Mach3는 백레쉬 보강이 될 때 축의 가속 파라미터에 대한 확고한 대응은 할 수 없으므로 스텝퍼 시스템에서는 다시 튜닝 하여서 스텝이 빠지는 위험을 방지 해야 할 것이다.

### 5.6.4 슬레이브 설정

켄트리 라우처 처럼 커다란 장비는 종종 한 축에 두개의 드라이브가 필요한데, 켄트리의 양끝에 하나씩 사용을 한다. 만약 스텝이 어긋난다면 축에 균열이 생겨 나머지 다른 축과 직각을 유지 못하게 될 것이다.

Mach3를 설정하여 한 축(X 라고 하자)에 “ 주 ” 드라이버 또다른 “ 부 ” 드라이브(슬레이브)를 정의 내릴 수 있다. 그림 5.19 참고.

일반적으로 같은 수의 스텝펄스가 주(마스터) 축과 부(슬레이브) 축에 보내져 속도 그리고 가속도가 둘 중에 느린 쪽을 기준으로 정해진다.

기준점 설정이 요청 됐을 때 주와 부가 한쪽의 홈 스위치에 닿을 때 까지 같이 움직일 것이다. 이때 한쪽은 다른 한쪽을 따라 같이 움직이게 되므로 한 쌍의 축은 홈 스위치에서도 서로 직각을 유지 하게 된다. 비록 Mach3는 주(마스터)와 부(슬레이브)를 스텝으로 관리 하지만 슬레이브 축의 DRO 는 툴 테이블에 적용된 옵셋을 나타내지 않을 것이다. 때문에 작업자로 하여금 혼선을 줄 수도 있다는 것을 알아 두기 바란다. 그러므로 Screen Designer 를 사용 하여 슬레이브 축 DRO 및 관련 컨트롤을 화면에서 나타내지 않도록 할것을 권한다. (단 Diagnostics 창 제외).

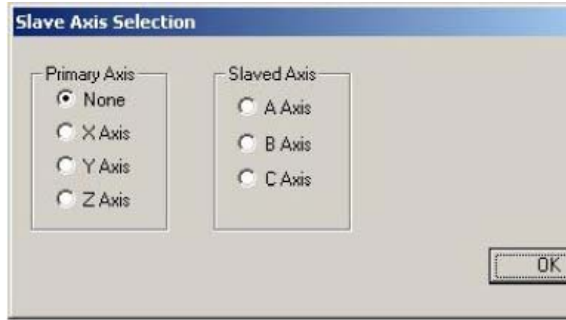


그림 5.19 슬레이브 설정

다른 이름으로 저장 하기에서 기본 값이 아닌 다른 이름을 설정 하여 저장 하고 View > Load Screen 메뉴를 사용하여 Mach3 에 로딩 할 수 있도록 한다.

### 5.6.5 툴 패스(tool path) 설정

Config > Toolpath 는 툴패스가 어떻게 표현되는지 정의 내릴 수 있도록 한다. 이 대화창은 그림 5.20과 같다. Origin Sphere 가 체크되면 : X=0, Y=0 Z=0로 표시되면서 툴패스 화면의 포인트 점에 등근 점을 보여 준다.

3D compass : 툴패스 화면의 X,Y,Z의 플러스 방향을 묘사하는 화살표를 보여준다.

Machine boundaries : 소프트리미트 설정에 해당하는 상자를 표시한다.(소프트 리미트의 on/off에 관계없이)

Tool Position : 화면상에서 현재의 툴(스핀들) 위치를 표시한다.

Jog follow Mode : 툴이 조그 됨에 따라 화면 상에 상대적으로 움직이는 툴패스의 선을 보여 준다. 달리 말하면, 툴패스 화면 창에서 툴의 위치는 고정되어 나타난다.

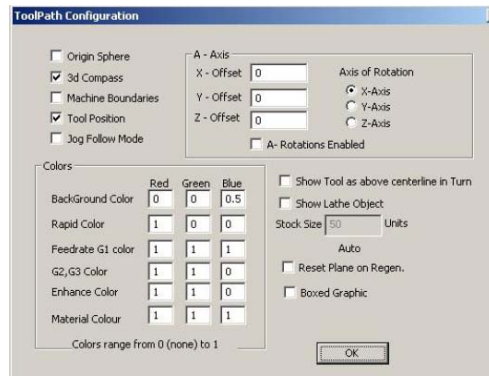


그림 5.20 툴패스 설정

Show Too as above centerline in Turn 은 Mach3Trun 프로그램(CNC 선반에 쓰이는)에서 사용 한다.

Show Lathe Object : 톨패스를 따라가는 절삭 대상물의 3D 렌더링을 가능 하게 한다.(단 Mach3Turn 에서만 가능)

Color : 각각의 화면 구성요소의 색은 설정 될 수 있다. 삼원색, 빨강, 녹색, 청색 의 밝기는 선의 타입에 따라 0에서 1까지 설정 될 수 있다. 힌트 : 포토샵 같은 프로그램으로 원하는 색을 만들고 RGB 값을 256 등분하여 Mach3에 적용 시킨다.

A axis 값은 A축이 로터리(rotary) 타입으로 설정되어 있다면 위치와 방향을 설정 할 수 있게 해준다.

Reset Plane on Regen 은 톨패스 화면을 재생성 시킬 때 마다 현재의 화면으로 복귀 시킨다.

Boxed Graphic 툴 움직임의 경계선에 박스를 표시해 준다.

### 5.6.6 초기 상태 설정

Config > State 에서 창을 열어 Mach3가 로딩 됐을 때 활성화된 모드를 정의 내릴 수 있다. (예 : 시스템의 초기 상태) 그림 5.21 참조.

**Motion mode** : Constant velocity 는 G64를 설정하고 Exact Stop 은 G61을 설정한다. 이에 대한 자세한 사항은 10장을 참고 하기 바란다.

**Distance mode** : Absolute 는 G90을, Inc 는 G91을 설정한다.

**Active plane** : X-Y 는 G17을, Y-Z는 G19을 X-Z는 G18을 설정한다.

**I/J Mode** : 추가로 호를 그리며 움직일 때 I&J에 해석을 넣을 수 있다. 이것은 다른 CAM의 후 공정(post process) 호환성을 위해 제공되며 다른 장비의 컨트롤러를 에뮬레이트 하기 위해서이다. Inc IJ mode 에서 I와 J(중심점)는 중심 형 호의 시작점으로부터 상대적으로 해석 되고 NIST EMC 와 호환을 이룬다. Absolute IJ mode 에서 I와 J는 현재의 좌표 계에서 좌표 중심이 된다.(즉, 작업이 끝난 후 장비와 G92는 옵션이 생긴다.) 만약 원이 화면에 제대로 나타나지 않거나 절삭 하는데 실패 한다면 IJ mode 가 파트프로그램과 호환 되지 않은 것이다.

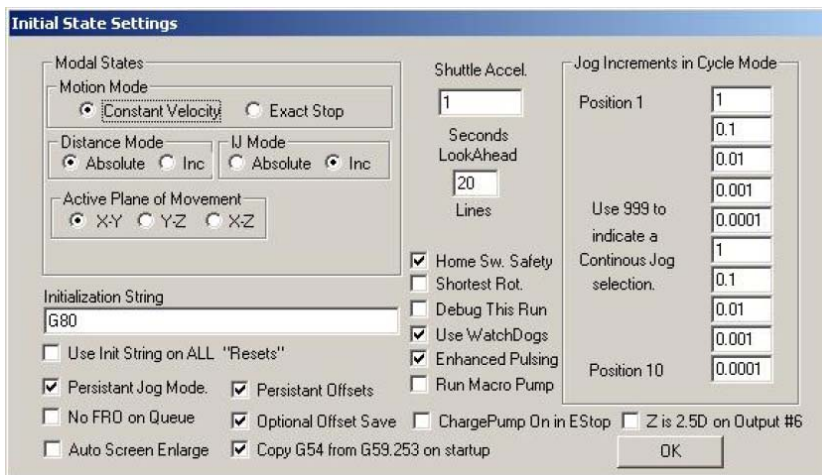


그림 5.21 초기 상태 설정

**이 설정에 관한 에러는 원을 자르려고 할 때 가장 많은 문제의 원인이 된다.**

**Initialization(Init) String** : 유용한 G code의 그룹으로서, Mach3가 시작 할 때 원하는 초기 상태를 설정한다. 이것은 값이 설정된 다음에 라디오 버튼을 체크하여 적용되고 또 반복 될 수도 있다. 라디오 버튼을 사용하여 언제든지 애매한 부분을 만들지 않도록 한다. 만약 “ Use Init on All “ Resets” 이 체크 되었다면 Mach3가 리셋 되어도 코드는 적용이 된다.

**다른 체크 항목 :**

Persistent Job Mode: Mark3Mill 프로그램이 진행 하는 동안 선택된 Jog Mode를 기억 한다.

Persistent Offsets : Mach2Mill 의 진행에 있어 선택된 작업대에서의 작업과 들 옵셋을 저장 한다. Optional Offset Save를 참조하라.

Optional Offset Save : 영구적 옵셋을 저장 할 것인가를 물어 본다.

Copy G54 from G59.253 on startup: 체크 되어 있다면 G54 옵셋(즉 작업 옵셋 1) 값을 Mach3가 시작 할 때 작업 옵셋 253 값으로 부터 다시 초기화 한다. 그전 사용자가 바꿔 놓거나 비 표준의 설정 값을 저장 하였다 하여도 G54가 고정된 좌표 시스템이 되게 하려면 이것을 체크한다. (예 : 장비의 좌표계)

Enhanced Plusing : 추가적인 CPU 프로세스 할당으로 타이밍펄스의 최고 정밀도를 확인 한다. 오직 1.2GHz 프로세서나 그 이상에서 만 사용 해야 한다.

Run Macropump : 현재 프로파일의 매크로 폴더에 있는 Macropump.m1s 파일을 찾기 시작하여 200 밀리초(milli second) 마다 한번씩 실행한다.

Auto Screen Enlarge : 현재 PC 스크린 화면 모드보다 적은 픽셀이라면 Mach3의 모든 화면과 그의 대상을 확대 하여 전체 화면에 꼭 차게 한다.

Charge pump On in EStop, Estop 이 탐지되도 충전펌프(charge pump) 출력(들)은 지속 될 것이다.

Z is 2.5D on output#6 가 체크되면 Output#6 가 제어 되는데 Z 축의 프로그램 좌표계의 현재 위치에 따라 좌우 된다.

만약 Z>0.0 면 Output#6 는 활성화 된다. 이 기능을 사용 하기 위해서는 반드시 Z 축이 설정되어야 하지만 스텝과 방향 출력은 가상의 핀으로 설정 할 수 있다. (예 : Pin 0, Port 0)

**Shuttle Accel** 은 G 코드 라인의 실행을 제어 하는 중, Mach3의 MPG 에 대한 반응을 제어한다.

**Lookahead** 는 해석기가 실행을 위해 버퍼링 할수 있는 G코드 라인의 수를 결정 한다. 일반적으로 튜닝이 필요치 않다.

**Jog Increments in Cycle Mode**: Cycle Jog Step 버튼은 리스트에 있는 값을 Step DRO 에 차례로 로딩 한다. 이것은 종종 Step DRO 타이핑 하는 것보다 수월 하게 쓰일 수 있다. Cont Jog Mode 로 전환 하기 위해 특수한 값 999를 코딩 한다.

**Reference Switch Loc**: 이 값은 장비의 좌표위치를 정의 내려 각 축의 홈 스위치에 닿은 후 기준 점에 설정되도록 한다. 이 값은 셋업 단위의 절대 위치가 된다.



### 5.6.7 다른 논리 항목 설정

Config > Logic 대화 창의 여러 기능들은 그림 5.22와 그 이하에 열거되어 왔다. 나머지 것들은 여기서 설명하도록 한다.

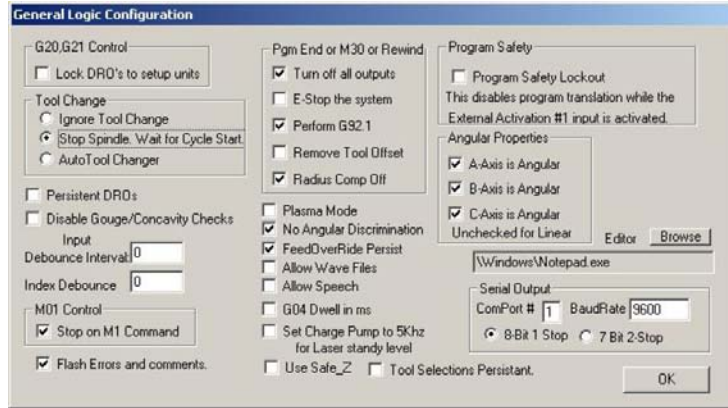


그림 5.22 - 로직 설정 대화 상자

**G20/G21 Control:** Lock DROs to set up Units이 체크 됐다면 비록 G20와 G21가 XYZ의 설정된 단위 방식이 바뀌더라도 DRO는 항상 Setup Unit 시스템에 표시 될 것이다.

**Tool Change:** M6 툴 체인지 요청은 무시되거나 M6 메크로(q.v.)를 불러들이기 위해 사용 될 수 있다. 만약 Auto Tool Change가 체크 되어 있다면 M6Start/M6End 메크로가 불러 질 수 있지만 Cycle Start 버튼은 누를 필요가 없다.

**Z-Inhibit :** Z가 허락하는 한도의 최저 값으로 리미트를 설정한다. 예를 들면, Z Inhibit -2.1로 설정되어 있다면 G0 Z-2.5는 Z=-2.1에 그대로 남게 된다.

**Angular properties:** 곡률로 정의되는 축은 “도(degree)” 로 측정된다.(말하자면, G20/G21 이 A,B,C의 해석(interpretation)을 바꾸지 않는다는 것이다.)

**Program end or M30 or Rewind:** 파트 프로그램을 되돌릴 때나 끝점에서 취할 순서를 정의 내린다. 필요한 기능을 체크 하라. 주의 : 옅색을 제거하거나 G92를 실행하기 위한 아이템에 체크 하기 전에 어떤 기능인지 확실히 알아야 하며 그렇지 않을 경우, 파트 프로그램을 다 끝내고 나서, 현재의 위치가 예상 했던 것과 현저히 다르다는 것을 알게 될 것이다.

**Debounce interval / Index Debounce :** Mach3의 펄스 수로, 스위치는 유효한 신호로 간주되기 위해 안정성이 있어야 한다. 그래서 35,000Hz로 운영하는 시스템에서는 100 은 3 밀리초(milliseconds) 디바운스(debounce) 되게 한다.(100 / 35,000 = 0.0029초) 인덱스 펄스와 다른 입력들은 서로 독립적 설정을 가지고 있다.

**Program safety:** Input #1을 커버 안전장치로 가동 시킨다.

**Editor :** G code 편집버튼으로 불러 올 수 있는 실행가능 편집기 파일. 브라우즈 버튼은 적절한 파일을 선택 할 수 있게 해준다. (예: C:\Windows\Notepad.exe)

**Serial output :** COM 포트 번호를 정의내려 직렬 출력 채널과 출력해야 할 때의 버드 레이트(baud rate)로 사용 할 수 있게 해 준다. 이 포트는 메크로의VB 스크립트로 작성될 수 있고 장비의 특수 기능 제어에 사용 할 수 있다. (예 : LCD 디스플레이, 툴 체인저, 축 클램프, 등등)

**G04 Dwell in Milli seconds** : 체크 되어 있을 때, G4 5000 명령어는 진행 중 5초간 머무르게 하며 체크 하지 않을 때는 1시간 23분 20초 동안 머무르게 한다.

#### 기타 체크 박스 :

Persistent DROs: 체크했다면, 축 DRO는 시작 할 때 Mach3 마지막으로 기록한 DRO 값을 갖게 된다. 알아둘 것은 장비에 전원을 off 시켰다면 DRO에 읽히는 값과 실제 장비 위치와는 같지 않을 것이다.

Disable Gouge/Concavity checks: 체크가 안 됐다면, 커터의 보정(G41, G42)을 실행 하고 있을 때, Mach3 는 엔드 밀의 직경이 안쪽 코너를 절삭 할 수 없을 만큼 크지 않은지 검사한다.

Plasma Mode: 체크 됐다면, 플라즈마 커터의 특성에 맞추기 위해 Mach3는 등속 운동 제어를 하게 된다.

No Angular Limit: 이것역시 일종의 등속 운동 작업 과 관련이 있다. 체크 되어 있지 않다면 Mach3는 CV Angular Limit DRO 에 있는 값 보다 큰 각도 값을 갖는 방향의 변화를 정확한 정지로 간주 하게 되어 예각인 코너의 과도한 라운딩이 되는 것을 방지 한다. 등속 운동(constant velocity)의 자세한 사항은 10장을 참고 한다.

FeedOverride Persists: 선택된 피드 오버라이드는 파트 프로그램의 끝 까지 유지 된다.

Allow Wave files: 체크 하게 되면, Windows 의 .wav 소리, 음성 파일을 Mach3 에서 실행 할 수 있다. 이것은 신호 에러나 혹은 주의를 요구 하는 창을 띄울 때 사용 할 수 있다.

Allow Speeh: 체크 하게 되면, Mach3는 Macrosoft Speech Agent 를 사용하여 정보 메시지나 오른 버튼 도움말 기능을 사용할 수 있게 한다. Windows 의 제어판에 가서 말하기 기능을 열어 사용할 음성을 설정 하기 바란다.

G40 Dwell param in Milliseconds: 체크 됐다면, G4 5000 명령어는 프로그램 중간에 5초간의 머무르게 할 것이다. 체크가 되어 있지 않다면 1시간 23분 20초동안 머무르게 한다.

Set charge pump to 5kHz for laser standby level : 충전 펌프 출력이 레이저와 호환성을 갖기 위해 표준 12.5kHz가 아니라 5kHz 를 출력 하게 된다.

Use safe\_Z 선택 되어 있다면 Mach3는 정의된 safe z 위치를 사용 할 것이다. 주의: 만약 장비의 초기 운영시 기준점을 잡지 않는다면 임의의 좌표를 사용 하는 것 처럼 이 옵션을 선택하지 않는 편이 낫다.

Tool selections Persistent, 선택 됐다면, Mach3 프로그램을 종료 할때 선택 된 툴을 기억하여 저장 한다.

## 5.7 프로파일 정보의 저장 방법

Mach3.exe 프로그램을 실행 시키면 어떤 프로파일을 진행 시킬 것인지 물어 볼 것이다. 이것은 일반적으로 Mach3 폴더에 있고 .XML 확장자로 되어 있다. 때문에 인터넷 익스플로러를 통해 프로파일 파일을 열어 볼 수도 있고 프린트 할 수도 있다.

단축 아이콘은 시스템 설치자에 의해 설치 되고 Mach3.exe는 어떤 프로파일을 실행 할 것인지 물어 본다. 각기 다른 프로파일을 다른 단축아이콘으로 만들 수 있어 한 컴퓨터로 여러 장비 툴을 제어 할 수 있다.

이것은 여러 대의 장비를 사용 할 때 모터의 튜닝 값이 장비마다 다르게 요구 되는 경우, 혹은 서로 다른 리미트, 흡스위치를 설정 할 때 상당히 유용하게 쓰일 수 있다.

Mach23 실행시켜 가능한 프로파일을 선택 하던지 아니면 사용할 특정 프로파일의 단축 아이콘을 만들어 사용할 수 있다..

단축아이콘에서 로딩할 프로파일을 오른쪽 마우스 버튼으로 클릭하여 속성을 열면 창의 목표 란에서 “ /p” 독립변수로 주어진 것이 보인다. 참고로, Mach3Mill 단축 아이콘을 오른 마우스 버튼을 눌러 속성을 살펴 보라.

.XML 파일은 외부 편집 프로그램으로 편집 될 수 있지만 하나 하나의 의미를 정확하게 알고 있지 않는 이상 편집 하지 않기를 강력히 권고 하는 바이다. 알아 둘 것은 일부 화면 배치 같은 버튼들은 기존 정해진 값들이 Mach3 메뉴를 사용하여 덮어 씌어질 경우에만 만들어 질 수 있다. Mach3의 설정메뉴를 사용 하여 XML 프로파일을 업데이트 하는 것이 훨씬 안전하다.

## 6. Mach3 컨트롤과 파트 프로그램 운영 방법

이번 장은 Mach3 가 제공하는 화면 제어를 통해, 장비의 작업 진행과 설정에 관해 알려준다. 이것은 주로 Mach3 상에서 파트 프로그램을 진행 하려 하는 장비 운영자와 파트 프로그래머에게 깊은 관련이 있다.

### 6.1 소개

이번 장을 통해 더 상세한 것을 배울 수 있을 것이다. 6.2를 건너뛰어, 화면 제어에 관한 자세한 내용을 보기 전에 파트프로그램의 입력과 편집에 관한 것들을 우선 보고 싶을 수도 있을 것이다.

### 6.2 제어에 관한 설명.

처음 보는 순간 옵션에 관한 선택 폭과 Mach3가 보여주는 데이터에 의해 어렵다는 부담을 느낄 수도 있지만, 사실 이것은 몇몇 논리적 그룹으로 구성 됐을 뿐이다.

우리는 이것을 앞으로 컨트롤 패밀리(control families)라고 부를 것이다. “ 컨트롤(control :제어)이라는 단어는 통해, Mach3가 운영하는 버튼과, 관련키보드 단축키 그리고 DRO, LED나 라벨 등에 의해 표시되는 정보에 대한 전반적인 것들을 다룰 것이다.

컨트롤 패밀리(control family)의 구성 요소의 참고를 위해 정의해 보았다. 패밀리는 ,대부분의 사용자들에게 있어, 중요한 순서대로 설명된다.

주의 해야 할 것은 Mach3 의 실제 화면은 그 패밀리가 사용중에 모든 컨트롤을 보이지는 않는 다는 것이다. 이것은 특정화면을 한눈에 쉽게 볼 수 있게 도와 주거나 생산 중인 파트에 실수로 변화를 가져다 주는 것을 피할 수 있도록 해 준다.

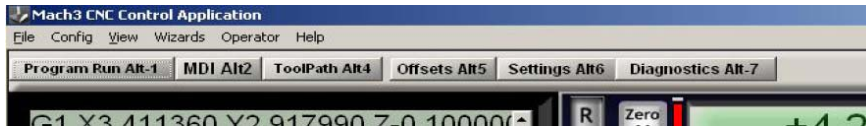


그림 6.1 각각의 스크린 탭 패밀리

Screen Designer 는 설정된 화면들 속의 컨트롤 항목 들을 추가/제거 할 수 있도록 해준다. 이것을 통하여 필요에 따라 화면을 처음 부터 디자인 하거나 편집 하여 어떤 컨트롤이든 특정 화면에 추가 할 수있다. 자세한 내용은 Mach3 Customization 메뉴얼을 참고 하기 바란다.

#### 6.2.1 화면 바꾸기

각 탭의 화면 마다 제어 버튼이 있다. 다른 화면으로 바꿀 수도 있고 현재 시스템의 상황에 관한 정보를 보여 준다.

##### 6.2.1.1 리셋

시스템이 리셋 되면 LED 가 일정한 색을 보인다. Charge pump pulse monitor 가 사용 가능으로 설정 되어 있다면, 펄스를 출력 할 것이면 선택된 “ enable output 이 활성화 된다.

### 6.2.1.2. 라벨

라벨은 마지막 에러 메시지, 현재 모드, 현재 진행중인 파트 프로그램 파일이름 그리고 사용되고 있는 프로파일을 나타내 준다.

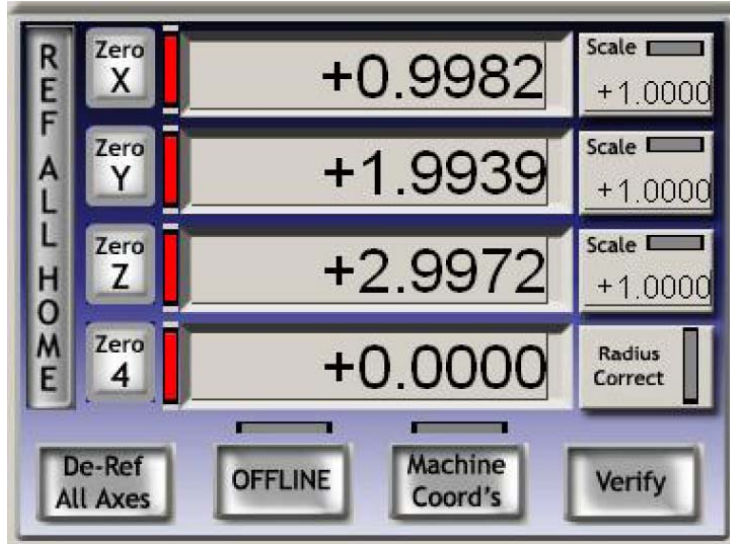


그림 6.2 축 제어 패밀리

### 6.2.1.3 화면 선택 버튼

이 버튼들은 이 화면에서 저 화면으로 바꿀 수 있다. 키보드 단축키는 각각의 이름 다음에 온다. 모든경우에 대한 명확성을 위해서 대문자를 사용하지만 단축키를 누르기 위해서 Shift 키를 함께 사용 해서는 안 된다.

### 6.2.2 축 제어 패밀리

이 패밀리는 툴의 현재 위치를 제어 한다.

#### 6.2.2.1 좌표 값 DRO

Config > Logic 대화창에서 단위 설정이 잠겨 있지 않는 이상 현재 단위(G20/G21)로 보여 준다. 여기서 표시되는 좌표 값은 현재 좌표계의 제어되는 지점을 표시한다. 일반적으로 현재 적용된 G92 오프셋과 작업 오프셋(Work Offset)(예:G54)의 조합으로 된 좌표계다. 하지만 이것은 절대좌표계로 변환 될 수있다.

축 DRO 창에 새로운 값을 입력 할 수 있다. 이로써, 현재의 작업 오프셋(Work Offset) 을 수정하여 설정한 값이 현재 좌표계가 되어 제어된 지점이 되도록 한다. 다중 좌표계에 완전히 익숙해 질 때까지 Work Offset(작업 오프셋)을 Offset screen(오프셋 화면) 에서 설정하라고 알려 주었다.

#### 6.2.2.2 정해진 기준점.

축의 기준점이 설정 됐다면 LED 가 녹색이 된다. 각 축은 “ Ref All” 버튼으로 기준 설정을 할 수 있다. 각 각의 축은 Diagnostics 화면에서 설정할 수 있다.

- 만약 홈/기준점(reference) 스위치가 축에 정의되어 있지 않다면 축은 움직이지 않을 것이지만 만약 Auto Zero DRO when homed 가 Config > Referencing(Homing) 에서 체크 되어 있다면 현재 축 위치의 장비 절대

좌표는 Config > State 대화창에 있는 Home/Reference 스위치 위치 테이블 에 정의 내려진 값이 될 것이다. 일반적으로 값은 0(영) 이 된다.

- 축에 home/reference 스위치가 정의내려져 있고 Ref 를 눌렀을 때 실제 입력이 없다면 축은 Config > Referencing(Homing) 에서 정의된 방향으로 입력이 작동 될때까지 움직일 것이다. 그리고 약간 반대 방향으로 다시 움직여서 입력이 비활성화 된다. 그리고 입력이 이미 작동 되었다면 단지 뒤로 조금 움직여 작동 되지 않은 상태로 위치할 것이다.

De-Ref All 버튼은 축을 움직이지 않지만 0 점 상태로 정지 시킨다.

### 6.2.2.3 장비의 좌표

Machine 버튼은 장비의 절대 좌표를 보여 준다. LED 는 절대 좌표가 화면에 나타나 있다는 것을 알려 준다.

### 6.2.2.4 눈금자(Scale)

어떤 축의 눈금 인자이든 G51 로 설정되고 G50 이 기본값이 된다. 1.0 을 제외한 다른 눈금 인자(scale factor) 가 설정 되면 G 코드로 나타날 때 좌표계에 적용 된다. (예 : X 문자 Y 문자 등등) 이 눈금 LED 는 축에 눈금이 설정 됐다는 것을 알리기 위해 깜박 인다. G51 에 의해 정의된 값은 눈금 DRO 에서 설정 될 수 있다. 마이너스 값은 연관된 축의 좌표에 그대로 반영된다. G50 버튼은 G50 명령을 실행하여 모든 눈금을 1.0 으로 만든다.

### 6.2.2.5 소프트리미트(soft limit)

Softlimit 버튼은 Config>home/limit 에 정의한 값을 활성화 시킨다.

### 6.2.2.6 조회 하기

Verify 버튼은 홈 스위치가 있을 때에만 적용이 되며 장비의 작동중 탈조된 스텝을 조회하기 위해 각 홈 스위치로 움직이게 한다.

### 6.2.2.7 반경/직경 수정

회전축(rotary axes)은 회전직경(Rotational Diameter) 콘트를 패밀리를 사용하여 정의된 대략적인 작업대상물 크기를 운영할 수 있다. 이 크기는 회전 축을 포함한 좌표계 운동 상에서, 복합적인 피드비율(feedrate) 을 계산 할 때 쓰인다. LED 는 값이 0 이 아니라는 것을 알려준다.

## 6.2.3 “ Move to ” 제어

화면에는 여러가지의 버튼이 있는데 돌을 특정 지점으로 쉽게 움직 일 수있도록 설계 되어 있다. 이런 버튼들은 Goto Zs 와같이 모든 축을 영점으로 움직이게 하거나, Goto Tool Change, Goto Safe Z, Goto Home 등으로 나타내 진다. 추가로 Mach3 는 다른 두 가지로 설정된 좌표계를 기억 하고 필요에 따라 선택 할 수있다. 이것은 Set Reference Point 와 Goto Ref Point 로 콘트를 되거나 Set Variable Position 과 Goto Variable Position 으로 제어 된다.

### 6.2.4 MDI 터치 컨트롤 패밀리(MDI teach control family)

G 코드라인을 MDI(Manual Data Input) 창에 입력하여 즉각적인 실행을 위해 사용 할 수 있다. 이것은 MDI 핫키를 누르거나 라인을 클릭한 후 사용 할 수 있다.

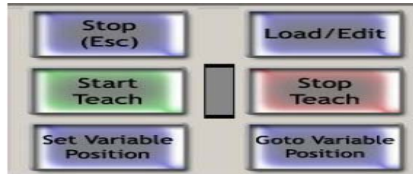


그림 6.4 제어점 기억과 터치



그림 6.5 MID 라인

MDI 창이 활성화 되면 색이 바뀌고 위에 임시 창이 떠서 과거에 입력했던 명령어들을 보여 준다. 그림 6.5 를 통해 예제를 볼 수 있다. 위, 아래 화살표 키를 통해 순서대로 내려진 명령어들을 다시 선택 하여 사용 할 수 있게 한다. Enter 키는 이때 넣은 입력어를 Mach3 가 실행 하게 하며 다음 명령어를 입력 할 때 까지 MDI 창은 계속 활성화가 될 것이다. Esc 키는 입력한 명령어를 취소 시키고 MDI 창에서 빠져 나오게 한다. 기억해 두어야 할 것은 이때 선택된 명령어는 어떤 키보드 형태이던 Mach3 를 제어한다기 보다는 MDI 라인에 입력되는 것이다. 특히, 키로 조깅 하는 것은 입력되어 지지 않는다.

Mach3 는 MDI 창에 입력되어 실행된 모든 명령어를 티치(teach) 기능을 사용 하는 파일에 저장한다. Start Teach 를 클릭한다. 그리고 필요한 명령어를 입력하고 Stop Teach 를 클릭한다. LED 가 깜빡 일때는 현재 티치 모드에 있다는 것을 알려준다. 입력된 명령어들은 다음과 같은 양식의 파일이름으로 저장 된다.  
 “ C:/Mach3/Gcode/MDITeach.tap” Load/Edit 을 클릭하면 Mach3 에 이 파일을 로딩하여 실행시키거나 혹은 편집 할 수 있게 한다. - 이때 Program Run Screen 으로 가서 확인 할 수 있다. 만약 이렇게 사용한 명령어들을 계속 저장 하여 사용 하고 싶다면 파일을 편집기로 열어 “ Save As” 로 저장하여 보관한다.

### 6.2.5 조깅 컨트롤 패밀리

조깅 컨트롤은 Tab 키를 누를 때 마다 나타나는 임시 창으로 보여 진다. 그림 6.6 을 참고 한다.

Jog on/off 버튼이 현재 화면에 나타날 때는 언제나 장비의 축은 다음과 같은 항목을 사용하여 조그 된다. a)조그 핫키. - 이때 키보드 에뮬레이터를 통해 연결된 MPG 도 포함된다. 핫키는 Configuration Axis 핫키에서 정의 된다. b)병렬포트의 인코더에 연결된 MPG 핸드휠.



그림 6.6 조그 제어 패밀리

c)조이스틱과 같은 USB 인터페이스. d)시뮬레이션 트랙볼 e)기존의 Windows 호환 아날로그 조이스틱

만약 Jog On/Off 버튼 이 화면에 없거나 Off 로 나타나 있다면 안전을 이유로 조깅은 사용 할 수 없게 된다.

#### 6.2.5.1 핫키 조그하기

모드에는 Continuous, Step 그리고 MPG, 이렇게 3 가지 모드가 있는데 이것들은 Jog Mode 버튼으로 선택 할 수 있고 LED 로 나타 난다.

연속(Continuous) 모드는 핫키가 눌러져 있는 동안 설정된 느린 속도로 축(들)을 움직인다. 조그 속도는 아래와 같이 정의 되지만 “ shift” + 핫키를 누르면서 바꿀 수 있다. Cont. LED 옆의 LED 는 현재 최대 조깅 속도로 진행됨을 알려준다.

스텝(Step) 모드는 축을 Jog Increment DRO 에 의해 정의된 대로 한번 누를때 마다 한 스텝씩 움직인다. F 문자에 의해 정의된 현재의 피드비율(current feedrate)이 축을 움직인다. 증분율을 Step DRO 창에 값을 직접 넣거나 Cycle Jog Step 버튼을 사용하여 사용자 정의가 된 값을 선택하면서 설정 가능 하다.

Incremental(증분) 모드는 토글(toggle) 버튼으로 선택하거나, 연속모드에서 조그를 하기전 Ctrl 키를 누르면서 할 수있다.

### 6.2.5.2 병렬포트 혹은 Modbus MPG 조깅

Encoder 세개가 설정됐다면 MPG Jog Mode 를 선택 하여 조그 휠로 사용할 수 있다. MPG 가 조그할 축은 LED 로 나타나며 Alt-A 버튼을 누르면 MPG1, Alt-B 는 MPG2, Alt-C 는 MPG3 등 장착된 축이 바뀌지며 시행 된다.

축 움직임에 있어 MPG Velocity 모드는 Mach3 와 함께 MPG 의 회전 속도와 관계가 있는데 이때 최고 속도와 축의 가속도의 요구에 응하는 것을 확인 하게 된다. 이렇게 함으로써 상호간의 교류에 대응하여 움직인다는 느낌을 줄 것이다. MPG Step/Velocity 모드는 현재 Velocity 모드 처럼 작동한다.

Single Step 모드에서 MPG 인코더로부터 한번씩 클릭 할 때 마다 설정된 거리만큼 증분하는 조그 스텝을 하게 한다. 한번에 한스텝 씩 움직이게 할 수 있다. 바꿔말하면, 이미 축이 움직이고 있는 중이라면 클릭 하여도 명령은 무시된다. Multi Step 모드에서는 클릭한 수만큼 쌓이고 기억 되어 그만큼 움직이게 된다. 클릭을 계속 하여 움직이는 모습을 보여 클릭을 멈추어도 클릭 수가 기억된 만큼 축이 움직일 것이다. 스텝은 MPG Feedrate DRO 창에 주어진 피드비율만큼 총축 될 것이다.

이런 스텝 모드들은 장비를 설정 할 때 아주 미세한 제어를 위해 주로 쓰인다. 일반적으로 Velocity Mode 를 사용 하길 권장 한다.

### 6.2.5.3 스핀들 스피드 컨트롤 패밀리

장비를 어떻게 디자인하는가에 따라 다르지만 스핀들 제어에 관해서는 3 가지 방법이 있다. a) 고정된 속도로 혹은 손으로 조절하거나 on/off 하면서 쓴다. b)고정된 속도로 혹은 손으로 속도 조절 한다. 하지만 스위치의 on/off 는 외부의 출력을 경유하여 M 코드에 의해 제어 된다. c) 속도가 PWM 혹은 스텝/방향 드라이브를 사용 하는 Mach3 에 의해서 설정된다.

이번 제어 패밀리는 오직 c 의 경우에만 해당된다.

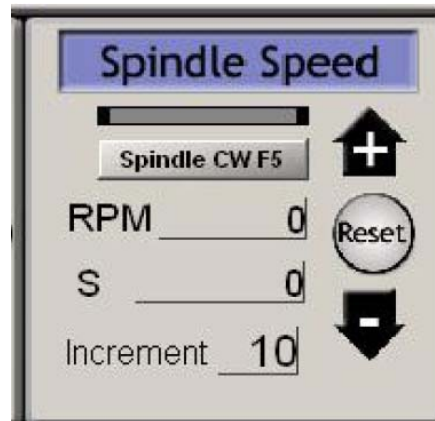


그림 6.6 스핀들 속도 컨트롤 패밀리

S DRO 는 S 문자가 파트 프로그램에서 사용 될 때 자체 값을 갖게 된다. 이것은 적절한 스핀들 속도가 된다. 그리고 DRO 창에 직접 입력 함으로써도 설정이 된다.

Mach3 는 어떤 경우에도 Config>Port&Pins > Spindle Setup 탭에 있는 Max Speed 값 보다 크거나 Min Speed 값보다 작은 속도로 셋팅 하는 것을 허락 하지 않는다.

만약 Index 입력 값이 설정되고 센서가 스핀들이 회전에 대한 정보가 해당 핀에 전달 된다면 현재 스핀들의 속도는 RPM DRO 창에 나타나게 된다. RPM DRO 창은 여러분에 의해 셋팅 될 수 없다. - S DRO 를 사용하여 속도에 대한 명령을 한다.

## 6.2.6 피드 제어 패밀리

### 6.2.6.1 피드 속도

Prog Feed DRO는 현재 단위(분(minute) 당 inch/millimeter)로 피드비율(feed rate)이 나타난다. 이것은 파트 프로그램에서 F 문자로 설정되거나 F DRO 에서 타이핑 할 수 있다. Mach3 는 이렇게 설정된 속도를 좌표상의 움직임에 적용 하려 할 것이다. 만약 이 속도가 나올 수 없는 속도라면 최대한 가능한 속도로 움직일 것이다.



### 6.2.6.2 피드 회전 속도

커터가 팁(tip) 대비 가공한 절삭에 의해 지정됨에 따라 아마도 회전당 피드를 지정해 주는 것이 더 편할 수도 있다.(즉 팁 당 피드 X 톨의 팁 개수) Prog feed DRO 창은 스핀들 회전수당 현재단위(inch/millimeters) 로 피드 비율을 나타낸다. 이것은 파트 프로그램에서 F 문자에 의해 셋팅 되거나 DRO 창에 직접 입력 한다. 스핀들의 회전은 S DRO나 혹은 카운트 되는 인덱스 펄스에 의해 측정된 속도로 결정한다. Config > Logic에 있는 체크박스가 있고 Mach3 가 어떤 것을 적용 할지 선택 할 수 있게 한다.

Feed unit/rev를 사용하기 위해, Mach3는 반드시 스핀들의 선택된 속도 값을 알아야 한다.(즉, 이것은 a). 스핀들 스피드 킷를 패밀리의 S DRO에 입력된 데이터에 의해서나 혹은 S 문자에 의해 정의되어 있거나, b). 실제 스핀들 속도를 측정하기 위해 Index 가 연결 되어 있어야 한다.)

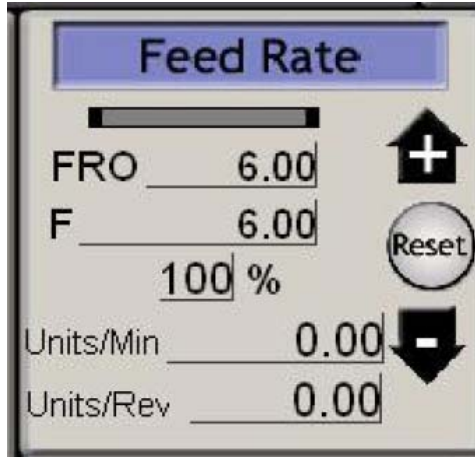


그림 6.7 피드 제어 패밀리

알아 두어야 할 것은 스핀들 속도가 1 rpm이 아닌 이상 컨트롤에 나타난 숫자 값은 매우 다를 것이다. 그러나 feed per minute 를 feed per rev 모드로 쓰게 된다면 엄청난 사고가 날 수도 있다.

### 6.2.6.3 피드 화면 표시(Feed display)

운영상의 실제 피드는 Unit/min, Unit/rev 의 화면에 모든 축의 좌표 움직임이 나타나 도록 한다. 만약 스핀들 속도가 설정되어 있지 않고 실제 스핀들 속도가 측정되어 있지 않다면 Feed per rev 은 의미가 없게 된다.

### 6.2.6.4 피드 번복하기, 덮어 쓰기(Feed override).

M49(feed override 사용 안함) 설정으로 하지 않는 이상, 피드 비율(feed rate)은 수동으로 번복 할 수 있고 20%에서 299%까지의 범위를 갖아 퍼센트 DRO에 입력 할 수 있다. 이 값은 10%의 스텝 안쪽으로 버튼이나 키보드 단축버튼에 의해 슬쩍 움직일 수도 있고 100%로 리셋 될 수도 있다. LED가 작업 중에 덮어쓰기 하는 것에 대해 경고메시지를 보낸다. FRO DRO 는 덮어쓰기한 퍼센트가 적용한 계산 결과를 set override 에 표시한다.

## 6.2.7 프로그램 동작 제어 패밀리

이 컨트롤은 로딩된 파트 프로그램을 실행 하거나 MDI 라인에서 명령을 한다.

### 6.2.7.1 작업시작 - Cycle Start

**안전 주의 :** 주의 해야 할것은 Cycle Start 버튼은 일반적으로 스핀들회전과 축을 작동 시킨다. 항상 두손으로 작동 되도록 설정되어야 하며 만약 핫키로 조작 하려 한다면 키보드를 동시에 2 개 눌러서 작동 하게 한다.

### 6.2.7.2 작업 일시 중지 - feed hold

Feedhold 버튼을 누르면 가능한 한 빨리 진행하고 있는 작업을 중단시키지만 제어가 가능하여 cycle start 버튼을 다시 누르면 멈추고 있던 작업이 계속 다시 진행 한다. 스핀들이나 쿨런트 는 이때 수동으로 조작 가능하다.

Feedhold 일 때는 축을 조그하여 움직일 수 있고, 툴이 부러졌다면 다시 교체 할 수도 있다. 만약 스펀들이나 쿨런트(냉매제)를 off 했다면 다시계속 진행 하기 전에 on 상태로 놓는다. Mach3 는 축의 위치를 기억 하기 때문에 파트 프로그램을 진행 하기 전에 feedhold 버튼을 누른 시점으로 다시 돌아 간다.

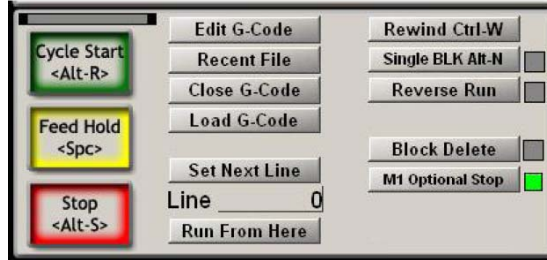


그림 6.8 프로그램 작동 패밀리

### 6.2.7.3 작업 중단 - Stop

Stop 은 최대한 빨리 동작을 멈춘다. Feed Hold 와는 달리 정지 하는 순간 스텝을 잃게 되고 다시 그 자리에서부터 시작 할 수 없다.

### 6.2.7.4 되돌리기- Rewind

현재 로딩된 파트 프로그램을 다시 돌린다.

### 6.2.7.5 싱글 블록 - Single BLK

SingBLK 는 LED 가 있는 톨 버튼이다. Single Block 모드에서는 Cycle Start 를 실행시키면 파트 프로그램의 다음 한줄을 실행 시키고 FeedHold 가 되게 한다.

### 6.2.7.6 뒤로 가기 - Reverse Run

Reverse Run 은 LED 가 있는 톨 버튼이다. 이것은 반드시 Feed Hold 나 Single BLK 후에 사용 되어 져야 하고 그후 Cycle Start 를 누르면 파트 프로그램을 거꾸로 진행 시키게 된다. 이것은 엔드밀이 부러졌을 때나 혹은 플라즈마 커터의 아크가 갑자기 나오지 않을 때 유용하게 사용 할 수 있다.

### 6.2.7.7 라인 번호 - Line Number

Line DRO 는 G 코드 화면 창에 나타나는 현재 라인의 서수 다(0 부터 시작). 알아 두어야 할 것은 N 문자 라인 숫자와 연관이 없다는 것이다. 현재 라인을 설정 하기 위해서 DRO 창에 타이핑 해서 넣을 수 있다.

### 6.2.7.8 여기서 시작 - Run from here

Run from here 는 파트 프로그램의 연습가공을 실행하여 형상의 상태(G20/G21, G90/G91 등등)를 알기위해 그리고 Line Number 의 라인 시작 을 위해 축을 움직여 정확한 위치 지점으로 옮긴다. 이 Run from here 는 서브루틴이 돌아 가는 중에 사용 해서는 안 된다.

### 6.2.7.9 다음 라인 설정 Set next line

Run from here 와 같지만 사전 준비 모드 설정이나 모션 없이 진행 된다.

### 6.2.7.10 블록 삭제 Block Delete

Delete 버튼은 Block Delete 스위치를 토글(toggle) 한다. 활성화 해 놓으면 /(슬레시) 로 시작 하는 G 코드 라인은 실행 되지 않는다.

### 6.2.7.11 Optional Stop

End 버튼은 Optional Stop 스위치를 토글 한다. 활성화 해 놓으면 M01 명령어는 M00 으로 간주 할 것이다.

### 6.2.8 파일 제어 패밀리

그림 6.9 와 같은 제어는 파트 프로그램의 파일에 관계되어 있다. 운영상에 분명하게 나타 난다.

### 6.2.9 툴 세부사항

그림 6.9 와 같이 툴 세부사항(Tool Detail) 그룹은 현재 툴에 대해 보여주고 길이 오프셋(offset) 과 직경 그리고 디지털 입력이 가능한 시스템에서 Z 평면에 자동으로 영점을 생성하도록 허가한다.

config>logic에서 툴 교체 요구가 생략 되지 않으면, M6에 도달 하였을 때, Mach3는 Safe Z 로 움직인 후 멈추고 Tool Change LED 가 깜빡인다. 그러면 이때 툴을 교체한 후에 Cycle Start 로 계속 진행 한다.



그림 6.9 툴 세부 사항

### 6.2.10 G 코드와 툴 패스 제어 패밀리

현재 로딩된 파트 프로그램은 G 코드 창에 나타난다. 현재 라인은 하이라이트 처리되고 창에서 스크롤 바를 움직여 움직일 수 있다.

툴패스 화면은, 그림 6.10 에서 보여 주는 것처럼, 제어된 점들의 패스를 X Y Z 면 상에서 움직일 것이다. 파트 프로그램이 패스(path)를 실행 할때 녹색으로 변해가며 움직이는 과정을 보여 준다. 이렇게 변하는 과정은 다이내믹 하여 화면을 움직일때 남아 있지 않고 툴패스의 뷰도 바뀌게 된다.

종종 화면에 나타나는 것이 계획된 패스로 정확히 따라 가지 않는 것을 볼 것이다. 그 이유는 다음과 같다. Mach3는 현재 하고 있는 일에 일단 중점을 둔다. 정확한 스텝 펄스를 장비 툴에 보내는 것이 첫째 임무다. 툴패스를 그리는 것은 우선순위에 떨어진다. Mach3의 툴패스 화면에 나타내는 그림은 시간적 여유가 있을 때 표시해 주며 각각의 점들을 직선으로 나타내는 것이다. 그러므로 계산 시간이 충분치 않다면 단지 몇 개의 점만 그려질 것이며 때문에 실제 원형이 때로는 다각형으로 보여 질 수도 있을 것이다. 이것은 걱정할 문제가 아니다.

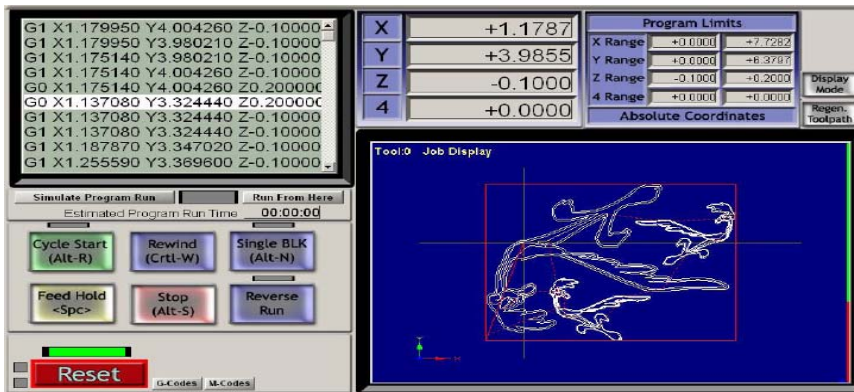


그림 6.10 툴 패스(tool path) 패밀리

Simulate program run 버튼은 G 코드를 실행 시키는 데, 이때 툴의 움직임은 없으며 파트를 만드는 시간을 알 수 있게 해 준다.

Program limits 데이터는 제어되는 각 점선의 최 외곽이 이탈 하지 않도록 체크 한다. (즉 테이블 상단을 밀링 하지 못하도록 한다.)

스크린 캡처 화면은 또한 DRO 와 Program Run 일부를 보여 준다.

여러분이 장비 테이블 사이즈에 맞는 소프트리밋을 설정 했다면 작업과 테이블 모드 사이를 토글(선택) 하여 툴 패스와 테이블의 관계를 보여 주는 Display Mode 버튼을 이용하는 것이 유용하다.

툴패스 화면은 왼쪽 마우스 버튼을 클릭하여 회전시킬 수 있으며 마우스를 그 안에서 끌기 할 수 있다. Shift + 왼 마우스 버튼을 하면 확대 축소를 하고 오른쪽 마우스 버튼을 사용 하여 화면을 움직일 수 있다.

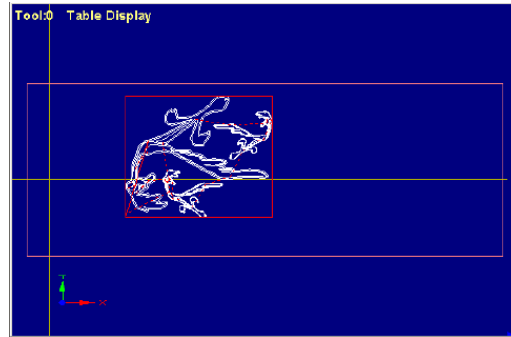


그림 6.11 - 테이블과 관련한 툴패스

Regenerate 버튼은 고정체에 적용된 G92 오프셋값이 적용된 G 코드로부터 툴패스 화면을 재 생성한다.

### 6.2.11 작업 오프셋(workoffset)과 툴 테이블(tool table) 제어 패밀리.

작업 오프셋(Work Offset) 과 툴테이블(Tool table)은 파트 프로그램과 오퍼레이터 메뉴로부터 액세스 할 수 있지만 이 패밀리를 통해 조작하는 것이 더 편할 수 있다. 7장 에서 자세히 살펴 보기 바란다.

G 코드의 정의 때문에 작업오프셋 이나 툴 테이블이 약간 다른 방식으로 작동 되는 것을 볼 수 있다.

경고: 작업 오프셋이나 툴 오프셋을 사용 중에 바꾼다면, DRO 창에 바뀌어진 값이 나오지만, 장비는 그렇게 움직이지 않는다. 그러나 새로운 오프셋을 설정한 후 G0, G1 등의 변화를 주면 새로운 좌표로 이동할 것이다. 여러분은 현재 어떤 작업을 하고 있느냐를 반드시 인식하고 있어야만 장비의 충돌을 피할 수 있다.

#### 6.2.11.1 작업 오프셋(Work Offsets)

Mach3 는 기본 적으로 작업 오프셋(work offset) 1번을 사용한다. 1~255까지, 임의의 숫자를 Current Work offset DRO 창에 입력하여 현재 사용할 작업 오프셋을 설정 할 수 있다. 작업 오프셋을 또 다른 말로 고정체 오프셋(fixture offset) 이라고 부르기도 한다.

DRO 창에 값을 입력 하는 것은 파트프로그램에서 G55 에서 G59까지, 혹은 G58.1에서 G59.253 까지 제거 하는 것과 동일 하다.

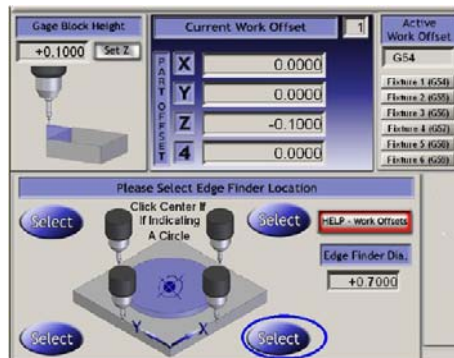


그림 6.12 작업 오프셋 패밀리

현재의 오프셋 시스템은 Fixture 버튼을 사용하여 셋팅 할 수도 있다.

오프셋 값은 바꿀 수 있어서 관련 Part Offset DRO 창에 넣으면 된다.

DRO 창의 값은 원하는 위치로 축을 움직이고 as Set or Select button 을 클릭하여 설정 할 수도 있다. X, Y와 Z축은 약간 달리 셋팅 한다. Z는 은 이해 하기 쉽게 때문에 먼저 설명 한다.

Z 오프셋 은 스피들에서 마스터툴로 셋업 한다(하나의 기준 툴로 설정한다). 그리고 나머지 툴 들은 작업테이블로 정정 한다.게이지 블록, 알루미늄 호일, 혹은 종이를 툴(여기서는 엔드밀) 과 절삭 물에 살짝 끼우고 Z를 살짝 터치하여 삽입 한 것이 빠져 나오지 못하면 Z=0 로 간주 된다. 이때 삽입한 게이지, 호일 등등의 두께를 Gage Block Height DRO 창에 입력한다 그리고 Set Z 버튼을 클릭 하면 된다. 이렇게 현재 작업 오프셋의 Z값을 설정하여 축이 주어진 높이에 있게 할 수 있다.

X, Y 도 비슷하지만 살짝 터치 하는 것이 4면중 어느 면도 될 수 있고 툴의 직경과 게이지의 두께가 계산에 들어 간다. 예를 들어, 직경이 0.5" 인 툴과 0.1" 의 게이지블록을 사용하여 절삭물의 모서리에서 y=0로 설정하려 한다. 그러면 Edge Finder Dia DRO 에는 0.7" 를 입력하고(툴 직경 과 게이지 블록 X 2) 클릭 한다.

Config>State 에 저장된 불변하는 오프셋(persitent offset) 과 새로운 오프셋값 들은 Mach3가 항상 기억하게 되어 다시 실행 하여도 그 값을 유지 한다.

### 6.2.11.2 툴(tools)

툴 번호는 0-255 까지 숫자를 사용 할 수 있다. 파트 프로그램에서 T 문자로 툴 번호는 선택되거나 T DRO 에 번호를 넣을 수 있다. 이것의 오프셋들은 Offset On/Off 토글 버튼에에서 On으로 설정되어 있을 때만 적용이 된다. 파트 프로그램의 G43와 G49 도 대등한 역할을 한다.

Mach3Mill 에서 Z offset 과 Diameter 만이 툴에 적용되어 사용된다. Diameter(직경)값은 DRO 창에 입력 가능 하고 Z offset(즉 툴길이에 대한 보정)은 직접 입력 하거나 터칭에 의해 입력 된다. Set Tool Offset 기능은 Z를 작업오프셋(work offset) 으로 설정하는 것과

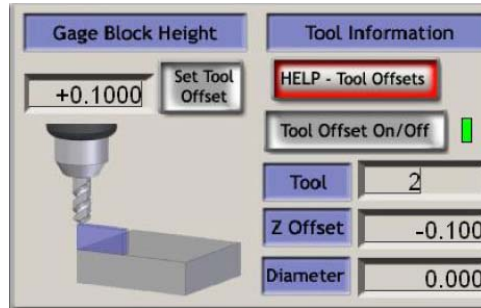


그림 6.13 툴 오프셋

일치 한다.

툴 오프셋 데이터는 작업 오프셋 데이터와 동일 한 방법으로 작업들 사이사이에 적용된다.

### 6.2.11.3 오프셋 테이블 직접 제어

Save Work Offset 과 Save Tool Offset 버튼을 이용해서

직접 열고 편집 할 수도 있고 혹은 Operator > Fixtures 와 Operator > tooltable 메뉴에서 테이블을 열고 편집 할 수 도 있다.

### 6.2.12 회전 직경 제어 패밀리

피드비를 제어 패밀리에서 설명한 것과 같이 회전하는 작업물의 대략 적인 크기를 정의 할 수 있는 것이 가능하기 때문에 회전 축 속도도 복합적인 피드비율에서 정확하게 표현

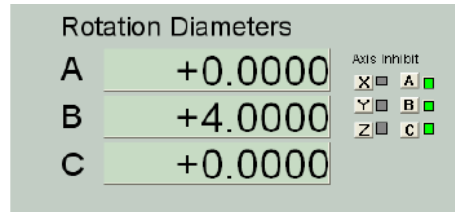


그림 6.14 회전 직경

될 수 있다.

관련된 직경은 이 패밀리의 DRO에 입력 되어진다. 축 제어 패밀리는 0 이 아닌 값이 입력 됐음을 알리는 LED로 경고 한다.

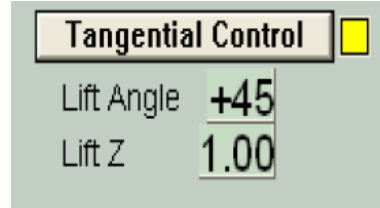
회전 운동이 선형 축들과 조합을 이루는 것이 아니라면 값은 필요하지 않다. 이런 경우 분당 몇도 움직이나 혹은 회전당 각도가 적합한 F 문자로 프로그램 되어야 한다.

**6.2.13 접선(tangential) 제어 패밀리**

비닐이나 섬유재질을 커팅 할 때 로터리(rotary) 축을 사용하여 칼날의 방향을 컨트롤 하는 것은 상당히 유용하다. X, Y 축이 움직이는 방향의 근 접선으로 커팅 하는 것이 가장 좋은 방법이다.

Mach3 는 A 축을 이와 같이 제어하여 G1 움직임을 표현할 것이다.

이기능은 Tangential Control 버튼이 활성화 되어야 사용 할 수 있다.



대부분의 경우 대상물에서 코너를 자를때 칼날의 회전 각은 제한되어 있다. **그림 6.15 접선 제어 패밀리**

이 값은 Lift Angle 에 정의 되어 있다. 코너의 각도가 Lift Angle 보다 더 크다면 Lift Z 의 값에 의해 Z 축을 올리는 것에 문제가 생긴다. 커터는 회전 하고 그리고 Z 축은 다시 그 지점으로 내려가 새로운 방향으로 자를 것이다.

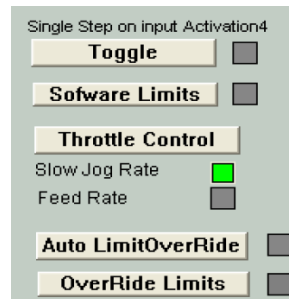
**6.2.14 리밋(Limit) 과 기타 제어 패밀리**

**6.2.14.1 입력 활성화(Input Activation) 4**

Input Activation 신호 4 는 Program Running 컨트롤 패밀리의 Single 버튼 과 동등한, 유선으로 연결된 Single Step 기능으로 사용 하기 위해 설정 할 수 있다.

**6.2.14.2 설정된 리밋 건너뛰기 (Override limits)**

Mach3 는 소프트웨어 적으로 입력 부에 연결된 리밋 스위치를 무시 할 수 있도록 할 수 있다. 이것은 말하자면, 리셋 후에 실행된 조깅은 축이 리밋 스위치에서 물러나기 전까지 리밋의 대상이 되지 않는다. 대체 리밋들이 패싱 될 수 있음에 따라 이것의 사용은 LED 로 경고를 한다.

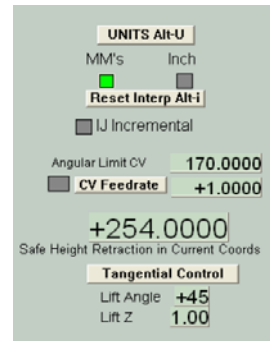


**그림 6.16 리밋 제어 패밀리**

알아두어야 할 것은 이런 컨트롤은 리밋 스위치가 드라이브 회로에 연결 되어 있거나 혹은 가동 중인 EStop 에 연결이 되어 있다면 적용 되지 않는다.

**6.2.15 시스템 셋팅 제어 패밀리(System Setting control family)**

주의 : 이 패밀리의 제어는 Mach3 의 스크린 화면 한곳에 모여 있지 않다. Program Run, Settings and Diagnostics screens 에서 찾아야 할 것이다.



**그림 6.17 시스템 셋팅**

### 6.2.15.1 단위

이 토글은 G20 와 G21 코드를 이행하여 현재의 축정 단위를 바꾼다. 파트 프로그램의 조그만 부분이 Work Offset 과 Tool Offset 테이블이 하나의 고정된 단위로 되어 있다는 사실 때문이 아니라면 사용 하지 않기를 강력히 권고 하는 바이다.

### 6.2.15.2 Z 축 제한(Safe Z)

이 패밀리는 작업물의 집게나 파트로부터 안전한 높이 값을 정의 내리게 해 준다. 원점 가거나 톨의 변환에 이용된다.

### 6.2.15.3 CV 모드/각도제한 CV Mode/Angular Limit

이 LED 는 Constant Velocity 모드에서 시스템이 진행 되고 있을 때 불이 들어 온다. 이것은 Exact stop 모드보다 더 부드럽고 빠른 오퍼레이션을 가능 하게 하지만 각이 진 코너에서 라운딩 현상의 원인이 될 수 있다. 시스템이 CV 모드에 있어도 방향 변화가 있는 코너에서, 마치 Exact Stop 이 선택 된 것처럼, Angular Limit DRO 에 주어진 값 보다 더 정밀하게 수행 할 것이다. 자세한 사항은 10 장 Constant Velocity 에 있다.

### 6.2.15.4 오프라인

이 토글과 경고 LED 는 Mach3 의 모든 출력 신호를 차단 한다. 이것은 장비의 설정과 테스트를 위함이다. 파트 프로그램을 진행 중인 상태에서 사용한다면 모든 좌표위치에 관한 문제가 생길 수 있다.

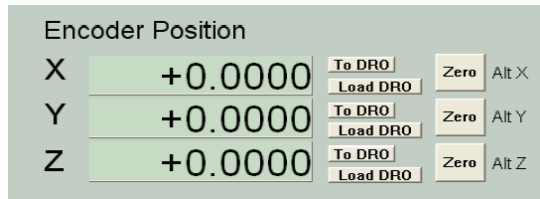


그림 6.18 - 인코더 제어 패밀리

### 6.2.16 인코더 제어 패밀리

이 패밀리는 축 인코더로 부터 값을 표시 해 주고 주 축의 DRO 들과 상호 신호 전달을 한다. Zero 버튼은 해당되는 인코더 DRO 를 0 으로 만든다.

To DRO 버튼은 주축 DRO 에 값을 복사한다. 즉 이 값을 G92 옵션에 적용 시킨다.

Load DRO 버튼은 대응하는 주축의 DRO 로부터 인코더 DRO 를 로딩 시킨다.

### 6.2.17 자동 Z 축 제어 패밀리

Mach3 는 Z 축의 최저 리밋을 설정 할 수 있게 한다. Config > Logic 대화 상자에 고정 설정 하는 Inhibit Z 값을 본다. G 코드 프로그램을 준비 하거나 실행 전에 Inhibit Z 값을 설정해 줄 수 있게 하는 컨트롤 패밀리도 있다. 그림 6.19 를 참고 한다.

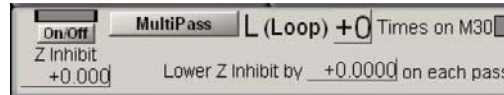


그림 6.19 오토매틱 Z 컨트롤

일반적으로 DXF 나 HPGL 형식의 파일의 프로그램을 코딩해서 한번의 커팅으로 만들거나 여러번 커팅해서 마침내 설계 했던 Z 높이가 되게 한다(예를 들면 작업물의 표면이 Z=0 일때 Z= -0.6 이라 놓는다). 마지막 명령어는 M30(퇴감기)가 되어야 한다.

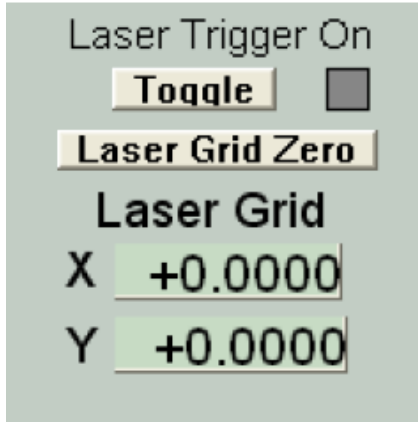
오토매틱 Z 컨트롤 을 사용하여 a)Z inhibit 값을 Z 의 처음 황삭 값으로 하여 설정 한다(대략 Z=-0.5 로 한다). b)lower Z inhibit 을 연속 커팅 깊이로 설정 한다(0.1 을 허용 한다). 전체 작업은 일곱 번의 가공으로 Z=-0.6 이 되니 c)이때 L(loop) 에 7 을 넣는다. Cycle Start 를 누르면 장비는 자동적으로 Z 깊이가 증가하면서 여러 번에 걸쳐 커팅을 하게 된다. 각 DRO 는 진행과정을 나타내 주고 L 은 Z inhibit 값이 업데이트 되어 감에 따라 감소

한다. 만약 주어진 L 값이 파트 프로그램의 Z r 값에 도달 하지 않으면 L DRO 를 업데이트 하여 다시 프로그램을 시작 한다.

### 6.2.18 레이저 점화 출력 패밀리(Laser Trigger output family)

Mach3 는 X 혹은 Y 축이 점화 시점( trigger point) 을 지날때 Digitize Trigger Out Pin 에 펄스를 출력 한다. Laser Trigger 그룹 컨트롤은 현재 단위로 그리드(grid)를 정의 내리고 임의의 기준점에 대응한다 . 원하는 그리드(grid) 원점과 제어 점이 일치 할 때 Laser Grid Zero 를 클릭 한다. X, Y 축의 그리드 선 위치를 정의 내리고 Toggle 을 클릭하여 축이 그리드 라인상에서 움직일 때 펄스의 출력을 가능 하게 한다.

이것은 경험적인 것이며 차후의 버전에서 바뀔 수도 있다.



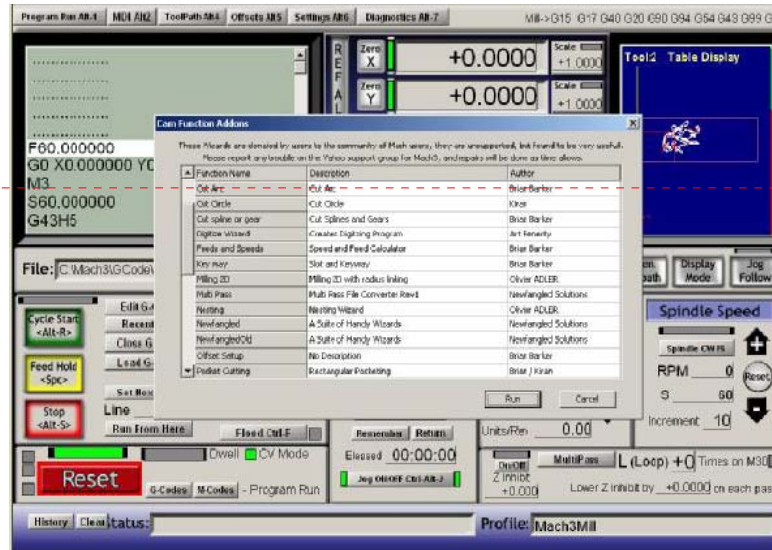
### 6.2.19 맞춤 사용자 제어 패밀리

Mach3 는 장비 설계자에게 전반적인 범위의 기능을 맞춤 스크린에 추가 할 수 있으며 이때 LED 나 DRO 버튼은 Visual Basic 스크립트 프로그램을 사용 한다. 자세한 사항은 Mach3 Customization manual 에서 볼 수 있다. 이런 예들은 Mach3 스크린이 각각 다른 응용프로그램에서, 궁극적으로 같은 기능 이라고 할 지라도 얼마나 다른지 보여 준다.



### 6.3 마법사(Wizard) 사용 하기

Mach3 마법사는 Teach 기능의 연장선 상으로, 다른 특별한 화면을 이용하여 머시닝 오퍼레이션을 할 수 있다. 마법사는 그리고 G 코드를 생성하여 필요한 절삭을 한다. 마법사의 예로는 원형 포켓 머시닝, 다중 구멍 드릴링 그리고 글자를 세기는 것 등등 많은 것들이 있다. 화면에서 Load Wizard 버튼을 누르거나 NFS 버튼을 누르면 시스템에 설치된 마법사 테이블을 보여준다. 그 중에 필요한 하나를 선택 하고 Run 을 클릭한다. 마법사 화면(혹은 여러개 중에 하나)이 나타난다.



**메모 [DP18]:** 바꿔말하면 이미 짜여진 템플릿 소프트웨어로 원 사각형 글자 세기 등 캠소프트웨어없이 Mach3에서 직접 만들어 바로 G코드를 생성해서 깎을 수 있다는 것이다. 그러므로 실제 적용할 수 있는 간단한 파트들은 CAD/CAM 없이 직접 Mach3에서 한번에 할 수 있다.

그림 6.21 마법사 선택하기

3 장의 포켓 밀링 예를 참고 할 수도 있고 아래 그림 6.22 은 텍스트 음각에 관한 마법사이다.

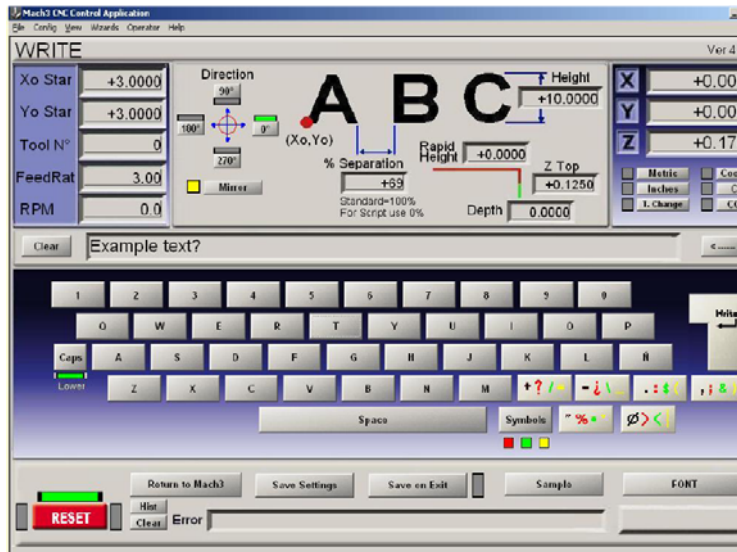


그림 6.22 글쓰기(Write)마법사 화면

마법사는 여러 제작자들에 의해 만들어 졌으며 그들의 의도에 따라 다르므로 컨트롤 버튼에 있어 약간의 차이가 있다. 대부분의 마법사는 여러분이 설정한 것을 저장 할 수 있게 하여 마법사를 다시 시도 할 때 같은 초기 값을 DRO 에 나타낸다.

그림 6.23 은 그림 6.22 처럼 Write 버튼을 누른 후의 Program Run 스크린 을 보여준다.

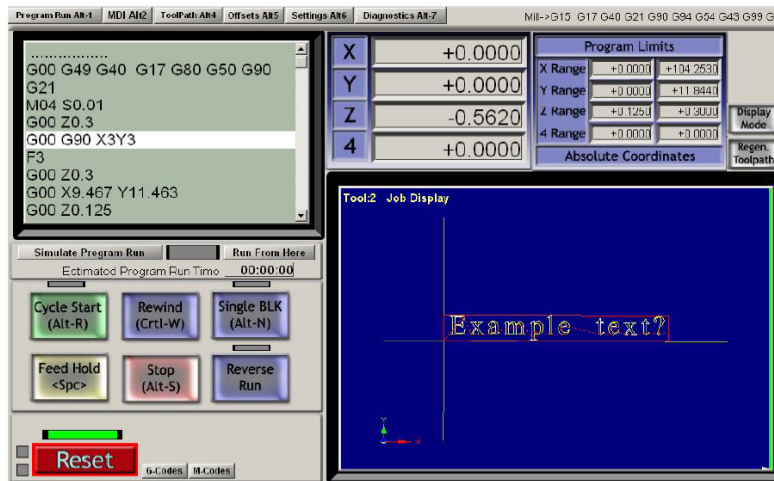


그림 6.23 (글 쓰기) Write 마법사 를 실행 후 화면

Last Wizard 버튼은 리스트에서 선택하는 것을 다시 반복 하지 않고 가장 최근의 마법사를 실행 한다.

Conversational 버튼은 Newfangled Solutiond 라는 회사 에서 디자인한 마법사들을 실행 한다. 이것들은 Mach3 가 제공하지만 실행을 위해서는 별도의 라이선스를 구매 해야 한다.

### 6.4 G 코드 파트 프로그램 로딩 하기

CAD/CAM 패키지나 혹은 직접 작성한 기존의 파트 프로그램이 있다면 Mach3 의 File > Load G code 메뉴에서 로딩 할 수있다. 파일이 선택 됐으면 Mach3는 로딩 한 후에 분석을 한다. 이것은 툴패스를 생성하여 화면에 나타내고 프로그램을 형성한다.

또한 Recent files 버튼을 눌러 최근에 사용한 파일들의 리스트를 로부터 표시 할 수도 있다.

로딩된 프로그램 코드는 G 코드 리스트 창에 표시 된다.

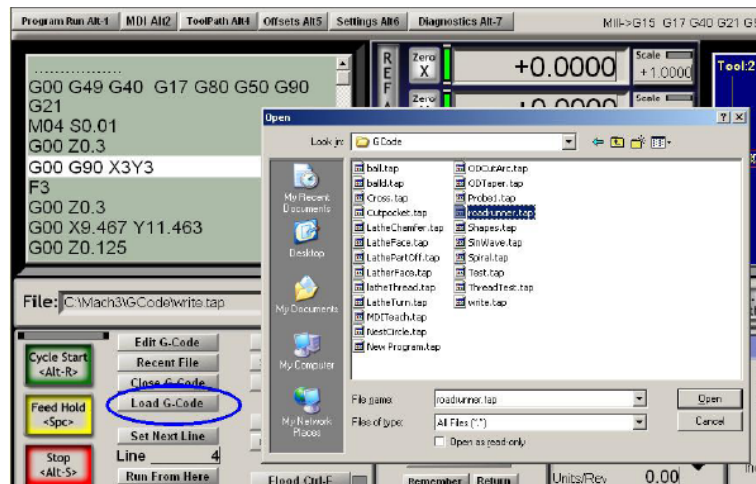


그림 6.24 G 코드 로딩

움직임은 하이라이트 된 라인을 따라 움직이는 것을 스크롤 할 수 있다.

## 6.5 파트 프로그램 편집 하기.

Config > Logic 에서 G 코드 편집기로 사용 하기 위한 프로그램을 선택하여 Edit 버튼을 눌러 코드를 편집 할 수 있다. 선정된 편집기는 새 창에 열려 선택한 코드를 보여 줄 것이다.

편집이 끝나면 파일을 저장 하고 편집기를 끝낸다.

편집하는 동안 Mach3 는 잠시 멈춘다. 만약 Mach3 창을 클릭한다면 잠기게 되지만 편집기를 닫으면 해제 된다.

편집기로 코드 변경 후 다시 분석하여 톨패스를 재생성 할 것이다. Regenerate 버튼을 이용하여 언제나 재생성 할 수 있다.

## 6.6 매뉴얼 작업으로 준비하고 파트프로그램 실행하기

### 6.6.1 직접 작성한 프로그램 입력

프로그램을 백지 상태에서 작성 하고자 한다면 Mach3 이외의 편집기에서 작성 하여 파일로 저장 하거나 파트프로그램이 로딩 되어 있지 않은 상태에서 Edit 버튼을 눌러 작성 한다. 이 경우 Save As 로 저장 하고 편집기를 닫아야 한다.

두 경우 다 Mach3 의 File > G code 로 로딩 해야 할 것이다.

경고: 코드 라인들의 에러 는 일반적으로 무시 된다. 세부 문장 체크에 의존 해서는 안 된다.

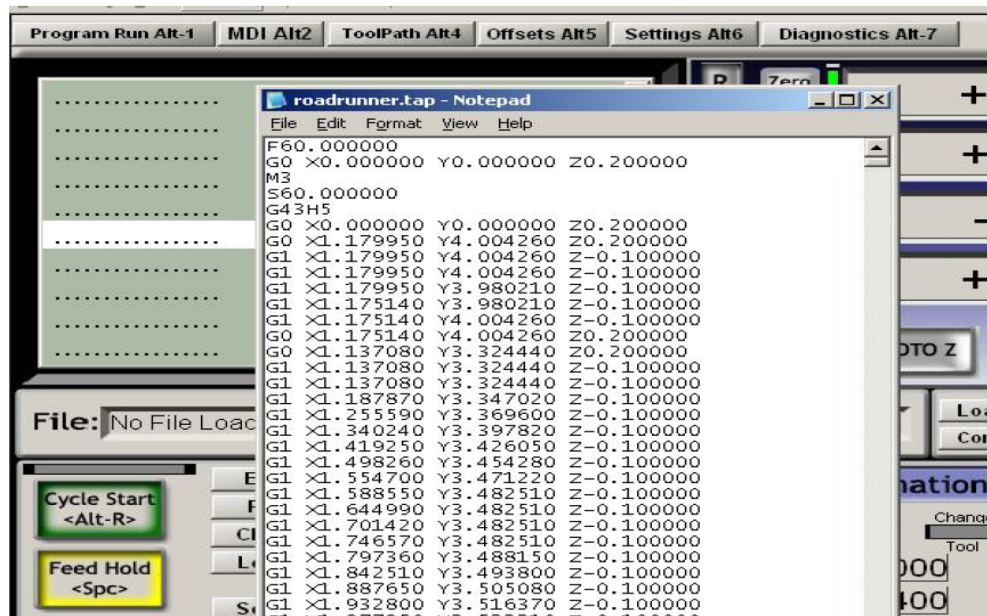


그림 6.25 windows 메모장으로 G 코드 편집 하기

### 6.6.2 파트프로그램을 실행 하기 전에

파트 프로그램이 시작 되기 전에 장비 상태에 관한 확인이 있어야 하기 때문에 G17/G18/G19, G20/G21, G40, G49, G61/G62, G90/G91, G93/G94 가 포함 되어야 한다.

Ref All 버튼을 이용하여 각 축들이 정해진 위치에 있는가를 확인해야 한다.

프로그램이 시작될 때 S 문자로 시작할 것인지 아니면 스피들 속도 값을 S DRO 에 입력하여 설정할 것인지에 관해 결정해야 한다.

적절한 피드 비율이 G01/G02/G03 명령어 수행 전에 설정되어 있는지 확인해야 한다. 이것은 F 문자 나 혹은 F DRO 에 데이터 입력에 의해 될 수 있다.

툴이나 작업 오프셋(work offset)을 선택할 필요가 있을 것이다.

마지막으로, 프로그램의 유용성이 증명되지 않았다면 가상으로 작업을 시도해서 잘못되는 것이 없는지 확인한다.

### 6.6.3 프로그램 실행

처음에는 프로그램을 잘 관찰하여 모니터링해야 한다. 피드비율을 다시 설정하거나 스피들 속도를 조절하여 소음을 줄여 제품을 최적화할 수 있는 방법을 찾을 수도 있을 것이다. 변화를 주고 싶다면 Pause 버튼을 누르거나 작업물로부터 높이 올려 바꾼 후에 다시 Cycle Start 를 클릭한다.

## 6.7 다른 파일을 불러들여 G 코드 만들기

Mach3 는 DXF, HPGL, JPEG 형식을 G 코드로 변환하여 커팅한다.

이것은 File > Import DXF/BMP/JPG 메뉴를 사용하여 진행된다. 파일 타입을 선택하고 초기 파일을 로드한다. 파트 프로그램에 포함될 변환, 피드, 쿨런트 명령어 등에 대한 정의를 내리기 위해서 파라미터가 나타날 것이다. Mach3 는 \*.TAP 파일을 새로 만들고 이것은 생성된 G 코드를 포함하게 되며 폴더와 파일 이름을 저장할 수 있게 한다. .TAP 파일은 Mach3 에 로딩되는 파트 프로그램으로 운영할 수 있다.



자세한 사항은 8 장에 기술되어 있다.

그림 6.27 불러들이기 필터로 선택하기

## 7. 좌표계, 툴 테이블 고정자

이번 장에서는 Mach3 가 주어진 위치에서 어떻게 움직이는지 설명 한다. 이것으로 좌표계의 개념을 설명 하고 장비 좌표계를 정의내리며 각 툴의 길이나 고정물에서 작업대상의 위치에 대한 구체화 작업, 그리고 필요에 따라 여러 옵션을 부가 할 수 있다. 처음 부분은 좀 어려울 수 있다. 그래서 직접 장비를 시도해 보라고 권유하고 싶다. 단지 이것은 앉아서 프로그램을 실행 하는 것만으로 알 수는 없으며 실제 어떤 위치로 가는지 직접 봐야 할 것이고, G 코드에 대한 이해도 필요로 한다.

Mach3 의 운영은, 이번 장의 구체적인 이해를 요구 하는 것은 아니지만, 이것에 대한 개념을 이해해서 하는 것을 알면 작업을 하는 것이 훨씬 빠르고 훨씬 신뢰성을 높일 수 있다는 것을 발견 하게 될 것이다.

### 7.1 장비 좌표계(Machine coordinate system)

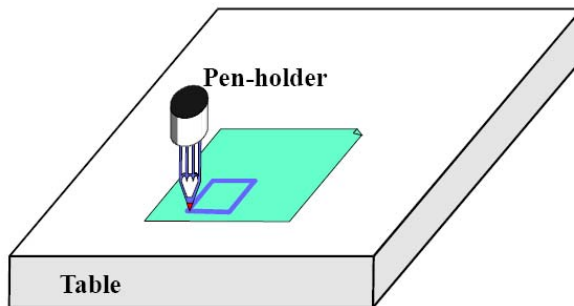


그림 7.1 기초 그림 그리기

대부분의 Mach3 화면은 X, Y 축 과 같은 DRO 가 있다는 것을 보아 왔다. 파트를 정확하게 만들려고 한다면 그리고 어떤 부분과도 장비가 충돌 하는 것을 막기 위해서는 진행하려는 파트 프로그램이나 작업을 실행 할때 각 값들이 어떤 의미를 가지고 있는 가를 정확히 알아야만 한다.

이것은 장비를 직접 보면서 이해 하는것이 가장 쉬운 방법이다. 우리는 그동안 가상의 장비를 운영하여 좌표계가 일하는 것을 보아왔다.

위의 그림은 카드보드나 종이위에 펜을 이용하여 그림을 그리는 것이다. 이것은 고정된 테이블과 X 축과 Y 축으로 움직일 수 있는 원형 펜 홀더로 구성되어 있다. 그림의 사각형은 종이위에 그린 그림을 보여 준다.

그림 7.2 는 왼쪽 아래 코너의 테이블 표면에서 부터 측정 하는 장비 좌표계(Machine coordinate system)를 보여 준다. 보기와 같이 종이의 왼쪽 아래 구석이 X=2, Y=1 그리고 Z=0(종이의 두께는 무시하고) 에 있다. 펜이 가리키는 점은 X=3, Y=2 그리고 Z=1.3 처럼 보인다..

만약 펜이 가리키는 점이 테이블의 코너에 있었다면, 이장비에서, 홈이나 기준위치가 될 것이다. 이 지점은 대개 스위치가 on 될때 축이 움직이는 홈 스위치의 위치로 정의 된다. 어떤 경우라도 각 축의 영점 위치는 절대 장비 영점(absolute machine zero)이라고 불리우는 곳이 있다.

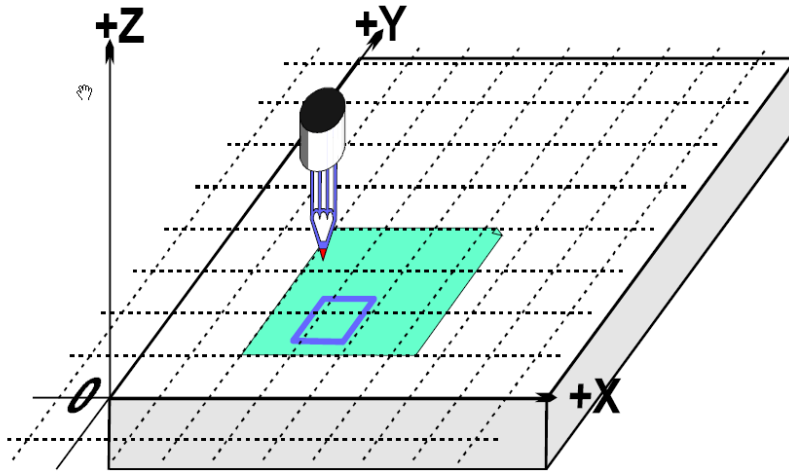


그림 7.2 장비 좌표계

펜이 가리키는 점은, 커팅툴의 끝부분과 길이, 실제 작업이 되는 부분이며 제어점(controlled point) 이라고 불린다. 축 DRO 는 항상 어떤 좌표계와 관련하여 제어점 좌표를 표시 한다. 이 장을 읽어야 하는 중요한 이유는 항상 장비의 고정된 곳에서 측정 좌표계의 영점 이 되는 것이 간단한 것 만은 아니란 것이다.

왜 그런지는 간단한 예제를 통해서 본다.

아래 파트 프로그램은 그림 7.1 의 1 인치 사각형 그림에 적합한 것이라는 것을 알 수 있다.

```

N10 G20 F10 G90 (특정 단위, 느린 피드비율 등등의 설정.)
N20 G0 Z2.0 (펜을 위로 올림)
N30 G0 X0.8 Y0.3 (아래로 내림, 사각형의 왼쪽)
N40 G1 Z0.0 (펜을 아래로)
N50 Y1.3 (1 을 방금 했기 때문에 G1 을 생략 할 수 있음)
N60 X1.8
N70 Y0.3 (시계 방향으로 회전)
N80 X0.8
N90 G0 X0.0 Y0.0 Z2.0 (펜을 움직여 위로 올린다)
N100 M30 (프로그램 끝)

```

모든 코드에 대한 이해가 부족해도 무슨 일이 일어 나는 지는 알 수 있다. 예를 들어 N30 라인 에서 장비는 제어점  $X=0.8$   $Y=0.3$  으로 움직이라고 되어 있다. N60 라인에서는 제어점이  $X=1.8$ ,  $Y=1.3$  으로 되어 있어 DRO 는 아래와 같이 읽는다.

X Axis 1.8000    Y Axis 1.3000    Z Axis 0.0000

물론 문제는 사각형이 그림 7.1 과 같이 그려지지 않았다는 것이다. 파트 프로그램 작성자는 종이의 구석에서 부터 측정 했지만 장비는 장비의 영점에서 부터 측정 한다.

## 7.2    작업 오프셋(Work Offset)

Mach3 는 다른 모든 장비 컨트롤러와 같이 좌표계의 원점을 움직이게 할 수 있도록 한다.

이것을 좌표계 오프셋 (offsetting) 이라고 한다.

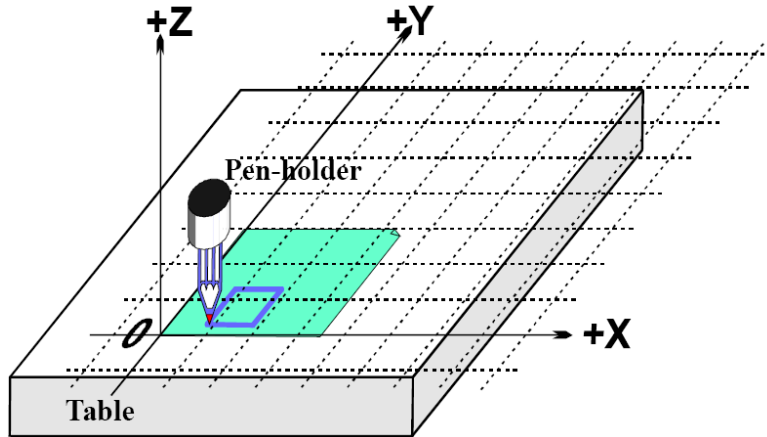


그림 7.3 종이의 코너에 좌표계의 원점을 옵셋하기

그림 7.3 은 현재 좌표계를 종이의 코너에 옵셋값을 주어 조절 한다면 어떻게 될지 보여 준다. 알아 뒤야 할 것은, G 코드는 항상 제어점을 현재의 좌표계에 주어진 숫자에 의해 움직인다는 것이다.

이때 옵셋을 작업 옵셋(work offset) 이라고 부르고 0,0,0 점을 이 좌표계의 원점 이라고 한다.

이 옵셋은 유용하여 Mach3 에서는 여러 가지 방법으로 설정할 수 있지만 모두 Offset 화면상에서 정돈되어 있다.(부록 1 을 참고)

### 7.2.1 작업 원점을 주어진 좌표에 설정하기

가장 분명한 방법은 두가지 단계로 되어 있다.

1. Offset 화면으로 간다. 제어점을 새로 만들고 싶은 원점 위치로 옮긴다. 조깅 하거나 G0 을 이용하여 얼마큼 움직여야 하는지 데이터를 입력 할 수 있다.

2. 화면의 Current Work Offset 부분에 있는 각 축 옆의 터치 버튼을 클릭한다. 처음 누를때 바뀐 현재 좌표가 Part Offset DRO 에 입력이 되고 축 DRO 는 0 으로 된다. 계속해서 다른 축에도 같이 적용 된다.

어떤 일이 있었는 가를 알고 싶다면 다음내용이 도움이 될 것이다. 작업 옵셋 값들은 항상 축 DRO 에 가산 되어(즉, 제어점의 현재 좌표계) 제어점의 절대 장비 좌표계를 만든다. Mach3 의 Machine Coords 버튼을 누르면 제어점의 절대 좌표점을 표시한다. LED 는 깜빡이며 절대좌표라는 것을 경고해 준다.

원점을 어디로 정할지 안다면 다른 방법을 사용하여 옵셋을 설정 할 수 있다.

종이의 코너는 눈으로 봤을때 기준점으로 부터 오른쪽으로 2.6인치 1.4 인치 위에 위치해 있다. 이정도로 충분히 정밀 하다고 가정 하자.

1. 2.6 과 1.4 를 X 와 Y 의 Offset DRO 에 각각 입력 한다. 축 DRO 값은 변할 것이다.. 기억 할 것은 제어점을 실제 위치로 움직이지 않았기 때문에 이것의 좌표는 원점으로 움직일 때 좌표값들은 변해야 한다.

2. 필요하다면 MDI 라인에서 G00 X0 Y0 Z0 를 사용 하여 확인 한다. 펜은 테이블 위 종이의 코너에 놓여질 것이다.

이것은 작업 옵셋 1 번을 이용하여 설명 하였다. 1 에서 255 까지의 번호를 사용 할 수 있다. 사용 중에는 오직 하나 만 선택 할 수 있으며 이것은 Offset 화면의 DRO 에 의해 선택 되어 지거나 파트 프로그램의 G 코드의 사용에 의해서 선택 할 수 있다.(G54 에서 G59 까지)

작업 옵셋을 설정 하는 마지막 방법은 축 DRO 에 값을 입력 하는 방법이다. 현재 작업 옵셋은 갱신되어 제어점은 축 DRO 에 의해 재 정립된다. 주의 할 것은 장비는 움직이지 않는다. 단지 좌표계의 원점이 바뀌었을 뿐이다. X=0, Y=0 등등, 버튼은 해당 되는 축 DRO 에 0 을 타이핑 하는 것과 동등하다.

Offset 화면에서 설정하는 옵셋에 익숙 하지 않다면 이 방법을 사용 하지 말기를 권유 하는 바이다.

작업 옵셋에 의한 현재 좌표계의 옵셋팅으로 사각형을 종이위의 원하는 위치에 그릴 수 있다.

### 7.2.2 실제 장비에서의 홈

위에 언급한 바와 같이 처음에는 단정하게 보여도 종종 Z 홈 위치를 테이블의 표면에 하는 것이 좋은 것은 아니다. Mach3 는 모든 축의 기준을 정할 수 있는 버튼이 있다(혹은 각 개인의 기준을 정할 수 있다). 홈 스위치가 장착된 장비에서 각 선형 축이나 혹은 선택된 축에서 스위치 방향으로 움직여 스위치를 작동 시킨후 살짝 스위치 off 위치로 움직이게 한다. 절대 장비 좌표계 원점(즉 장비 영점) 주어진 X, Y ,Z 값에 설정된다(일반적으로 0.0). 사실 0 이 아닌 다른 숫자로 홈을 설정 할 수있지만 여기서는 잊어버린다.

Z 홈 스위치는 일반적으로 테이블에서 가장 높은 곳에 위치 한다. 물론 기준 점은 장비 좌표계의 Z=0 이 되고 모든 작업 위치는 그보다 아래가 되며 장비 좌표계에서는 Z가 마이너스 값을 갖게 된다.

다시 말하지만 이것이 분명히 이해가 가지 않아도 걱정 할 필요는 없다. 제어점이 홈 위치에 있을때 방해가 되는 것이 없게 하는 것이 편리 하며 테이블의 작업물에 있어서 편리한 좌표계를 설정하기 위해 작업 옵셋을 사용하는 것이 쉽다.

## 7.3 엔드밀의 서로 다른 길이에 관하여

지금까지의 내용에 문제가 없다면 다음 경우를 생각해 볼 때가 됐다.

그림 7.4 와 같이 빨간색 사각형을 그린다고 하자. Z 축을 조그 하여 빨간색 펜을 넣는다. 그때 빨간색 펜의 길이가 그 전 것보다 길다.

때문에 현재 좌표계 원점에서 끝 부분이 테이블과 충돌 할 것이다. (그림 7.5)

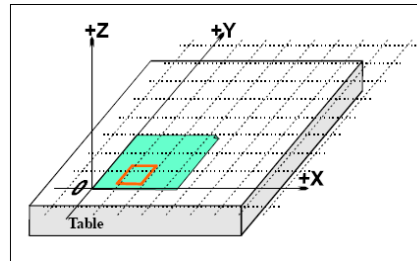


그림 7.4 다른 색으로 한다.

Mach3 는 다른 CNC 컨트롤러와 같이, 툴에 관한 정보를 저장 할 수 있는 기능이 있다. 이 툴 테이블(Tool Table)은 각기 다른 256 가지의 툴에 관한 정보를 저장 한다.

Offset 화면에서 툴(Tool) 번호와 정보에 관한 화면을 볼 수 있을 것이다. DRO 에는 Z offset, 직경(Diameter) 그리고 T 라는 라벨이 있을 것이다. DRO Touch Correction 이나 그와 관련 On/Off 버튼은 생략 해도 상관 없다.

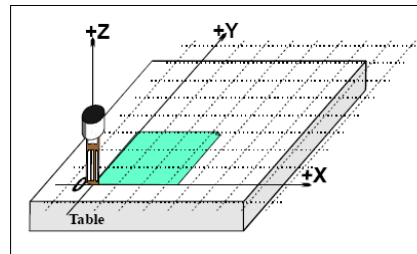


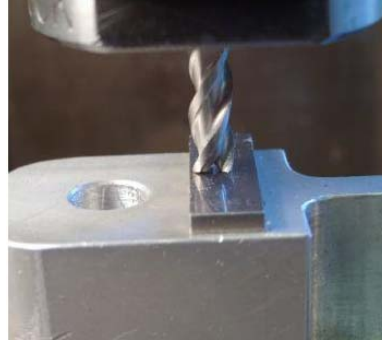
그림 7.5 교체 후 충돌 한 화면



기본적으로 Tool#0 이 선택 되어 있지만 이것의 오프셋 스위치는 Off 로 되어 있을 것이다. 툴의 직경에 관한 정보는 커터 보정(Cutter Compensation) 으로 사용 된다.

### 7.3.1 사전에 설정 가능한 엔드밀(tool)

여기에서, 여러분의 장비에는 툴 홀더 시스템이 있고 툴을 넣을 때 정확하게 같은 위치에 놓인다고 가정 한다. 이것은 자동 잠금 척(chuck) 이나 혹은 여러개의 척이 있는 밀링 장비가 될것이다. 툴의 위치가 매번 바뀐다면 매번 오프셋을 바꿔 주어야 한다. 이것은 나중에 기술 하기로 한다.



여기 그림 그리는 장비에서 , 1 인치 깊이로 펜 홀더에 펜이 있다고 가정 해 보자. 그리고 빨간 펜은 4.2 인치 길이 이고 파란 펜은 3.7 인치 길이 이다.

1. 장비가 막 기준점과 홈 포지션이 잡혀 있고 작업 오프셋이 Z=0.0 으로 종이의 코너에 정의 내려지고 있다고 가정한다. Z 축을 5 인치 조그해서 올리고 파란 펜에 맞게 한다. 툴 번호 DRO 에 1 을 넣고 Offset On/Off 에서 On 으로 놓는다. Z 를 조그해서 펜이 닿게 한다. **그림 7.6 사전설정 가능한 엔드밀**

Z 축 DRO 는 펜이 홀더로부터 2.7 인치 떨어져 있기 때문에 2.7 로 나타날 것이다. 그리고 Touch 버튼을 누른다. 이것으로 Tool #1 의 Z 오프셋에 2.7 인치가 적용 됐다. Offset On/Off 은 LED 로 상태를 나타내고 툴 오프셋을 적용시킨다. 그러므로 Z 축 DRO 는 0.0 이라고 표시된다. 그 전과 같이 예제 파트 프로그램을 진행 할 수 있다.

2. 다음으로 빨간 펜을 사용 하기 위해 Z 축을 위로 올린 후 파란 펜을 빼내고 빨간 펜을 넣는다. 물리적으로 펜을 교체 하는 것은 축 DRO 들을 변화시키지 않는다. 툴 오프셋 LED 를 off 시키고 Tool #2 를 선택한 후 종이의 코너로 조그 한 다음 Touch 를 누른다. 이것은 툴 #2 의 오프셋을 3.2 인치로 만든다. Tool #2 의 오프셋을 다시 On 시키면 축 DRO 에 Z=0.0 이 되어 파트 프로그램이 빨간 사각형을 그릴 수 있게 된다.

3. 툴 1 과 2 가 설정되었으므로 원하는 만큼 바꿀 수 있고 적절한 툴 번호 선택과 오프셋을 on 함으로서 현재의 좌표계에 맞출 수 있게 된다. 이런 툴 선택과 오프셋의 on/off 는 파트 프로그램에서 할 수 있으며(T 문자, M6, G43 그리고 G94) DRO 들은 표준 Program Run 화면에 나타나 있다.

### 7.3.2 사전에 설정 불 가능한 엔드밀(tool)

일부 툴 홀더는 매번 툴(팁)을 교환 할때 정확하게 그 자리에 맞지 않는 경우가 있다. 예를 들면 라우터의 콜렛은 일반적으로 구멍이 깊게 나 있어 안쪽 부분에 닿기 까지 힘들 경우가 있다. 이런 경우 매번 교체 할때 마다 오프셋을 설정 해야 할 것이다. 이럴경우에도 하나 이상의 오프셋을 사용할 수있다. 만약 물리적인 고정체가 없다면 엔드밀을 교체 할 때 마다 작업 오프셋의 Z 를 다시 설정하는 것이 더 편할 수 있을 것이다.

## 7.4 오프셋값이 저장 되는 방법

254 개의 작업 오프셋은 Mach3 의 테이블 하나에 저장 된다. 255 개의 툴 오프셋과 직경을 다른 테이블에 저장 할 수 있다. 오프셋 화면에서 Work Offsets Table 과 Tool Offsets Table 을 사용하여 이 테이블 들을 볼 수 있다. 이 테이블들에는 현재 Mach3 에서 쓰이지 않는 추가적인 정보를 위한 빈 공간이 있다.

Mach3 는 일반적으로 각기 다른 프로그램 진행때 마다 모든 작업과 툴 오프셋 값을 기억 하지만 각 프로그램을 달기 전에 값들을 저장 할 것인지 물어 본다. Config > State 대화 상자에서 이렇게 자동으로 저장 할 수 있게 하던지, 아니면 저장 하지 않게 하던지를 선택 할 수 있도록 설정 할 수 있다. 하지만 자동 저장 옵션이 선택 되어도 저장을 하게 하는 테이블 화면이 있는 대화 상자에서 Save 버튼을 눌러 저장 할 수도 있다.

## 7.5 같은 작업 여러 번 반복 하기 - 고정자 사용

이제 여러 장의 종이에 같은 그림을 그린다고 가정한다. 같은 위치에 종이를 테이프로 붙이는 것은 어려울 것이며 매번 옵셋을 설정하는 것도 번거로울 것이다. 더 좋은 방법은 핀을 고정할 수 있는 판을 준비하고 미리 구멍이 뚫린 종이에 맞춰 사용하는 것이다. 이것이 머신샵 에서 오랜 기간 동안 사용해 왔던 방법이란 것을 알게 될 것이다. 그림 7.7은 고정체에 대한 예제이다.

종종 고정체는, 못과 같은 것으로, 테이블 상에 항상 같은 위치에 고정 되어 있다.

현재 좌표계는 작업 옵셋 #1 을 고정체 상에서 종이의 코너 부분에 설정 함으로써 움직일 수 있다. 예제 프로그램을 진행 하면 그 전과 똑 같이 사각형을 그리는 것을 볼 수 있을 것이다.

이것은 물론 두께에 의한 Z 의 위치도 고려 되어야한다.

새로운 종이를 핀에 고정시키고 사각형을 또 다른 설정 없이 똑 같이 그린다.

2개 대신 3개의 홀이 있는 종이를 그림 7.8과 같이 놓고 사용할 수 있다. 물론 고정체의 어떤 점에서도 옵셋 좌표계의 원점을 설정 할 수 있다. 위의 그림 그리는 장비에서 종이의 왼쪽 아래 구석을  $X=0, Y=0$  로 놓을 수 있고 고정물의 표면은  $Z=0$  로 놓는다.

하나의 물리적 고정체에 여러 개의 작업을 수행 할 수 있도록 사용하는 것이 일반적이다. 그림 7.9는 2,3 개의 복합 구멍 고정물을 나타 낸다.

물론 사용 하게 될 각각 분리된 두 개의 작업 옵셋을 가진다. 그림 7.8은 현재 좌표계가 2 개의 홀을 가진 종이를 사용 하고 있는 것을 보여 준다.

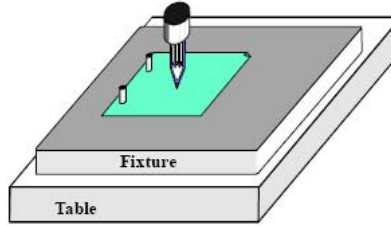


그림 7.7 두개의 고정 핀이 있는 장비

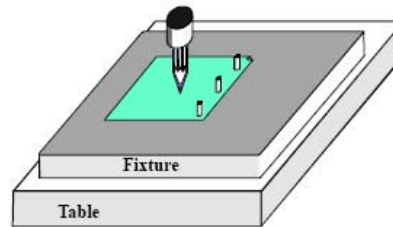


그림 7.8 3 개의 고정핀

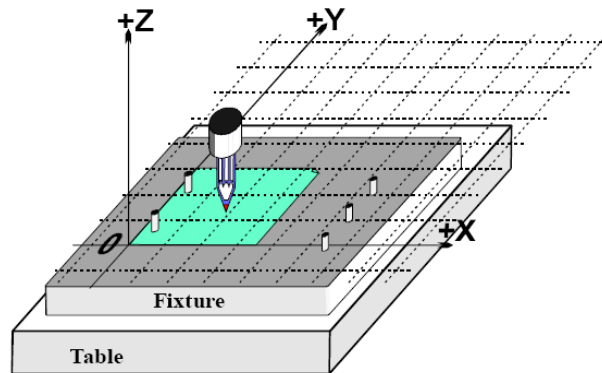
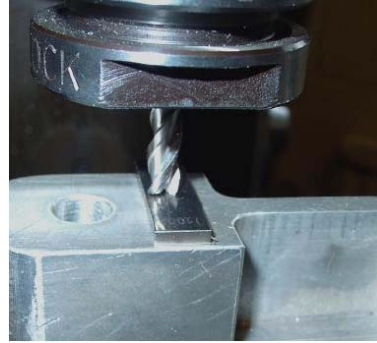


그림 7.9 두개의 고정물

## 7.6 터칭의 실용성

### 7.6.1 엔드밀

수동 장비에서 작업물에 터칭 하는 것을 핸들로 느낄 수 있지만 더 정확한 작업을 위해서는 종이나 혹은 얇은 플라스틱 같은 필러 게이지를 이용 하는 것이 좋다. 그림 7.10 을 참고 하라.



Offset 화면에서 Touch Correction DRO 에 이 필러나 슬립 게이지의 두께를 계산에 넣을 수 있다 (Set Tool Offset 버튼 옆에 있다). 그리고 On/Off 를 클릭 하여 On 으로 한다. touch 를 할 때는 언제나 LED 가 반짝이며 정정이 됐음을 알려 준다.

예를들어 DRO Z= -3.518 에 0.1002" 슬립게이지가 살짝 고정되어 있다고 하자. 툴 DRO 에 3 을 입력해 Tool #를 #3 로 만든다. Gage Boock Height 에 0.1002 를 DRO 에 넣는다. Touch 를 클릭 하면 DRO 는 Z=0.1002 라고 읽는다.

그림 7.10 슬립 게이지-밀이 Z 옵셋에 닿을 때

즉 컨트롤 포인트가 0.1002 가 된다.

그리고 Z 옵셋은-0.1002 이 된다. 그림 7.11 은 Set Tool Offset 을 클릭 하기 전의 프로세스를 보여 준다.

만약 정밀한 원통형 게이지가 있고 적절한 크기의 평평한 모양의 작업물이 있다면 이것을 이용 하는 것이 조그 다운 해서 필러 게이지에 닿는 것보다 낫을 것이다. 조그 다운해서 롤러가 툴 아래에서 왔다 갔다 통과 하지 않도록 한다. 그리고 툴 바로 아래에서 살짝 닿을 듯 구를 때 까지 아주 천천히 조그 업 한다. 그리고 Touch 버튼을 누른다.

조그 다운해서 필러 게이지에 닿는 것은 툴의 커팅 면에 손상을 줄 수 있으니 유념한다.

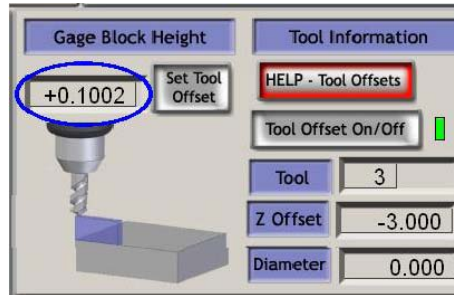


그림 7.11 Z 옵셋 입력

### 7.6.2 점점 찾기(Edge finding)

엔드밀의 플러트로 인해서 정확하게 X 나 Y 를 설정 하기 힘들다. 특수한 점점 찾기를 (edge finder) 사용하는 그림이 아래 7.12 와 같다.

Touch Correction 이 이때에도 사용 될 수 있다. 이때 프로브 팁의 반경을 알아야 하고 필러(슬립)게이지의 두께도 알아야 한다.



## 7.7 G52&G92 옵셋

제어점(controlled point)의 옵셋 설정에는 또 다른 방법이 있다. 그것은 G 코드의 G52 와 G92 의 사용이다.

G52 를 실행함으로써 Mach3 는 어떤 제어점 값이던,

X,Y 혹은 Z 의 주어진 값을 더함으로써 실제 장비의 위치 옵셋으로 정한다. 그림 7.12 장비에서 표면 찾기

G92 를 사용함으로써, Mach3 는 현재 제어점의 좌표계가 X, Y , Z 에 의해 어떤 값이 될지를 정한다.

G52 이던 G92 이던 현재 좌표계의 원점에 다른 오프셋 값들을 추가로 설정했다하여 움직이지 않는다.

### 7.7.1 G52 사용하기.

G52 의 간단한 예로는,작업 대상물에 2 개의 동일한 모양을 대상물위의 각각 다른 위치에서 작업 한다고 생각해 보자. 먼저 번에 1 인치 짜리 사각형을 코너가 X=0.8, Y=0.3 에서 그렸었다.

G20 F10 G90( 단위, 느린 피드레이트 등등 설정)

G0 Z2.0( 펜 올리기)

G0 X0.8 Y0.3(사각형의 아래 왼쪽으로 빨리 옮기기)

G1 Z2.0(펜 내기리)

Y1.3(G1 생략 가능 - 중복)

X1.8

Y0.3(시계방향으로 라운드 모양)

X0.8

G0 X0.0 Y0.0 Z2.0(펜을 멀리 떨어지게 하고 올린다)

다른 사각형을 그리고자 하며 이때 코너가 X=3.0 Y=2.3 이라면 위의 코드를 똑 같이 쓸 수 있지만 두 번째 카피로 들어가기 전에 적용할 G52 와 오프셋을 사용 한다.

G20 F10 G90

G0 Z2.0(펜 올리기)

G0 X0.8 Y0.3(사각형의 아래 왼쪽으로 빨리 옮기기)

G1 Z0.0(펜 올리기)

G52 X2.2 Y2(두 번째 사각형을 위한 임시적 오프셋)

G0 X0.8 Y0.3(사각형의 아래 왼쪽으로 빨리 옮기기)

G1 Z0.0(펜 올리기)

Y1.3(G1 생략 가능 - 중복)

X1.8

Y0.3(시계방향으로 라운드 모양)

X0.8

G52 X0 Y0(임시 오프셋 제거)

G0 X0.0 Y0.0 Z2.0 ( 펜을 멀리 떨어지게 하고 올린다)

G 코드 서브루틴 생성은 가능 하므로 일반적인 코드는 한번 쓰여져서 여러 번 사용이 가능하다. 서브루틴 버전은 아래 와 같다. 펜 업 다운 명령어는 고정되어 있고 서브루틴은 0.0에서 두 사각형의 코너설정을 위해 G52가 사용된다.

```
G20 F10 G90 (단위, 느린 피드레이트 등등 설정.)
G52 X0.8 Y0.3 (처음 사각형 그리기 시작)
M98 P1234 (처음 위치에서 사각형의 서브루틴 부르기)
G52 X3 Y2.3 (두번째 사각형 시작)
M98 P1234 (두번째 위치에서 사각형의 서브루틴 부르기)
G52 X0 Y0 (중요 - G52 옵션 제거)
M30 (프로그램의 끝에서 다시 돌아가기)
O1234(서브루틴 1234의 시작)
G0 X0 Y0 (사각형의 아래 왼쪽으로 빨리 옮기기)
G1 Z0.0 (펜 다운)
Y1 (G1 생략 - 중복)
X1
Y0 (시계 방향으로 라운드 모양)
X0
G0 Z2.0 펜 올리기)
M99 (서브루틴에서 돌아가기)
```

알아 둘 것은 각각의 G52 는 새로운 옵션을 적용 하고 그 전에 적용 했던 G52 는 무시 된다.

### 7.7.2 G92 사용 하기

가장 간단한 G92 의 예는, 주어진 점에서, X, Y 를 0 으로 설정하는 것이지만 어떤 값이던 원하는 대로 할 수 있다. G92 옵션을 취소 하는 가장 간편한 방법은 G92.1 을 MDI 라인에 입력 하는 것이다.

### 7.7.3 G52 와 G92 의 처리

옵션은 원하는 축에 원하는 만큼 그 축의 값을 포함시켜 설정 할 수 있다. 축 이름이 주어 지지 않을 때에는 바뀌지 않는다.

Mach3 는 G52 와 G92 옵션을 위해 같은 내부 메커니즘을 사용한다. 단지 여러분의 X Y Z 문자와 계산이 다를 뿐이다. 만약 G52 와 G92 를 같이 사용 하게 된다면 아마도 혼동이 되어(Mach3 도 마찬가지로) 결국은 사고가 발생 할 수도 있다. 어떻게 작동하는지 확인 하고 싶다면 옵션을 설정하고 제어점을 좌표계로 옮겨본다. 예로서 X=2.3 Y=4.5 라고 하자. 있어야 할 절대 장비 좌표계를 예상하고 이를 확인하기 위해 Mach3 화면의 장비 좌표의 Mach 버튼을 누른다.

옵션을 사용한 후 지우는 것을 잊지 말라.

경고 : G92 옵션으로 되는 대부분의 것들은, 작업 옵션이나 G52 옵션으로 더 잘 사용 할 수 있다. 왜냐하면 G92 는 제어점이 있는 곳과 G92 가 실행될 때의 축 문자에 의존 하기 때문이고 프로그램의 변화는 충돌을 유도하는 버그를 쉽게 만든다. 많은 작업자들이 세 개의 옵션을 끊임없이 파악하고 있는 것이 쉽지 않다는 것을 알게 되며, 만약 혼동이 되면 쉽게 툴(팁)을 손상 시키거나 장비에 좋지 않은 영향을 가하게 될 것이다.

## 7.8 툴의지름(Tool diameter)

그림의 파란색 사각형이 아이들의 그림판 높이에 맞는 조각이라고 가정하자. G 코드는 제어점을 움직인다는 것을 기억하기 바란다. 예제의 파트 프로그램은 1 인치 사각형을 그린다. 만약 툴이 두껍게 느껴지는 펜이라고 한다면 1 인치 사각형 보다 작은 홀이 형성 될 것이다. 그림 7.12 참고.

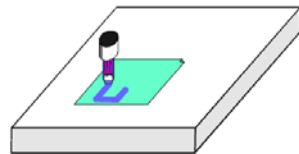


그림 7.12 지름이 넓은 툴(팁)

같은 문제가 엔드밀이나 슬롯 드릴에도 발생 할 것이다. 포켓커팅 하거나 혹은 중간에 조그만 모양만 만들어 질 것이다. 이런 것들은 다른 방법의 보완이 필요하다.

이것이 쉽게 들릴지 모르지만 실기에 있어서는 시작과 끝에 있어서 아주 힘든 것들을 만나게 된다. 이런 것들은 대개 마법사나 혹은 각자의 CAD/CAM 소프트웨어를 이용하여 문제를 다루게 한다. Mach3 는 파트 프로그램이 실제 절삭중인 툴의 직경을 보정 할 수 있도록 해 준다. 이런 기능은 파트 프로그램의 작성자가 사용할 커터의 직경을 정확히 모를 때 중요하다. 툴 테이블은 툴의 직경을 정의 내릴 수 있도록 해주고 또는 실제 사용하는 명목상의 차이도 정의 내릴 수 있도록 해준다. 자세한 내용은 Cutter Compensation 을 보기 바란다.

## 8. DXF, HPGL 그리고 이미지 파일 불러오기(Import)

이 장은 파일 불러오기와 Mach3 에 의한 파일 변환에 대해 설명한다. 그리고 간단한 G 코드와 그에 관한 기능에 대해 제한적인 지식을 갖고 있다고 간주 할 것이다.

### 8.1 소개

지금까지 보아 왔던 것처럼 Mac3Mill 은 파트 프로그램을 이용하여 장비에서 톨의 움직임을 제어한다. 여러분은 지금까지 파트 프로그램을 직접 작성 했을 수도 혹은 CAD/CAM 을 이용하여 만들었을 수도 있을 것이다.

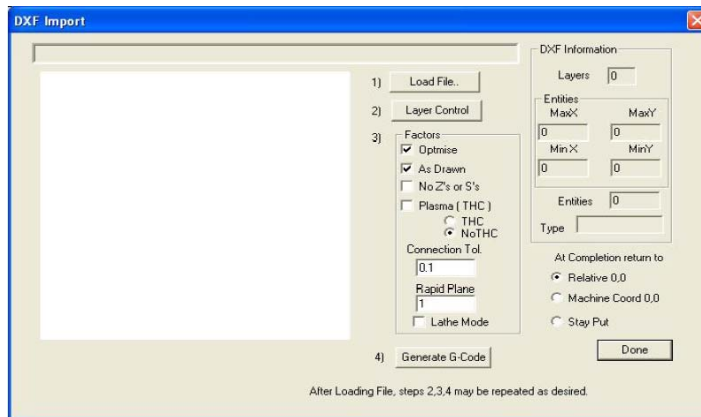
DXF, HPGL, BMP 혹은 JPEG 포맷으로 정의되는 그래픽 파일을 불러오는 것은 프로그래밍의 중간 레벨을 원한다. 직접 코딩 하는 것 보다는 쉽지만 CAD/CAM 패키지 에 의한 프로그램 출력 보다 장비의 제어에 더 제한적이다.

자동 Z 제어 기능과 Inhibit-Z 값을 감소시키는 재연적 실행은 불러들여진(imported) DXF 와 HPGL 파일을 바탕으로 하는 일련의 황삭 커팅을 하는데 강력한 기능이 된다.

### 8.2 DXF 불러오기(Import)

대부분의 CAD 프로그램은 비록 CAM 기능을 제공하지 않더라도 DXF 포맷으로 파일을 생성 할 수 있다. 파일에는 라인과 호 등등 그림의 시작과 끝에 대한 설명과 레이어에 대한 정보가 포함되어 있다. Mach3 는 그런 파일을 불러들여 각 레이어의 특정 톨과 피드 비율 그리고 커팅의 깊이를 지정한다. DXF 파일은 반드시 바이너리가 아닌 텍스트 포맷으로 되어 있어야 한다. 그리고 Mach3 는 **선, 폴리라인, 원, 그리고 호(arc) 만 불러들이며 텍스트는 불러들여 지지 않는다.**

임포트 하는 동안 a)스핀들 공회전 움직임을 줄이기 위해 선의 순서를 최적화 할 수 있고 b)그림의 실제 좌표를



사용 하거나 혹은 옵션을 주어 하단 왼쪽이 0.0 이 되게 할 수 있고 c) 선택적으로 플라즈마/레이저커터의 아크/광선을 제어하기 위해 코드를 넣을 수 있으며 d)도면을 Z/X 로 변환해서 회전 작업을 만들 수 있다. DXF 임포트(Import)는 화면 상단 File 메뉴에 있다. 대화상자는 그림 8.1 과 같다.

그림 8.1 DXF 임포트 대화 상자

### 8.2.1 파일 로딩

이번에는 파일을 불러 들이는 4 개의 단계를 보여 준다. 1 단계는 DXF 파일을 로딩 하는 것이다. Load File 버튼을 클릭하면 파일 열기 대화 상자가 나타난다. 그림 8.2는 두개의 사각형과 원을 나타내주는 파일 예제다.

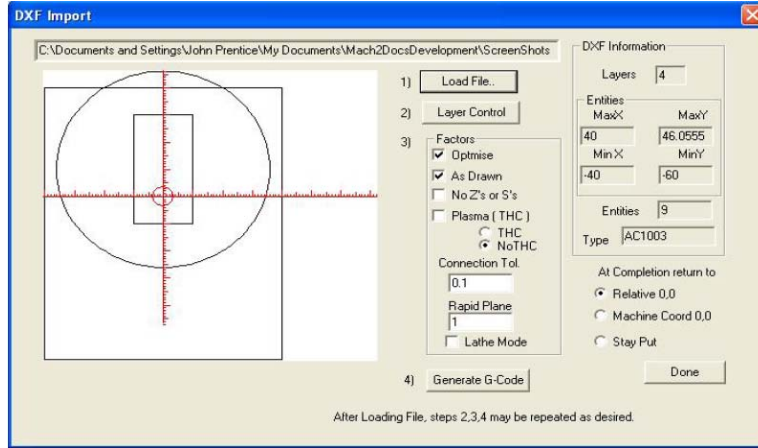


그림 8.2 - 8 개의 선과 하나의 원

### 8.2.2 각각의 레이어를 위한 작업 지정

2 단계는 그림의 각 레이어 별, 선들이 어떻게 다루어 지는 가를 정의 내리는 것이다. Layer Control 버튼을 클릭하여 그림 8.3 과 같은 대화 상자가 나타나도록 한다.

커팅 하고자 하는 선이 있는 레이어를 ON 시키고, 사용할 툴과 절삭 깊이, 사용할 피드 비율, 스피indel 속도(스텝/방향이 있는 것이나 PWM 스피indel 컨트롤러가 있는 것에 한해서) 그리고 커팅할 레이어의 순서를 선택한다. 알아둘 것은, 절삭 깊이값은 Z 값이 사용 되기 때문에 작업대상물의 표면이 Z=0 라면 마이너스 값을 갖게 될 것이다.

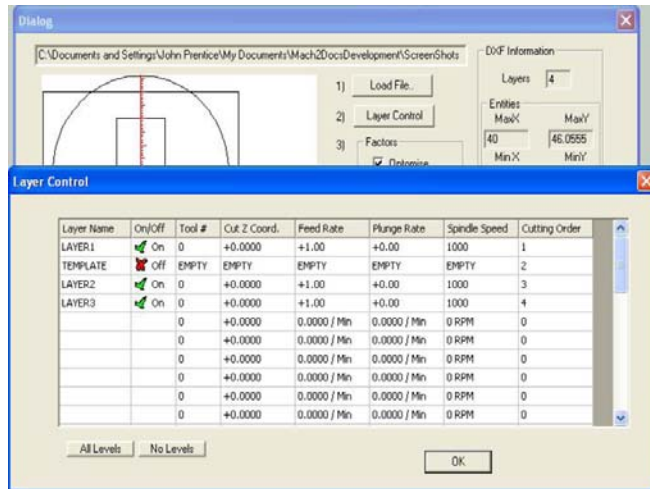


그림 8.3 각 레이어를 위한 옵션



### 8.2.3 변환(Conversion) 옵션

다음으로 변환 과정을 위한 옵션을 선택한다.(그림 8.2의 step 3 참고)

**DXF Information** : 파일의 일반적 세부사항을 포함하며 진단 목적을 위해 유용하다.

**Optimize**: 이것이 선택되지 않으면 선과 같은 개체들은 DXF 파일에 나타나는 선의 순서대로 자르게 될 것이다.

**As Drawn** : 이것이 선택되지 않으면 G 코드의 0점 좌표는 드로잉(그림)의 하단 좌측이 될 것이다. 물론 선택 되면 G 코드가 정하는 좌표가 된다.

**Plasma Mode** : 이것이 선택 되면 M3, M5 명령어가 생성 되어 아크/레이저가 자르기 하는 동안 ON, OFF 하게끔 한다. 만약 선택 되지 않는다면 스피들 이 파트 프로그램의 처음에서 시작해서 톨 교환 할 때나 혹은 프로그램의 끝에서 멈춘다.

**Connection To**. 같은 레이어의 두개의 선이 두 개의 양 끝 지점이 지정한 거리 안에 있을 경우 연결된 것으로 간주 한다. 이것은 Rapid Plane 으로 움직이지 않고 선의 중간을 매김 함으로써 커팅을 한다.

**Rapid Plane** : 이 컨트롤은 Z 값을 정의 하여 그림의 개체 사이를 빨리 움직일 수 있도록 적용시킨다.

**Lathe mode** : 이것이 선택 되면 드로잉(그림)의 수평(플러스 X)방향이 G 코드에서 Z 로 코딩 되고 수직(플러스 Y)방향이 마이너스 X 로 코딩 되어 그림의 수평축에 그려진 파트의 아웃 라인이 Mach3Turn 에서 정확하게 절삭 되도록 한다.

### 8.2.4 G 코드 생성

마지막으로 Generate G-code 를 클릭하여 스텝 4 를 실행하게 한다. 보통 생성된 G 코드는 .TAP 확장자로 파일을 저장하지만 이것은 반드시 그런 것은 아니며 Mach3 는 자동으로 확장자를 만들지 않는다.

스텝 2 에서 4 까지 반복하거나 1 에서 4 까지 반복해 보고 끝났으면 Done 을 클릭한다.

Mach3 는 생성된 마지막 G 코드 파일을 로딩한다. 이름과 작성한 날짜 에 관한 코멘트를 참고 하기 바란다.

주의:생성된 G 코드는 불러들여진 레이어에 따라 피드비율을 갖는다. 만약 스피들이 S 문자에 반응하지 않으면 스피들 속도를 수동으로 설정하고 톨 교환 할 때 속도를 바꾼다.

- DXF 입력은 간단한 형태에 좋으며 이것은 단지 입력 파일을 만들 기초적인 CAD 프로그램을 요구하며 만든 그림에 입각하여 정밀하게 작동 된다.

- DXF 는 사용할 톨의 직경이 매우 작을 때 레이저나 플라즈마 커팅에 적합하다.

- 밀링에 있어서, 커터의 직경에 관한 범위를 수동으로 직접 만들어야 한다. DXF 선들은 커터의 중심선의 경로가 된다. 하지만 이것은 복잡한 형태의 커팅을 할 경우 수월하지는 않을 것이다.

- DXF 파일에 의해 형성된 프로그램은 파트를 황삭 하거나 포켓의 중심을 깔끔하게 할 다중 경로를 가지고 있지 않다. 이것을 자동으로 해결하기 위해서는 CAM 프로그램을 사용해야 한다.

- DXF 파일이 텍스트를 포함한다면 작성한 프로그램에 따라 크게 두 가지 형태가 된다. 문자가 연속적 라인으로 되는 경우. 이것은 Mach3 로 불러들이기가 가능하다. 문자들이 DXF 텍스트 대상물 일 경우. 이런 경우는 무시 된다. 돌출 어는 경우도, 비록 글자체 가 작은 v-point 나 혹은 둥근날 커터에 맞다고 해도, 초기 드로잉(그림) 에서 사용된 글자체로 조각 하는 G 코드를 만들지 못한다. 플라즈마나 레이저는 글자의 외각선을 자를 수 있을 만큼 폭이 좁다. 그렇지만 0 이나 A 와 같은 문자는 자르기 전에 어떻게 할 것인지 확인을 해야 한다.

### 8.3 HPGL 불러오기

HPGL 파일은 하나나 그 이상의 펜으로 그려진 선을 가지고 있다. Mach3Mill 은 모든 펜들에 대한 절삭을 한다. HPGL 파일은 대부분의 CAD 소프트웨어에 의해 제작 될 수 있고, 종종 파일 확장자가 .PLT 나 .HPL 이 된다.

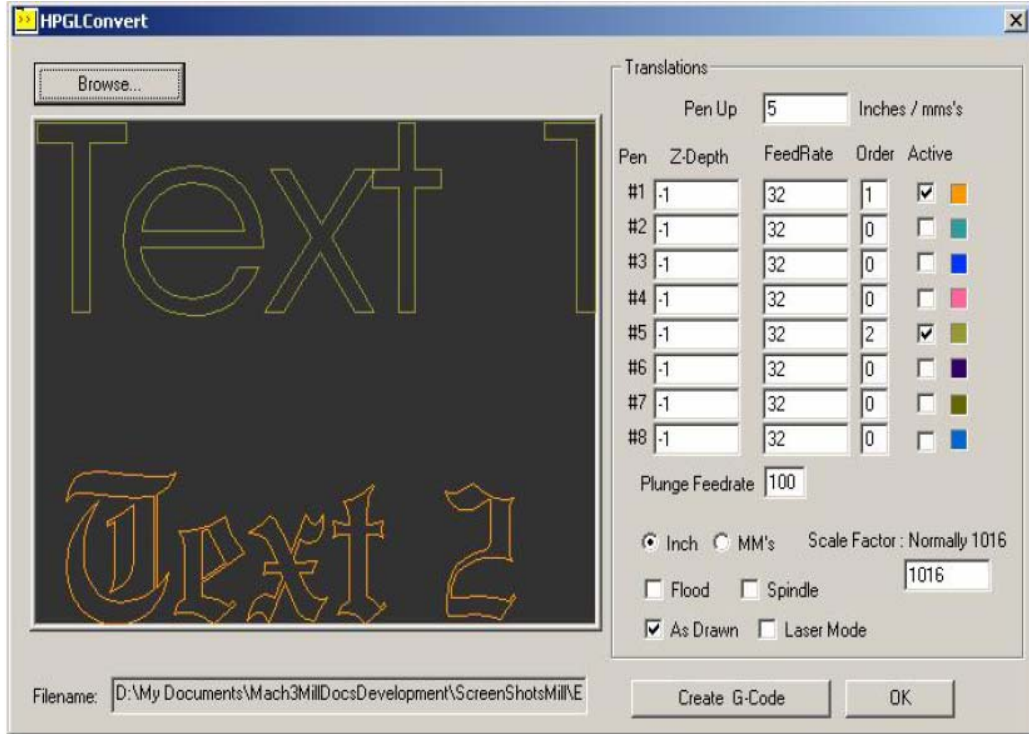


그림 8.4 - HPGL 임포트 필터

#### 8.3.1 HPGL 에 관하여.

HPGL 은 DXF 보다 덜 정밀한 작업을 하며 원을 그릴 때 둥근 모양을 작은 선 단위로 나누어 모양을 만든다. HPGL 의 불러오기 과정은 DXF 파일과 비슷하여 HPGL 로부터 생성된 G 코드를 포함하는 .TAP 파일을 만든다.

#### 8.3.2 불러올 파일 선택 하기.

불러오기(import) 필터는 File>Import HPGL/BMP/JPG 로 들어가서 HPGL 버튼을 on 시킨다. 그림 8.4는 임포트 대화 상자를 보여준다.

처음에는 만들어진 HPGL 파일에 대응하는 Scale 을 선택한다. 대개 40HPGL 단위 당 밀리미터(1016 개체 대비 인치) 가 된다. 이것을 다른 HPGL 포맷에 맞게 변화 시키거나 G 코드 파일에 비례하게 한다. 예를 들어, 40 보다 20 을 선택하는 것은 정의된 개체의 크기를 두배가 되게 한다.

자 이제 HPGL 데이터가 포함된 파일 이름을 넣거나 브라우저를 사용한다. 이 기본 확장자는 .HPG 가 되어 파일 이름을 만들기 편리하게 한다.

### 8.3.3 파라미터 불러오기

Pen up, Pen down 컨트롤은 현재 Mac3 가 사용하는 단위에서 동작할 때 Z 값을 나타낸다. Pen Up 은 일반적으로 작업물의 바로 위에 위치하며 Pen Down 은 원하는 절삭 깊이를 나타낸다.

각기 다른 깊이의 커팅 과 피드 비율은 그림을 그리기 하는 펜에 따라 프로그래밍 될 수 있다. 또, 어느 것을 먼저 커팅 할 것인지 순서를 정의 할 수도 있다. 그러므로 대상물의 최 외곽을 절삭해 내기 전에 내부를 먼저 커팅 할 수 있게 한다.

Check only for laser table 이 설정되면 G 코드는 Pen Down Z 위치로 움직이기 전에 M3(시계 방향으로 스피ن들을 회전)을 실행시키고 Pen Up 위치로 움직이기 전에 M5(스핀들 정지)를 실행시킨다.

주어진 Feedrate 생성된 G 코드가 시작할 때 F 문자가 입력될 것이다.

### 8.3.4 G 코드 파일 작성하기.

마지막으로, 불러들이기 해석이 정의된 상태에서 Import File 을 클릭하여 Mach3Mill 로 데이터를 불러들이기 한다. 생성된 코드를 저장할 파일 이름을 표시하라는 메시지가 나올 것이다. 원하는 파일이름과 확장자를 넣는다. 일반적으로 확장자는 .TAP 가 된다.

주의:

- 불러오기(import) 필터는 Mach3 가 멈춰 있을 때나 사용중인 필터 프로그램에 의해 진행된다. 만약 실수로 클릭 하여 Mach3Mill 화면으로 전환되면, 순간 락업(lock up) 될 것이다 이때 간단하게 Windows 작업 관리자(Cntl + Alt + Del 키)로 가서 필터 프로그램을 다시 작동하게 하여 작업을 끝낸다. 이것은 파트 프로그램이 진행될 때의 편집기와 비슷한 상황이 된다.

- .TAP 파일이 이미 있고, Mach3 에 열려 있다면 불러오기 필터는 같은 파일을 쓸 수 없다. 예를 들어, 불러오기를 테스트 한 파일이 있고 다시 불러오기 하여 해석에 변화를 주려고 한다면 이때 Mach3Mill 에 있는 같은 .TAP 파일을 다시 불러오기 하기 전에 닫아야 한다.

- 일반적으로 HPGL 파일을 불러오기 하는 전 과정에서 미터법을 쓰는 것이 가장 쉽다.

- Laser Table 옵션으로 레이저나 플라즈마를 사용한다면 M3/M5 의 순서를 확인해야 하며 Z 방향이 절삭의 시작과 끝이 정확하게 범위안에 있는지 확인 해야 한다.

- 밀링에 있어서 커터의 직경에 관한 범위를 수동으로 직접 만들어야 한다. HPGL 선들은 커터의 중심선의 경로가 된다. 하지만 이것은 복잡한 형태의 커팅을 할 경우 간단하지 않을 것이다.

- HPGL 파일에 의해 형성된 프로그램은 파트를 황삭 하거나 포켓의 중심을 깔끔하게 할 다중 경로를 가지고 있지 않다. 이것을 자동으로 해결하기 위해서는 CAM 프로그램을 사용해야 한다.

## 8.4 BMP&JPEG 파일 불러오기

이 옵션은 사진을 불러오기 해서 G 코드 프로그램을 생성하고 절삭 깊이를 다르게 하여 명암을 렌더링 한다. 결과물은 실제 사진과 같은 조각을 할 것이다.

### 8.4.1 불러오기 할 파일 선택 하기.

불러오기 필터는 File>Import HPGL/BMP/JPEG 에서 실행되며 JPEG/BMP 버튼을 대화 상자에서 On 시킨다.

첫 단계는 Load image file 버튼을 눌러 이미지가 있는 파일을 선택한다. 파일이 로딩되면 작업대상의 범위를 이미지에 맞게 선택하게 한다. 생성된 파트 프로그램을 진행 할 때 필요한 G20/21 모드에 따라 메트릭이나 인치를 사용해도 상관없다. 그림 8.5는 이 대화상자를 보여 준다.

Maintain Perspective 를 체크하면 X 사이즈 값을 정할 때 자동으로 Y 크기를 정하고 원본 사진의 비율에 맞게 한다.

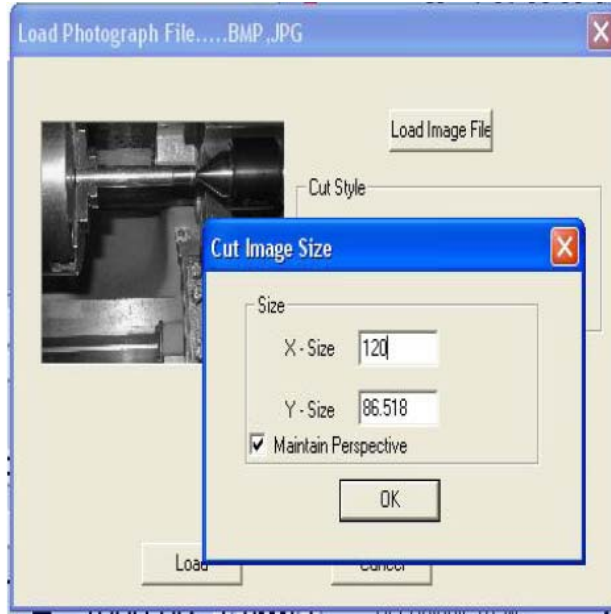


그림 8.5 임포트한 사진의 사이즈.

반대의 경우도 마찬가지다. 이미지가 칼라면 불러오기 할 때 흑백으로 전환 된다.

### 8.4.2 렌더링 타입 선택 하기

다음으로 이미지를 렌더링 할 방법을 선택한다. 이것은 이미지를 레스터화(rasterize) 하면서 톨의 경로를 정의 내린다. Raster X/Y 는 X 축을 따라 커팅 하며 끝나면 X의 한줄이 끝날 때 마다 하나씩 Y 축으로 옮긴다. Raster Y/X 는 위의 경우와 반대가 된다. Spiral 은 이미지의 외곽선에서 시작 하여 나선을 그리며 중심으로 움직인다. 각 각의 레스터(raster)라인은 Z 좌표의 높이와 일련의 직선으로 구성 된다.

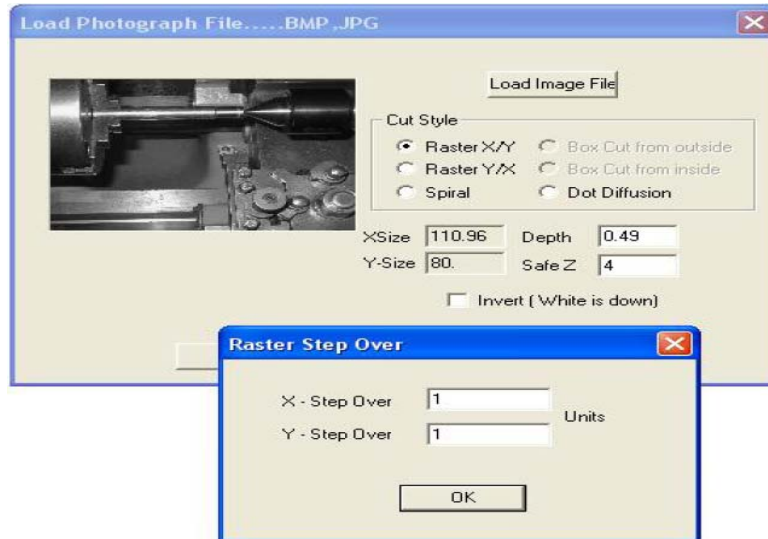


그림 8.6 Step Over 정의 하기

### 8.4.3 레스터(raster)과 나선(spiral)형 렌더링.

래스터 방식 중 하나를 선택함에 따라 대화 상자에서 step-over 값을 정의하라고 나올 것이다. 그림 8.6 참조. 이렇게 하여 선들의 간격과 그 선을 구성하는 작은 점들 사이의 간격을 정의 내린다. 전체 움직임 수는  $(X\ Size / X\ Step\ Over) \times (Y\ Size / Y\ Step\ Over)$  가 되고 물론, 물론 물체 사이즈의 제곱으로 증가 하고 스텝 오버(step over) 사이즈의 역제곱으로 증가한다. 가능한 적당한 크기의 해상도로 시작해야 파일 사이즈가 너무 크게 되거나, 커팅 시간이 아주 길게 되는 것을 방지 할 수 있다.

### 8.4.4 점 발산 렌더링(Dot diffusion Rendering)

점 발산 렌더링 방식을 선택 하면 구체적인 내용에 대한 입력이 요구된다. 점 발산은 일반적인 격자(grid) 안에서 일련의 드릴링 한 점으로 나타내 진다. 일반적으로 V-포인트나 혹은 둥근 모양(bull-nosed) 툴에 의해 형성된다. 각 점들의 깊이는 이미지의 명암에 따라 결정된다. 영역에 필요한 점들의 숫자는 필터에 의해 계산되며 이때 툴의 형태와 선택한 조각의 깊이에 따라 결정된다. 그림 9.7은 필요한 데이터를 보여준다. 각각의 점들은 Z의 움직임으로 구성되는데 이때 Z가 들어가는 깊이와 작업을 위에서의 움직임을 가리킨다. 점 발산에 따른 계산 로딩시간을 컨트롤 하기 위해 적절한 픽셀 수를 만들 수 있는 사진 편집기로 제작된 이미지가 있어야 한다. Check Stats 버튼을 눌러 얻어진 통계는 얼마큼 적절한 파라미터를 설정했는지 알려 준다.

자 이제 설정한 렌더링 테크닉을 정의함으로써 작업물의 위로 움직이는 Safe Z는 완성 될 것이며 흑과 백의 가장 깊어야 할 기준 지점을 선택 하게 해준다.

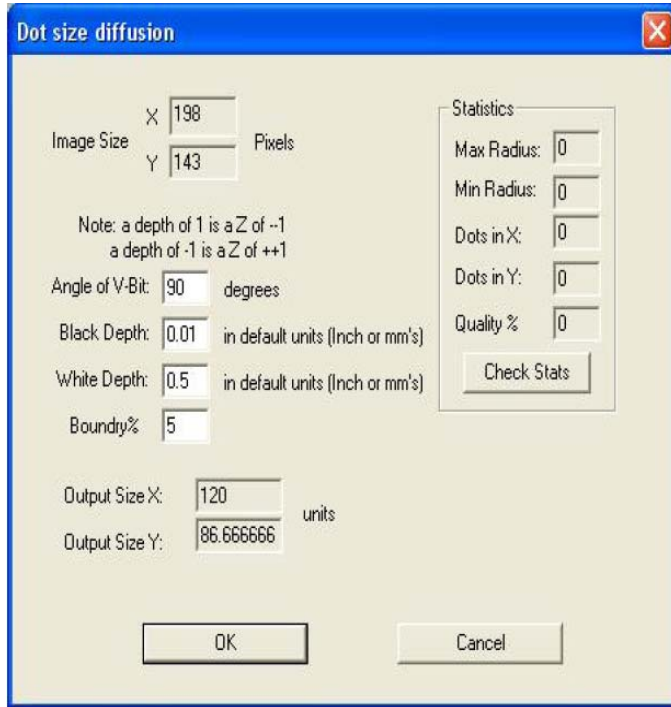


그림 9.7 점 발산 파라미터

### 8.4.5 G 코드 파일 작성하기

마지막으로 Convert 를 클릭하여 Mach3Mill 에 데이터가 불러오기 된다. 사용할 파일 이름을 정하고 생성된 코드를 저장 하게 된다. 파일이름과 확장자를 작성한다. 일반적으로 확장자는 .TAP 가 된다.

주의 :

- 불러오기 필터는 Mach3 가 멈춰 있을 때나 사용중인 필터 프로그램에 의해 진행된다. 만약 Mache3Mill 화면을 전환하면, 예를 들어, 실수로 클릭한다고 할 때, 순간 락업(lock up) 될 것이다 이때 간단하게 Windows 작업 관리자로 가서 필터 프로그램을 다시 작동하여 작업을 끝낸다. 이것은 파트 프로그램이 진행될 때의 편집기와 비슷한 상황이 된다.

- .TAP 파일이 이미 있고 Mach3 에서 열려 있다면 불러오기 필터는 같은 파일을 쓸 수 없다. 예를 들어, 불러오기를 테스트 한 파일이 있고 다시 불러오기 하여 해석에 변화를 주려고 한다면 이때 Mach3Mill 에 있는 같은 .TAP 파일을 다시 불러오기 하기 전에달아야 한다.
- 사용할 피드비율은 MDI 를 사용하거나 진행하기 전에 파트 프로그램을 편집하여 정의 내린다.
- 점 발산은 Z 축을 많이 움직이게 한다. 반드시 Safe Z 를 가능한 낮게 설정하여 움직이는 거리를 최소화 하고 Z 축 모터의 회전이 매우 안정적이게 설정한다.

## 9. 커터 보정(Cutter Compensation)

커터 보정은 Mach3의 한 기능으로 아마도 사용 할 일이 없을 수도 있다. 대부분의 CAD/CAM 프로그램은 근소한 차이의 톨 직경을 입력 받아 파트 프로그램을 출력해서 포켓이나 파트의 외곽을 커팅 하게 된다. 때문에 CAD/CAM 소프트웨어는 전반적으로 Mach3보다 더 낫은 기능들이 있어 예각의 안쪽 코너를 자를 때 역시 더 잘 처리 한다.

Mach3에서, 보정은 a) 프로그램된 것으로부터 다른 직경의 톨을 사용하거나 혹은 b) 톨의 중심 경로 보다는 원하는 외곽에 관한 설명이 있는 파트 프로그램 사용 하게 할 것이다. 그러나 보정이 사소한 것이 아니기 때문에, 이 장에서 설명하여 사용하게 될 경우를 대비 하는 것이다. 이 기능은 개발 단계에 있으며 Mach3의 최종 버전에, 많은 변화가 있을 수 있다는 것을 말하고 싶다.

### 9.1 보정에 관한 소개.

과거 Mach1은 제어점의 움직임을 컨트롤 해 왔다. 실전에서는, V 모양 커터를 제외하고, 톨은 포인트 점이 아니기 때문에, 커터의 반경에 따라 제어점 대비 다른 위치에서 커팅을 하게 된다. 가장 손쉬운 방법은 CAD/CAM을 이용하여 포켓이나 형체의 외곽을 절삭하는 것이다. Mach3는 커터의 반경(직경) 계산에 대해 지원한다.

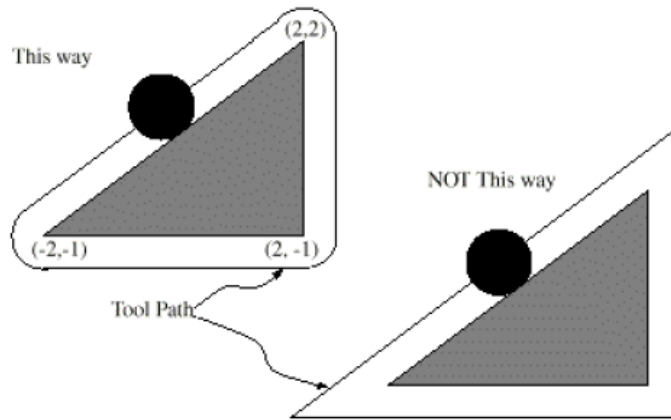


그림 9.1 삼각형을 절삭 하는 두가지 방법의 톨 패스

보정은 다른 파트 프로그램의 생산에서 요구 되기 보다는, 장비 작업자에 의해 가능해 진다.

이것을 알면 문제는 쉽게 풀릴 수 있다. 해야할 것은 단지 X,Y에 의한 제어점에 옵션을 주는 것이며 이것으로 톨의 반경을 허용 하게 된다. 간단한 삼각법으로 거리를 구할 수 있는데 이것은 각도와 축에서 절삭하는 방향에 따라 결정된다.

실전에서 이런 것들은 그리 쉬운데 아니다. 몇가지 문제가 있을 수 있지만 주요 문제는 장비는 커팅이 시작 되기 전에 Z 위치가 설정 되어야 하는데 그때 톨이 어느 방향으로 움직일지 모른다는 것이다. 이 문제는 사용하지 않는 파트를 이용하여 사전 움직임을 시행 하므로써 해결 할 수 있다.

이것으로 실제 파트의 외곽선이 커팅 되기 전에 보정 계산이 이루어 질 수 있다. 파트의 외곽선 에서 부드럽게 작동 하는 파트의 선택이 또한 표면의 마무리를 최적화 하게 한다. 종료 동작이 때로는 마지막 커팅에서 사용 되어 마무리가 잘 될 수 있도록 한다.

## 9.2 두 종류의 윤곽선(Contour)

Mach3는 두가지 타입의 윤곽선을 위한 보정을 운영할 수 있다.

- 파트 프로그램 코드의 윤곽선은 절삭 되어서는 안될 대상물의 표면이다. 이런 타입을 “ 재료의 경계 윤곽선” 이라고 명명 한다. 이런 부류의 코드는 대개 직접 작성된 것들이다.

- NC 코드에 주어진 윤곽선은 정확한 톨 반경에 의해 진행될 톨 경로가 된다. 이런 타입을 “ 톨 패스 윤곽선” 이라 부른다. 의도 하는 커터의 직경을 알고 있다면, 이런 종류의 코드는 CAD/CAM 프로그램이 만들 수 있는 것이다.

중간 해석 프로그램은 어떤 타입의 윤곽선을 사용할지 결정 할 수 있는 설정기능을 가지고 있지 않다. 하지만 같은 파트의 지형에서 윤곽선의 숫치적 기술은 물론 두 타입 사이를 다르게 만들 것이며 톨 테이블 에서 직경 값들은 두 타입에게 다를 것이다.

### 9.2.1 재료의 경계 윤곽선

대상물의 표면이 윤곽선이 된다면, 표면의 외곽선은 파트 프로그램에서 설명 되어진다. 재료의 경계 윤곽선에서 톨 테이블에 있는 직경 값들은 실제 톨의 직경 값이 된다. 값들은 반드시 플러스 값이 된다.

#### 예제 1:

위의 그림 9.1 과 같이 삼각형의 외곽을 깎는 NC 프로그램이 있다. 이 예제에서, 커터 보정 반경은 실제 사용하는 톨의 반경으로 0.5 값이 된다. 톨 테이블의 직경 값은 반경 의 두배가 되니 1 이다.

N0010 G41 G1 X2 Y2 (보정을 작동 시키고 초기 움직임 시작)  
N0020 Y-1 (삼각형의 오른쪽 면을 따라 간다.)  
N0030 X-2 (삼각형의 아래 면을 따라 간다.)  
N0040 X2 Y2 (삼각형의 빗변을 따라 간다.)  
N0050 G40 (보정 중지)

이것의 결과는 경로를 따라 초기 움직임을 시작하여 시계 방향으로 따라 가는 것을 보여 준다. 알아 들것은 대상물의 삼각형 좌표는 NC 코드에 나타난다. 또 하나는 톨 패스가 3개의 호를 포함하고 있으며 이것은 프로그램 된 것이 아니다. 단지 자동으로 생성된 것이다.

### 9.2.2 톨 패스 윤곽선

윤곽선이 톨패스 윤곽선이 된다면, 패스(경로)는 파트 프로그램에 기술 되어 진다. 초기 움직임을 제외하고, 패스( 경로) 가 일부 파트 지형을 만들것으로 예상된다. 경로는 원하는 의도에 따라 파트의 지형을 고려하여 수동으로 생성 되거나 혹은 CAD/CAM으로 생성한다. 그림 9.1의 왼쪽 에서 보여지는 것처럼, Mach3 작업에서 톨 패스는 파트 지형의 표면과 접촉이 된다. 그림 9.1 오른쪽 과 같이 톨패스가 파트 지형에 항상 접촉 하지 않는다면 해석프로그램은 작은 사이즈의 톨을 이용하게 될 때 적절히 보완 못할 것이다.

톨 패스 윤곽선에서, 톨 테이블에서 커터의 직경 값은 선택한 톨이 약간 클때 약간의 양수가 될 것이며 약간 작을 경우 약간의 음수가 될 것이다. 적용된 것과같이 커터의 직경 값이 음수 이면 해석 프로그램은 프로그램 된 윤곽선의 반대쪽으로 보정을 하게 하며 주어진 직경의 절대값을 사용 하게 된다. 실제 톨이 정확한 사이즈라면 테이블 상의 값은 0 이 된다.

#### 톨 패스 윤곽선 예제:



커터의 직경이 현재 스피들에서 0.97 이라고 가정하자. 그리고 톨패스에서 생성 하는 직경은 1.0 이라고 하자. 그때 톨 테이블의 직경 값은 -0.03이 된다. 여기 아래에 삼각형의 외각선을 조각 하는 NC 프로그램이 있다.

N0010 G1 X1 Y4.5 (조정 동작)  
N0020 G41 G1 Y3.5 (보정을 작동 시키고 초기 움직임 시작)  
N0030 G3 X2 Y2.5 I1 (두번째 움직임 시작)  
N0040 G2 X2.5 Y2 J-0.5 (톨패스의 맨위에서 호를 따라 커팅)  
N0050 G1 Y-1 (톨패스의 오른쪽을 따라 커팅)  
N0060 G2 X2 Y-1.5 I-0.5 (톨패스의 오른쪽 아래 호를 따라 커팅)  
N0070 G1 X-2 (톨 패스의 아래면을 따라 커팅)  
N0080 G2 X-2.3 Y-0.6 J0.5 (톨패스의 왼쪽 아래 호를 따라 커팅)  
N0090 G1 X1.7 Y2.4 (톨패스의 빗면을 따라 커팅)  
N0100 G2 X2 Y2.5 I0.3 J-0.4 (톨패스의 맨위에서 호를 따라 커팅)  
N0110 G40 (보정 중지)

이것은 교정 동작을 시작으로 두번의 초기 움직임을 만들고 다음 패스의 약간 안쪽에서 따라 간다. 그림 10.1 과 같이 시계 방향으로 진행 된다. 이 경로는 비록 G41이 프로그램 되어 있어도 프로그램된 경로는 경로(패스) 의 오른쪽에 있게 된다. 왜냐하면 직경 값이 음수 이기 때문이다.

### 9.2.3 진입 동작 프로그래밍

일반적으로 정렬 동작과 진입 동작은 정확한 보정을 시작하기 위해 필요하다. 톨은 진입동작이 시작 되기전 마무리 절삭 으로부터 적어도 직경정도의 거리가 필요하다.

## 10. Mach3 G, M 코드 참고

---

이번 섹션에서는 Mach3 에서 이해하고 해석 되는 언어(G코드)에 대해 정의 한다.  
장비를 위한 일부 NIST NMC(Next Generation Controller) 아키텍처 기능들은 여기 Mach3에 적용되지 않아 이렇게 회색으로 처리 되었다. 이런 기능들이 여러분의 어플리케이션에 중요하다면 ArtSoft 주식회사에 알려 주길 바라며 그러면 다음 개발 계획에 참고 할 것이다.

### 10.1 용어 정의

#### 10.1.1 선형 축

X, Y 그리고 Z 축은 직교 하는 축의 표준 오른손 좌표계로 구성된다. 3차 선형운동 메커니즘의 위치는 이 축들의 좌표를 이용하여 기술 된다.

#### 10.1.2 회전 축

회전 축은 X, Y 혹은 Z 축의 플러스 방향에서 보았을때 반시계 방향으로 선형축을 말려 가듯 하므로써 각도로 표시가 된다. 여기서 “ 선형 축을 말려 가듯” 이란 뜻은 축이 반시계 방향으로 회전 함에 따라 각의 위치가 무한정 증가 할 때(플러스 방향으로 무한정 갈때) 와 시계 방향으로 회전 함에 따라 무한정 감소 할 때(마이너스 방향으로 무한정 갈때)를 말한다. 말린 선형 축들은 기계적 회전 제한이 있고 없음을 떠나 사용 된다. (이 말은 회전 하는 물체의 위에서 보는 것이 아니라 측면 관점에서 본다면 이해가 가능해 진다.)

시계 방향 혹은 반시계 방향 은 작업물의 관점에서 보는 것이다. 작업물이 회전판에 고정되어 회전축을 따라 돌아 간다면, 반 시계 방향 회전은 작업물의 관점에서 볼때 회전판이 돌아 감에 따라 장비 주위의 사물들은 시계 방향으로 돌아 가는 것 처럼 보이게 된다.

#### 10.1.3 스케일 입력(scaling input)

각 축에 있어서 스케일 설정은 가능하다. 이것은 X, Y, Z A, B, C, I, J 그리고 R 문자의 값들이 입력 될 때마다 적용된다. 또한 가공된 형상의 크기 조절과 이미지 미러링(mirroring)을 가능하게 한다. 스케일 조절은 값들로 적용되며 피드비율(feedrate)과 같은 것들은 항상 비율조절 값을 기초로 한다.

고정, 툴 테이블에 저장된 옵션은 사용하기 전에는 비율조절 되지 않는다. 비율조절은 물론 값이 입력 됐을 때 적용된다.

#### 10.1.4 제어점(controlled point)

제어점은 위치와 모션 비율이 제어된 지점을 말한다. 툴 길이 옵션이 0(기본값)일때 이 지점이 스피들 축에서(종종 게이지 포인트라고 불리움) 스피들 끝 위로 일정하게 고정된 거리가 있으며 주로 스피들틀에 꼭 맞물리는 툴 홀더의 끝부분 근처에 있다. 제어점의 위치는 툴 길이 옵션값을 지정함으로써 스피들 축을 따라 움직일 수 있다. 값의 크기는 대체로 사용 중인 커팅 툴의 길이가 되어 제어점이 커팅 툴의 끝점이 되게 한다.

### 10.1.5 선형운동의 조화

지정된 경로로 툴을 움직이기 위해, 미시닝 장비는 여러 축을 조화롭게 움직여야 한다. 여기서 “ 선형운동의 조화” 란 말은 다음의 내용을 기술 하기 위해 쓰여졌다.

대개, 각 축이 일정한 속도로 움직이고 모든 축이 처음과 끝이 같게 움직이는 것으로서 “ 선형” 이란 단어를 써서 직선 운동을 할 수있게 한다. 그러나 실제 동작에 있어서는 일정한 속도를 유지하는 것은 종종 불가능 한데 그것은 가속, 감속이 운동의 처음과 끝에서 필요 하기 때문이다. 축 제어는 필요한 거리를 각각의 다른 축에도 똑 같이 움직이게 한다. 이 동작을 통해 툴을 같은 경로를 움직이게 하고, 그러므로 이런 동작을 선형운동의 조화 라고 부른다.

선형운동의 조화는 피드비율이나 혹은 빠른 왕복운동 비율(rapid traverse rate)에서 수행된다. 축 속도의 물리적 한계로 인해 원하는 속도비율이 나오지 않는다면, 모든 축이 가능한 속도로 움직여 원하는 패스(경로)를 유지 하도록 한다.

### 10.1.6 피드비율 Feed Rate

제어점에서의 비율이나 축의 축의 움직임은 일반적으로 사용자의 설정에 의해 일정한 비율을 갖는다. 해석 프로그램에서 피드비율의 해석은 다음과 같다. 단 G 93 모드 가 사용 되지 않을 때이다.

- 하나 이상의 선형 축들의 움직임에 관련하여, (X, Y, Z 그리고 추가적으로 A, B,C), 회전축 동작이 없는 경우, 피드비율은 프로그램된 선형 XYZ(ABC) 경로를 따르는 속도(milimeter,inch/min)를 의미 한다.

- 하나 이상의 선형 축들의 움직임에 관련하여, (X, Y, Z 그리고 추가적으로 A, B, C), 회전축 동작이 있는 경우, 피드비율은 파이(pi =3.1415..) 곱하기 축의 직경(Correction Diameter) 곱하기 회전축의 각속도 조합하여 프로그램된 선형 XYZ(ABC) 경로에 따른 속도(inch,millimeter/second)를 의미한다.

- 하나의 회전 축 운동이 있고 X,Y,Z축이 정지되어 있다면 피드비율은 회전축의 회전각도당 분을 의미한다.

- 하나 혹은 두개의 회전 축과 XYZ 축이 정지 되어 있다면 피드비율은 다음과 같이 적용된다. dA, dB 그리고 dC를 각도로 놓고 여기서 A,B,C 축들을 통하여 움직인다. D 는 루트(dA제곱 + dB제곱 + dC제곱)으로 놓는다. 개념적으로 D는 토탈 회전 동작의 측정이 되고 이때 유클리드의 미터법을 사용한다. T 를 각도당 분을 사용 하는 현재 피드비율에서 D각도 만큼 움직일 때 필요한 시간으로 놓는다. 회전 축은 선형 운동 조화로 움직여야 하므로 모션의 시작과 끝으로부터 경과한 시간은 T 에 감, 가속에 필요한 시간을 더한 값이다.

### 10.1.7 호 (Arc Motion)

각각 면을 구성하는 선형 축들이(XY, YZ, XZ) 면에서 원의 호를 따라 움직일 수 있도록 제어한다. 이렇게 되는 동안, 세번째 선형 축과 회전 축이 동시에 일정한 비율로 효율적으로 움직이게 제어 한다. 좌표 선형 운동에 있어 운동이 제어됨에 따라 감속, 가속이 경로에 영향을 미치지 못하게 한다.

회전축이 움직이지 않으면, 세번째 선형 축이 움직이지만, 주어진 제어점은 나선이 된다.

호 운동 하는 동안 피드비율은 먼저 번의 Feed Rate에 기술했었다. 나선 운동의 경우, 비율(rate)은 나선을 통해 적용된다. 시스템에 따라 해석됨이 다른 것에 유의 하기 바란다.

### 10.1.8 냉매제(coolant)

유체, 기체 콜러트(냉각제)는 독립적으로 작동 시킬 수 있다. OFF 될때 같이 꺼진다.

### 10.1.9 머무르기 (Dwell)

머시닝 시스템은 일정 시간 동안 잠시 머무르기 하도록 명령 받을 수 있다. 머무르기를 필요로 하는 곳은 대부분 깨끗하게 칩들을 제거하나 혹은 스피들의 속도를 일정하게 올릴 때다. 머무르기에 지정하는 단위들은 초나 혹은 밀리초가 되며 Configure > Logic 의 정의에 따른다.

### 10.1.10 단위 (Units)

X, Y 그리고 Z 축에 사용 되는 거리의 단위는 밀리미터나 인치로 사용 될 수 있다. 사용중인 머신 컨트롤에서는 단위를 바꿀 수 없다.

스핀들 속도는 분당 회전수로 측정이 되고 회전축의 위치는 각도로 측정된다. 피드비율은 현재 길이 단위 대비 분으로 표시 되거나 각도당 분으로 먼저 언급 한 바와 같이 표시 된다.

**경고** : 툴과 고정체가 테이블에 놓여 있을 때나, 파트 프로그램이 실행 되는 동안 오프셋이 적용 될때 단위를 바꾸는 것에 대해 장비가 어떻게 반응하는지 잘 관찰 하기 바란다.

### 10.1.11 현재 위치

제어점은 항상 어떤 위치를 현재 위치라고 부르고 Mach3는 항상 그것이 어디인지 기억하고 있다. 현재 위치를 표시하는 수치가, 축이 움직이지 않는 동안 다음과 같은 경우, 조정 된다.

- 길이 단위가 바뀔때
- 툴 오프셋 이 바뀔때
- 좌표계 오프셋이 바뀔때

### 10.1.12 선택된 좌표면(plane)

항상 선택된 좌표면은 있기 마련이며 이때 XY 면 YZ 면 혹은 XZ 면이될 것이다. Z 축은, 물론 XY 면에 수직이 되고 X 축은 YZ면에 Y 는 XZ 면에 수직이 된다.

### 10.1.13 툴 테이블(Tool Table)

툴 테이블에 있는 각 슬롯에 지정된 툴은 1 혹은 0 이다.

### 10.1.14 툴 교체(Tool change)

Mach3는 필요에 따라 손으로 직접 툴을 교체 할 수도 매크로를 사용하여 자동적으로 툴 교체를 할 수도 있다.

### 10.1.15 팔렛 셔틀(Pallet Shuttle)

Mach3 는 매크로를 이용하여 팔렛 셔들을 사용 할 수 있게 한다.

### 10.1.16 경로 제어 모드(Path Control Modes)

머시닝 시스템에는 두가지중 하나의 경로 제어 모드를 사용할 수 있다. 1) 정확한 정지 모드(exact stop mode) 2) 등속 모드(constant velocity mode). 정확한 정지 모드에서, 머신은 프로그램된 동작의 끝에서 정확하게 멈춘다. 등속 모드에서는, 각진 코너에서 살짝 라운딩 하여 피드비율을 유지 할 수있도록 한다. 이런 모드들은 사용자에게 코너회전을 적절하게 제어 할 수 있게 하는데 실상에서 장비는 관성력에 기인 하여 유한한 가속도를 가지기 때문이다.

Exact stop(정확한 정지) 은 말 그대로다. 장비는 방향이 바뀔 때 마다 순간 멈출 것이고 툴은 명령받은 경로를 정확히 따라 갈 것이다.

constant velocity (등속운동)은 새로운 방향으로 진행할 때의 가속과 현재의 방향에서 감속할 때 중복되어 명령된 피드비율을 유지한다. 이것은 어떤 코너 라운딩도 빠르고 부드러운 커팅을 유도한다. 때문에 플라즈마나 라우터 커팅에 중요 하다.

장비축의 가속도를 낮출수록 라운딩 되는 코너의 반경은 커진다.

플라즈마 모드에서( Configure Logic 대화 상자에서 설정) 시스템은 독립된 알고리즘에 의한 플라즈마 커팅을 위해 코너동작을 최적화 하려 할 것이다.

각도를 제한 하는 것도 가능하여 제한된 각을 넘어 갈 때, 비록 등속(Constant Velocity) 이 선택되어도, Exact Stop(정확한 정지)로 간주하도록 정의 할 수 있다. 이것은 완만한 코너를 더욱 부드럽게 하지만 하나 이상의 축에서 저 가속 일때 날카로운 코너에서의 과도한 라운딩을 피한다. 이 기능은 Configure Logic 대화상자에서 가능하게 하며 각도의 제한은 DRO에서 설정한다. 이 설정은 장비의 특성이나 각 작업의 톨패스에 따라 경험적으로 설정 한다.

## 10.2 제어를 통한 해석 프로그램의 상호 작용

### 10.2.1 피드와 속도를 번복하기

Mach3의 피드와 속도를 번복하는 스위치 명령을 통해 활성화(M48) 또는 비 활성화(M49) 할 수 있다. 일부 머시닝 운영을 위해서 이런 제어 스위치들을 유용하게 사용 할 수 있다.

이것을 실행 하는 이유는 최적의 설정이 프로그램에 내장되어 있다면 작업자가 그것들을 바꾸지 않도록 하기 위함이다.

### 10.2.2 블록 삭제 제어 (Block Delete control)

이 옵션이 On 되어 있으면, " /" (슬래시)로 시작하는 코드 라인들은 실행이 되지 않는다. Off 되어 있다면 그런 라인들은 실행을 하게 된다.

### 10.2.3 옵션 프로그램 정지 제어 (Optional Program Stop control)

옵션 프로그램 정지 제어(Config> Logic)는 다음과 같이 작동된다. 이 컨트롤이 On 되어 있다면 그리고 입력 라인에 M1 코드가 포함되어 있다면 프로그램 실행은 Cycle Start 버튼을 누르기 전까지는 해당 라인의 명령어의 끝에서 정지된다.

## 10.3 톨 파일(Tool File)

Mach3 는 256가지의 톨을 사용 할 수 있는 톨파일을 관리 한다.

파일의 데이터 라인은 하나의 톨을 위한 정보를 가지고 있다. 이것은 톨 길이, 직경, 팁 반경 등에 대한 정보를 정의 할 수 있게 한다.

## 10.4 파트 프로그램의 언어

### 10.4.1 개략

언어는 라인의 코드에 기초한다. 각 라인(블록: block 이라고 부름)은 명령어를 포함하여 장비 시스템에서 다른것들을 수행 하기도 한다. 코드 라인들은 프로그램을 만들기 위해 파일에 저장되기도 한다.

일반적인 코드 라인은 하나 이상의 단어에 뒤이어 추가적 라인 번호로 구성된다.

단어는 문자로 구성이 되고 숫자가 뒤에 오게 된다. 단어는 명령을 주거나 혹은 명령어에 인수를 제공 하기도 한다. 예를 들어, G1 X3는 두개의 단어로 된 가능한 코드 라인이다. G1의 명령어는 " 프로그램된 피드비율에서 직선으로 움직여라" 라는 뜻이고 X3는 인수 값을 제공한다. 대부분의 명령어들은 G 나 혹은 M 으로 시작 한다(General , Miscellaneous). 이런 명령어 단어들은 G 코드나 M 코드 라고 한다.

언어는 두개의 명령어들로 (M2 나 혹은 M30), 그중 하나가 프로그램을 끝낸다. 프로그램은 파일이 끝나기 전에 끝날 수도 있다. 프로그램의 끝 후에 발생하는 파일의 라인들은 일반적인 흐름으로 실행 되지 않아 일반적으로 서브루틴의 파트가 된다.

Parameter number	Meaning	Parameter number	Meaning
5161	G28 home X	5261	Work offset 3 X
5162	G28 home Y	5262	Work offset 3 Y
5163	G28 home Z	5263	Work offset 3 Z
5164	G28 home A	5264	Work offset 3 A
5165	G28 home B	5265	Work offset 3 B
5166	G28 home C	5266	Work offset 3 C
5181	G30 home X	5281	Work offset 4 X
5182	G30 home Y	5282	Work offset 4 Y
5183	G30 home Z	5283	Work offset 4 Z
5184	G30 home A	5284	Work offset 4 A
5185	G30 home B	5285	Work offset 4 B
5186	G30 home C	5286	Work offset 4 C
5191	Scale X	5301	Work offset 5 X
5192	Scale Y	5302	Work offset 5 Y
5193	Scale Z	5303	Work offset 5 Z
5194	Scale A	5304	Work offset 5 A
5195	Scale B	5305	Work offset 5 B
5196	Scale C	5306	Work offset 5 C
5211	G92 offset X	5321	Work offset 6 X
5212	G92 offset Y	5322	Work offset 6 Y
5213	G92 offset Z	5323	Work offset 6 Z
5214	G92 offset A	5324	Work offset 6 A
5215	G92 offset B	5325	Work offset 6 B
5216	G92 offset C	5326	Work offset 6 C
5220	Current Work offset number		
5221	Work offset 1 X		<i>And so on every 20 values until</i>
5222	Work offset 1 Y		
5223	Work offset 1 Z	10281	Work offset 254 X
5224	Work offset 1 A	10282	Work offset 254 Y
5225	Work offset 1 B	10283	Work offset 254 Z
5226	Work offset 1 C	10284	Work offset 254 A
5241	Work offset 2 X	10285	Work offset 254 B
5242	Work offset 2 Y	10286	Work offset 254 C
5243	Work offset 2 Z	10301	Work offset 255 X
5244	Work offset 2 A	10302	Work offset 255 Y
5245	Work offset 2 B	10303	Work offset 255 Z
5246	Work offset 2 C	10304	Work offset 255 A
		10305	Work offset 255 B
		10306	Work offset 255 C

그림 10.1 시스템에 정의된 파라미터들

#### 10.4.2 파라미터

Mach3 머시닝 시스템은 10,320 숫자 파라미터 배열을 관리한다. 대부분 특정 사용처가 있다. 고정체와 관련된 파라미터는 영구적이다. 다른 파라미터들은 Mach3 가 로딩될때 정의내려지지 않는다. 파라미터들은 해석 프로그램이 리셋 될때 보존된다. Mach3에 의해 정의된 파라미터들의 의미는 그림 10.1을 참조한다.

### 10.4.3 좌표계

머시닝 시스템에는 절대 좌표계와 254 작업 옵셋(고정체) 시스템이 있다. 툴 옵셋은 G10 L1 P~ X~ Z~에 의해 설정 될 수 있다. P 문자는 설정할 툴 번호를 정의한다.

고정물 시스템의 옵셋을 G10 L2 P~ X~ Y~ Z~ A~ B~ C~를 이용하여 설정 할 수 있다.

P 문자는 설정할 고정물을 정의 내린다. X, Y, Z 등등의 문자는 절대 좌표계의 관점에서 축을 위한 원점의 좌표이다.

처음 일곱개의 작업 옵셋을 G54, G55, G56, G57, G58, G59에 의해 설정 할 수 있다. 255 작업 옵셋 중 아무것이나 G59 P~에 의해 선택 될 수 있다. (예 G59 P23은 고정체 23을 선택하게 된다.)

절대 좌표계는 G59 P0로 설정 될 수 있다.

G92 혹은 G92.3으로 현재 좌표계에 옵셋을 줄 수 있다. 이 옵셋은 작업 옵셋 좌표 계의 맨 위에 적용 하게 된다. 이 옵셋은 G92.1나 혹은 G92.2로 취소 시킬 수있다.

문자	의미
A	장비의 A 축
B	장비의 B 축
C	장비의 C 축
D	툴 반경 보정 숫자
F	피드비율
G	일반 기능 (테이블 5 참고)
H	툴 길이 옵셋 인덱스
I	호의 X 축 옵셋 G87 고정 싸이클(canned cycle) 에서의 X 옵셋
J	호의 Y 축 옵셋 G87 고정 싸이클(canned cycle) 에서의 Y 옵셋
K	호의 Z 축 옵셋 G87 고정 싸이클(canned cycle) 에서의 Z 옵셋
L	고정 싸이클 에서의 반복 숫자/G10 으로 사용된 서브루틴 키
M	기타 기능 (테이블 7 참고)
N	라인 번호
O	서브루틴 라벨 번호
P	고정 싸이클에서의 대기 시간 G4 대기 시간 G10 으로 사용한 키
Q	G83 고정 싸이클에서의 피드 증분 서브루틴 호출의 빈도
R	호의 반경 고정 싸이클 의 철회 단계
S	스핀들 속도
T	툴 선택
U	A 와 동의어
V	B 와 동의어
W	C 와 동의어
X	장비의 X 축
Y	장비의 Y 축
Z	장비의 Z 축

그림 10.2 문자의 초기 알파벳

절대 좌표계에서 G53 과 G0 혹은 G53과 G1을 이용하여 직선 운동을 시킬 수있다.

## 10.5 라인 포맷

가용한 입력 코드 라인은 다음과 같은 순서대로 구성되며 라인에 허용되는 문자의 숫자에는 제약이 있고 최대의 숫자는 256 가지이다.

- “ /” 슬레쉬는 라인을 생략 하게 한다.
- 추가적 라인 번호
- 단어의 숫자, 파라미터 설정 그리고 코멘트
- 라인의 끝을 알리는 마크(운영자 되돌림(carriage return) 혹은 라인 피드 혹은 모두)

명확하게 허용되지 않은 입력은 해석 프로그램이 에러의 메시지를 보내는 원인이 되거나 라인을 무시 하게 할 수있다.

스페이스 와 탭 키는 코드라인에 사용 가능하고 코멘트 안의 것을 제외하고는 라인의 성격을 바꾸지 않는다. 이것은 입력라인을 조금 이상하게 보일지도 모른다. 예를 들어, 라인 `g0x +0. 12 34y 7` 과 `go x+0.1234 y7`은 동등하다. 공란이 있는 라인도 사용 가능하다. 하지만 무시 하고 넘어 간다. 입력은 코멘트를 제외 하고는 대소문자를 가린다.

### 10.5.1 라인번호

라인 번호는 0과 99999 사이의5자리 까지의N 문자 뒤의 정수로 나타낸다. 라인 번호는, 보통 정상적인 순서로 이용되지만, 반복 되거나 순서없이 사용 될 수있다. 그리고 이것은 반드시 사용해야 하는 것은 아니지만 사용하게 된다면 반드시 적절한 위치에 있어야만 한다.

### 10.5.2 서브루틴 라벨

서브루틴 라벨은 0과 99999 사이의 5자리 까지의O문자 뒤의 정수로 나타낸다. 서브루틴 라벨은 특정한 순서에 관계 없이 사용 될 수있으나 프로그램에서 이의 위반에 대한 에러를 띄우지 않을 수 있어도 유일해야 한다. 코멘트를 제외 하고는 같은 라인에서 어떤 것도 서브루틴 라벨과 같아 서는 않된다. 알아두어야 할 것은 라인 번호는 현재의 버전에서는 허용 되지 않는다.

### 10.5.3 문자

실제 값 앞에 오는 문자는 N과 O를 제외한 문자가 된다.

문자는 그림 11.2에 보여진 것과 같이 어떤 문자로도 시작 할 수있다. 테이블은 N 과 O를 포함하여, 비록 라인번호가 문자가 아니더라도, 위에 정의된 바와 같이 완성하고 있다. 몇개의 문자(I, J, K, P, R)는 다른 문장에서 다른 뜻을 가질 수 있다.

실제 값은 프로세스 될 일부 문자(character)의 집합체로 숫자를 만들어 낸다. 실제 값은 함수 값 일 수 있고(예 341 혹은 -0.8807) 파라미터 값, 기호 표식(expression), 혹은 단일 연산값(unary operation value) 이 될 수 도 있다. 이러한 것들에 대한 정의는 즉각 나타 난다. 숫자를 만들어 내는 프로세스 문자는 수치 계산 이라고 부른다. 함수는 스스로 수치화 한다.

#### 10.5.3.1 숫자

다음의 법칙이 함수에 사용된다. 이런 법칙에서 숫자는 0 과 9 사이의 단일 문자가 된다.

- 숫자는 1)추가적인 플러스나 마이너스 기호로 구성되고 2) 영에서 부터 일정한 숫자 가 뒤를 이으며3) 소숫점이 따라 올 수 있고 4)다시 뒤를 이어 많은 숫자가 올 수도 있다.
- 숫자는 두가지 종류가 있다 : 정수와 소수. 정수는 소숫점을 포함하지 않는다. 소수는 포함한다.



- 숫자는 어떠한 아라비아 숫자로도 가능하며 라인길의 제한에 따라 달라진다. 대략 17개의 중요기능만 보존이 된다.
- 기호가 없는 0이 아닌 숫자는 양수로 간주한다.

알아 둘것은 앞부분(소숫점 이전과 0이 아닌 처음 숫자) 과 뒷 부분(소숫점 이후와 0이 아닌 마지막 숫자) 의 0을 사용해도 되지만 불필요 하다. 앞과 뒤에 붙은 0은 없는 것처럼 간주 하여 있는 숫자만 읽게 된다.

Mach3 에 의해 쓰여지는 특별한 숫자는 일부 제한된 값이나 일정 범위의 값으로 규정된다. 많은 사용되는 것으로, 소숫점은 반드시 정수주위에 위치 해야 한다 : 이것은 인덱스, M 코드, G코드 에 10을 곱한값을 포함한다. 소숫점은 0.0001의 정수안에 있을 경우 충분히 가까운 것을 간주 한다.

### 10.5.3.2 파라미터 값

이하 번역 중략

### 10.5.3.3 표식과 이진 연산

이하 번역 중략

### 10.5.3.4 일진 연산 값

이하 번역 중략

### 10.5.4 파라미터 설정

이하 번역 중략

### 10.5.5 주석과 메시지

이하 번역 중략

### 10.5.6 반복 아이템

이하 번역 중략

### 10.5.7 아이템 순서

이하 번역 중략

### 10.5.8 명령어와 장비 모드

이하 번역 중략

## 10.6 형식상의 그룹

형식 명령은 “ 형식 그룹” 으로 분류하고 오직 한 그룹만 주어진 시점에서 유효하다. 일반적으로 형식 그룹은 논리적으로 불가능한, 동시에 진행 할 수 없는 명령어들을 포함한다. 예를 들면 인치 단위와 밀리미터 단위로 측정한다 등등. 머시닝 시스템은 동시에 여러 모드에 있을 수 있는데 그 중 각 형식 그룹이 유효한 모드가 있을 수도 있다.

<p>G 코드를 위한 형식 그룹</p> <ul style="list-style-type: none"> <li>• group 1 = {G00, G01, G02, G03, G38.2, G80, G81, G82, G84, G85, G86, G87, G88, G89} 동작</li> <li>• group 2 = {G17, G18, G19} 평면 선택</li> <li>• group 3 = {G90, G91} 거리 모드</li> <li>• group 5 = {G93, G94} 피드 비율 모드</li> <li>• group 6 = {G20, G21} 단위</li> <li>• group 7 = {G40, G41, G42} 커터의 반경 보정</li> <li>• group 8 = {G43, G49} 툴 길이 옵셋</li> <li>• group 10 = {G98, G99} 고정 (canned) 사이클의 리턴 모드</li> <li>• group 12 = {G54, G55, G56, G57, G58, G59, G59.xxx} 좌표계 선택</li> <li>• group 13 = {G61, G61.1, G64} 경로 제어 모드</li> </ul>
<p>M 코드를 위한 형식 그룹 :</p> <ul style="list-style-type: none"> <li>◆ group 4 = {M0, M1, M2, M30} 멈추기</li> <li>◆ group 6 = {M6} 툴 체인지</li> <li>◆ group 7 = {M3, M4, M5} 스피indle 회전</li> <li>◆ group 8 = {M7, M8, M9} 쿨런트 (special case: M7 and M8 may be active at the same time)</li> <li>◆ group 9 = {M48, M49} 피드와 속도 번복 제어 활성화, 비활성화</li> </ul>
<p>위의 형식 그룹에 추가 하여 비형식 그룹이 아래와 같다.</p> <p><b>G codes:</b></p> <ul style="list-style-type: none"> <li>◆ group 0 = {G4, G10, G28, G30, G53, G92, G92.1, G92.2, G92.3}</li> </ul>

그림 10.3 형식 그룹

여러 형식 그룹에서 장비 시스템이 명령을 대기 하고 있을 때 하나의 그룹은 유효 해야 한다. 이런 형식 그룹을 위한 기본적인 셋팅이 있다. 장비 시스템이 On 되거나 혹은 다시 시작했다면 기본 값들은 자동적으로 효력을 가진다.

그룹1, 테이블의 첫 그룹으로 동작을 위한 G 코드 그룹이다. 이들중 하나는 항상 유효하다. 그리고 그것은 현재 동작 모드라고 한다.

축 문자를 사용 하는 그룹 0과 그룹 1, 모두에서 비롯한 G 코드를 같은 라인에 넣으면 에러가 발생한다.

그룹 1에서 사용하는 축 문자가 잠재적으로 유효성이 있고 그룹 0의 G 코드가 축 문자를 라인에서 사용 한다면 그룹 1의 G 코드는 그 라인에서 중지 하게 된다.

그룹 0에서 축문자를 사용하는 G 코드는 G10, G28, G30 그리고 G92 가 된다.

Mach3 각 스크린의 맨 위에 현재 모드를 표시한다.

## 10.7 G codes

Mach3 의 G 코드 입력 언어에 관한 내용과 설명은 그림 10.4 에 나와 있다. 해설에는 명령어 원형을 포함하고 있으며 있고 전언(courier) 형식으로 되어 있다.

명령어 원형에서 기호” ~” 가 뜻하는 것은 실제 값을 나타낸다. 먼저 설명했던것과 같이, 실제 값은 양 함수( 예를 들어 4.4) 가 될 수있고, 수식도 될 수 있고 (예 [2+2.4]) 파라미터 값(예 #88) 이 될 수도 있고 단항 함수 값 (예 acos[0]) 이 될 수도 있다.

대부분의 경우, 만약 축 문자(X~, Y~, Z~, A~, B~, C~, U~, V~, W~) 가 주어 진다면, 목적지점을 지정한다. 축 번호는, 명확한 절대좌표로 기술 되어지지 않으면, 현재 사용 하는 좌표계에 관계된다. 축 문자가 옵션 이면, 현재 생략된 축은 현재의 값을 갖게 된다. 명령어 원형의 어떤 아이템은 옵션이라고 명시 되지 않는한 무엇이든 필요로 한다. 그리고 필요한 아이템이 생략 됐다면 에러가 난다.

U, V 와 W 는 각각 A, B 그리고 C 와 같다. A와 U를 같이 사용하고, B 를 V와 같이 사용하면 잘못된 것이다. (마치 A를 두번 사용 하는 것과 같다.) 코드 U, V, W에 관해서는 매 번 명백히 명시 되지 않지만 A, B, C 를 의미한다.

원형(prototype)에서, 문자 뒤에 오는 값들은 양 함수(explicit number)로 주어 진다. 특별히 다르게 명기되지 않았다면, 양 함수는 실제 값이 될 수 있다. 예를 들어, G10 L2는 G[2\*5] L[1+1] 과 같다. 파라미터 100의 값이 2였다면, G10 L#100은 같은 뜻이 된다. 예제와 같이 양 함수가 아닌 실 값을 사용 하는 것은 그렇게 유용하지 못하다.

L~ 이 원형에 쓰여져 있다면 “ ~” 는 L 넘버로 참조 될 수 있다. 또, “ ~” 가 H~에 있다면 H 넘버로 불리워 진다.

배율(scale factor)이 축에 적용 된다면 해당 하는 X, Y, Z, A/U, B/V, C/W 의 값에 적용 된다.

G 코드 요약	
G0	빠른 위치 잡기
G1	선형 보간
G2	시계방향 원형/ 나선형 보간
G3	반시계 방향 원형/나선형 보간
G4	머무르기(Dwell)
G10	좌표계 원점 설정
G12	시계 방향 원형 포켓
G13	반시계 방향 원형 포켓
G15/G16	극 좌표가 G0 와 G1 에서 움직인다.
G17	XY 좌표면 선택
G18	XZ 좌표면 선택
G19	YZ 좌표면 선택
G20/G21	인치/밀리 단위
G28	홈으로 리턴
G28.1	기준 축들
G30	홈으로 리턴
G31	직선 프로브
G40	커터 반경 보정 취소
G41/G42	커터 반경 보정 시작 왼쪽/오른쪽
G43	툴 길이 옵션 적용(+)

G49	틀 길이 옵션 취소
G50	모든 배율 요소(scale factor)를 1.0 으로 리셋
G51	축 데이터 입력 배율 요소 설정
G52	임시 좌표계 옵션
G53	절대 장비 좌표계에서 이동
G54	고정체(fixture) 옵션 1 사용
G55	고정체(fixture) 옵션 2 사용
G56	고정체(fixture) 옵션 3 사용
G57	고정체(fixture) 옵션 4 사용
G58	고정체(fixture) 옵션 5 사용
G59	고정체(fixture) 옵션 6 사용/일반적 고정체 번호 사용
G61/G64	정확한 정지/등속 모드
G73	고정 싸이클(canned cycle) - 때려서(peck) 뚫기
G80	동작 모드 취소( 고정 싸이클 포함)
G81	고정 싸이클 - 뚫기
G82	고정 싸이클 - 대기 하고 뚫기
G83	고정 싸이클 - 때려서(peck) 뚫기
G84	고정 싸이클 - 오른손 암나사 깎기
G85	고정 싸이클 - 보이링, 대기 없음, 피드 아웃(feed out)
G86	고정 싸이클 - 보이링, 스핀들 정지, 빠르게 빠져 나옴(rapid out)
G88	고정 싸이클 - 보이링, 스핀들 정지, 수동으로 전환(manual out)
G89	고정 싸이클 - 보이링, 대기, 피드 아웃
G90	절대 거리 모드
G91	증분 거리 모드
G92	옵션 좌표 와 파라미터 설정
G92.x	G92 등 취소
G93	반비례 타임 피드 모드
G94	피드/분 모드
G95	피드/회전 모드
G98	초기 레벨이 고정 싸이클 후 리턴
G99	R 포인트 레벨이 고정 싸이클 후 리턴

## 그림 10.4 G 코드 테이블

### 10.7.1 고속 선형 운동(Rapid Linear Motion) - G0

1)고속 선형 운동에서, 프로그램 G0 X~ Y~ Z~ A~ B~ C~, 이때 모든 축 문자가 옵션이면, 적어도 하나는 사용되어야 한다. G0 는 현재 모션 모드가 G0 일때 옵션이 된다. 이것은 좌표화된 선형 운동을 목적지까지 일정 비율로 보여 줄 것이다. G0 일때는 커팅하지 않는다.

2) 만약 G16 이 극 좌표 원점을 설정하도록 실행 됐다면, 반경과 각도 G0 X~ Y~에 의해 기술되는 지점까지 고속 선형 운동이 사용될 수 있다. X~는 G16 극 좌표원점에서부터 시작하는 선의 반경이고 Y~는 3 시 방향에서 반시계 방향으로 증가하는 값으로 측정된 각도 이다. G16 이 실행되는 시점에서의 위치 지정 좌표는 극 좌표원점이다. 모든 축의 단어가 생략되어 있다고 한다면 에러가 난다.

만약 커터 반경 보정이 작동 중이라면, 모션은 위와 다를 것이다. 커터 보정을 참고 하라. G53 이 같은 선 상에서 프로그램 됐다면 모션은 또한 다를 것이다. 절대 좌표를 참고 하라.

### 10.7.2 피드 비율에 따른 선형 운동 - G1

1) 선형운동을 위한 피드비율(절삭 할때나 아닐때) 프로그램 G1 X~ Y~ Z~ A~ B~ C~ 에서, 모든 축 문자는 옵션이나 반드시 하나 이상은 사용 되어야 한다. G1 은 현재 모션 모드가 G1 이라면 옵션이 된다. 이것은 좌표화된 선형 운동으로, 현재 피드비율로 목적 지점 까지 움직일 것이다.

2) G16 이 극좌표원점을 설정하기 위해 실행 됐다면 피드비율 상의 선형 운동은 반경과 각도 G0 X~ Y~로 기술되는 지점까지 사용될 수있다. X~는 G16 극 좌표 원점에서 부터 시작 되는 선의 반경이고 Y~는 3 시 방향에서 부터 반시계 방향으로 증가하는 값으로 측정되는 각도 이다.

G16을 실행 할 때의 현재 위치 좌표는 극 좌표원점 이 된다.

-모든 축의 단어가 생략되어 있다고 한다면 에러가 난다.

만약 커터 반경 보정이 작동 중이라면, 모션은 위와 다를 것이다. 커터 보정을 참고 하라. G53 이 같은 선 상에서 프로그램 됐다면 모션은 또한 다를 것이다. 절대 좌표를 참고 하라

### 10.7.3 피드비율에 따른 호 - G2, G3

원형이나 나선형 호는 G2(시계방향 호) 나 G3(반시계 방향 호)로 지정된다. 원이나 나선의 축은 반드시 장비좌표계의 X, Y 혹은 Z 축에 평행을 이루어야 한다. 축은 G17(Z축, XY 면) G18(Y 축, XZ 면) 혹은 G19( X 축, YZ 면) 으로 선택된다.

만약 코드 라인이 호를 만들고 회전 축 동작을 포함 한다면, 회전축은 일정 비율로 회전 하여 회전 동작이 XYZ 동작과 동시에 시작 하고 끝나게 한다. 이런 종류의 라인은 좀처럼 프로그램 되지 않는다.

만약 커터 반경 보정이 작동 중이라면, 모션은 위와 다를 것이다. : 커터 보정을 참고 한다.

호를 지정하는 것에는 두가지 포맷이 허용 된다. 중심 포맷과 반경 포맷 이다. 두가지 포맷에서 G2, G3 는 만약 현재 동작 모드로 되어 있다면 옵션이된다.

#### 10.7.3.1 반경 형태의 호

반경 포맷에서, 선택된 표면 상의 호 끝점의 좌표는 호의 반경을 따라 지정 된다. 다음과 같이 프로그램 한다. G2 X~ Y~ Z~ A~ B~ C~ R~. R은 반경이다. 축 문자는 모두 옵션 이지만 반드시 선택 된 면에서 하나나 그 이상을 사용 해야 한다. R 넘버는 반경이다. 양 수의 반경은 호가 180 도나 혹은 이하로 회전 하는 동안 음 수의 반경은 180 도에서 359.999 까지 이다. 호가 나선형이라면 나선축과 평행한 좌표축에서 호의 끝점 값 또한 지정된다.

다음과 같은 경우에는 에러가 난다.

- 선택된 면의 축을 위한 두개의 축문자가 생략 됐을때.
- 호의 끝점이 현재 지점 일때.

반경 포맷이 거의 원에 가깝거나 혹은 반원에 가깝거나 할 때 프로그램 하는 것은 좋지 않다. 왜냐하면 끝점의 위치에서 작은 변화는 원의 중심위치에서 훨씬 커다란 변화를 주기 때문이다. 확대 효과는 충분한 크기 때문에 일정 범위에 있는 라운딩에러 일지 라도 허용범위를 넘어가는 질삭을 한다. 거의 완전한 원과 반원도 좋지 않지만 그 이외의 크기의 호는 적절하다고 할 수있다. (작은 호에서 165도 까지 혹은 195에서 345 까지)

G17 G2 X 10 Y 15 r 20 z 5.

위의내용은 반경 포맷 명령어로 호를 밀링 하는 예제이다. 시계 방향 원호나 나선호가 Z 축과 평행하고 X=10, Y=15, Z=5에서 반경 20 으로 끝난다는 뜻이다. Z 값이 5라면 이것은 원의 호로 XY 면과 평행이 된다. 그렇지 않을 경우는 나선 호가 된다.

### 10.7.3.2 중심점 형태의 호.

중심포맷에서, 선택된 면의 호 끝점 좌표는 현재 위치로 부터 호의 중심 옵셋을 따라간다. 이런 포맷에서 호의 끝점이 현재 좌표라고 해도 좋다. 다음과 같은 경우는 예러가 난다.

-호가 선택된 좌표면에 투영 됐을때 현재 지점에서 중심점까지 거리와 끝점과 중심점 사이의 거리가 0.0002 인치혹은 밀리 미터 이상 차이가 날때.

중심은 I, J 문자를 사용함으로 지정 된다. 이런 것들을 해석하는 방법은 두가지가 있다. 일반적인 방법은 I 와 J 가 호가 시작 할때 현재 지점에 상대적으로 중심이 되는 것이다. 이것은 Incremental IJ mode(점중 IJ 모드) 라고 불리기도 한다.

두번째 방법은 I 와 J가 현재 시스템에서 실질적 좌표로 중심을 지정하는 것이다. 이것을 잘못 부를 수도 있지만 절대 IJ 모드(Absolute IJ mode) 라고 부른다. IJ 모드는 Config > State ... 에서 설정한다. 모드의 선택은 상업용 컨트롤러와 호환성을 띄어야 한다. 실 사용에서는 Incremental (점중) 모드가 더 낫다는 것을 알게 될 것이다. 절대 모드에서도 물론 I 와 J 문자가 호의 중심이 원점에 있지 않는이상 둘다 사용 할 필요성이 있다.

XY 좌표면이 설정될 때, 다음과 같이 프로그램 한다. G2 X~ Y~ Z~ A~ B~ C~ I~ J~. 축 문자는 는 적어도 X와 Y 가 사용 될때 옵션이 될 수 있다. I 와 J 는 현재의 위치나 좌표로 부터 옵셋을 나타내지만 원 중심의 IJ 모드에 달려 있다. I, J 는 한개 이상의 문자가 사용할 때 옵션이 될 수 있다. 다음의 경우는 예러가 난다.

- X, Y 가 둘다 생략 된다.
- I, J 가 둘다 생략 된다.

XZ 좌표면이 선택 될 때, 다음과 같이 프로그램 한다.

G2 X~ Y~ Z~ A~ B~ C~ I~ K~

축 문자는 X 나 Z 가 사용 될때 옵션이 될 수있다. I 와 K는 현재의 위치나 좌표로 부터의 옵셋을 나타내지만 원의 중심 IJ 모드에 달려있다. I 와 K 는 둘중 하나 이상이 쓰일때 옵션이 될 수있다.

다음의 경우는 예러가 난다.

- X와 Z가 둘다 생략 된다.
- I 와 K 가 둘다 생략된다.

YZ 면이 선택 될때, 다음과 같이 프로그램 한다.

G2 X~ Y~ Z~ A~ B~ C~ J~ K~

축 문자는 Y 나 Z 가 사용 될때 옵션이 될 수있다. J 와 K는 현재의 위치나 좌표로 부터의 옵셋을 나타내지만 원의 중심 IJ 모드에 달려있다. J 와 K 는 둘중 하나 이상이 쓰일때 옵션이 될 수있다.

다음의 경우는 예러가 난다.

- Y와 Z가 둘다 생략 된다.
- J 와 K 가 둘다 생략된다.

아래의 예제는 중심 포맷 명령을 사용하여 점중IJ모드로 호를 밀링 한다.

G17 G2 x10 y16 i3 j4 z9

이것의 의미는 시계 방향 원호나 나선호가 Z 축과 평행하고 X=10, Y=6, Z=9에서 X 방향으로, 현재 X 위치에서 3만큼의 중심옵셋을 갖고 Y는 4만큼 옵셋을 갖는 다는 뜻이다. 현재 위치가 초기 시점에서 X=7, Y=7을 갖는다면, 중심은 X=10, Y=11이 될 것이다. Z의 시작값이 9라면, 이것은 원호가 되며 그렇지 않으면 나선호가 될 것이다. 호의 반경은 5가 된다. 위의 호를 절대 IJ 모드로 나타내면

G17 G2 x10 y16 i10 j11 z9 이된다.

중심 포맷에서, 호의 반경은 지정되지 않았지만 원의 중심에서 현재 지점이나 혹은 호의 끝점까지의 거리로 인해 쉽게 알 수있을 것이다.

#### 10.7.4 머무르기(Dwell) – G4

대기의 프로그램을 다음과 같이 해 보라. G4 P~. 이것은 축을 P에서 지정한 대로 몇 초 혹은 밀리 초 동안 움직이지 않게 할 것이다. 이 시간 단위는 Config > Logic 대화 상자에서 설정한다.  
예를 들어, 단위가 Second(초) 단위로 설정됐다면, G4 P0.5 프로그램은 0.5초 동안 대기 하게 될 것이다. P 값이 음수로 된다면 에러가 날 것이다.

#### 10.7.5 좌표계 데이터 툴과 작업 오프셋 테이블 설정(Set Coordinate System Data Tool and work offset tables) – G10

좌표계의 더 자세한 정보는 툴과 작업 오프셋에 관한 내용을 살펴 보라.

툴의 오프셋 값을 설정하려 한다면 다음과 같이 프로그램한다.

G10 L1 P~ X~ Z~ A~, P 값은 반드시 0 ~ 255 사이의 정수가 되어야 한다. P 값에 의해 지정된 툴 오프셋은 주어진 값으로 리셋 한다. A 값은 툴 팁의 반경을 리셋 한다. 오직 축 문자에 포함된 그 값들만이 리셋 할 것이며 툴 지름은 이런 방법으로 설정 할 수 없다.

고정물 좌표계의 원점을 위해 좌표 값을 설정 하려면 다음과 같이 프로그램 한다.

G10 L2 P~ X~ Y~ Z~ A~ B~ C~. 이때 P 값은 1~255 사이의 정수가 되어야 한다. - 고정물 숫자( G54 와 G59 에 대응하는 값 1~6) 그리고 모든 축 문자는 옵션이다. P 값에 의해서 지정된 좌표 원점은 주어진 좌표 값으로 리셋 한다. 라인에 축문자가 포함된 좌표들 만 리셋이 된다.

P 숫자가 0~255까지의 정수가 아니라면 에러가 난다.

원점 오프셋(G92, G92.3)이 G10 이 사용되기 전에 작동 되었다면 사용 후에도 마찬가지 일 것이다.

G10 명령어에 의해 원점이 설정된 좌표계는 G10 이 실행 될 당시 작동 혹은 작동 하지 않을 수 도 있다. 설정된 값은 Table 화면에 있는 버튼으로 툴이나 고정물 테이블을 저장 하지 않았다면 계속 남아 있지 않는다.

예: G10 L2 P1 x3.5 y17.2. 이것은 처음 좌표계의(G54에 의한) 원점을 X=3.5 Y=17.2(절대 좌표로)에 설정한다. 원점의 Z 좌표는 라인이 실행되기 전의 어떤 원점 좌표도 될 수 있다.

#### 10.7.6 시계/반시계 방향 회전 포켓(Clockwise/Counterclockwise circular pocket) – G12, G13

이러 회전 포켓 명령어는 일종의 닫힌 싸이클로, 사용되고 있는 툴(커터) 보다 큰 원형 홀을 만들거나 혹은 O 링과 같은 내부 홀을 만들 수 있다.

G12 I~ 와 같이 프로그램하여 시계 방향으로 움직이게 하고 G13 I~로 프로그램 하여 반시계 방향으로 움직이게 한다.

만약 I 문자와 그 방향의 원형 절삭이 초기 X, Y 좌표가 중심으로 지정 됐다면, 툴은 주어진 값에 의해 X 방향으로 움직이게 된다. 툴은 다시 중심으로 돌아 가게 된다. 현재 좌표면이 XY 가 아니라면 결과는 정의되지 않는다.

#### 10.7.7 극(Polar) 모드로 들어가고 나가기(Exit and Enter Polar mode) – G15, G16

G0, G1은 X Y 좌표면을 움직이는데 이것은 반경과 각도로 임시 중심점에 상대적인 좌표를 지정하기 위해서이다. G16이 이런 모드가 되도록 프로그램한다. 제어점의 현재 좌표는 임시적인 중심이 된다.

G15를 프로그램 하여 일반 데카르트 좌표로 되돌린다.

G0 X10 Y10 // G0 는 10,10 으로 움직인다.

G16 // 폴라 모드의 시작.

G10X10Y45

(이것은 X17.xxx Y 17.xxx 로 움직이며 초기 좌표 10, 10의 초기 좌표로부터 반경 10 각도 45도 인 원에서의 하나의 점이 된다.)

이것은, 예를 들어, 구멍들로 만든 원을 드릴링 할 때 상당히 유용할 수 있다. 아래의 코드는 50mm 반경에 X=10, Y=5.5 좌표를 가지는 원을 펙 드릴링 높이 Z=-0.6 인 매회 10도 마다 구멍을 내어 원을 만든다.

```
G21 // 메트릭
G0 X10Y5.5
G16
G1 X50 Y0 // 반경 50 각도0 인 곳으로 폴라(극)이 움직임
G83 z-0.6 //펙 드릴
G1 Y10 // 원점 중심에서부터 10 도 움직임...
G83 z-0.6 //
G1 Y20

G1 Y30

G1 Y40
>... etc...
G15 //데카르트 좌표로 복귀.
```

주의:

G61 이 작동 중일 때는 X 혹은 Y를 G0 나 G1이외의 것으로 움직여서는 안 된다.

G16은 Fanuc 실행과 달라 현재 좌표를 극좌표 중심(polar center)로 사용한다. Fanuc 버전은 0,0 이 아닌 원의 중심이 아닌 것에 대해서는 무수히 많은 원점 이동을 필요로 하여 원하는 결과를 얻는다.

#### 10.7.8 좌표면 선택(Plane Selection) – G17, G18, G19

G17은 XY 좌표면을 선택하고, G18은 XZ 면을, G19은 YZ 면을 선택한다. 선택한 면의 효력은 G2/3 과 닫힌 사이클(Canned cycle)에 설명되어 있다.

#### 10.7.9 길이 단위(Length Unit) – G20, G21

G20을 프로그래밍하여 길이 단위를 인치로 하고 G21을 프로그래밍하여 밀리미터 단위로 프로그래밍 한다. G20 혹은 G21은 어떤 동작이 발생하기 전에 프로그래밍하여 사용하는 것이 바람직하며 프로그램내에서 다른 것을 바꿔 사용하는 것은 좋지 않다.

#### 10.7.10 홈으로 돌아 오기 (Return to Home) – G28, G30

홈의 위치는 파라미터 5161-5166 에서 정의되었다. 파라미터 값은 절대 좌표계에 의한 것 이지만 길이 단위는 지정되지 않는다.

프로그래밍된 위치를 경유하여 홈의 위치로 돌아가기 위해서 다음과 같이 프로그래밍 한다.

G28 X~ Y~ Z~ A~ B~ C~ 혹은 G30을 사용한다. 모든 축 문자는 옵션이다. 경로는 현재 위치에서 프로그래밍된 위치까지 가로지르는 운동을 하고 뒤이어 홈 위치로 가로 지른다. 어떤 축 문자도 프로그래밍 되지 않으면 중간 지점이 현재의 지점이 되어 한번의 움직임만 있을 것이다.

#### 10.7.11 축 기준점 G28.1

G28.1 X~ Y~ Z~ A~ B~ C~ 를 프로그래밍하여 주어진 축 기준을 참고 할 수있다. 축들은 현재 피드레이트로 Configuration 에서 정의된 바와 같이 홈 스위치로 전진한다. 절대좌표계가 축 문자에 의해 주어진 값에 접근하게 되면 피드레이트는 Configure> Config Referencing 에 의해 정의된 값으로 설정된다. 주어진 현재 절대 좌표위치가 거의 정확 하다면, 기준 스위치에서 soft stop(limit?) 을 설정할 수 있게 한다.



## 10.7.12 수직 프로브(straight probe) - G31

### 10.7.12.1 수직 프로브 명령어

G31 X~ Y~ Z~ A~ B~ C~ 를 프로그램 하여 직선 프로브 오퍼레이션을 수행한다. 회전 축 문자가 허용 되지만 생략 하는 것이 좋다. 만약 회전 축 문자가 사용 된다면, 현재의 위치 값이 회전축 값과 같아 회전축이 움직이지 않게 한다. 선형 축 문자는 옵션이다. 하지만 그 둘중 하나 이상은 반드시 사용 되어야 한다. 스프indle에 붙어 있는 툴은 반드시 프로브가 되어야 한다.

다음의 경우는 예러가 난다.

- 현재의 위치가 프로그램된 위치로부터 0.254 밀리미터 나 0.01 인치 보다 작을 경우
- G31 역 타임 피드레이트 모드(inverse time feed rate mode) 로 사용될때
- 회전축이 움직이도록 명령 되어 질때
- X, Y, Z 축 문자 중 어떤 것도 사용 되지 않을 때

이런 명령어에 대응하여 장비는 제어점( 프로브 팁의 끝 점)을 프로그램된 지점으로 현재의 피드 레이트에 의해 직선으로 움직이게 된다. 만약 프로브가 프로그램된 위치에 닿게 되면, 닿은 지점으로 부터 살짝 뒤로 물러 서게 된다. 그리고 프로그램된 위치를 넘어서도 작동되지 않으면 예러 신호를 띄운다.

프로브가 성공적으로 설정되면, 2000에서 2500 까지의 파라미터는 프로브가 닿은 지점에서 제어점의 위치 좌표로 설정될 것이고 X, Y 그리고 Z를 설정하는 트리플릿(triplet)은, M40에 의해 열렸다면(macro/OpenDigFile() 기능 참고), 트리플릿(triplet)파일에 기록 될 것이다.

### 10.7.12.2 직선 프로브(Straight Probe) 명령어 사용 하기

직선 프로브 명령어 를 사용 할 때, 만약 프로브 뭉치가 Z 축과 평행 하고(즉, 회전축들이 0점에 있다면) 툴 길이 옵션이 프로브에 사용 됐다면 제어점은 프로브 팁 끝이 된다.

- 프로브에 관한 추가적인 정보가 없어도, XY 좌표 면에 대한 파트 면의 평행위치는 ,예를 들어, 찾을 수 있다.
- 프로브 팁의 반경을 개략적으로 알 수 있다면, YZ 나 혹은 XZ 면에 대한 파트 면의 평행위치는 ,예를 들어, 찾을 수 있다.
- 프로브 뭉치(shank)가 Z 축에 잘 정렬되어 있고 프로브 팁 반경을 개략적으로 안다면, 원형 홀의 중심은 ,예를 들어, 찾을 수 있다.
- 프로브 뭉치가 Z 축에 잘 정렬 되어 있고 프로브 팁 반경을 정확하게 안다면, 원형 홀의 직경을 찾는 것과 같은 것들에 직선 프로브(straight probe) 명령어를 더 많이 사용 하게 된다.

프로브 뭉치의 일직선을 고 정밀도로 정렬할 수 없다면 프로브 팁의 유효한 반경을 적어도+X, -X, +Y 그리고 - Y 방향으로 아는 것이 바람직하다. 이런 값들은 파라미터 파일에 포함 되어 저장되거나 혹은 Mach2 프로그램에 의해 저장될 수 있다.

회전 축을 0 점으로 설정하지 않고 프로브를 사용 하는 것 또한 적용 시킬 수 있다. 하지만 이렇게 하는 것이 0점으로 설정하는 것 보다 복잡하며 여기서는 언급 하지 않기로 한다.

### 10.7.12.3 예제 코드

사용 가능한 예제로서, 원형 홀의 직경과 중심을 찾는 코드가 아래 예제 10.5에 나와 있다. 정확한 결과를 산출 하기 위해 프로브 뭉치(shank)는 Z 축에 잘 정렬되어 있어야 하고 , 프로브 팁에서 가장 넓은 단면은 원형에 되어야 하며, 프로브 팁의 반경(즉, 원형 단면의 반경)은 정확히 알아야 한다. 프로브 팁의 반경을 개략적으로만 알게 된다면 , 홀 중심의 위치는 정확 할 지라도, 홀 직경은 그렇지 못할 것이다.

N010 (프로브는 원형 홀의 중심과 직경을 찾는다)  
 N020 (이 프로그램은 여기에 주어진 대로 진행 하지 않는다 )  
 N030 (i <숫자의 기술>.에 숫자를 넣어야만 한다.)  
 N040 (그 후 N020, N030, N040 라인을 지운다.)  
 N050 G0 Z <후퇴한 지점의 Z 위치> F <피드레이트>  
 N060 #1001=<홀 중심의 X 값>  
 N070 #1002=<홀 중심의 Y 값>  
 N080 #1003=<홀 안의 어떤 Z 값>  
 N090 #1004=<프로브 팁 반경>  
 N100 #1005=[<홀 직경>/2.0 - #1004]  
 N110 G0 X#1001 Y#1002 (근소한 홀 중심으로부터 위로 움직임r)  
 N120 G0 Z#1003 (조심히 홀로 들어감,여기에서 G1 을 G0로 대체 )  
 N130 G31 X[#1001 + #1005] (프로브, 홀의 +X 가장자리)  
 N140 #1011=#2000 (결과 저장)  
 N150 G0 X#1001 Y#1002 (홀의 중심으로 돌아가기)  
 N160 G31 X[#1001 - #1005] (프로브, 홀의 -X 가장자리)  
 N170 #1021=[[#1011 + #2000] / 2.0] (적절한 홀의 중심 X 값 찾기)  
 N180 G0 X#1021 Y#1002 (홀의 중심으로 돌아가기)  
 N190 G31 Y[#1002 + #1005] (프로브,홀의 +Y 가 장자리)  
 N200 #1012=#2001 (결과 저장)  
 N210 G0 X#1021 Y#1002 (홀의 중심으로 돌아가기)  
 N220 G31 Y[#1002 - #1005] (프로브,홀의-Y 가장자리)  
 N230 #1022=[[#1012 + #2001] / 2.0] (적절한 홀의 중심 Y 값 찾기)  
 N240 #1014=[#1012 - #2001 + [2 \* #1004]] (Y 방향으로 홀의 직경 찾기)  
 N250 G0 X#1021 Y#1022 (홀의 중심으로 돌아가기)  
 N260 G38.2 X[#1021 + #1005] (프로브, 홀의 +X 가장 자리)  
 N270 #1031=#2000 (결과 저장)  
 N280 G0 X#1021 Y#1022 (홀의 중심으로 돌아가기)  
 N290 G31 X[#1021 - #1005] (프로브, 홀의 -X 가장자리)  
 N300 #1041=[[#1031 + #2000] / 2.0] (절적인 홀의 중심 X 값 찾기)  
 N310 #1024=[#1031 - #2000 + [2 \* #1004]] (X 방향으로 홀의 직경 찾기)  
 N320 #1034=[[#1014 + #1024] / 2.0] (홀의 평균 직경 찾기)  
 N330 #1035=[#1024 - #1014] (홀 직경들의 오차 찾기)  
 N340 G0 X#1041 Y#1022 (홀의 중심으로 돌아 가기)  
 N350 M2 (끝)

#### 예제 10.5

예제 10.5 에서, <숫자의 기술> 이 의미하는 것은 서로 매칭되는 실 수를 입력 하는 것이다. 이번 섹션의 코드를 실행하고 나면, 중심의 X 값은 파라미터 1041에서 찾을 수 있을 것이고 중심의 Y 값은 1022 파라미터에 있을 것이다. 그리고 직경은 파라미터 1034에 있다. 추가적으로, X 축과 평행한 직경은 파라미터 1024에 있고 Y축에 대한 직경은 1014 파라미터에 있으며 원형 정도의 차이에 관한 파라미터는 1035에 있게 된다. 프로브 팁은 홀의 XY중심에 있게 된다.

예제는 스피들 안에 집어 넣을 다른 톨 교환에 대해서는 포함 하지 않는다. 필요한 경우 톨 교환에 대한 코드를 시작에 넣기 바란다.

#### 10.7.13 커터 반경 보정(Cutter Radius Compensation) – G40, G41, G42

커터 반경 보정을 off 하려면 G40. 을 프로그램 한다.

커터 반경 보정은 XY 좌표면이 활성화 될 때만이 작동 할 수있다.

커터 반경 보정을 왼쪽으로 on 하기위해서(즉, 툴 반경이 양수일때, 커터는 프로그램된 경로의 왼쪽에 머무르게 된다.), G41 D~ 로 프로그램 하고 커터 반경 보정을 오른쪽으로 on 하려면(즉, 툴 반경이 양수 일때 프로그램된 경로의 오른쪽에 머무르게 된다.) G42 D~로 프로그램 한다.

D 문자는 옵션이다. 만약 D 문자가 없다면 현재 스피들에 장착된 툴의 반경이 사용된다. 만약 사용되었다면 D 넘버는 일반적으로 필요 하지 않더라도 스피들 안의 툴 슬롯 넘버가 되어야 한다. D 넘버가 0 이 될 수도 있다; 0 의 반경 값이 사용될 것이다.

G41과 G42는 P 문자에 의해 충족되어야 한다. 이것은 현재 툴 테이블 엔트리에 주어진 툴 직경 값을 덮어 쓰기 할 것이다.

다음과 같은 경우는 예러가 난다.

- D 넘버가 정수가 아닐때, 그리고 음수 가 되거나 컨베이어 슬롯 넘버 보다 클경우.
- XY 좌표면 이 활성화 되지 않았을때
- 커터 반경 보정이 이미 on 상태에서 다시 on 으로 명령되어 질 때.

커터 반경 보정이 ON 되어 있을때의 머시닝 시스템의 작동에 대한 설명은 9장에서 설명 되었다. 알아 들것은 들어가고 나가는 동작의 유용한 프로그래밍에 대한 중요성이다.

#### 10.7.14 툴 길이 옵셋(tool length offset) – G43, G44 그리고 G49

툴 길이 옵셋을 사용 하기 위해 G42 H~를 프로그램 한다. 이때 H 넘버는 툴 테이블의 적절한 인덱스 가 된다. 이 테이블의 모든 엔트리는 양수가 될 것으로 예상된다. H 넘버는, 반드시는 아니지만, 스피들 안에 있는 툴의 슬롯 번호와 같아야 한다. H 넘버가 0 이 되어도 된다; 0의 옵셋 값이 사용 되어 진다. H를 삭제 하는 것은 0 값과 같은 효과를 낸다.

G44 는 호환성을 위한 것이고 테이블의 엔트리가 음수 옵셋일때 사용 된다. 다음의 경우에는 예러가 난다.

H 넘버가 정수가 아니거나, 음수 이거나, 컨베이어 슬롯의 넘버보다 큰 경우.

G49 을 프로그램 하면 툴 길이 옵셋이 없음을 나타낸다.

이미 사용 중인 같은 옵셋을 프로그램 할 수도 있고 현재 아무 것도 사용 되지 않으면 툴 길이 옵셋이 없다고 프로그램해도 된다.

#### 10.7.15 배율계수(scale) G50, G51

X, Y, Z, A, B, C, I&J 문자가 사용되기 전, 이곳에 적용될배율 요소를 정의 하기 위해 다음과 같이 프로그램 한다.

G51 X~ Y~ Z~ A~ B~ C~ 이때 X, Y, Z 등등의 문자는 주어진 축의 배율 요소가 된다. 이런 값들은 물론, 자동으로 배율되지 않는다.

G2나 혹은 G3로 타원형을 만들기 위해 비등한 배율 요소를 사용 하는 것은 허용 되지 않는다.

모든 축의 배율 요소를 1.0 으로 리셋 하기 위해서는 G50을 프로그램 한다.

#### 10.7.16 임시 좌표계 옵셋(Temporary Coordinate system offset) – G52

현재 지점에 주어진 음수나 양수 거리만큼의 옵셋을 주기 위해서(움직이지 않고) 다음과 같이 프로그램 한다.

G52 X~ Y~ Z~ A~ B~ C~.

이때 축 문자는 원하는 옵셋을 포함 하게 된다. 모든 축문자는 옵션이지만 하나 이상은 사용 되어야 한다.

축문자가 주어진 축에 사용 되지 않을 경우, 이 축 상에서 현재 위치 좌표는 바뀌지 않는다.

다음과 같은 경우는 예러가 난다.

- 모든 축 문자가 생략 될 경우.

G52 와 G92는 Mach2에 있는 공통의 내부 메커니즘을 사용 하고 모두 같이 사용 되지 않을 수도 있다.

G52가 실행 될 때, 현재 활성화 된 좌표계의 원점은 주어진 값으로 움직인다.

G52의 영향은 G52 X0 Y0 등으로 프로그램 하여 취소 시킬 수 있다.

여기 예제를 보자. 지정된 좌표계에서 현재 위치가 X=4 라고 가정 하자. 그러면 G52 X7은, X 축으로 7만큼의 오프셋을 설정하게 되어 현재 지점의 X 좌표는 -3 이 된다.

축 오프셋은 어떤 고정물(fixture) 좌표계를 이용하던 절대 거리 모드에 지정된 모션이 있을 때 항상 사용하게 된다. 그러므로, 모든 고정물 좌표계는 G52에 의해 영향을 받는다.

#### 10.7.17 절대 좌표계에서의 동작(Move in Absolute Coordinate) - G53

절대 좌표에 기술된 한 지점으로 움직이는 선형 운동을 위해 다음과 같이 프로그램 한다. G1 G53 X~ Y~ Z~ A~ B~ C~ (혹은 G1 대신 G0와 유사한). 이때 모든 축 문자는 옵션이지만 하나 이상은 사용 되어야 한다. 현재 모션 모드에 있을 경우 G0 혹은 G1은 옵션이 된다. G53은 형식이 아니며 반드시 활성화 되어지는 각 라인에 프로그램 되어야 한다. 이것은 프로그램 된 지점으로 좌표 운동을 하게 된다. 만약 G1이 활성화 됐다면, 모션 스피드는 현재의 피드 레이트(혹은 장비가 그렇게 빨리 움직일 수 없다면 천천히 움직인다) 로 움직일 것이다.

다음과 같은 경우는 예러가 난다.

- G53 이 G0 혹은 G1이 활성화 되지 않은 상태에서 사용된 경우
- G53 이 커터 반경 보정 이 on 상태 일때 사용된 경우.

좌표계의 개요에 관한 장을 참고 하기 바란다.

#### 10.7.18 작업 오프셋 좌표계 선택(Select Work Offset Coordinate System) - G54 에서 G59 와 G59 P~ 까지

작업 오프셋(work offset) #1 을 선택 하기 위해서 G54를 프로그램 한다. 그리고 유사한 방법으로 처음 6개의 오프셋을 프로그램 한다. 시스템 번호 - G - 코드 쌍은, (1-G54), (2-G56), (3-G56), (4-G57), (5-G58), (6-G59) 과 같다.

254(1-254) 개의 어떤 작업 오프셋에 액세스 하기위해 G59 P~ 를 프로그램 한다. 이때 P 문자는 필요한 오프셋 넘버를 제공한다. 그러므로, G59 P5는 사실상 G58과 동일 하다.

다음과 같은 경우는 예러가 난다.

- 커터 반경 보정이 on 일때 위의 G 코드 중 하나가 사용될 때.
- 좌표계의 개요에 관한 장을 참고 하기 바란다.

#### 10.7.19 경로 제어 모드 설정(Set Path Control Mode) - G61, G64

G61을 프로그램하여 머시닝 시스템을 정확 정지 모드(exact stop mode) 로 만들거나, 혹은 G64를 사용하여 등속 모드(constant velocity mode) 로 한다. 이미 활성화된 모드를 프로그램 해도 상관이 없다. 이런 모드들은 위에서 설명 되어 졌다.

#### 10.7.20 좌표계 회전 G68, G69

G68 A~ B~ I~ R~ 을 프로그램 하여 프로그램 좌표계를 회전 시킨다.

현재 좌표계에서 회전 중심은 A~ 는 X 좌표로, B~ 는 Y 좌표로 나타난다.

R~ 은 각도로 표현 되는 회전 각이다. (Z축의 위에서 볼 때 + 방향은 반시계 방향이 된다)

I~ 는 옵션으로 값은 사용 되지 않는다. 만약 I~가 존재한다면 주어진 R 값은 G68에 의해 설정된 어떠한 회전에 대해서도 값이 더해 진다.

예: G68 A12 B25 R45 은 좌표계가 45도 회전하도록 되며 이때 Z=12 Y=25가 된다.

이어서 G68 A12 B35 I1 R40 은 좌표계를 85도 회전 시켜 X=12, Y=25가 된다.

프로그램 G69는 회전을 취소 시킨다.

주의 :

- 이 코드는 오직 회전만 시키며 XY 평면 일 때 가능 하다.
- I 문자는 중심점이 이전에 사용 했던 것과 달라도 사용 가능 하지만, 이런 경우 결과를 주의 깊게 봐야 한다.

### 10.7.21 길이 단위 - G70과 G71

G70을 프로그램하여 길이단위를 인치로 한다. G71을 프로그램하여 밀리미터 단위로 한다.

G70 혹은 G71을 프로그램 시작 하기 전에 사용 하는 것이 좋다. 그리고 이후에는 사용 하지 않는 것이 바람직 하다. 현재 단위가 적절히 사용 되는 것에 대한 모든 것은 사용자의 책임이다. 유사한 G20/G21을 살펴 보기 바란다.

### 10.7.22 고정 사이클 - 고속 펙 드릴(Canned Cycle - High Speed Peck Drill) G73

G73 사이클은 깊이 뚫기(deep drilling) 나 대폐질 과 같은 밀링을 위해 사용된다. G83 또한 참고 하기 바란다. 위 아래로 움직이고 갈수록 깊이 들어가면서 밀링은 하지만 결코 홀의 바깥으로 나오지는 않는다. 틀에는 긴 홈이 파여 있는 것이 적합 한데 그것은 홀에서 나온 나무 파편들을 깨끗하게 한다. 이런 사이클은 Q 넘버를 사용하는데 이것은 Z축을 따라가는 델타 증감량을 나타낸다. 다음을 프로그램 한다.

G73 X~ Y~ Z~ A~ B~ C~ R~ L~ Q~

- 예비운동, G81에서 89까지 닫힌 사이클에서 기술 된 바와 같다.
- Z 축을 델타만큼 혹은 Z 위치까지 현재 피드레이트 로 아래를 향하여 움직인다. 이때 어떤 것이 됐던 더 낮은 깊이를 선택한다.
- Setting 화면의 G73 Pullback DRO에 의해 정의된 거리 만큼 다시 뒤로 빨리 움직인다.
- 빨리 현재 홀의 바닥으로 내려가고, 다시 살짝 올라간다.
- 처음 스텝 1,2 를 Z 위치가 스텝 1에 도달 할 때 까지 반복 한다.
- Z 축을 트래버스 비율(traverse rate) 로 올려 사이에 아무것도 없게 한다.

다음의 경우는 예러가 난다.

- Q 넘버가 음수 이거나 혹은 0 일때.

### 10.7.23 형식 모션 취소하기 Cancel Modal Motion - G80

G80을 프로그램하여 어떤 축의 움직임도 없게 한다. .

형식그룹 0 G 코드 가 축 문자를 사용하여 프로그램 되어 있지 않은 경우, G80 이 활성화 되어 있을 때 축 문자가 프로그램 되어있다면 예러가 발생한다.

### 10.7.24 고정 사이클 (Canned Cycles) - G81 에서 G89

닫힌 사이클 G81에서G89까지에 관해 이 섹션에서 설명한 바와 같이 적용되어 왔다. 두 가지의 예제가 G81의 설명과 아울러 아래에 나타나 있다.

모든 닫힌 사이클은 현재 선택된 좌표면에 대하여 수행 된다. 모든 좌표면은 선택 가능 하다.(XY, YZ, XZ). 이번 섹션 전체를 통해 대부분의 설명은 XY 좌표면을 사용해서 설명되어 있다고 간주 하기 바란다. 나머지 YZ, XZ 좌표면의 성질 또한 유사하다.

회전축 문자는 닫힌 사이클에서 사용이 허용되지만, 가능하면 생략 하는 것이 더 바람직 하다. 만약 회전축 문자가 사용됐다면, 숫자는 반드시 현재 위치 숫자와 동일 하게 만들어 회전 축이 움직이지 않도록 해야 한다.

모든 닫힌 싸이클은 NC 코드에 X, Y, 그리고 Z 숫자를 사용한다. 이런 숫자들은 X, Y, R 그리고 Z 위치를 결정하기 위해 사용된다. R(일반적으로 후진을 의미) 위치는 축을 따라 현재 선택된 면과 수직을 나타낸다(Z 축은 XY 면에, X 축은 YZ 면에, Y 축은 XZ 면에). 일부 닫힌 싸이클은 추가적인 인수를 사용한다.

닫힌 싸이클들에서, 같은 싸이클 들이 여러 줄에 걸쳐 차례로 사용될 때, 숫자는 첫 번째에서는 사용되어야 하지만, 나머지 줄에서는 옵션이 된다. 이때 우리는 그 숫자를 “ sticky(스틱키)” 라고 부르기로 한다. 스틱키 숫자는 특별히 다르도록 프로그램 되어 있지 않는 이상 나머지 라인에서도 같은 값을 유지 한다. R 숫자는 항상 스틱키가 된다.

증분 거리 모드(Incremental distance mode)에서 : XY 좌표면이 선택 되어 질 때, X, Y 그리고 R 숫자는 현재의 위치까지 증분하는 값으로 간주 되고 Z 는 Z의 움직임이 시작 되기 전의 Z 위치로부터 의 증분 되는 것으로 간주 된다. - YZ .혹은 XZ 좌표면이 선택 되면, 축 문자는 위와 유사하게 간주된다. 절대 거리 모드(absolute distance mode)에서, X, Y, R 그리고 Z 숫자들은 현재 좌표계에서의 절대 위치를 가리킨다.

L 숫자는 옵션이고 반복하는 횟수를 나타낸다. L=0는 허용 되지 않는다. 반복 기능이 사용될 때, 일반적으로 증분 거리 모드(Incremental distance mode) 가 사용되어, 같은 동작 순서가 직선 상에서 일정하게 빈 공간을 반복하게 만든다. 절대 거리 모드(absolute mode)에서, L > 1 뜻하는 것은 같은 곳에서 여러 번 반복 하라는 것이다. L 문자를 생략 하는 것은 L=1 과 같은 효과가 있다. L 숫자는 스틱키가 아니다.

XY 좌표면이 선택된 증분 모드(Incremental mode)에서 L>1 일때, X, Y 위치는 주어진 X, Y 값에 현재 X, Y 위치(첫 번째 회전 지점)를 더하거나 혹은 바로 전 회전지점의 끝(반복 되는 지점들)에서의 X Y 값을 더해서 결정된다. R과 Z 위치는 반복 될 때 변화 하지 않는다.

각각 반복 하는 곳에서 Z를 위로 올려 안전 하게 하는 높이는 후진모드(retract mode)의 설정에 따라 정해 진다. 후진모드가 G98이고 Z 원점이 R 보다 높을때 Z 원점으로 하거나 혹은 R 위치로 한다.

다음과 같은 경우는 예러가 난다.

- X, Y 그리고 Z 문자들이 닫힌 싸이클 이 진행되는 동안 전부 없을 때.
- 하나의 P 숫자가 필요로 하고 음수 P 숫자가 사용될 때
- 양수로 정해 지지 않은 L 숫자가 사용될 때
- 닫힌 싸이클을 진행 하는 동안 회전 축의 움직임이 있을 때
- 닫힌 싸이클을 진행 하는 동안 Inverse time feed rate 가 활성화 될 때
- 닫힌 싸이클을 진행 하는 동안 커터 반경 보정이 활성화 될 때

XY 좌표면이 활성화 될 때, Z 숫자는 스틱키이다라는 전제에서 다음과 같은 경우 예러가 난다.

- Z 숫자가 없고 같은 닫힌 싸이클이 이미 활성화 되지 않았을 때
- R 숫자가 Z 숫자 보다 작을 때

XZ 좌표면이 활성화 될 때, Y 숫자는 스틱키이다라는 전제에서 다음과 같은 경우 예러가 난다.

- Y 숫자가 없고 같은 닫힌 싸이클이 이미 활성화 되지 않았을 때
- R 숫자가 Y 숫자 보다 작을 때

YZ 좌표면이 활성화 될 때, X 숫자는 스틱키이다라는 전제에서 다음과 같은 경우 예러가 난다.

- X 숫자가 없고 같은 닫힌 싸이클이 이미 활성화 되지 않았을 때
- R 숫자가 X 숫자 보다 작을 때

#### 10.7.24.1 예비 동작 과 중간 동작(Preliminary and In-Between Motion)

닫힌 싸이클(Canned Cycle) 의 모든 초기 실행에서, XY 좌표면이 선택 되고, 만약 현재 Z 위치가 R 위치 아래 에 있다면, Z 축은 R 위치로 이동 하게 된다. 이것은 L 값에 관계 없이 단 한번 발생한다.

추가적으로, 첫 싸이클의 시작과 각각의 반복하는 곳에서, 다음의 중 하나 이상의 동작이 발생한다.

- XY 좌표면 주어진 점에서 면과 평행한 직선 이동
- R 위치에 현재 있지 않다면, Z 축이 R 위치로 직선 이동을 한다.

XZ 혹은 YZ 좌표면이 활성화 되어 있다면, 사전, 중간 동작은 위와 유사 하다.

#### 10.7.24.2 G821 싸이클

G81 싸이클은 드릴링을 하기 위함이다. G81 X~ Y~ Z~ A~ B~ C~ R~L~ 을 프로그램 해 본다.

- 사전 동작은 위의 설명과 같다.
- Z의 위치를 현재 피드 비율로 움직인다.
- 되돌아 가기 비율로 Z를 움직여 공간을 만든다.

**예제 1.** 현재의 위치가 1,2,3 이고 XY 면이 선택 됐고 아래 NC 코드가 해석 된다.

G90 G81 X4 Y5 Z1.5 R2.8

이렇게 하면 절대 거리 모드 G90과 과거의 “ Z” 철회 모드 G98 을 호출 하고 G81, 드릴링 싸이클을 한 회 시행 할것을 요구한다. X 숫자와 X 위치는 4가 된다. Y 숫자와 Y 위치는 5가 된다. Z는 1.5가 된다. R 숫자와 Z의 틸름은 2.8이 된다. 그리고 다음 동작이 따른다.

- XY 평면에 평행으로 가로 질러 4,5,3 위치로 간다.
- Z 축에 평행으로 가로 질러 4,5,2.8 위치로 간다.
- Z 축에 평행으로 피드를 주어 4,5,1.5 위치로 간다.
- Z 축에 평행으로 가로질러 4,5,3 위치로 간다.

**예제 2.** 현재 위치가 1,2,3 이고 XY 평면이 선택 됐고, 아래 NC 코드가 해석 됐다고 하자.

G91 G81 G98 X4 Y5 Z-0.6 R1.8 L3

이것은 증가하는 거리모드 G91과 과거 Z 철회 모드 를 호출하고 G81 드릴링 싸이클이 3회 반복 되는 것을 요청 할 것이다. X 숫자는 4, Y는 5 Z는 -0.6 그리고 R은 1.8 이 된다. 초기 X 위치는 5(=1+4), 초기 Y 위치는 7(=2+5) 그리고 Z 틸름 위치는 4.8(=1.8+3) 그리고 Z 위치는 4.2(=4.8-0.6) 과거 Z는 3.0 이 각각 된다.

첫 움직이는 Z축을 따라 가로질러 1,2,4.8에 위치하는데 그것은 과거 Z가 틸름 Z 보다 작기 때문이다. 처음 반복 동작은 3단계로 움직인다.

- XY 평면에 평행으로 가로질러 5,7,4.8 로 위치한다.
- Z축에 평행으로 피드를 주어 5,7,4.2 로 위치한다.
- Z축에 평행으로 가로질러 5,7,4.8로 위치한다.

두번째 동작은 3단계로 움직인다. X의 위치는 9(=5+4)로 리셋이 되고 Y는 12(=7+5)가 된다.

- XY평면에 평행으로 가로질러 9,12,4.8 로 위치한다.
- Z축에 평행으로 피드를 주어 9,12,4.2 로 위치한다.
- Z축에 평행으로 가로질러 9,12,4.8로 위치한다.

세번째 동작은 3단계로 움직인다. X의 위치는 13(=9+4)로 리셋이 되고 Y는 17(12+5)이 된다.

- XY평면에 평행으로 가로질러13,17,4.8 로 위치한다.
- Z축에 평행으로 피드를 주어 13,17,4.2 로 위치한다.
- Z축에 평행으로 가로질러 13,17,4.8로 위치한다.

#### 10.7.24.3 G82 싸이클

G82 싸이클은 드릴링을 위함이다. G82 X~ Y~ Z~ A~ B~ C~ R~ L~ P~ 를 프로그램 해 본다.

- 사전동작은 위의 설명과 같다.
- Z축이 오직 현재의 피드비율로 위치에 이동한다.

- P 초만큼 머무르기 한다.
- 가로지르기 비율로 Z를 철회하여 틈새 Z로 움직인다.

#### 10.7.24.4 G83 싸이클

G83 싸이클은 깊은 구멍 뚫기나 밀링 하는 것을 위함이다. G73 를 참고 한다. 이싸이클에서 철회 동작은 구멍안에 있는 파편들을 제거한다. 이 싸이클은 Q 숫자 만큼 Z 축을 따라 “ 델타” 값 만큼 증가 한다.

G83 X~ Y~ Z~ A~ B~ C~ R~ L~ Q~ 을 프로그램 한다.

- 사전동작은 위의 설명과 같다.
- Z축을 현재의 피드 비율로 아래 방향으로 델타 만큼 움직인다.
- 빠른 동작으로 틈새 Z 까지 올라 간다.
- 빠른 동작으로 구멍의 바닥까지 내려간다음 약간 물러선다.
- 스텝 1,2,3 를 Z 위치에 갈때까지 반복한다.
- Z 축을 가로지르기 비율로 철회하여 틈새 Z 까지 올린다.

Q 숫자가 음수이거나 0 이면 에러가 발생한다.

#### 10.7.24.5 G84 싸이클

G84 싸이클은 태핑 톨로 오른손 방향 태핑 하기 위함이다. G84 X~ Y~ Z~ A~ B~ C~ R~ L~ 을 프로그램 한다.

- 사전동작 은 위의 설명과 같다.
- 속도- 피드 동조 를 시작한다.
- Z 축을 Z의 위치에 현재 피드 비율로 이동한다.
- 스펀들을 정지 한다.
- 스펀들을 반시계 방향으로 회전시킨다.
- Z 축을 틈새 Z까지 피드비율로 철회 한다.
- 속도- 피드가 싸이클이 시작되기 전에 동조 되어 있지 않다면 정지 시킨다.
- 스펀들을 정지 한다.
- 스펀들을 시계 방향으로 회전 시킨다.

스핀들은 반드시 이 싸이클을 시작 하기 전에 시계 방향으로 회전해야 한다. 만약 스펀들이 이 싸이클이 시행 되기 전에 시계 방향으로 회전 하지 않는 다면 에러가 발생한다.

이번 싸이클로, 프로그래머는 나사산이 만들어지는 피치(pitch)를 정확히 맞추기 위해 속도와 피드를 정확하게 프로그램 해야 한다. 스펀들 속도는 피드 비율 곱하기 피치와 동일하다. 예를 들어, 밀리미터당 피치(pitch)에 2개의 나사산이 있다면 활성화된 길이 단위는 밀리미터가 되고 피드 비율은 명령어 F150으로 설정되면 속도는 명령어 S300 으로 설정한다. (150 X2)

만약 피드와 속도가 반복 스위치가 활성화 되었고 100%로 설정 되어 있지 않다면, 상대적으로 아래에 값으로 설정된것이 유효할 것이다. 스피드와 속도 비율은 계속 동조 되어 있을 것이다.

#### 10.7.24.6 G85싸이클

G85 싸이클은 보어링(boring) 이나 리밍(reaming) 작업을 하기 위함이지만 드릴링이나 밀링에도 사용 할 수 있다.

G85 X~ Y~ Z~ A~ B~ C~ R~ L~ 을 프로그램 한다.

- 사전동작은 위의 설명과 같다.
- Z축을 현재의 피드 비유와 Z 위치로 움직인다.
- Z축을 틈새Z 위치로 현재의 피드 비율로 움직인다.

#### 10.7.24.7 G86 싸이클

G86싸이클은 보어링(boring)을 위함이다. 이 싸이클은 P 초 만큼 머무르기 한다. G86 X~ Y~ Z~ A~ B~ C~ R~ L~ P~ 를 프로그램 한다.



- 사전 동작은 위의 설명과 같다.
- Z 축을 현재의 피드 비율로 Z 위치로 이동한다.
- P 초 만큼 머무르기 한다.
- 스피들 회전을 멈춘다.
- Z 축을 틈새 Z 만큼 가로지르기 비율로 철회 한다.
- 스피들을 회전 했던 방향으로 다시 시작 한다.

스핀들은 이사이클이 사용되기전에 회전 해야 하고 만약 그렇지 않다면 예러가 발생할 것이다.

#### 10.7.24.8 G87 싸이클

G87 싸이클은 백 보어링(back boring)을 위함이다.

G87 X~ Y~ Z~ A~ B~ C~ R~ L~ J~ K~

이 상황은 그림 10.6 과 같이 구멍을 완전히 뚫은 다음 바닥면으로 부터 카운터 보어링을 하기 위함이다. 이렇게 하려면 L 형태의 스피들 커터의 뒷 부분을 이용하여 절삭 하는 것을 사용해야 한다.

이 싸이클은 I와 J 숫자를 사용하여 툴의 장착위치를 알려 준다. I와 J는 거리모드 셋팅에 도 불구하고 X 위치와 Y 위치로 부터 항상 증가하게 된다. 이 싸이클은 또한 K 숫자를 사용하여 Z축 카운터 보어의 제어된지점을 따라 위치를 가르쳐준다. K 숫자는 절대 거리 모드에서 현재 좌표계를 나타내는 Z 값이 되고 Z 축에서 증가하는 거리 모드에서 증가 하는 수치를 나타 낸다.

- 사전 동작은 위의 설명과 같다.
- XY 평면을 가로지르기 비율로 평행하게 이동하여 I 와 J 가 가르키는 지점에 위치 한다.
- 특정 방향에서 스피들을 정지 시킨다.
- Z 축을 가로지르기 비율로 움직여 Z를 위치 시킨다.
- XY 평면을 가로지르기 비율로 움직여 XY 좌표에 위치 시킨다.
- 스피들을 전에 회전했던 방향으로 회전 시킨다.
- Z 축을 주어진 피드 비율로 위로 움직여 K가 지시하는 위치에 놓는다.
- Z 축을 주어진 피드 비율로 다시 아래로 움직여 Z를 위치 시킨다.
- 스피들을 전과 같은 방향에서 정지 시킨다.

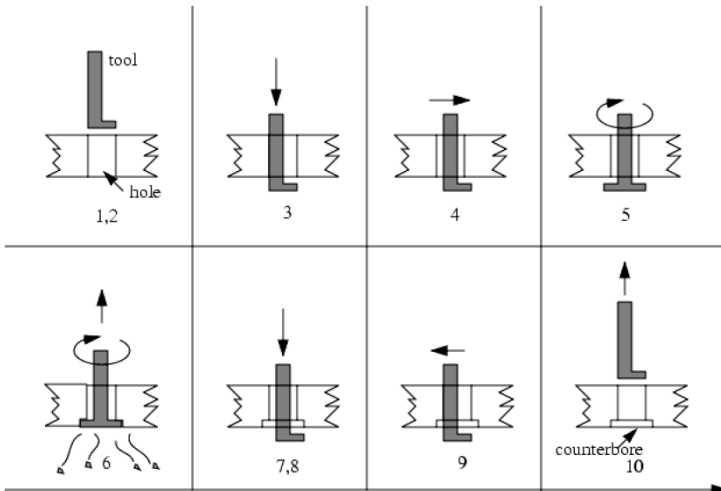


그림 10.6 G87 백 보어링 순서

- 가로지르기 비율로 XY 평면에 평행하게 이동하여 I와 J가 가르키는 방향으로 이동한다
- Z 축을 가로지르기 비율로 움직여 틸새 Z로 위치한다.
- 가로지르기 비율로 XY 평면에 평행하게 이동하여 특정 XY 지점으로 이동한다.
- 전에 회전 했던 방향으로 스피ن들을 다시 회전 시킨다.

이 싸이클을 프로그램 할때 I와 J 숫자를 선택하여 톨이 어떤 방향 위치로 정지 할때 구멍을 통하여 맞게 한다. 왜냐하면 각기 다른 커터는 다 달리 만들어졌기 때문에 분석과 실험을 하여 적절한 I 와 J 값을 결정한다.

#### 10.7.24.9 G88 싸이클

G88 싸이클은 보어링을 위함이다. 이 싸이클은 P 문자를 사용하며 이때 P는 머무르기(dwell) 하는 초 수를 가르킨다. G88 X~ Y~ Z~ A~ B~ C~ R~ L~ P~를 프로그램 해 본다.

- 사전 동작은 위의 설명과 같다.
- Z축을 위치로 오직 현재의 피드 비율로 움직인다.
- P에 나타난 초 만큼 머무르기 한다.
- 스피ن들 회전을 정지 시킨다.
- 프로그램을 정지 시켜 오퍼레이터가 스피ن들을 매뉴얼로 철회 할 수 있도록 한다.
- 스피ن들을 회전 하던 방향으로 제 시작한다.

#### 10.7.24.10 G89 싸이클

G89 싸이클은 보어링을 위함이다. 이 싸이클은 P 문자를 사용하며 이때 P는 머무르기(dwell) 하는 초 수를 가르 킨다. G89 X~ Y~ Z~ A~ B~ C~ R~ L~ P~ 를 프로그램 해 보자.

- 사전 동작은 위의 설명과 같다.
- Z축을 위치로 오직 현재의 피드 비율로 움직인다.
- P에 나타난 초 만큼 머무르기 한다.
- Z 축을 현재의 피드 비유로 철회 하여 틸새 Z로 움직인다.

#### 10.7.25 거리 모드 설정하기 - G90과 G91

Mach3 코드의 해석은 두가지 거리 모드중 하나가 될 수 있다. “ 절대” 와 “ 증분” 가 그 두가지 이다.

절대 거리 모드로 들어 가기 위해 G90를 프로그램 한다. 절대 거리모드에서 축 숫자( X, Y, Z, A, B, C)는 주로 현재 좌표계에 의한 위치를 말한다. 그 법칙에 대한 예외는 이번 섹션에서 G코드와 같이 설명 할 것이다.

증분 거리 모드로 들어 가려면, G91을 프로그램 한다. 증분 거리모드에서는 축 숫자들(X, Y, Z, A, B, C)이 번호의 현재 값으로부터 증분함을 나타낸다. I와 J 숫자는 거리모드의 설정에도 불구하고 항상 증분을 나타낸다. K 숫자는 G87 보어링 싸이클이외에는 모두 증분을 나타 내며 이때 거리 모드를 변형한다.

#### 10.7.26 IJ 모드 설정 G90.1 G91.1

G0 와 G03 코드에 있는 IJK 값의 해석은 두가지 거리 모드중 하나가 된다. “ 절대” 와 “ 증분” 이 이 두가지가 된다.

절대 IJ 모드로 들어 가기 위해서 G90.1을 프로그램 한다. 절대 거리 모드 에서, IJK 숫자들은 현재 활성화된 좌표계에 의한 위치로 나타내 진다.

증분 IJ 모드로 가기 위해서 G91.1 을 프로그램 한다. 증분 거리 모드에서, IJK 숫자들은 주로 현재 제어점 으로부터 증분을 나타 낸다.

이 모드의 정확하지 않은 설정은 톨 패스에서 정확하지 않은 호를 그리게 된다.

### 10.7.27 G92 옵셋 G92, G92.1, G92.2, G92.3

이번 섹션은 자세한 내용을 정확히 모른다면 그리고 어떤 축이라도 옵셋이 적용 됐다면 절대로 사용 하지 않기를 바란다.

현재의 지점을 원하는 좌표에 놓기 위해(동작없이) G92 X~ Y~ Z ~ A ~ B~ C~ 를 프로그램 하여 원하는문자가 원하는 축의 숫자가 되게 한다. 모든 축 문자는 옵션이지만 적어도 하나는 반드시 사용되어야 한다. 만약 축 문자가 주어진 축에 사용 되지 않으면 현재 지점에서 축의 좌표는 바뀌지 않는다. 만약 모든 축 문자가 생략 되었다면 에러가 난다.

G52 와 G92 는 Mach3에서 공통적인 내부 메터니즘을 사용하여 함께 쓰이지 않을 수도 있다.

G92가 실행된다면, 현재 활성화된 좌표계의 원점은 이동한다. 이렇게 하기 위해 원점 옵셋을 조정해야 하여 현재 지점의 좌표가 움직인 좌표에 대하여 G92를 포함하는 라인 선상에 명시 되어야 한다. 추가로, 파라미터 5211에서 5216까지는 X,Y,Z,A,B, C 축 옵셋으로 셋팅한다. 축의 옵셋은 원점이 움직여야하는 크기가 되어 축의 제어된 지점의 좌표가 특정한 값을 갖게 한다.

여기 예제가 있다. 현재 지점이 X=4라고 한다 그리고 현재 X축 옵셋은 0이 되고 G92 X7은 X축 옵셋을 -3 그리고 5211을 -3으로 셋팅 한다. 그리고 현재 지점의 X 좌표는 7이 되게 한다.

축 옵셋은 고정체 좌표계를 사용하는 절대좌표 모드에서 동작이 명시 된다면 항상 사용된다. 그러므로 모든 고정체 좌표계는 G92에 의해 영향을 받는다.

증분 거리 모드는 G92 에 영향을 받지 않는다.

0이 아닌 옵셋은 G92가 호출 됐을 때 이미 영향을 받았을 수도 있다. 새로운 값이 적용되기 전에 사실상 버려지기 때문이다. 수학적으로 각 옵셋의 새로운 값은 A+B이며 이때 A는 전의 옵셋값이 0 일때의 옵셋값을 나타내고 B는 전의 옵셋값을 나타낸다. 먼저 번의 예를 들어, 현재 지점의 X 값은 7 이었다. 만약 G92 X9가 프로그램 된다면, 새로운 X 축 옵셋은 -5가 된다.  $[[7-9]-3]$

축 옵셋을 0으로 리셋하려면, G92.1 혹은 G92.2 를 프로그램하여 5211 파라미터에서 5216 파라미터까지 0으로 셋팅한다 이때 G92.2 가 현재 값이 되도록 남겨진다.

축 옵셋 값이 파라미터 5211에서 5216에 주어진 값으로 셋팅 되기 위해 G92.3를 프로그램 한다.

축 옵셋을 프로그램에서 설정 할 수 있고 같은 옵셋을 다른 프로그램에 사용 할 수도 있다. G92를 처음 프로그램에서 사용하고 이때 5211에서 5216까지 파라미터들을 셋팅한다. G92.1을 처음 프로그램의 나머지에서 사용하지 않도록 한다. 파라미터 값들은 처음 프로그램을 종료할 때 저장이 되고 두번째 프로그램이 시작 될 때 복구 된다. G92.3를 두번째 프로그램 시작 근처에 사용해 본다. 그러면 처음 프로그램에 저장된 옵셋을 복구 시킬것이다.

### 10.7.28 피드 비율 모드 설정 - G93, G94 그리고 G95

3가지 피드 비율 모드가 있다. 시간의 반비례모드, 분당 길이 단위(unit per minute)모드 와 스피들의 회전 당 단위 모드가 이 3가지 이다. G93를 프로그램 하여 시간의 반비례 모드를 시작 한다. G94를 프로그램하여 분당 길이 단위 모드를 시작한다. G95를 프로그램하여 회전당 단위 모드를 시작한다.

시간의 반비례모드에서 F 문자의 의미는 동작이 몇분안에 끝나는 것을 의미한다[1/F 숫자] 예를 들어 F 숫자가 2.0 이면 움직임은 30 초 안에 끝나야 한다.

회전 당 단위 피드 비율 모드에서 라인에 있는 F 문자는 제어점이 분 당 인치, 분당 밀리미터 혹은 분당 각도의 동작으로 움직이는 것으로 해석되며 이때 길이 단위와 어느 축이 움직이는 가에 따라 다르다.

회전 피드 비율 당 단위는 위의 내용과 유사하다.

시간의 반비례 피드비율모드가 활성화 되었다면 F 문자는 G1, G2, G3모션이 있는 모든 라인에 있어야 하고 G1, G2, G3모션이 없는 라인의 F 문자는 생략 된다. 이 모드에서 G0 동작은 영향을 주지 않는다. 그리고 이 모드가 활성화 되어 있고 G1, G2, G3 라인에 F 문자가 없다면 에러가 난다.

### 10.7.29 고정 싸이클 회귀 높이 - G98, G99

스핀들이 고정 싸이클(canned cycle)에서 회귀하는 동안 얼마큼 회귀 하는가에 대한 선택이 있다.

1. 선택된 면에 직각으로 회귀하여 R문자에 나타나는 위치로 간다. 혹은
2. 선택된 면에 직각으로 회귀하여 고정 싸이클이 시작하기 전의 축의 위치로 간다.

1번을 실행하려면 G99을 프로그램하고 1번을 실행하려면 G98을 프로그램한다. R문자는 절대거리 모드와 증분 거리 모드에서 다른 뜻을 포함하는 것을 잊지 않도록 한다.

## 10.8 내장된 M코드

M 코드	의미
M0	프로그램 정지
M1	선택적으로 프로그램 정지
M2	프로그램 끝
M3/4	스핀들 시계/반시계 방향으로 회전
M5	툴 교체(두개의 매크로에 의해)
M6	분무기 냉매제 on
M7	분무기 냉매제 off
M8	유체 냉매제 on
M9	모든 냉매제 off
M30	프로그램을 끝내고 다시 감기
M47	첫 라인부터 프로그램 다시 시작
M48	속도와 피드 번복 활성화
M49	속도와 피드 번복 불능화
M98	서브루틴 호출
M99	서브루틴/반복으로부터 회귀

그림 10.7 내장된 M 코드

Mach3에서 해석되는 M 코드는 위의 그림 10.7 과 같다.

### 10.8.1 프로그램 정지와 끝 M0, M1, M2, M30

진행 중인 프로그램을 일시적으로 정지 하려면 M0를 프로그램 한다. (이때 정지 스위치 옵션 설정을 무시하고) 진행 중인 프로그램을 일시적으로 정지 하려면 M1을 프로그램 한다. (이때 정지 스위치 옵션 설정이 on 일때만)

MDI에 M0, M1을 했다하더라도 즉각 눈에 띄게 나타나지는 않을 것이다. 왜냐하면 MDI의 일반적 실행은 입력한 라인을 마치고 난 후 정지 하기 때문이다.

프로그램 이 M0와 M1에 의해 정지 했다다면 cycle start 버튼을 누르는 것은 다음 라인에서부터 프로그램을 다시 시작 하게 한다.

프로그램을 끝내려면 M2, M30 를 프로그램 한다. M2는 다음 라인을 M2라인으로 실행시키도록 남겨둔다. M30 는 G 코드 라인을 되돌리기 한다. 이런 명령어들은 다음 효과가 있고 configure> logic 대화 상자의 선택된 옵션에 의존한다.

- 축움셋은 G92.2와 같이 0으로 설정되고 원점 움셋은 G54와 같이 기본으로 설정 된다.
- 선택된 좌표면은 XY 로 설정 된다. (G17)
- 거리 모드는 절대 모드로 설정된다. (G90)
- 피드 비율 모드는 분당 단위 모드로 설정된다. (G94)

- 피드 와 속도 번복은 ON 으로 설정된다. (M48)
- 커터 보정은 Off 로 설정 된다.(G40)
- 스핀들이 정지 된다. (M5)
- 현재의 동작 모드는 G1이 된다.(G1)
- 냉매제는 off 가 된다(M9)

더 이상의 파일의 코드는 M2 혹은 M3 명령어가 실행된 후로 이루어 지지 않는다. Cycle start 버튼을 누르면 프로그램이 계속 되거나(M2) 파일의 처음 부분에서 다시 시작 된다.(M30)

### 10.8.2 스핀들 컨트롤 M3, M4, M5

스핀들을 현재 프로그램된 속도에 시계 방향으로 회전 하려면 M3를 프로그램 한다.  
또 반대로 회전 시키려면 M4를 프로그램 한다.

PWM 혹은 스텝/방향 스핀들을 위한 스피드는 S 문자에 의해 프로그램 된다. On/off 스핀들 제어는 기어링/풀리에 의해 설정 된다.  
스핀들을 정지 시키려면 M5를 프로그램 한다.

스핀들 속도가 0으로 설정되어 있다면 M3 혹은 M4를 사용 할 수 있다.

### 10.8.3 톨 교체 M6

톨교체에 대한 요청은 무시 되어서는 않된다. Mach3는 명령어가 나타나면 매크로 M6Start를 호출 한다. 그리고 cycle start 버튼이 눌러 질 때까지 기다리고 매크로M6End 을 실행 하여 파트프로그램을 계속 진행한다.  
매크로에 비주얼 베이직 코드를 넣어 장비의 톨 교체 를 직접 할 수 있도록 해 본다.

톨 교체가 무시 되도록 config>logic 에서 설정되면 M6는 아무 효과가 없다.

### 10.8.4 냉매제 제어 M7, M8, M9

위의 표를 참고 한다.  
어떤 종류의 냉매제를 사용 하든 이 명령어를 사용 할 수 있다.

### 10.8.5 첫 라인에서 다시 시작 하기. M47

파트프로그램에서 M47 을 만나게 되면 처음 라인부터 계속 진행 할 것이다. M47이 서브루틴에서 실행 되면 에러가 난다.  
Pause 나 Stop 버튼을 눌러 임시정지 혹은 정지 할 수 있다.

### 10.8.6 제어 번복 하기 - M48, M49

속도와 피드 번복을 가능하게 하려면 M48을 프로그램 한다. 두가지 모두 번복 불가능하게 하려면 M49을 프로그램 한다.

### 10.8.7 서브루틴 호출 M98

번역 생략

### 10.8.8 서브루틴에서의 회귀

## 10.9 매크로 M코드

### 10.9.1 매크로 개요

만약 위의 내장된 M 코드 이외의 코드를 사용한다면 Mach3는 “ Mxx.M1S “ 라는 파일을 매크로 폴더 안에서 찾으려고 할 것이다. 파일을 찾으면 비주얼베이직 스크립트를 실행한다. Operator>Macros 메뉴 아이템에는 현재 설치한 매크로를 보여 주는 대화상자가 나타난다. 그리고 편집 저장 등등 할 수 있게 한다. 대화 상자는 또한 Help 버튼이 있어 Mach3로 불러 들일 수 있는 비주얼베이직 기능을 보여준다. 예를 들어 축의 위치, 축이동등 입출력 신호를 보내고 제어한다.

새로운 매크로는 매모장 같은 외부 편집기 프로그램을 이용하여 매크로 폴더에 저장 하거나 Mach3에 기존의 매크로를 로드 하여 완전히 새로 작성하여 다른 이름으로 저장 할 수 있다.

## 10.10 기타 입력 코드

### 10.10.1 피드 비율 F 설정

번역 생략

### 10.10.2 스피들 속도 S 설정

번역 생략

## 10.11 에러 처리하기

Mach3에서 만약 명령어가 실행 되지 않는다면 입력하는 중 타이핑이 잘 되었는지 확인 해 본다.. 종종 0과 O를 바꿔서 타이핑 하는 경우가 많다. Mach3는 스스로 과잉운동이나 과잉 속도, 피드등을 체크 하지 않는다. 그러나 작업한 내용을 직접 확인하기 바란다.

## 10.12 실행 순서

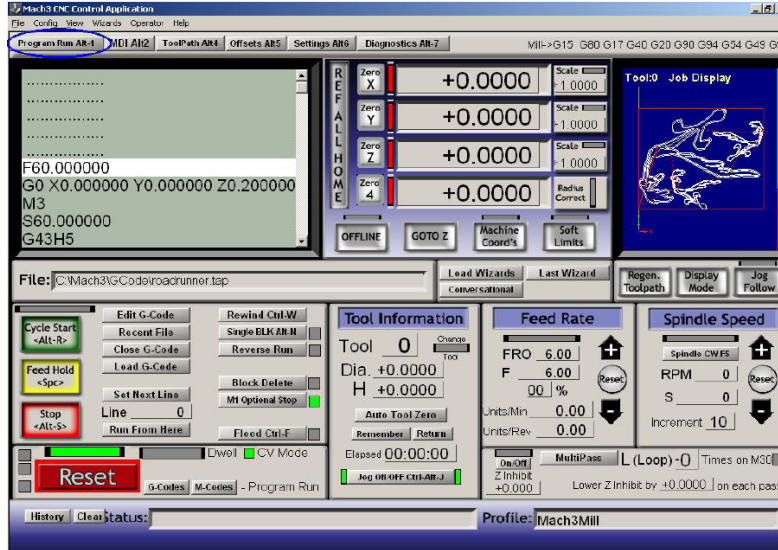
라인 상에서 항목의 실행 순서는 장비의 효율과 안전을 위해서 아주 중요하다. 같은 라인 상에 항목들이 있다면 아래 10-9 와 같은 순서로 실행한다.

순서	항목
1	코멘트
2	피드 비율 모드 설정(G93, G94, G95)
3	피드 비율 설정(F)
4	스핀들 스피드 설정(S)
5	툴 선택
6	툴 체인지(M6) 그리고 M code 매크로 실행
7	스핀들 On/Off(M3, M4, M5)
8	냉매제 On/Off(M7, M8, M9)
9	번복(override) 가/불가(M48, M49)
10	머무르기(dwell) G4
11	활성 좌표면 설정 G17, G18, G19)
12	길이 단위 설정(G20, G21)
13	커터반경 보정 on/off(G40, G41, G42)

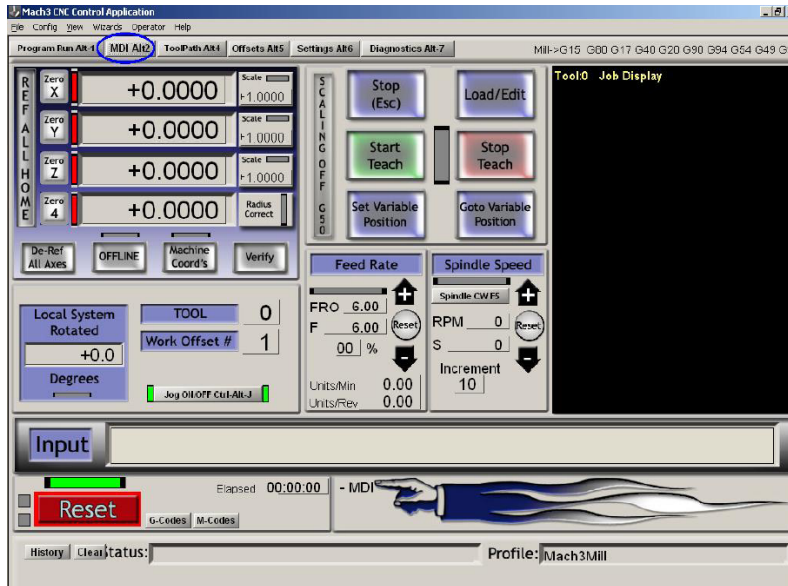
14	툴 테이블 옵션 on/off (G43, G49)
15	고정자 테이블 선택(G54-G58 & G59 P~)
16	경로 제어 모드 설정(G61, G61.1, G64)
17	거리 모드 설정(G90, G91)
18	고정 싸이클 회귀 높이 모드 (G98, G99)
19	홈, 혹은 좌표계 데이터 변환 (G10)혹은 옵션설정 (G92, G94)
20	동작 실행(G0, G1, G2, G3, G12, G13, G80에서 G89까지)
21	정지나 반복(M0, M1, M2, M3, M47, M99)

**10.9 테이블 - 라인상의 실행 순서.**

## 11. 부록 1 - Mach3 스크린 샷

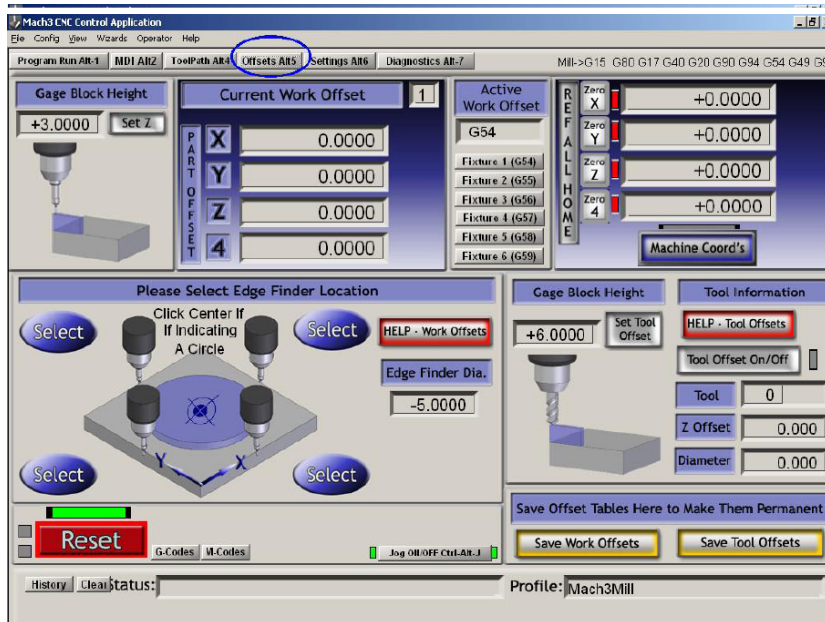


Program Run 스크린

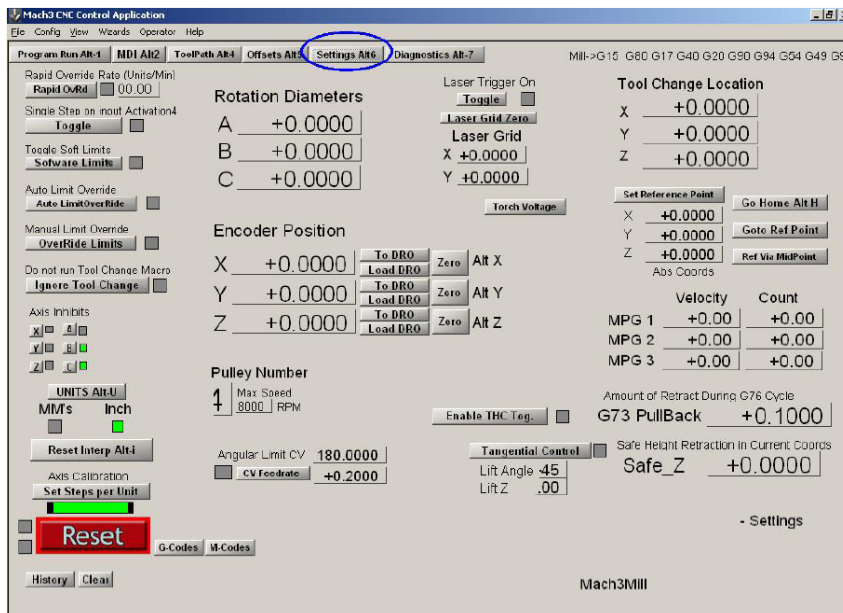


MDI 스크린

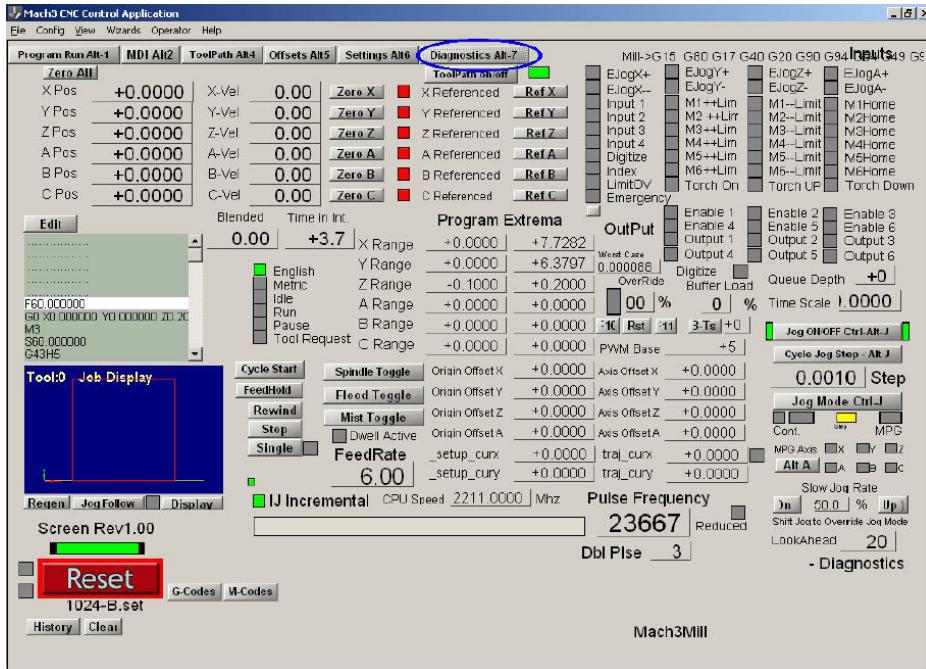




오프셋(offset)스크린



셋팅 스크린



검사 (Diagnostics) 화면

## 부록 2 - 예제 개략도

### 12.1 릴레이를 사용한 Estop 과 리밋

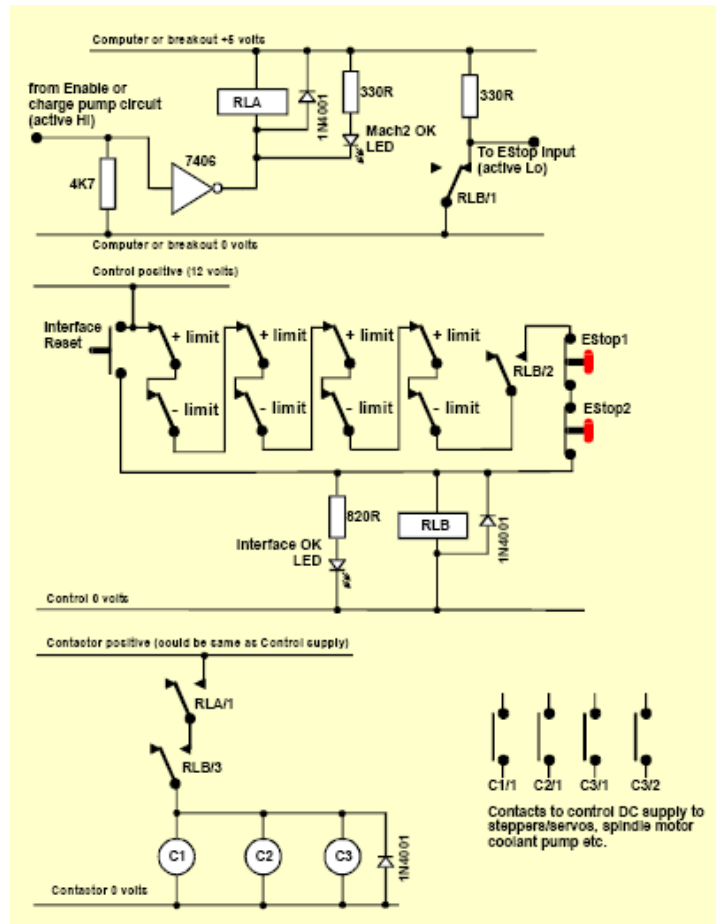


그림 12.1 Estop 과 리밋 스위치 연결

주의 :

1. 이 회로는 외부적으로 연결된 리밋 스위치를 위한 가능한 해결책 중의 하나다. 기존 스위치가 필요하다면 이런 것들은 Mach3에 별도로 연결되어야 한다.
2. 릴레이 접선은 전력이 없는 위치에 있다. 리밋 스위치와 누름 버튼은 작동 되지 않는다.
3. 인터페이스 리셋을 누르고 있으면 Mach3 리셋 버튼은 눌러 질 것이고 리밋 스위치로부터 조그 되어 스위치 off 될 것이다. 인터페이스 리셋은 걸리게 된다.
4. 릴레이 A는 하나의 NO 접선 하게 된다. 이것은 5V, 최소 150옴이 되어야 한다. Omron G6H

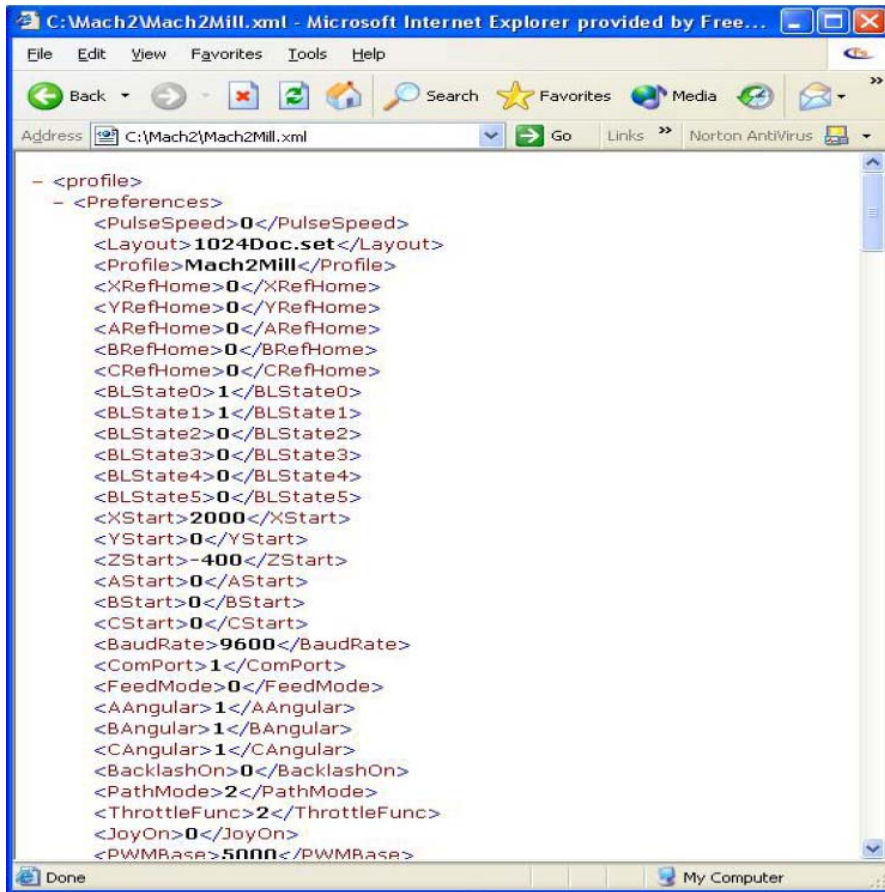
- 2100-5 모델이 적절할 것이다.
5. 릴레이B는 하나의 NC 와 2 NO 접선이 필요하다. 적절한 전압을 공급하기 위해 어떤 형식의 코일을 사용할 수 있다. 이것의 공통적으로 PC 0 볼트 선상에 있지 말아야 하는데 그것은 Estop 과 리미트 스위치의 긴 선으로 노이즈를 갖을 수 있기 때문이다. 5암페어 220V AC 정도 되는 Omron MY4 시리즈 모델이 적절할 것이다.
  6. LED는 옵션이지만 어떤일이 진행 되는지 알기 위해 유용하게 사용 될수 있다. 인터페이스 OK LED 전류를 제한 하는 저항기는 24볼트가 쓰인다면 1.8킬로옴이 된다.
  7. 코일 전압이 적합하다면 접촉기는 컨트롤을 이용하여 플러스와 뉴트럴전압을 선택한다.
  8. 접촉기의 배열은(C1, C2, C3) 드라이버의 전원 공급장치와 장비의 모터 결선에 달려 있다. 즉 각각적인 정지를 확실히 하기 위해 콘덴서를 고르게 한다음 서보나 스텝퍼로 DC 전압을 바꾼다. 스프링이나 냉매제를 위한 모터를 다시 결선 하여 제어 접촉기가 회로에 갑자기 전압을 차단 하는 것을 방지 하고자 할 수 있다. 주 접촉기 작동 이후로 모터의 도선을 스위칭 한다. 전기에 대해 모른다면 꼭 충고를 구하기 바란다.
  9. 릴레이와 접촉기 코일에 걸려 있는 다이오드는 코일에서 전류를 바꿀때 역류 emf 를 흡수하기 위해 필요하다.

### 13. 부록 3 - 설정 저장하기

Mach3의 설정은 항상 종이에 직접 메모를 해 두기 바란다.

Mach3의 완전한 설정은 상세한 많은 내용을 포함하고 있다. 컴퓨터를 업그레이드 한다면 하나씩 다시 처음부터 설정하고 싶은 생각은 없을 것이다.

Mach3 프로파일들은 .XML 파일로 되어 있고 보통 Mach3 폴더 안에 있다. 예를 들어 밀링 머신의 프로파일은 mach3mill.xml 형식으로 되어 있다. 그러므로 이 .XML 파일은 따로 복사하여 잘 보관해 두기 바란다. .XML 파일을 더블 클릭 하면 인터넷 웹 브라우저를 통해 파일이 열리고 아래 그림과 같이 보일 것이다.



XML 파일은 메모장 같은 텍스트 편집기로 열릴 수 있지만 결코 권하지 않는다. 프로파일 파일은 차후 장비의 문제를 알기 위해 ArtSoft 사에 이메일로 보내 주면 기술적 도움을 받는데 좋은 자료가 될 것이다.