

T.C.

GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ



12 12 93

CAD/CAM SİSTEMLERİNDEN 5 EKSEN CNC
KODLARINA VERİ DÖNÜŞTÜRÜLMESİ, BİR
UYGULAMA YAZILIMININ GELİŞTİRİLMESİ

Bülent KAYA

YÜKSEK LİSANS TEZİ

TASARIM VE İMALAT MÜHENDİSLİĞİ

ANABİLİM DALI

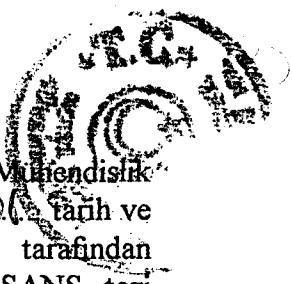
T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANİASYON MERKEZİ

TEZ DANIŞMANI

Doç.Dr. E. Murat ESİN

GEBZE

2002



.....Bülent Kaya.....'ın tez çalışması, G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 20.03.2001 tarih ve 2001/25 sayılı kararıyla oluşturulan jüri tarafından Tecrüme ve İnceleme Mühendislik Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE
(Tez Danışmanı)

: Doç. Dr. E. Murat Eşih
Uludağ

ÜYE

: Doç. Dr. Fehmi Erzinanlı
Gmi

ÜYE

: Ural. Doç. Dr. Babür Özcelik


ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 14.01.2002 tarih ve 2002/2 sayılı kararı.



ÖZET

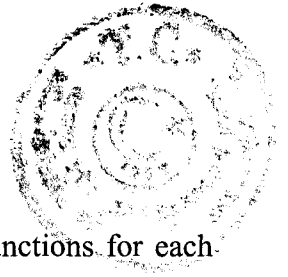


CNC tezgahlarında parça işlemek için, kesici takımın fonksiyonları her adımı temsil edecek şekilde bir dizi kodla ifade edilir. Özellikle 5 eksen CNC tezgahlar için bu kodlarının hatasız yazılması oldukça zordur. Günümüzde parçanın tasarımı için bilgisayar destekli teknik çizim programları yaygınca kullanılmaktadır. Tasarımı yapılan parçanın bilgileri, bilgisayar destekli imalat sistemleri vasıtasıyla bir dizi genel kesme veri dosyası şekline dönüştürülür. Bu dosya daha sonra, postprosesör olarak adlandırılan yazılımla, belirli bir tezgah-kontrolör kombinasyonu için işlenir. Her bir tezgah-kontrolör kombinasyonu için ayrı bir postprosesöre ihtiyaç duyulmaktadır.

En son kuşak Bilgisayar Destekli İmalat (CAM) yazılımları, birisi Postprosesör Generator yazılımı olmak üzere, birçok özelliği beraberinde taşır. Bu yazılım, herhangi bir tezgah-kontrolör kombinasyonu için, son kullanıcının yazılım programlama tecrübesi gerektirmeden, özel postprosesörler oluşturmasını sağlar. Bununla birlikte yazılımın pahalı olması, özellikle 5 eksen işleme için esnek ve kullanımının kolay olmaması gibi birtakım dezavantajları söz konusudur.

Bu çalışmada, belirli bir CAM sistemi ve 5 eksen universal bir işleme merkezi için Postprosesör program geliştirilmiştir. Programın geliştirilmesi aşamasında, C++ Builder 3.0 genel programlama yazılımı, homojen transformasyon matrisleri ve ters kinematik kullanılmıştır. Çalışma sonucunda elde edilen veriler, tezgahta başarılı bir şekilde test edilmiştir.

SUMMARY



For machining a component on the CNC machine tools, the cutter tool functions for each step are represented by a set of codes. Writing these codes correctly is difficult, especially for 5 axis CNC machine tools. Nowadays, Computer Aided Desing (CAD) software packages are commonly used to desing a component. In Computer Aided Manufacturing (CAM) systems, information about the component is converted into a generalised set of cutter location data. This data is then processed for a specific machine tool-controller combination by a software called as Postprocessor. For each combination of machine tool and controller, a separate postprocessor is required.

The latest generation of Computer Aided Manufacturing (CAM) software packages contain many features, one of which is a customised postprocessor generator. This additional software allows the end user, without having any software programming expertise, to create postprocessors for any combination of numerical control machine tool-controller combinations. However, this kind of software packages are expensive, quite inflexible, and not easy-to-use especially for 5 axis machining.

In this study, a Postprocessor was developed for a specific CAM software package and 5 axis machine tool. Borland C++ Builder general programming language, homogeneous transformation matrices and inverse kinematics were used for development process of the Postprocessesor. The result data taken from the Postprocessor was succesfully tried on the 5 axis machine tool.

TEŞEKKÜR



C programlama dili ve bilgisayar grafikleri konusuyla tanıştıran, her konuda destekleyen,
Sayın Doç. Dr. E. Murat Esin' e;

Bölümdeki CNC tezgahla ilgili eğitim ve tez çalışmalarına verdikleri destekleri nedeniyle
Sayın Doç. Dr. Fehmi Erzincanlı, Sayın Yrd. Doç. Dr. Babür Özçelik, Sayın Yrd. Doç. Dr.
A. Zafer Şenalp ve Sayın Dr. Hasan Kurtaran' a;

Yardımlarından dolayı Sayın Yrd. Doç. Dr. Salim Öncel' e, Tasarım&İmalat ve Bilgisayar
Mühendisliğinde görevli tüm arkadaşlarıma teşekkür ederim.

Ayrıca manevi desteklerinden dolayı sevgili eşim Şule Kaya' ya da sonsuz teşekkürler...

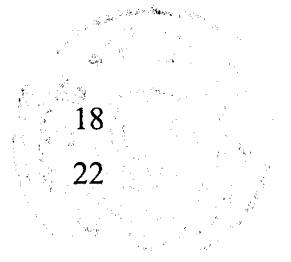
İÇİNDEKİLER DİZİNİ

SAYFA NO



ÖZET	iv
SUMMARY	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR DİZİNİ	xi
ŞEKİLLER DİZİNİ	xii
TABLolar DİZİNİ	xiv

1.GİRİŞ	1
1.1. Bilgisayar Bütünleşik Tasarım İmalat Sistemlerinden CNC tezgahlarına veri aktarımı	1
1.2. Postprosesör Yazılımları	2
2. SAYISAL KONTROLLÜ TEZGAHLAR	5
2.1. Bir NC Tezgahın Konfigürasyonu	6
2.2. NC Sistem Tipleri	7
2.3. NC/CNC/DNC	8
2.4. Parça Programlamanın Temel Kavramları	9
2.4.1. Koordinat Sistemleri	10
2.4.2. Parça Programlama	11
2.4.3. Parça Program Yapısı	12
2.5. Parça Programlama Yöntemleri	14
2.5.1 Elle Parça Programlama	15
2.5.2 Bilgisayar Yardımlı Parça Programlama	15
2.5.3 APT Dili	17



2.5.3.1. Geometrik Komut Satırları

18

2.5.3.2. Hareket Komut Satırları

22

2.5.4. CAD Veritabanından Parça Programlama

27

2.5.4.1. Takım Yolu Oluşturulması

30

2.5.4.2 Takım Yolu Benzetimi Ve Doğrulaması

37

3.CAD/CAM SİSTEMLERİNDEN CNC TEZGAHLARINA

VERİ AKTARIMI

41

3.1. Giriş

41

3.2. Bilgisayar Destekli 5 Eksen İşleme

41

3.2.1. NC Freze Tezgahlarının Sınıflandırılması

42

3.2.2. 5 Eksen İşleme İhtiyacı

42

3.3. 5 Eksen Tezgahın Elemanları

44

3.3.1. 5 Eksen Tezgahın Konfigürasyonu

44

3.3.2. Makina Kontrolörü

45

3.3.3. CAD/CAM Yazılımı

46

3.3.4. Personel Eğitimi

47

3.4. 5 Eksen Freze Tezgahlarının Kinematığı

48

3.4.1. Eğilmeli-Döner Tabla Tipinde (RRTTT) Freze

Tezgahları İçin Düz Kinematik

50

3.4.2 Eğilmeli-Döner Tabla Tipinde Bir Tezgahın Ters Kinematığı

53

3.4.3. Bilek Tipi (TTTRR) Freze Tezgahının Düz Kinematığı

56

3.4.4. Bilek Tipi Freze Tezgahının Ters Kinematığı

58

3.5. Takım Kesme Dosyası ve Postprosesör Kavramı

60

3.5.1. Özel Postprosesörler

65

3.6. Özel Bir Postprosesörün Tasarımı

67

3.7. Tezgah Tanımlanma Safhası

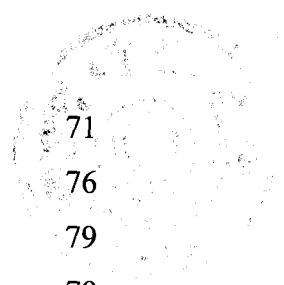
67

3.8. Kesme Dosyası Özellikleri

69

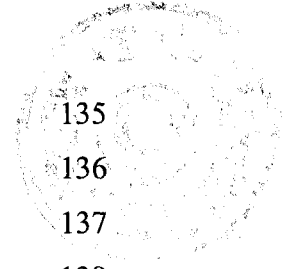
3.8.1. Lineer Hareketler

69

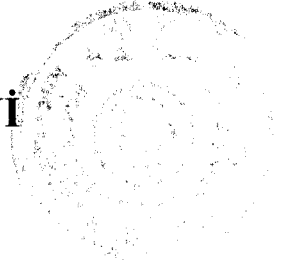


3.8.2. Dairesel Hareketler	71
3.8.3 Kapalı Döngüler	76
3.8.3.1. Operasyonun derinliği parametresi (DEPTH)	79
3.8.3.2. Kesme Hızı Parametresi (FEED)	79
3.8.3.3. Boşluk Yüksekliği (CLEAR)	80
3.8.3.4. RETURN Opsiyonu	81
3.8.3.5. RAPTO Opsiyonu	81
3.8.3.6. DWELL Opsiyonu	81
3.8.3.7. ORIENT Opsiyonu	82
3.8.4. Yardımcı Komutlar	94
3.9. Özel Postprosesörün Optimizasyonu	106
 4. SONUÇLAR/ÖNERİLER	 107
 KAYNAKLAR	 108
 ÖZGEÇMİŞ	 109
 EKLER	 110
EK1. EIA Standartlarında G Kodları	111
EK2. EIA Standartlarında M Kodları	112
EK3. Elle Programlamada Kullanılan Müsvedde (Manuscript) Örneği	113
EK4. ISO 3592-1978 ve ISO 4343-1978 Kesme Dosyasında Komut Kayıt Sınıfları	115
EK5. CATIA Yazılımı Kesme Dosyası Örneği	116
EK6. SIMATRON Yazılımı Kesme Dosya Örneği	118
EK7. EUCLID Yazılımı Kesme Dosyası Örneği	120
EK8. MASTERCAM Yazılımı Kesme Dosyası Örneği	122
EK9. UNIGRAPHICS Yazılımı Kesme Dosyası Örneği	124
EK10. ProEngineer Yazılımı Kesme Dosyası Örneği	126
EK11. DECKEL MAHO DMU 60P Üniversal İşleme Merkezi Teknik Özellikleri	128
EK12. POINTS_ONLY Formunda Örnek Çıktı Kesme Dosyası.	130
EK13. POINTS&ARC Formunda Örnek Çıktı Kesme Dosyası.	132
EK14. APT Formunda Örnek Çıktı Kesme Dosyası.	134

EK15. ARC_ONLY Formunda Örnek Çıktı Kesme Dosyası.	135
EK16. DRILL Komutu İçin Kesme Ve Makina Veri Dosya Örneği	136
EK17. DEEP Komutu İçin Kesme Ve Makina Veri Dosya Örneği	137
EK18. BRKCHP Komutu İçin Kesme Ve Makina Veri Dosya Örneği	138
EK19. THRU Komutu İçin Kesme Ve Makina Veri Dosya Örneği	139
EK20. FACE Komutu İçin Kesme Ve Makina Veri Dosya Örneği	140
EK21. BORE Komutu İçin Kesme Ve Makina Veri Dosya Örneği	141
EK22. REAM Komutu İçin Kesme Ve Makina Veri Dosya Örneği	142
EK23. CSINK Komutu İçin Kesme Ve Makina Veri Dosya Örneği	143
EK16. TAP Komutu İçin Kesme Ve Makina Veri Dosya Örneği	144
EK25. ProEngineer PPRINT Tablosu Örneği	145
EK26. INCH ve MM Komutlarına Örnek, Kesme Ve Makina Kontrol Veri Dosyaları	146



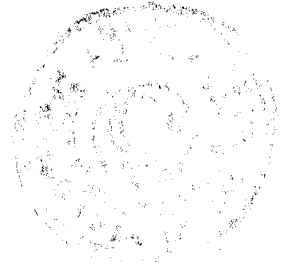
SİMGELER VE KISALTMALAR DİZİNİ



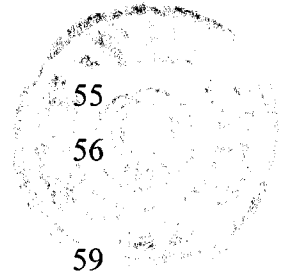
Simgeler

NC	SAYISAL KONTROL (NUMERICAL CONTROL)
MCU	Makina Kontrol Ünitesi (Machine Control Unit)
DPU	Veri İşleme Ünitesi (Data Processing Unit)
CLU	Döngü Kontrol Ünitesi (Control Loop Unit)
CNC	Computer Numerical Control
CPU	Central Processor Unit
DNC	Doğrudan-Dağıtık Sayısal Kontrol (Direct-Distributed Numerical Control)
ASCII	American Standar Code For İnformation Exchange
ISO	International Standarts Organization
EIA	Electronic Industries Association
EOB	Satır Sonu Karakteri
APT	Automatically Programmed Tools
CAD	Bilgisayar Destekli Tasarım (Computer Aided Desing)
CAM	Bilgisayar Destekli İmalat (Computer Aided Manufacturing)
CLDATA file	Kesme Veri Dosyası (Cutter Location Data File)

ŞEKİLLER DİZİNİ



<u>Sekil</u>	Sayfa
1.1. Tasarım Ve İmalat Süreci Elemanları	2
2.1. NC Tezgahın Konfigürasyonu	6
2.2. DNC Yapısı	9
2.3. Sağ Ve Sol El Koordinat Sistemleri	11
2.4. CNC Sistemlerinde Takım Yarıçap Telafisi	15
2.5. APT Dilinde Nokta Tanımı	19
2.6 APT Dilinde Çizgi Tanımlaması	20
2.7. APT Dilinde Çizgi Tanımlama	21
2.8 APT Dilinde Daire Tanımlaması	22
2.9. APT Dilinde Üç Yüzey Tanımlaması	23
2.10. APT Dili TANTO Niteleyici Tanımlaması	24
2.11. TANTO Niteleyicisinin Kullanımı	25
2.12. Kesici Takım Bağlı Hareket Komutları	26
2.13. 2 Ve 3 Eksende Yapılan Freze İşlemleri	29
2.14. Birinci Tip Kaba Boşaltma İşlemi	31
2.15 Cep İşleme İçin Takım Yolu Çıkarılması	31
2.16. Çeşitli Tip Takım Yolları	32
2.17. Paso Genişliği Ve Yüzey Pürüzlülüğü	33
2.18. Adım Uzunluğu	33
2.19. Kesme Temas Noktaları Toleransları	34
2.20. Küresel Uçlu Bir Takım İçin CL Noktası	34
2.21 Kesici Takımın İstenmeyen Hareketi	36
2.22. Takımın Komşu Yüzeydeki Dalma Hareketi	37
2.23. Takımın Komşu Yüzeydeki Dalma Hareketinin Engellenmesi	37
2.24. Temel Takım Yolu Simülasyonu	39
2.25. Benzetimde Nokta Vektör Tekniği.	40
3.1. 5 Eksen Tezgahta Kesicinin İş Parçasına Göre Konumlanması	43
3.2. 5 Eksen Tezgah Tipleri	49
3.3. 5 Eksen Tezgahtaki Koordinat Sistemleri	50
3.4. Eğilmeli-Döner Tabla Tip Tezgah İçin Doğru A ve C Rotasyon Seçimi İçin	



Yaklaşım Çeyreği	55
3.5. 5 Eksen Bilek Tip Freze Tezgahının Kinematiği	56
3.6. Bilek Tip Tezgah İçin Doğru A Ve C Rotasyon Seçimi İçin Yaklaşım Çeyreği	59
3.7. Tasarım Ve İmalat Sürecinde Postprosesörün Yeri	61
3.8. 2000 Sınıfı Kayıt Yapısı	62
3.9. Postprosesör Generator Yazılımlarının Genel Yapısı	64
3.10. Deckel-Maho DMU 60P İçin Koordinat Sistemi	68
3.11. RAPID Ve GOTO Komutları İçin Kullanılan Algoritma	66
3.12. Bir Yayın Tolerans Değeri İle Doğru Parçalarına Bölünmesi	72
3.13. Dairesel Enterpolasyon İçin Kullanılan Algoritma	75
3.14. DRILL Döngüsü Tanımlaması	83
3.15. DRILL Döngüsü İçin Kullanılan Algoritma	84
3.16. DEEP Komutu Tanımlaması	86
3.17. DEEP Komutu İçin Uygulamada Kullanılan Algoritma	86
3.18. FACE Komutu İçin Kullanılan Algoritma	89
3.19. BORE Komutu Ve Makina Kodlarında Karşılığı	90
3.20. REAM Komutu Tanımlanması	91
3.21. TAP Komutu Tanımlaması	93
3.22. LOADTL Komutu İçin Kullanılan Algoritma	96
3.23. Takım Eksen Vektörü	98
3.24. Doğru Rotasyon Değerleri Analizi İçin Tasarlanan Test Parçası	100
3.25. XY Düzleminde $A < 30^\circ$ İçin Kesin Rotasyon Analizi	101
3.26. XY Düzleminde $A > 30^\circ$ İçin Kesin Rotasyon Analizi	101
3.27. ZX Düzleminde A İçin Doğru Rotasyon Değer Analizi	102
3.28. Postprosesör Programındaki Ana Form Düzeni	105

TABLÖLAR DİZİNİ



Tablo

Sayfa

3.1. Kesme Dosyasında Bulunan Özel Karakterler

63

3.2. Düzlemlere Göre Dairesel Hareket Yönü

74

3.3. CAM Sistemleri Kapalı Döngü Tipleri

76

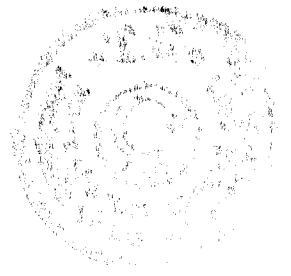
3.4. Soğutma Sistemi Kontrol Kodları

94

3.5. Tezgah Mili Kontrol Kodları

95





1. GİRİŞ

1.1. Bilgisayar Bütünleşik Tasarım Ve İmalat Sistemlerinden CNC Tezgahlarına Veri Aktarımı

1940'lı yıllarda başlayan bilgisayar teknolojisindeki gelişmelerin imalat sektörüne yansması sonucu, 1950'li yıllarda ilk sayısal denetimli tezgah yapıldı ve böylece Sayısal Kontrol (Numerical Control) kavramı doğmuş oldu. Sayısal Kontrol programlanmış otomasyon yöntemlerinden biridir. Sayısal Kontrolde, takım tezgahları sayısal olarak tanımlanan kodlarla çalıştırılır ve kontrol edilir.

Sayısal Kontrol ile imalat sisteminde, işlenecek parça ile ilgili birtakım bilgiler, kesici takımın kontrollü hareketini sağlayan bir dizi sayısal kod haline çevrilmelidir. Parça programı olarak adlandırılan bu sayısal komut dizisi temelde üç şekilde üretilmektedir;

1. Elle programlama vasıtasıyla;
2. Bilgisayar destekli programlama (APT ve benzeri ileri seviye diller) vasıtasıyla;
3. CAD / CAM sistemleri vasıtasıyla.

Elle programlama yönteminde operatör, belirli bir tezgah üzerinde, tezgaha bağlı dili (komutlar dizisi) kullanarak parça programını yapar. Bununla birlikte günümüzde bu yöntem yerini, verimliliğin çok daha fazla olduğu diğer yöntemlere bırakmıştır.

Elle programlama yöntemi haricinde diğer yöntemler kullanılarak oluşturulan parça programları, ilk etapta tezgah kontrolörü tarafından anlaşılabilir bir formda

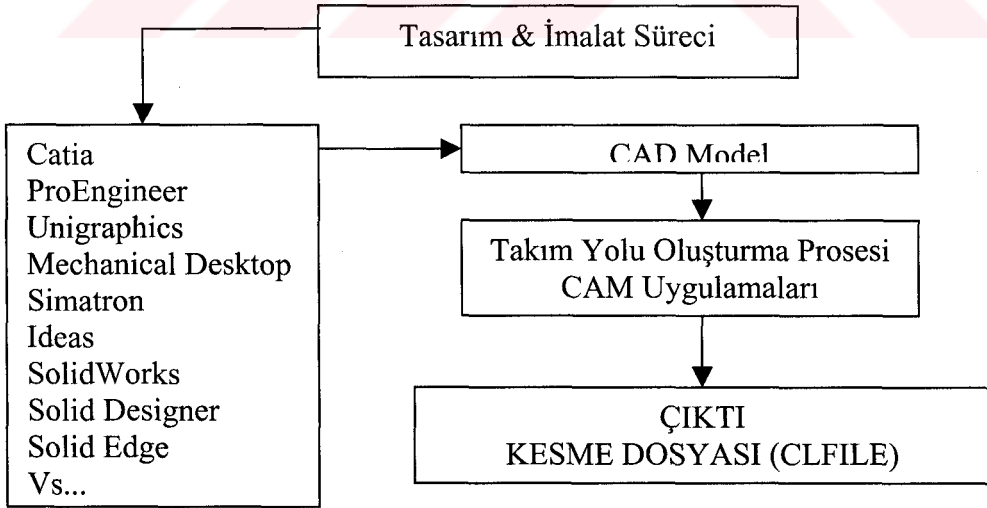
değildir ve parça programının bu özel formuna kesme veri dosyası (Cutter Location Data File) adı verilir.

Kesme dosyasının taşıdığı veriler geneldir, bir başka deyişle kesme dosyaları aynı türde tüm tezgahlara hitap eder. Ancak daha sonra, Postprosesör adı verilen bir program vasıtasıyla son bir işlemden geçirilerek, belirli tezgah diline uygun hale çevrilir. Dolayısıyla bilgisayar destekli tasarım ve imalat sürecinin son kademesindeki bu programlar süreçte önemli bir yer teşkil eder.

Postprosesörler, kesme dosyasındaki verileri, belirli bir tezgah veya kontrolör tarafından tanınan, kesin bir formdaki çeşitli kodlara çevirirler.

1.2. Postprosesör Yazılımları

Şekil 1.1'de gösterildiği üzere, günümüzde çok değişik CAD/CAM sistemleri, tasarım ve imalat sürecinde farklı yöntemler kullansalar bile bir kesme dosyası oluşturmaktadırlar.



Şekil 1.1. Tasarım Ve İmalat Süreci Elemanları

Çok çeşitli CAM yazılımları farklı yapıda kesme dosyaları oluştururlar. Esasen kesme dosyaları, belirli standartlarda komutlar kullanılarak oluşturulur ancak komut satırlarının fiziksel yapıları farklıdır. CAM yazılımları, bu standart komutlardan sadece bir kısmını kullanırken ayrıca standart dışı, kendine has komutlar da kullanabilir.

Tezgah üreticisi firmalar ise, ürettikleri tezgahların konfigürasyonuna bağlı olarak değişik kontrolörler kullanırlar. Tezgahlar temelde bazı standart komutlarla kontrol edilir ancak bu durum yine üretici firmaya göre değişmektedir. Kısacası tezgahın kontrolünü sağlayan dil, üretici firmalara bağlı olarak değişmektedir.

Yukarıda bahsedilen konulardan dolayı CAM sistemleri, sonuçta postprosesör yazılımı kullanmak zorundadır. Tipik bir postprosesör programın şu görevleri vardır; [Mahieddine ve Webb, 1989].

1. Kesme dosyası içinde verileri okur, sıralar ve kontrol eder;
2. Genel kesme dosyası içindeki koordinatları, makina koordinat sistemine taşıyacak transfer işlemlerini yapar, takım için uygun ilerleme hızını hesaplar, doğrusal ve dönel hareketleri gözlemleyerek tezgahın mekanik limit değerlerini kontrol eder;
3. Hazırlık ve yardımcı fonksiyon komutlarını genişletir;
4. Kesme dosyalarındaki harici bilgi akışlarını izler ve zaman hesabı yapar;
5. Makina kontrol veri dosyası ve liste dosyası oluşturur.

Postprosesörlerin kompleks yapısı, tezgahın kapasitesi ve karmaşıklığına bağlı olarak değişir. Her bir farklı CAM yazılımı ve tezgah kontrolörü kombinasyonu özel bir postprosesör programı gerektirir. Bu durum postprosesör yazılımlarını, özellikle 4 ve 5 eksen işleme merkezleri için pahalı yapabilir [Edwards ve Sharp, 1991].

Günümüzde Postprosesör Generator olarak adlandırılan yazılımlar yardımıyla çok değişik tipteki tezgahlar ve CAM sistemleri için özel postprosesör oluşturulabilir. Bu tür yazılımların, pahalı olması, yeterince esnek ve özellikle 5 eksen işleme merkezleri için kolay kullanımının olmaması gibi birtakım dezavantajları vardır.

Bu tez çalışması, belirli bir CAM sistemi ve 5 eksen universal işleme merkezi için özel bir postprosesörün geliştirilmesini kapsamaktadır. Uygulamada programın geliştirilmesi için C++ Builder3.0 kullanılacaktır. Uygulama sonucunda elde edilecek veriler tezgah üzerinde test edilecektir.

2. SAYISAL KONTROLLÜ TEZGAHLAR



Üretimin artırılması ve üretim kalitesinin insan faktörüne olan bağımlılığının azaltılması, üretim tezgahlarında gelişmeyi yönlendiren en önemli etkenlerdendir. Bu yöndeki gelişmelerin başlangıcı 1820'lere kadar uzanmaktadır [Karagöz, 2000].

Bilgisayarlar 1940'lı yıllarda hesaplayıcı makineler olarak geliştirilmeye başlandı. Başlangıçta çok yavaş ve büyük olmaları nedeniyle önemli uygulamalar yapılamadı. 1950'lerin ikinci yarısında, lambalı devreler kullanılarak, bilgisayarlar yapıldı. 1950'li yılların ağır ekonomik bunalımı, gittikçe artan maliyetler, işletme zorlukları ve rekabet endüstriyel kuruluşları klasik tasarım, imalat ve işletme yöntemleri dışına yöneltti. Amaç, birim maliyeti düşürerek ve üretim hızını artırarak daha üretken yöntemler aramaktır. O yıllarda elektronik sanayiinde ve bilgisayardaki gelişmeler bu yeni arayışlara yön verdi. Takım tezgahlarının bir kontrol düzeni altında hatasız, çabuk ve kendi başına parça işlemesi endüstri devriminin önemli hedeflerindendir. Bu konuda ki öncü çalışma Parson tarafından 1940'lı yılların sonlarında gerçekleştirilmiştir [Karagöz, 2000].

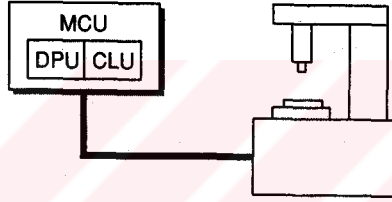
1940'lı yıllarda John Parsons, hava taşıtları kanat formları gibi çapraşık ve ince ayrıntılı formları işlemek için bir yöntem geliştirdi. Bu yöntem, bir kart üzerine delinen çeşitli deliklerin, makine kesici takımını kontrol etmesine dayanıyordu. 1947 yılında Amerikan Hava Kuvvetlerinin projeleri dahilinde istediği formların, Amerikan metal kesme endüstrisi tarafından imal edilememesi, Parsons'un makinada kullandığı fikri cazip kıldı. Böylelikle, Amerikan Hava Kuvvetleri bir kontrol sistemi geliştirmek üzere Parsons Corporation ile anlaşma yaptı. 1951 de Parsons Corporation, Massachusetts Institute Of Technology (MIT) Servomekanik laboratuvarı ile bir alt anlaşma yaparak, 1952 de 3 eksen Cincinnati Hydrotel freze tezgahını yaptı ve böylece Sayısal Kontrol terimi ortaya çıkmış oldu [McMahon, 1993]. Sayısal Kontrolün tanımı, EIA (Electronic Institutes Association) tarafından aşağıdaki şekilde yapılmıştır;

“NC sistemi, bazı noktaların sayısal verilerinin direkt sisteme girilmesiyle, tüm sistem aktivitelerin kontrol edilmesidir. Sistem mutlaka en azından bu verilerin bir kısmını otomatik olarak çevirmelidir.”

Sayısal veriler, bir parçanın imali için gereklidir ve “parça programı” adı verilen program formundan elde edilir. Parça programı, makine kontrol sistemi tarafından okunabilen ve kontrolörün bunu, tezgah mili ile makina kesicisini hareket ettirdiği çeşitli sinyallere çevirdiği bir dizi satırdan oluşmaktadır. Parça programı, parçanın geometrik bilgileriyle, iş parçasına (hammadde) göre kesici takımın hareket ettirildiği hareket bilgilerini içerir.

2.1. Bir NC Tezgahın Konfigürasyonu

Şekil 2.1 de görüldüğü üzere, bir NC tezgah, Makine Kontrol Ünitesi (MCU) ve tezgahın kendisinden oluşur.



Şekil 2.1. NC Tezgahın Konfigürasyonu

MCU tezgahın beyini olarak tanımlanabilir. Parça programlarını okur ve tezgahı kontrol eder. Bu iki fonksiyon Makine Kontrol Ünitesi tarafından iki bölüme ayrılır; Veri İşleme Ünitesi (DPU-Data Processing Unit) ve Döngü Kontrol Ünitesi (CLU - Control Loop Unit). Veri İşleme Ünitesi, kağıt veya teyp bandından okuduğu parça programı cümlelerini çözümler, çözümlediği bilgileri işler. Tezgahın tüm eksenlerinin alacağı pozisyonun, hareket ve hızının taşındığı bu bilgiler Makina kontrol ünitesine gönderilir. Döngü kontrol ünitesi, veri işleme ünitesinden aldığı bilgileri kontrol sinyallerine çevirir, makine hareket mekanizmasını çalıştırır, her bir eksenindeki gerçek hız ve pozisyonla ilgili sinyallerin geri beslemesini yapar ve operasyonu bitirdiğinde Veri İşleme Ünitesine talimat vererek, parça programından yeni direktif okumasını sağlar.

Bir veri işleme ünitesi, kağıt-teyp okuyucu, veri okuma devresi ve istenen eksenin hareketlerini tayin eden devre çözücü elemanları içerir. İlk ve ikinci kuşak NC veri



işleme ünitelerinde, parça programlarının taşındığı delikli şeritlerin okunabildiği bir sürücü mevcuttur.

Döngü kontrol ünitesi, bir enterpolator, tüm eksen hareketleri için pozisyon kontrol döngüsü, hız kontrol döngüsü, yavaşlama ve tepki emici devreler ve yardımcı fonksiyonlardaki kontrol cihazlarından oluşur.

Enterpolator, takım hareketi için belirli veri noktalarını temin eder. Pozisyon kontrol devresi tüm hareketli ekseninde bulunur ve eksenindeki konumlanmayı kontrol eder. Hız kontrol devreleri ise ilerleme kontrolünde kullanılır. Yardımcı fonksiyonlara sahip cihazlar ise soğutma sıvısı açılıp kapanması, dişli değiştirilmesi ve tezgah mili dönme hareketi kontrolü gibi çeşitli fonksiyonları sağlarlar.

2.2. NC Sistem Tipleri

NC tezgahlarda, noktadan noktaya veya sürekli hareket (kontur) kontrolörü olmak üzere başlıca iki tip NC kontrolör vardır. Bir noktadan noktaya NC kontrolör, takım yolunun iş parçasına göre önemli olmadığı durumda kullanılmaktadır. Bu genelde, takımın bir noktadan diğerine hareketinde iş parçasına temas etmediği durumlarda söz konusudur. Bu durumlara verilecek en tipik örnekler, delme, punch pres, diş çekme ve bir baskı devre üzerine eleman yerleştirme işlemi olabilir. Noktadan noktaya esaslı hareket kontrolörü basit bir yapıya sahiptir ve bu yüzden bu tür kontrolörlü bir NC tezgah pahalı değildir. Ayrıca, bir noktadan diğerine gidiş hızını kontrol etmemizi sağlayan bir hız kontrol mekanizması sisteme eklenmesiyle kanal açma gibi basit frezeleme işlemleri yapılabilir.

Takım hareketi işlenecek parça üzerinde bir konturu takip edecekse, sürekli tip kontrolör sistemi gereklidir. Sürekli kontrol sistemlerine sahip tezgahlara örnekler, freze, torna ve taşlama tezgahları, plazma kesme gibi alevli kesiciler, kaynak tezgahları şeklinde verilebilir.

Bu tezgahlarda, iki yada daha fazla eksenin eşzamanlı kontrol edilebilmesi nedeniyle her bir ekseninde farklı hızlar sağlanabilir. Örneğin, XY düzleminde bir eğriyi

takip eden takım, bu hareketini hızın iki elemanını V_x ve V_y ’ yi kontrol ederek yapar. Bu yeteneği sayesinde NC kontrolör, örneğin, daire merkezi yeri, yarıçap, ve yayın son noktaları belirlendiğinde dairesel bir yörünge çizebilir.

2.3. NC/CNC/DNC

İlk kuşak vakum tüp teknolojisine sahip NC tezgahlar ile ikinci kuşak katı hal devre teknolojisi tabanlı NC tezgahlarda, parça programı, makina kontrol ünitesine delikli şerit vasıtasıyla aktarıldı. NC Tezgahlar olarak tanımlanan bu tür tezgahlarda, makina kontrol ünitesi içinde, parça programının saklanamaması ve aynı anda sadece bir komutun işlenebilmesi gibi sakıncalar söz konusuydu.

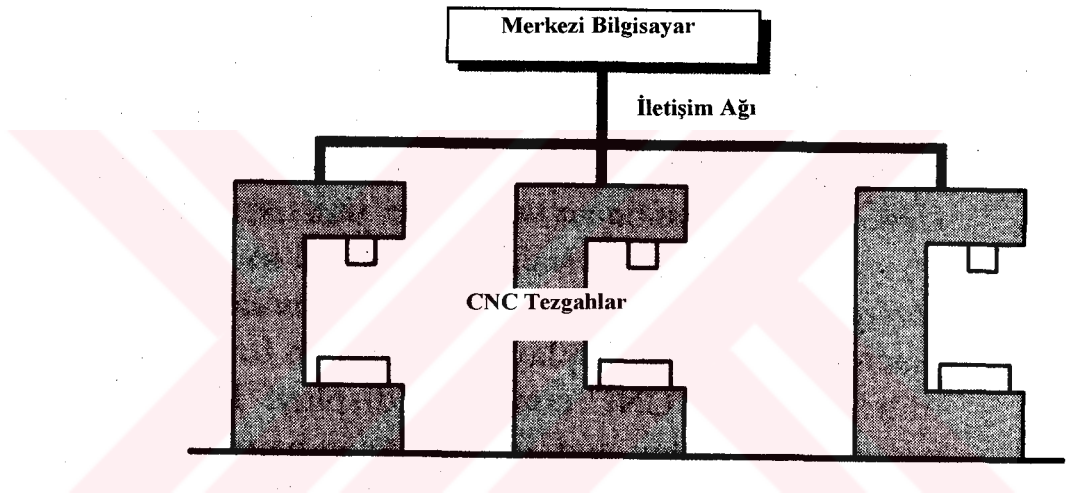
Üçüncü kuşak tezgahlar ise, bilgisayar donanımlarında genişçe kullanılan bellek ve entegre devre teknolojisine sahiptir. 1970’lerde, bu şekilde bilgisayarın kontrolcüyle birleşmesi günümüz modern kontrolörün önderliğini yapmıştır. Bu tür tezgahlar, Bilgisayar Kontrollü Tezgahlar (CNC - Computer Numerical Control) olarak adlandırılmışlardır. Bir belleğe sahip bu tür tezgahlarda parça programı bir kez makina kontrol ünitesine yüklendikten sonra, kolayca istenildiği kadar kullanılabilir. Buna ek olarak, bu tür tezgahlar farklı diğer ünitelerle (Robot ve CNC tezgahlar, Bilgisayarlar) iletişim kurabilir, veri alışverişi yapabilirler. Daha sonraları, CNC tezgahlarda operatöre görsel parça programlaması ve hata analizi imkanını veren monitörler eklenmiştir. Daha karmaşık tezgahlar ayrıca takım yolunun grafiksel doğrulamasını yapabilir.

Günümüzde modern CNC tezgahı bir kişisel bilgisayarla birleştirilmiş ve gerçekte, kontrol edilebilen mekaniği ile özel amaçlı bir bilgisayardan farksızdır. Bir Merkezi İşlem Birimine (CPU - Central Processor Unit) sahip olmalarının yanı sıra, ROM (Read Only Memory), rasgele erişimli bellek (Ram - Random Access Memory), sabit disk, iletişim portları ve monitöre de sahiptirler.

ROM’un avantajı “kapalı döngü” adı verilen programlanabilir makine kod döngülerine imkan sağlamasıdır. Döngüler standart alt-rutin olarak tanımlanır ve tezgah kütüphanesinde saklanır. Herhangi bir NC program içerisinde, özel bir kodla bu döngüler çalıştırılabilir. “Kapalı” kelimesi alt-rutin önceden yazıldığını ve

saklandığını, “döngü” kelimesi ise bu alt-rutinin tekrar ve tekrar kullanılabilirliğini ifade eder. Delik delme, kılavuz, raybalama, bara, dış açma bu tür döngülere örnektir.

Doğrudan Sayısal Kontrol (DNC / Direct Numerical Control) imalat sistemi, Şekil 2.2.’de gösterildiği gibi, merkezi bilgisayardan eşzamanlı olarak yönlendirilebilen bir çok tezgahtan oluşmuştur. Doğrudan Sayısal Kontrolde, ana bilgisayar ya kendi sabit diskinden yada dışarıdaki herhangi bir üniteden parça programı verilerini okur ve sonra, tasarlanan parçayı imal etmek üzere, parça programı direktiflerini bloklar halinde NC tezgaha gönderir. Bu teknik, merkezi bilgisayarın tezgahlara hizmet etmesi üzerine kurulmuştur.



Şekil 2.2. DNC Yapısı

Doğrudan Sayısal Kontrol sistemi artık yerini Dağıtık Sayısal Kontrol sistemine bırakmıştır. Dağıtık Sayısal Kontrolde, merkezi bilgisayar tüm parça programını CNC tezgahlarına aktarır. Bu tezgahlar, lokal belleklerine bir yada daha fazla parça programını tuttukları için, merkezi bilgisayardan bağımsız çalışırlar.

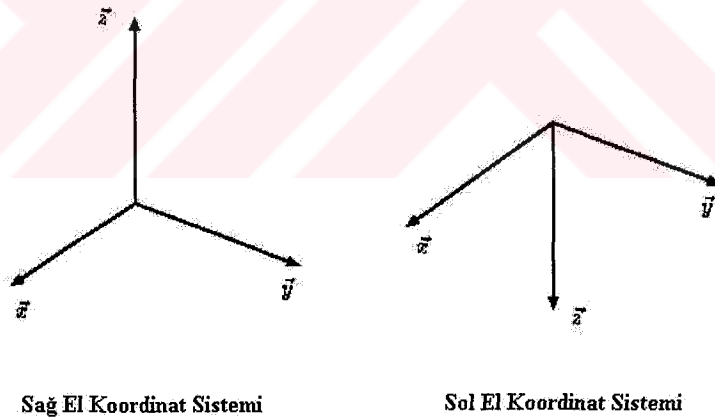
2.4. Parça Programlamanın Temel Kavramları

Parça programı, işlenecek model hakkında geometrik bilgiler ve iş parçasına bağımlı olarak kesici takımın hareket yapmasını sağlayan hareket bilgilerinden oluşmaktadır. Bu nedenle, parça programlayıcısı bir şekilde bu bilgileri program içinde

tanımlamak zorundadır. Bu durumda yapılacak ilk şey, nokta konumlarının tayinini sağlayan bir koordinat sistemi belirlemektir. Geometrik ve kesici hareket bilgileri bu koordinat sistemine göre belirlenecektir. Eğer parça programlayıcısının belirlediği koordinat sistemi ile tezgahın varsayılan koordinat sistemi arasında bir uyumsuzluk söz konusuysa, bu durumda hatalar meydana gelecektir. Bu yüzden NC programlayıcısı, takım yolunun geometrik koordinat sistemi ile tezgah koordinat sisteminin özdeşliğini, uygun bir NC program için temin edebilmelidir. Eğer takım yolu bir CAD (Computer Aided Design) veritabanından oluşturulduysa, modelleme için oluşturulan koordinat sistemi ile tezgahın varsayılan koordinat sistemi aynı olmalıdır.

2.4.1. Koordinat Sistemleri

Takımın iş parçasına göre hareketi, tezgah kesici takımının kayma hareketleriyle elde edilir. Üç ana eksendeki hareketler, X, Y ve Z, genelde sağ el kuralı referans alınarak tayin edilir (Şekil 2.3).



Şekil 2.3. Sağ Ve Sol El Koordinat Sistemleri

X eksenini; torna gibi, iş parçasının döndüğü tezgahlarda, takımın hareket ettiği doğrultudur ve pozitif yönü takımın iş parçasından uzaklaştığı yöndür. Dik freze ve matkap tezgahlarında, pozitif x yönü, programcının yüzü tezgaha dönük pozisyonda iken, sağ tarafı işaret eder. Yatay freze tezgahlarında ise x eksenini tezgah tablasına paraleldir.

Y eksenini; y ve z eksenlerine bağı olarak tayin edilir.

Z eksenini; iş parçasının döndüğü torna tezgahı gibi tezgahlarda, ayna eksenine paralel ve pozitif yönü, kesicinin iş parçasından uzaklaştığı doğrultudadır. Frezeleme, delme, raybalama işlemlerindeki gibi, takımın döndüğü tezgahlarda, z eksenindeki pozitif yönlü hareket, takımın iş parçasından uzaklaştığı harekettir.

Bazı tezgahlarda, birincil eksen olarak kabul ettiğimiz X, Y, Z eksen hareketlerine ek olarak, U, V ve W ile ifade edilen ikincil eksen hareketleri de söz konusudur. Ayrıca X, Y, Ve Z eksenlerine paralel eksenler etrafında, A, B, C olarak belirtilen açısıl dönme hareketleri de vardır. Bütün bu ekstra eksen simgeleri EIA standartlarına göre tanımlanmıştır.

Geleneksel olarak tüm bu tezgahlar, kesici takımın konumlanmasını sağlayan eksen sayısına göre sınıflandırılır. Örneğin terminolojide, 2-eksen, 3-eksen, 5-eksen, tezgah kontrolörünün eşzamanlı olarak kaç eksenini kontrol ettiğini göstermektedir. Başka bir deyişle, eğer tezgah kontrolörü sadece iki ekseninde simültane hareket yapabiliyorsa 2-eksen tezgah olarak tanımlanır. Bu durumda takım, bağımsız olan üçüncü bir ekseninde kontrol edilir. 3-eksen bir tezgahta, simültane 3 eksen hareketi vasıtasıyla takım herhangi bir eğri boyunca hareket edebilir. Takımın açısıl konumlanmasını sağlayacak hareket 3-eksenden fazlasını gerektirmektedir. Günümüzde 9 eksene kadar açısıl konumlama sağlayan tezgahlar mevcuttur.

2.4.2. Parça Programlama

Parça programı, bir parçanın tam olarak işlenmesi için tüm bilgileri içerir. Bir program, blok adı verilen ve tek bir sıraya yazılan bir takım cümlelerden meydana gelmektedir. Her blok farklı tipte bilgi içerebilir. Genelde bir programda üç çeşit bilgi vardır;

- Takım veya parçanın konumunu, yaptığı hareketin şeklini ve yönünü kapsayan, **geometrik bilgiler**;

- Takım, ilerleme hızı ve kesme hızını içeren, **teknolojik bilgiler**;
- Kesme hareketinin gerçekleşmesini (tornada parça, frezede takımın dönmesi), takım değiştirmesini, kesme sıvısının kullanılmasını içeren, **yardımcı bilgiler**.

2.4.3. Parça Program Yapısı

NC tezgah ve sistemlerde programlama, ASCII (American Standart Code For Information Interchange) kod esasına göre oluşturulan ISO, EIA ve DIN gibi aynı içerikli kod sistemine dayanmaktadır. Bununla beraber NC sistemleri için kontrol üniteleri üreten firmalar arasında, yani kontrol sistemleri arasında bazı farklılıklar vardır. Ancak tüm bu sistemde programlamanın esası aynıdır.

Temel olarak, tezgahlar ilerleme hızı, soğutma sıvısı durumu, fener mili dönme hızı gibi parametrelerini parça programındaki direktiflerden, bu direktifleri ise blok adı verilen ve parça programındaki sıralı komut satırlardan alırlar. Her bir blok satır içindeki birçok komuttan meydana gelebilir. Satırdaki her bir komut belirleyici bir harf ile tanımlanır. Bu belirleyici harfler şu şekildedir;

Program Adı, programın adını temsil eder ve rakamlardan oluşur.

Blok Numarası, N harfi ve bunu takip eden bir sayı ile ifade edilir ve her bir bloğun tanımlanmasında kullanılır. Genelde sayı, 1'den 9999'a kadar dört rakamdan oluşur. Sayılar düzenli ve ard arda olabildiği gibi düzensizde olabilir. Kontrol ünitesinde işlem görmedikleri için blok numaralarının konulması mecbur değildir. Ancak koyulması programın kullanılması açısından kolaylıklar sağlar. Blok içindeki komutların hızlıca işlenmesini sağlarlar.

Hazırlayıcı fonksiyonlar (G kodları) makina kontrol ünitesini operasyona hazırlar. Operasyon genellikle kesici takım hareketlerinden oluşmaktadır. Bu kodların EIA RS-273 standardına göre tanımlanmaları, Ek 1'de detaylı bir şekilde verilmiştir. Bununla birlikte, aynı G kodu farklı kontrolörler için farklı anlamlar taşıyabilir. G kodu ile ifade edilen hazırlık fonksiyonları, blok içinde komutların anlamını tayin eder. Buna

göre hazırlık fonksiyonu: hareket tipini, boyutların mutlak veya eklemeli, milimetre veya inç olarak verilmesi, çalışma şeklini, ve benzeri özel işlemleri belirtir. Fonksiyonun sayısal kısmı 0'dan 99'a kadar 2 basamaktan oluşur. Her sayı kod şeklindedir; yani sabit bir anlam taşır. Bir takım hazırlık fonksiyonları, hemen hemen tüm kontrol sistemlerinde aynı anlamda; bazıları ve boş olanları farklı anlamda kullanılmaktadır.

Koordinat kodları yani, X, Y, Z, A, B ve C kodları vasıtasıyla kesici takım konumu belirlenmektedir. X, Y, Z kodları 3 eksenle konumlamayı sağlarken, 3 eksenle fazla eksene sahip tezgahlarda A, B ve C gibi ekstra kodlar kullanılır. Bu kodların koordinat değerleri gerçel sayı olarak verilir ve belirli duyarlılıkları vardır. Sayıda virgülden sonra ifade edilen bu duyarlılık, NC Tezgah hassasiyetine bağlıdır.

İlerleme hız (F) kodu; kesicinin ilerleme hızını belirleyen koddur. İlerleme hızı, duruma göre metrik veya inç birim sistemlerinde verilebilir.

Dönme devir kodu (S); tezgah milinin dönme hızını belirler. Genelde birimi, dakikadaki dönme devri (Revolution Per Minute-RPM) olarak verilir.

Takım seçme (T) kodu; Belirli bir operasyon için istenen kesici takımın kullanılması bu kod vasıtasıyla sağlanmaktadır.

Yardımcı fonksiyon (M Kodları) kodları; CNC tezgah ve sistemlerinin çalışmasını kontrol eder. Bu kodların etkisi daha çok aç/kapa (on/off) şeklinde olur. Örneğin iş milinin dönmesi veya durması, kesme sıvısının açılması veya kapanması, programın durması vb. gibi. M kodunun sayısal kısmı 0'dan 99'a kadar iki rakamdan oluşur. Bu kodlar, EIA RS-273 standardına göre Ek 2'de detaylı şekilde verilmiştir. Karşılaşılabilecek diğer kodlar tezgah üreticisinin belirlediği, tezgaha has kodlardır.

2.5. Parça Programlama Yöntemleri

2.5.1 Elle Parça Programlama

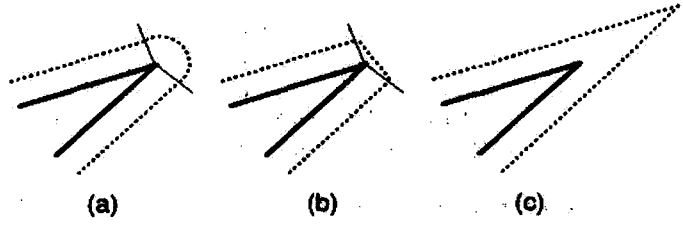
Elle programlama yöntemi, parça programcısının bilgisayar yardımı almaksızın, doğrudan işlenecek parça programını, G kodlarıyla oluşturmasıdır. Burada program herhangi bir belge gibi elle yazılabilir. Ancak programlamayı kolaylaştırması ve hataları ortadan kaldırması maksadıyla müsveddeler hazırlanabilir. Programlama müsveddelerdeki her bir satır bir bloğu ifade eder ve blok sonunda, satır sonu karakteri (EOB) bulunur. Çeşitli formlarda hazırlanabilen müsveddelere bir örnek Ek 3'te verilmiştir.

Programcı elle programlama yaparken tüm dikkatini toplamak zorundadır çünkü, bu tür bir yöntemde hata analizi yapmak zordur. Programcının müsveddelere yazdığı kodlar direkt kontrolöre aktarılacaktır.

Elle programlamada temel sorun, programlayıcının parçanın konfigürasyonunu (parça şekli ölçüleri vs.) değil de bu konfigürasyona göre takım hareketlerini belirlemek zorunda olmasından kaynaklanır. Bu beraberinde, hata nedeni olan ve genelde can sıkıcı birçok aritmetik-trigonometrik hesabın yapılmasını getirir. Elle programlamadaki bu sıkıntıları bir nebze kaldıran günümüzün CNC teknolojisinin özelliklerini şu şekilde toparlanacak olursa;

- Birçok CNC sistemi takım yarıçapı telafisi sağlamaktadır. Bu parça geometrisi konturlarını ötelemeden, doğrudan parça geometrisinden program yapılmasını sağlar. Başka bir deyişle, programcı parça geometrisi açısından programı hazırlar, kontrolör ise bu yolu takım yarıçapına göre öteler ve takımı ötelenen bu eğri üzerinde hareket ettirir. Bu özellik, parça geometrisinden ötelenen kenarların kesişmediği durumlarda, ötelenen eğrileri uzatan veya kırpan başka bir özellikte çalışır.

Şekil 2.4.(a), bu durumda ekstra bir yayın eklenmesini, Şekil 2.4(b), bir doğru parçasının eklenmesini, Şekil 2.4(c), ise eğrilerin uzatılmasını örneklemiştir. Ancak bu özellikler sadece basit geometriler için geçerli olmaktadır.



Şekil 2.4. CNC Sistemlerinde Takım Yarıçap Telafisi

- Çeşitli tekrarlanan hareketlerden oluşan ortak işleme operasyonları, döngüler gibi kaydedilebilir ve bu sayede parça programlarında ihtiyaç duyulduğunda çağrılabilir. Geleneksel programlama dillerinde bu döngüler standart alt-rutin kütüphanesine denktir.
- Kapalı Döngülere ek olarak, programcının tanımladığı çeşitli komutlarla oluşturulan makrolar, değişken parametreleri değiştirilerek, parça programında tekrarlanılarak çağrılabilir.

Bütün bu özelliklerine rağmen, son derece kolay parçalar haricinde, bilgisayar desteğiyle parça programı yapmak, elle programlama yapmaya göre elbette programcının iş yükünü azaltacaktır.

2.5.2 Bilgisayar Yardımlı Parça Programlama

Elle programlamaya bir alternatif, tezgaha göre değişen alışılmadık kodlarla parça geometrisini ve takım yolunu tanımlamak yerine, yüksek seviyeli programlama dilleri kullanmaktır. Bu dillerin esasını İngilizce kelimeler ve kullanımı kolay matematiksel tanımlar oluşturur. Buna ek olarak, bu tür diller genel amaçlı tüm bilgisayarlarda çalıştırılabilir. Bu dilleri kullanan programlayıcının temelde iki görevi vardır. Birincisi,

çeşitli noktalar, çizgiler, yaylar vb. geometrik elemanlar vasıtasıyla parçanın geometrik açıdan tanımlanmasıdır. İkincisi ise, kesici takıma, bu geometrik elemanlar boyunca takip edeceği yolu göstermektir. Parça geometrisinin ötelenmesi sistem tarafından otomatik olarak hesapladığından elle programlamaya nazaran daha güçlü ve kolay bir sistemdir.

Herhangi bir NC kontrolörü bu tür bir yüksek seviyeli dil ile programlarken şu prosedürlere uyulur;

- Programlayıcı parça geometrisini, genel kesici takım hareketlerini, kesici parametrelerini, ilerleme ve fener mili dönme hızlarını tanımlar.
- Programlayıcı herhangi bir programlama dilini kullanarak bu tanımlar vasıtasıyla kodlar oluşturur. Bu kodlara “kaynak” adı verilir. Bu amaçla kullanılan en popüler dil APT (Automatically Programmed Tools) dilidir.

Kaynak, makine-bağımsız kesici takım ve yardımcı kontrol bilgilerinden oluşan ve CLDATA (Cutter Location Data File-Kesme Dosyası) olarak bilinen bir dosya oluşturulmak üzere derlenir. Kesme dosyaları (CL data file) ikilik sisteme göre oluşturulan dosyadır ancak, ayrıca okunabilir haldeki bir başka versiyonu da vardır. Kesme dosyaları, detayda takım hareketlerini gösterir bir dizi mutlak lineer GOTO veya göreceli GOLDDTA hareketlerinin yanı sıra, fener mili, soğutma sıvısı, ilerleme hız kontrolü vb. postprosesör komutlarından meydana gelir. Kesme dosyası formu, ISO (International Organization For Standardization) tarafından belirlenmiştir.

Kesme dosyası, belirli bir tezgah için, makine kontrol verilerine dönüştürülmek maksadıyla postprosesör vasıtasıyla işleminden geçirilir. Burada makina kontrol verileri, elle programlama yönteminde elde edilen verilerle aynıdır. Makine kontrol verileri formunun yanı sıra belirli bir G ve M kodu direktifinin olmamasından dolayı, postprosesörler, belirli tezgah-kontrolör kombinasyonuna has, özel programlardır. Parça geometrisinin ve takım yolunun bir programlama diliyle kodlanması yerini, CAD veritabanından doğrudan kesme dosyası oluşturulması yöntemine bırakmasına rağmen,

bilgisayar destekli parça programlama tekniđi, günümüzde geçerli bir yöntem olarak hala kullanılmaktadır.

2.5.3. APT Dili

NC programlama için bir çok programlama dili geliştirilmiş olmasına rağmen, APT (Automatically Programmed Tool) dili, en kapsamlı ve geniş kullanıma sahip dildir.

APT sisteminin ilk prototipi MIT tarafından 1956 yılında geliştirildi. Programın geliştirilmesinde MIT' in asistanlığının yanında, 21 şirket AIA'nın (Aerospace Industries Association) sponsorluğunda birleşti. Bu çalışmaların sonunda APT II 1958 yılında ve bundan çok daha etkili olan bir sistem APT III 1961 yılında geliştirildi. Bu tarihlerden sonra, IITRI (The Illinois Institute Of Technology Research Institute), programı doğrudan gelecek içinde genişletmek ve geliştirmek için seçildi ve program kapasitesi sürekli bir şekilde artırıldı [Lee, 1999]. Günümüzde APT dili, 5 eksen kadar tezgahları kontrol edebilecek kapasitededir. Bir APT program 5 kısımdan oluşur;

1. Tanımlama komutları; parça (model) ismi, kullanılacak tezgah ve postprosesör tipini belirleyen komutlar;
2. Geometrik Komutlar; işleme teknikleriyle ilgili, parça geometrisinin belirlendiđi komutlar;
3. Takım Hareket Komutları; parça geometrisine bağımlı kesici takım hareketlerini belirleyen komutlar;
4. Postprosesör Komutları; ilerleme ve tezgah mili dönme hızı ve soğutma sıvısı kontrol komutları gibi tezgah parametreleriyle ilgili, hiçbir deđişiklik yapılmadan kesme veri (CLDATA) dosyasına aktarılan komutlar;

5. Yardımcı Komutlar; kesici takım tanımlaması, tolerans değeri gibi yardımcı tezgah fonksiyonlarının belirtildiği komutlar.

2.5.3.1. Geometrik Komut Satırları

Geometrik komut satırlarının genel yapısı aşağıdaki gibidir.

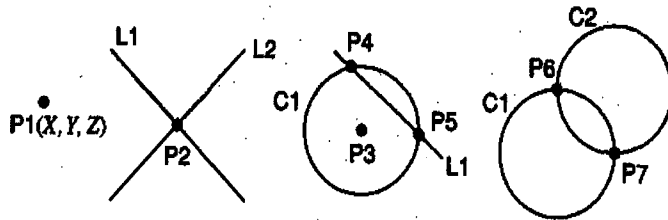
Sembol = geometri_kelimesi / tanımlayıcı veri

Sembol, geometri elemanının ismidir ve öteki yüksek-seviye programlama dillerindeki gibi bir değişken ismi rolü oynar. Sembol, bir harf ile başlayan ve en fazla altı karaktere kadar bir kattan oluşturulur.

Geometri_kelimesi, noktalar, çizgiler, çemberler, konikler, küreler, ruled yüzeyler, tabulated silindirler gibi geometri tiplerini tanımlarlar.

Tanımlayıcı veri, nesne tanımlamak için gerekli olan sayısal veriyi içerir. Bu arada nesne tanımında kullanılan diğer nesneler veya INTOF gibi tali (minor) kelime olarak adlandırılan tanımlamalar dikkate alınır. Tali kelimeler, kullanılmakta olan geometri tanımı tipini işaret etmekte kullanılır.

Genel geometri_kelimeleri tanımlama yöntemleri sadece nokta, çizgi, çember nesneleri şu şekilde yapılır;



Şekil 2.5. APT Dilinde Nokta Tanımı

Şekil 2.5'te gösterildiği üzere bir noktanın tanımlanmasında bir çok yöntem kullanılabilir. Şekildeki nokta tanımlamaları şu formdadır;

$P1 = \text{POINT}/x,y,z$; koordinatların belirtilmesiyle ;

$P2 = \text{POINT}/L1,L2$; iki doğrunun kesişim noktasının ifadesi;

$P3 = \text{POINT}/\text{CENTER}, C1$; çemberin merkezini belirleyen nokta;

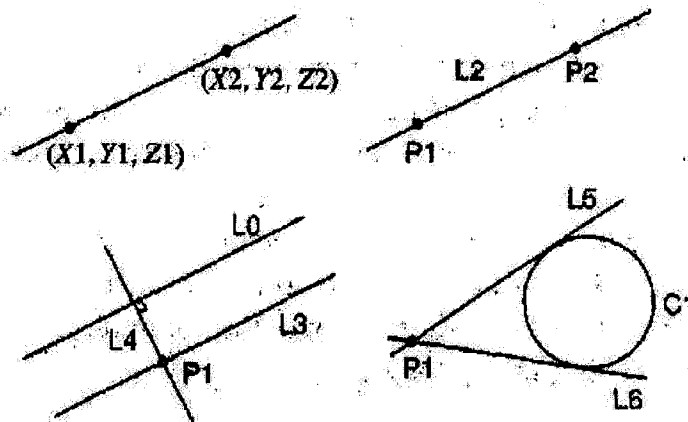
$P4 = \text{POINT}/\text{YLARGE}, \text{INTOF}, L1, C1$; bir doğru ve çemberin kesişim ifadesi; koordinatta y değeri daha büyük;

$P5 = \text{POINT}/\text{XLARGE}, \text{INTOF}, L1, C1$; bir doğru ve çemberin kesişim ifadesi; koordinatta x değeri daha büyük;

$P6 = \text{POINT}/\text{YLARGE}, \text{INTOF}, C1, C2$; iki çemberin kesişim ifadesi, koordinatta y değeri daha büyük;

$P7 = \text{POINT}/\text{XLARGE}, \text{INTOF}, C1, C2$; iki çemberin kesişim ifadesi, koordinatta x değeri daha büyük;

Şekil 2.6 ve Şekil 2.7'de gösterildiği gibi bir çizginin APT dilinde belirlenmesinin birçok yöntemi vardır;



Şekil 2.6. APT Dilinde Çizgi Tanımlaması

$L1 = \text{LINE}/X1, Y1, Z1, X2, Y2, Z2$; noktalar arasındaki $(X1, Y1, Z1 - X2, Y2, Z2)$ doğru parçası ifadesi;

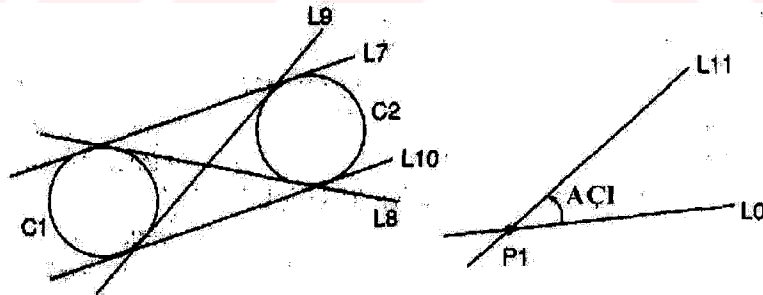
$L2 = \text{LINE}/P1, P2$; daha önce belirlenen $P1$ ve $P2$ noktaları arasındaki doğru parçası ifadesi;

$L3 = \text{LINE}/P1, \text{PARLEL}, L0$; $P1$ noktasında geçen ve önceden tanımlanan $L0$ doğrusuna paralel doğru ifadesi;

$L4 = \text{LINE}/P1, \text{PERPTO}, L0$; $P1$ noktasında geçen ve $L0$ doğrusuna dik doğrunun ifadesidir.

$L5 = \text{LINE}/P1, \text{LEFT}, \text{TANTO}, C1$; $P1$ noktasından geçen, $C1$ çemberine teğet ve $P1$ noktasından bakıldığında sol tarafta kalan doğru parçası ifadesidir.

$L6 = \text{LINE}/P1, \text{RIGHT}, \text{TANTO}, C1$; $P1$ noktasından geçen, $C1$ çemberine teğet ve $P1$ noktasından bakıldığında sağ tarafta kalan doğru parçası ifadesidir.



Şekil 2.7. APT Dilinde Çizgi Tanımlama

$L7 = \text{LINE}/\text{LEFT}, \text{TANTO}, C1, \text{LEFT}, \text{TANTO}, C2$; $C1$ çemberinden bakıldığında, $C1$ ve $C2$ çemberlerine sol taraftan teğet doğrunun ifadesi;

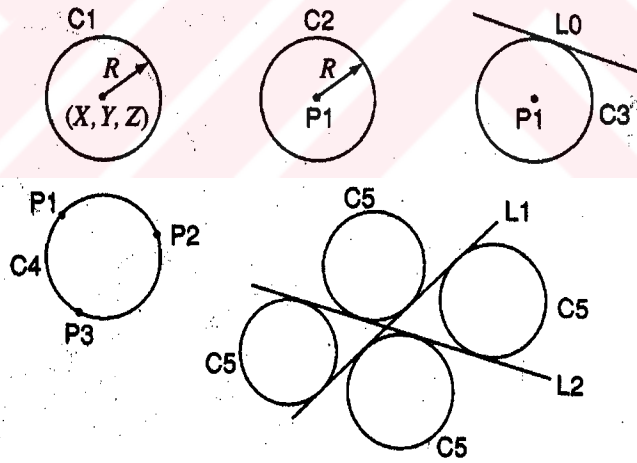
L8 = LINE/LEFT, TANTO, C1, RIGHT, TANTO, C2 ; C1 çemberinden bakıldığında, C1 çemberine soldan, C2 çemberlerine sağ taraftan teğet doğrunun ifadesi;

L9 = LINE/RIGHT, TANTO, C1, LEFT, TANTO, C2 ; C1 çemberinden bakıldığında, C1 çemberine sağdan, C2 çemberlerine sol taraftan teğet doğrunun ifadesi;

L10 = LINE/RIGHT, TANTO, C1, RIGHT, TANTO, C2 ; C1 çemberinden bakıldığında, C1 ve C2 çemberlerine sağ taraftan teğet doğrunun ifadesi;

L11 = LINE/P1, ATANGL, DEGREE, L0 ; P1 noktasında geçen, L0 doğrusuyla saat yönünün tersinde açı yapan doğru parçasının ifadesi.

APT dilinde bir çember tanımlanması için, Şekil 2.8 baz alındığında, şu yöntemler kullanılabilir;



Şekil 2.8. APT Dilinde Daire Tanımlaması

C1 = CIRCLE/X, Y, Z, R ; merkez noktası (X, Y, Z) ve yarıçapı (R) belirli çember denklemini;

C2 = CIRCLE/CENTER, P1, RADIUS, R ; P1 noktası merkez olarak atanmış, yarıçapı (R) belirli çember ifadesi;

C3 = CIRCLE/CENTER, P1, TANTO, L0 ; P1 noktasında merkezlenen ve L0 doğrusuna teğet doğru denklemi;

C4 = CIRCLE/P1, P2, P3 ; önceden belirlenen üç noktadan geçen çember;

$$C5 = \text{CIRCLE/} \begin{Bmatrix} XSMALL \\ XLARGE \\ YSMALL \\ YLARGE \end{Bmatrix}, L1, \begin{Bmatrix} XSMALL \\ XLARGE \\ YSMALL \\ YLARGE \end{Bmatrix}, L2, \text{RADIUS, R ;}$$

Şekil 2.8'te görüldüğü üzere çember C5, XSMALL, XLARGE, YSMALL ve YLARGE niteleyicileri kullanılarak dört farklı biçimde tanımlanabilir. Bu niteleyiciler çemberin merkez noktası, çizgi ve çember teğet noktası ile çizgi arasındaki ilişkiyi göstermektedir.

APT dilinde bu temel üç geometrik elemanın dışında düzlem ve yüzey gibi tanımlanabilecek bir çok geometri nesnesi bulunmaktadır. Bu geometrilerin tanımı yine benzer tekniklerle yapılmaktadır.

2.5.3.2. Hareket Komut Satırları

Yukarıda izah edildiği gibi çeşitli yöntemlerle, bir kere parça geometrisi tanımlandıktan sonra hareket komutları vasıtasıyla kesici takıma yön verilir. Her hareket komut satırı, yeni bir noktaya yada bir yüzey boyunca hareketi sağlar. İki grup hareket komutundan söz edilebilir, bunlar; noktadan noktaya ve kontur hareketlerdir.

Noktadan noktaya hareketi sağlayan üç hareket komutu vardır:

FROM/ Nokta_konumu

GOTO/ Nokta_konumu

GODLTA/ $\Delta x, \Delta y, \Delta z$

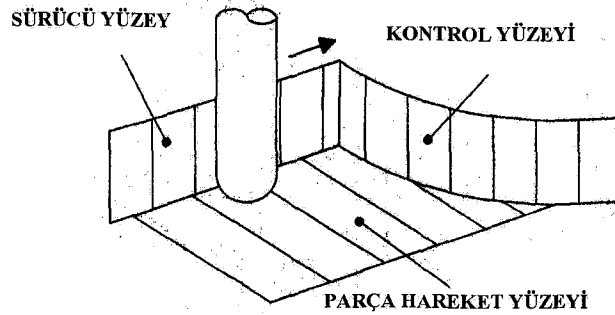
Nokta_konumu, X, Y, Z koordinat değerleri veya geometri komutlarıyla daha önceden tanımlanan sembolik bir değer olabilir.

FROM komutu, ilk hareket komutudur. Hareketin başlayacağı ilk noktayı belirtir.

GOTO komutu, kesici takımı bulunduğu noktadan belirtilen bir diğer noktaya hızlı şekilde hareket ettirir.

GODLTA komutu, kesici takımı bulunduğu konumdan Δx , Δy ve Δz ile belirtilen artırımlı mesafe değerleriyle takımı hareket ettirir.

Kontur tipi işlemlerde, Şekil 2.9'da gösterildiği gibi, üç yüzeyin belirlenmesi gerekmektedir. Bunlar, Parça Hareket Yüzeyi, Sürücü Yüzey ve Kontrol Yüzeyleridir. Parça Hareket Yüzeyi; kesici takımın uç noktasıyla temasın gerçekleştiği parça yüzeyidir. Parçadaki işlenecek yüzeyi tanımlar. Sürücü Yüzey; kesici takımın, yüzey boyunca kayarak yaptığı harekete yön verir. Kesici takım bu yüzey vasıtasıyla yönlendirir. Kontrol Yüzeyi; takım hareketini sınırlayan, parça hareket yüzeyinde bir sınır oluşturmaya yarayan yüzeydir. Kesici takım hareketini, kontrol yüzeyi boyunca sürdürür.



Şekil 2.9. APT Dilinde Üç Yüzey Tanımlaması

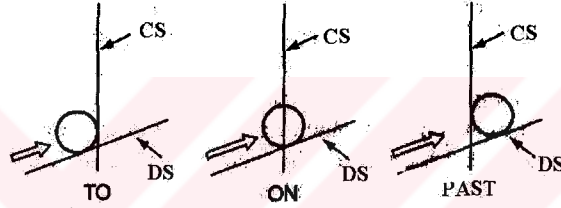
Kesici takımın kontrol yüzeyleri üzerinde hareket ettirilmesinden önce, takımın yüzeylere gönderilmesi gerekir. Bu üç yüzeyin hepsi GO komutu içinde tanımlanmaktadır;



$GO/\begin{Bmatrix} TO \\ PAST \\ ON \end{Bmatrix}$; sürücü yüzey, $\begin{Bmatrix} TO \\ PAST \\ ON \end{Bmatrix}$, Parça yüzeyi, $\begin{Bmatrix} TO \\ PAST \\ ON \\ TANTO \end{Bmatrix}$, Kontrol Yüzeyi

TO, PAST, ON, TANTO gibi niteleyici sözcükler, kontrol yüzeylerine bağlı olarak, kesici takımın konumlandırılacağı yeri gösterir. Örneğin, Şekil 2.10'da gösterildiği gibi, kesici takım, "CS" ile gösterilen kontrol yüzeyinde, "DS" ile gösterilen yönlendirici yüzey boyunca hareket ettirildiğinde üç farklı şekilde konumlanabilir.

TANTO niteleyici sözcüğü, sadece kontrol yüzeyi ile birlikte kullanılır.

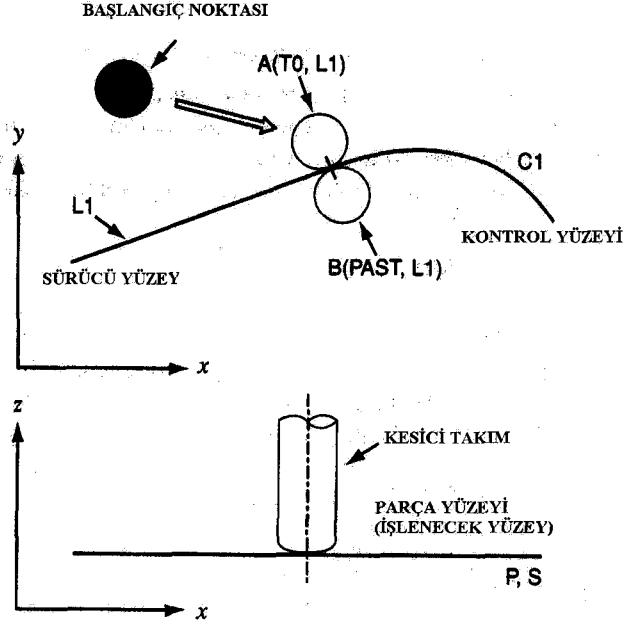


Şekil 2.10. APT Dili TANTO Niteleyici Tanımlaması

Şekil 2.11'de TANTO niteleyicisinin kontrol yüzeyine bağlı olarak takımı konumlandırması gösterilmektedir. Kesici takım konumları olan A ve B' ye çeşitli komutlar vasıtasıyla ulaşılmaktadır.

Konum A: GO/TO, L1, TO, PS, TANTO, C1

Konum B: GO/PAST, L1, TO, PS, TANTO, C1



Şekil 2.11. TANTO Niteleyicisinin Kullanımı

Şimdiye kadar üç kontrol yüzeyine göre kesici takımın başlangıç noktasına gönderilmesi göz önünde tutulmuştur. Takımın bir önceki hareket yönüne bağlı olarak hareket ettiğini düşünersek; bu durumlar için kullanılan komutlar Şekil 2.12'ye göre;

GOLFT/ ; önceki hareket yönünden sola ve yönlendirici yüzey boyunca hareket ettirme komutu,

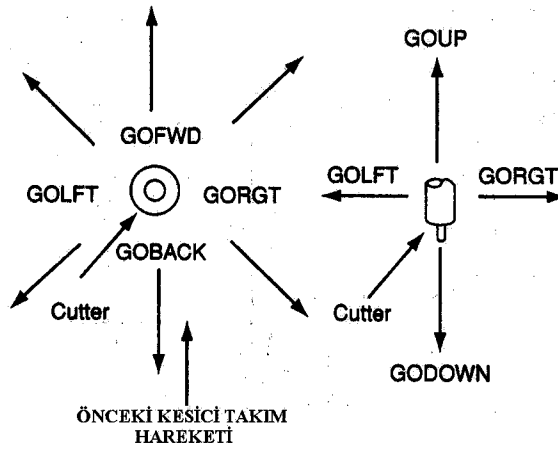
GORGT/ ; önceki hareket yönünden sağa ve yönlendirici yüzey boyunca hareket ettirme komutu,

GOUP/ ; yönlendirici yüzey boyunca yukarı doğru hareket (parça yüzeyinden uzaklaşma) hareketi,

GODOWN/ ; yönlendirici yüzey boyunca aşağıya doğru hareket (parça yüzeyine yaklaşma) hareketi,

GOFWD/ ; teğet bir konumdan, teğet doğrultu boyunca ileriye doğru hareket,

GOBACK/ ; teğet bir konumdan, teğet doğrultu boyunca geriye doğru hareket,



Şekil 2.12. Kesici Takım Bağıl Hareket Komutları

İlave APT komutları; postprosesör komutları olarak adlandırılan, tolerans, kesici bilgileri, başlangıç ve bitiş komutları gibi takım kesme (CL) dosyasına doğrudan aktarılan komutlardır.

APT diline ek olarak, benzer işlevi daha az veya daha çok yerine getiren diğer birçok program mevcuttur. Bunlardan en yaygın kullanılanlarından kısaca bahsetmek gerekirse;

ADAPT (Adaptation of APT), APT programlama sisteminin daha düşük kapasiteli bilgisayarlara adaptasyonu için IBM tarafından Amerikan Hava kuvvetleri desteğiyle geliştirdiği ilk dildir. ADAPT esnek modüler bir yapıya sahiptir. Alt-rutinlerin eklenip çıkarılabilmesi, küçük ve orta seviye bilgisayarlar için bu dili uygun hale getirmiştir. ADAPT konumlanma ve iki veya üç eksenle kontur hareketi için idealdir ancak üçten fazla eksen için programlama yeteneğine sahip değildir [Groover, 1984]

AUTOSPOT (AUTOMATIC System for Positioning Tools), yine IBM tarafından geliştirildi ve ilk defa 1962’ de tanıtıldı. Delik delme gibi konumlanma işleri için popülerdir. Sürekli takım hareketi uygulamalarını yapabilmesi amacıyla sonradan, ADAPT ile birleştirilmiştir.

EXAPT (Extended alt-rutinset of APT) 1964 yılında, APT dilinin Avrupa'da kullanılması için Almanya da birçok üniversitenin ortak katılımıyla geliştirilmiştir. APT ile uyumludur ve bu uyum dolayısıyla APT sisteminde kullanılan aynı postprosesör kullanılabilir. Üç versiyonu vardır; EXAPT I, noktadan noktaya konumlanan makinalar için, EXAPT II, tornalama operasyonu için, EXAPT III, üç eksen kontur operasyonları içindir.

COMPACT, manufacturing data systems, inc. (MDSI) tarafından geliştirilmiştir. COMPACT II ve APT en çok kullanılan programlama dilleridir. COMPACT II komutları İngilizce'ye benzer ve makinacılık terminolojisine yakın sözcüklerden oluşur. Bu dilin en büyük özelliği doğrudan makina kontrolörüne gönderilebilmesidir. Bu yüzden postprosesör kullanmaya gerek yoktur.

SPLIT (Sundsstrand Processing Language Internally Translated) programlama dili, Sundstrand şirketi tarafından, kendi tezgahlarında kullanılmak üzere geliştirilmiştir. 5 eksene kadar konumlanma ve kontur kapasitesine sahiptir. Postprosesör ise programın içinde oluşturulduğu için her tezgah kendine has SPLIT programı kullanır.

MAPT (Micro-APT) alt seviyeli bir APT dilidir. Mikro-bilgisayarlarda kullanılmak üzere geliştirilmiştir. APT ile aynı özellikleri taşır.

2.5.4. CAD Veritabanından Parça Programlama

APT diliyle program yapılmasıyla ilgili daha önceki örneklerde de görüldüğü üzere, programın büyük bir kısmının, parça geometrisinin tanımlanmasına ayrıldığı söylenebilir. Eğer parça geometrisinin herhangi bir CAD sisteminde dizayn edildiği düşünülürse, programlayıcının CAD veritabanında saklanmış olan bu geometrik verileri, parça programlamasında kullanmaya çalışması da son derece doğaldır. Parça geometrisi bir CAD sistemi ile modellenmemiş olsa bile, APT dili gibi herhangi bir programlama diliyle, özellikle çeşitli kompleks eğri ve yüzeylerle sınırlanıyorsa, parça geometrisinin tanımlanması yerine, bir CAD sisteminde modelleme daha uygundur.

Bu bahsedilen konularla altı çizilmek istenen fikirler sayesinde günümüz CAD/CAM entegrasyonlu sistemler gelişimini sağlamıştır. Entegre bir CAD/CAM sisteminde, bir NC program vasıtasıyla CAD veritabanından alınan geometri bilgileri, takım hareketlerine dönüştürülür.

Bir CAD/CAM sisteminde parça programlamadaki temel yaklaşımlar şu şekilde ifade edilebilir;

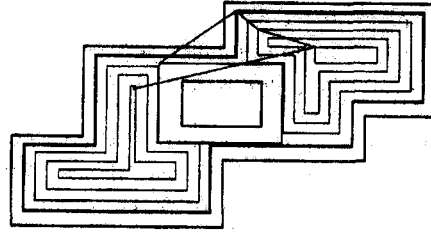
1. İşlenebilirlik açısından parça geometrisinin görünüşü önemlidir ve parça modelinde işlenecek geometri belki de ayrı bir katmana atılarak izole edilir. Geometriyi oluşturan parametreler değiştirilebilir yada takım hareketinin sınırlandıran ek bir geometri ilave edilebilir. Parça programlama için gerekli olan geometrik bilgiler, kullanılan NC tezgaha göre değişir. Sözelimi, torna tezgahında yapılabilecek çeşitli operasyonlar sadece 2 boyutlu profil geometri tanımlanmasını bekler. Bu profil geometri, bilgisayar destekli teknik resim programlarından doğrudan alınabilir.

Torna operasyonlarına benzer olarak, 2-2^{1/2} eksen freze ve delme işlemlerinde yine iki boyutlu CAD model iş görecektir. Örneğin Şekil 2.13(a) ve Şekil 2.13(b)'de gösterilen cep boşaltma işlemleri ile Şekil 2.13(c)'de gösterilen profil işlemleri, 2 boyutlu veya 3 boyutlu geometriden çıkarılabilir.

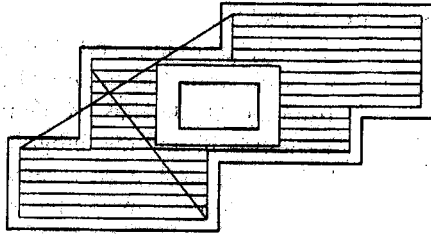
3 veya 5 eksenli yüzey işleme gerektiren operasyonlarda geometriyle ilgili daha fazla bilgiye ihtiyaç vardır ve bu bilgiler NC programa, yüzey modelleme veya katı modelleme yeteneklerine sahip bir CAD sistemi tarafından aktarılır. Bu durumda daha önceleri APT dilinde parça yüzeyi olarak ifade edilen işlenecek yüzey, bu yüzeye komşu ve APT dilinde kontrol ve yönlendirme yüzeyleri olarak isimlendirilen yüzeylerin enteraktif olarak programcı tarafından seçilmesi söz konusudur. Bahsi geçen bu enteraktif olaylar, CAD/CAM entegrasyonlu bir sistemin ne kadar büyük avantajlara sahip olduğunu kanıtlar. APT gibi bir dil kullanılarak, kompleks bir yüzeyin tanımlanması çok zordur hatta pratikte imkansızdır.

2. CAD/CAM sistemlerinde bir başka adım kesici takım geometrisinin tanımlanmasıdır. Genelde NC sistemleri, birçok değişik tipte kesici takım barındıran bir

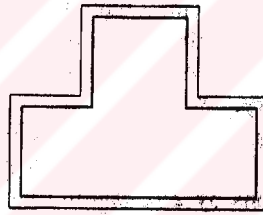
takım kütüphanesine sahiptir. Programlayıcı bu kütüphaneden uygun takım çağırabileceği gibi belli parametreleri tanımlayarak kesici takım oluşturabilir.



(a)



(b)



(c)

Şekil 2.13. 2 Ve 3 Eksende Yapılan Freze İşlemleri.

3. Kullanıcı gerekli takım yolunu, işleme operasyonlarının tipini ve planlarını, uygun kesme parametrelerinde tanımlayarak oluşturur. Takım yolu, kesici takımın izlemek zorunda olduğu ve bulunduğu başlangıç konumundan parça üzerinde işleme yapıp sonra tekrar başlangıç noktasına döndüğü yoldur. Yol genelde, her bir pasoda iş parçasından talaş kaldıracak şekilde, birbirini takip eden tekrarlardan meydana gelir. Bazı NC programları, basit parça ve delik delme, tornalama gibi operasyonlar için otomatik takım yolu planlama imkanı vermektedir.

4. Bir kez takım yolu planlaması yapıldıktan sonra, tanımlı kesici takım ve parça geometrisi vasıtasıyla takım yolu üzerindeki gerekli X, Y ve Z noktaları NC sistem tarafından hesaplanır. Takım yolu üzerinde bir çok noktanın operasyonda kullanılması

sonucunda, tam bir yüzey elde edilecektir. Bununla birlikte nokta sayısının artması büyük bir kesme dosyası oluşmasına ve yavaş DNC veri transferine sebep olacaktır.

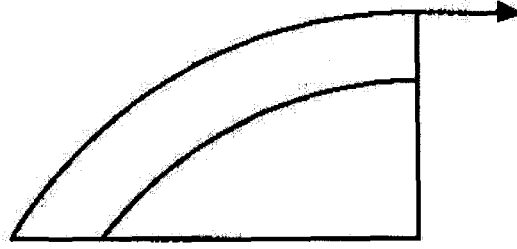
5. Takım yolu oluşturulduğu zaman grafik ekranda doğruluk analizi yapılabilmektedir. Analiz genelde kesici takımın, oluşturulan takım yolu boyunca, parça üzerindeki hareketlerinin simülasyonunu içeren bir safhadır. Analiz esnasında bir hata bulunursa takım yolu uygun bir şekilde değiştirilir. Makro komutları veya diğer detaylar, belirli bir işleme döngüsüne veya operasyona eklenir.

6. Bir kesme yerleşimi veri (CLDATA) dosyası, takım yolunun dosyaya yazdırılmasından oluşur. Kesme dosyası, tezgaha aktarılacak olan makina kontrol veri dosyasına çevrilmek üzere postprosesör programdan geçirilir.

2.5.4.1. Takım Yolu Oluşturulması

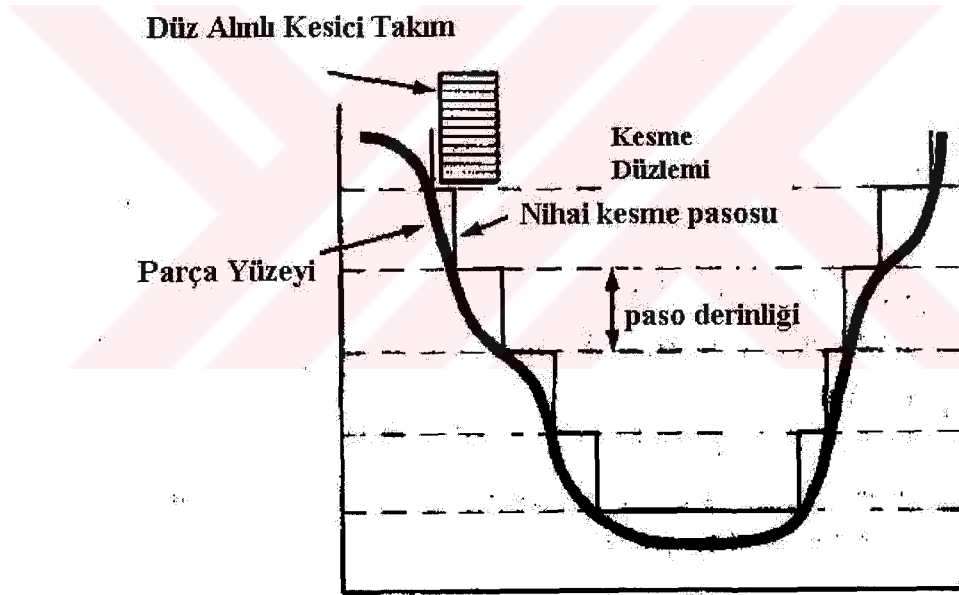
Freze tezgahlarında yapılan, 3 veya 5 eksen kontur yüzey işlemleri, birçok takım yolu oluşturulmasını gerektirebilir. Önemli miktarlarda talaş kaldırılması istendiğinde, normalde iki grup takım yolu hesaplanır: bunlardan biri kaba kesme işlemidir ve iş parçasından büyük hacimlerde talaş kaldırılır, diğeri ise nihai kesme işlemidir ve iş parçasını istenen son şekline getirir. Bazı durumlarda ara kesme işlemleri (semi-finish) gerekebilir.

Kaba kesme işlemi için, iki tip kesme prosesi söz konusudur. Birinci tip kesme işlemi, Şekil 2.14'de verildiği gibi, dökme parçalar gibi, hammaddenin (iş parçasının) zaten tasarlanan parça şekline yakın olduğu durumlarda kullanılır. Parça yüzeyleri (işlenecek yüzeyler) öncelikle, parça son halindeki yüzeylerin belirli bir aralıkta ötelenmesiyle belirlenir. Kaba kesme işlemi için takım yolu bu ötelenen yüzeyler yoluyla hesaplanır.



Şekil 2.14. Birinci Tip Kaba Boşaltma İşlemi

Şekil 2.15 de görüldüğü üzere, her bir katman için takım yolu, cep boşaltma işlemi (Şekil 2.13) gibi hesaplanır. Bu durumda, her bir katman için cebin sınırlarını oluşturan eğri, işlenecek yüzey ile katmana karşılık gelen yatay düzleminden çıkarılır. Böylece bir kez cebin sınırları belirlendikten sonra takım yolu, bu dış sınır eğrisinin, her bir seferde, belirli miktarlarda (paso genişliği) iç tarafa doğru ötelenmesiyle oluşturulur.



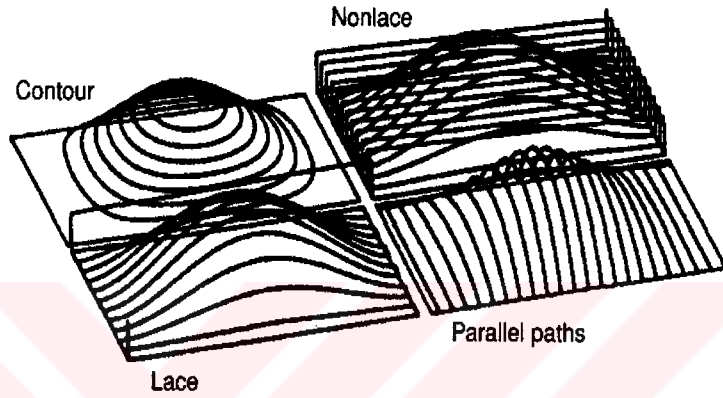
Şekil 2.15. Cep İşleme İçin Takım Yolu Çıkarılması

Cep işleme takım yolunu çıkarmadaki bir diğer yöntem; her biri cep sınırı ile çevrilmiş, paralel düz doğru parçalarının hesaplatılmasıyla sağlanabilir.

Nihai kesme için takım yolu hesaplanması, işlenecek olan parça yüzeyinde, bir dizi düz doğru parçaları vasıtasıyla eğriler uydurulması ve bu doğru parçaları boyunca kesici takımın yüzey üzerinde hareket ettirilmesini içerir. Bu yüzden kesme işleminin

hassasiyeti doğrusal yaklaşıma bağlıdır. Bu faktör, doğru parçalarıyla asıl eğri arasındaki maksimum sapmayı belirleyen bir tolerans değeriyle belirlenir.

Düz doğru parçaları vasıtasıyla yüzeye uydurulan eğrileri seçmek için bir çok yöntem mevcuttur. Fiziksel olarak bunun anlamı, bir yüzeyde takımın birkaç türlü hareket edebileceğidir. Şekil 2.16'da yüzeyde oluşturulan çeşitli takım yolları gösterilmektedir.



Şekil 2.16. Çeşitli Tip Takım Yolları [McMahon ve Browne, 1993]

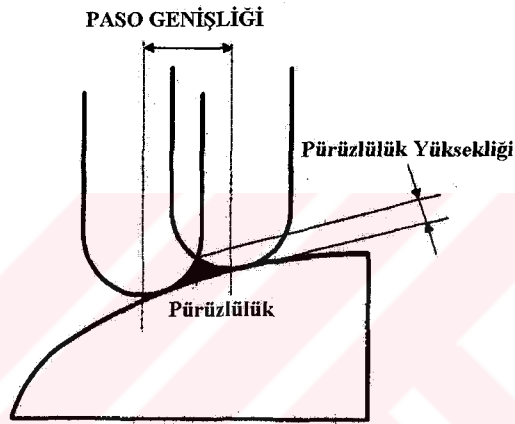
İşlenecek yüzey üzerinde bir kesici şu şekillerde hareket edebilir;

- Bir yüzey boyunca, herhangi bir yüzey parametresi değerinin sabit tutulduğu eğriler ve Lace kesme olarak adlandırılan, her iki ileriye veya geriye doğru hareket şeklinde veya Nonlace kesme olarak adlandırılan bütün kesme hareketinin bir uçtan diğerine aynı yönde olduğu hareket şeklinde olabilir.
- Yüzey üzerinde çevresel (kontur) hareket veya;
- Yüzey ile paralel düzlemlerin kesişmesiyle sağlanan takım yolu hareketi.

Genellikle, kullanıcılar takım yolu planlama işlemi boyunca bu üç tip kesici hareketinden birini seçerler. Seçilen örneklerde, eğriler arasındaki mesafe, istenilen

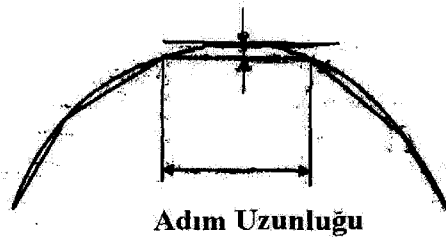
hassasiyet esas alınarak, NC program vasıtasıyla hesaplanır. Şekil 2.17’de görüldüğü gibi bu aralığa paso genişliği adı verilir ve yüzey işlenmesi bittikten sonra yüzey pürüzlülüğünü tayin eden bir faktördür.

Yüzeye uydurulan eğriler elde edildikten sonra eğri üzerindeki noktaların hesaplatılması gerekir. İki komşu noktayı birbirine bağlayan herhangi bir düz doğru parçası ile uydurulan eğri parçası arasındaki mesafe belirlenen toleranstan daha az olmalıdır. Bahsi geçen bu noktalarda kesici takım yüzeyle temas ettiğinden, bu noktalara kesme temas noktaları (Cutter Contact, CC-Points) adı verilmiştir.



Şekil 2.17. Paso Genişliği Ve Yüzey Pürüzlülüğü

Şekil 2.18’de gösterildiği gibi, orijinal eğri ile doğru parçası arasındaki sapma, adım uzunluğu olarak adlandırılan kesme-temas noktaları arasındaki mesafeye bağlı olarak artar. Daha küçük adım uzunlukları daha iyi bir hassasiyet sağlamalarına rağmen, izin verilen maksimum adım uzunlukları daha cazip olan kısa kesme dosyalarını oluşturur.



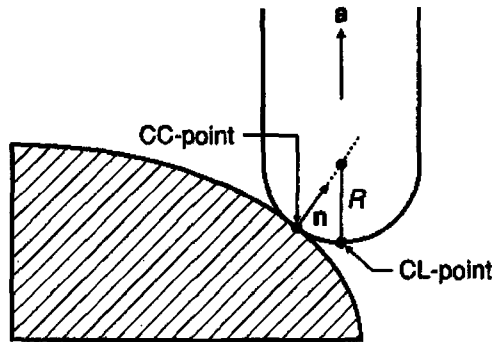
Şekil 2.18. Adım Uzunluğu

Şekil 2.19’da gösterildiği gibi, işlenecek yüzeydeki bir eğri Bezier eğrisi şeklinde temsil edilir. Eğri, tüm konveks kabuk (Şekil 2.19’da taralı bölgeler) kalınlığı belirli tolerans değerlerinden aşağı oluncaya kadar parçalara ayrılır ve bu parçaları oluşturan noktalar, tolerans değeri dikkate alınarak (Şekil 2.19’da D4 ve D7), kesici temas noktası (CC-point) olarak seçilir.



Şekil 2.19. Kesme Temas Noktaları Toleransları

Kesme temas noktalarının tayininden sonra kesme konum noktaları hesaplanır. Kesme konum noktaları NC kontrolör için gerekli noktalardır. Şekil 2.20’de gösterilen, küresel uçlu bir kesici takım için CL noktası, belirlenen CC noktasından Denklem 2.1’de verilen formülle hesaplanabilir.



Şekil 2.20. Küresel Uçlu Bir Takım İçin CL Noktası

$$\mathbf{r}_{cl} = \mathbf{r}_{cc} + R[n(u,v) - \mathbf{a}] \quad (2.1)$$

burada; $\mathbf{r}_{cl}$ ve $\mathbf{r}_{cc}$ sırasıyla CL noktanın ve CC noktasının pozisyon vektörleri,

R , kesici takımın yarıçapı,

$N(u,v)$, u ve v parametrelere değerlerine uygun olarak CC noktasında parça yüzeyine normal birim vektörün ifadesi,

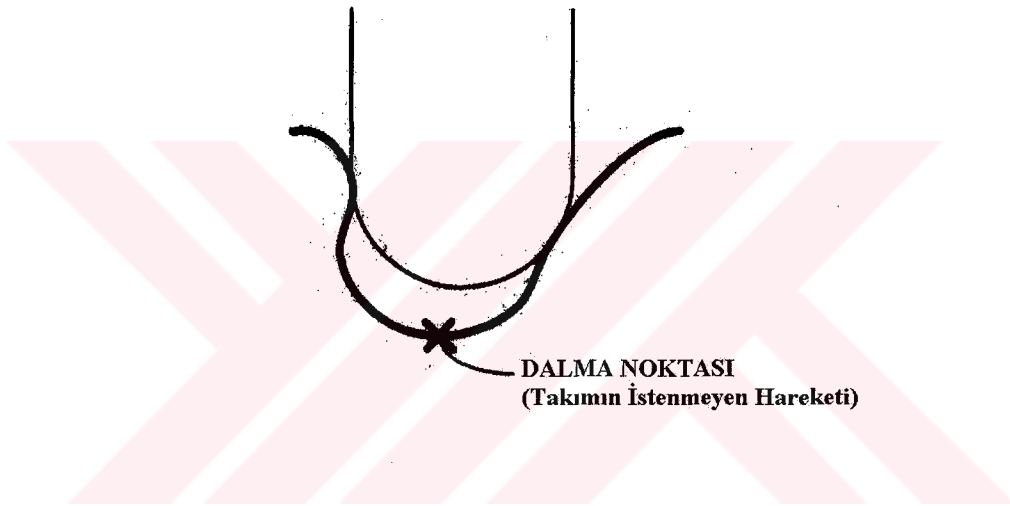
$\mathbf{a}$, takım eksenı boyunca birim vektör,

Üç eksen frezeleme için takım eksenı vektörü ($\mathbf{a}$) genellikle (0,0,1) değerlerindedir ancak 5 eksen frezelemede bu değerler değişiklik gösterir. 3 eksen işlemecilikte, takım eksenı her zaman iş parçasına göre sabit bir açıda, normal olarak z eksenıyle hizalandırılmıştır.

Başka tiplerdeki kesiciler için, benzer şekilde CC ve CL noktaları arasında bağlantı kurulabilir. 5 eksen işlemlerde, işlenen yüzeye uygun olarak kesici eksenı değişir. Prensipde kesici, yüzey pürüzlülüğünü azaltacak şekilde, yüzey normali ile hizalandırılır ancak pratikte, kesici eğri bir açı yapar. Bu yüzden, kesme işlemi takımın alt kısmında değil de yan kısmında oluşur. Bu şekilde bir kesim daha avantajlıdır çünkü; yuvarlak uçlu takımda olabildiğince geniş bir yay, iş parçasından talaş kaldırır ve ayrıca maksimum kesme hızlarına ulaşılmış olur. Bu işlem için 5 eksen işlemde, takım ekseninde dönmeler söz konusu olduğundan, takım yolu hesaplatılması daha karmaşıktır.

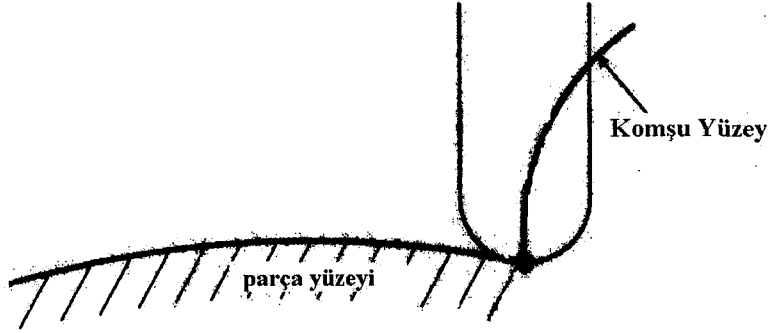
Buraya kadar, oluşturulan CC-noktalarından CL noktalarının nasıl hesaplatıldığı anlatılmıştır. Bununla birlikte CL noktaları hesaplanmasındaki farklı bir yaklaşımda, özellikle küresel uçlu takımlar için, işlenecek yüzeyin ötelenmesidir. Bu, işlenecek parça yüzeyinin, küresel uçlu kesici takımın yarıçapı kadar ötelenmesi ve buna ek olarak, Şekil 3.12'de gösterilen yöntemdeki gibi, ötelenen bu yüzey üzerinde birtakım eğrilerin oluşturulmasıyla sağlanır. Bundan sonra eğriler üzerinde hesaplatılan noktalar, kesici takımın merkezinin konumlandığı noktalardır ve CL noktalar ise bu noktaların pozisyon vektörlerine, Denklem 2.1 eşitliğinden elde edilen $-R\mathbf{a}$ değerinin eklenmesiyle elde edilir.

Kesici takımın hesaplanan takım yolu üzerinde hareketi esnasında, bazen dalma (gauge) adı verilen istem dışı kesme hareketleri de gerçekleşebilir. Şekil 2.21'de gösterildiği gibi dalma, işlenecek olan konkav bir yüzeyin eğriliğinin, kesici takım yarıçapından daha küçük olması sonucunda oluşur. Bu problem, yüzey eğriliğinden daha küçük yarıçapa sahip kesici takım kullanılarak çözümlenebilir. Bununla birlikte takımın çok küçük seçilmesi halinde, verimsiz bir kesme işlemi gerçekleşebilir. Bu durumda işlenecek yüzey parçalara bölünebilir ve yüzeyin önemli bir kısmı büyük bir takım ile işlenirken, geride kalan yüzey eğriliği düşük olan bölge daha küçük bir takım ile işlenebilir.

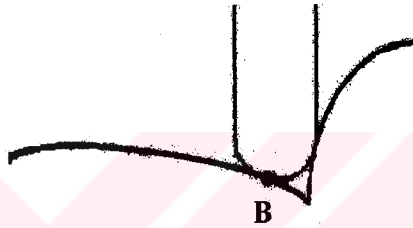


Şekil 2.21. Kesici Takımın İstenmeyen Hareketi

Bir diğer dalma tipi ise, Şekil 2.22'de gösterildiği gibi, işlenecek parça yüzeyiyle komşu yüzeylerin birleştiği durumlarda söz konusudur. Problem, Şekil 2.22'de A ile gösterilen ve iki yüzeyin kesişimi olan bu noktanın, CC noktası olarak hesaplanmasından kaynaklanmaktadır. Bu problemde Şekil 2.23'de gösterildiği gibi, A noktasındaki CC noktasının B noktasına kaydırılmasıyla çözülebilir. Kesici takımın limit noktası olan bu nokta, yüzeylerin kesişmesi ile elde edilir. Eğer hem işlenecek parça yüzeyi hem de bu yüzeye komşu olan yüzeyler, takım yarıçapı kadar ötelenir ve bu ötelenen yüzeylerin kesişim noktaları hesaplatılabilirse, dalmanın engellenileceği nokta bulunmuş olur. Ancak değişik takımlar için bu hesaplama çok daha karmaşıklmaktadır.



Şekil 2.22. Takımın Komşu Yüzeydeki Dalma Hareketi.



Şekil 2.23. Takımın Komşu Yüzeydeki Dalma Hareketinin Engellenmesi.

5 eksen işlemede dalma kontrolü analizi, çok daha karmaşıktır çünkü; her bir oryantasyon da tüm takım geometrisinin düşünülmesi ve kesici takımın tutucular, fener mili, tabla gibi çeşitli ekipmanlara çarpıp çarpmayacağı kontrol edilmelidir.

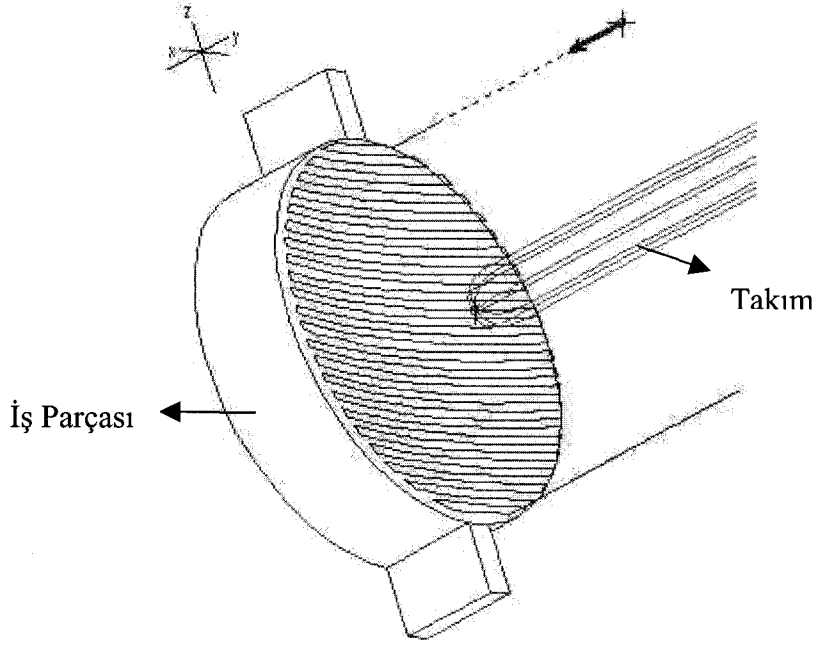
2.5.4.2 Takım Yolu Benzetimi Ve Doğrulaması

Takım yolu genellikle, gerçek bir işleme operasyonu boyunca takıma rehberlik eden bir çok koordinat değerine sahiptir. Bu değerleri elle kontrol edip doğrulamak mümkün değildir. Geçmişte operatörler programın doğruluğunu, çok zaman harcayarak, ağaç veya plastik malzemeleri işleyerek test etmişlerdir. Günümüzde doğruluk kontrolünü yapan yazılımlar mevcuttur. Dolayısıyla doğruluk kontrolü için çok fazla zaman alan bu test yerini, takım yollarının görülebildiği, grafik ekranda görsel doğruluk kontrol yazılımlarına bırakmıştır. Bu yazılımlar sayesinde tezgahlar, sadece gerçek işlenecek parçalar için kullanılmaktadır. Ayrıca bu imkan, parça programcısını görsel olarak şunları kontrol etmesine olanak tanır;

- Kesicinin sadece, hammaddenin (iş parçasının) gerekli kısımlarından talaş kaldırıp kaldırmadığını;
- Kesicinin mengene veya tutucu gibi herhangi bir ekipmana çarpıp çarpmadığını;
- Kesici takımın bir cebin kenarlarından veya tabanından talaş kaldırıp kaldırmadığını ve;
- Takım yolunun verimli-etkili olup olmadığını kontrol edebilir.

Şekil 2.24'te gösterildiği gibi, görselliğin ve herhangi bir talaş kaldırma prosesi benzetiminin en kolay yolu, takım yolunun, modellenen parça üzerinde birtakım doğru parçaları şeklinde gösterimidir. Gösterilen her bir parça çizgi, CL dosyadan satır satır okunan değerlerin tekabülüdür. Bu tür bir benzetimde parça programlayıcısı, kesici takımın planlandığı şekilde hareket edip etmediği hakkında genel bir bilgi edinir, ancak, benzetimde sadece kesici konumları gözüktüğünden başka bir ifadeyle, iş parçasındaki değişiklikler gözükmediğinden, dalma kontrolü ve artan malzemeye ait bilgileri elde edemez.

En ideal simülasyon şekli katı model halindeki iş parçasının, takım hareketleriyle tasarlanan parçaya dönüştürüldüğü şekildir. Bu, takım hareketlerinden oluşturulan süpürme hacimsel modellerin, iş parçası modelinden çıkarılmasıyla sağlanır. Programlayıcı ayrıca iş parçası ile tasarlanan parça arasında karşılaştırma yaparak (bu iki model arasında Boolean işlemleri yapmak gibi) takım yollarının doğruluğunu kontrol edebilir. Bu yaklaşımın temelde problemi hesap edilebilirlik açısından masraflı olmasıdır. Benzetimin yükü takım hareketleri sayısının dördüncü kuvvetiyle orantılıdır. Tipik bir CL kesme dosyası içinde 10000 takım hareketi olduğu düşünüldüğünde, bu durumun görsel olarak hesaplanması, sistem için aşırı derecede yük demektir.

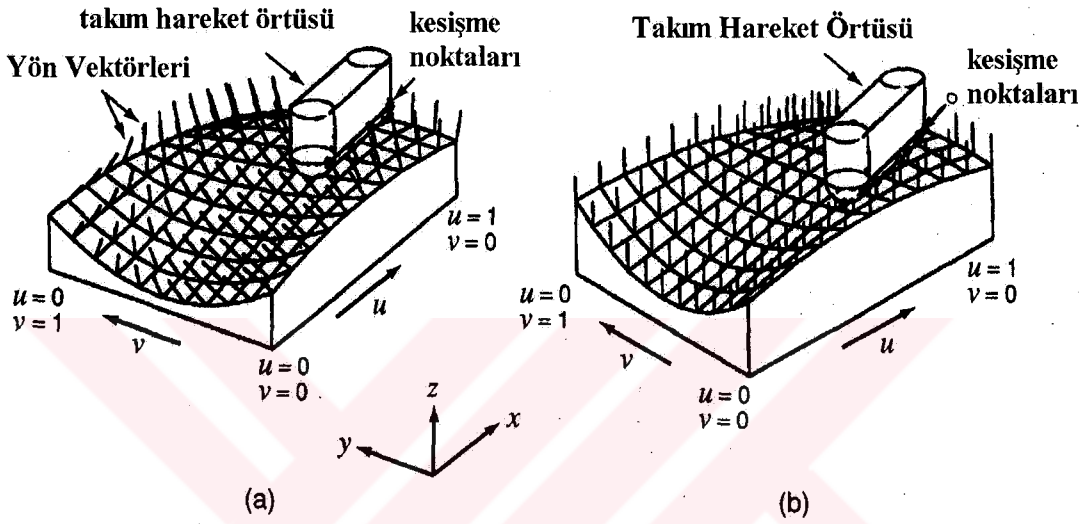


Şekil 2.24. Temel Takım Yolu Simülasyonu

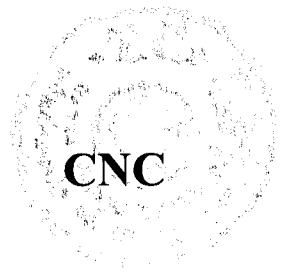
İş parçasını işlendiği gibi göstermedeki etkinliği artırmak için, çeşitli simülasyon teknikleri tasarlanmıştır. Bu tahmini yöntemlerde, simülasyon zamanı takım hareketleriyle doğrusal olarak artar. Bu metotlardan birinde, parça çeşitli karelere bölünür ve iş parçası ise dikdörtgen şeklinde sütunlarla ifade edilir. Bu tür yapılara 3 boyut “Histogram” adı verilmiştir. Bu yaklaşımın temel fikri; takım hareketleriyle bu dikdörtgen şeklindeki sütunların boyutlarının değiştirilmesidir. Bu yüzden her bir kolon iş parçasının yüksekliği ile başlar ve her bir takım hareketinde, sütunların yükseklikleri aşağıya doğru değişir. Bu şekilde iş parçası, takım hareketleriyle sanki kesiliyormuş gibi davranır [Anderson, 1978].

Bir başka yaklaşım ise nokta vektör tekniğidir. Bu yaklaşımda işlenecek yüzey yaklaşık olarak bir dizi noktalarla ifade edilir ve yön vektörleri hem iç hem de dış tarafı gösterecek şekilde, her noktada, yüzey normallerine dik olarak oluşturulur. Her bir CL dosya satırını takip eden, talaş kaldırma simülasyonu boyunca, bu vektörlerle, takım hareketlerinin yaklaşık örtüsü arasındaki kesişme noktaları hesaplanır. Eğer bir vektör örtü ile kesişiyorsa uzunluğu azalır. CL dosyadaki son satırda işlendiğinde, son vektörün kesin uzunluğu, kalan hammaddenin miktarına tekabül eder (iş parçasının yukarısında) veya (iş parçasının altında) dalma derinliğini verir. Bu yöntem 3 eksen talaş kaldırma işlemine uygulandığı kadar, 5 eksen talaş kaldırma işlemine de uygulanabilir. 5 eksen

işleme takım örtü yapısı, 3 eksen işlemeden daha karmaşıktır. 3 eksen işleme simülasyonu, genelde Z yönünde olan takım eksenindeki yön vektörlerinin kullanılmasıyla çok daha basitleştirilmiştir. Şekil 2.25(a)'da yön vektörlerinin yüzeye normal olması, Şekil 2.25(b)'de ise yön vektörlerinin her yerde z eksenı boyunca olduğu gösterilmiştir [Jerad et al, 1989].



Şekil 2.25. Benzetimde Nokta Vektör Tekniği.



3. CAD/CAM SİSTEMLERİNDEN CNC TEZGAHLARINA VERİ AKTARIMI

3.1. Giriş

Gerek APT, gerekse CAD/CAM sistemleri ile bilgisayar üzerinde gerçekleştirilen tasarım ve imalat sürecinin son basamağında postprosesör yazılımları bulunmaktadır. Belirli bir CNC tezgahıyla, CAD/CAM sistemlerinin iletişimi ancak bu yazılım sayesinde olmaktadır.

Bu tezde uygulama olarak, 5 eksen bir tezgah için, C++ Builder 3.0 genel programlama dili kullanılarak bir postprosesör yazılımı geliştirilmiştir. Dolayısıyla bu bölümde konuyla ilgili 5 eksen işleme mantığı, kesme dosyası özellikleri, belirli tezgahların kinematikleri gibi konular işlenmiştir.

3.2. Bilgisayar Destekli 5 Eksen İşleme

5 eksen işleme, kompleks yüzeylerin işlenmesinde 3 eksen geleneksel işleme yöntemlerine göre birçok avantajlar sağlar. İşleme zamanını azaltır, nihai yüzey kalitesini artırır.

CNC tezgahları genellikle yontma (sculptured) yüzey denilen kompleks yüzeylerin imalatında kullanılmaktadır. 3 eksen bir tezgahta bu işlem küresel uçlu kesici takımlar kullanılarak yapılırken, 5 eksen bir tezgahta düz alınlı kesici takımlar kullanılabilir.

3.2.1. NC Freze Tezgahlarının Sınıflandırılması

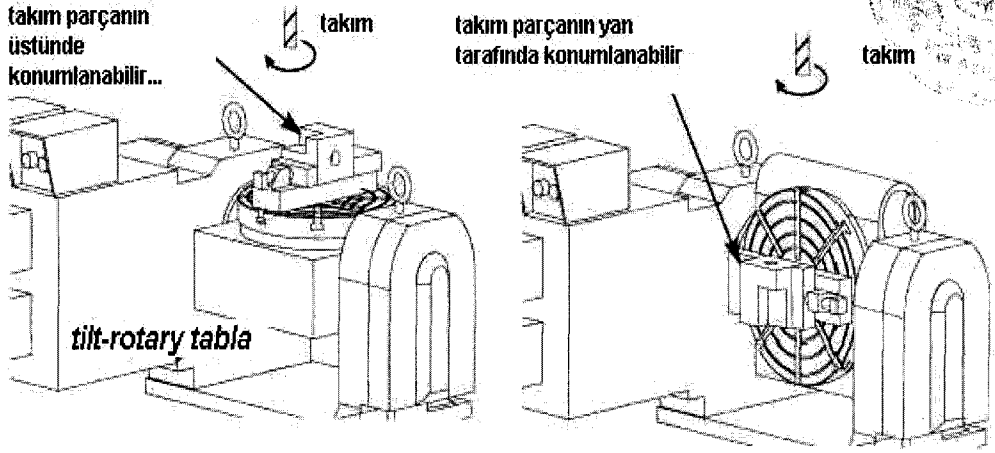
CNC tezgahları sahip oldukları eksen sayıları ile tanımlanırlar. Bir 3 eksen freze tezgahı iş parçasına göre 3 yönde hareket etme yeteneğine sahiptir. Özel bağlama aparatlarının ve küresel uçlu kesici takımların kullanılması ile bu tezgahlar oldukça esnektir ve düşük veya yüksek miktarda hacimsel boşaltmalar için çok uygundur. 5 eksen bir tezgahta kesici takım, 3 boyutlu uzay içinde oryantasyon hareketleri yapabilir. 5 eksen tezgahlar özellikle düşük miktarlardaki hacim ve daha kompleks parçaların üretiminde kullanılırlar.

Üç eksen işleme teknolojisinden 5 eksen işleme teknolojisine geçiş ilk anda zor ve yıldırıcı gelebilir. Bununla birlikte 5 eksen tezgahlar 3 eksenle sağlanamayacak kadar esneklik ve etkinlik sağlarlar. 5 eksen bir tezgahta, kompleks parçaların üretimi sadece basit bir bağlamayla sağlanır. Böylelikle kompleks ve pahalı tutucular kullanmaya gerek kalmaz. 3 eksen tezgahlarda ancak özel formlara sahip takımlarla yapılabilecek geometrik formlar, bu tür tezgahlarda rahatça işlenebilir. Bir uçak motoru türbini gibi sadece 5 eksende işlenebilecek parçalar işlenebilir ve en önemlisi serbest form yüzeyler, en iyi 5 eksen tezgahlarla elde edilebilirler.

3.2.2. 5 Eksen İşleme İhtiyacı

Şekil 3.1'de gösterildiği gibi, 5 eksen bir tezgahta kesici takım, iş parçası üzerindeki değişik konumlara, parça düzeni değiştirilmeden ulaşabilir. Örneğin 5 eksen tezgahta parçanın yüzeylerine basitçe parçayı döndürerek ulaşılabilir.

3 eksen bir tezgahta parça üretilirken istenen tutucu ve çeşitli ayar değişikliklerine, bu tür tezgahlarda gerek duyulmaz.



Şekil 3.1. 5 Eksen Tezgahta Kesicinin İş Parçasına Göre Konumlanması

5 eksen tezgahlar serbest formdaki yüzeylerin işlenmesinde çok belirgin bir avantaj sağlarlar. Geleneksel 3 eksen işlemede aynı yüzey, küresel uçlu bir takımla birbirine çok yakın paso genişlikleri kullanarak işlenir. Takımın her geçişinde yüzeyde silindirik bir iz kalır. Bu izler nedeniyle tam olarak serbest form yüzey işleyebilmek mümkün olmaz. Küresel uçlu kesici takım, kesme işleminin çoğunu, kesme yüzeyindeki teğetsel hızının en az olduğu taban merkezine yakın olan kısımlarda yapar. Bu yüzden yüzey kalitesi düşüktür ve yüzeylerde izler oluşmaktadır. Genellikle 3 eksen işlemeden sonra daha yumuşak ve kesin bir yüzey oluşturmak için, taşlama veya polisaj gibi ekstra işlemlere ihtiyaç duyulmaktadır. 5 eksenli bir tezgahta ise, yüzeyde elipssel izler bırakan düz alınlı kesici takımlar, yüzeye uygun şekilde açısal olarak konumlandırılabilirler. Eğim açısının değiştirilmesiyle, izlerin ölçüleri değişir ve tasarlanan yüzeye çok daha yaklaşık bir yüzey sağlanır. Buna ek olarak kesme işlemi, kesme hızının en yüksek olduğu, açısal konumlandırılmış takımın kenarında gerçekleşir. Böylece küresel uçlu takımlara göre çok daha kaliteli bir nihai yüzey ortaya çıkar.

Daha az takım geçişleri vasıtasıyla nihai yüzey işlemindeki zamanın azaltılması, tüm modern imalat endüstrisinde baştan sona üretimin maliyetini düşüren bir rol taşır. Bahsi geçen serbest form yüzeyler ise genellikle kesici takım ve kalıp kullanılarak üretilen tüm modern ürünlerin ortak elemanıdır. Kalıp üretiminin sadece Amerika Birleşik Devletlerinde 20 milyon dolarlık bir endüstrisi vardır. Basit

küçük bir kalıbın yapılması 1300 saati tutabilir. Bu zamanın %66'sı polisaj veya taşlama işlemi de dahil olmak üzere nihai işlemeye harcanmaktadır. Bu alanda 5 eksen işleme 3 eksen işlemeye göre çok daha kuvvetlidir, çünkü kesici takımın daha az pasosunu gerektirir ve daha kaliteli bir yüzey oluşturur [Warkentin et al, 2001].

3.3. 5 Eksen Tezgahın Elemanları

5 eksen bir işleme sisteminin 4 elemanı vardır, bunlar;

- Tezgahın Tipi,
- Tezgah Kontrolörü,
- CAD/CAM Yazılımı,
- Personel,

şeklinde ayrılabilir. Her 5 eksen sistem, her türlü iş için uygun değildir. Uygun elemanların seçimi, 5 eksen işleme vasıtasıyla elde edilecek olan verimliliği artıracaktır.

3.3.1. 5 Eksen Tezgahın Konfigürasyonu

Değişik tiplerdeki 5 eksen tezgahlar, kendilerine göre birtakım avantajlar taşırlar. Burada düşünülmesi gereken en önemli konu sistemin rijitliği, çalışma hacmi, ve sistemin hassasiyetidir. Buna ek olarak 3 eksen de olduğu gibi, otomatik takım değiştiricisi, ilerleme ve dönme devri gibi çeşitli özellikler, 5 eksen içinde önemlidir.

Rijitlik tüm freze tezgahlarında istenen bir özelliktir çünkü; konumlanma hassasiyetini arttırır ve büyük oranlarda talaş kaldırmaya izin verirler. 5 eksen de rijitlik bir problem olarak karşımıza çıkabilir çünkü; tezgahdaki dönebilen mafsallar genellikle lineer kaymalı mafsallardan daha esnektirler. Genel olarak 5 eksen sistem ne kadar rijitse o kadar pahalıdır. İstenen rijitliğe, kesilecek olan malzeme tipine ve kesme ölçülerine bağlı olarak karar verilebilir.

Tezgahın çalışma hacmi ayrıca büyük önem taşır. Bu ise mafsal hareketlerinin erişimiyle alakalıdır. Bu hareket menzili iş parçasının maksimum ölçülerini belirler ve takımın iş parçası üzerinde ulaşabileceği noktaları etkiler.

Hassasiyet, 5 eksen bir tezgahta en kritik özelliktir. Dönel eksenlerdeki konumlanma hataları, dönme merkeziyle olan uzaklığa bağlı olarak büyümektedir. Bunun bir sonucu olarak, tezgahla bağlantılı hata, çalışma alanındaki pozisyona göre değişir. Bu olay, iş parçasında beklenen hassasiyetin önceden söylenmesini zorlaştırır.

3.3.2. Makina Kontrolörü

Bir CNC tezgah kontrolörü tezgahı kontrol eder, G kodlarını çalıştırır ve tezgah ile operatör arasında kullanıcı arayüz sağlar. Kontrolör 5 eksen de eşzamanlı hareketi sağlayabilmelidir. Bunun anlamı, bir kesme hareketi boyunca 5 eksenin her biri, pozisyondan pozisyona doğrusal biçimde hareket edebilmelidir. Örneğin, bir hareketin ortasındaki noktada, tüm eksenler yarı yol almış olmalıdır. Bu özellik sayesinde, daha yumuşak kesme hareketleri sağlanacaktır.

Bazı kontrolörler sadece 5 eksen konumlanmayı sağlarlar. Bu durumlarda kontrolör lineer enterpolasyonu desteklemez ve eksenler eşzamanlı hareketi sağlayamazlar. Bu tür tezgah kontrolörü daha basittir ancak, serbest yüzey işlenmesi gibi birtakım operasyonları gerçekleştirme kapasiteleri azalmaktadır.

46

Kontrolörler verileri olabildiğince hızlı işlemelidir. 5 eksen serbest yüzey işleme operasyonunda, birbirine çok yakın kesici takım konumlamaları söz konusu olabilir. Bunun anlamı, birkaç milimetrelik bir kısım üzerinde, kontrolör, takım pozisyonunu, örneğin 100 defa değiştirmek zorunda kalabilir. Eğer kontrolör, bu takım konumlamalarını yeterince hızlı işleyemezse, ilerleme hızını düşürecektir. Bu durum işleme prosesini yavaşlatacak ve nihai yüzey kalitesini değiştirecektir. Ayrıca kontrolörler, 5 eksende iş parçası yanlarındaki aşırı, ağır kesmeler boyunca eksenlerini kilitleme yeteneğine sahip olmalıdır.

3.3.3. CAD/CAM Yazılımı

CAD/CAM yazılımları CNC tezgah ile kullanıcı kişi arasında arayüz sağlarlar. CNC tezgahları ise G kodları adı verilen, özel kodlar vasıtasıyla programlanır. CAM programları, CAD modellerinden doğrudan G kodları oluşturmak için kullanılmaktadır. Bir CAM sistemi, G kodlarını iki aşamada oluşturur. Birincisi, iş parçası koordinat sistemine göre takım konumlarının listelendiği, kesme (CLDATA) dosyasının oluşturulması. İkincisi ise, bir postprosesör yardımıyla bu kesme (CLDATA) dosyasının, makina G kodlarına dönüştürülmesidir.

5 eksen işleme için uygun CAD/CAM paketinin seçilmesi önemlidir. Bir çok CAM paketi 2 yada $2^{1/2}$ eksen için takım yolu çıkartabilir. Bu gibi CAM sistemlerinde, XY düzleminde eşzamanlı hareket elde edilebilir ancak, Z yönünde sadece konumlama yaparlar. Basit parçaların işlenmesi için bu yeterlidir. Bununla birlikte, bu tür sistemler serbest yüzey işleme kapasitesine sahip değildir.

Günümüzde birçok CAM yazılımı, 5 eksen işleme kapasitesine sahip olduğunu iddia eder. Bununla birlikte, bu sistemlerin taşıdığı özellikler birbirlerine göre büyük değişiklikler göstermektedir. Bu paketlerden bir çoğu, ancak 5 eksen yanal işlemeye uygun olan, eksen konumlanma hareketini sağlayabilirler. Bunun yanı sıra, 5 eksen serbest yüzey işleme yeteneğine sahip değildir [Choi et al, 1993].

Bazı CAM paketleri, kesici takım ucunu yüzey üzerine yerleştirerek ve takım ekseninin yüzey normaliyle hizalandırılmasıyla serbest yüzey işlemesi yapabilir. Bu teknik kullanıldığında, tasarlanan yüzeydeki konkav alanlarda, kesici takımın dalması söz konusudur. Bu tür sistemlerde kesici takımın, dalma limitine kadar ötelenmesi mümkün olabilir ancak bu yüzey kalitesinde bir miktar azalmaya neden olacaktır. En karmaşık CAM paketleri, takıma eğim vererek nihai işlemeyi gerçekleştirebilir. Burada takım yüzey normaliyle belli bir açı yapabilir ve bu açıyla yüzeye temas eder. Bu teknik çok daha etkili bir talaş kaldırma sağlar. Bununla birlikte günümüzdeki en son teknoloji CAM paketleri, literatürde sunulmakta olan ileri seviye 5 eksen işleme tekniklerinden hala geridedir.

Takım yolu oluşturmada CAD/CAM paketlerine ek olarak, sayısal doğrulama yazılımları, 5 eksen işlemede önemli bir rol oynamaktadır. Doğal olarak kompleks olan 5 eksen takım hareketleri, tezgah, kesici takım, iş parçası ve tutucular arasındaki çeşitli çatışma problemlerinin sezilmesi maksadıyla grafik ekranda benzetim (simülasyon) edilebilir. Bu tür yazılımlarla talaş kaldırma prosesi de görülebilmektedir. Bu özellikle dalmaların fark edilmesinde kullanışlıdır. Tasarlanan yüzeye herhangi bir dalma söz konusu olduğunda, dalma giderilecek şekilde takım pozisyonunun değiştirilmesi gerekir. Talaş kaldırma simülasyonu ayrıca, kesme işleminden sonra yüzeydeki kalıntıların ölçülerini değerlendirmede kullanılır. Bu optimum ölçülerdeki takım ve ilerleme hızının tayinine yardım edecektir.

3.3.4. Personel Eğitimi

5 eksen işleme sisteminin en son elemanı, tezgahı programlayan ve kontrol eden personelin eğitimidir. 5 eksen işleme 3 eksen işlemeden çok daha kompleksir ve tezgah hareketlerinin iyi anlaşılmasını gerektirir. 5 eksen yardımıyla parça imalatı, görsellik ve uygun takım yolu inşası için 3 boyutta iyi bir beceri gerektirir. CAM yazılımı için esaslı bir eğitim, özellikle işlenecek yüzeylerde 5 eksen işleme açısından birçok yaklaşım olabileceğinden dolayı çok önemlidir. Kesici takım boylarının belirlenmesi ve iş parçasının dönel mafsallara bağlı olarak yerleştirilmesi gibi, tezgah ayarları yapılırken özel bir özen gösterilmelidir.

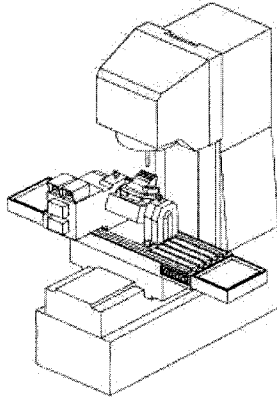
3.4. 5 Eksen Freze Tezgahlarının Kinematiki

5 eksenin faydası, kesici takımın iş parçasına bağılı olarak, herhangi bir eksen de oryantasyon yapabilme yeteneğıyle ortaya çıkar. 5 eksen işleme büyük bir oranda tezgahın konfigürasyonu ile bağımlıdır. Hemen hemen tüm 5 eksen CNC tezgahlar, farklı postprosesörler kullanılır ve böylece dönel eksenlerdeki hareketleri kontrol edebilirler. Gerçekte postprosesörün genel kesme verilerini belirli bir tezgaha bağılı G-kodlarına çevirmesi için, kesici takım ve iş parçası düzeni hakkındaki bilgilere ihtiyacı vardır. Bu nedenlerden dolayı, bu kısımda tezgah kinematikleri incelenecektir.

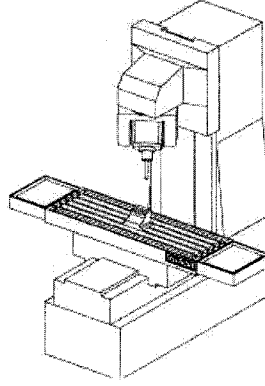
5 eksen freze tezgahları, sahip oldukları lineer (T) ve dönel (R) eksenlerinin sırası ve kombinasyonuna bağılı olarak sınıflara ayrılır. Örneğın bir tezgah üç lineer transformasyon ve iki rotasyon eksene sahipse, TTTRR şeklinde ifade edilecektir. 5 eksen bir freze tezgahını meydana getirirken, olası bir çok kombinasyonda eksen kullanılabilir. Bununla birlikte gerçekte ortak kullanımda, sadece üç tip tezgah konfigürasyonu söz konusudur.

1. RRTTT; bir açısal eğilmeli ve dönel hareket yapabilen tablanın, üç lineer eksen üzerine monte edilmesiyle oluşturulan ve eğilmeli-döner (tilt-rotary) tip tezgah olarak ifade edilen konfigürasyonda 5 eksen tezgahlar.
2. TTTRR; üç eksen ile kesici takımın iki dönel eksen de hareket ettirildiğı ve genellikle bilek (wrist veya Euler) tipi olarak tanınan tezgahlar.
3. RTTTR; bir döner tablanın üç lineer eksen üzerine monte edildiğı ve diğeri bir rotasyon ekseninin de kesici takımda kullanıldığı 5 eksen tezgahlar.

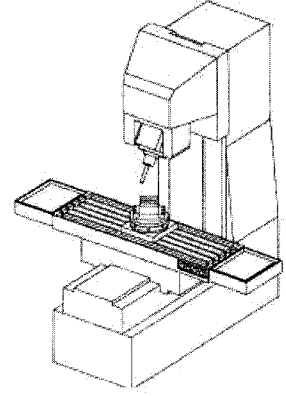
Yukarıda bahsedilen bu üç tip 5 eksen tezgah konfigürasyonu Şekil 3.2’de gösterilmiştir. Bir başka olası tip, örneğın TRTTR gibi bir tezgah konfigürasyonundan bahsedilmemiştir çünkü, bu tip rotasyon ve transformasyon eksenlerinin karışımından oluşan tezgah konfigürasyonunun tasarımı zordur.



Eğilmeli-döner tablalı tip



Bilek tipi



Döner tablalı tip

Şekil 3.2. 5 Eksen Tezgah Tipleri

Her bir tezgah konfigürasyonun kendine göre avantajları ve dezavantajları söz konusudur.

Bilek (wrist) tip tezgahlar, en kolay programlanabilir tezgahlardır ve çok geniş iş parçalarının işlenebileceği bir çalışma alanına sahiptirler. Serbest form yüzeylerin işlenmesi için en uygun, ancak diğer konfigürasyonlara göre en az rijitliğe sahip tezgahlardır.

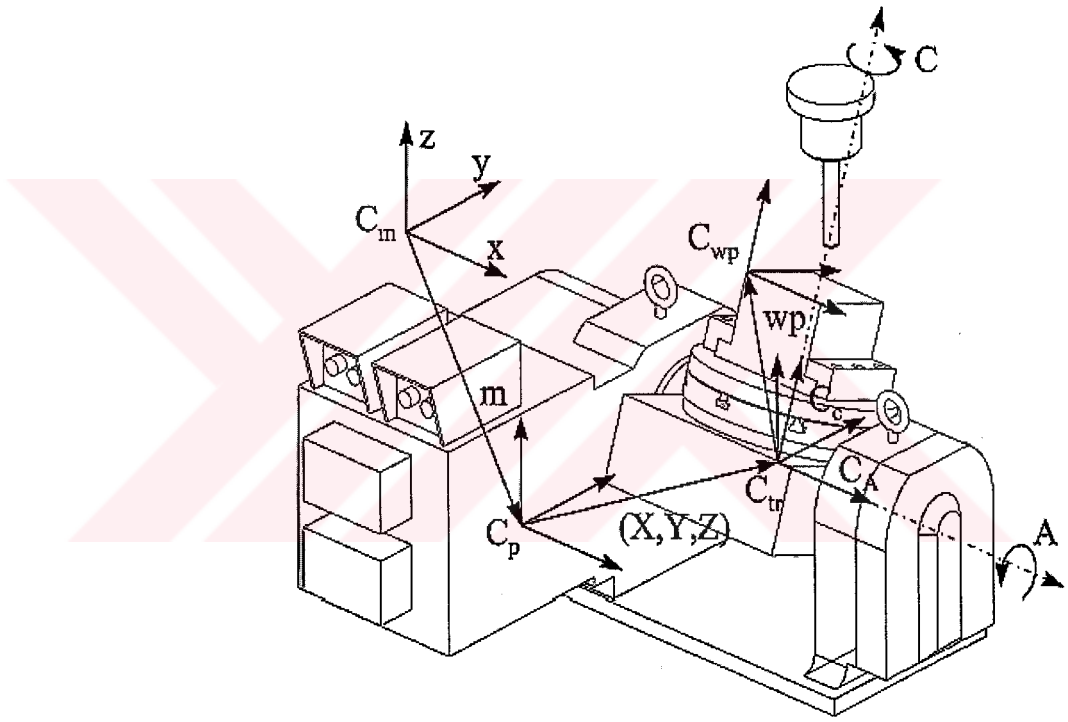
Eğilmeli-döner tabla tipindeki tezgahlar 5 eksen yanal işlemler için çok uygundur ve diğer konfigürasyonlara göre daha sağlam bir yapıya sahiptir. Bununla birlikte, ayarlama hatalarına daha elverişlidir ve geniş bir çalışma alanına sahip değildir.

Bir CNC tezgahı programlanırken, takımın iş parçasına bağlı tasarlanan konum ve oryantasyonu sağlamak için, her bir mafsallık hareketinin belirli bir sıra ile tanımlanması gerekir. Bu durum, bir ters kinematik problemini işaret etmektedir. Programlayıcı ayrıca, verilen bir dizi mafsallık komutuna göre takım yolunun doğruluğunu kontrol etmek için, sonuçtaki takım konumunu ve oryantasyonu bilmek zorundadır. Bu durum ise düz kinematik problemini işaret etmektedir.

Takip eden kısımda en çok karşılaşılan eğilmeli-döner tablalı ve bilek tipindeki 5 eksen freze tezgahları, düz ve ters kinematik açısından incelenecektir.

3.4.1. Eğilmeli-Döner Tabla Tipinde (RRTTT) Freze Tezgahları İçin Düz Kinematik

Tilt-rotary tipindeki 5 eksen bir tezgahı modellerken, Şekil 3.3'te gösterilen, tezgah koordinat sistemleri düşünülür.



Şekil 3.3. 5 Eksen Tezgahtaki Koordinat Sistemleri

Şekil 3.3'te makina koordinat sistemi C_m , CNC tezgahında çoğu pozitif konumda olacak şekilde çalışma hacmi içinde sabitlenen, değiştirilemez bir noktadır. Tezgaha gönderilen tüm komutlar makina koordinat sistemi açısından değerlendirilir. Tüm diğer koordinat sistemleri tezgah kullanıcılarına kullanım kolaylığını sağlamak içindir. Şekil 3.3'te, program koordinat sistemi C_p ile makina koordinat sistemi arasına m ile ifade edilen vektör yerleştirilmiştir. Bu vektörün, eğilmeli-döner tabla dönme merkezinde takımla uyumlu şekilde yerleştirilmiş olması esastır. Vektör

taşındıktan sonra, programlanan koordinat sistemine göre kontrolöre (0, 0, 0) pozisyonuna hareketi gerektiren bir komut verildiğinde, takım ucu dönme merkezinde olacak şekilde konumlanacaktır. Eğilmeli-döner tabla koordinat sistemleri, C_{tr} ve dönel koordinat sistemleri C_A ve C_C eğilmeli-döner tablanın merkezine bitişiktir.

Dönel eksenlerin başlangıç konumuna bağlı olarak, bir eğilmeli-döner tablanın farklı birçok şekilde düzenlenebileceğine dikkat edilmelidir. En temel konfigürasyonlar, tablanın başlangıçta yatay ve dikey olduğudur.

İş parçası koordinat sistemi C_{wp} , eğilmeli-döner tabla ile hareket eder. İş parçası öteleme vektörü w_p , eğilmeli-döner tabla koordinat sistemine göre iş parçası koordinat sisteminin konumunu verir.

Bir takım yolu kesme dosyası (CLDATA) postprosesörle işlendiğinde, postprosesör iş parçası öteleme vektörünü (W_p) G-kodlarına çevirmek için kullanmaktadır. Her bir G konumlanma kodu X, Y, Z, A ve C olmak üzere takım pozisyonunu program koordinat sistemine göre tanımlayan elemanlardan oluşur.

Eğilmeli-döner tabla koordinat sistemi, program koordinat sistemine göre, X, Y, Z komutlarıyla transfer edilir. İş parçası koordinat sistemi ise, eğilmeli-döner tabla koordinat sistemi içinde A ve C komutları vasıtasıyla X ve Z eksen etrafında döndürülebilir. CNC kontrolör, program koordinat sistemine göre verilen komutları makina koordinat sistemine m vektörü vasıtasıyla çevirir.

CNC tezgahın kinematik modelinin belirlenmesinde, koordinat sistemleri arasındaki bağlantıyı kurmak için homojen transformasyon matrisleri kullanılır.

$$\begin{bmatrix} P_x^{tr} \\ P_y^{tr} \\ P_z^{tr} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(A) & -\sin(A) & 0 \\ 0 & \sin(A) & \cos(A) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(C) & -\sin(C) & 0 & 0 \\ \sin(C) & \cos(C) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & wp_y \\ 0 & 1 & 0 & wp_y \\ 0 & 0 & 1 & wp_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x^{wp} \\ p_y^{wp} \\ p_z^{wp} \\ 1 \end{bmatrix} \quad (3.1)$$

Eğilmeli-döner tabla tipindeki tezgah örneği için, iş parçası koordinat sistemi C_{wp} içindeki, p^{wp} noktasının pozisyonunun, eğilmeli-döner tabla koordinat sistemine göre (C_{tr}) Denklem 3.1’de verildiği gibidir.

Daha sonra eğilmeli-döner tabla koordinat sistemi C_{tr} içinde bulunan p^{tr} noktası, program koordinat sistemine C_p Denklem 3.2 vasıtasıyla transfer edilir.

$$\begin{bmatrix} p_x^p \\ p_y^p \\ p_z^p \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 1 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x^{tr} \\ p_y^{tr} \\ p_z^{tr} \\ 1 \end{bmatrix} \quad (3.2)$$

Son olarak, program koordinat sistemi (C_p) içindeki nokta (P_p), makina koordinat C_m sistemine makina öteleme vektörü m yardımıyla transfer edilir (Denklem 3.3);

$$\begin{bmatrix} p_x^m \\ p_y^m \\ p_z^m \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & m_x \\ 0 & 1 & 0 & m_y \\ 0 & 0 & 1 & m_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x^{tr} \\ p_y^{tr} \\ p_z^{tr} \\ 1 \end{bmatrix} \quad (3.3)$$

Denklem 3.3’te bulunan P_m , makina koordinat sisteminde bulunan noktadır. Bütün bu transformasyon matrislerinin birleştirilmesiyle, P_{wp} ile gösterilen ve iş parçası koordinat sisteminde bulunan bir nokta, makina koordinat sistemine Denklem 3.4’te verilen ifade ile transfer edilir.

$$\begin{bmatrix} p_x^m \\ p_y^m \\ p_z^m \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(c) & -\sin(C) & 0 & x+m_x \\ \cos(A)\sin(C) & \cos(A)\cos(C) & -\sin(A) & y+m_y \\ \sin(A)\sin(C) & \sin(A)\cos(C) & \cos(A) & z+m_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x^{wp} \\ p_y^{wp} \\ p_z^{wp} \\ 1 \end{bmatrix} \quad (3.4)$$

Bu transformasyon iş parçası koordinat sisteminde bulunan bir nokta ile makina koordinat sistemi arasındaki ilişkiyi belirler. Bu bağıntının farklı tezgahlar için bir yaklaşım olduğu düşünülürse, belirli bir tezgahta doğrulanması için talaş kaldırma işlemi gerekecektir. Postprosesör için düşündüğümüzde, iş parçası koordinat sistemi içinde takımın doğru yerde konumlanması için mafsallı pozisyonlarının tayininde bu bağıntının tersi yani ters kinematik ifadesi gerekir.

3.4.2. Eğilmeli-Döner Tabla Tipinde Bir Tezgahın Ters Kinematiği

5 eksen işlemede kullanılan takım yolu, iş parçası koordinat sistemi içinde bir dizi takım konumlarından, ve bunlara karşılık gelen bir dizi oryantasyon vektörlerinden (*taxis*) oluşmaktadır. Postprosesör bu bilgileri, tezgah kesicisinin iş parçası koordinat sistemine bağlı olarak doğru yönde ve konumda olması için, açısallı (A ve C) ve doğrusal (X, Y, Z) elemanlara çevirmelidir. Tilt-Rotary tip tezgahlarda kesici oryantasyonu program koordinat sistemi içinde Z eksenini boyunca sabitlendiğinden, doğru oryantasyon ancak takım yönü Z eksenine hizalanana kadar iş parçasının A ve C eksenleri etrafında döndürülmesiyle elde edilir. Başka bir deyişle takım eksen vektörü, *taxis*, eğilmeli-döner koordinat sistemi içinde [0, 0, 1] ile ifade edilebilir vektördür. Bu şekilde, rotasyon açıları A ve C Denklem 3.5'te verilen transformasyon matris ifadesindeki A ve C değerleri çözülerek bulunabilir .

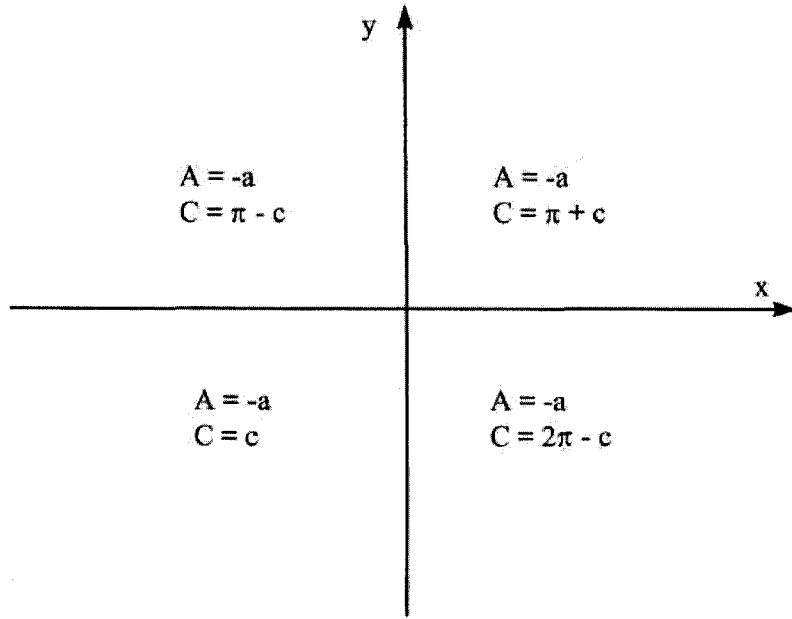
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(C) & -\sin(C) & 0 & 0 \\ \cos(A)\sin(C) & \cos(A)\cos(C) & -\sin(A) & 0 \\ \sin(A)\sin(C) & \sin(A)\cos(C) & \cos(A) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \textit{taxis}_x \\ \textit{taxis}_y \\ \textit{taxis}_z \\ 1 \end{bmatrix} \quad (3.5)$$

Bununla birlikte, bu eşitlikler transandantal denklemlerdir ve bunları kullanarak sonuçta A ve C rotasyon değerlerinin doğru işaretlerini tayin etmek zordur. Denklem 3.6'da verilen matrisin ilk satırı, C dönmesi hesabında kullanılmaya elverişlidir.

$$C = \pm \tan^{-1} \left(\frac{taxis_x}{taxis_y} \right) \quad (3.6)$$

Denklem 3.6'da verilen eşitliğe dikkat edildiğinde, C değeri için doğru olan işaretin tayin edilmesi gerektiği görülebilir. Bundan başka, bir 5 eksen tezgah A rotasyonundaki takım konum ve yön hareketini, negatif ve pozitif olmak üzere iki farklı şekilde sağlayabilir. Bir örnek vermek gerekirse, bir takım eksen yön vektörü, taxis, [0.577, 0.577, 0.577] ile belirlendiğinde, takımın bu vektörle hizalanması için A ve C rotasyonlarının sırasıyla (45° , 54.731°) veya (-135° , -54.731°) gibi farklı iki değerle sağlanabileceği doğrudur. Bu nedenle, öncelikle rotasyonların büyüklüklerinin hesaplanması daha iyidir. Daha sonra, takım yön vektörü dikkate alınarak, koordinat sistemindeki XY düzleminde, her bir çeyrek (Şekil 3.4) baz alınarak rotasyonların doğru işaretleri bulunabilir. A rotasyonunun her zaman negatif ve takım yön vektörünün her zaman yukarıyı gösterecek şekilde olduğu farz edildiğinde, aşağıdaki ifade A ve C değerlerinin doğru işaretinin tayin edilmesini sağlar. Bu yaklaşımla, takım yön vektörü ile pozitif z eksen arasındaki açı; denklem 3.7 yardımıyla hesaplanabilir [Warkentin et al, 2001].

$$A = \cos^{-1} \left(\frac{\sqrt{taxis_x^2 + taxis_y^2}}{\sqrt{taxis_x^2 + taxis_y^2 + taxis_z^2}} \right) \quad (3.7)$$



Şekil 3.4. Eğilmeli-Döner Tabla Tip Tezgah İçin Doğru Ave C Rotasyon Seçimi İçin Yaklaşım Çeyreği

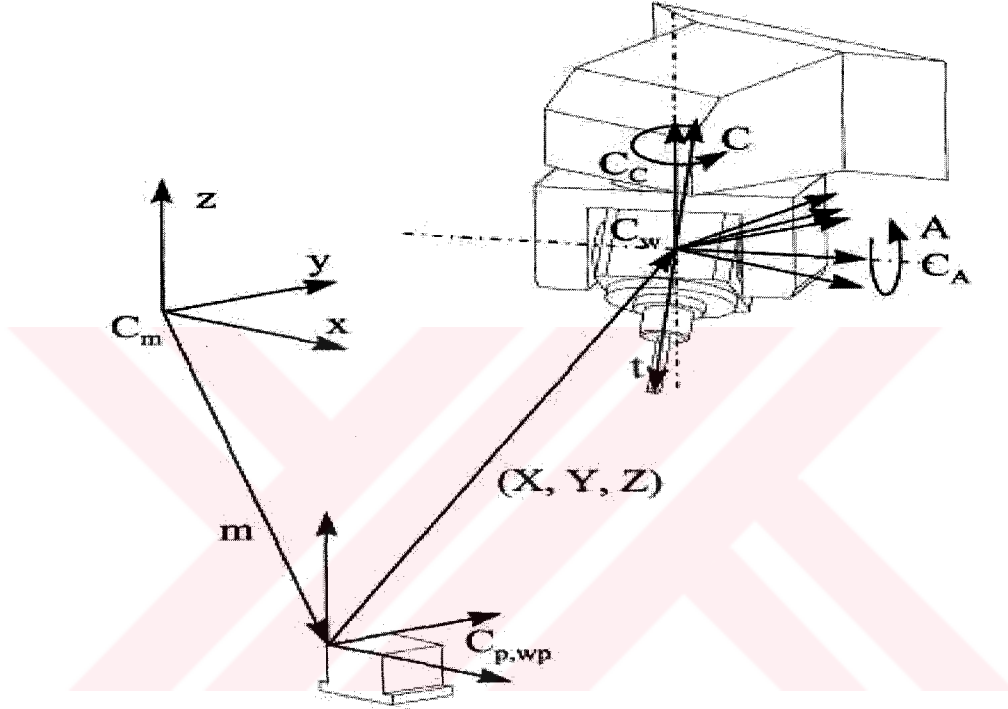
Takım yön vektörü ile y eksenini arasındaki açı ise; Denklem 3.8’de verilen eşitlikle hesaplanabilir [Warkentin et al, 2001].

$$c = \cos^{-1} \left(\frac{|taxis_y|}{\sqrt{taxis_x^2 + taxis_y^2}} \right) \quad (3.8)$$

Bu noktadan sonra iş parçası koordinat sistemine göre her bir çeyrek yardımıyla takım yön vektörü belirlenebilir.

3.4.3. Bilek Tipi (TTTRR) Freze Tezgahının Düz Kinematığı

Bilek tipi (TTTRR) 5 eksen tezgahın kinematığını modellemek daha kolaydır çünkü, iş parçası koordinat sistemi dönmemektedir. Şekil 3.5’de bilek tipi 5 eksen freze tezgahı modellenirken kullanılan koordinat sistemi gösterilmektedir.



Şekil 3.5. 5 Eksen Bilek Tip Freze Tezgahının Kinematığı

Her zamanki gibi, makina koordinat sistemi C_m , çalışma hacmi içindeki bir pozitif noktada konumlandırılmıştır. Program koordinat sistemi C_p , m vektörüyle, bileğin dönme merkezinin iş parçası koordinat sistemi ile başlangıçta aynı olacak şekilde konumlandırılmıştır. Bilek koordinat sistemleri C_w ve rotasyon koordinat sistemleri C_A ve C_C , bileğin dönme merkezi ile bitişiktir. Takım öteleme vektörü t , bilek koordinat sistemine göre takım ucunun konumunu vermektedir.

Takım yolu kesme verileri öncelikle takım uzunluğunun bilinmesiyle G kodlarına çevrilebilir. Tabla, program koordinat sistemine göre X, Y, Z pozisyon komutlarıyla konumlanırken, takım ise bilek etrafında A ve C komutları kullanılarak döndürülür. Bu komutlar makina koordinat sistemine kontrolör tarafından çevrilecektir. Vektör cebri kullanılarak, bilek koordinat sistemi C_w içindeki takım ucu konumu t Denklem 3.9 vasıtasıyla hesaplanır.

$$\begin{bmatrix} t_x^w \\ t_y^w \\ t_z^w \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(C) & -\sin(C) & 0 & 0 \\ \sin(C) & \cos(C) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(A) & -\sin(A) & 0 \\ 0 & \sin(A) & \cos(A) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \\ 1 \end{bmatrix} \quad (3.9)$$

Benzer olarak, bilek koordinat sistemi (C_w) içindeki takım ucu, t^w , program koordinat sistemine (C_p), Denklem 3.10 vasıtasıyla çevrilir;

$$\begin{bmatrix} t_x^p \\ t_y^p \\ t_z^p \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x^w \\ t_y^w \\ t_z^w \\ 1 \end{bmatrix} \quad (3.10)$$

Son olarak, program koordinat sistemi (C_p) içindeki takım uç pozisyonu, makina koordinat sistemine (C_m) Denklem 3.11 yardımıyla transfer edilir.

$$\begin{bmatrix} t_x^m \\ t_y^m \\ t_z^m \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & m_x \\ 0 & 1 & 0 & m_y \\ 0 & 0 & 1 & m_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x^p \\ t_y^p \\ t_z^p \\ 1 \end{bmatrix} \quad (3.11)$$

Tüm transformasyon matrislerinin birleştirilmesiyle, makina koordinat sistemi içindeki takım uç konumlanması denklem 3.12’de verildiği gibidir.

$$\begin{bmatrix} t_x^m \\ t_y^m \\ t_z^m \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(C) & -\sin(C)\cos(A) & \sin(C)\sin(A) & X + m_x \\ \sin(C) & \cos(C)\cos(A) & -\cos(C)\sin(A) & Y + m_y \\ 0 & \sin(A) & \cos(A) & Z + m_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \\ 1 \end{bmatrix} \quad (3.12)$$

Bu transformasyon takım ucunun makina koordinat sistemindeki yerini tespit etmektedir.

3.4.4. Bilek Tipi Freze Tezgahının Ters Kinematiki

Eğilmeli-döner tablalı tip tezgahta olduğu gibi, genel takım yolları postprosesörde işlenerek uygun mafsallara çevrilmelidir. Tekrar etmek gerekirse, tpos vektörü takım ucu pozisyonunu verirken, taxis vektörü takım yönünü göstermektedir. Bilek tipi bir tezgah için doğru yön, takımın, takım yön vektörüyle hizalanana kadar A ve C eksenleri etrafında döndürülmesiyle elde edilir. Bu değerler, A ve C için, Denklem 3.13’te verilen eşitliğin çözülmesiyle elde edilir.

$$\begin{bmatrix} taxis_i \\ taxis_j \\ taxis_k \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(C) & -\sin(C)\cos(A) & \sin(C)\sin(A) & 0 \\ \sin(C) & \cos(C)\cos(A) & -\cos(C)\sin(A) & 0 \\ 0 & \sin(A) & \cos(A) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{t}_x \\ \hat{t}_y \\ \hat{t}_z \\ 1 \end{bmatrix} \quad (3.13)$$

Denklem 3.13’te, takım ucunun başlangıç noktasında normalleştirilmesi $\hat{t}$ ile ifade edilmiştir. Bununla birlikte eşitliklerin transandantal olması ve olası iki yönde kesici takımın konumlanabilmesi nedeniyle, bu yöntemde birtakım zorluklar ortaya çıkmaktadır. Böyle bir durum için eğilmeli-döner tabla tipi tezgah için kullanılan yaklaşımın aynısı kullanılabilir.

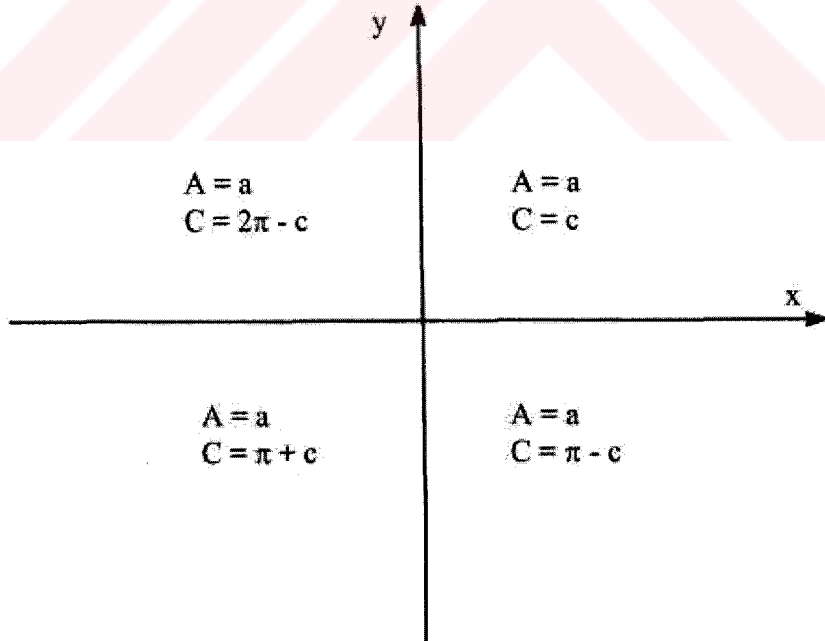
İş parçası koordinat sistemi içinde takım ile -Z eksenini arasındaki açı Denklem 3.14 yardımıyla hesaplanabilir.

$$A = \cos^{-1} \left(\frac{\sqrt{taxis_x^2 + taxis_y^2}}{\sqrt{taxis_x^2 + taxis_y^2 + taxis_z^2}} \right) \quad (3.14)$$

Takım yön vektörü ile y eksenini arasındaki açı ise Denklem 3.15 yardımıyla hesaplanabilir.

$$C = \cos^{-1} \left(\frac{|taxis_y|}{\sqrt{taxis_x^2 + taxis_y^2}} \right) \quad (3.15)$$

Açılar hesaplandıktan sonra aşağıdaki Şekil 3.6'de verilen bölgelere göre, iş parçası koordinat sistemi içinde doğru takım konumları, dolayısıyla doğru A ve C rotasyonları hesaplanabilir.



Şekil 3.6. Bilek Tip Tezgah İçin Doğru A ve C Rotasyon Seçimi İçin Yaklaşım Çeyreği

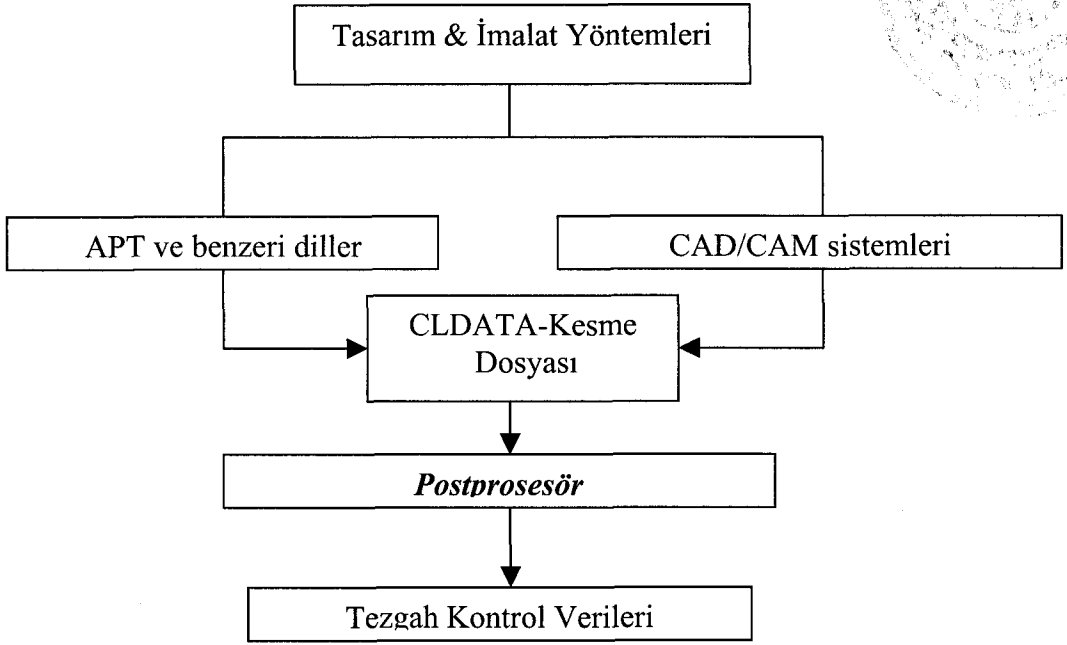
Buraya kadar iki tip tezgah için genel kinematik özellikler çıkarılmış ve koordinat transformasyonları formülize edilmiştir. Aynı tezgah tipinde bile olsa, denklemlerden farklı sonuçlar elde edilebilir. Bunun nedeni, denklemlerin herhangi bir teste tabii tutulmaması ve her tezgahın farklı iş alanı hacmi, farklı açısal hareket kapasitesi, tezgah montajında oluşan farklı kaçıklık mesafe değerlerine sahip olmasından kaynaklanır. Tezin uygulama kısmında eğilmeli-döner tip bir tezgah için postprosesör geliştirilirken, özellikle doğru rotasyon analizlerinde verilen ifadelere göre birtakım uyumsuzluklar görülmüş ve bu durum tezgahta yapılan birkaç denemeyle düzeltilmiştir.

3.5. Takım Kesme Dosyası ve Postprosesör Kavramı

Postprosesör ihtiyacının çıkış noktası, tezgahların farklı kontrolör kullanmalarından doğar. Bazen aynı yapım ve aynı model kontrolörler bile müşteri istekleri doğrultusunda farklı opsiyonlarla donatılabilir. Bu durum kullanıcıların, aynı kapasitede fakat farklı yapılarıdaki tezgahlarda, aynı parça için, aynı NC programının kullanılmasını engellemektedir. Bu yüzden oluşturulan kesme dosyası bir şekilde, belirli bir tezgahın anlayabileceği kontrol verilerine çevrilmelidir.

Buradan yola çıkarak postprosesör programın tanımı yapılacak olursa, kesme dosyasını (CLDATA), belirli bir tezgah için istenen NC tezgah kontrol verilerine çeviren bir programdır.

Şekil 3.7’de gösterildiği gibi hangi yöntemle parça imal etmek istersek isteyelim, birkaç istisna yöntem haricinde, parça programı oluşturmak için postprosesöre ihtiyaç duyulmaktadır.



Şekil 3.7. Tasarım Ve İmalat Sürecinde Postprosesörün Yeri

Kullanıcı, tezgah üreticileri ve CAD/CAM yazılım firmaları arasında bağlantı kurmak maksadıyla CLDATA, ISO (International Organization For Standardization) tarafından belirlenen iki standart ISO 3592-1978 ve ISO 4343-1978 ile ilk kez tanımlanmıştır [Edwards ve Sharp, 1991].

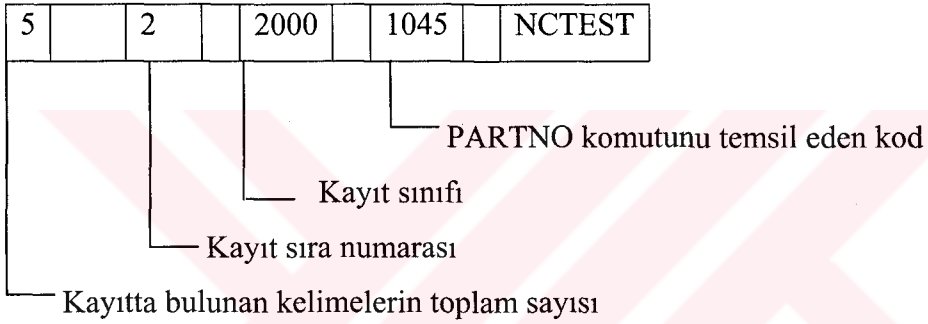
Bir takım kesme dosyası, bir dizi komut içeren blok halindeki mantıksal kayıtlardan meydana gelir. ISO 3592-1978 standardına uygun olarak bu komutlar 18 ayrı sınıfta kategorize edilmişleridir. Ek 4'te detaylıca verilen bu sınıflardan takım yolu tayininde kullanılan, 1000, 4000, 7000, 8000 ve 12000 sınıfı kategorileri, artık günümüzde kullanılmamakta ve dolayısıyla kesme dosyası içinde yer almamaktadır.

ISO standartları, bir mantıksal kaydın fiziksel yapısını belirlememiştir [Chang ve Melkanoff, 1989]. Bu yüzden farklı yazılım firmaları farklı tip kayıt formunda kesme dosyası oluşturur. Günümüzde çeşitli CAM yazılımlarının kesme dosyası örnekleri Ek 5 ile Ek 10 arasında detaylı olarak verilmiştir.

Günümüz CAM sistemleri genelde kesme dosyası kayıt yapılarını, 2000 veya 20000 sınıf içeriğine göre sunmaktadır. 2000 sınıfı kayıtlarındaki veriler, tam sayı ve katar karışımı ile gerçel sayılardan meydana gelir. 20000 sınıfındaki kayıtlarda ise veriler yalın bir ifade ile, karakter katarlarından meydana gelir. Örneğin 20000 sınıfında;

PARTNO NCTEST

Şeklindeki bir kayıt içinde verilen komutun 2000 sınıfındaki karşılığı Şekil 3.8'deki gibidir.



Şekil 3.8. 2000 Sınıfı Kayıt Yapısı

Kayıtlardaki komutlar ise asıl (majör) ve tali (minör) olmak üzere ikiye ayrılırlar. Asıl komutlar genellikle 2000 sınıfı kayıtlar içinde tanımlıdır. Bir asıl kelime postprosesör komutları içinde ilk parametredir. Örneğin;

SPINDL/350, RPM,CLW

Komut satırında, SPINDL sözcüğü asıl kelimedir. Tali kelimeler ise asıl kelimeleri takip eden herhangi sözcüklerdir. Yukarda verilen örnekte, RPM ve CLW tali sözcüklerdir.

Kesme dosyaları APT dilindeki kullanımı ile aynı olan bir takım özel karakterler kullanırlar. Bunlar ve kullanım amaçları Tablo 3.1’de verilmiştir.

Tablo 3.1. Kesme Dosyasında Bulunan Özel Karakterler.

Sembol	Kullanımı
/	Asıl komut sözcüğü ile diğer parametreleri ayırır.
,	Parametreleri ayırır.
[]	Opsiyonel parametreleri içine alır.
()	Parametreleri ayırırken öncelikleri değiştirir.
\$\$	Yorum yapılan bir kayıt olduğunu bildirir.
\$	Sürekliliği ifade eden karakterdir.

Farklı CAD/CAM sistemlerinin birbirinden bağımsız kesme dosyaları oluşturmaları buna ilaveten, her tezgahın neredeyse kendine göre bir programlama dilinin olması, postprosesör yazılımına büyük önem kazandırmıştır. Eğer imal etmek için parça tasarımı yapılıyorsa uygun tezgah için uygun postprosesör olmadığında, parçanın üretimi mümkün olmayacaktır.

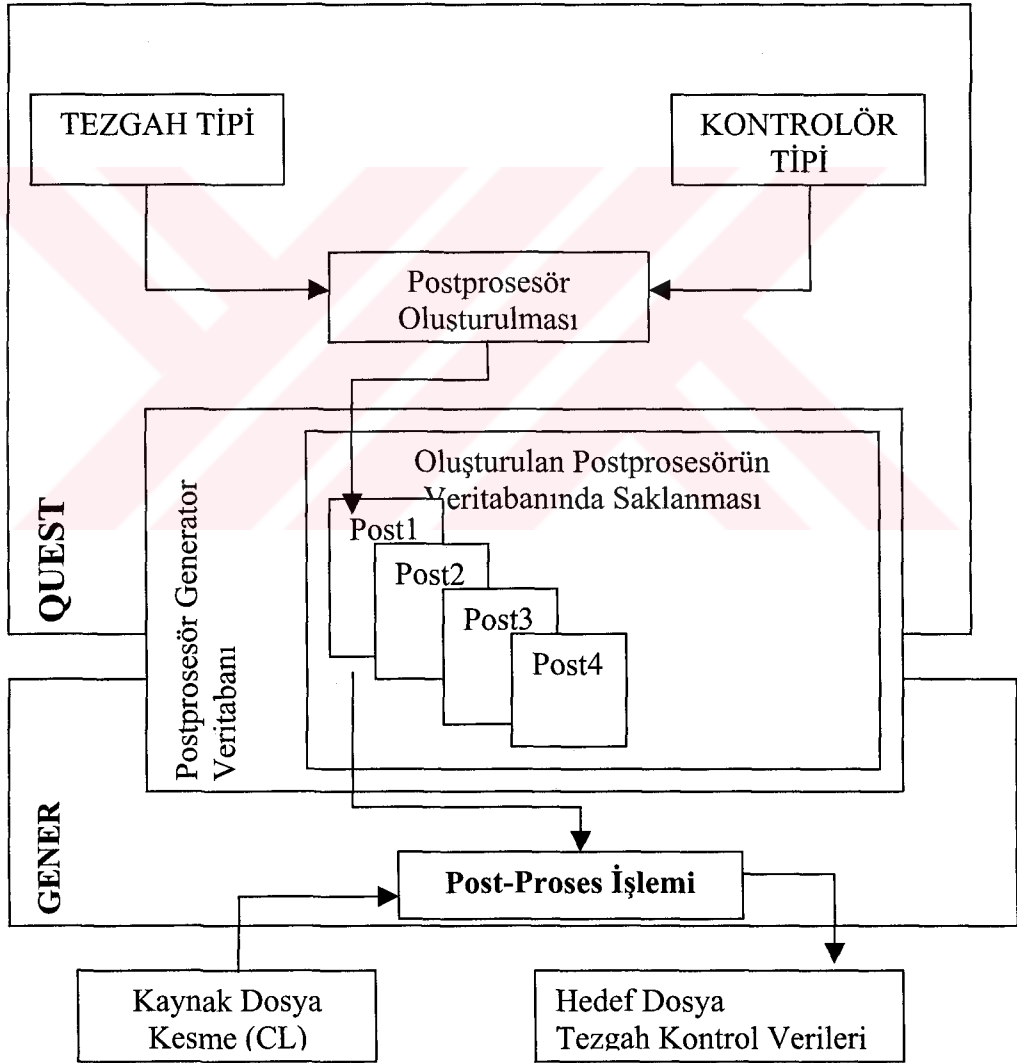
Çok çeşitli CAD/CAM sistemlerinin bulunması, takım kesme dosyalarının farklılaşması açısından ortaya bir problem olarak çıktığında, Postprosesör Generator yazılımları çözüm olarak sunulmaktadır.

Postprosesör Generator programları, son kullanıcının C, Fortran, Pascal gibi yapısal programa dilleri bilmesini gerektirmeden belirli herhangi bir tezgah ve CAM sistemi için postprosesör oluşturmalarını sağlayan genel yazılımlardır. Bir işletmede birkaç çeşit farklı tezgahın ve CAM sisteminin olduğu düşünüldüğünde bu yazılımın sağladığı avantajlar büyüktür. Kullanım itibariyle yazılım iki sınıfta incelenebilir;

- Belirli bir CAD/CAM sistemiyle bütünleşik Postprosesör Generator yazılımları; günümüzde çoğu CAD/CAM sistemi kendisine entegre bu tür bir yazılımı barındırır. Dolayısıyla sistemin oluşturduğu takım kesme dosyası komutlarının tanıtılması gibi bir problem yoktur.

- Üçüncü parti çözümü Postprosesör Generator yazılımları; tüm CAD/CAM sistemlerine cevap vermek amacını taşırlar. Çok daha pahalı, kullanımı daha karmaşık yazılımlardır. CAD/CAM sistemlerinin çok hızlı bir şekilde değişmesi ve yenilenmesi nedeniyle, sürekli olarak güncellenmesi gerekir. Bu maddi açıdan ek bir yük getirir.

Bir Postprosesör Generator yazılımı, Şekil 3.9'da gösterildiği gibi çalışmaktadır.



Şekil 3.9. Postprosesör Generator Yazılımlarının Genel Yapısı



Postprosesör Generator yazılımlar “Quest” ve “Gener” olarak adlandırılan iki kısımdan meydana gelir.

Quest, veritabanında bir yada daha fazla NC tezgah ve bunların kontrol bilgilerinin girilmesini ve saklanmasını sağlayan enteraktif programdır. Bu kısımda kullanıcıya bir dizi sorular sorulur. Tezgah tipine göre değişen birçok sorunun, adım adım cevaplanmasıyla tezgah, Generator yazılımına tanıtılmış olur ve veritabanına kaydedilir.

Yazılımın Gener kısmı ise, bu oluşturulan veritabanını kullanarak, belirli bir tezgah için kesme dosyasını tezgah kontrol verilerine çevirir.

3.5.1. Özel Postprosesörler

Postprosesör generator yazılımları fazla sayıda ve çeşitli tipte tezgahlara postprosesör oluşturmaya yarayan sistemlerdir ve bu yüzden pahalıdır. Birkaç adet ve tipte tezgah kullanan işletmeler için diğer bir alternatif, tezgahlara ve CAM sistemlerine uygun özelleştirilmiş postprosesör yazılımı kullanmaktır. Esasen kabaca bakıldığında postprosesörün yaptığı iş, okunabilen bir yazı dosyasını, yine okunabilen bir başka yazı dosyasına çevirmekten başka bir şey değildir. Bu mantıkla yaklaşıldığında postprosesörler, yapısal programlama dilleri kullanılarak oluşturulabilirler. Postprosesör Generator yazılımıyla hazırlanan bir postprosesör ile programlama dilleriyle tezgaha özel hazırlanan postprosesörleri mukayese edildiğinde şu sonuçlar çıkar;

- Generator yazılımlarının maliyeti yüksektir. Özellikle 5 eksen destekleyen bir Generator yazılımının fiyatı, 5 ila 25 bin dolar arasında değişmektedir. Konu 5 eksen olduğunda bu rakamlara göre özel postprosesörler fiyat açısından çok daha avantajlıdır.



- Generator yazılımları son kullanıcının yapısal programlama dillerinden herhangi birini bilmesini gerektirmez. Özellikle 3 ve 5 eksenli tezgah konfigürasyonları için postprosesör hazırlamada büyük kolaylık sağlar.

- Generator yazılımları yeteri kadar esnek, basitçe geliştirilebilen yazılımlar değildir. Bunun nedeni yazılımın yapısı gereği, belirli sayıda soruya cevap beklemesinden dolayıdır. Oysa CAM teknolojisi sürekli gelişmekte olduğundan, postprosesörün etkinliğini artırmak için esnek bir Generator yazılım gereklidir. Bu mecburiyet son kullanıcının, Generator içinde yine yazılıma has makro dilleri kullanması zorunluluğunu getirir. Makro dillerini öğrenmek ise yine yazılımla ilgili iyi seviyede eğitim almayı gerektirir.

- Generator yazılımları, reklamı yapıldığı gibi sadece soruların cevaplanıp buna göre postprosesörün hazırlanabileceği yazılımlar değildir. Program numarasının girilmesi, X, Y, Z ötelemelerinin sorulup cevaplarının işleme tabi tutulması, takımın ne zaman değiştirileceği, universal bir tezgah için kafanın hangi durumlarda çalışma düzlemini değiştireceği ve kesme dosyasında bazı anlaşılamayan komutların yazılıma tanıtılması gibi, postprosesörün optimizasyonu ile ilgili çok önemli konularda mutlaka makro dili kullanılmalıdır. Örneğin, sadece sorulara cevap verilerek tanıtılan 5 eksenli bir universal tezgah için, tezgah mili ancak başlangıç düzleminde çalışacaktır.

- Özel bir postprosesör, tasarlama hayal gücü ve programlama yetenekleriyle orantılı şekilde esnektir çünkü, bir yapısal programlama dili, Generator yazılımına nazaran baştan aşağıya bir dizi makro ile oluşturulur. Belli standartları destekleyen tezgah kontrolörü haricinde herhangi bir nümerik kontrollü tezgahın CAM sistemine adaptasyonu ancak bu tip bir yöntemle mümkündür. Örneğin, herhangi bir işte kullanılan ve Acrobasic dili ile hareket ettirilebilen 3 eksenli bir robot için Generator yazılımları kullanılamaz. Aynı şekilde günümüz CAM sistemlerinin desteklediği koordinat ölçme cihazları için takım yolunun çıkarılması ve neticesinde oluşan dosyanın tezgah diline çevrilmesi ancak özel postprosesör yardımıyla yapılmaktadır.

3.6. Özel Bir Postprosesörün Tasarımı

Uygulamada belirli bir tezgah ve belirli bir CAM sistemi için özel bir postprosesör geliştirilmiştir. Bundan sonraki konularda bu uygulamanın çeşitli safhaları izah edilecektir.

Daha öncede belirtildiği gibi özel bir postprosesör yapısal programlama dilleri kullanılarak çevrilir. Programlama açısından bakıldığında bir ASCII yazı dosyasının okutulup, başka bir ASCII yazı dosyasına çevrilmesi gerekir. Bu dosya okuma ve yazma işlemleri hemen hemen tüm programlama dilleriyle yapılabilir. Uygulama için C++ Builder 3.0 kullanılmıştır. Uygulamada, standart C kodları kullanılmış ve buna Builder' in verdiği arayüz tasarımındaki çeşitli rahatlıklar eklenmiştir. C dilinin en büyük avantajlarından biri platform bağımsız olmasıdır. Dolayısıyla standart C diliyle yazılan kodlar ufak tefek değişiklikler yapıldığında Unix veya Linux işletim sistemlerinde de kullanılabilir.

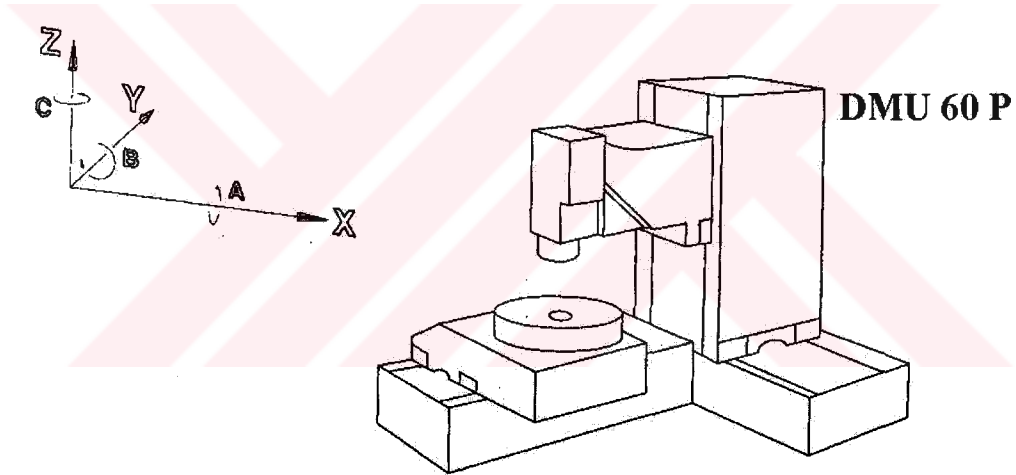
Bir programlama dili bilgisine hakim olma ihtiyacından başka belirli CAM sistemleri bilgisine de hakim olmak, özel postprosesör tasarımında değişmez temel ilkedir. Dolayısıyla bu bir aşama olarak düşünülmemiş ve bu ilkeleri açıklamaktan kaçınılmıştır. Bunun haricinde diğer aşamalar, tezgahın tanımlanması ve CAM sistemi kesme dosyası özelliklerinin bilinmesidir.

3.7. Tezgah Tanımlanma Safhası

Hangi yöntemle postprosesör hazırlanırsa hazırlansın, tezgah özelliklerinin iyi bilinmesi gerekir. Bu özellikler tezgah çalışma alanı, maksimum ilerleme hızları, fener mili dönme devri gibi fiziksel kapasite özellikleri ile kontrolör kumanda dili özellikleridir. Uygulamada, 5 eksen Deckel-Maho DMU 60P tezgah için bir postprosesör hazırlanmıştır. Bu veriler ışığında Şekil 3.10'da gösterildiği gibi söz konusu tezgah için temelde şu özellikler geçerlidir ;

- Tezgah 3 lineer eksen yanında 2 rotasyon eksenine sahip eğilmeli-döner tabla olarak adlandırılan tiptedir. A eksenini olarak adlandırılan birincil eksen, X eksenini etrafında her iki yönde de açısal hareket yapabilmektedir. C eksenini olarak atanan eksen ise tabla ya da iş parçası, Z eksenini etrafında her iki yönde de açısal hareket yapabilir.

- Tezgah koordinat sistemi sağ el prensibine uymaktadır (Şekil 3.10.).
- Tezgah başlangıç çalışma düzlemi XY düzlemidir , istenildiğinde çalışma düzlemi program içindeki bir komut vasıtasıyla ZX düzlemine çevrilebilir.



Şekil 3.10. Deckel-Maho DMU 60P İçin Koordinat Sistemi

Bu bilgilere ilaveten, uygulamada geçerli tezgahın ayrıntılı teknik bilgileri Ek 11’de verilmiştir.

Ayrıca, tezgahla ilgili bilinmesi gereken bir başka husus, tezgah kontrol komutlarıdır. Tezgah kontrolörü, Ek 1 ve Ek 2’de detaylı olarak verilen, söz konusu standart komutları büyük bir oranda desteklemektedir. Desteklenmeyen birtakım komutlardan ise, ileriki konularda yeri geldiğinde bahsedilmektedir.

İlgili tezgahın özellikleri tanındıktan sonra bir sonraki adım, tezgah kontrol verilerine (G kodlarına) çevrilecek olan dosyanın özelliklerinin belirlenmesidir. Çeşitli CAM yazılımları farklı yapılar da kesme dosyası oluşturduğundan, uygulamada, dünyanın en yaygın kullarımdaki yazılımlarından biri olan, ProEngineer yazılımı kesme dosyalarını baz almıştır. Bundan sonraki kısımda bu yazılımın kendine has takım kesme dosyası incelenecektir.

3.8. Kesme Dosyası Özellikleri

ProEngineer takım kesme dosyasında, her bir kayıt maksimum 80 karakter uzunluğundadır. Bu uzunluğun aşıldığı durumlarda kayıt sonuna “\$” işareti konur. Böylece ilgili kaydın bir sonraki kayıt ile birlikte ele alınması gerektiği belirtilir. Bir CAD/CAM yazılımı olarak ProEngineer, ISO standartlarında belirlenen komutlarla oluşturulan kayıtlarla bir kesme dosyası oluşturur. Kaynak dosya olarak kullanılacak kesme dosyasını birkaç kısımda incelenebilir.

3.8.1. Lineer Hareketler

Kesme dosyalarında doğrusal hareketler “GOTO” komutu ile tanımlanır. GOTO komutu kesme dosyalarının belkemiğini teşkil eder ve kesme dosyaları içinde en sık rastlanan komuttur. Sık sık karşılaşılan bu komut, takımın belirli bir noktada konumlanmasını sağlayan direktif ve koordinat bilgilerini içerir.

CNC tezgahlarında, tüm dairesel ve kapalı döngü hareketlerin, doğrusal olarak programlanabilirliği düşünülürse, tüm kesme dosyası sadece GOTO komutu kullanılarak oluşturulabilir. Ancak bu durum kesme ve makina kontrol veri dosyasının hacminin artmasına neden olur.

Doğrusal hareketler CNC tezgahlarında, noktadan noktaya ve kontur olmak üzere iki bölüme ayrılabilir. Noktadan noktaya konumlanma, tezgahta hızlı konumlanma olarak geçer ve G0 ile kodlanır. Kontur hareket ise, belirli bir

ilerlemeyle yapılan düzlemde eşzamanlı harekettir ve G1 ile kodlamaktadır. Kesme dosyasında bu hareketler için farklı tanımlamalar kullanılır. Örneğin hızlı konumlanmanın kesme dosyasındaki formu şu şekildedir;

RAPID

GOTO / x, y, z

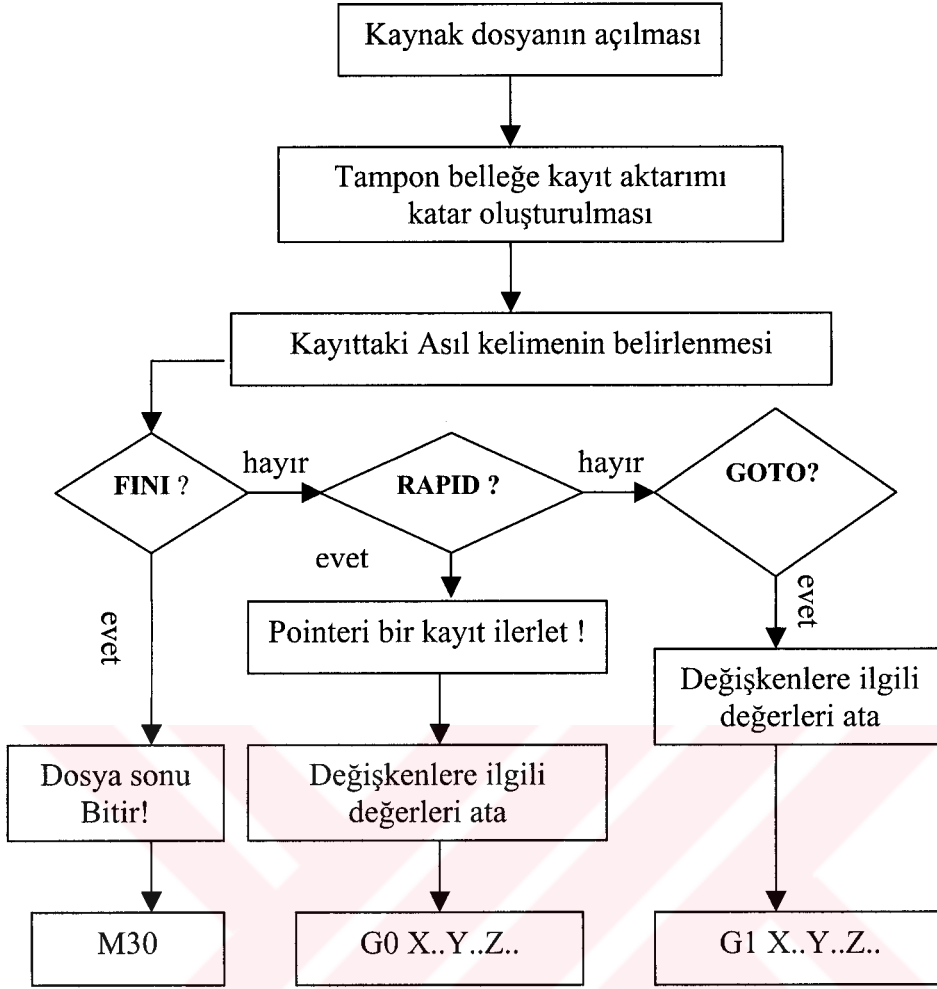
Lineer kontur hareketi ise tek başına bir GOTO komut satırından oluşmaktadır;

GOTO / x, y, z

Asıl komut olan “GOTO” komutu 5000 kayıt sınıfı içinde tanımlıdır. Burada x, y ve z simgeleri, kartezyen koordinat sisteminde takım konumunu belirleyen değerleridir.

GOTO komutu kesme dosyası içinde takım konumlanmasını tanımladığından tüm diğer komutlarda içindede kullanılır. Genel kullanıma haiz bu komut için alt rutin tanımlayıp program içinde bu alt rutini çağırmak avantaj sağlayacaktır. Uygulamada, doğrusal hareket için Şekil 3.11’de verilen algoritma kullanılmıştır.

Ek 10’da verilen takım kesme dosyası örneğinde, bu komutun geçtiği satırlar gösterilmektedir. GOTO komut satırındaki koordinat değerleri, varsayılan olarak noktadan sonra 10 rakamdan oluşan bir duyarlılığa sahiptir ve C dilinde bu değerler bir değişkene atanırken float veya double olarak tanımlanabilir. Bu değişkenler uygulama içinde double olarak tanımlanmıştır. C dilinde float değerler noktadan sonra 6 rakam duyarlılığına sahip olduğundan dolayı uygulama, özellikle 5 eksen kesme hareketlerinde yüksek hassasiyet sağlamak için değişkenleri, duyarlılığı daha fazla olan double ile tanımlamayı yeğlemiştir. Gerekli görüldüğü takdirde ProEngineer, bu değerlerin hassasiyetlerini istenilen şekilde ayarlamaya imkan tanır. Bunun için konfigürasyon dosyasında ilgili değişikliğin yapılması gerekmektedir.



Şekil 3.11. RAPID Ve GOTO Komutları İçin Kullanılan Algoritma

3.8.2. Dairesel Hareketler

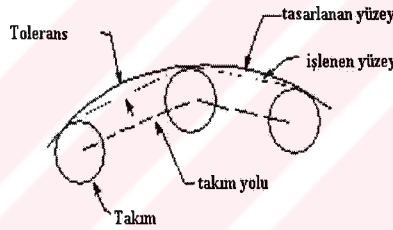
CNC tezgahları ve dolayısıyla CAM sistemleri, dairesel hareket olarak adlandırılan, bir diğer kontur hareket tarzını desteklerler. CAM sistemleri dairesel hareketler için CIRCLE komutu kullanırlar. Kesme dosyası içinde parçalı 5000 kayıt sınıfı veya parçasız 15000 sınıf kayıtlarıyla tanımlanırlar.

Çoğu CAM yazılımında olduğu gibi ProEngineer, dairesel takım hareketi için birkaç değişik formda çıktı vermektedir.

POINTS_ONLY kesme dosyası formu; dairesel enterpolasyonun sadece bir dizi noktalarla ifade edildiği kesme dosyası formudur. Bu kesme dosyası şekli dairesel enterpolasyon yapma kapasitesinden yoksun kontrolöre sahip tezgahlar için kullanılır.

Şekil 3.12’de gösterildiği gibi, bu tip kesme dosyalarında, takımın takip edeceği yaylar, belirlenen tolerans değerlerine uygun olacak şekilde, bir dizi düz çizgisel hareketin birleştirilmesi ile oluşturulur. Düz çizgisel hareketler doğrusal hareket komutu GOTO ile tanımlanacağından, tüm dairesel hareket bir dizi GOTO komutundan meydana gelecektir.

POINTS_ONLY formundaki bir kesme dosyası, Ek 12’de verilmiştir.



Şekil 3.12. Bir Yayın Tolerans Değeri İle Doğru Parçalarına Bölünmesi.

POINTS_&_ARC formu; artırımlı tip dairesel enterpolasyon yapan tezgahlar için geçerlidir. CIRCLE asıl komutunun bulunduğu kaydı takip eden kayıtlarda, tolerans değerine uygun olarak maksimum sayıda nokta tanımlı yapılır.

POINTS_&_ARC formundaki bir kesme dosyası örneği , Ek 13’ verilmiştir.

APT_FORMAT; postprosesörün dairesel hareket tanımlamasını APT formunda beklediği durumlarda kullanılır. ProEngineer yazılımı varsayılan dairesel enterpolasyon komutu APT formundan farklıdır ve CIRCLE komutu ile tanımlanır.

INDIRV / i, j, k

TLON

GOFWD / (CIRCLE / x, y, z, r), ON, (LINE / x1, y1, z1, x2, y2, z2)

Burada;

i, j, k — birim vektör

x, y, z — daire merkezi koordinatları

r — daire yarıçapı

x1, y1, z1 ve x2, y2, z2 — çıkış çizgilerini belirleyen iki nokta koordinatları.

APT_FORMAT kullanılarak oluşturulan bir dosya örneği Ek 14'te verilmiştir.

ARC_ONLY formu; tam dairesel enterpolasyon yapabilen tezgahlar için kullanılır. Bu formda, sadece CIRCLE komutunun bulunduğu satır ve bunu takip eden, postprosesör için gerekli minimum nokta kesme dosyasında yer alır. Bu noktaların sayısı NUMBER_OF_ARCS_PTS parametresi ile belirlenir.

Uygulamada, CAM sisteminde varsayılan ARC_ONLY formu kullanılmıştır. Uygulamada kullanılan tezgah, tam dairesel enterpolasyon yapamadığından, bir yayın eşit iki yay halinde tanımlanmasını sağlayan, NUMBER_OF_ARC_PTS parametresi, 2 olacak şekilde kullanılmıştır. CNC tezgahta dairesel enterpolasyon metodu ise yarıçap metodu olarak tayin etmiştir.

GOTO / x, y, z

CIRCLE / cx, cy, cz, i, j, k, r

GOTO / x, y, z

x, y, z

Burada;

cx, cy, cz — daire merkezinin koordinatları.

i, j, k — düzlem vektörü.

r — yarıçap (pozitif ve 0'dan farklı bir sayı olmalıdır).

CIRCLE komutundan önce gelen satırdaki GOTO komutu ile yayın başlangıç noktası belirlenir. Takip eden GOTO satırlarında yayın son noktası belirlenmektedir.

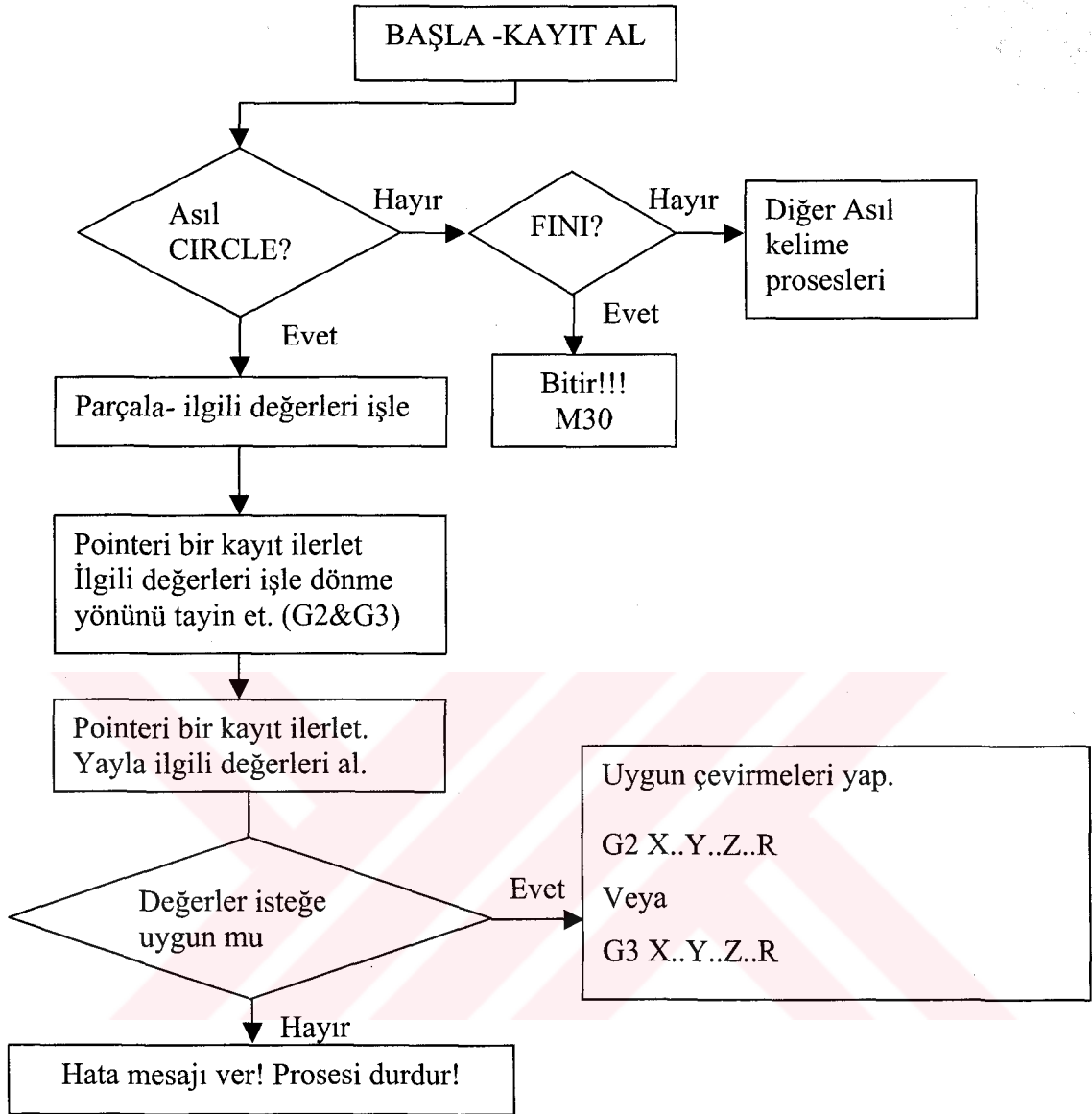
Dairesel hareketi incelendiğinde, başka bir faktör olarak takım hareket yönünün belirlenmesi çıkar. CNC tezgahlarında takım hareketi, saat yönünde (CLW) G2 veya saat yönünün tersi (CCLW) G3 kodları vasıtasıyla sağlanır. Kesme dosyasında ise hareket yönü, Tablo 3.2'te verildiği şekilde, sağ el kuralı yardımıyla düzlem vektörünün kullanması ile tayin edilir.

Tablo 3.2. Düzlemlere Göre Dairesel Hareket Yönü

i	j	k	Düzlem	Tezgah düzlemi	Yön	Tezgah yön kodu
0	0	1	XY	G17	CCLW	G3
0	0	-1	XY	G17	CLW	G2
0	1	0	ZX	G18	CCLW	G3
0	-1	0	ZX	G18	CLW	G2

ARC_ONLY formundaki bir kesme dosyası ve makina kontrol verisine çevrilmiş hali Ek 15'te verilmiştir.

Dairesel hareket için Şekil 3.13'te gösterilen algoritma kullanılmıştır.



Şekil 3.13. Dairesel Enterpolasyon İçin Kullanılan Algoritma

Dikkat edilirse algoritma içinde, üçüncü kayıt içeriğinin kontrolü söz konusudur. Bu sayede kullanıcının CAM yazılımı içinde NUMBER_OF_ARC_PTS parametresini 2 olarak ayarlayıp ayarlamadığı kontrol edilir.

Yukarıda bahsi geçen dairesel enterpolasyon tipleri sadece kesme dosyası formuyla ilgilidir. Eğrisel bir geometri boyunca gerçek takım hareketi her zaman verilen tolerans değerine göre oluşturulmaktadır. Yani hangi enterpolasyon şeklini kullanırsanız kullanın sonuçta aynı hassasiyete sahip kesme yapılacaktır, bununla

birlikte değişen tek şey kesme ve tezgah kontrol veri dosyalarının bellekte tuttuğu hacim olacaktır.

3.8.3 Kapalı Döngüler

Kesme dosyası içinde kullanılan CYCLE komutu standart delik delme, kılavuz salma, raybalama ve benzeri delme operasyonları için kullanılmaktadır. Bu operasyonlar uygulamada kullanılan tezgah kontrolörü tarafından G81-G89 arasındaki döngü komutları ile otomatik olarak oluşturulmaktadır.

CAM sistemlerindeki tüm kapalı döngüler, temelde benzer hareketleri yaparlar. Tezgah, takımı belirlenen bir başlangıç noktasına hızlıca konumlandırarak getirir, kesici takım bu noktadan sonra parça içinde belirli bir hızda ilerleyerek kesme yapar ve operasyon bittiğinde kesiciyi tekrar başlangıç noktasına çeker. Başlangıç konumlanması, derinlik ve bekleme zamanı tanımlamaları tüm kapalı döngüler için aynı şekilde belirlenir.

Döngülerin bir diğerinden farkı, takımın parça içinde ilerlerken ve geri çekilirken kullandıkları farklı yöntemlerden kaynaklanmaktadır.

Tablo 3.3. CAM Sistemleri Kapalı Döngü Tipleri

DRILL	Belirli hızda kesme, hızlı harekette geri çekilme
FACE	Belirli hızda kesme, bekleme, hızlı çekilme
TAP	Belirli hızda kesme, fener mili tersine hareket, belirli hızda geri çekilme
BORE	Belirli hızda kesme, çeşitli hareketler, hızlı çekilme
REAM	Belirli hızda kesme, opsiyonel bekleme, hızlı çekilme

NC tezgah açısından kapalı döngü tanımlanmasında ifadesinde iki yaygın yöntem kullanılır. Standart metot, döngü blok satırı içinde, döngü tanımlaması ve takım konumlanma bilgilerinin belirtildiği yöntemdir. Aktivatör metodunda ise, ilk

blok satırı, döngü tipi, aralık mesafesi, derinlik ve diğer döngü tanımlamada kullanılan parametrelerden oluşur. Bunu takip eden diğer blokta ise takım konum verileri bulunur (genellikle G79 kodu ile belirlenir). Uygulama da kullanılan tezgah aktivatör metodunu kullanmaktadır.

Cam yazılımlarında genel kapalı döngü tipleri ve hareket karakteristikleri şu şekildedir.

Kesme dosyasında DRILL tali komutu ile tanımlanan standart delme döngüsünde, kesici takım belirli bir ilerleme hızıyla parçaya dalarak kesme yapar, başladığı noktaya tekrar hızlı hareketle geri çekilir.

FACE döngüsü, DRILL döngüsüne benzer aradaki fark, geri çekilmeden önce kesici takımın bir süre bekletilmesidir.

TAP veya kılavuz salma olarak adlandırılan döngü, deliklere diş açılması için kullanılmaktadır. Bu döngüde kılavuz takım belirli bir derinliğe kadar tezgah milinin belirli dönme hızıyla iner, burada fener mili dönme yönünü değiştirir ve aynı ilerlemeyle başlangıç noktasına kadar çıkılır.

BORE döngüsü, delik çapının yüksek hassasiyet gerektirdiği durumlarda kullanılmaktadır. Standart BORE döngüsünde takım belli bir derinliğe kadar belli bir ilerleme hızıyla iner, burada fener milini durdurur ve takımı hızlı konumlama ile başlangıç noktasına geri çeker. DWELL BORE döngüsü, standart BORE döngüsüne benzerdir aradaki tek fark, fener mili durdurulmadan önce takımın bir süre bekletilmesidir. ORIENT BORE döngüsünde ise, takım son derinliğe ulaştığında, fener mili belirlenen bir açı ile döndürülerek, takım ucunun silindirik duvarını temizlemesi sağlanır, ardından takım geri çekilir. Bu üç olası yöntemde de fener mili, takım ancak geri çekildikten sonra dönmesine devam eder.

REAM veya raybalama olarak bilinen döngü ise yüzey kalitesi yüksek deliklerin oluşturulmasında kullanılır. Döngüde kesici takım belirli bir ilerlemeyle

yine belirlenen derinliğe iner, buradan başlangıç noktasına aynı ilerleme ile geri çekilir. Döngüde isteğe bağlı olarak takım belirli bir süre bekletilebilir.

DEEP (Kademeli delik delme) ve BRKCHP (talaş kırma) döngüleri, derin deliklerin bir dizi kademeyeyle oluşturulması maksadıyla kullanılır. Kademeli delik delme döngüsünde kesici takım, her bir kademedan sonra başlangıç noktasına kadar hızlı hareket ederek bir gir-çık hareketi yapar ve böylece oluşan talaşın dışarıya atılması sağlanır. Talaş kırma döngüsü ise kademeli delik delme döngüsüne benzerdir. Ancak aradaki fark, delik derinliğindeki her bir artırımı hareketin sonunda takımın geri çekilme düzlemine kadar bütün bir yolu katetmesi yerine, belirlenen bir mesafeye kadar geri çekilmesinin sağlanmasıdır.

THRU döngüsü, belirli bir aralıktaki iki veya daha fazla plakalar halindeki parçaların delinmesinde kullanılır. Aynı eksendeki plaka deliği belirli bir ilerleme ile delinirken, bir sonraki plakadaki deliğe hızlı hareket ile takım yaklaştırılır. Bütün katmanlardaki delikler delindikten sonra takım ilk katmandaki başlangıç noktasına hızlı hareketle geri çekilir.

Çoğu CAM yazılımında olduğu gibi ProEngineer, CYCLE komutu açısından iki tipte kesme dosyası oluşturur. Sabit parametrelili döngü formu, ISO 4343-1978 standartlarına göre verilir. Eğer konfigürasyon opsiyonu olan “Cycle_Format”, “Fixed” olarak ayarlanırsa bu durumda, kesme dosyasında döngüler için aşağıdaki formda çıktı alınacaktır;

$$\text{CYCLE} / \left[\begin{array}{c} \text{DRILL} \\ \text{DEEP} \\ \text{BRKCHP} \\ \text{THRU} \\ \text{FACE} \\ \text{CSINK} \\ \text{TAP} \\ \text{BORE} \\ \text{REAM} \\ \text{OFF} \end{array} \right], z, f, \left[\begin{array}{c} \text{IPM} \\ \text{IPR} \\ \text{MMPM} \\ \text{MMPR} \end{array} \right], t, r, i$$

Burada;

z — takımın ilerlediği derinlik;

f — ilerleme hızı;

t — adım;

r — r derinliğine kadar hızlı hareket;

i — derinlik artırımı;

Bir diğer form ise önerilen ISO-4343 standardının gözden geçirilmesiyle yenilen Couplet standardıdır [Chang ve Melkanoff, 1989]. ProEngineer kesme dosyasında döngüler için varsayılan döngü çıktısı şekli Couplet formudur. uygulamada bu form kullanılmıştır.

CAM sistemlerinde kullanılan tüm döngülerin ortak parametreleri vardır. Couplet formuna göre bu parametreler kesme dosyası içinde asıl komutlar olarak bulunmaktadır. Bu parametreler şunlardır;

3.8.3.1. Operasyonun derinliği parametresi (DEPTH)

Kesme dosyası içinde operasyonun derinliği, her zaman döngü tipinden hemen sonra gelir ve isteğe bağlı bir parametre değildir. Operasyonun başlangıç noktasından, takımın ulaşacağı en son nokta arasındaki mesafe bu tali kelime değeriyle belirlenir.

[DEPTH,] *değer*

3.8.3.2. Kesme Hızı Parametresi (FEED)

Kesme dosyasında, döngü tanımlama kaydı içinde DEPTH tali kelimesinden sonra gelen, kullanımı zorunlu parametredir. İlerleme tipi ve ilerleme değeri olarak, gibi iki kısımda tanımlanmaktadır;

$$\begin{bmatrix} IPM \\ IPR \\ MMPM \\ MMPR \\ TPI \end{bmatrix}, \text{değer}$$

IPM	Dakikada inç olarak ilerleme
IPR	Her bir dönüş için inç olarak ilerleme
MMPM	Dakikada mm olarak ilerleme
MMPR	Her bir dönüş için mm ilerleme
TPI	İnç cinsinden adım

3.8.3.3. Boşluk Yüksekliği (CLEAR)

Boşluk yüksekliği, döngüde takımın harekete başladığı en üst nokta ile delik başlangıç yüzeyi arasındaki mesafedir. Operasyon sonunda, takım bu noktada tekrar konumlanacaktır.

[CLEAR,] *değer*

Boşluk mesafesi opsiyonel bir parametre değildir ve döngü tanımı kaydı içinde ilerleme hızından sonra gelen CLEAR tali kelimesiyle belirlenir.

CAM yazılımlarında, bahsedilen kullanılması zorunlu parametreler haricinde birde döngüdeki her bir kontrol noktasında çeşitli işlemler yapmayı sağlayan opsiyonlar bulunmaktadır. Bu opsiyonlar kayıt yapısı içinde zorunlu parametrelerden sonra herhangi bir sırayla gelebilir. Her bir opsiyon, farklı tipteki döngüler içinde aktif durumda olabilir.

3.8.3.4. RETURN Opsiyonu

Boşluk mesafesi ile belirlenen noktadan farklı, opsiyonel olarak takımı istenen noktaya çekmesini sağlar. Başka bir deyişle, belirlenen geri çekilme düzleminden farklı bir noktanın geri çekilme mesafesi olarak kullanılmasına yarar. Bu yüzden eğer varsayıldığı gibi sıfır değerinde bırakılırsa, takım geri çekilme düzlemine kadar çıkar.

[RETURN [, *değer*]]

3.8.3.5. RAPTO Opsiyonu

Bu opsiyon, döngüdeki her bir konumlama noktası için ekstra hızlı konumlanma noktası oluşturmada kullanılır. Bu opsiyon boşluk mesafesinin belirlendiği düzlemden (clearance plane) deliğin başlangıcına kadar mesafede, herhangi belirlenen bir noktada takımın hızlı hareketini sağlar.

[RAPTO [, *değer*]]

Bu opsiyon kullanıldığında, takım boşluk düzlemi üzerinde konumlandıktan sonra, RAPTO parametresi ile belirlenen derinliğe kadar hızlı hareket yapar.

3.8.3.6. DWELL Opsiyonu

DWELL opsiyonu, kesici takım belirli bir derinliğe indiğinde, programlanan bir süre için bekletilmesini sağlar. Bu programlama süresi, saniye veya tezgah mili dönme devir sayısı türünden programlanabilir.

$$\left[\left(\begin{array}{c} DWELL, \text{saniye} \\ REV, \text{tezgah_mili_devir_sayı} \end{array} \right) \right]$$

Bu opsiyon DRILL, FACE, BORE, REAM ve CSINK döngü tipleri ile kullanılabilir.

3.8.3.7. ORIENT Opsiyonu

ORIENT opsiyonu, kesici takımın durdurulup, belirli bir açıyla döndürülmesini sağlayan opsiyondur ve tolerans değeri düşük deliklerin oluşturulmasında kullanılabilir.

[ORIENT [, *açı*]]

Görüldüğü gibi döngüler, içlerinde birçok parametreyi barındırmaktadır. Asıl kelimenin tanımladığı döngü tipi tezgah kontrolörü tarafından tanınsa bile, döngü içinde genellikle opsiyonel tali kelimelerin tanımladığı prosesler kontrolör tarafından tanınmayabilir. Tezgahı tanımayan kullanıcı, isteğe bağlı bu parametreleri kullanabilir. Uygulamada tezgah hareketini etkilemeyen tali kelimelerin kullanılmasına izin verilmiş ve karşılaşılabilecek bu tür durumda, gerekli uyarı verilerek kullanıcı bilgilendirilmiştir. İsteğe bağlı tali kelimelerin döngü içinde yer alması döngünün tanımlanması için birden fazla alternatifi beraberinde getirir ve programcılık açısından kaydın parçalanmasını zorlaştırır.

ProEngineer kesme dosyalarında, kapalı döngüleri tanımlayan asıl komutlar şu şekildedir;

DRILL komutu, standart delik delme döngüsünü gerçekleştirir. CAM yazılımında döngü tipi opsiyonları içinde Drill-Standart kullanıldığında elde edilir.

Kesme dosyasındaki formu şu şekildedir;

CYCLE / DRILL, DEPTH, a, FEED, b, CLEAR, c, RAPTO, d, RETURN, e
GOTO / x, y,z

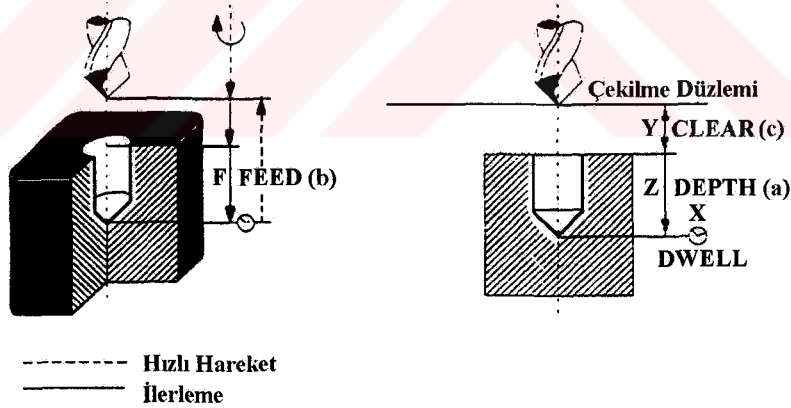
CYCLE / OFF

Burada;

- a — CAM yazılımlarının hesapladığı derinlik.
- b — CUT_FEED parametresi ile belirlenen ilerleme hızı.
- c — CLEAR_DIST parametresi ile belirlenen boşluk mesafesi.
- d — Opsiyonel RAPTO_DIST parametresi ile belirlenen değer.
- e — PULLOUT_DIST; parametresi ile belirlenen değer.

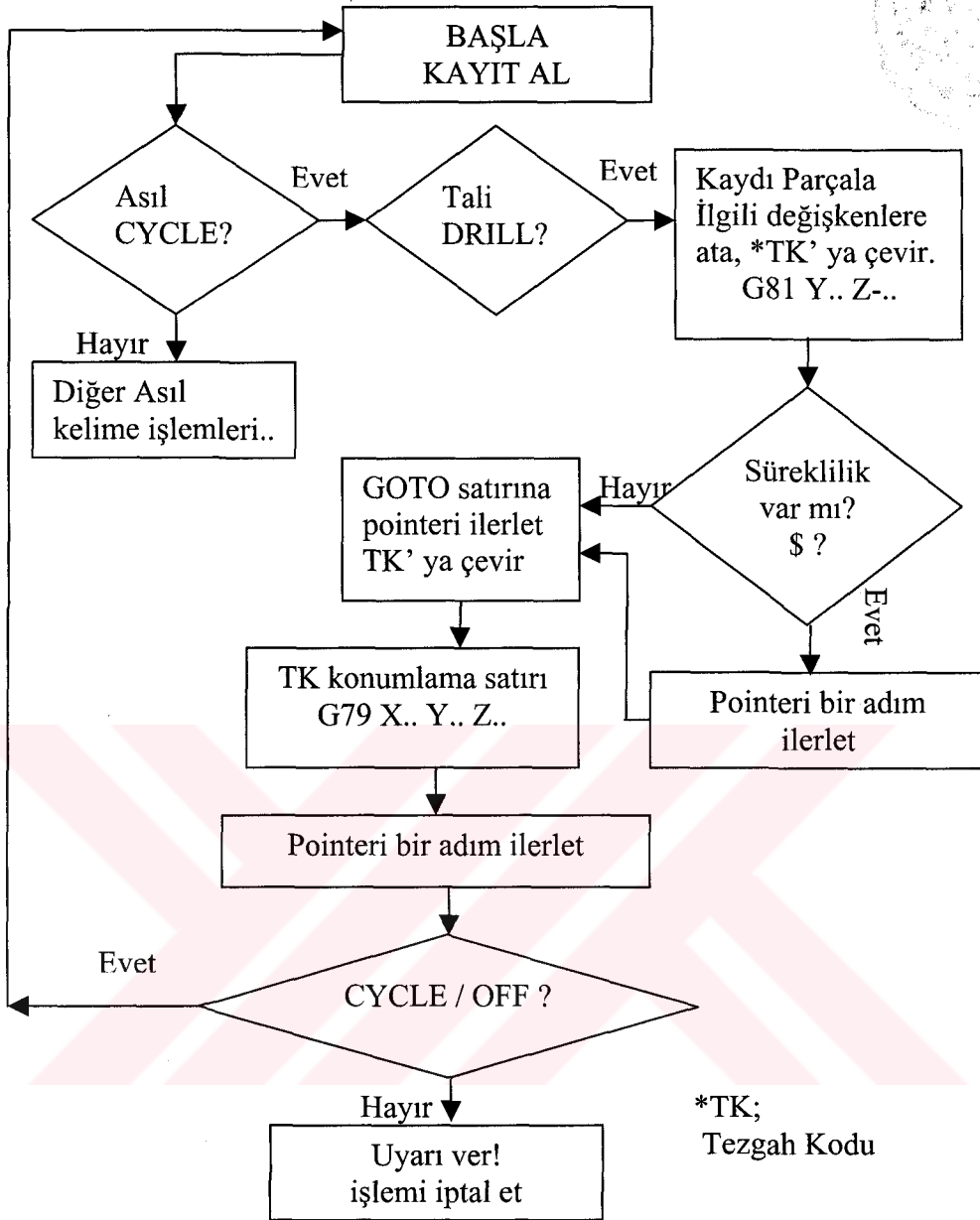
DRILL komutunun belirlediği rutin delik delme döngüsü tezgahta G81 ile tanımlanan döngüdür. Kesme dosyasındaki bu komut, Şekil 3.14’de gösterilen parametreler yardımıyla tezgah kontrol kodlarına çevrilir. döngünün tanımlı olduğu örnek bir kesme dosyası çıktısı ve makina kodlarına çevrilmiş hali Ek. 16’da verilmiştir.

N.. G81 Y(c) Z-(a) F(b)
N.. G79 x y z aktivatör satırı (GOTO)



Şekil 3.14. DRILL Döngüsü Tanımlaması

Standart delme (DRILL) döngüsü için, uygulama da Şekil 3.15’te gösterilen algoritma kullanılmıştır.



Şekil 3.15. DRILL döngüsü için kullanılan algoritma

Bazı opsiyonel parametrelerin döngü içinde yer alması tezgah için dolayısıyla postprosesör için bir anlam ifade etmez. Örneğin, tezgah döngüsü içinde PULLOUT_DIST parametresine karşılık gelen herhangi bir kod yoktur. Bu yüzden uygulamada, bu tali komut ve dolayısıyla değeri dikkate alınmamıştır. Seri imalatta faydası olabilecek ve nadiren kullanılan RAPTO tali komutu ise, uygulamada dikkate alınmamıştır.

RAPTO tali komutu değeriyle, Y değeri arasında aşağıdaki şekilde bağıntı vardır;

$$Y = c - d$$

Y değeri, RAPTO opsiyonu belirlendiğinde, CLEAR opsiyon değerinden çıkarılarak elde edilir. Eğer RAPTO değeri sıfır olarak belirlenmişse tezgah güvenli geri çekilme düzlemine kadar kesici takımı çekecektir. Bu durum sistemde varsayılan olarak güvenli kullanılabilir. Ancak RAPTO değerinin, CLEAR değerden büyük olması durumunda takım delik başlangıç yüzey seviyesinden daha aşağı seviyedeki noktaya hızla hareket edecektir. Başka bir deyişle takım, hızlı hareketle iş parçasına çarpacaktır. CAM yazılımlarında delik delme döngüleri için dalma kontrolörü analizinin yapılmadığı ve takım yolunun grafiksel testi aşamasında döngüyü oluşturan tüm kontrol noktalarının net şekilde görülemediği düşünülürse, opsiyonların tecrübesiz kullanımları, çeşitli takım çarpmalarına neden olacaktır. Her şeyin iyi planlandığı düşünülse bile, sadece birkaç milimetrelik hızlı hareket için tezgah ve takım güvenliği uygulamada riske atılmak istenmemiştir.

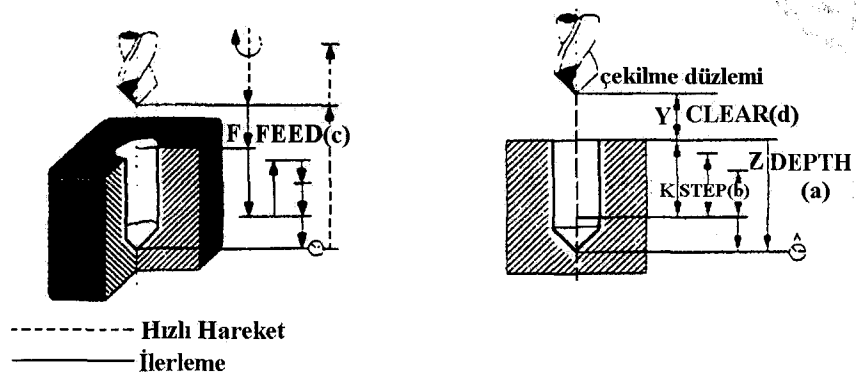
DRILL döngüsünde olduğu gibi diğer döngü kayıtları içindeki, opsiyonel RAPTO ve RETURN tali kelimeleri uygulamada dikkate alınmamıştır.

DEEP komutu, derin deliklerin kademeli şekilde delinmesini tanımlar. CAM yazılımında aşağıdaki formda bulunur.

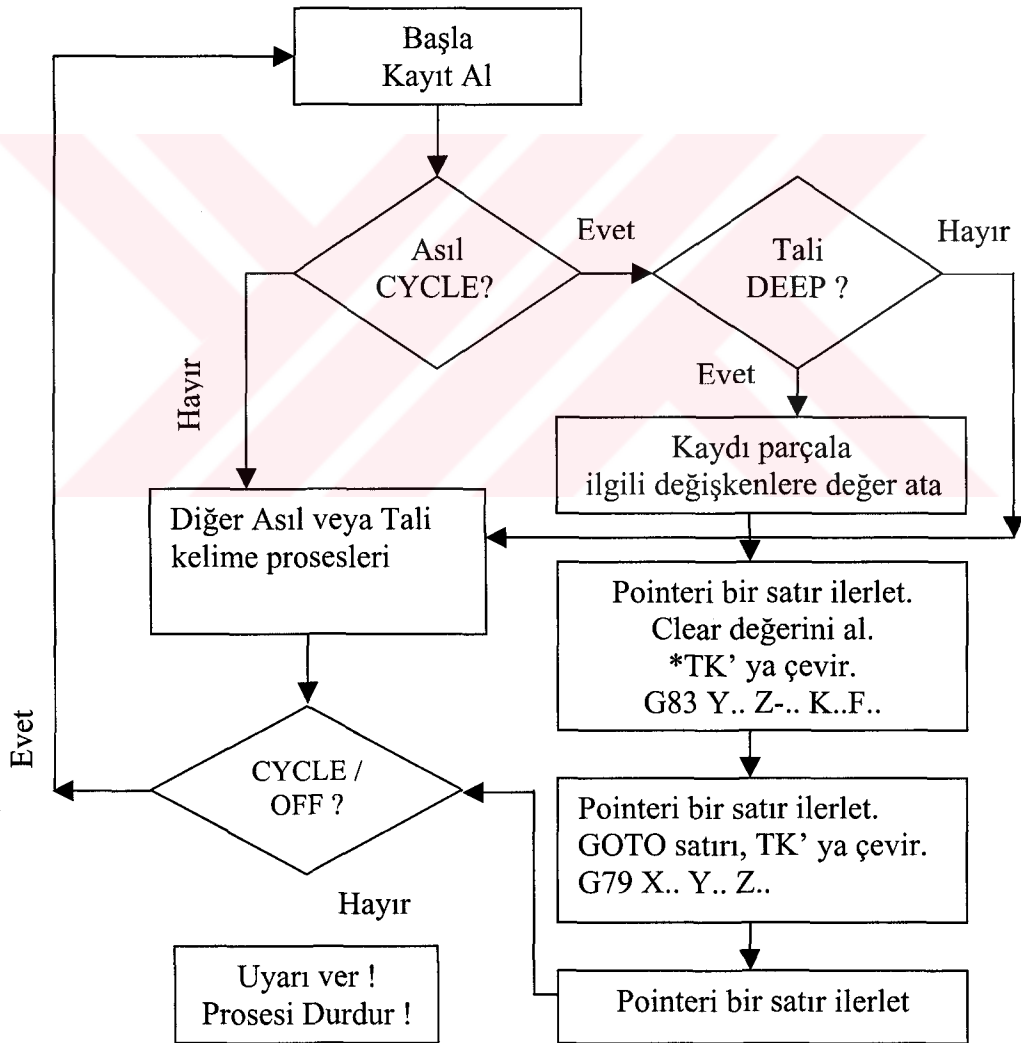
CYCLE / DEEP, DEPTH, a, STEP, b, FEED, c, CLEAR, d, RAPTO, e, RETURN, f
GOTO / x, y, z
CYCLE/ OFF

DRILL komutunda farklı olarak buradaki tek fark STEP tali komutudur ve bu komut değeri b, yazılım içinde PECK_DEPTH parametresi ile verilen artırımlı derinlik mesafesidir.

N.. G83 Y(d) Z-(a) K(b) F(c)
N..G79 x y z (GOTO aktivatör satırı)



Şekil 3.16. DEEP Komutu Tanımlaması



*TK; tezgah kodu

Şekil 3.17. DEEP Komutu İçin Uygulamada Kullanılan Algoritma

DEEP komutu ve tezgahdaki karşılığı ile ilgili bir örnek Ek.17'te verilmiştir.

BRKCHP komutu, talaş kırma döngüsünü tanımlar. Tüm parametrelerle tanımlandığında kesme dosyasında aşağıdaki formda bulunur;

CYCLE / BRKCHP, DEPTH, a, STEP, b, FEED, c, CLEAR, d, RAPTO, e,
RETURN, f, DWELL | BACK | REV, g, TIMES, h

GOTO /x, y, z

CYCLE / OFF

Bu döngü her ne kadar ISO 1056 standardı içinde G83 kodu ile tanımlı olsa da, uygulamada kullanılan tezgah kontrolörü döngüyü tanımaz. Döngünün yaptığı işlem benzer bir döngü olan kademeli delik delme döngüsü kullanılarak yapılabilir ya da kesme dosyası içinde COUPLET yerine sadece GOTO komutlarıyla tanımlanabilir.

Uygulama içinde, BRKCHP komutuyla karşılaşıldığında kullanıcıya, döngünün derin delik döngüsüne çevrilmesi gerektiği veya kesme dosyasında döngü formunun GOTO şeklinde ayarlanması gerektiğine dair bir uyarı veren yordam kullanılmaktadır.

Bu komut ve tezgah kontrol verilerine göre karşılığı Ek.18'te verilmiştir.

THRU komutuyla, CAM yazılımında döngü tiplerinden WEB döngüsü seçildiğinde karşılaşılır. Kesme dosyasındaki formu aşağıdaki şekildedir.

CYCLE / THRU, DEPTH, a1, a2, a3, ..., FEED, b, CLEAR, c, RAPTO, d,
RETURN, e

GOTO /x,y,z

CYCLE / OFF

Diğer döngülerden farklı olarak, a1, a2, a3 değerleri katmanlar arası mesafeyi gösterir.

THRU komutunun, ISO 1056 standardında tanımı yoktur. Dolayısıyla bir çok kontrolör tarafından tanınmaz. Uygulamada komutla karşılaşıldığında, bir uyarı mesajı verilerek, döngü kullanımını iptal eden bir yordam kullanılmıştır.

FACE komutunun kullanımı, DRILL komutuna benzerdir. Kullanımdaki tek fark, kesme dosyasında komut tanımına, bekleme süresi, DWELL tali komutunun eklenmesidir. FACE komutu, ISO 1056 standardında G82 kodu ile tanımlanan döngüdür, ancak uygulamada kullanılan tezgah kontrolörü bu kodu tanımaz. Bu yüzden uygulamada döngü, tezgah kontrolöründe tanımlı G81 kod içeriğine bekleme süresi eklenerek DRILL komutuna benzetilerek oluşturulur.

Kesme dosyası içinde aşağıdaki formda bulunur;

CYCLE / FACE, DEPTH, a, FEED, b, CLEAR, c, RAPTO, d, RETURN, e, DWELL
| REV, f
GOTO / x,y,z
CYCLE / OFF

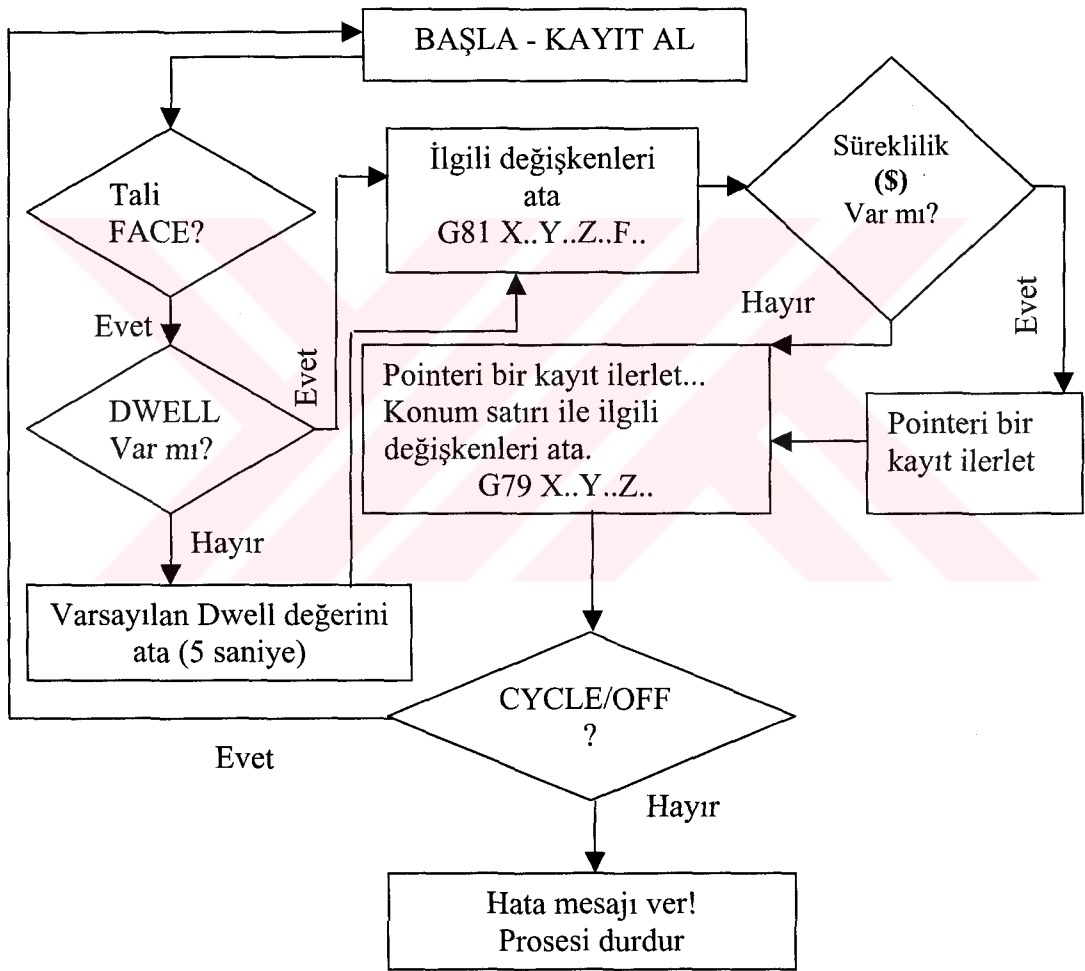
FACE komutu satır sentaksı içinde görüldüğü gibi bekleme süresi (DWELL), iki şekilde belirlenebilir. CNC tezgahlarında bekleme süresi, tezgah mili dönme devri veya saniye türünden tanımlanabilir. Uygulamada kullanılan tezgah için geçerli olan, bekleme sürelerinin saniye türünden verilmesidir. Uygulamada kullanıcı bekleme süresi olarak, tezgah mili dönme devrini kullanırsa bu durumda, varsayılan saniye cinsinden bir değer (5 saniye) bekleme süresi olarak atanacaktır. Ayrıca, kesme dosyası içinde FACE asıl komutu ile karşılaşıldığı, ancak bu komut içinde DWELL tali komutunun kullanılmadığı durumlarda, yine bekleme süresiyle ilgili varsayılan atama gerçekleştirilecektir.

FACE komutu, DRILL komutuna benzetildiğinde aşağıdaki makina kod formu oluşturulur;

N..G81 X(f) Y(c) Z-(a) F(b)

N..G79 x y z

Makina kontrol kodları içinde bekleme süresi, X kodu ile tanımlanır. Uygulamada, yordam içinde Şekil 3.18’de verilen algoritma kullanılmıştır.



Şekil 3.18. FACE Komutu İçin Kullanılan Algoritma

Ek 20’de FACE komutu ve tezgah kodlarına çevrilmiş biçimi verilmiştir.

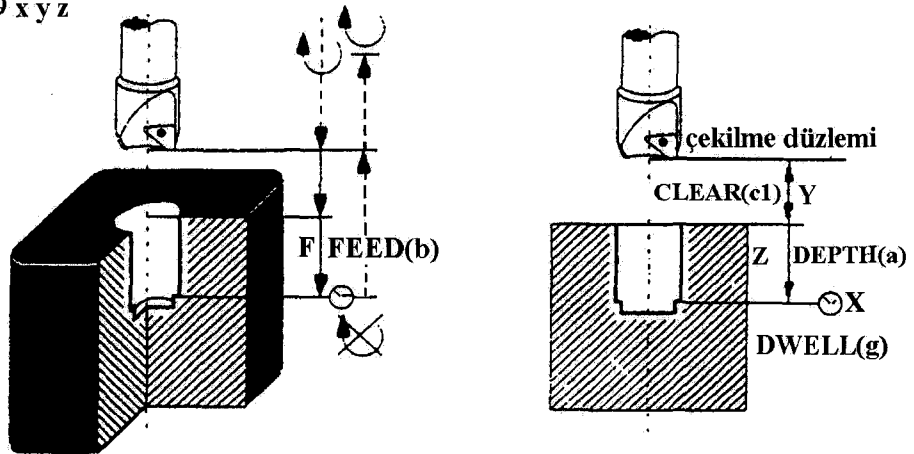
BORE komutu, ISO 1056 standardında G86 ile tanımlanmıştır. Kesme dosyası içinde aşağıdaki formda bulunur;

CYCLE / BORE, DEPTH, a, FEED, b, CLEAR, c1, c2 ,RAPTO, d, RETURN, e,
ORIENT, f, DWELL | REV, g
GOTO / x, y, z
CYCLE / OFF

Diğer komutlardan farklı olarak, kayıt yapısı içindeki “c2”, jog mesafesi olarak tanımlanan, kesici takımın belirlenen değerde delik ekseninden kaydırılmasını sağlayan değer ve f, ORIENT_ANGLE parametresi ile belirlenen değerdir.

BORE döngüsü kayıt yapısı içinde, opsiyonel DWELL ve ORIENT tali komutları birlikte kullanılmaz. Normalde postprosesörün bu durumda hata mesajı verip işlemi durdurması gerekir. Ancak uygulamada kullanılan tezgah için belirlenen bu opsiyonlardan sadece DWELL tali komutu, yani bekleme süresi geçerli olduğundan, uygulamada sadece bu opsiyon araştırılır. Diğer opsiyonların varlığı dikkate alınmaz ve işlem durdurulmaz. BORE komutu ve makina kodlarına çevrimi Şekil 3.19’da gösterildiği gibidir.

N.. G86 X(g) Y(c1) Z-(a) F(b)
N.. G79 x y z



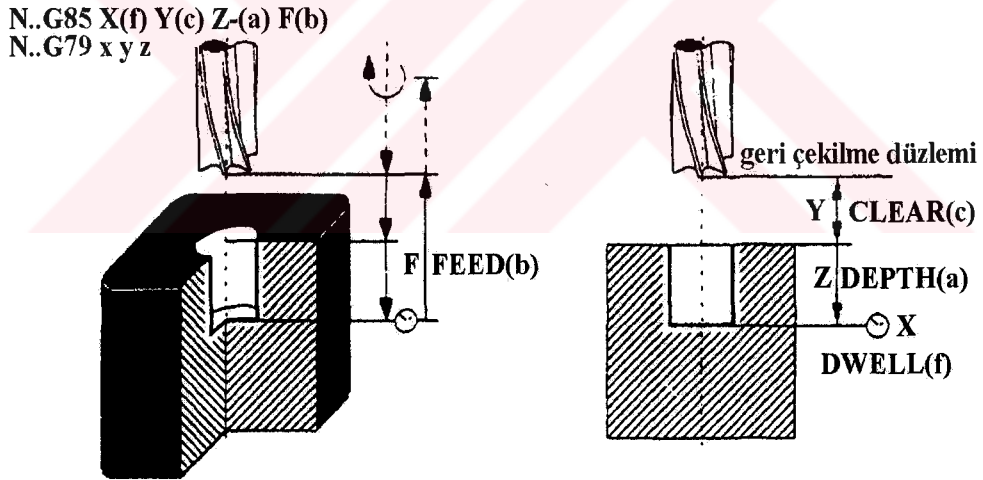
Şekil 3.19. BORE Komutu Ve Makina Kodlarında Karşılığı

Uygulamada, FACE döngünde kullanılan algoritmaya benzer bir algoritma kullanılmıştır. Aradaki fark, BORE kaydı, DWELL-tali komutunun opsiyonel olarak bırakılmasıdır. Kayıt içinde bu tali varsa işlenir yoksa geçilir. Bu döngü ile ilgili bir kesme dosyası örneği ve makina kodlarındaki karşılığı Ek 21’de verilmiştir.

REAM komutu, ISO 1056 standardında G85 kapalı döngüsünü temsil eder. Bu komutun kesme dosyası içindeki genel formu şu şekildedir;

CYCLE / REAM, DEPTH, a, FEED, b, CLEAR, c, RAPTO, d, RETURN, e,
 DWELL | REV, f
 GOTO \ x,y,z
 CYCLE \ OFF

REAM komutu ve makina kodlarına çevrimi Şekil 3.20’de gösterildiği gibidir.



Şekil 3.20. REAM Komutu Tanımlanması

REAM komutu kayıt yapısı, FACE komutu kaydına benzerdir. Uygulamada REAM komutu için, FACE komutuna benzer bir algoritma kullanılmıştır. Döngü içinde tanımlanabilecek bekleme süresi ise yine opsiyoneldir.

Ek 22’de REAM komutu ve tezgah kodlarına çevrilmiş biçimi verilmiştir.



CSINK komutu, deliklere pah kırmayı sağlayan döngüyü tanımlar. ISO 1056 standardı içinde G81 koduyla tanımlanmıştır. Kesme dosyasında şu formda bulunur.

CYCLE / CSINK, DIAMET, a, TLANGL, b, FEED, c, CLEAR, d, RAPTO, e,
RETURN, f, DWELL | REV, g
GOTO / x,y,z
CYCLE / OFF

Diğer döngülerden farklı parametrelerin tanımı şöyledir;

a — kesici takım çapı.

b — kesici takım uç açısı.

Kesme dosyasında verilmeyen derinlik değeri, postprosesör tarafından hesaplanmalıdır. Z değeri şu eşitlik vasıtasıyla hesaplanır;

$$Z = [a * \sin(b) * -1] \quad (3.16)$$

Diğer parametreler ve uygulamada bunlara yaklaşım, önceki konularda anlatılan döngülere aynıdır. Döngü için kullanılan yordamın algoritması, DRILL komutunda verilenle benzer ancak, Z derinliğinin yordam içinde hesaplanması gerekir;

G81 X(g) Y(d) Z(değer) F(c)

G79 x y z

CSINK komutunun bulunduğu bir kesme dosyası ve makina kodlarına çevrilmiş hali Ek 23'te verilmiştir.

TAP komutu, ISO 1056 standardında G84 koduyla belirlenen, dış açma yada kılavuz salma olarak adlandırılan döngüyü tanımlar.

Kesme dosyası içinde bu komut kaydının genel formu şu şekildedir;

CYCLE / TAP, DEPTH, a, , b, CLEAR, c, RAPTO, d, RETURN, e

GOTO / x, y, z

CYCLE / OFF

Diğer döngülerden farklı olarak belirlenen, TPI, MMPR, IPR, tali komutları, CAM yazılımında CUT_UNITS ile belirlenen vida adım birimleridir. Vida adımı değeri ise “b” ile gösterilmiştir.

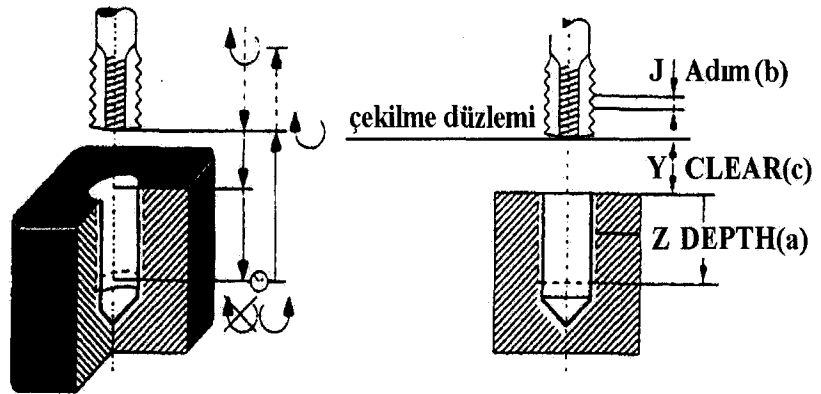
İstenen adımda diş açılması, döngü içindeki ilerleme hızı, tezgah mili dönme devri ve adım değerine bağlı olarak postprosesör tarafından hesaplanacaktır;

$$F = \text{Tezgah Mili Dönme Devri (S)} * \text{Adım Değeri (J)} \quad (3.17)$$

TAP komutunun, tezgah kontrol verilerine göre karşılığı Şekil 3.21’de verildiği gibidir.

N..G84 J(b) Y(c) Z(a) F(değer)

N..G79 x y z



Şekil 3.21. TAP Komutu Tanımlaması

TAP komutu ile ilgili bir kesme dosyası örneği Ek 24’te verilmiştir.

3.8.4. Yardımcı Komutlar

Kesme dosyalarında hareket komutlarının yanı sıra, bu hareketleri destekleyen, takım değiştirme ilerleme hızı, dönme devri, soğutma sıvısı gibi bir dizi yardımcı komutlar bulunur;

COOLNT komutu vasıtasıyla tezgahın soğutma sıvısı kontrol edilmektedir. Kesme dosyası içinde şu formda bulunabilir;

COOLNT / $\left[\begin{array}{c} ON \\ OFF \\ FLOOD \\ MIST \\ TAP \\ THRU \end{array} \right]$

Kesme dosyasında bu komutla karşılaşıldığında, Tablo 3.5'te verilen, tezgahın soğutma sıvısıyla ilgili kodlara çevrilir. COOLNT asıl komutundan sonra gelen diğer tali TAP ve THRU komutları tezgah kontrolörü için tanımlı değildir. ON tali komutu ise, uygulamada dikkate alınmamış, bunun yerine, tezgah mili dönme devri ile soğutma sıvısı kontrolünü aynı anda yapan kodlar kullanılmıştır.

Tablo 3.4. Soğutma Sistemi Kontrol Kodları

Tali komut	Tezgah kontrol kodu	Açıklaması
OOF	M09	Soğutma sıvısını kapatır.
FLOOD	M07	Sıvı soğutmasını aktif eder.
MIST	M08	Hava soğutmasını aktif eder.

SPINDL komutu, tezgah mili dönme devrini ve yönünü kontrol eden asıl komuttur. Kesme dosyası içinde şu formda bulunur;

$$SPINDL / \begin{bmatrix} RPM \\ SFM \\ SMM \end{bmatrix}, [\text{hız değeri}], \begin{bmatrix} CLW \\ CCLW \end{bmatrix}$$

SPINDL / OFF

Tezgah mili dönme devri, dakikada dönme devri ve kesici takım temas noktası hızının bir fonksiyon olarak verildiği, iki şekilde tanımlanabilir. Birinci tip, RPM olarak adlandırılır. İkinci tip ise SFM veya SMM (dakikada yüzeyde dönme devir hız birimleri) olarak adlandırılır. Uygulamada sadece RPM tali kelimesi dikkate alınmıştır. Diğer tipteki dönme devirleri, kesici takımın ayrıntılı bilgilerini taşıyan ilave bir kesme dosyası komutu gerektirir. Varsayılan olarak CAM sistemi içinde böyle bir komut bulunmamaktadır. Uygulamada bu parametrelerin kullanılması durumunda, sadece uyarı mesajı verilmekte ve dönme değeri olarak belirlenen değer yine RPM olarak atanmaktadır.

Bir diğer tali komut olan, OFF komutu ise, tezgah soğutma sistemini kapatır. Tablo 3.5'te tali komutlar ve tezgah için karşılıkları verilmiştir.

Tablo 3.5. Tezgah Mili Kontrol Kodları

Tali komut	Tezgah kontrol kodu	Açıklaması
OOF	M5	Fener milini durdurur.
CCLW	M14	saat yönünün tersine hareket
CLW	M13	Saat yönünde hareket

Tezgah mili, M14 ve M13 kodları haricinde M3 ve M4 komutlarıyla da kontrol edilebilir. Ancak uygulamada, tezgah mili ile birlikte soğutma sıvısı sistemini kontrol edebilen bu kodlar kullanılmıştır.

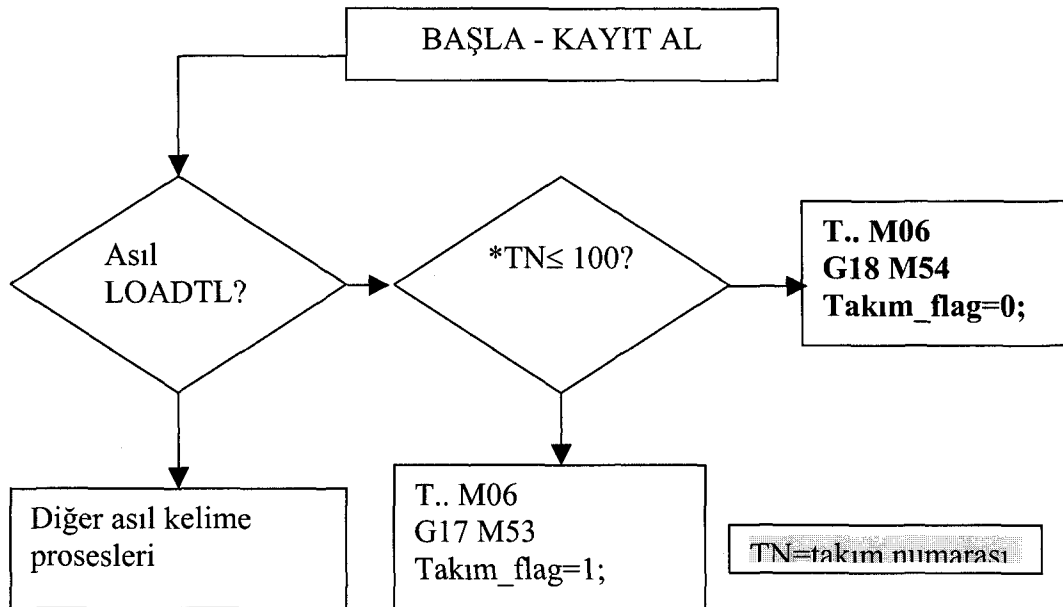
LOADTL komutu, tezgah magazinindeki herhangi bir kesici takımın çağırılmasını sağlar.

LOADTL / takım_numarası

Takım numarası kullanıcıya bağlı olarak atanan bir tamsayı değerdir ve bu değer kesici takımın magazindeki yerini (adresini) temsil eder.

Bu komutun, uygulamada takım değiştirme haricindeki çok önemli bir diğer fonksiyonu, tezgah çalışma düzlemi değişimini kontrol etmesidir. Gerçekte komutun böyle bir fonksiyonu olmadığı gibi çalışma düzlemini değiştirmeyi sağlayan herhangi bir kesme dosyası komutu da yoktur. Bu yüzden uygulamada, çalışma düzleminin belirlenmesi, yani tezgah kafasının yatay veya dikey olarak çalıştırılması, bu komutla ilişkilendirilmiştir. Eğer kullanıcı, operasyonda kullanacağı takım numarasını 100 veya daha küçük değerle belirtirse, bu durumda tezgah için ilgili yatayda çalışma (ZX düzlemi) kodları kullanılacaktır. Kullanıcı takım numarasını 100'den büyük belirlerse, bu durumda tezgahı XY düzleminde yani dik pozisyonunda çalıştıracak kodlar kullanılır.

Tezgah milinin yatay ve dikey pozisyonlarında, takım konumlanması farklı olduğundan, Şekil 3.22'de algoritması verilen yordam içinde flag kullanılmış ve böylece, komutla ilgili diğer fonksiyonlarda da çağırılarak kullanılmıştır.



Şekil 3.22. LOADTL Komutu İçin Kullanılan Algoritma

MULTAX komutu 9000 kayıt sınıfı içinde tanımlanır ve kesme dosyası oluşturan herhangi bir CAM sisteminde 5 eksen hareketi işaret eder. Her bir hareket için takım yönünü gösteren takım eksen vektörleri, bu komuttan sonra düzenli X, Y ve Z lineer koordinatlarına ek olarak hesaplanarak, kesme dosyasına yazılır. Bu komut sayesinde postprosesörler kesme (CLDATA) dosya içinde çok eksenli bir hareketin olduğunu anlar ve proseste ona göre hesaplamalar yaparlar.

Bir kesme dosyasında, gerek lineer, dairesel, gerekse döngüsel tüm hareketlerde, takım konumlanması için GOTO asıl komutu kullanılır. Buraya kadar bahsedilen tüm komutlarda, takım konumlanmaları 3 eksen üzerine bina edilir ve tezgahta konumlama için, sadece X, Y ve Z ile tanımlanan koordinat değerleri kullanılır. Söz konusu 5 eksen hareket için, GOTO asıl komutuna takım eksen vektörünün verilerini taşıyan birim vektör bilgileri ilave edilir;

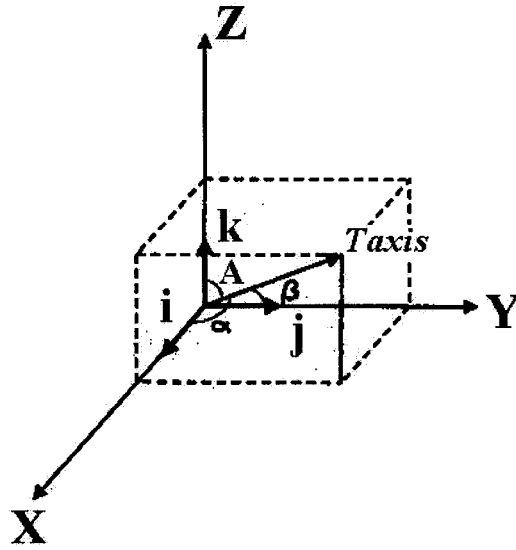
GOTO / x, y, z
i, j, k

Burada i, j ve k değerleri takım eksen vektörünü tanımlar. CAM yazılımları, bu vektörü birim vektör olarak verirler. Takım eksen vektörü ve tezgah kinematiği bilgileri birleştirilerek, kesme dosyasındaki takım konumları, postprosesör tarafından tezgaha uygun doğru konumlara çevrilmelidir.

Yapılması gereken ilk iş, A ve C tabla rotasyon açılarının belirlenmesidir.

A rotasyon değeri için, Denklem 3.12’de verilen eşitlik kullanılabilir. Ancak uygulamada, A rotasyon değerinin tam olarak, doğrudan hesaplanmasını sağlayan bir başka yöntem kullanılmıştır.

Bu yöntem matematikte, doğrultu açıları ve doğrultu kosinüsleri kavramı tanımından çıkarılmıştır [Görgülü,2000]. Bu yöntemle göre CAM yazılımında takım eksen vektörünü tanımlayan üç vektörden, k vektörünün doğrultu kosinüsü bize doğru açıyı verecektir (Şekil 3.23).



Taxis: takım eksen vektörü

Şekil 3.23. Takım Eksen Vektörü

Öncelikle üç eksen takım yolu için düşünüldüğünde, takım eksen vektörü tezgahın Z eksenine hizalanmış olacaktır. Bu durumda birim vektör $(0\mathbf{i}+0\mathbf{j}+1\mathbf{k})$ vektör bileşenleri ile ifade edilir. Eğer tezgah konumlaması için gerekli A açısını, doğrultu kosinüsüne göre hesaplanırsa;

$$\cos(A) = \left[\frac{k}{|T_k|} \right] \quad (3.18)$$

Burada T_k , birim vektördür (takım eksen vektörü).

Denklem 3.18'de verilen eşitliği, $T_k = 0\mathbf{i} + 0\mathbf{j} + 1\mathbf{k}$ için uygulanacak olursa;

$$|T_k| = \sqrt{0^2 + 0^2 + 1^2} = 1$$

değeri elde edilir.

Pratikte bu işlemi yapmaya gerek yoktur çünkü, Cam yazılımı T_k vektörünü zaten birim vektör olarak tanımlar. Dolayısıyla her zaman, $|T_k| = 1$ olacaktır. Bu durumda;

$$\cos(A) = \left[\frac{1}{1} \right] = 1 \quad \text{olur.} \quad (3.19)$$

Buradan A açısı, 0° olarak hesaplanır. Yani tezgah tablası açısal hareket yapmamaktadır. 5 eksen takım yolunda ise i, j, k vektörleri değişir dolayısıyla A açısal hareketinin değeri de değişmektedir. Bu yöntemin avantajı, A değerlerinin doğrudan hesaplanabilmesidir.

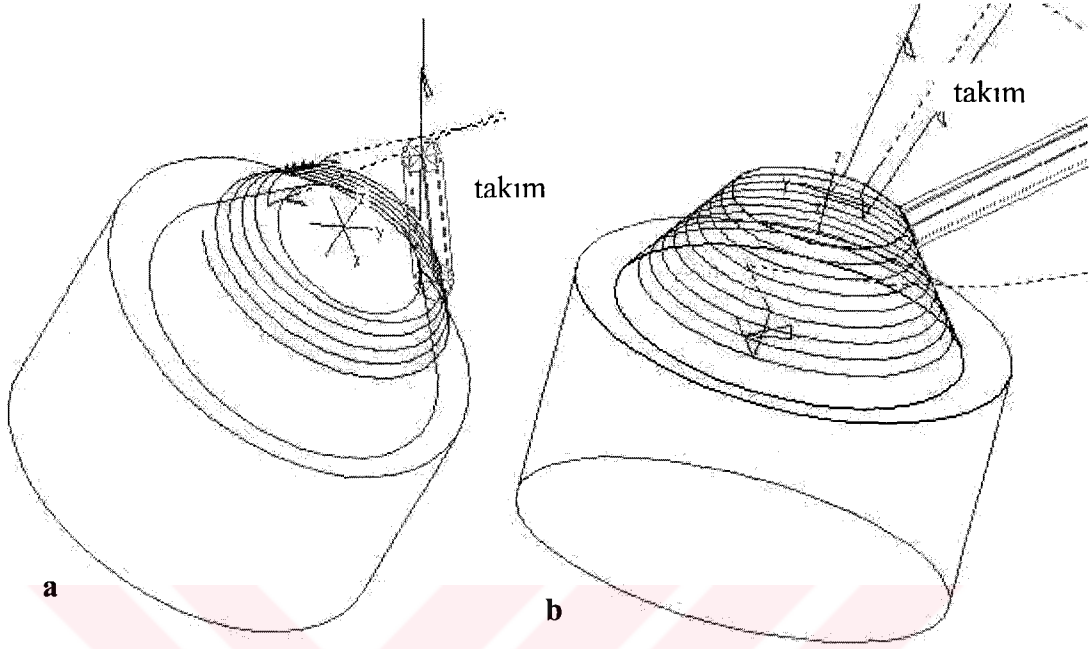
Uygulamada, tezgahın C rotasyon değerini hesaplamak için denklem 3.8'de verilen eşitlik kullanılmaktadır.

5 eksen eşzamanlı hareket için, A ve C rotasyonları kesin değerlerinin, verilen bu denklemlerle hesaplanması mümkün değildir. Doğru konumlama için öncelikli olan doğru açıların hesaplanması olduğundan, bir kere A ve C eksenini için rotasyon değerleri hesaplandıktan sonra, değerlerin tezgah koordinat sistemine göre analizlerinin yapılması gereklidir.

Uygulamada, doğru A ve C rotasyon değerleri, tezgahta çeşitli denemeler yapılarak bulunmuştur. Takım eksenini yön vektörünü tanımlayan i, j, k vektörlerinin, makina koordinat sistemi üzerindeki yeri, yani hangi bölgede bulunduğu bizim için önemlidir. Uygulamadaki tezgah için doğru rotasyon analizi temelde ikiye ayrılır. XY düzleminde ve ZX düzleminde A ve C rotasyon değeri analizleri farklı şekilde sonuçlanmaktadır.

Eşzamanlı 5 eksen takım yolu oluşturmak maksadıyla, Şekil 3.24'te gösterilen, test parçası tasarlanmıştır. Parça üzerinde oluşturulan koni, tezgah tarafından hem dik (Şekil 3.25(a)) hem de yatay pozisyonda (Şekil 4.25(b)) işlenecek şekilde takım

yolu oluşturulmuştur. Bu test parçası düşünülerek önce XY düzlemindeki, daha sonra ZX düzlemindeki doğru rotasyon analizleri incelenecektir.



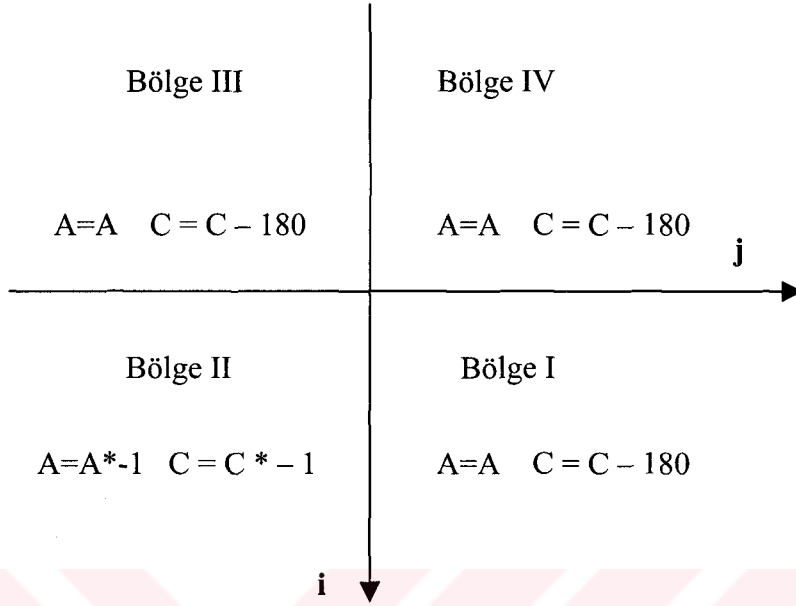
Şekil 3.24. Doğru Rotasyon Değerleri Analizi İçin Tasarlanan Test Parçası

XY düzlemi için, özellikle A rotasyon değerinin, doğru takım konumlaması açısından önemi büyüktür. Uygulama yapılan tezgahta tabla, X ekseninde saat yönünde 30° , tersinde 45° olmak üzere, toplamda 75° açıyla, iki yönde hareket yapabilir.

Tablanın bu iki yöndeki hareketi, iki alternatif yol ortaya çıkarmaktadır. Örneğin, tezgah koordinat sistemine göre 10° açı yapan bir yüzey A ekseninin her iki yöndeki hareketiyle de işlenebilir. Bu durumda bir kabullenme yapılması gerekir. Uygulamada, A rotasyon değeri 30° veya daha küçük hesaplandığında negatif kabul edilir. Böylece takım, C ekseninde daha az hareket edecektir. A ekseninde yapılabilecek bu alternatif iki hareket nedeniyle, takımın doğru konumlaması için farklı iki analiz yapılması gerekir.

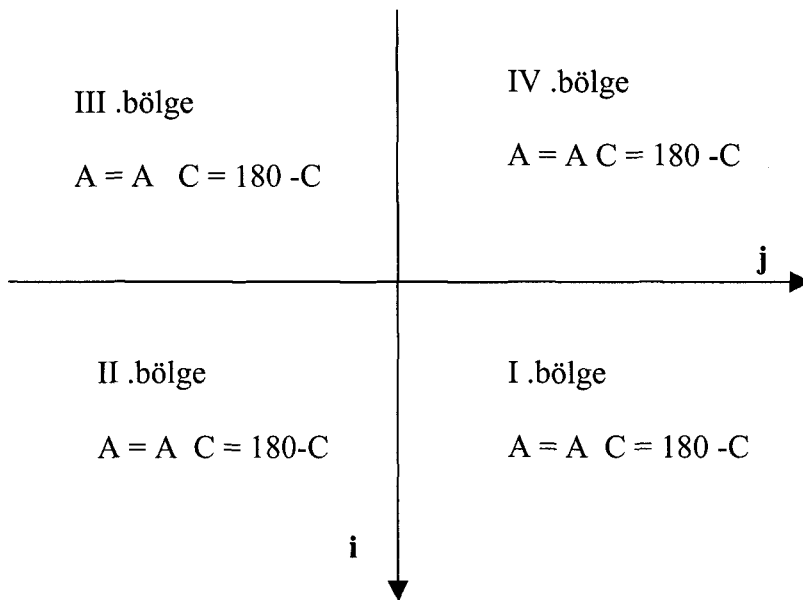
Uygulamada, A ve C rotasyon değerleri, verilen denklemler yardımıyla hesaplandıktan sonra doğru rotasyon değeri analizi tezgahta çeşitli denemelerden

sonra belirlenmiştir. Analizde belirlenen bölgeler, tezgah koordinat sistemine göre, Şekil 3.25 ve Şekil 3.26’da gösterildiği gibi gibidir.



Şekil 3.25. XY Düzleminde $A < 30^\circ$ İçin Kesin Rotasyon Analizi

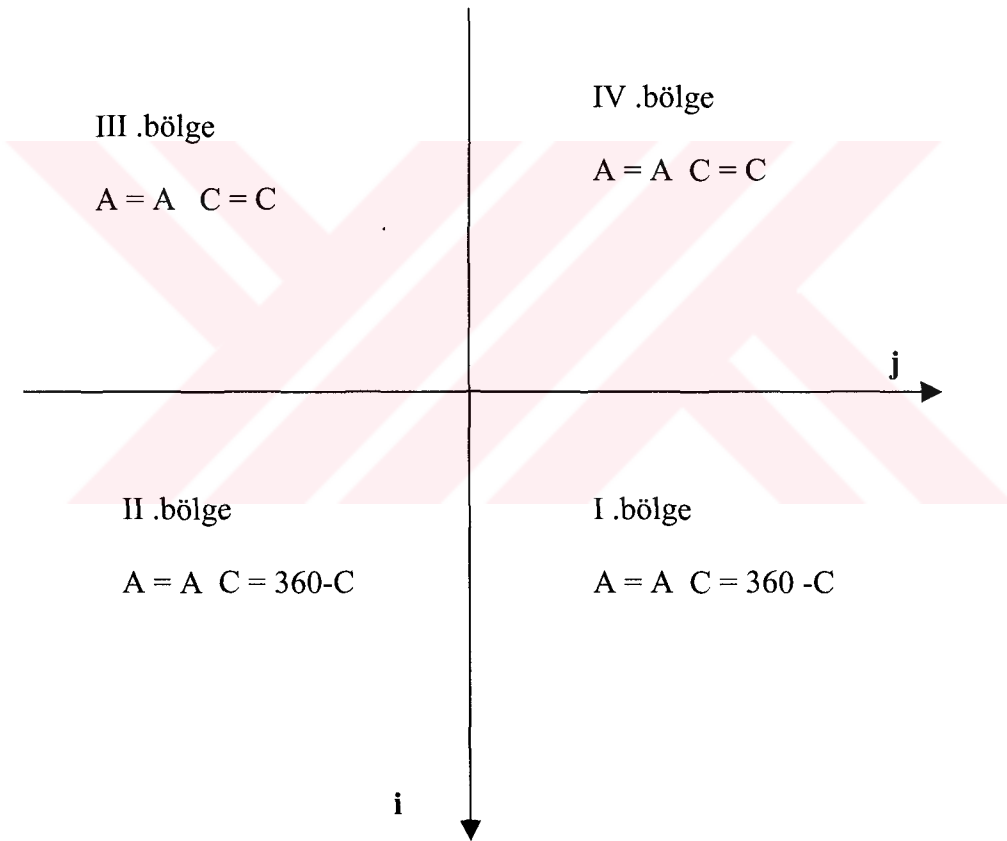
Hesaplanan A rotasyon değeri, 30° ’den küçükse bu durumda Şekil 3.25’te gösterilen analiz yapılır. Değer daha büyükse, bu durumda Şekil 3.26’da verilen analiz kullanılır. Dikkat edilirse A rotasyon değerleri pozitifdir ve hiçbir değişikliğe uğramamıştır.



Şekil 3.26. XY Düzleminde $A > 30^\circ$ İçin Kesin Rotasyon Analizi

A ve C için doğru rotasyon hesaplamaları yapıldıktan sonra bir diğer işlem tezgah kinematiğine göre takımın, X, Y, Z lineer koordinatlarını hesaplamaktır. Bu koordinatlar, eğilmeli-döner tabla kinematiğine uygun olarak Denklem 3.1'de verilen eşitlik kullanılarak hesaplanır.

ZX düzlemindeki çalışmalarda ise, koordinat sistemi düşünüldüğünde, A rotasyon değerinin 90° 'den çıkartılması gerekir. Diğer düzlemdeki gibi, Şekil 3.27'de gösterilen analiz, tablanın doğru A ve C rotasyon değerini hesaplamayı sağlayacaktır.



Şekil 3.27. ZX Düzleminde A İçin Doğru Rotasyon Değer Analizi

Doğru açılar hesaplandıktan sonra XY düzleminde kullanılan, Denklem 3.4 eşitliği kullanılarak X, Y ve Z lineer koordinatları hesaplanır.

UNITS komutu, kesme dosyasındaki veri ve ifadelerin hangi birimde olduğunu gösterir. 9000 sınıfı kaydı olan bu komut kesme dosyası içinde şu formda bulunur;

$$\text{UNITS} / \begin{bmatrix} \text{INCH} \\ \text{FEET} \\ \text{MM} \\ \text{CM} \end{bmatrix}$$

Uygulamada tezgah kontrolörü tarafından tanınan, iki temel birim sistemini tanımlayan INCH ve MM tali kelimeleri dikkate almıştır.

Tali Komut

Tezgah Kontrol Verisi

INCH

G70

MM

G71

Uygulamada tali kelimeye bağlı olarak yordam içinde, bir flag tanımlanmış, böylelikle tezgah kontrol veri dosyasındaki çeşitli sayısal verilerin, basamak sayısı, kesme ana birim sistemine ters düşen birimlerin çevrilmesi ve uyarı verilmesi gibi durumlar kontrol edilmiştir. İnç ve metrik birimdeki kesme dosyaları ve makina kontrol verilerinde karşılıkları Ek 26'da verilmiştir.

PPRINT komutu, tezgah kontrol veri dosyasında kullanıcıya (genelde operatöre) mesaj vermek için kullanılmaktadır. Kesme dosyaları içinde şu formda bulunurlar;

PPRINT / katar

Form içinde belirtilen katar, operatöre mesaj verilmesi istenen herhangi bir bilgiyi temsil etmektedir. ProEngineer, PPRINT komutu için, Ek 25'te gösterilen kendine has tablosunu kullanır. Kullanıcı bu tablodan, takım ismi ve numarası, operasyon ismi ve sırası gibi aslında çok önemli bilgileri taşıyan bilgileri kesme dosyasına ekleyebilir. Ancak genelde kullanıcılar bu önemli özelliği kullanmaktan

çekinirler çünkü, bu komut makina kontrol veri dosyasının hacminin artmasına neden olacaktır. Uygulamada sadece takım ismi bilgisini veren PPRINT komutu, makina kontrol veri dosyasına aktarılacak şekilde işlenmiştir. Yazara göre PPRINT bilgileri içinde en önemli bilgi takım ismi bilgisidir. Diğer tüm bilgiler makina kontrol veri dosyasına aktarılmadan arayüz içindeki ayrı bir bölümde bilgi şeklinde kullanıcıya sunulur. Uygulamadaki bu avantajdan dolayı, tüm PPRINT tablo bilgileri, kesme dosyasına eklenebilir. Böylece kesme dosyasından, takım yolu ile ilgili tüm bilgiler alınmış olur.

CUTTER komutu, 6000 sınıfı kaydı içinde tanımlıdır ve kesme dosyasında takımın geometrisini tanımlar. ProEngineer kesme dosyasında, bu komutun önünde “\$” karakteri olduğu için yorum cümlesinden sayılarak postprosesör tarafından dikkate alınmaz.

\$\$-> CUTTER / kesici takım çapı

Uygulamada, Postprosesör Generator yazılımlarının kullanamadığı bu komut kullanılarak, PPRINT komutu kullanılsa bile, hangi satırda hangi çapta takımın kullanıldığına dair bilgi verilir. Böylelikle kullanıcı (operatör) hangi işlem için hangi kesici takımın kullanıldığını rahatça görür ve buna göre ayarlamalar yapabilir.

FINI komutu, kesme dosyasının sonuna geldiğini gösterir. Uygulamada, bu komutla karşılaşıldığında tezgahta operasyonu bittiren “M30” kod satırı dosya sonuna eklenir. Ayrıca bu kod satırından sonra, dosya sonuna, DNC protokolü gereğince, veri alış verişinin sonuna geldiğini gösteren “\$” işareti konulur. Makina kontrol veri dosyası sonuna bu karakter eklendikten sonra, dosyaya yazılan birtakım bilgiler, CNC belleğine aktarılmazlar.

3.9. Özel Postprosesörün Optimizasyonu

Optimizasyon aşaması, postprosesörün kullanıcıya birtakım kolaylıklar sağlayacak şekilde geliştirilmesi sürecidir. Normalde buraya kadar anlatılan konular, bir postprosesörün işlevini yerine getirmesi için yeterlidir.

Makina kontrol veri dosyasına, isteğe bağlı birtakım opsiyonlar eklenmesinin yanı sıra, program için bir arayüz tasarlanması kullanıcıya büyük rahatlıklar sağlayacaktır. Örneğin uygulamada, C++ Builder yerine herhangi bir C derleyicisi kullanılsaydı, Windows işletim sisteminin verdiği kolaylık ve görsellikten yararlanılamayacaktı.

Uygulamada arayüz için birkaç form kullanılsa da, bunlar arasında en önemli olanı, Şekil 3.28’de gösterilen formdur.

GYTE Deckel-Maho Post Processor

Kaydet Tamam

1
GOTO / 36.2243086615, -27.6819324475, -28.0000043416, \$I
-0.3603506918, 0.3630218017, 0.8592802514I
GOTO / 35.0094849260, -29.1053022912, -27.9999982611, \$I
-0.3439600833, 0.3811788466, 0.8581341084I

2
N893 G1 X:0.000 Y56.150 Z-44.743 A29.690 C-90.000
N894 G1 X-0.224 Y56.149 A29.691 C-91.470
N895 G1 X-0.695 Y56.142 Z-44.748 A29.701 C-94.560
N896 G1 X-1.163 Y56.128 Z-44.756 A29.722 C-97.640
N897 G1 X-1.627 Y56.106 Z-44.769 A29.752 C-100.719
N898 G1 X-2.085 Y56.077 Z-44.786 A29.793 C-103.790

3
\$\$* Pro/CLfile Version 2000i - 1999260I
İslenecek parçanın adı: TEZ4U
3 eksen için sıfır noktası G55.. atandı..
Kesme dosyası birimi:mm : G71 atandı..
satır5 dönme devri(S)=6500
10 satırında ilerleme(F)500.000 MMPM

4
4 satırdaki 1.takimin numarası:101 takimin çapı:10.000000

Şekil 3.28. Postprosesör Programındaki Ana Form Düzeni

Form üzerinde 1 ile gösterilen alan kaynak dosyanın (kesme dosyasının) gösterildiği alandır. 2 ile gösterilen alan ise makina kod veri dosyasının aktığı alandır. Uygulamada bu iki alan eşzamanlı olarak çalışır. Bir başka deyişle kaynak dosya 1. alanda satır satır akarken, 2. alanda bu satırların makina kodlarına dönüşümü, o anda görülebilir. 3. alanda ise kesme dosyası ile ilgili birtakım bilgiler (kesme dosyası yolu, işlenen parçanın adı, hangi satırda hangi ilerleme hızı ve dönme devri olduğu gibi) yine kesme dosyasında eşzamanlı olarak akmaktadır. 4. bölüm ise kesme dosyasında kullanılan takımları ve çaplarını gösterir. Takım numarası değiştirilecek kesici takımlar, bu alandaki bilgiler vasıtasıyla bulunur ve ilgili değişiklik, 2.alanda yapılır.

Bütün gerekli düzeltmeler yapıldıktan sonra, 2. alanda bulunan makina kontrol veri dosyası, form üzerindeki “kaydet” düğmesi vasıtasıyla, dosya uzantısı “gyte” olacak şekilde kaydedilir. İstenildiğinde hiçbir kayıt işlemi yapılmadan “tamam” düğmesi ile program kapatılabilir.

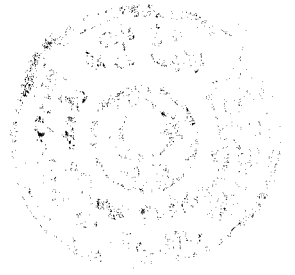
4. SONUÇ/ÖNERİLER

Bu çalışmada C++ Builder, homojen transformasyon matrisleri ve ters kinematik kullanılarak, eğilmeli döner tip tablalı bir CNC tezgah için Postprosesör yazılımı geliştirilmiştir.

Yazılımın geliştirilme süreci ilk etapta uzun sürebilir çünkü, bu süreç, bir programlama dili ve CAD/CAM sistemine hakim olmanın yanı sıra, tezgah özelliklerinin dolayısıyla 5 eksen mantığının iyi kavranmasını gerektirir. Bir kere herhangi bir CAM sistemi için postprosesör geliştirildikten sonra diğer CAM sistemleri için çok daha kısa sürede Postprosesör hazırlanabilir. Uygulamada ProEngineer için bir postprosesör hazırlanmış ve tezgahta test ederek başarılı sonuçlar elde edilmiştir. Ayrıca daha sonra Unigraphics yazılımı kesme dosyaları içinde postprosesör hazırlanmış ve tezgahta başarıyla test edilmiştir. Ancak sadece lineer hareketleri kapsayan bu postprosesörden, tez içinde bahsedilmemiştir.

Uygulama için geliştirilen Postprosesör sonuçları, ayrıca ICAM Postprosesör Generator paket programının verdiği sonuçlarla mukayese edilmiş ve doğru rotasyon analizi açısından birtakım farklar görülse de, sonuçta aynı kesici konumlanması sağlanmıştır.

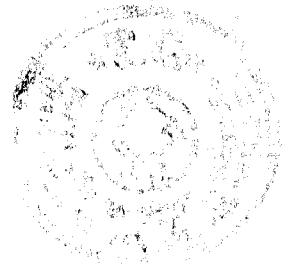
Sonuç olarak; son kullanıcılar, çok pahalı olan Generator programlarını kullanmak yerine, kendilerinin de hazırlayabileceği özel postprosesörleri geliştirebilirler.



EKLER

Ek 1

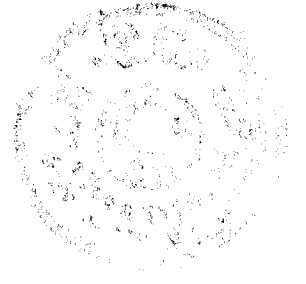
EIA Standartlarında G Kodları (Koren,1983)



G00	Noktadan noktaya hızlı hareket
G01	Doğrusal enterpolasyon
G02	Dairesel enterpolasyon saat yönünde
G03	Dairesel enterpolasyon saat aksi yönde
G04	Oyalanma (saniye veya devir sayısı türünden)
G08	Hızlandırma (belirli bir oranda ilerlemeyi hızlandırır)
G09	Yavaşlatma (belirli bir oranda ilerlemeyi yavaşlatır)
G10	G01'e benzer ancak verilen tüm ölçüler 10 ile çarpılır.
G11	G01'e benzer ancak verilen tüm ölçüler 10 ile bölünür.
G13	Eşzamanlı hareket yapamayan tezgahlarda belirli bir eksende işlem yapılmasını sağlar.
G14	
G15	
G16	
G17	XY düzleminde çalışma
G18	ZX düzleminde çalışma
G19	YZ düzleminde çalışma
G20	G10'a benzer, saat yönünde dairesel hareket
G21	G11'a benzer, saat yönünde dairesel hareket
G30	G10'a benzer, saat yönü aksinde dairesel hareket
G31	G11'e benzer, saat yönü aksinde dairesel hareket
G40	Takım yarıçap telafisi iptali
G41	Soldan takım yarıçapı telafisi
G42	Sağdan takım yarıçapı telafisi
G43 ~ G49	Takım boyu telafisi (düzeltmeleri)
G60 ~G79	Noktadan noktaya sistemler için ayrılan kod
G81~G89	Kapalı döngüler (delme, raybalama kılavuz salma vs.)
G90	Mutlak ölçülendirme
G91	Zincirleme ölçülendirme

Ek 2

EIA Standartlarında M Kodları (Koren,1983)



M00	Program durdurma
M01	Opsiyonel (planlı) program durdurma
M02	Program sonu
M03	Tezgah milini saat yönünde döndürür.
M04	Tezgah milini saatin aksi yönünde döndürür.
M05	Tezgah milini durdurur.
M06	Takım değiştirme komutu
M07	Sıvı soğutma sistemini devreye sokar.
M08	Hava soğutma sistemini devreye sokar.
M09	Soğutma sistemini durdurur.
M10	Otomatik olarak tutucu çenelerini sıkıştırır.
M11	Tutucu çenelerini gevşetir.
M13	Saat yönünde, tezgah milini ve soğutma sıvısını devreye sokar.
M14	Saat yönü aksine, tezgah milini ve soğutma sıvısını devreye sokar.
M19	Belirlenen bir açıda tezgah milini durmasını sağlar.
M30	Program sonu
M32	Sabit kesme hızı sağlar.
M40	Kullanılıyorsa, dişli değiştirir.

[illegible]

Ek 4

ISO 3592-1978 ve ISO 4343-1978 Kesme Dosyasında Komut Kayıt Sınıfları

Kayıt Tipi	Kayıt İsmi	İçeriği
1000	Kayıt sıra numarası	Parça programında her bir kayıt için tanım ve sıra bilgilerini verir.
2000	Postprosesör komutları	Postprosesör tarafından işleme tabi tutulan ve çevrilen direktifleri içerir.
3000	Yüzey verileri	Parça geometrisi tanımlama
4000~5000	Takım pozisyonu	Kontrol ve parça yüzeylerine bağlı göre takım pozisyonu ve takım hareket vektörü bilgileri
6000	Tolerans veya kesici bilgileri	Takım yolu oluşturmada kullanılan bilgiler
7000	Başlangıç hareketi komutu	Başlangıç hareketinde belirli yüzeye göre, takım pozisyonu verilmesi
8000	Hareket bilgileri	Son takım hareketine göre takım yönünü gösteren bilgi
9000	Eksen şekli veya birim	3 eksen veya çok eksen bilgisi veya birimler
12000	Özel program parametreleri	Özel programlar için çeşitli komutlar için ayrılan sınıf
14000	FINI kaydı	Parça programını bitirir
15000	Parçasız takım yolu	Doğrusal olmayan takım yolu bilgisi
16000	İş parçası çevresi tanımlanması	İş parçası çevresini tanımlayan parametre yada veriler.
17000	Takım tanımlanması	Takım, materyal ve tezgah tanımlanması için ayrılan kayıt tipleri
18000	Materyal tanımlaması	
19000	Tezgah tanımlaması	
20000	Literal tipte postprosesör komutları	Sade ifadelerden oluşan postprosesör komutları
28000~ 32000	Tescil edilmiş özel komutlar	Özel amaçlar için ayrılan kayıt tipleridir.

EK 5



CATIA Yazılımı Kesme Dosyası Örneği.

```
$$=====
$$      GENERATED ON TUESDAY MAY 22, 2001      AT 14.08.55
$$=====
PPRINT MODEL = DWG_1J378_04-D04_202_CAM
PPRINT PART OP = 25
PPRINT MACHINE = STANDARD_5_AXIS_NC_MACHINE
$$*CATIA0
$$*AXS505
$$      .00000      -1.00000      .00000      8.00000
$$      -.10453      .00000      .99452      .00000
$$      -.99452      .00000      -.10453      .00000
PPRINT MFG_START_MACRO
PARTNO PART TO BE MACHINED
COOLNT/ON
CUTCOM/OFF
MULTAX
PPRINT OPERATION NUMBER : 1
FEDRAT/.08,IPR
PPRINT OPERATION NUMBER : 2
RAPID
FROM /      .00000,      .00000,      30.00000,      .622840,      .000000,      .782349PT      1
PPRINT OPERATION NUMBER : 3
PPRINT TOOL = 3fm
CUTTER/ 3.000000,      .000000,      1.500000,      .000000,      .000000,$
      .000000,      5.000000
TOOLNO/1,3.00000,.00000,,5.00000,5.00000,.00000,5.00000,2,.0000,NOTE
TPRINT/3fm
LOADTL/1
PPRINT OPERATION NUMBER : 4
PREFUN/54
PPRINT OPERATION NUMBER : 5
SPINDL/910,CLW,RANGE,2
PPRINT OPERATION NUMBER : 6
SELECT/01
PPRINT OPERATION NUMBER : 7
MULTAX/ON
PPRINT OPERATION NUMBER : 8
SPINDL/910.00000,RPM,CLW
FEDRAT/.08000,MMPR
GOTO /      .00000,      .00000,      30.00000,      .000000,      .000000,      1.000000PT      2
PPRINT OPERATION NUMBER : 9
PPRINT OPERATION NUMBER : 10
RAPID
GOTO / 12.95522,      .69946,      34.03697,      .622840,      .000000,      .782349PT      3
RAPID
GOTO /      .49842,      .69946,      18.38998,      .622840,      .000000,      .782349PT      4
GOTO / 1.67195,      .69946,      17.45572,      .622840,      .000000,      .782349PT      5
GOTO / -2.26766,      15.04356,      16.16581,      -.622843,      .000000,      .782347PT      135
GOTO / -1.09413,      15.04356,      15.23155,      -.622843,      .000000,      .782347PT      136
GOTO / -1.09413,      13.54356,      15.23155,      -.622843,      .000000,      .782347PT      137
GOTO / -1.09413,      2.19946,      15.23155,      -.622843,      .000000,      .782347PT      138
GOTO / -1.09413,      .69946,      15.23155,      -.622843,      .000000,      .782347PT      139
RAPID
GOTO / -1.48531,      .69946,      15.54297,      -.622843,      .000000,      .782347PT      140
RAPID
GOTO / 1.62889,      .69946,      19.45471,      -.622843,      .000000,      .782347PT      141
PPRINT OPERATION NUMBER : 33
RAPID
GOTO / 1.03340,      15.04356,      19.84285,      -.622843,      .000000,      .782347PT      142
RAPID
GOTO / -2.08080,      15.04356,      15.93110,      -.622843,      .000000,      .782347PT      143
GOTO / -.90728,      15.04356,      14.99684,      -.622843,      .000000,      .782347PT      144
GOTO / -.90728,      13.54356,      14.99684,      -.622843,      .000000,      .782347PT      145
GOTO / -.90728,      2.19946,      14.99684,      -.622843,      .000000,      .782347PT      146
GOTO / -.90728,      .69946,      14.99684,      -.622843,      .000000,      .782347PT      147
RAPID
```

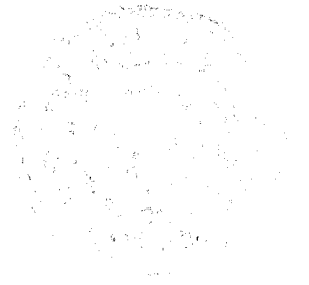
GOTO / -1.29845, .69946, 15.30826, -.622843, .000000, .782347PT 148
RAPID
GOTO / 11.15835, .69946, 30.95525, -.622843, .000000, .782347PT 149

•
•
•

PPRINT OPERATION NUMBER : 34
FEDRAT/.08000,MMPR
GOTO / .00000, .00000, 30.00000, -.609108, -.208849, .765095PT 150
PPRINT OPERATION NUMBER : 35
PLABEL/OPTION,37,TO,1
PPRINT OPERATION NUMBER : 36
SPINDL/650.00000,RPM,CLW
FEDRAT/.08000,MMPR
GOTO / .00000, .00000, 30.00000, .000000, .000000, 1.000000PT 151
PPRINT OPERATION NUMBER : 37
PLABEL/OPTION,37,TO,0
PPRINT OPERATION NUMBER : 38
MULTAX/OFF
PPRINT OPERATION NUMBER : 39
INSERT G91G28X0Y0Z0\$
PPRINT MFG_END_MACRO
SPINDL/OFF
REWIND/0
END



Ek 6



CIMATRON Yazılımı Kesme Dosyası Örneği.

\$\$p01355177l|]>pdE'l4\$n]OA9/,!|}&'

UNIT/INCH

TPCIM/"TP_MODEL"

NEXT_TOOLPATH

PRCIM/1,"PROFILE ","FACE TOP"

PRCIM2D/0.0,0.0039,2.5,0.0,0.0,0.0

NEXT_PROCEDURE

PPRINT/PROFILE

PPRINT/FACE TOP

UCS/0.0,0.0,0.0,1,0,0,0,1

UCS_MULTI/1,0.0,0.0,0.0,1,0,0,0,1

CUTTER/5.0,0.0,0.0,0.0,0.0,0.5906

LOADTL/2,ADJUST,2,LENGTH,3.937,AUTO_PLN,FIRST

FROM/0.0,0.0,0.1

GOHOME

TOOLINF/"5" FACEMILL",2,5.0,0.0,1.1811,0.5906,"5" FACEMILL"

RAPID

SPINDL/CLW,RPM,2500

COOLNT/FLOOD

GOTO/8.5,0.0,0.1,0.0,0.0,1.

RAPID

GOTO/8.5,0.0,0.02

FEDRAT/6,FPM

GOTO/8.5,0.0,0.0

SETUP/START,1

FEDRAT/20,FPM

GOTO/5.5,0.0,0.0

GOTO/-5.5,0.0,0.0

GOTO/-8.5,0.0,0.0

GOTO/-8.5,2.0,0.0

•

•

•

PRCIM/9,"DRILL ", "TAP 1/4-20"

NEXT_PROCEDURE

PPRINT/DRILL

PPRINT/TAP 1/4-20

MULTAX/ON

CUTTER/0.25,0,0,0,0,0.65

LOADTL/8,ADJUST,8,LENGTH,3.937,AUTO_PLN,LAST

TOOLINF/"1/4-20 CUT TAP",8,0.25,0.0,1.1811,0.65,"1/4-20 CUT TAP"

MULTAX/ON

RAPID

GOTO/-3.5,-2.0,0.2,0.0,0.0,1.

CYCLRET/INIT

CYCLE/TAP,1.0,FPM,25,0.2,0.2

SPINDL/CLW,RPM,500

COOLNT/FLOOD

GOTO/-3.5,-2.0,0.0,0.0,0.0,1.

GOTO/-3.5,2.0,0.0,0.0,0.0,1.

GOTO/3.5,2.0,0.0,0.0,0.0,1.

GOTO/3.5,-2.0,0.0,0.0,0.0,1.

CYCLE/OFF

RAPID

GOTO/3.5,-2.0,0.2,0.0,0.0,1.

MULTAX/OFF

\$\$ STOP

\$\$ END

Ek 7

EUCLID Yazılımı Kesme Dosyası Örneği.

```
UNIT/MM
PARTNO/1234
MACHIN/MILL_3AXIS
PPRINT/END MILL
CUTTER/ 10.00000, 0.0, 5.0, 0.0, 0.0, 0.0, 80.00000
LOADTL/1, LENGTH, 0.00000
SPINDL/ 1200.00000, CLW
COOLNT/ON
ORIGIN/ 0.00000, 0.00000, 0.00000
TLAXIS/ 0.00000, 0.00000, 1.00000
RAPID
GOTO / -145.0000, 95.0000, 53.879
RAPID
GOTO / -145.0000, 95.0000, 31.879
FEDRAT/MMPM, 300.00000
GOTO / -145.000, -95.000, 31.879
GOTO / -136.212, -95.000, 31.879
GOTO / -136.212, 95.000, 31.879
GOTO / -127.424, 95.000, 31.879
GOTO / -127.424, -95.000, 31.879
GOTO / -118.636, -95.000, 31.879
GOTO / 101.061, 95.000, 31.879
GOTO / 101.061, -95.000, 31.879
GOTO / 109.848, -95.000, 31.879
GOTO / 109.848, 95.000, 31.879
GOTO / 118.636, 95.000, 31.879
GOTO / 118.636, -95.000, 31.879
GOTO / 127.424, -95.000, 31.879
GOTO / 127.424, 95.000, 31.879
CUTCOM/LEFT, XYPLAN
GOTO / -59.887, 35.388, 66.559
AUTOPS
INDIRV/ -0.99994, 0.01111, 0.00000
GOFWD / (CIRCLE/ -59.9978, 25.38889, 66.55948,
10.00000 ) $
, ON , $
(LINE/ -59.99784, 25.38889, 66.55948, $
-69.99722, 25.49999, 66.55948 )
AUTOPS
INDIRV/ -0.01111, -0.99994, 0.00000
GOFWD / (CIRCLE/ -25.0000, 25.00000, 66.55948,
45.00000 ) $
, ON , $
(LINE/ -25.00000, 25.00000, 66.55948, $
19.99722, 24.50001, 66.55948 )
AUTOPS
INDIRV/ 0.01111, -0.99994, 0.00000
GOFWD / (CIRCLE/ -59.9978, 24.61111, 66.55948,
10.00000 ) $
, ON , $
(LINE/ -59.99784, 24.61111, 66.55948, $
-59.88674, 14.61172, 66.55948 )
```

CUTCOM/OFF

GOTO / -59.000, 25.000, 66.559
RAPID

•

•

•

CUTTER/ 6.00000, 0.0, 3.0, 0.0, 30.0, 0.0, 60.00000

LOADTL/2, LENGTH, 0.00000

SPINDL/ 2500.00000, CLW

COOLNT/ON

ORIGIN/ 0.00000, 0.00000, 0.00000

TLAXIS/ 0.00000, 0.00000, 1.00000

RAPID

GOTO / -175.000, 125.000, 0.00000

PPRINT/DRILLING

CYCLE /DRILL, 10.00000, MMPM, 300.00000, DWELL, 500.00000,

\$

2.00000, 2.00000

GOTO / -175.000, 125.000, 0.000

GOTO / -175.000, -125.000, 0.000

GOTO / 175.000, -125.000, 0.000

GOTO / 175.000, 125.000, 0.000

CYCLE/OFF

RAPID

GOTO / 175.000, 125.000, 53.879

CYCLE /DRILL, 10.00000, MMPM, 300.00000, DWELL, 500.00000,

\$

2.00000, 2.00000

GOTO / -175.000, 125.000, 0.000

CYCLE/OFF

COOLNT/OFF

SPINDL/OFF

PPRINT/DEEP-DRILL

CUTTER/ 6.00000, 0.0, 3.0, 0.0, 30.0, 0.0, 60.00000

LOADTL/3, LENGTH, 0.00000

SPINDL/ 2500.00000, CLW

COOLNT/ON

ORIGIN/ 0.00000, 0.00000, 0.00000

TLAXIS/ 0.00000, 0.00000, 1.00000

RAPID

GOTO / -175.000, 125.000, 0.000

PPRINT/DEEP DRILLING

CYCLE /DEEP, 10.00000, INCR, 2.5, MMPM, 300.00000, DWELL,

500.00000, \$

2.00000, 2.00000

GOTO / -175.000, 125.000, 0.000

GOTO / -175.000, -125.000, 0.000

GOTO / 175.000, -125.000, 0.000

GOTO / 175.000, 125.000, 0.000

CYCLE/OFF

RAPID

GOTO / 175.000, 125.000, 53.879

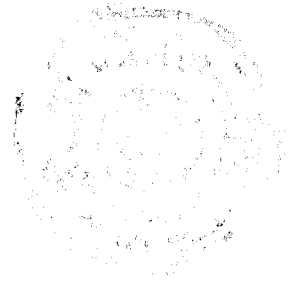
COOLNT/OFF

SPINDL/OFF

FINI

REMARK/\$\$ This is the end.

Ek 8



MASTERCAM Yazılımı Kesme Dosyası Örneği.

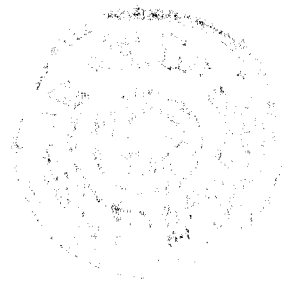
```
1050
8 33 11 01 2001 12 30 28 E:\MCAM8\MILL\MC8\SAMPLES\2D
MACHINING\CONTOUR RAMP ZIGZAG AND REMACHINING.MC8
1011
0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
1012
2 0 0 0 0 0 0 0 0 0
1013
0 1. 0. 1 1 0. 0. 0. 2 C:\MCAM81\MILL\TOOLS\FLATMILL.MC8
1014
1. 0. 0. 0. 1. 0. 0. 0. 1.
1016
3 10 1 1 0. 0. 0. 41 1 0 4 0 0. -1 0 3
1017
1. 0. 0. 0. 1. 0. 0. 0. 1.
1025
0 0 0 0 0 0 0 0 0 0
1020
7.25696 3.00413 0.75 0.30412 0.45514 0. 0 0. 0. 0. 0 5000 NONE
20001
1" FLAT ENDMILL
20002

20003

20004
3 10 1 0 1. 0. 0. 180. 43 3 6.4176 6.4176 6.4176 534 1 4
20006
0 50. 50. 25. 25. 0. 0. 0.
20007
0. 2. 3. 2.5 1. 2. 1. 0 100. 100. 0
1001
0 100 2 3 43 3 0 534 6.4176 1 4.632601 -1.287199 0.25 0. 0. 0. 0 0.
0
0 4.632601 -1.287199 0.25 -2. 0
0
0 4.632601 -1.287199 0.1 -2. 0
1
0 4.632601 -1.287199 -0.77 6.4176 0
3
0 0 3.932601 -0.587199 3.932601 -1.287199 -0.77 -1. 0
1
0 3.332643 -0.587199 -0.77 -1. 1000
1
0 3.095824 -0.990867 -0.77 -1. 0
2
0 0 2.815932 -1.216272 2.658497 -0.734302 -0.77 -1. 0
2
0 0 2.200328 -1.2316 2.482524 -0.195576 -0.77 -1. 0
2
0 0 1.896253 -0.998958 2.33358 -0.742392 -0.77 -1. 0
0 0.25 0. 1 1 0. 0. 0. 2 C:\MCAM81\MILL\TOOLS\FLATMILL.MC8
1014
```

1. 0. 0. 0. 1. 0. 0. 0. 1.
 1016
 4 10 1 1 0. 0. 0. 41 1 0 4 0 0. -1 0 1
 1017
 1. 0. 0. 0. 1. 0. 0. 0. 1.
 1025
 0 0 0 0 0 0 0 0 0 0
 1020
 7.25696 3.00413 0.75 0.30412 0.45514 0. 0 0. 0. 0. 0 5000 NONE
 1000
 0 100 2 1 41 1 0 2139 6.4176 1 2.055081 -0.603598 0.1 0. 0. 0. 0 0.
 0
 0 2.055081 -0.603598 0.1 -2. 0
 1
 0 2.055081 -0.603598 -0.7 6.4176 0
 1
 0 2.054646 -0.541099 -0.7 -1. 0
 3
 0 0 2.047774 -0.516146 2.004648 -0.541447 -0.7 -1. 1000
 1
 0 1.873986 -0.219917 -0.7 -1. 0
 3
 0 0 1.847082 -0.219833 1.860509 -0.227824 -0.7 -1. 0
 2
 0 0 1.683134 -0.451804 0.289516 0.707068 -0.7 -1. 0
 3
 0 0 1.672301 -0.475309 1.721579 -0.483773 -0.7 -1. 0
 1
 0 1.661721 -0.536908 -0.7 -1. 0
 0
 0 1.661721 -0.536908 0.1 -2. 0
 1011
 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 1012
 2 0 0 0 0 0 0 0 0 0
 1013
 0 0.25 0. 1 1 0. 0. 0. 2 C:\MCAM81\MILL\TOOLS\FLATMILL.MC8
 1014
 1. 0. 0. 0. 1. 0. 0. 0. 1.
 1016
 4 10 1 1 0. 0. 0. 41 1 0 4 0 0. -1 0 1
 1017
 .
 .
 .
 3
 0 0 1.847082 -0.219833 1.860509 -0.227824 -0.75 -1. 0
 2
 0 0 1.683134 -0.451804 0.289516 0.707068 -0.75 -1. 0
 3
 0 0 1.672301 -0.475309 1.721579 -0.483773 -0.75 -1. 0
 1
 0 1.661721 -0.536908 -0.75 -1. 0
 0
 0 1.661721 -0.536908 0.1 -2. 0
 1003
 0. 0. 0.

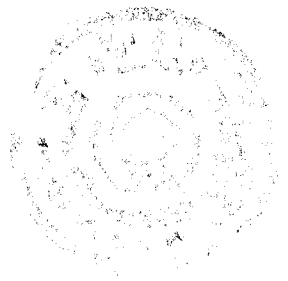
Ek 9



UNIGRAHICS Yazılımı Kesme Dosyası Örneği.

```
TOOL PATH/ROUGH, TOOL, BALLMILL, .750
TLDATA/MILL, 0.7500, 0.3750, 3.0000, 0.0000, 0.0000
MSYS/0.0000, 0.0000, 0.0000, 1.0000000, 0.0000000, 0.0000000, 1.
0000000, 0.0000000
PAINT/PATH, DASH, 100.0000, 0.0000000, 0.0000000, 1.0000000
PAINT/SPEED, 10
PAINT/TOOL, FULL, 8
PAINT/COLOR, 4
FROM/0.0000, 0.0000, 3.0000, 0.0000000, 0.0000000, 1.0000000
LOAD/TOOL, 0
SPINDL/RPM, 4000, CLW
FEDRAT/IPM, 400.0000
GOTO/0.0424, -5.2266, 3.0000, -0.0035580, 0.4385885, 0.8986809
PAINT/COLOR, 6
GOTO/0.0547, -6.7370, -0.0949
PAINT/COLOR, 3
GOTO/0.0000, -6.7375, -0.0947, 0.0000000, 0.4386151, 0.8986750
GOTO/-0.3395, -6.7289, -0.0947, 0.0220998, 0.4380579, 0.8986751
GOTO/-0.4562, -6.7220, -0.0947, 0.0297013, 0.4376082, 0.8986751
GOTO/-0.5748, -6.7129, -0.0947, 0.0374189, 0.4370159, 0.8986751
GOTO/-0.6951, -6.7015, -0.0947, 0.0452516, 0.4362744, 0.8986751
GOTO/-0.8172, -6.6877, -0.0947, 0.0531982, 0.4353769, 0.8986751
GOTO/-0.9410, -6.6715, -0.0947, 0.0612570, 0.4343163, 0.8986751
GOTO/-1.0664, -6.6526, -0.0947, 0.0694262, 0.4330855, 0.8986751
GOTO/-1.1936, -6.6309, -0.0947, 0.0777033, 0.4316772, 0.8986751
GOTO/-1.3224, -6.6064, -0.0947, 0.0860859, 0.4300840, 0.8986751
GOTO/-1.4527, -6.5790, -0.0947, 0.0945710, 0.4282982, 0.8986751
GOTO/-1.5845, -6.5485, -0.0947, 0.1031550, 0.4263121, 0.8986752
GOTO/-1.7179, -6.5148, -0.0947, 0.1118343, 0.4241179, 0.8986752
GOTO/-1.8526, -6.4778, -0.0947, 0.1206046, 0.4217078, 0.8986752
GOTO/-1.9886, -6.4373, -0.0947, 0.1294610, 0.4190737, 0.8986752
GOTO/-2.1259, -6.3933, -0.0947, 0.1383984, 0.4162076, 0.8986752
GOTO/-2.2644, -6.3456, -0.0947, 0.1474110, 0.4131015, 0.8986752
GOTO/-2.4039, -6.2941, -0.0947, 0.1564925, 0.4097474, 0.8986752
GOTO/-2.5443, -6.2386, -0.0948, 0.1656362, 0.4061373, 0.8986752
GOTO/-2.6856, -6.1791, -0.0948, 0.1748345, 0.4022632, 0.8986753
GOTO/-2.8276, -6.1154, -0.0948, 0.1840796, 0.3981174, 0.8986753
GOTO/-2.9702, -6.0474, -0.0948, 0.1933627, 0.3936923, 0.8986753
GOTO/-3.1133, -5.9751, -0.0948, 0.2026748, 0.3889803, 0.8986753
GOTO/-3.2566, -5.8982, -0.0948, 0.2120059, 0.3839742, 0.8986753
GOTO/-3.4001, -5.8166, -0.0948, 0.2213456, 0.3786671, 0.8986753
GOTO/-3.5435, -5.7304, -0.0948, 0.2306828, 0.3730524, 0.8986753
GOTO/-3.6867, -5.6393, -0.0948, 0.2400056, 0.3671238, 0.8986754
GOTO/-3.8295, -5.5433, -0.0948, 0.2493018, 0.3608756, 0.8986754
GOTO/-3.9717, -5.4424, -0.0948, 0.2585583, 0.3543023, 0.8986754
GOTO/-4.1130, -5.3363, -0.0948, 0.2677614, 0.3473994, 0.8986754
GOTO/-4.2534, -5.2252, -0.0948, 0.2768970, 0.3401626, 0.8986754
GOTO/-4.3924, -5.1088, -0.0948, 0.2859501, 0.3325884, 0.8986754
GOTO/-4.5300, -4.9873, -0.0948, 0.2949054, 0.3246741, 0.8986754
GOTO/-4.6658, -4.8604, -0.0948, 0.3037471, 0.3164177, 0.8986755
GOTO/-4.7996, -4.7283, -0.0948, 0.3124586, 0.3078181, 0.8986755
```

GOTO/-4.9312,-4.5910,-0.0948,0.3210234,0.2988751,0.8986755
GOTO/-5.0602,-4.4483,-0.0948,0.3294241,0.2895895,0.8986755
GOTO/-5.1865,-4.3005,-0.0948,0.3376433,0.2799630,0.8986755
GOTO/-5.3097,-4.1474,-0.0948,0.3456633,0.2699984,0.8986755

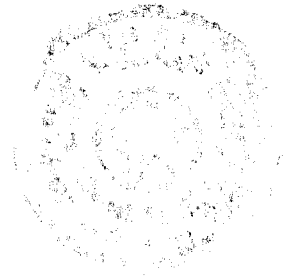


-
-
-

CIRCLE/0.0006,0.0005,-
1.6456,0.0000000,0.0000000,1.0000000,0.1719,0.0020,0.8333,0.7500,0.3
750
GOTO/0.1567,-0.0715,-1.6456
CIRCLE/0.0019,-0.0027,-
1.6456,0.0000000,0.0000000,1.0000000,0.1694,0.0020,0.8333,0.7500,0.3
750
GOTO/0.0004,-0.1721,-1.6456
PAINT/COLOR,7
GOTO/0.0004,-0.1721,3.0000
PAINT/COLOR,4
GOHOME/0.0000,0.0000,3.0000
SPINDL/OFF
REWIND
FINI
PAINT/SPEED,10
PAINT/TOOL,NOMORE
END-OF-PATH



Ek 10



ProEngineer Yazılımı Kesme Dosyası Örneği.

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ4U
PARTNO / TEZ4U
$$-> FEATNO / 107
MACHIN / MILL, 01
UNITS / MM
LOADTL / 101
$$-> CUTTER / 10.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 6500.000000, CLW
RAPID
GOTO / 131.6656161617, 100.0000000000, 120.0000000000
RAPID
GOTO / 131.6656161617, 100.0000000000, 105.0000000000
FEDRAT / 750.000000, MMPM
GOTO / 131.6656161617, 100.0000000000, 96.0000000000
GOTO / 131.6467767539, 99.0027055566, 96.0000000000
GOTO / 131.5523407162, 97.5443587136, 96.0000000000
GOTO / 131.3790632406, 96.1000000991, 96.0000000000
CIRCLE / 0.0000000000, -0.0000000000, 0.0000000000, $
0.0000000000, 0.0000000000, 1.0000000000, 25.6852420000
GOTO / 0.0000000000, 25.6852420000, 0.0000000000
GOTO / 0.0000000000, 25.6852420000, 10.0000000000
RAPID
GOTO / 42.7360080000, 0.0000000000, 10.0000000000
FEDRAT / 500.000000, IPM
GOTO / 42.7360080000, 0.0000000000, 0.0000000000
CIRCLE / -0.0000000000, 0.0000000000, 0.0000000000, $
0.0000000000, 0.0000000000, 1.0000000000, 42.7360080000
GOTO / -42.7360080000, 0.0000000000, 0.0000000000
GOTO / -42.7360080000, 0.0000000000, 10.0000000000
SPINDL / OFF
$$-> END /
$$-> FEATNO / 11
LOADTL / 1
$$-> CUTTER / 10.000000
SPINDL / RPM, 6500.000000, CLW
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
MULTAX / ON
RAPID
GOTO / 20.0000000000, 100.0000000000, 151.0489618531, $
-0.7913295320, 0.0000000000, 0.6113898689
RAPID
GOTO / 78.1600126348, 100.0000000000, 106.1138986894, $
```

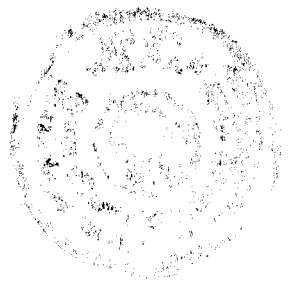
-0.7913295320, 0.0000000000, 0.6113898689
FEDRAT / 650.000000, MMPM

.

.

.

GOTO / 128.1022874873, 64.6306610629, 72.0000000000, \$
-0.2515916177, 0.4597295103, 0.8516750761
GOTO / 68.5015972616, 173.5380896198, 273.7572081451, \$
-0.2515916177, 0.4597295103, 0.8516750761
SPINDL / OFF
\$\$-> END /
FINI



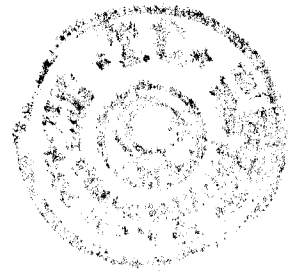
Ek 11



DECKEL MAHO DMU 60P Üniversal İşleme Merkezi Teknik Özellikleri.

ÇALIŞMA SAHASI	
X eksen (boyuna)	600 mm
Y eksen (enine)	700 mm
Z eksen (dikey)	550 mm
PARÇA BAĞLAMA TERTİBATLARI (NC Eğimli Döner Tabla)	
Bağlama yüzeyi	Ø 630 mm
Dönüş aralığı	360°
Eğilme aralığı (tezgah miline doğru)	+45° / -30°
Taşıma kapasitesi	500 kg
ANA TAHRİK	
Devir sayısı (standart)	20 – 12000 dev/dak
Max. Döndürme momenti	130 Nm
Takım tutucu	ISO 40. DIN 69871
İLERLEME TAHRİKLERİ	
İlerleme hızı	1 – 10000 mm/dak
Hızlı hareket	24 m/dak
İlerleme kuvveti (X / Y / Z)	10 / 10 / 10kN
TEZGAH AĞIRLIĞI	
NC döner tabla ve 30 takımlı magazinle birlikte	7800 kg.
CNC KUMANDA	
Standart	DECKEL MAHO MillPlus

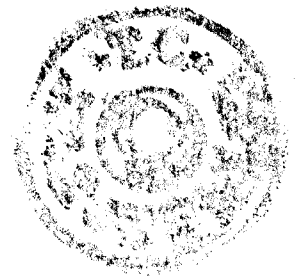
Ek 12



ProEngineer kesme dosyasında, 10 mm yarıçapında 270°'lik bir yay üzerinde hareket eden kesici takım için POINTS_ONLY formunda örnek çıktı.

```
$$*          Pro/CLfile  Version 2000i2 - 2000370
$$-> MFGNO / CIRCLE
PARTNO / CIRCLE
$$-> FEATNO / 39
MACHIN / GEBZE1, 25
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / MM
LOADTL / 102
$$-> CUTTER / 25.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2500.000000, CLW
RAPID
GOTO / -5.0000000000, 0.0000000000, 10.0000000000
RAPID
GOTO / -5.0000000000, 0.0000000000, 5.0000000000
FEDRAT / 200.000000, MMPM
GOTO / -5.0000000000, 0.0000000000, 0.0000000000
GOTO / -4.9600533049, 0.6307703327, 0.0000000000
GOTO / -4.8408515150, 1.2514617893, 0.0000000000
GOTO / -4.6442993172, 1.8521565409, 0.0000000000
GOTO / -4.3735373560, 2.4232562795, 0.0000000000
GOTO / -4.0328920494, 2.9556355861, 0.0000000000
GOTO / -3.6278064592, 3.4407877433, 0.0000000000
GOTO / -3.1647533176, 3.8709606610, 0.0000000000
GOTO / -2.6511316016, 4.2392807445, 0.0000000000
GOTO / -2.0951483074, 4.5398627259, 0.0000000000
GOTO / -1.5056873129, 4.7679037024, 0.0000000000
GOTO / -0.8921674256, 4.9197598808, 0.0000000000
GOTO / -0.2643918822, 4.9930048000, 0.0000000000
GOTO / 0.3676082939, 4.9864681030, 0.0000000000
GOTO / 0.9937345755, 4.9002542376, 0.0000000000
GOTO / 1.6039822922, 4.7357407875, -0.0000000000
GOTO / 2.1886004923, 4.4955564600, 0.0000000000
GOTO / 2.7382477497, 4.1835390833, 0.0000000000
GOTO / 3.2441414280, 3.8046742824, 0.0000000000
GOTO / 3.6981980148, 3.3650158162, 0.0000000000
GOTO / 4.0931622863, 2.8715888456, 0.0000000000
GOTO / 4.4227232354, 2.3322776814, 0.0000000000
GOTO / 4.6816149139, 1.7556998030, 0.0000000000
GOTO / 4.8657005749, 1.1510681627, 0.0000000000
GOTO / 4.9720387730, 0.5280439748, 0.0000000000
GOTO / 5.0000000000, 0.0000000000, 0.0000000000
GOTO / 4.9600533049, -0.6307703327, 0.0000000000
GOTO / 4.8408515150, -1.2514617893, 0.0000000000
GOTO / 4.6442993172, -1.8521565409, 0.0000000000
GOTO / 4.3735373560, -2.4232562794, 0.0000000000
GOTO / 4.0328920495, -2.9556355860, 0.0000000000
```

GOTO / 3.6278064592, -3.4407877433, 0.0000000000
GOTO / 3.1647533177, -3.8709606609, 0.0000000000
GOTO / 2.6511316017, -4.2392807445, 0.0000000000
GOTO / 2.0951483074, -4.5398627259, 0.0000000000
GOTO / 1.5056873129, -4.7679037024, 0.0000000000
GOTO / 0.8921674256, -4.9197598808, 0.0000000000
GOTO / 0.2643918822, -4.9930048000, 0.0000000000
GOTO / -0.3676082939, -4.9864681030, 0.0000000000
GOTO / -0.9937345754, -4.9002542376, 0.0000000000
GOTO / -1.6039822922, -4.7357407875, 0.0000000000
GOTO / -2.1886004923, -4.4955564600, 0.0000000000
GOTO / -2.7382477497, -4.1835390832, 0.0000000000
GOTO / -3.2441414280, -3.8046742824, 0.0000000000
GOTO / -3.6981980149, -3.3650158161, 0.0000000000
GOTO / -4.0931622863, -2.8715888455, 0.0000000000
GOTO / -4.4227232354, -2.3322776813, 0.0000000000
GOTO / -4.6816149139, -1.7556998029, 0.0000000000
GOTO / -4.8657005749, -1.1510681626, 0.0000000000
GOTO / -4.9720387730, -0.5280439748, 0.0000000000
GOTO / -5.0000000000, 0.0000000000, 0.0000000000
GOTO / -5.0000000000, 0.0000000000, 10.0000000000
SPINDL / OFF
\$\$-> END /
FINI



Ek 13



ProEngineer kesme dosyasında, 10 mm yarıçapında 270°'lik bir yay üzerinde hareket eden kesici takım için POINTS_&_ARC formunda örnek çıktı.

```
$$*          Pro/CLfile  Version 2000i2 - 2000370
$$-> MFGNO / CIRCLE
PARTNO / CIRCLE
$$-> FEATNO / 39
MACHIN / GEBZE1, 25
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / MM
LOADTL / 102
$$-> CUTTER / 25.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2500.000000, CLW
RAPID
GOTO / -5.0000000000, 0.0000000000, 10.0000000000
RAPID
GOTO / -5.0000000000, 0.0000000000, 5.0000000000
FEDRAT / 200.000000, MMPM
GOTO / -5.0000000000, 0.0000000000, 0.0000000000
CIRCLE / -0.0000000000, 0.0000000000, -0.0000000000, $
-0.0000000000, 0.0000000000, -1.0000000000, 5.0000000000
GOTO / -4.9600533049, 0.6307703327, 0.0000000000
-4.8408515150, 1.2514617893, 0.0000000000
-4.6442993172, 1.8521565409, 0.0000000000
-4.3735373560, 2.4232562794, 0.0000000000
-4.0328920494, 2.9556355861, 0.0000000000
-3.6278064592, 3.4407877433, 0.0000000000
-3.1647533176, 3.8709606609, 0.0000000000
-2.6511316017, 4.2392807445, 0.0000000000
-2.0951483074, 4.5398627259, 0.0000000000
-1.5056873129, 4.7679037024, 0.0000000000
-0.8921674256, 4.9197598808, 0.0000000000
-0.2643918823, 4.9930048000, 0.0000000000
0.3676082939, 4.9864681030, 0.0000000000
0.9937345754, 4.9002542377, -0.0000000000
1.6039822921, 4.7357407875, -0.0000000000
2.1886004922, 4.4955564600, -0.0000000000
2.7382477497, 4.1835390833, -0.0000000000
3.2441414279, 3.8046742824, -0.0000000000
3.6981980148, 3.3650158162, -0.0000000000
4.0931622863, 2.8715888456, -0.0000000000
4.4227232354, 2.3322776814, -0.0000000000
4.6816149139, 1.7556998030, -0.0000000000
4.8657005749, 1.1510681627, -0.0000000000
4.9720387730, 0.5280439749, -0.0000000000
4.9989303644, -0.1034176576, -0.0000000000
4.9459456569, -0.7332268127, -0.0000000000
4.8139312762, -1.3513199726, -0.0000000000
4.6049966374, -1.9478208257, -0.0000000000
```

4.3224802401, -2.5131980770, -0.0000000000
3.9708963226, -3.0384177453, -0.0000000000
3.5558627313, -3.5150875148, -0.0000000000
3.0840111542, -3.9355908325, -0.0000000000
2.5628811558, -4.2932086114, -0.0000000000
2.0007997046, -4.5822265922, -0.0000000000
1.4067481190, -4.7980266495, -0.0000000000
0.7902185582, -4.9371605838, -0.0000000000
0.1610623495, -4.9974052187, -0.0000000000
-0.4706674227, -4.9777979245, -0.0000000000
-1.0948765517, -4.8786519999, -0.0000000000
-1.7015910007, -4.7015516658, -0.0000000000
-2.2811162750, -4.4493267513, -0.0000000000
-2.8241923268, -4.1260074772, -0.0000000000
-3.3221415186, -3.7367600579, -0.0000000000
-3.7670072807, -3.2878041528, -0.0000000000
-4.1516812463, -2.7863134837, -0.0000000000
-4.4700168339, -2.2403012085, -0.0000000000
-4.7169274617, -1.6584918815, -0.0000000000
-4.8884678242, -1.0501820468, -0.0000000000
-4.9818969333, -0.4250916912, -0.0000000000
-5.0000000000, 0.0000000000, 0.0000000000
GOTO / -5.0000000000, -0.0000000000, 10.0000000000
SPINDL / OFF
\$\$-> END /
FINI

Ek 14

ProEngineer kesme dosyasında, 10 mm yarıçapında 270°'lik bir yay üzerinde dairesel hareket eden kesici takım için APT formunda örnek çıktı ve tezgah kodlarındaki karşılığı.

```
$$*          Pro/CLfile  Version 2000i2 - 2000370
$$-> MFGNO / CIRCLE
PARTNO / CIRCLE
$$-> FEATNO / 39
MACHIN / GEBZE1, 25
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / MM
LOADTL / 102
$$-> CUTTER / 25.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2500.000000, CLW
RAPID
GOTO / -5.0000000000, 0.0000000000, 10.0000000000
RAPID
GOTO / -5.0000000000, 0.0000000000, 5.0000000000
FEDRAT / 200.000000, MMPM
GOTO / -5.0000000000, 0.0000000000, 0.0000000000
INDIRV / 0.0000000000, 31.4159265359, 0.0000000000
TLON
GOFWD / ( CIRCLE / -0.0000000000, 0.0000000000, -0.0000000000,$
5.0000000000 ), $
      ON, 2, INTOF, ( LINE / -0.0000000000, 0.0000000000, -
0.0000000000,$
          -5.0000000000, -0.0000000000, 0.0000000000 )
GOTO / -5.0000000000, -0.0000000000, 10.0000000000
SPINDL / OFF
$$-> END /
FINI

%PM999
N999
N1 G59
N2 G71
N3 G17
N4 M54
N5 T2 M06
N6 S2500 M13
N7 G0 X-100 Z50 Y0 A0 C0
N8 G2 X100 Y0 Z50 R100 F200
N9 G2 X-100 Y0 Z50 R100
N10 M30
$
```

Ek 15

10 mm yarıçapında 270°'lik bir yay üzerinde dairesel hareket eden kesici takım için, ProEngineer kesme dosyası ARC_ONLY formu örneği ve tezgah kodlarındaki karşılığı.

```
$$*          Pro/CLfile  Version 2000i2 - 2000370
$$-> MFGNO / CIRCLE
PARTNO / CIRCLE
$$-> FEATNO / 46
MACHIN / GEBZE1, 25
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / MM
LOADTL / 103
$$-> CUTTER / 2.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2500.000000, CLW
RAPID
GOTO / 0.0000000000, -2.5000000000, 10.0000000000
RAPID
GOTO / 0.0000000000, -2.5000000000, 4.0000000000
FEDRAT / 250.000000, MMPM
GOTO / 0.0000000000, -2.5000000000, 0.0000000000
CIRCLE / 0.0000000000, 0.0000000000, 0.0000000000, $
0.0000000000, 0.0000000000, -1.0000000000, 2.5000000000
GOTO / -1.7677669530, 1.7677669530, 0.0000000000
          2.5000000000, 0.0000000000, 0.0000000000
GOTO / 2.5000000000, 0.0000000000, 10.0000000000
SPINDL / OFF
$$-> END /
FINI
```

```
%PM999
N999
N1 G59
N2 G71
N3 G17
N4 M54
N5 T3 M06
N6 S2500 M13
N7 G0 X0 Y-2.5 Z4
N8 G1 X0 Y-2.5 Z0 F250
N9 G2 X-1.768 Y1.768 Z0 R2.5
N10 G2 X2.5 Y0 Z0 R2.5
N11 G1 X2.5 Y0 Z10
N12 M30
$
```

Ek 16

DRILL Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 5
MACHIN / MILL, 01
UNITS / MM
LOADTL / 101
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 780.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / DRILL, DEPTH, 40.000000, MMPM, 250.000000, CLEAR, 5.000000,
RAPTO,$
  3.000000, RETURN, 2.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / -30.0000000000, -0.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Tezgah kontrol verileri

```
%PM99
N99
N1 G59
N2 G71
N3 G17
N4 M54
N5 T1 M06
N6 S780 M13
N7 G0 X0 Y-30 A0 C0
N8 G81 Y5 Z-40 F250
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

Ek 17

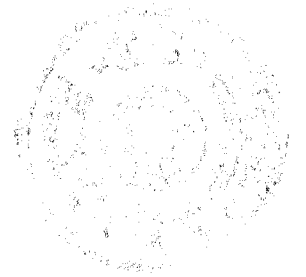
DEEP Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 29
MACHIN / MILL, 01
UNITS / MM
LOADTL / 105
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 850.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / DEEP, DEPTH, 40.000000, STEP, 2.000000, MMPPM, 250.000000,
CLEAR,$
    5.000000, RAPTO, 3.000000, RETURN, 2.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Makina kontrol veri dosya formu

```
%PM111
N111
N1 G59
N2 G71
N3 G17
N4 M54
N5 T5 M06
N6 S850 M13
N7 G0 X0 Y-30 A0 C0
N8 G83 Y5 Z-40 K2 F250
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

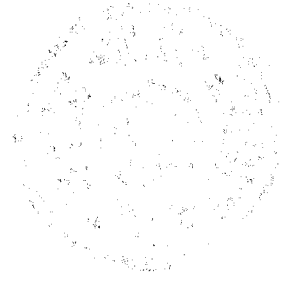
Ek 18



BRKCHP Komutu İçin Kesme Dosya Örneği.

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 16
MACHIN / MILL, M0001
UNITS / MM
LOADTL / 104
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 850.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / BRKCHP, DEPTH, 40.000000, STEP, 5.000000, MMPM, 200.000000,
CLEAR,$
    5.000000, RAPTO, 3.000000, RETURN, 8.000000, DWELL, 13.000000,
BACK,$
    7.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

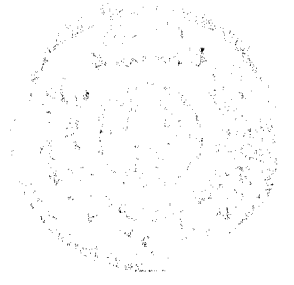
Ek 19



THRU Komutu İçin Kesme Dosyası Örneği.

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 19
MACHIN / MILL, 01
UNITS / MM
LOADTL / 104
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 850.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / THRU, DEPTH, 40.000000, MMPM, 200.000000, CLEAR, 5.000000,
RAPTO,$
  3.000000, RETURN, 8.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Ek 20



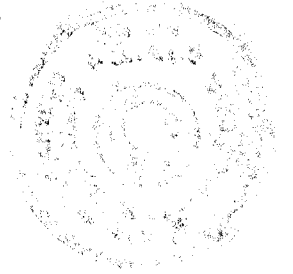
FACE Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ HOLES
PARTNO / TEZ HOLES
$$-> FEATNO / 13
MACHIN / MILL, 01
UNITS / MM
LOADTL / 103
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 580.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / FACE, DEPTH, 40.000000, MMPM, 250.000000, CLEAR, 4.000000,
RAPTO,$
  2.000000, RETURN, 1.000000, DWELL, 25.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Tezgah kontrol verileri

```
%PM99
N99
N1 G59
N2 G71
N3 G17
N4 M54
N5 T3 M06
N6 S580 M13
N7 G0 X0 Y-30 A0 C0
N8 G81 X25 Y4 Z-40 F250
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

Ek 21



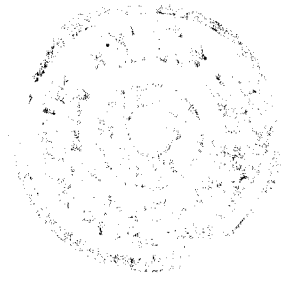
BORE Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$$          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 15
MACHIN / MILL, 01
UNITS / MM
LOADTL / 104
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 560.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / BORE, DEPTH, 40.000000, MMPM, 250.000000, CLEAR, 5.000000,
4.000000,$
  RAPTO, 3.000000, RETURN, 1.000000, ORIENT, 2.000000, DWELL,
22.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Tezgah kontrol verileri

```
%PM99
N99
N1 G59
N2 G71
N3 G17
N4 M54
N5 T4 M06
N6 S560 M13
N7 G0 X0 Y-30 A0 C0
N8 G86 X22 Y5 Z-40 F250
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

Ek 22



REAM Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 26
MACHIN / MILL, 01
UNITS / MM
LOADTL / 105
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 950.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / REAM, DEPTH, 40.000000, MMPM, 200.000000, CLEAR, 5.000000,
RAPTO,$
  3.000000, RETURN, 2.000000, DWELL, 5.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Tezgah kontrol verileri

```
%PM99
N99
N1 G59
N2 G71
N3 G17
N4 M54
N5 T5 M06
N6 S950 M13
N7 G0 X0 Y-30 A0 C0
N8 G85 X5 Y5 Z-40 F200
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

Ek 23



CSINK Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 31
MACHIN / MILL, 01
UNITS / MM
LOADTL / 101
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 850.000000, CLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / CSINK, DIAMET, 12.000000, TLANGL, 118.000000, MMPM,
250.000000,$
  CLEAR, 5.000000, RAPTO, 3.000000, RETURN, 2.000000, DWELL, 5.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Tezgah kontrol verileri

```
%PM99
N99
N1 G59
N2 G71
N3 G17
N4 M54
N5 T1 M06
N6 S850 M13
N7 G0 X0 Y-30 A0 C0
N8 G81 X5 Y5 Z-3.605 F250
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

Ek 24



TAP Komutu İçin Kesme Ve Makina Veri Dosya Örneği.

```
$$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 10
MACHIN / MILL, 01
UNITS / MM
LOADTL / 102
$$-> CUTTER / 5.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 100.000000, CCLW
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
CYCLE / TAP, DEPTH, 40.000000, MMPR, 0.800000, CLEAR, 2.000000,
RAPTO,$
  3.000000, RETURN, 2.000000
GOTO / 0.0000000000, -30.0000000000, 0.0000000000
CYCLE / OFF
RAPID
GOTO / 0.0000000000, -30.0000000000, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Tezgah kontrol verileri

```
%PM2324
N2324
N1 G59
N2 G71
N3 G17
N4 M54
N5 T2 M06
N6 S100 M13
N7 G0 X0 Y-30 A0 C0
N8 G84 Y4 J.8 Z-40 F80
N9 G79 Z0
N10 G0 Z20
N11 M30
$
```

Ek 25



ProEngineer, PPRINT Tablosu Örneği.

! PPRINT ITEM	FLAG	COMMENT
PART_NAME	YES	-
DATE_TIME	YES	-
SCALE	YES	-
TRANSLATE	YES	-
ROTATE	NO	-
OPERATION_NAME	NO	-
OPERATION_COMMENTS	YES	-
LAYER_NAME	YES	-
UDF_NAME	YES	-
TOOL_TABLE	YES	-
ONLY_OUTPUT_USED_TOOLS	YES	-
CUTCOM_GEOMETRY_TYPE	YES	-
NC_SEQUENCE_NAME	YES	-
NC_SEQUENCE_COMMENTS	NO	-
FEATURE_ID	YES	-
SEQUENCE_TYPE	YES	-
CUTCOM_REGISTER	NO	-
SPINDLE_SPEED	YES	-
CONVEX_CORNER_FINISH_TYPE	NO	-
CUT_FEEDRATE_&_UNITS	YES	-
ARC_FEEDRATE_&_UNITS	YES	-
FREE_FEEDRATE_&_UNITS	YES	-
RETRACT_FEEDRATE_&_UNITS	NO	-
PLUNGE_FEEDRATE_&_UNITS	NO	-
SCAN_TYPE	NO	-
RETRACT_HEIGHT	YES	-
TOOL_NAME	NO	-
TOOL_TYPE	YES	-
TOOL_POSITION_NUMBER	NO	-
TOOL_COMMENTS	YES	-
TOOL_PARAMETERS	NO	-
TOOL_OFFSET_NUMBER	YES	-
CHAMFER_LENGTH	YES	-
CORNER_RADIUS	NO	-
CSINK_ANGLE	YES	-
CUTTER_DIAM	YES	-
DRILL_DIAMETER	NO	-
DRILL_LENGTH	NO	-
END_ANGLE	NO	-
END_OFFSET	YES	-
GAUGE_Z_LENGTH	NO	-
GAUGE_X_LENGTH	NO	-
HOLDER_TYPE	NO	-
INSERT_LENGTH	NO	-
LENGTH	YES	-
LENGTH_UNITS	YES	-
NOSE_RADIUS	NO	-
NUM_OF_TEETH	NO	-
POINT_ANGLE	NO	-
SHANK_DIAMETER	NO	-
SIDE_ANGLE	NO	-
SIDE_WIDTH	YES	-
TOOL_MATERIAL	NO	-

Ek 26



INCH ve MM komutlarına örnek, kesme ve makina kontrol veri dosyaları

Metrik dosya

```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 36
MACHIN / MILL, 01
UNITS / MM
LOADTL / 103
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2500.000000, CLW
RAPID
GOTO / 18.2740064004, 21.7358690695, 20.0000000000
RAPID
GOTO / 18.2740064004, 21.7358690695, 5.0000000000
FEDRAT / 200.000000, MMPM
GOTO / 18.2740064004, 21.7358690695, 0.0000000000
GOTO / 18.2740064004, -19.5164169245, 0.0000000000
GOTO / -20.2697961698, -19.5164169245, 0.0000000000
GOTO / -20.2697961698, 21.7358690695, 0.0000000000
GOTO / 18.2740064004, 21.7358690695, 0.0000000000
GOTO / 18.2740064004, 21.7358690695, 20.0000000000
SPINDL / OFF
$$-> END /
FINI
```

Metrik sistemde makina kontrol veri dosyası

```
%PM99
N99
N1 G59
N2 G71
N3 G17
N4 M54
N5 T3 M06
N6 S2500 M13
N7 G0 X18.274 Y21.736 A0 C0
N8 Z5
N9 G1 Z0 F200
N10 Y-19.516
N11 X-20.27
N12 Y21.736
N13 X18.274
N14 Z20
N15 M30
$
```

İnç sistemde kesme dosyası



```
$$*          Pro/CLfile  Version 2000i - 1999260
$$-> MFGNO / TEZ_HOLES
PARTNO / TEZ_HOLES
$$-> FEATNO / 36
MACHIN / MILL, 01
UNITS / INCHES
LOADTL / 103
$$-> CUTTER / 12.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,
$
          0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2500.000000, CLW
RAPID
GOTO / 0.7194490709, 0.8557428768, 0.7874015748
RAPID
GOTO / 0.7194490709, 0.8557428768, 0.1968503937
FEDRAT / 200.000000, MPPM
GOTO / 0.7194490709, 0.8557428768, 0.0000000000
GOTO / 0.7194490709, -0.7683628710, 0.0000000000
GOTO / -0.7980234713, -0.7683628710, 0.0000000000
GOTO / -0.7980234713, 0.8557428768, 0.0000000000
GOTO / 0.7194490709, 0.8557428768, 0.0000000000
GOTO / 0.7194490709, 0.8557428768, 0.7874015748
SPINDL / OFF
$$-> END /
FINI
```

İnç sistemde makina kontrol veri dosyası

```
%PM998
N998
N1 G59
N2 G70
N3 G17
N4 M54
N5 T3 M06
N6 S2500 M13
N7 G0 X0.7194 Y0.8557 A0 C0
N8 Z0.1969
N9 G1 Z0 F8
N10 Y-0.7683
N11 X-0.798
N12 Y0.8557
N13 X0.7194
N14 Z0.7874
N15 M30
$
```