

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO**

**UM ESTUDO SOBRE IMPLEMENTAÇÃO DE CICLOS DE USINAGEM
ATRAVÉS DE PROGRAMAÇÃO PARAMETRIZADA EM MÁQUINAS
DE COMANDO NUMÉRICO COMPUTADORIZADO**

DISSERTAÇÃO DE MESTRADO

Marco Aurélio da Fontoura Gonçalves

Santa Maria, RS, Brasil

2007

**UM ESTUDO SOBRE IMPLEMENTAÇÃO DE CICLOS DE USINAGEM
ATRAVÉS DE PROGRAMAÇÃO PARAMETRIZADA EM MÁQUINAS
DE COMANDO NUMÉRICO COMPUTADORIZADO**

por

Marco Aurélio da Fontoura Gonçalves

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Engenharia de Produção, Área de Concentração em Tecnologia de Informação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Mestre em Engenharia de Produção.**

Orientador: Prof. Dr. Alexandre Dias da Silva

Santa Maria, RS, Brasil

2007

**Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Engenharia de Produção**

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

**UM ESTUDO SOBRE IMPLEMENTAÇÃO DE CICLOS DE USINAGEM
ATRAVÉS DE PROGRAMAÇÃO PARAMETRIZADA EM MÁQUINAS
DE COMANDO NUMÉRICO COMPUTADORIZADO**

elaborada por
Marco Aurélio da Fontoura Gonçalves

como requisito parcial para obtenção do grau de
Mestre em Engenharia de Produção

COMISSÃO EXAMINADORA:

Prof. Alexandre Dias da Silva, Dr. (UFSM)
(Presidente/Orientador)

Prof. Leandro Costa de Oliveira, Dr. (UFSM)

Prof. Inácio da Fontoura Limberger, Dr. (UFSM)

Santa Maria, 26 de julho de 2007.

À Márcia, esposa e companheira.
Ao meu filho, Igor Von Ende Gonçalves.

AGRADECIMENTOS.

Ao professor e orientador Alexandre Dias da Silva, pelo apoio, confiança e dedicação na condução do desenvolvimento e realização deste trabalho.

A minha família, pelo apoio, compreensão, alegria e estímulo constante.

Aos meus pais, Marilú da Fontoura Gonçalves e Romeu Almeida Gonçalves, in memoriam.

A direção e colegas do Colégio Técnico Industrial (CTISM), da Universidade Federal de Santa Maria (UFSM).

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Engenharia de Produção
Universidade Federal de Santa Maria

UM ESTUDO SOBRE IMPLEMENTAÇÃO DE CICLOS DE USINAGEM ATRAVÉS DE PROGRAMAÇÃO PARAMETRIZADA EM MÁQUINAS DE COMANDO NUMÉRICO COMPUTADORIZADO

AUTOR: MARCO AURÉLIO DA FONTOURA GONÇALVES

ORIENTADOR: ALEXANDRE DIAS DA SILVA

Santa Maria, 26 de julho de 2007.

As máquinas de comando numérico computadorizado (CNC) são programáveis e possuem funções que facilitam a obtenção de diferentes superfícies com geometria variada na manufatura de peças mecânicas. Este trabalho tem como finalidade apresentar uma metodologia de aplicação dos recursos avançados de programação no desenvolvimento de rotinas parametrizadas dedicadas ao processo de fresamento. A programação parametrizada mostrou-se uma ferramenta eficaz para criar ciclos de usinagem a serem implementados como recursos adicionais às funções de programação disponíveis no comando da máquina. Por usar recursos tradicionais de programação, a técnica reduz o número de linhas de códigos em relação aos métodos computacionais de geração automática de programas de operação do equipamento. O trabalho mostrou que a técnica de programação parametrizada aumenta a eficiência operacional da máquina, através do aumento dos recursos de programação, agregando maior valor na utilização da mesma para realizar operações produtivas.

Palavras-chave: comando numérico computadorizado, programação paramétrica, ciclos de usinagem, fresamento.

ABSTRACT

Dissertação de Mestrado
Programa de Pós-Graduação em Engenharia de Produção
Universidade Federal de Santa Maria

A STUDY ON IMPLEMENTING OF MANUFACTURING CYCLE THROUGH PARAMETERIZED PROGRAMMING IN COMPUTER NUMERICAL CONTROL MACHINES

AUTHOR: MARCO AURÉLIO DA FONTOURA GONÇALVES
ADVISOR: ALEXANDRE DIAS DA SILVA
Santa Maria, 26 de julho de 2007.

The computer numerical control (CNC) machines can be programmed. They have functions that help to get different superficies of varied geometry in the manufacturing process of mechanical parts. This research has the purpose to present a methodology of application of the resources of parametrized programming for the developing of manufacturing cycles to a machine (CNC). The parametrized programming seems to be an efficient tool to implement routines that the machine does not have. By using traditional programming resources, it reduces the number of programming lines in relation to more used methods. This dissertation shows a technique that increases the machine operational efficiency by increasing the programming resources, and also add value to the use of the machine in the operational production process.

Key-words: computer numerical control; parametrized programming; manufacturing cycles; manufacture.

LISTA DE FIGURAS

Figura 1 – Eixos programáveis de um torno CNC	21
Figura 2 – Eixos lineares programáveis de uma fresadora CNC.....	22
Figura 3 – Zero peça (zero programa).	22
Figura 4 – Exemplo de programação coordenada absolutas	23
Figura 5 – Movimento de uma ferramenta no sistema de coordenada incremental	23
Figura 6 – Exemplo de programação coordenada incrementais.....	24
Figura 7 – Movimento circular G02 e G03	30
Figura 8 – Definição de I e K	30
Figura 9 – Modelo de perfil para torneamento	32
Figura 10 – Modelo de perfil para fresamento	32
Figura 11 – Ciclo de usinagem G74	34
Figura 12 – Equação da elipse para cópias retangulares	45
Figura 13 - Trajetória em arco com programação parametrizada	46
Figura 14 - Elipse gerada pelo programa.....	47
Figura 15 – Centro de usinagem vertical ROMI Discovery 308.....	49
Figura 16 - Movimento da ferramenta tradicional para rebaixo.....	50
Figura 17 - Movimento da ferramenta proposto para rebaixo.....	50
Figura 18 - Movimento da ferramenta com “w” largura de corte	51
Figura 19 - Fluxograma Cavidade Retangular	52
Figura 20 - Movimento da ferramenta para Rampa (usinando)	53
Figura 21 - Movimento da ferramenta para Rampa (traçado).....	53
Figura 22 – Seqüência de movimento para a rampa.....	54
Figura 23 - Vista lateral (Esquema do movimento da ferramenta)	54
Figura 24 - Vista superior (Esquema do movimento da ferramenta)	55
Figura 25 – Fluxograma para a rampa.....	56
Figura 26 - Trajetória da ferramenta para rebaixo elíptico.....	57
Figura 27 - Cálculo dos pontos da elipse.....	57
Figura 28 – Fluxograma da elipse	59
Figura 29 – Esquema para o cálculo da cópia múltipla de elipse.....	60
Figura 30 – Trajetória da ferramenta para gerar cópias múltiplas de elipse.....	60
Figura 31 – Fluxograma cavidade elípticas múltiplas	62

Figura 32 – Concavidade Cônica.....	62
Figura 33 – Movimento em espiral da ferramenta	63
Figura 34 – Vista lateral da Cavidade Cônica	63
Figura 35 – Trajetória da ferramenta para a Cavidade Cônica.....	63
Figura 36 – Fluxograma para cavidade cônica.....	65
Figura 37 – Rasgo semi-circular.....	66
Figura 38 – Esquema para o desenvolvimento do perfil semi-circular	66
Figura 39 – Vista frontal do perfil semi-circunferência	67
Figura 40 – Vista superior do movimento da ferramenta.....	67
Figura 41 – Triângulo retirado da Figura 39	67
Figura 42 – Trajetória tradicional da ferramenta para usinar a peça.....	68
Figura 43 – Fluxograma cavidade semi-cilíndrica	69
Figura 44 Comando da fresadora Romi.....	70
Figura 45 - - Software CAD utilizado para analisar os projetos	71
Figura 46 – Software CAD/CAM, utilizado para gerar o código CNC	71
Figura 47 – Traçado para cavidade retangular (X,Y).....	73
Figura 48 – Traçado da cavidade retangular (Y,Z)	73
Figura 49 – Seqüência de usinagem da cavidade retangular	74
Figura 50 – Programas da cavidade retangular para comparação	74
Figura 51 – Traçado da rampa (X,Y).....	76
Figura 52 – Traçado da rampa (Y,Z).....	76
Figura 53 – Programa gerado pelo CAD/CAM para rampa.....	77
Figura 54 – Traçado da elipse de raio maior	79
Figura 55 – Traçado elíptico com redução do raio.....	79
Figura 56 – Seqüência de usinagem da elipse	80
Figura 57 – Programa gerado pelo CAD/CAM para a cavidade elíptica	80
Figura 58 – Programa com Raio maior 200 e Raio menor 100.....	81
Figura 59 – Traçado cópias múltiplas de elipse	82
Figura 60 – Seqüência de usinagem da cavidade elípticas múltiplas	83
Figura 61 – Programa gerado pelo CAD/CAM para a cavidades elípticas múltiplas	83
Figura 62 – Movimento espiral gerado pelo comando (X,Y).....	85
Figura 63 – Comando gerando profundidade cônica (Y,Z).....	85
Figura 64 – Usinagem da cavidade cônica	86
Figura 65 – Planilha de simulação para gerar a espiral	87

Figura 66 – Programa gerado pelo CAD/CAM para a cavidade cônica	87
Figura 67 – Simulação gráfica da cavidade circular (X,Y)	89
Figura 68 - Simulação gráfica da cavidade circular (X,Z)	89
Figura 69 – Programa gerado pelo CAD/CAM para a cavidade cilíndrica.....	90

LISTA DE TABELAS

Tabela 1 – Caracteres (Siglas) presentes em um programa CNC.....	24
Tabela 2 – Códigos G (General or preparatory) pelo Padrão ISO 1056.....	25
Tabela 3 – Códigos M (miscellaneous) pelo Padrão ISO 1056.....	26
Tabela 4 – Descrição de um programa CNC.....	31
Tabela 5 – Descrição de um programa CNC (Torno)	32
Tabela 6 – Descrição de um programa CNC (Fresadora)	33
Tabela 7 – Variáveis do exemplo do ciclo G74.....	34
Tabela 8 – Funções Aritméticas	39
Tabela 9 – Funções Lógicas	40
Tabela 10 – Programa exemplo com "Custom macro B"	44
Tabela 11 – Programa para gerar cópias retangulares de elipse.....	45
Tabela 12 - Variáveis do programa	46
Tabela 13 - Definindo equações e as posições	46
Tabela 14 - Programa parametrizado para geração de arcos	47
Tabela 15 - Programa para gerar elipse	48
Tabela 16 - Variáveis do programa cavidade retangular.....	51
Tabela 17 - Variáveis para o programa de rampa.....	54
Tabela 18 - Variáveis para o programa elipse	58
Tabela 19 - Variáveis para o programa copia múltiplas de elipse.....	61
Tabela 20 - Variáveis para o programa cavidade cônica.....	64
Tabela 21 - Variáveis para o programa rasgo cilíndrico	68
Tabela 22 - Programa parametrizado CNC para rebaixo retangular	72
Tabela 23 - Programação paramétrica CNC para perfil de rampa	75
Tabela 24 - Programação paramétrica CNC para elipse.....	78
Tabela 25 - Programação paramétrica CNC para cópia múltiplas de elipse	82
Tabela 26 - Programação paramétrica CNC para cavidade cônica	84
Tabela 27 - Programação paramétrica CNC para rasgo simi circular	88
Tabela 28 – Comparação do número de linhas gerado pelos dois processos.....	91

LISTA DE SIGLAS

ANSI	American National Standards Institute
APT	<i>Automatically Programmed Tools</i> (Ferramentas Programadas Automaticamente)
ATC	<i>Automatic Tool Changer</i> (Troca de Ferramentas Automática)
Auto Prompt	<i>Automatic Programming of Machine Tools</i>
CAD	<i>Computer Aided Design</i> (Desenho Assistidos por Computador)
CAE	<i>Computer Aided Engineering</i> (Engenharia Assistida por Computador)
CAM	<i>Computer Aided Manufacturing</i> (Manufatura Assistida por Computador)
CAP	<i>Computer Aided Planning</i> (Planejamento Assistido por Computador)
CAPP	<i>Computer Aided Process Planning</i> (Planejamento do Processo Assistido por Computador)
CLP	<i>Programmable Logical Computer</i> (Controlador Lógico Programável)
CN	Controle ou Comando Numérico
CNC	Controle ou Comando Numérico Computadorizado
DNC	<i>Direct Numeric Control</i>
EIA	Eletronic Industries Association
ISSO	<i>International Organization for Standardization</i> (Organização Internacional para Normalização)
M.I.T.	Instituto de Tecnologia de Massachusetts
MDI	<i>Manual Data Input</i> (Entrada manual de dados)
NAFA	Núcleo de Automação e Processos de Fabricação
PC	Computador Pessoal
UFSM	Universidade Federal de Santa Maria

SUMÁRIO

	Pág.
RESUMO	
ABSTRACT	
LISTA DE FIGURAS	
LISTA DE TABELAS	
LISTA DE SIGLAS	
1. INTRODUÇÃO	13
2. REVISÃO BIBLIOGRÁFICA	15
2.1 Comando Numérico Computadorizado – CNC	15
2.2 Formas de programação CNC	18
2.2.1 Programação manual.....	18
2.2.2 Programação manual assistida.....	18
2.2.3 Programação MDI.....	19
2.2.4 Sistemas Gráficos Interativos.....	20
2.2.5 Programação automatizada por computador.....	20
2.3 Transmissão do programa para o comando	21
2.4 Programa CNC	22
2.5 Sistema ISO de programação, linguagem G	25
2.5.1 Os três tipos de movimento mais básicos.....	29
2.5.2 Exemplos de programas.....	32
2.6 Ciclos de usinagem	35
2.7 Programação Parametrizada do CNC	36
3 METODOLOGIA	51
3.1 Cavidade Retangular	52
3.2 Rampa	55
3.3 Cavidade Elíptica	58
3.4 Cavidades Elípticas Múltiplas	62
3.5 Cavidade Cônica	64
3.6 Cavidade Semi-Cilíndrica	68
4 – RESULTADOS E DISCUSSÃO	72
4.1 Cavidade Retangular	74
4.2 Rampa	77
4.3 Cavidade Elíptica	80
4.4 Cavidades elípticas múltiplas	83
4.5 Cavidade Cônica	86
4.6 Cavidade Semi-Cilíndrica	89
4.7 Restrições dos ciclos	93
5 – CONCLUSÃO	94
REFERÊNCIAS BIBLIOGRÁFICAS	96

1. INTRODUÇÃO

Máquinas que necessitam de posicionamento, velocidade e aceleração controlados podem ser automatizadas por meio de um comando numérico computadorizado (CNC). Na atualidade, máquinas-ferramenta acionadas por CNC, são equipamentos muito utilizados em sistemas produtivos. Furadeiras, puncionadeiras, dobradeiras, tornos, fresadoras e mandriladoras são alguns exemplos.

Para se produzir uma peça usinada com uma máquina comandada por CNC, são necessários vários procedimentos. Um dos principais é a elaboração do programa CNC. Este programa contém informações geométricas e tecnológicas para a usinagem prevista, e deve propiciar boa qualidade de acabamento, redução de custo operacional e redução do tempo de operação.

A programação CNC é feita com varias técnicas, dependendo do nível tecnológico e financeiro disponível, podendo ser desde a entrada manual de dados, diretamente no comando da máquina, até as técnicas mais avançadas, como sistemas CAD/CAM (Desenho Assistido por Computador/Manufatura Assistida por Computador), que integra as etapas de projeto e fabricação. Os *softwares* CAD/CAM processam arquivos de desenho de peças em sistemas CAD, para obter automaticamente o programa CNC, possibilitam a simulação da usinagem com seleção de ferramentas e transmitem os comandos para a máquina no momento da produção.

Apesar da eficiência do CAD/CAM, a programação manual é bastante empregada, principalmente na programação de geometrias regulares. Nessa forma de programação, existem funções que auxiliam o programador. Exemplos dessas funções são os ciclos de usinagem para operações de desbaste e execução de cavidades ou ressaltos regulares.

A programação parametrizada é uma forma de programação manual avançada que permite incluir em programas CNC cálculos matemáticos, variáveis computacionais e desvios condicionais. Isto permite implementar lógica algorítmica na operação da máquina de comando numérico.

Muitas máquinas CNC não suportam tecnologias modernas de programação por limitação na capacidade de memória, pois foram originalmente projetadas para uma programação manual contendo poucas linhas de comando capazes de produzirem geometrias mais simples. As técnicas de programação modernas e automatizadas, que facilitam obter geometrias complexas, tornam-se difíceis de serem aplicadas a estas máquinas por gerarem

programas CNC extensos. Além disso, a exigência de mão de obra qualificada e a necessidade de aquisição de equipamentos (computadores) e softwares de custo elevado, são limitações que inviabilizam o investimento nessas tecnologias, principalmente para empresas de pequeno e médio porte.

Este trabalho tem como objetivo propor uma metodologia para aumentar a potencialidade de máquinas CNC através da aplicação de recursos de programação parametrizada no desenvolvimento de ciclos de usinagem.

Com este estudo tem-se uma referência à aplicação de uma técnica de programação que embora pouco difundida, quando adequadamente utilizada permite gerar rotinas para geometrias padronizadas ou parametrizadas.

A metodologia proposta será demonstrada através de rotinas implementadas no comando de um centro de usinagem vertical Romi Discovery 308, com comando CNC Mach8, pertencente ao Núcleo de Automação e Processos de Fabricação da Universidade Federal de Santa Maria (NAFA/UFSM).

O trabalho de pesquisa proposto está estruturado em cinco capítulos, cujos conteúdos, encontram-se a seguir:

No primeiro capítulo, encontra-se a definição e a natureza do problema em estudo, é relatado o objetivo do trabalho e a importância para o setor.

No segundo capítulo, faz-se as revisões mais aprofundadas do problema, bem como a fundamentação teórica sobre CNC e a programação paramétrica do CNC.

No terceiro capítulo, segue a metodologia, como será feito o trabalho, as ferramentas utilizadas, bem como a descrição dos estudos feitos.

No quarto capítulo, faz-se testes de viabilidade das rotinas desenvolvidas, analisando-se os resultados obtidos.

No quinto capítulo, têm-se as considerações finais da presente pesquisa, a partir do objetivo geral que se pretendia alcançar e dos resultados obtidos, sugerindo recomendações para trabalhos futuros.

2. REVISÃO BIBLIOGRÁFICA

2.1 Comando Numérico Computadorizado - CNC

Com a revolução industrial, as máquinas reduziram o trabalho físico dos operários no processo industrial. Atualmente computadores com grande poder de processamento de dados, auxiliam os programadores na elaboração de difíceis algoritmos. A união destes dois desenvolvimentos por alguns foi chamada de segunda revolução industrial.

Em 1949, a Força Aérea Norte-americana observou uma pequena empresa fabricante de hélices e rotores para helicópteros a "*Parsons Corporation*". E tinha dificuldades para produzir gabaritos para os rotores na velocidade necessária de produção, havia experimentado uma forma rudimentar de controle por números em uma máquina de usinagem convencional. (FIGUEIRA, 2003)

A Força Aérea Americana, percebendo um grande avanço na fabricação de aviões e material bélico, contratou a *Parsons* e patrocinou estudos e desenvolvimento do controle numérico. Planejaram e executaram as adaptações de controle numérico para uma fresadora de três eixos, a *Hydrotel*, da *Cincinnati Milling Machine Company*. Os controles e comandos convencionais foram retirados e substituídos pelo comando numérico, dotado de leitora de fita de papel perfurado, unidade de processamento de dados e servomecanismos nos eixos. Deste modo foi criado um protótipo de uma máquina CN que foi apresentado em 1953 no (MIT) Instituto de Tecnologia de Massachusetts. Os resultados foram excepcionais e demonstraram uma mudança revolucionária nas indústrias de transformação. (WILSON,1980)

Após este período, a Força Aérea Norte-americana teve um desenvolvimento extraordinário, pois as peças complexas e de grande precisão, empregadas na fabricação das aeronaves, principalmente os aviões a jato de uso militar, passaram a ser produzidos de forma simples e rápida, reduzindo-se os prazos de entrega do produto desde o projeto, até o acabamento final. Em 1956 surgiu o trocador automático de ferramentas. Mais tarde em 1958, os equipamentos com controle de posicionamento ponto a ponto e a geração contínua de contornos, foram melhorados por este sistema em desenvolvimento.

A partir de 1957, houve nos Estados Unidos, uma grande corrida na fabricação de máquinas comandadas por CN. Até então os industriais investiam em adaptações do CN em máquinas convencionais. Este novo processo foi cada vez mais usado na rotina de manufatura, fazendo com que surgissem novos fabricantes, os quais já produziam seus próprios sistemas e linguagem de comandos. Devido ao grande número de fabricantes,

começaram a surgir os primeiros problemas, sendo o principal a falta de uma linguagem única e padronizada.

A falta de padronização era bastante sentida nas empresas com mais de uma máquina, fabricadas por diferentes fornecedores, onde cada um deles tinha uma linguagem própria. Com isso, havia a necessidade de uma equipe técnica especializada para cada tipo de comando, elevando os custos de fabricação. (CASSANIGA, 2000)

Em 1958, por intermédio da EIA (*Electronic Industries Association*) organizaram-se estudos no sentido de padronizar os tipos de linguagem. A linguagem destinada à programação de máquinas era a APT (*Automatically Programed Tools*), desenvolvida pelo Instituto de Tecnologia de Massachusetts em 1956. Daí para frente foram desenvolvidas outras linguagens para a geração contínua de contornos como *AutoPrompt* (*Automatic Programming of Machine Tools*), ADAPT, *Compact II*, *Action*, e outros que surgiram e continuam surgindo para novas aplicações. Os códigos de máquinas CNC mais utilizados no mundo foram padronizados inicialmente pela "ISO 1056:1975" e posteriormente atualizada pela "ISO 6983-1:1982" ou seja, pela (*International Organization for Standardization*) e de forma muito semelhante pela ANSI "NAS995-1996" (*American National Standards Institute*) e também pela EIA. (Historia do CNC, 2007)

Com o aparecimento do circuito integrado, houve grande redução no tamanho físico dos comandos e sensível aumento da capacidade de armazenamento, comparando-se com os controles transistorizados.

Em 1967, as primeiras máquinas controladas numericamente chegaram ao Brasil, vindas dos Estados Unidos. A partir daí, observa-se uma evolução contínua e notável concomitantemente com os computadores em geral, fazendo com que os comandos CNC mais modernos empreguem em seu conceito físico (*hardware*) tecnologia de última geração.

O CNC é um equipamento eletrônico que recebe informações de como a máquina vai realizar uma operação, por meio de linguagem própria, denominado programa CNC, processa essas informações e devolve-as ao sistema através de impulsos elétricos. (CASSANIGA, 2000)

Os sinais elétricos são responsáveis pelo acionamento dos motores que darão à máquina os movimentos desejados com todas as características possíveis, realizando a operação na seqüência programada sem a intervenção do operador.

Atualmente, a utilização do CNC, é a saída mais apropriada para a solução dos mais complexos problemas de usinagem. Onde anteriormente se exigia uma máquina ou uma ferramenta especial, atualmente é feito com o CNC de uma forma relativamente simples.

O CNC não é apenas um sistema que atua diretamente no equipamento, ele deve ser encarado como um processo que deve ser responsável por mudanças na cultura da empresa. Isto quer dizer que, para que se tenha um melhor aproveitamento de um equipamento CNC, é interessante que se tenha uma boa organização, principalmente no que se refere ao processo de fabricação, controle de ferramentais (fixação, corte e medição) e administração dos tempos padrões e métodos de trabalho. (CASSANIGA, 2000)

Vantagens da utilização de máquinas CNC:

- Redução do tempo de fabricação, maior controle na seqüência das operações, fazendo com que os tempos padrões previstos sejam mais seguros. Com os tempos padrões mais seguros, tem-se uma maior precisão nos cálculos de custos, nos controles de carga máquina, nos controles de carga homem. (Usinagem Brasil, 2007)
- Maior controle no consumo de ferramentais. Por se tratar de uma usinagem com esforços, velocidades constantes, uniformes, e repetitivos, fazem com que os desgastes estejam sob controle. Isto facilita ao controle do estoque, ao permitir melhor desenvolvimento e teste de fornecedores, melhor controle por desgaste ao processo das ferramentais, evitando retrabalhos e refugos.
- Redução nos tempos de preparação (*set-up*) tornando viável a produção de pequenos lotes.
- Redução de itens acabados no estoque, por se possibilitar produção de pequenos lotes.
- Redução nos tempos e na freqüência, com que as inspeções de qualidade são efetuadas.

2.2 Formas de programação CNC

Os comandos de um programa CNC são os responsáveis pelo controle de uma máquina CNC, informando todas as etapas de fabricação de uma determinada operação de uma peça. Uma linha de comando de um programa CNC pode conter informações sobre o movimento da ferramenta (movimento rápido, interpolação, etc.), informações tecnológicas (velocidade, avanço, etc.), ou informações que ativem funções auxiliares (ligar refrigerante, eixo da árvore, etc.). A obtenção dessas informações depende sobre tudo dos dados da peça a

ser usinada, considerando-se as limitações da máquina, as características do CNC e da ferramenta. (FIGUEIRA, 2003)

Os métodos ou técnicas de programação incluem uma variedade de procedimentos, que vão desde o método manual até o método altamente automatizado. Groover (1987) e Lynch (1997), por exemplo, classificam as técnicas de programação em cinco tipos: programação manual, programação manual assistida, programação MDI (Manual Data Input), programação via CAD/CAM e programação automatizada por computador.

2.2.1 Programação manual

Segundo Gibbs (1994), a programação manual refere-se à preparação de um programa, sem a utilização de recursos computacionais para determinar trajetórias de ferramentas, pontos de intersecção de perfis, avanços, velocidades, etc.. Nesse tipo de programação, as instruções de processamento são documentadas em um formulário chamado manuscrito do programa, que é uma listagem das posições da ferramenta em relação à peça, que a máquina precisa seguir para executar o processamento. A listagem também pode incluir outros comandos com informações sobre velocidades, avanços, ferramental, etc.. Este tipo de programação é ideal para aplicações ponto a ponto em trabalhos de torneamento e fresamento que envolvam o processamento de peças geometricamente simples e que requeiram poucas operações (Lynch, 1994).

2.2.2 Programação manual assistida

Na programação assistida por computador o trabalho de cálculo requerido na programação manual é automatizado. Para peças com geometrias complexas ou trabalhos com muitas etapas de processamento, o uso do computador resulta em economias significativas de tempo de programação. A escolha entre programação manual e programação assistida por computador depende do tipo e complexidade das peças produzidas, do número de eixos programados em cada máquina CNC, do volume de programas CNC elaborados por período e do tempo médio de elaboração dos programas (Lynch, 1997).

2.2.3 Programação MDI

O sistema MDI (Entrada manual de dados) é um procedimento no qual o programa CNC é inserido diretamente no comando da máquina-ferramenta. Consequentemente, o uso de sistemas de transferência de dados é desnecessário. O procedimento de programação é

simplificado para permitir que, além dos programadores, os operadores de máquina também façam ou alterem o programa. O operador/programador insere os dados do programa e comandos na unidade de controle da máquina. Os sistemas MDI são entendidos como sendo o caminho para as pequenas fábricas introduzirem a tecnologia CNC nas suas operações, sem a necessidade de adquirir equipamentos especiais de programação e contratar programadores. Isso permite à fábrica fazer um investimento inicial mínimo, para iniciar a transição em direção à tecnologia CNC. É um sistema bastante adequado para ferramentarias e empresas que prestam serviços de usinagem. Uma consideração importante em relação à aplicação do MDI, é que a máquina CNC não fique parada enquanto o operador/programador está entrando com as instruções de programação. O uso eficiente do sistema dita que a programação da próxima peça seja executada enquanto se esteja usinando uma outra peça. Isso reduz o tempo de troca de um trabalho para outro (Lynch, 1997). É fundamental observar que nem todos os comandos CNC têm essa característica.

Ainda no campo da programação MDI, existem as "máquinas eletrônicas" que além de poderem ser programadas via MDI, possuem um controle com ciclos fixos, podem ser operadas manualmente e também tem recursos para serem programadas via programação por aprendizagem. O operador usina a primeira peça conduzindo a máquina como se esta fosse convencional, e os movimentos são gravados automaticamente no controle. Os dados são compilados num programa e armazenados no computador, permitindo a execução da segunda peça e peças subseqüentes através de um arquivo, para uso futuro (Mason, 1995).

2.2.4 Sistemas Gráficos Interativos

Este tipo de programação é uma forma avançada da programação assistida por computador. Nele um sistema gráfico interativo, equipado com *software* de programação CNC, é usado para facilitar a atividade de programação. Neste método, o programador trabalha numa estação CAD/CAM para selecionar ferramentas, métodos e procedimentos para usinar a peça projetada no CAD. Os sistemas de programação CNC com base em sistemas CAD/CAM foram criados inicialmente para solucionar as falhas da programação manual. Atualmente são concebidos para programar tanto peças simples como complexas, de forma mais rápida, mais fácil e com um grau mais alto de precisão. Suas aplicações são bem sucedidas nas indústrias de fabricação de moldes e aeroespacial, que trabalham com usinagem de peças tridimensionais complexas e cujos programas são muito extensos (Lynch, 1994). Segundo Anderl et al (1998), podem ocorrer neste tipo de programação, problemas relacionados à confiabilidade de transferência dos dados geométricos do sistema CAD para a

máquina CAM, à velocidade de cálculo e confiabilidade das trajetórias geradas e à complexidade do processo.

2.2.5 Programação automatizada por computador

Este tipo automatiza a atividade de programação completa, usando um *software* que é capaz de tomar decisões lógicas sobre como a peça deva ser usinada. Na programação CNC usando o sistema CAD/CAM, muitos aspectos do procedimento foram automatizados. O objetivo aqui é tornar possível automatizar o procedimento completo da programação CNC. A interação entre esta atividade e o CAPP, o Planejamento do Processo Assistido por Computador (*Computer Aided Process Planning*) é fundamental para o sucesso da técnica (Lynch, 1997).

2.3 Transmissão do programa para o comando

Todos os programas, sejam criados manualmente num editor de texto, ou automaticamente a partir de um programa CAM, têm de alguma forma, que ser transferidos para o comando da máquina CNC.

Para isso, os controladores dispõem de um ou mais dispositivos de entrada, nomeadamente:

- porta serial (RS-232);
- placa de rede (com entrada BNC ou RJ-45);
- *drive* de disquetes.

A porta serial é o dispositivo mais comum nas máquinas CNC e o envio de programas faz-se através do protocolo DNC (*Direct Numeric Control*). O protocolo DNC consiste basicamente no envio do programa, bloco por bloco, em modo texto. O computador que envia o programa apenas aguarda os sinais “X-ON” e “X-OFF”: quando o controlador envia o sinal “X-ON”, o PC envia blocos, até que o controlador o mande aguardar através do sinal “X-OFF”. Sempre que o *Buffer* fica “cheio”, a memória armazena os blocos em fila para serem executados. Para a comunicação via porta serial em modo DNC, não basta, normalmente, a ligação DATA-In e DATA-Out cruzada com GROUND ligado a GROUND, já que os sinais X-ON e X-OFF são comunicados por condutores próprios. Além disso, os chamados “*S hunts*” ligam pinos da mesma porta, de forma que o controlador possa verificar a presença ou ausência do cabo.

Desde algum tempo, os controladores passaram a ser construídos baseados em PC's e sistemas operativos como o Windows NT, embora com configurações especiais. Esses

controladores dispõem de discos rígidos e podem receber, tal como qualquer PC, placas de rede, tornando o envio de programas muito mais fácil, robusto e prático. Contudo, este tipo de controladores não vem equipado de origem com a placa de rede e se não for incluída na compra inicial da máquina CNC, a instalação pode custar centenas de vezes mais do que o preço de uma simples placa de rede para um PC.

Para pequenos programas, o *drive* de disquetes é bastante simples e fácil de utilizar, quando o formato utilizado é comum com os PC's. Contudo, a limitação de espaço de um disquete impede a transmissão de programas grandes, que são normalmente gerados pelos programas CAM. (FIGUEIRA, 2003)

2.4 Programa CNC

O programa CNC contém as informações geométricas e tecnológicas da tarefa a ser executada em cada operação. As informações geométricas da peça são transmitidas ao comando com o auxílio de um sistema de coordenadas cartesianas.

Torno

Os movimentos no plano X,Z para a ferramenta é mostrado na Figura 1.

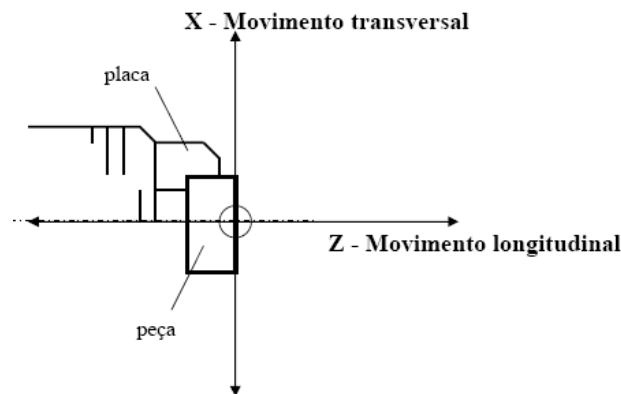


Figura 1 – Eixos programáveis de um torno CNC
Fonte: (COSTA, 2006)

Fresadora

Os movimentos no plano x , y são dados pela mesa (peça) e o movimento em z é dado pela ferramenta (Figura 2).

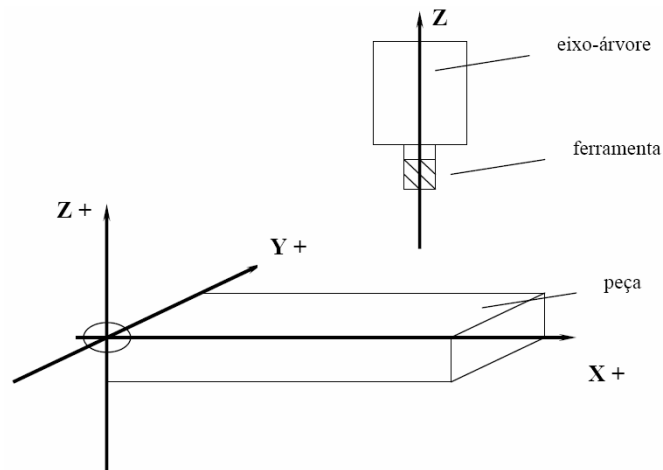


Figura 2 – Eixos lineares programáveis de uma fresadora CNC
Fonte: (COSTA, 2006)

O sistema de coordenadas é definido no plano, formado pelo cruzamento de uma linha paralela ao movimento longitudinal (Z) com outra paralela ao movimento transversal (X).

Para o torneamento, todo o movimento da ponta da ferramenta é descrito neste plano XZ em relação a uma origem pré-estabelecida (X_0, Z_0). Sendo assim, é conveniente adotar-se que X é sempre a medida do diâmetro. (COSTA, 2006)

Sistema de coordenadas absolutas

Neste sistema, a origem é estabelecida em função da peça a ser executada, para tanto, pode-se estabelecê-la em qualquer ponto do espaço facilitando a programação. Este processo denomina-se “Zero Flutuante”.

Como vimos a origem do sistema foi fixada como sendo os pontos X_0, Z_0 . O ponto X_0 é definido pela linha de centro-árvore. O ponto Z_0 é definido por qualquer linha perpendicular à linha de centro do eixo-árvore.

Durante a programação, normalmente a origem (X_0, Z_0) é pré-estabelecida no fundo da peça (encosto das castanhas) ou na face da peça (Figura 3).

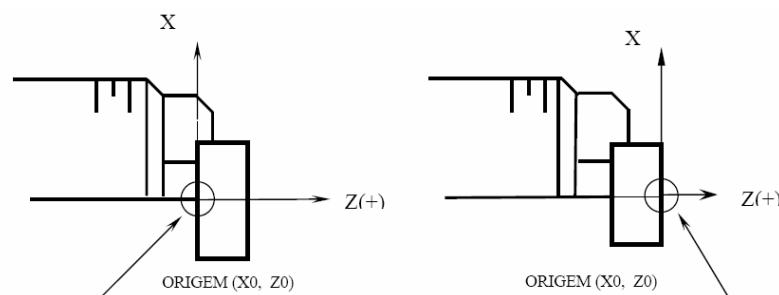


Figura 3 – Zero peça (zero programa).
Fonte: (COSTA, 2006)

EXEMPLO DE PROGRAMAÇÃO:

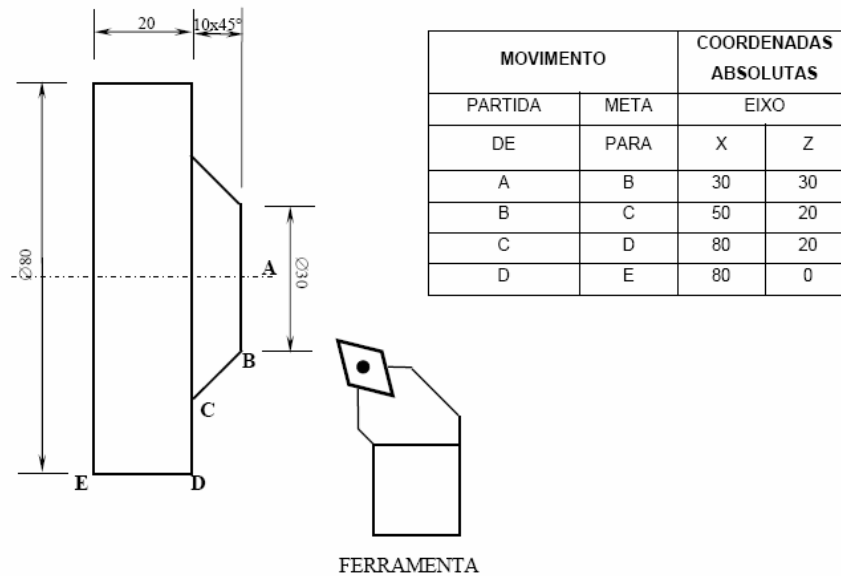


Figura 4 – Exemplo de programação coordenada absolutas
Fonte: (COSTA, 2006)

Sistemas de coordenadas incrementais

A origem deste sistema é estabelecida para cada movimento da ferramenta. Após qualquer deslocamento haverá uma nova origem, ou seja, para qualquer ponto atingido pela ferramenta, a origem das coordenadas passará a ser o ponto alcançado. Todas as medidas são feitas através da distância a ser deslocada.

Se a ferramenta desloca-se de um ponto A até B (dois pontos quaisquer), as coordenadas a serem programadas serão as distâncias entre os dois pontos, medidas (projetadas) em X e Z.

Note que o ponto A é a origem do deslocamento para o ponto B (Figura 5). Após a máquina chegar ao ponto B, esse será a origem para o deslocamento até o ponto C, e assim sucessivamente. (COSTA,2006)

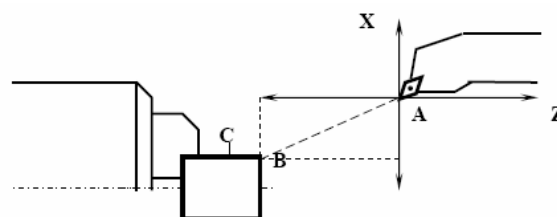


Figura 5 – Movimento de uma ferramenta no sistema de coordenada incremental
Fonte: (COSTA, 2006)

EXEMPLO DE PROGRAMAÇÃO:

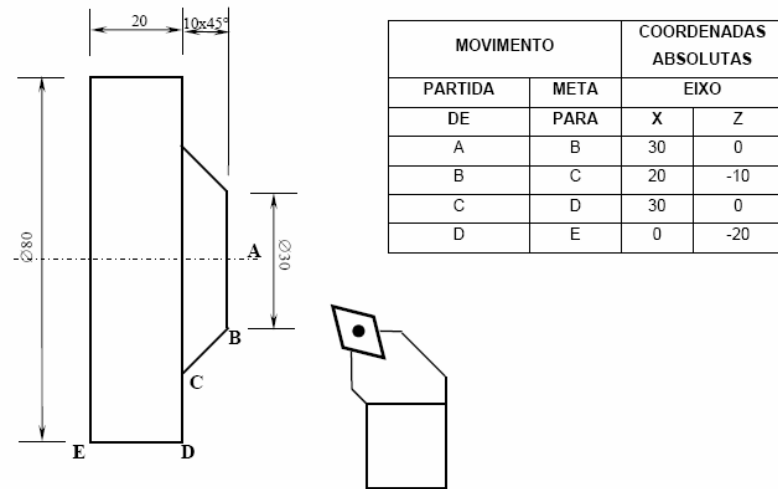


Figura 6 – Exemplo de programação coordenada incrementais
Fonte: (COSTA, 2006)

2.5 Sistema ISO de programação, linguagem G

Um programa de usinagem CNC é uma lista de instruções codificadas que descrevem como a peça projetada será usinada. Cada linha do programa é chamada de bloco, e estes blocos são executados seqüencialmente. Nem todos os códigos estão ainda sob controle da norma internacional ISO. Por isso, o mesmo código pode ter um significado diferente dependendo do comando (*Siemens, Fanuc, Mach, etc.*) ou da mudança de máquina (torno e fresadora). A Tabela 01 fornece uma lista de caracteres que vão estar presentes em um programa CNC. (Mundo CNC, 2007)

Tabela 1 – Caracteres (Siglas) presentes em um programa CNC

CARACTERE	SIGNIFICADO
O	Número do programa
N	Número da linha (bloco)
G	Função preparatória
X	Movimento no eixo X
Y	Movimento no eixo Y
Z	Movimento no eixo Z
I	Distância incremental paralela ao eixo X
J	Distância incremental paralela ao eixo Y
K	Distância incremental paralela ao eixo Z
R	Raio da trajetória da ferramenta/ Recuo da ferramenta no ciclo de desbaste - torno

M	Função miscelânea
H	Altura da ferramenta
T	Número da ferramenta
S	Rotação do eixo-árvore
F	Avanço
U	Profundidade de corte em X no raio /material deixado em X para acabamento - torno
W	Material deixado em Z para acabamento - torno
P	Número de um sub-programa
; ou #	Fim de bloco

Fonte: <http://www.mundocnc.com.br/>

As funções G (preparatórias) e M (miscelâneas) são funções que compõem basicamente um programa CNC. As funções “G” definem a máquina o que fazer preparando-a para executar movimentos e reconhecer unidades de medida. As funções “M” funcionam como botões liga/desliga; a exemplo temos a função M08 que liga o refrigerante de corte e M09 que desliga. As Tabelas 02 e 03 mostram algumas funções G e M usadas regularmente em programas de usinagem CNC.

Tabela 2 – Códigos G (General or preparatory) pelo Padrão ISO 1056.

Código G	Função
G00	Posicionamento rápido
G01	Interpolação linear
G02	Interpolação circular no sentido horário (CW)
G03	Interpolação circular no sentido anti-horário (CCW)
G04	Temporização (Dwell)
G05	Não registrado
G06	Interpolação parabólica
G07	Não registrado
G08	Aceleração
G09	Desaceleração
G10 a G16	Não registrado
G17	Seleção do plano XY
G18	Seleção do plano ZX
G19	Seleção do plano YZ
G20	Programação em sistema Inglês (Polegadas)
G21	Programação em sistema Internacional (Métrico)
G12 a G24	Não registrado
G25 a G27	Permanentemente não registrado
G28	Retorna a posição do Zero máquina
G29 a G32	Não registrados
G33	Corte em linha, com avanço constante
G34	Corte em linha, com avanço acelerando
G35	Corte em linha, com avanço desacelerando
G36 a G39	Permanentemente não registrado

G40	Cancelamento da compensação do diâmetro da ferramenta
G41	Compensação do diâmetro da ferramenta (Esquerda)
G42	Compensação do diâmetro da ferramenta (Direita)
G43	Compensação do comprimento da ferramenta (Positivo)
G44	Compensação do comprimento da ferramenta (Negativo)
G45 a G52	Compensações de comprimentos das ferramentas
G53	Cancelamento das configurações de posicionamento fora do zero fixo
G54	Zeragem dos eixos fora do zero fixo (01)
G55	Zeragem dos eixos fora do zero fixo (02)
G56	Zeragem dos eixos fora do zero fixo (03)
G57	Zeragem dos eixos fora do zero fixo (04)
G58	Zeragem dos eixos fora do zero fixo (05)
G59	Zeragem dos eixos fora do zero fixo (06)
G60	Posicionamento exato (Fino)
G61	Posicionamento exato (Médio)
G62	Posicionamento (Grosso)
G63	Habilitar óleo refrigerante por dentro da ferramenta
G64 a G67	Não registrados
G68	Compensação da ferramenta por dentro do raio de canto
G69	Compensação da ferramenta por fora do raio de canto
G70	Programa em Polegadas
G71	Programa em metros
G72 a G79	Não registrados
G80	Cancelamento dos ciclos fixos
G81 a G89	Ciclos fixos
G90	Posicionamento absoluto
G91	Posicionamento incremental
G92	Zeragem de eixos (mandatório sobre os G54...)
G93	Avanço dado em tempo inverso (Inverse Time)
G94	Avanço dado em minutos
G95	Avanço por revolução
G96	Avanço constante sobre superfícies
G97	Rotação do fuso dado em RPM
G98 e G99	Não registrados

Fonte: <http://www.mundocnc.com.br/>

Tabela 3 – Códigos M (miscellaneous) pelo Padrão ISO 1056

Código M	Função
M00	Parada programa
M01	Parada opcional
M02	Fim de programa
M03	Liga o fuso no sentido horário (CW)
M04	Liga o fuso no sentido anti-horário (CCW)
M05	Desliga o fuso
M06	Mudança de ferramenta
M07	Liga sistema de refrigeração número 2
M08	Liga sistema de refrigeração número 1
M09	Desliga o refrigerante

M10	Atua travamento de eixo
M11	Desliga atuação do travamento de eixo
M12	Não registrado
M13	Liga o fuso no sentido horário e refrigerante
M14	Liga o fuso no sentido anti-horário e o refrigerante
M15	Movimentos positivos (aciona sistema de espelhamento)
M16	Movimentos negativos
M17 e M18	Não registrados
M19	Parada do fuso com orientação
M20 a M29	Permanentemente não registrado
M30	Fim de fita com rebobinamento
M31	Ligando o "Bypass"
M32 a M35	Não registrados.
M36	Acionamento da primeira gama de velocidade dos eixos
M37	Acionamento da segunda gama de velocidade dos eixos
M38	Acionamento da primeira gama de velocidade de rotação
M39	Acionamento da segunda gama de velocidade de rotação
M40 a M45	Mudanças de engrenagens se usada, caso não use, Não registrados.
M46 e M47	Não registrados.
M48	Cancelamento do G49
M49	Desligando o "Bypass"
M50	Liga sistema de refrigeração número 3
M51	Liga sistema de refrigeração número 4
M52 a M54	Não registrados.
M55	Reposicionamento linear da ferramenta 1
M56	Reposicionamento linear da ferramenta 2
M57 a M59	Não registrados
M60	Mudança de posição de trabalho
M61	Reposicionamento linear da peça 1
M62	Reposicionamento linear da peça 2
M63 a M70	Não registrados.
M71	Reposicionamento angular da peça 1
M72	Reposicionamento angular da peça 2
M73 a M89	Não registrados.
M90 a M99	Permanentemente não registrados

Fonte: <http://www.mundocnc.com.br/>

As funções “G” podem ser modais e não-modais. As funções modais, uma vez programadas, permanecem na memória do comando, valendo para todos os blocos posteriores. Já as não-modais, todas as vezes que requeridas, devem ser programadas, ou seja, são válidas somente nos blocos que as contêm.

O código G01(interpolação linear) e G00(movimento rápido) parecem atuar de modo igual, porém, se existem os dois e com descrições diferentes com certeza é porque eles atuam diferentemente, às vezes um programador ou operador de máquina CNC tem o sentimento

que a única diferença é o fato que em G00 a máquina movimenta-se na velocidade mais rápida possível e em G01 a mesma máquina movimenta-se em velocidade controlada. De certo modo isto é verdade, porém algumas outras diferenças sutis podem causar um movimento inesperado na máquina, pois o programador no CAD/CAM vê um movimento rápido do mesmo modo que uma interpolação linear. (Mundo CNC, 2007)

2.5.1 Os três tipos de movimento mais básicos

Uma máquina CNC pode ter vários tipos de movimentos (dependendo de sua aplicação), revisaremos os três mais comuns disponíveis em quase todas as formas de equipamento de CNC. Estes tipos de movimentos têm em comum duas coisas. Primeiro, eles são todo modais. Isto significa que eles permanecem em efeito até outro comando interrompê-los. Por exemplo, se vários movimentos do mesmo tipo forem usados consecutivamente, a função G correspondente só aparecerá no primeiro comando.

Segundo, o ponto final do movimento é especificado em cada comando de movimento. A posição atual da máquina será tomada como o ponto de partida. (COSTA, 2006)

Movimento rápido G00

Este tipo de movimento é usado para comandar movimento à taxa de avanço mais rápida da máquina. É usado para minimizar tempos não produtivos durante o ciclo de usinagem. Usos comuns para movimento rápido incluem posicionamento da ferramenta para se iniciar um corte, movimentos de desvios de partes auxiliares tais como grampos, fixadores e outras obstruções, e em geral, qualquer movimento não cortante durante o programa.

O manual do construtor da máquina determinar a velocidade de movimentação rápida. Normalmente esta velocidade é extremamente rápida (algumas máquinas ostentam valores de mais de 25m/min). O operador deve ser cauteloso ao verificar comandos de movimento rápidos. Felizmente, há um modo para o operador anular a valor rápido durante verificação de programa.

O comando que quase todas as máquinas CNC usam para iniciar movimento rápido é o G00. Dentro do comando de G00, o ponto final para o movimento é determinado. Os fabricantes de controle variam com respeito ao que de fato acontece se mais de um eixo é incluído no comando de movimento rápido. Com a maioria dos controles, a máquina moverá tão rápido quanto possível em todos os eixos comandados. Neste caso, um eixo alcançará seu ponto de destino provavelmente antes dos outros. Com este tipo de comando rápido,

movimento de linha reta não acontecerá durante movimentos rápidos neste sentido o programador deve ter muito cuidado se há obstruções para evitar. Com outros controles, acontecerá movimento de linha reta entre o ponto inicial e final do movimento, até mesmo nos comandos de movimento rápidos. (COSTA, 2006)

Movimento em linha reta G01

Movimento de linha reta com avanço programado pode ser usado a qualquer momento, um movimento cortante reto é requerido enquanto se faz uma furação, um faceamento e ao fresar superfícies retas.

O método pelo qual a velocidade de avanço é programada varia de um tipo de máquina para o outro. Em geral, centros de usinagens só permitem que a taxa de avanço seja especificada em formato (polegadas ou milímetros por minuto). Os centros de torneamento também permitem se especificar taxa de avanço em formato (polegadas ou milímetros por revolução).

A palavra G01 é normalmente usada para especificar movimentação em linhas retas. No G01, o programador incluirá o ponto final desejado em cada eixo. (COSTA, 2006)

Movimento circular G02 G03

Neste tipo de movimento a ferramenta executa trajetórias na forma de um caminho circular. Dois códigos G são usados com movimento circular. G02 é usado para especificar movimentos circulares à direita (sentido horário) enquanto G03 é usado para especificar os movimentos circulares a esquerda (sentido ante horário). Para se avaliar o qual usar, simplesmente precisa-se ter uma visão do movimento da mesma perspectiva que a máquina verá o movimento. Por exemplo, se fazendo um movimento circular em XY em um centro de usinagem, simplesmente veja o movimento do ponto de vista do fuso. Fazendo se um movimento circular em XZ em um centro de torneamento, simplesmente veja o movimento de sobre o fuso (Figura 7). (COSTA, 2006)

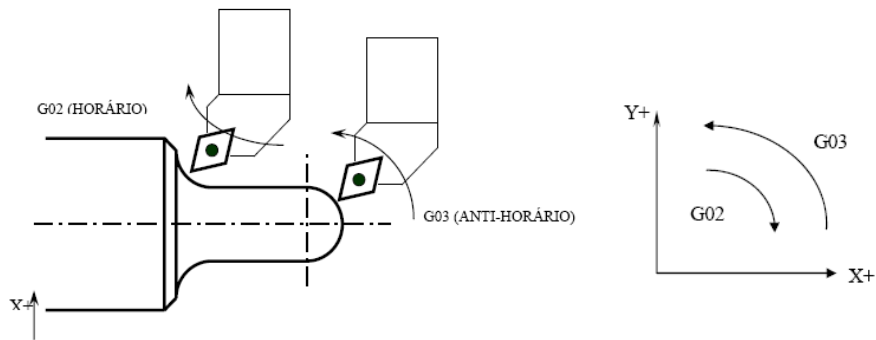


Figura 7 – Movimento circular G02 e G03
Fonte: (COSTA, 2006)

Adicionalmente, movimento circular requer que por um meio ou outro, o programador especifique o raio do arco a ser gerado. Com os controles CNCs mais novos isto é controlado por um "R" que simplesmente declara o raio. Com controles mais antigos, vetores direcionais (especificado por I, J e K) digam ao controle o local do ponto de centro do arco (Figura 8). Desde que controles variam com respeito a como são programados vetores direcionais, e desde que a palavra de R está ficando mais popular para designação de raio, nossos exemplos mostrarão o uso do R.

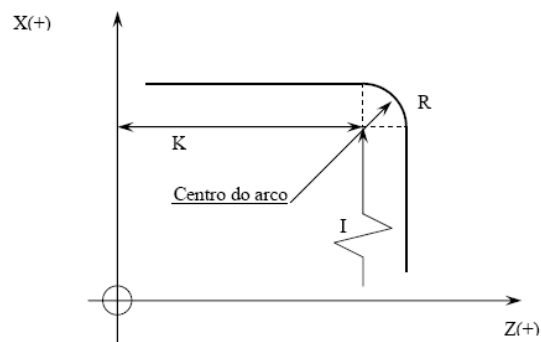


Figura 8 – Definição de I e K
Fonte: (COSTA, 2006)

2.5.2 Exemplos de programas

A Tabela 04 mostra uma listagem de um programa exemplo com linguagem “G”, que mostra o uso dos movimentos básicos G00, G01, G02 e G03 citados anteriormente.

Tabela 4 – Descrição de um programa CNC

Bloco de comando	Descrição do bloco
O0002	Número de Programa
N005 G54 G90 S350 M03	Seleciona o sistema de coordenadas, modo absoluto e fuso gira a 350 RPM sentido horário
N010 G00 X -.625 Y -.25	Rápido para o ponto 1
N015 G43 H01 Z -.25	Ativar a compensação de comprimento da ferramenta, rápido até superfície de trabalho
N020 G01 X5.25 F3.5	Máquina em movimento direto para o ponto 2, taxa de avanço 3.5 pol/min.
N025 G03 X6.25 Y.75 R1.0	Movimento circular no sentido ante horário CCW para o ponto 3
N030 G01 Y3.25	Máquina em movimento direto para o ponto 4
N035 G03 X5.25 Y4.25 R1.0	Movimento circular no sentido ante horário CCW para o ponto 5
N040 G01 X.75	Máquina em movimento direto para o ponto 6
N045 G03 X -.25 Y3.25 R1.0	Movimento circular no sentido ante horário CCW para o ponto 7
N050 G01 Y.75	Máquina em movimento direto para o ponto 8
N055 G03 X.75 Y -.25 R1.0	Movimento circular no sentido ante horário CCW para o ponto 9
N060 G00 Z.1	Rápido ao longo do eixo Z
N065 G91 G28 Z0	Vai para o ponto de referência da máquina em Z
N070 M30	Fim de programa, rebobinamento da fita

Fonte: <http://www.mundocnc.com.br/>

Os controles CNC variam com respeito a limitações dos tipos de movimento. Por exemplo, alguns controles têm regras rígidas que regem o quanto de um círculo completo lhe permitem para fazer dentro de um comando circular. Alguns requerem vetores direcionais para comandos de movimento circulares em vez de permitir o "R". Alguns têm ciclos fixos para fazer chanfros e concordâncias circulares em cantos, minimizando o número de comandos de movimento que devem ser dados. (COSTA, 2006)

Exemplo de um programa para torno, do perfil da Figura 9.

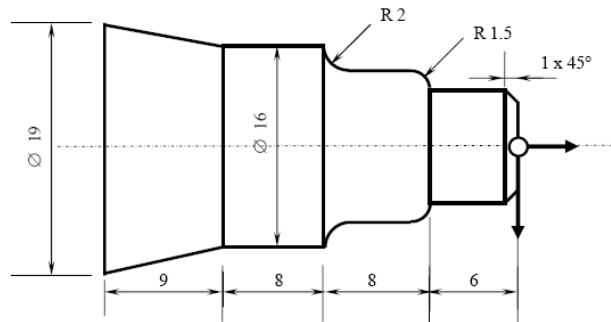


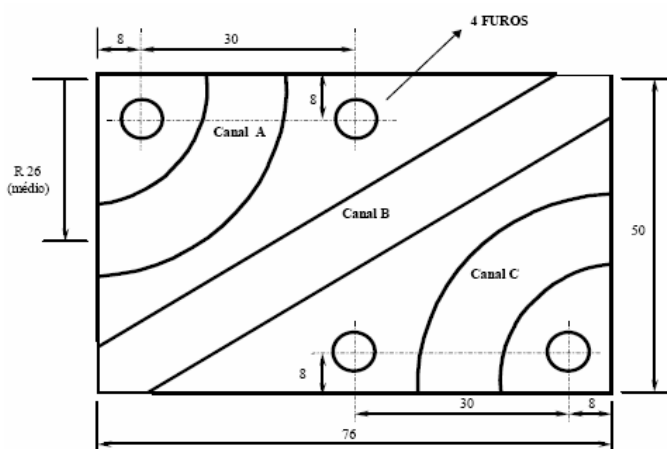
Figura 9 – Modelo de perfil para torneamento

Tabela 5 – Descrição de um programa CNC (Torno)

Programa CNC	
N10 O005	N130 X9 Z-1
N20 G21	N140 Z-6
N30 [BILLET X19 Z50	N150 G03 X12 Z-7.5 R1.5
N40 G98	N160 G01 Z-12
N50 G28 U0 W0	N170 G02 X16 Z-14 R2
N60 M06 T0101	N180 G01 Z-22
N70 G97 S2500 M03	N190 X19 Z-31
N80 G00 X19 Z0.5	N200 G00 X19.5
N90 G71 U0.3 R0.25	N210 G70 P110 Q200
N100 G71 P110 Q200 U0.2 W0.2 F90	N220 M05
N110 G00 X7 F70	N230 G28 U0 W0
N120 G01 Z0	N240 M30

Fonte: COSTA, 2006

Exemplo de um programa para fresadora, referente a Figura 10.



Dados da peça:

- Canais A e C: largura 8 mm e profundidade 1.6 mm;
- Canal B: largura 10 mm e profundidade 1.9 mm;
- Quatro furos de \varnothing 6 mm e profundidade 1.5 mm;
- O material da peça é alumínio e o da ferramenta é aço-rápido;
- Ferramentas a serem usadas:
 - Fresa de topo de \varnothing 10 mm (T04);
 - Fresa de topo de \varnothing 6 mm (T03);
 - Fresa de topo de \varnothing 8 mm (T08).

Figura 10 – Modelo de perfil para fresamento

Tabela 6 – Descrição de um programa CNC (Fresadora)

Programa CNC	
N10 O0011	N250 G03 X26 Y50 R26
N20 G21	N260 G01 Y56
N30 [BILLET X76 Y50 Z20	N270 Z-1.6
N40 [TOOLDEF T04 D10	N280 Y50
N50 [TOOLDEF T03 D6	N290 G02 X0 Y24 R26
N60 [TOOLDEF T08 D8 (médio)	N300 G00 Z5
N70 G91 G28 X0 Y0 Z0	N310 X50 Y-4
N80 M06 T04	N320 Z-0.8
N90 G43 H04	N330 G01 Y0 F120
N100 G90 G00 X-7 Y0 Z3 S2500 M03	N340 G02 X76 Y26 R26
N110 Z-0.95	N350 G01 X82
N120 G01 X0 F120	N360 Z-1.6
N130 X76 Y50	N370 X76
N140 X83	N380 G03 X50 Y0 R26
N150 Z-1.9	N390 G00 Z5 M05
N160 X76	N400 G91 G28 X0 Y0 Z0
N170 X0 Y0	N410 M06 T03
N180 G00 Z5 M05	N420 G43 H03
N190 G91 G28 X0 Y0 Z0	N430 G90 G00 X8 Y42 Z5 S2500 M03
N200 M06 T08	N440 G99 G81 X8 Y42 Z-1.5 R2 F70
N210 G43 H08	N450 X38
N220 G90 G00 X-6 Y24 Z 3 S2500 M03	N460 Y8
N230 Z-0.8	N470 X68
N240 G01 X0 F120	N480 G91 G80 G28 X0 Y0 Z0 M05
Fonte: COSTA, 2006	N490 M30

2.6 Ciclos de usinagem

Os ciclos de usinagem são rotinas de programação, com a finalidade de diminuir as linhas do programa, para situações de repetição de movimento. Como exemplo o ciclo (G74) Ciclo desbaste de perfil simples (Figura 11), com variáveis definidas na Tabela 07.

Tabela 7 – Variáveis do exemplo do ciclo G74

Variáveis	Definição
X	Diâmetro final
Z	Comprimento final
I	Incremento por passada
U1	Recuo angular dos eixos
F	Avanço

Fonte: SILVA, 2004

Programa.

N010 G00 X95. Z2. M08

N020 G74 X50. Z-45. I5. U1 F.25

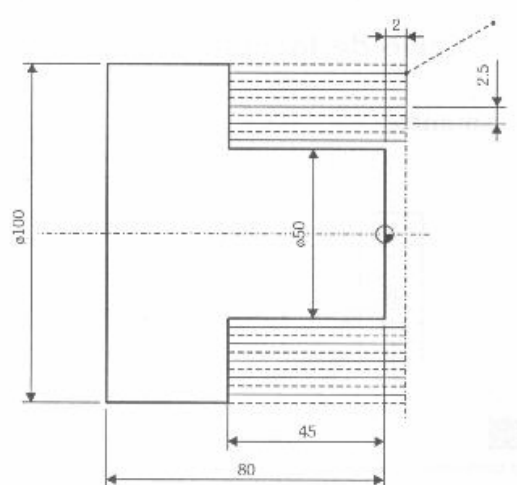


Figura 11 – Ciclo de usinagem G74

Fonte: SILVA, 2004

2.7 Programação Parametrizada do CNC

Programação parametrizada pode ser comparada a qualquer linguagem de programação como as linguagens BASIC, linguagem C ou PASCAL. (Mundo CNC)

Porém, esta linguagem de programação reside no controle do CNC e pode ser acessado ao nível do código G, podem combinar técnicas de programação manuais com técnicas de programação paramétricas.

Características relacionadas aos computadores como as variáveis, aritmética, declarações de lógica, e os *loopings* estão disponíveis nesta linguagem. Como todas as linguagens de programação a programação paramétrica possui várias versões. A mais popular é *Custom Macro B* (usado pela *Fanuc* e controles *Fanuc* compatíveis). Outros incluem *User Task (Okuma)*, *Q Routine (Sodick)*, e linguagem de programação Avançada [APL] (G & L).

Além de ter muitas rotinas relacionadas ao computador, a maioria das versões de programação paramétrica tem rotinas relacionadas ao CNC com relativa profundidade. Por exemplo, macros que permitem ao usuário de CNC ter acesso a muitas propriedades do controle CNC (ferramenta de compensação, posicionamento dos de eixo, alarmes, geração e edição de código G codifica, e proteção de programa) que permite a edição interna do programa CNC.

Os usuários deste conceito se restringem aos construtores de máquinas de usinagem, e fabricantes de controles, pois é quase nula a informação que se obtém sobre isto nos meios acadêmicos a não ser grupos de estudos muito isolados.

Aplicações

· Famílias de peças

Tendo peças semelhantes, porém, com dimensões variáveis, deste modo o programador deverá referenciar em um quadro no desenho as cotas variáveis e propor-las em um programa parametrizado, que será acionado conforme as solicitações das peças a serem produzidas. Fazendo isto, tem-se uma aplicação perfeita para rotinas de usuário.

• *Ciclos fixos (inclusive referenciando um código G)*

Possuindo peças que requeiram operações de usinagem semelhantes pelo menos. Ou talvez deseje que seu controle CNC tivesse mais (ou melhores) ciclos fixos.

Com macros de usuários, podem-se desenvolver rotinas de propósitos gerais para operações como usinagem em linha, padrões de furos de roscas específicas ou entalhes.

• *Movimentos complexos*

Quando o controle CNC é incapaz de gerar um movimento necessário com facilidade.

Executar uma usinagem em linha de precisão, por exemplo, seu controle tem que ter a habilidade para formar um movimento espiralado em XY enquanto forma um movimento linear em Z (movimento helicoidal não bastará neste caso).

A maioria dos controles de CNC não possui interpolação em espiral, com macro de usuário é possível gerar este movimento desejado, permitindo criar suas próprias formas de interpolação.

• *Utilidades*

Macro de usuário pode ajudar reduzir a cronometragem da organização, tempo dos ciclos, tempo de transferência de programa, e em geral, facilitar o uso de seu equipamento. Alguns exemplos de aplicações que se ajustam a esta categoria incluem contadoras de peças, gerenciamento de vida de ferramenta, mordentes automáticos inclusos as máquinas, usando as saídas padrões dos próprios controles.

Paramacros é um recurso de linguagem de programação do CNC MACH8 que oferece ao programador maiores facilidades na geração de seus programas. Através dele é possível:

- Trabalhar com variáveis computáveis;
- Usar funções computáveis em qualquer tipo de bloco;
- Ter acesso a certos parâmetros modais do sistema para computação;
- Utilizar operadores e expressões aritméticas para computação;

- Efetuar desvios adicionais, chamadas de sub-rotinas e sub-programas dependendo do resultado de uma função lógica;
- Programar sub-rotinas e subprogramas parametrizados;
- Programação de Ciclos Fixos parametrizados;
- Definir Prompts para auxiliar na geração e execução do programa, através dos quais será feita a entrada dos parâmetros para o programa principal;
- Cálculos utilizando funções lógicas e aritméticas.

Expressões e operadores

A programação com linguagem Paramacros permite valores numéricos para quase todas as funções, com exceção de N e G, em qualquer tipo de bloco, e de P, num bloco com G39 ou G79.

Qualquer outra função pode ser computada através do argumento de uma expressão aritmética, através do uso dos parênteses. (Manual ROMI, 1998)

Por exemplo:

$$X (2.53*6.2)$$

Note que não existe espaço em branco entre os caracteres.

A expressão aritmética pode ter qualquer tamanho, desde que caiba na página *Prompt* Editor. Da mesma forma pode-se ter um bloco com quantas funções desejar. É permitido ainda escrever uma expressão com alinhamento de parênteses.

Por exemplo:

$$L ((2.7+14.123)* 17)$$

As ordens da precedência das operações são: “!” “depois “*“ e “/“ e por fim” +“e “-“.

Os parênteses têm precedência sobre qualquer um dos operadores. Expressões com operadores sem ordem de precedência sobre os outros. São resolvidos da esquerda para direita.

Operadores Unitários:

Os operadores unitários + e - só podem ser usados numa expressão fechada por parênteses. Se for uma expressão alinhada, todos os parênteses abertos devem ser fechados.

Operadores Binários:

- + Adição
- Subtração
- * Multiplicação
- / Divisão
- ! Exponenciação (inclusive potências fracionárias)

Os operadores não podem aparecer consecutivamente, eles devem ser separados utilizando parênteses.

Por exemplo:

$3!(-4)$ é correto, enquanto que $3!-4$ não é correto.

O *Prompt* Editor fornece um significativo auxílio na programação e checagem de sintaxe. A sintaxe é checada a priori, assim que for pressionada a tecla EOB. A sintaxe é checada toda vez que o bloco é alterado, basta pressionar ENTER. A assistência à programação inclui o balanceamento de parênteses, o qual só é suspenso quando estiver sendo exibida a mensagem ERRO DE SINTAXE.

A checagem da sintaxe é feita varrendo o bloco do começo ao fim. Assim que for encontrado um erro. Este é exibido na linha de comentário, ao pé da tela, permanecendo o cursor sobre o caractere errado. Uma mensagem de erro “ERRO DE SINTAXE” aparecerá então na parte de baixo da tela.

Após a correção do bloco basta pressionar ENTER para que o controle cheque novamente. Durante a execução, todos os campos do bloco são computados antes de serem executados. Se um bloco contém a função L (função de repetição), as demais funções não são recomputadas a cada repetição do bloco. Da mesma forma. G39 e G79 não têm seus valores computados enquanto todo o bloco tenha sido avaliado.

A precisão de uma expressão aritmética é de sete dígitos, de 0.0000001 à 9999999., todavia o resultado pode ser truncado pelo formato da função, que usará esse resultado.

Funções lógicas e aritméticas

Com Paramacros é possível acessar as funções lógicas e aritméticas, as quais podem ser usadas numa expressão, (Tabela 8). Estas funções sempre serão seguidas pela expressão entre parênteses.

Tabela 8 – Funções Aritméticas

SIGLA	DESCRIÇÃO
ABS	Valor absoluto
INT	valor interno (Valor à esquerda do ponto decimal)
FRA	valor fracionário (Valor à direita do ponto decimal)
LOG	Logaritmo na base 10
LN	Logaritmo na base E
EXP	Exponenciação na base E
SQR	raiz quadrada
SIN	seno do ângulo em graus
COS	cosseno do ângulo em graus
TAN	tangente do ângulo em graus
ASN	Arc-seno com resultado em graus
ACS	Arc-cosseno com resultado em graus
ATN	Arc-tangente com resultado em graus

Fonte: Manual de programação ROMI

EXEMPLOS:

A (ATN(3.5/4.2))

a variável local "A" assumirá o valor 39.80557

W (SQR(3.532!2+2.785 !2))-0.2714

a variável local "W" assumirá o valor 4.226516

As funções lógicas Tabela 9, assumem 0 (zero) se o resultado for falso e assumem 1 (um) se o resultado for verdadeiro. Esse valor pode então ser usado em cálculos aritméticos. O resultado da função lógica pode também resultar desvios condicionais, chamada de sub-rotina ou de subprograma. A comparação lógica de duas variáveis tem curacidade 10, por exemplo:

EQ (0.0000001) será considerado verdadeiro.

Tabela 9 – Funções Lógicas

SIGLA	DESCRIÇÃO
EQ	Igual a zero
NE	Diferente de zero
GT	Maior que zero
LT	Menor que zero
GE	Maior ou igual a zero
LE	Menor ou igual a zero

Fonte: Manual de programação ROMI

Exemplos:

A (EQ(27.01-22.2))

A variável “A” assumirá o valor 0, pois a função lógica é falsa, já que 27.01-22.2 é diferente de zero.

A (EQ(22-3-19))

A variável “A” assumirá o valor 1, pois a função lógica é verdadeira, já que 22-3-1-9 é exatamente zero.

Variáveis

As variáveis são letras com as quais o programador pode atribuir valores, utilizando-as para realizar cálculos ou desvios no programa.

Na programação com PARAMACROS são disponíveis três tipos de variáveis para se trabalhar. As variáveis são referenciadas por duas letras, das quais a primeira corresponde ao tipo e a segunda corresponde a variável propriamente dita. As variáveis especificadas com única letra são assumidas como tendo prefixo “L”, indicando tratar-se de variáveis locais. Portanto, “F” é o mesmo que “LF”.

Os três tipos de variáveis a disposição são:

LOCAL: LA-LZ ou A-Z (exceto G,E,H,N e P)

GLOBAL: GA-GZ (exceto G, E, H, N e P)

SISTEMA: SA-SZ e SA -SZ (apenas para leitura)

Variáveis Locais: são variáveis que estão disponíveis somente no programa corrente ou sub-rotina. Pode-se definir Variáveis Locais em até 6 níveis: o programa principal e os 5 níveis de sub-rotinas permitidos. Quando um programa sai de determinado nível, as Variáveis Locais para esse nível são perdidas.

Variáveis Globais: são variáveis que estão disponíveis durante todo o programa e suas sub-rotinas e subprogramas.

Variáveis do Sistema: são os parâmetros atuais do sistema, podendo ser usadas em computação. Elas não podem ser usadas via programa. Exceto para a segunda letra das variáveis do sistema, o fato de se usar maiúsculas ou minúsculas não tem importância.

A precisão de todas as variáveis Local e Global são 7 dígitos, de 0.0000001 à 9999999. A precisão das variáveis do Sistema depende do parâmetro específico a que ela se refere (por exemplo, o número de um programa pode ir de 0 a 250, já a coordenada absoluta do eixo X pode ir de 0.000 à 9999.999).

Definição de variável local e sub-rotina paramétrica G39

A função G39 tem duas aplicações, definição de variável Local e programação de Sub-Rotinas e Sub-Programas parametrizados. (Manual ROMI, 1998)

Um bloco com função G39 que tenha chamada de sub-rotina ou subprograma através das funções E, H estará chamando uma sub-rotina ou subprograma parametrizado. Todas as letras, exceto N, G, E, H e P podem armazenar valores para a correspondente variável local da sub-rotina ou subprograma chamado.

Exemplo:

G39 P7 X2.75 Y(LOG(2.65/2))

Esse bloco chama o subprograma número 7 e especifica para o mesmo que as variáveis Local “LX” valerá 2.75 e “LY” valerá 0.2116229 (que corresponde ao logaritmo de (2.65/2) na base 10). Assim dentro do subprograma esses valores paramétricos poderão ser usados.

Por exemplo, um comando do tipo G01 X(LX), dentro do subprograma, provocará um deslocamento no eixo X até a coordenada 2.75, ou ainda, um comando do tipo G81 Z(2.3+LY) provocará um ciclo fixo de furação até a coordenada 2.511 na direção do eixo Z.

Um bloco G39 que não contenha uma chamada de sub-rotina ou subprograma servirá para especificar variáveis locais. Assim, todas as funções, exceto N, G, E, H, P podem armazenar valores para correspondente variável local.

Exemplo:

G39 J27.5 U(4.5/2) B(ASN(0.5)+25)

Este bloco especifica que a variável “LJ” vale 27.5, “LU” “vale 2.25 e” LB “vale 55 (que é o arc-seno de 0.5 mais 25)”. Esses valores podem ser referenciados dentro do programa, sub-rotina ou subprograma corrente, usando variáveis locais.

Exemplo:

G01 X(LJ) Y(LB+5) será executado como G01 X27.5 Y60.

Definição de variáveis global e ciclos fixos parametrizados

A função G79 tem duas finalidades, definir variáveis Globais e Ciclos Fixos parametrizados.

Um G79 que contenha uma chamada de sub-rotina ou subprograma, usando E, H e/ou P, define um ciclo fixo parametrizado. Todas as letras, exceto N, G, E, H e P, podem armazenar valores para a correspondente variável Global da sub-rotina ou subprograma chamado.

Exemplo:

G79 P7 X2.27 Y(LQG(2.65)/2)

Este bloco especifica uma chamada ao subprograma número 7, o qual funcionará como um ciclo fixo. A variável local “LX”, para esse subprograma, valerá 2.27 e “LY” valerá 0.2116 (que é o logaritmo de (2.65/2)). Dessa forma esses valores poderão ser usados dentro de subprograma, por exemplo, um comando G01 X(LX) posicionará o eixo “X” na coordenada 2.27 ou ainda, um comando G81 Z(2.3+LY) executará um ciclo de furação com “Z” igual a 2.511.

Note que o G79 não é recomputado toda vez que o Ciclo Fixo é executado, apenas quando o G79 é executado sozinho. O mesmo ocorre com outros Ciclos Fixos como G81 Z(PZ).

Da mesma forma, G79 H100 E200 L7 ativará um ciclo fixo através de um salto para uma sub-rotina que começa em N100 e termina em N200, podendo-se fazer uso de uma variável Local, por exemplo, "LL" para referenciar o valor programado junto a função, "L" (Função de repetição).

O fato do G79 ativar um Ciclo Fixo traz o seguinte inconveniente, a cada movimento rápido no programa principal, após um comando G79 com chamada de sub-rotina ou subprograma, fará com que esta sub-rotina ou subprograma seja executado como se fosse um ciclo fixo.

Exemplo:

G79 J27.5 U(4.5/2) B(ASN(0.5)+25.)

Este bloco atribui a variável Global, GJ "o valor 27.5, "GU "o valor 2.25 e "GB" o valor 55. Estes valores podem ser usados em qualquer lugar do programa ou de seus subprogramas, através das variáveis globais, por exemplo: G01 R(GJ) A(GB/5) será executado como se fosse um bloco G01 R27.5 A11.

Desvio condicional

Com Paramacros é possível programar desvios condicionais para uma sub-rotina ou subprograma com base no resultado de uma função lógica.

Se um código "M" precede uma expressão com uma função lógica. Então o desvio é baseado no resultado dessa função lógica. Se a função for verdadeira o desvio ou chamada ocorre, se ela é falsa, o controle processará o próximo bloco da seqüência. (Manual ROMI, 1998)

Exemplo de programa parametrizado do CNC, com a linguagem "Custom macro B".

A linguagem "Custom macro B" de programação parametrizada tem o nome das variáveis como números mostrados na Tabela 10. Por exemplo, a variável que corresponde ao diâmetro do furo corresponde ao número #100.

Tabela 10 – Programa exemplo com "Custom macro B"

Custom macro B	Descrição
O0001	(número de Programa)
#100=1.	(diâmetro final do furo)
#101=3.0	(X posicionam do furo)
#102=1.5	(Y posicionam do furo)
#103 = .5	(profundidade do furo)
#104=400	(velocidade em RPM)
#105=3.5	(avanço em IPM)
#106=3.	(número de compensação do comprimento da ferramenta)
#107=2.0	(diâmetro do furo)
G90 G54 S#104 M03	(seleção do modo absoluto, coordenada de sistema, rotação inicial)
G00 X#101 Y#102	(posição corrente X e Y do centro do furo)
G43 H#106 Z.1	(aciona a compensação de comprimento da ferramenta, para chegar ao Z corrente)
G01 Z-#103 F[#105 / 2]	
Y[#102 + #107 / 2 - #100 / 2] F#105	
G02 J-[#107 / 2 - #100 / 2]	
G01 Y#102	
G00 Z.1	
M30	

Fonte: <http://www.mundocnc.com.br/>

Exemplo de programa parametrizado do CNC, com MACH.

A linguagem MACH de programação parametrizada tem o nome das variáveis representadas por letras como mostra a Tabela 11. Por exemplo, a variável que corresponde ao raio maior da elipse corresponde a LA.

O programa usa a equação da elipse para gerar a geometria e distribui em reproduções (Figura 12).

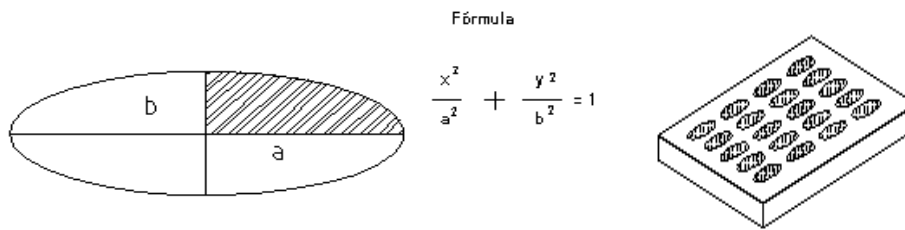


Figura 12 – Equação da elipse para cópias retangulares
Fonte: (IATAGAN,2007)

Tabela 11 – Programa para gerar cópias retangulares de elipse

MACH	
;ELIPSES	G00X(LX+LV)Y(-LY-LW)
G99	G01Z06.F800
G90	G01Z(LZ)
G17	N100
GZO	G39Y(SQR((LA*LA-LX*LX)*LB*LB/LA/LA))
T02M06	G01X(LX+LV)Y(-LY-LW)
O2S2000M3	G39X(LX+LI)
G39A37.0000B17.900V+081.0W+85.0X0.0Y0.	M(LT(LX-LA))H100
0I1.0	N200
;A = raio-maior	G39Y(SQR((LA*LA-LX*LX)*LB*LB/LA/LA))
;B = raio- menor	G01X(LX+LV)Y(+LY-LW)
;V = posicionamento	G39X(LX-LI)
-no-eixo-x	M(GE(LX+LA))H200
;w= posicionamento-no-eixo-y	G39V(LV+LA*2+LR*2+LD)
;X-Y-Eixos	G39O(LO-1)
;l=Incremento.	G00Z20.
G39R5.0D10.0O4.0Q5.0S8.0Z06	M(GT(LO))H050
;R=raio-fresa	G39V(LT)O(LU)
;D=distancia-entre-cavidade	G39W(LW+LB*2+LR*2+LD)
;O=	G39Q(LQ-1)
;Q=	G00Z20.
;S= Número-de-passadas em (Z)	M(GT(LQ))H050
;Z=Eixo	G39V(LT)O(LU)W(LK)Q(LL)
;	G39S(LS-1)Z(LZ-1)
G39T(LV)U(LO)K(LW)L(LQ)A(LA-LR)B(LB-LR)	G00Z20.
G00X0.Y0.	M(GT(LS))H050
G00Z20.F1000.	GZO
N050	
G39X(-LA)	
G39Y(SQR((LA*LA-LX*LX)*LB*LB/LA/LA))	
Fonte: (IATAGAN,2007)	

M02; (fim do programa)

O próximo programa também feito com MACH, realiza a trajetória de um arco do ângulo inicial ao final, calculando ponto a ponto, usando a linguagem parametrizada (Figura 14). A Tabela 12 mostra as variáveis usadas no programa. A Tabela 13 faz uma relação das equações de cálculo para pontos iniciais e finais de (X) e (Y) para pontos determinados. A Tabela 12 mostra o programa descrito para realizar o arco de circunferência.

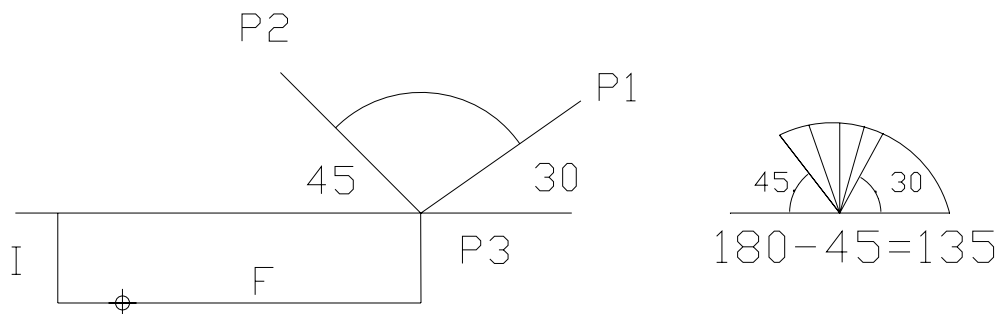


Figura 13 - Trajetória em arco com programação parametrizada
Fonte: Manual ROMI, 1998

Tabela 12 - Variáveis do programa

Variáveis	Valor	Definições
R	100	Raio do arco
T	50	Ângulo inicial
U	30°	Ângulo final
F	45°	Posição x centro do arco
I	50	Posição y centro do arco

Fonte: Manual ROMI, 1998

Tabela 13 - Definindo equações e as posições

Variáveis	Definição	Equações	Equações
A	posição x inicial	$100+(\cos 30^{\circ} \times 50)$	$A(LF+(LR \times \cos(LT)))$
B	posição x final	$100-(\cos 45^{\circ} \times 50)$	$B(LF-(LR \times \cos(LU)))$
C	posição y inicial	$50+(\sin 30^{\circ} \times 50)$	$C(LI+(LR \times \sin(LU)))$
D	posição y final	$50+(\sin 45^{\circ} \times 50)$	$D(LI+(LR \times \sin(LT)))$

Fonte: Manual ROMI, 1998

Tabela 14 - Programa parametrizado para geração de arcos

Programa	Dfinitiones
G39 R50. T30. U45. F100. I50.	Definindo variáveis locais
G39 A(LF+(LRxcos(LT))) C(LI+(LRxsin(LU)))	Definindo A e C
G39 B(LF-(LRxcos(LU))) D(LI+(LRxsin(LT)))	Definindo B e D
G00 x (LA) Y (LC)	Posicionamento rápido
Z2	Aproximação rápida
G01 z- .2 F500	Movimento em Z

Fonte: (Manual ROMI, 1998)

Exemplo de programa parametrizado do CNC, com HEIDENHAIN.

O programa parametrizado do CNC, para realizar uma geometria de elipse com linguagem HEIDENHAIN (Figura 14), esta listado na Tabela 15. Na linguagem HEIDENHAIN as variáveis são armazenadas com a letra Q mais um número correspondente. Existem ciclos definidos que podem ser acionados no decorrer do programa.

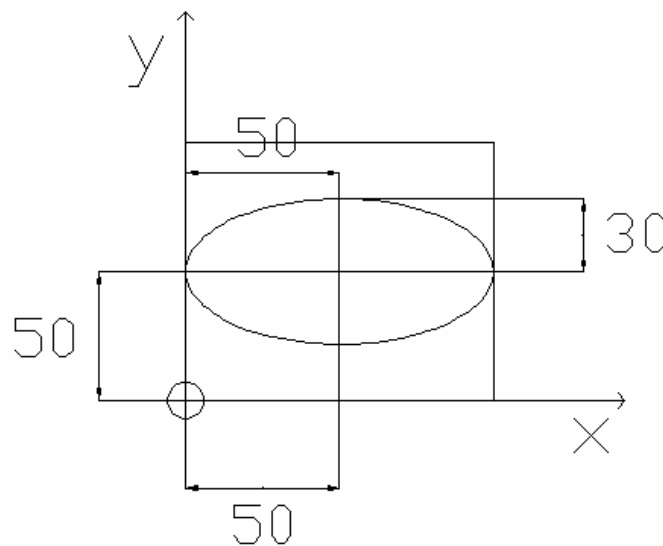


Figura 14 - Elipse gerada pelo programa

Tabela 15 - Programa para gerar elipse

HEIDENHAIN	
1 FN 0: Q1=+50 Centro em x	26 Q35=(Q6 - Q5) / Q7 Incremento angular
2 FN 0: Q2=+50 Centro em y	27 Q36=Q5 Cópia angulo inicial
3 FN 0: Q3=+50 Semi eixo em x	28 Q37=0
4 FN 0: Q4=+30 Semi eixo em y	29 Q21=Q3* COS Q36 Ponto inicial x
5 FN 0: Q5=+0 Angulo inicial	30 Q22=Q4* SIN Q36 Ponto inicial y
6 FN 0: Q6=+360 Angulo final	31 L X+Q21 Y+Q22 R0 F MAX M3
7 FN 0: Q7=+40 Número steps	32 L Z+Q12 R0 F MAX
8 FN 0: Q8=+0 Rotação da elipse	33 L Z-Q9 R0 FQ10
9 FN 0: Q9= +5	34 LBL 1
10 FN 0: Q10=+100	35 Q36= Q36 + Q35 Update ângulo
11 FN 0: Q11=+350	36 Q37= Q37 + 1
12 FN 0: Q12= +2	37 Q21= Q3 * COS Q36
13 BLK FORM 0.1 Z X+0 Y+0 Z-20	38 Q22= Q4 * SIN Q36
14 BLK FORM 0.2 X+100 Y+100 Z+0	39 L X+Q21 Y+Q22 R0 FQ11
15 TOOL DEF 1 L+0 R+2.5 Ferramenta	40 FN 12: IF +Q37 LI +Q7 GOTO LBL 1
16 TOOL CALL 1 Z S4000	41 CYCL DEF 10.0 ROTATION
17 L 2+250 R0 F MAX	42 CYCL DEF 10.1 ROT+0
18 CALL LBL 10	43 CYCL DEF 7.0 DATUM SHIFT
19 L Z+100 R0 FMAX M2	44 CYCL OFF 7.1 x+0
20 LBL 10 Subprograma 10	45 CYCL OFF 7.2 Y+0
21 CYCL DEF 7.0 DATUM SHIFT	46 L Z+Q12 R0 F MAX
22 CYCL DEF 7.1 X+Q1	47 LBL 0
23 CYCL DEF 7.2 Y+Q2	48 END PGM ELLIPSE MM
24 CYCL DEF 10.0 ROTATION	
25 CYCL DEF 10.1 ROT+Q8	

Fonte: Manual Conversational Programming HEIDENHAIN

3 METODOLOGIA

Com o objetivo de demonstrar a metodologia proposta, são apresentados neste capítulo exemplos de criação de rotinas com a linguagem parametrizada do CNC MACH, para serem utilizadas como ciclos de usinagem. A título de comparação com a técnica de sistemas CAD/CAM, programas CN semelhantes também são gerados por esses sistemas.

O equipamento usado para testes foi um centro de usinagem vertical ROMI Discovery 308 (Figura 15), instalada no NAFA da UFSM.



Figura 15 – Centro de usinagem vertical ROMI Discovery 308

3.1 Cavidade Retangular

O primeiro caso é a elaboração de uma rotina para usinagem de cavidade retangular, embora a máquina já possua uma rotina para esse fim, essa nova função traz uma alternativa ao usuário quanto à seqüência de corte. Enquanto a função disponibilizada pelo comando executa uma operação que move a ferramenta em trajetória *offset* (Figura 16), a proposta gera um movimento alternativo da ferramenta (Figura 17). Este movimento é de zigue zague até completar a condição de limite de profundidade. Considerando que estas rotinas desenvolvidas são de relativa simplicidade, onde o foco se manteve na verificação do traçado da ferramenta, não foi acrescentado nos módulos a compensação da ferramenta e ciclos de acabamento.

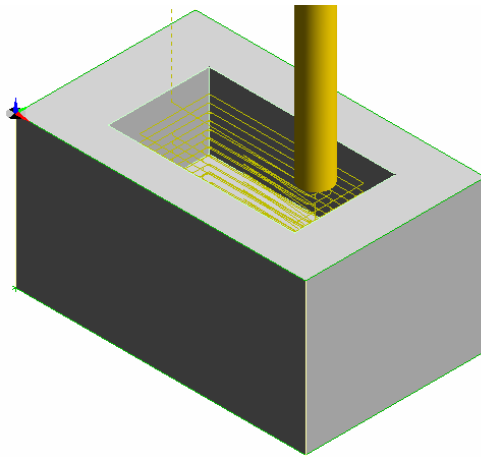


Figura 16 - Movimento da ferramenta tradicional para rebaixo

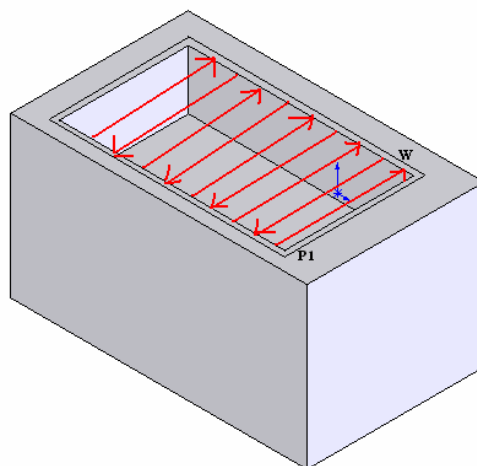


Figura 17 - Movimento da ferramenta proposto para rebaixo

Para a elaboração do programa CNC paramétrico com o objetivo de realizar a cavidade retangular, precisa-se estabelecer as variáveis necessárias ao sistema (Tabela 16), e identificado na Figura 18.

Tabela 16 - Variáveis do programa cavidade retangular

Variáveis	Definição
X	Comprimento
Y	Largura
Z	Profundidade
W	Largura de Corte
U	Profundidade de Corte

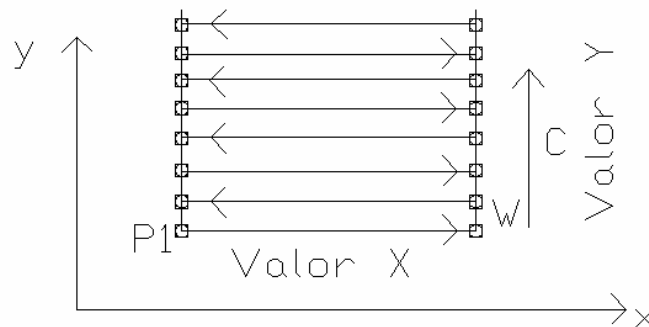


Figura 18 - Movimento da ferramenta com “w” largura de corte

Algoritmo

A ferramenta é posicionada pelo operador no ponto P1 (ponto inicial para o deslocamento da ferramenta), conforme identificado na Figura 18 e representado em forma de fluxograma na Figura 19. No próximo movimento, a ferramenta avança em (Z) até definir a profundidade de corte (U). No plano XY ocorre o avanço com o valor de (X), que corresponde ao comprimento da cavidade. Ao valor da variável que armazena o valor de (X) é aplicado o inversor (B). Calcula o valor do novo (Y). É verificada a condição de (Y) máximo ter sido atingido, se a condição não for satisfeita, o avanço é feito em (Y) com o valor da largura de corte (W). É realizado novo avanço em (X) com o sentido invertido. Nesta condição são realizadas tantas passadas quanto for necessário para satisfazer a condição de (Y) máximo. Quando a condição de (Y) máximo for alcançada, o ciclo é encerrado. È feito o

ultimo movimento no valor de (Y) máximo. Concluída esta fase do desbaste, a cavidade está com uma profundidade igual à profundidade de corte (U). Para realizar os de mais desbastes na direção do eixo (Z), a ferramenta é novamente posicionada na coordenada (Y) do ponto (P1). Esta posição pode ser coincidente a (P1), como pode ser oposta, dependendo do lado que finalizou a ferramenta no seu avanço. Verifica se a profundidade (Z) foi alcançada, se não foi, redireciona para o ponto de partida e repete a rotina.

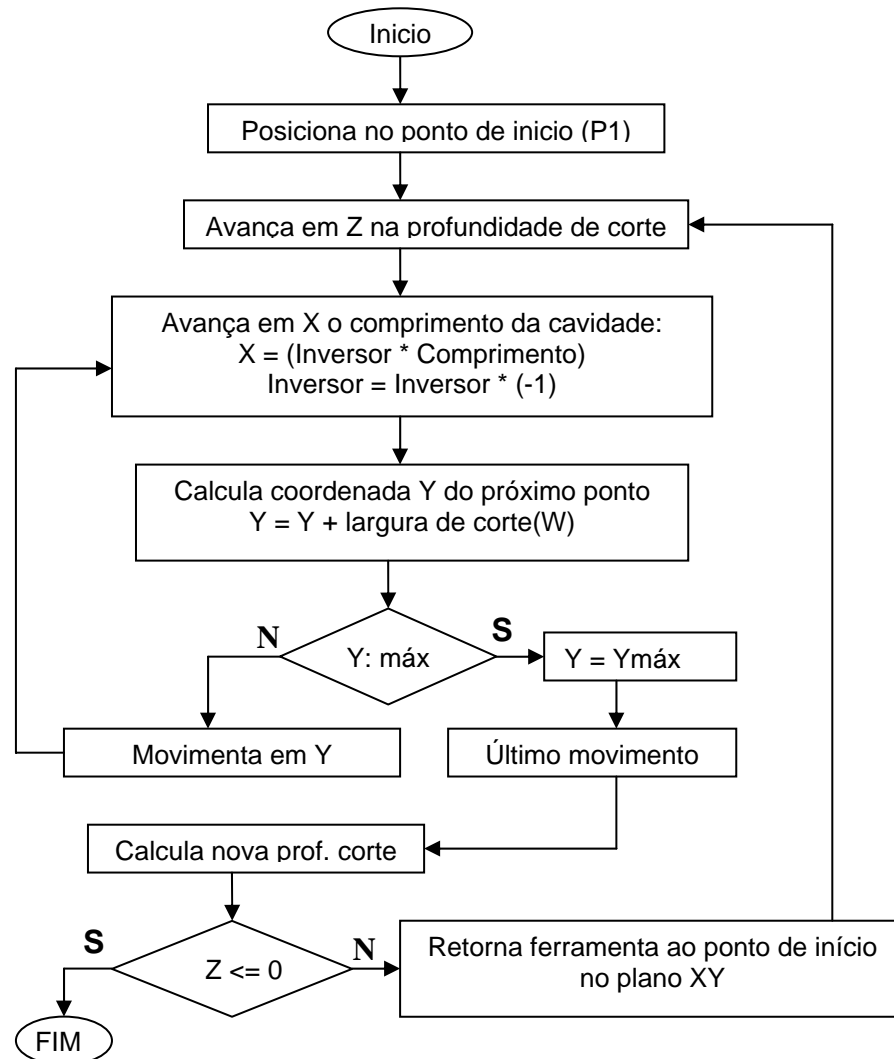


Figura 19 - Fluxograma Cavidade Retangular

3.2 Rampa

O segundo estudo é uma rotina de programação parametrizada para usinar uma rampa. A Figura 20 mostra o trabalho da ferramenta, gerado graficamente. Na Figura 21, pode-se observar o traçado da ferramenta.

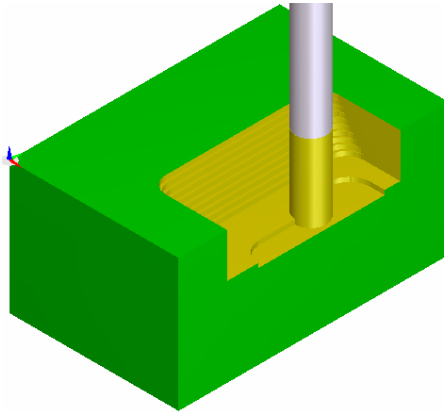


Figura 20 - Movimento da ferramenta para Rampa (usinando)

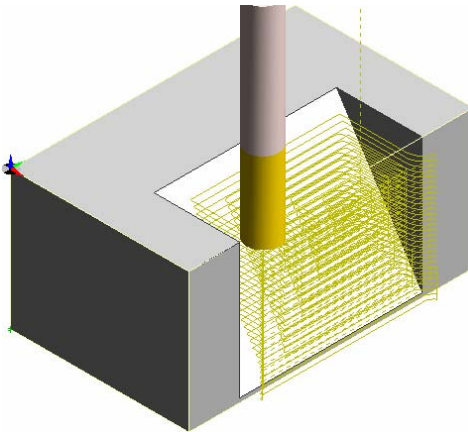


Figura 21 - Movimento da ferramenta para Rampa (traçado)

O movimento da ferramenta proposto e o limite do plano inclinado é mostrado na Figura 22.

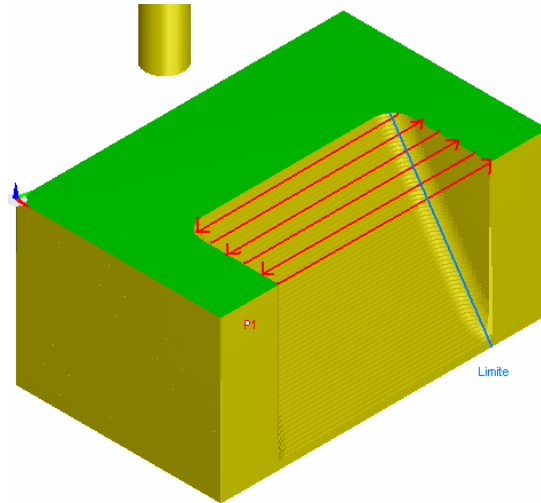


Figura 22 – Seqüência de movimento para a rampa

Para geração do programa CNC parametrizado para usinagem da rampa as variáveis usadas estão relacionadas na Tabela 17 e identificadas nas Figuras 23 e 24.

Tabela 17 - Variáveis para o programa de rampa

Variáveis	Descrição
X	Comprimento em x
Y	Comprimento em y
Z	Altura
W	Largura de corte
U	Profundidade de corte

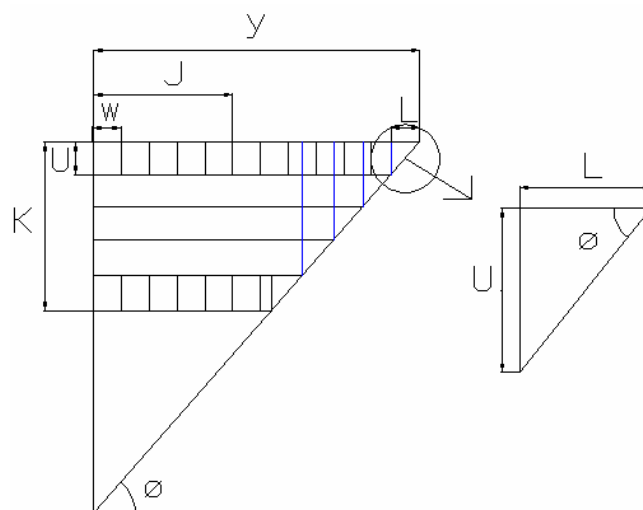


Figura 23 - Vista lateral (Esquema do movimento da ferramenta)

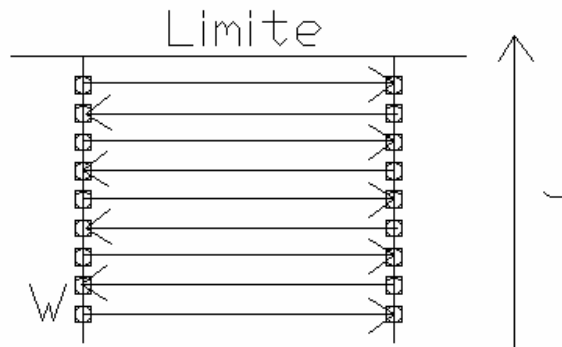


Figura 24 - Vista superior (Esquema do movimento da ferramenta)

Algoritmo

A ferramenta em seu deslocamento avança do ponto inicial em (Z), na profundidade de corte estabelecida por (U), representado no fluxograma da Figura 25. O programa realiza o cálculo para determinar o valor de (Y) em relação ao final da rampa ($Y=Y-L$), pois a linha que determina o final da rampa é inclinada (Figura 23). O valor é calculado pelas fórmulas, onde a variável (T) é calculada pela fórmula (2), e a variável (L) em relação de (T) pela fórmula (3). Ocorre o avanço no valor de (X). É aplicado o inversor para mudar o sentido de (X). Calcula-se o valor do novo valor de avanço (Y). Se a condição de (Y) máximo não for alcançada, avança (Y) com a largura de corte estabelecida (W). Calcula-se o novo valor limite para (Y) e repete o ciclo. Alcançado o valor de (Y) máximo, realiza último movimento em (Y) máximo, retorna para o ponto de início do plano (X,Y). Se a condição de (Z) máximo não for estabelecida, redireciona-se para o início no plano XY e começa novamente a rotina.

$$\text{Tan}(\theta) = \frac{U}{L} \quad (1)$$

$$T = \text{Tan}(\theta) = \frac{LU}{LL} \quad (2)$$

$$L = \frac{LU}{\text{Tan}(\theta)} = \frac{LU}{LT} \quad (3)$$

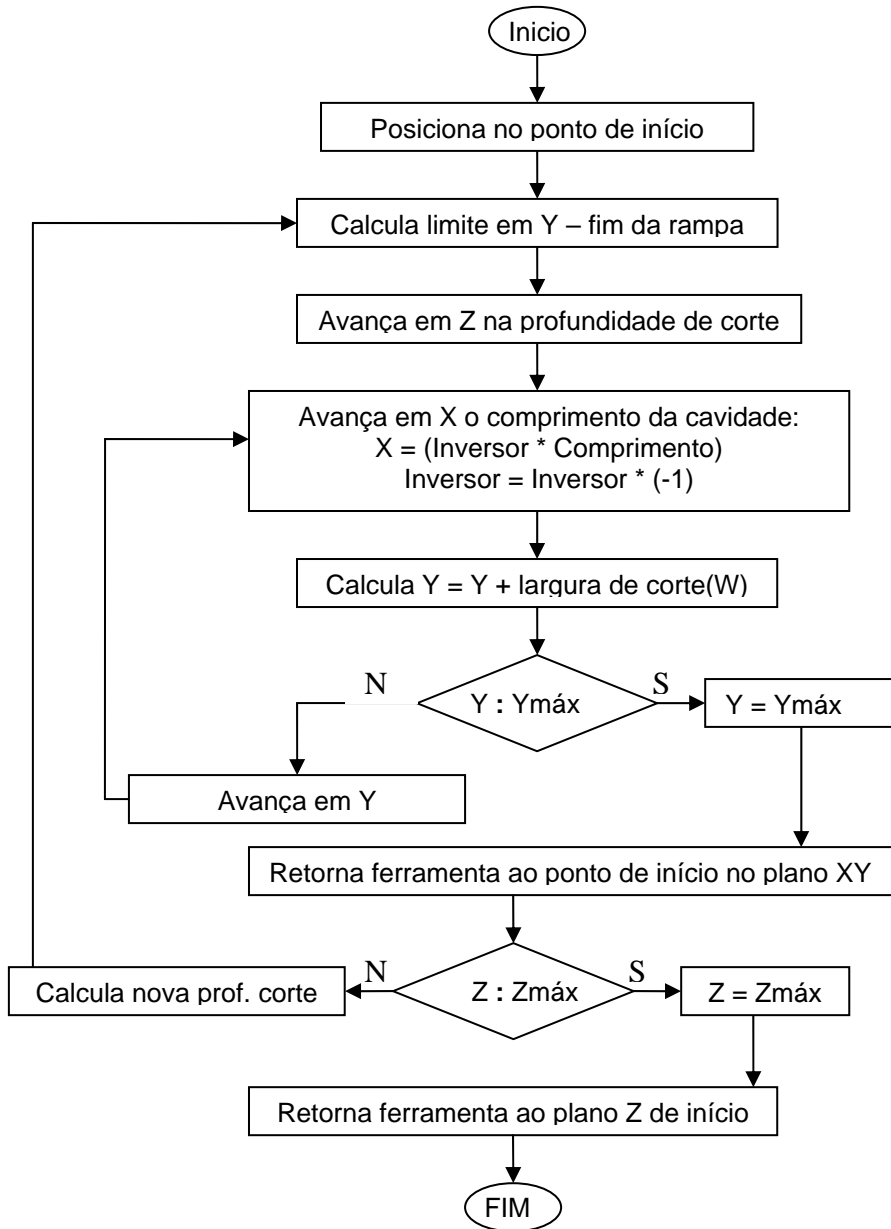


Figura 25 – Fluxograma para a rampa

3.3 Cavidade Elíptica

Este estudo compreende uma rotina para gerar uma elipse com profundidade estabelecida, utilizando a programação parametrizada do CNC, Figura 26.

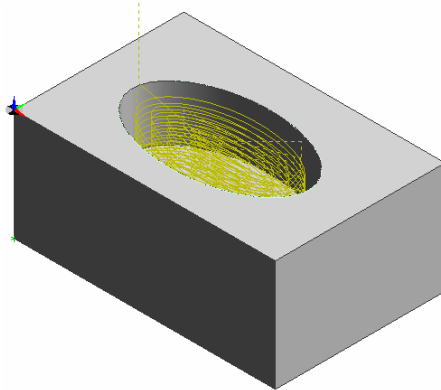


Figura 26 - Trajetória da ferramenta para rebaixo elíptico

A Figura 27 mostra a geometria para o procedimento que determina a trajetória da ferramenta em forma de elipse com uma profundidade definida.

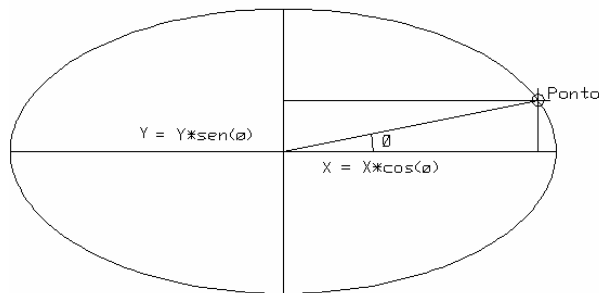


Figura 27 - Cálculo dos pontos da elipse

O ponto “P” da Figura, em relação ao centro da elipse, é calculado por:

$$Para(X) = Raiomaior * (\cos(\theta)) \quad (4)$$

$$Para(Y) = Raiomenor * (\sen(\theta)) \quad (5)$$

As variáveis necessárias para a definição do algoritmo estão relacionadas na Tabela

Tabela 18 - Variáveis para o programa elipse

Variáveis	Definições
S	Incremento Z
J	Contador Z
M	Raio maior
S	Raio menor
R	Variável raio maior
C	Variável raio menor
A	Ângulo inicial
B	Ângulo final
I	Incremento angular
D	Decremento raios
Z	Altura Z

Algoritmo

Para a usinagem de uma cavidade em forma de elipse, a ferramenta percorre todo o traçado da elipse com raio maior, ponto a ponto, avança em relação ao centro o valor do raio maior da elipse (X), que será o ponto de partida. Os movimentos estão indicados no fluxograma da Figura 28. Avança o valor de (Z) definido pela profundidade de corte (S). Calcula os valores para (X) e (Y) para o ângulo inicial (A) pelas equações (4) e (5). Avança para os valores calculados (X,Y). Se a condição de o ângulo (A) ser maior ou igual ao ângulo (B) não for satisfeita, ocorre um incremento (I) no ângulo (A) e recalculam-se novamente os valores de (X) e (Y) para este novo ângulo (A+I). Avança para os novos valores de (X) e (Y). Quando a condição anterior for satisfeita, avança para (X,Y) com relação ao ângulo (B) para completar a elipse. Se a condição de o raio da elipse ser menor ou igual à zero, tiver como resposta negativo, a variável do ângulo (A) é zerada. Calculam-se os novos raios da elipse com decremento de (D). Avança em (X) para a nova posição e começa o ciclo da elipse novamente, até que a condição do raio da elipse seja satisfeita então a ferramenta faz um ultimo movimento para (Y) máximo e retorna ao plano XY inicial, entra em (Z) e começa a rotina novamente, até atingir a condição de (Z) máximo onde terá fim o programa.

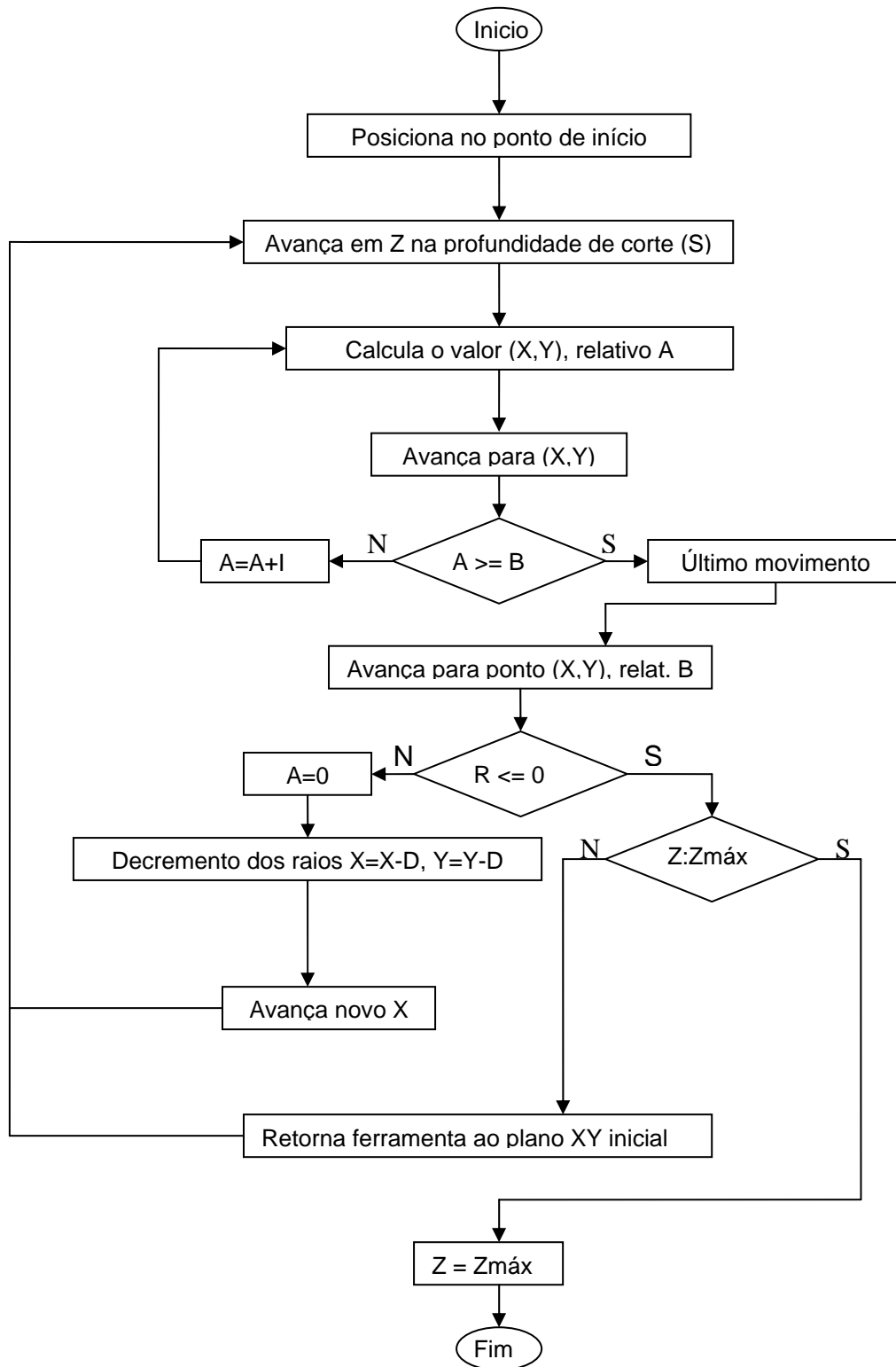


Figura 28 – Fluxograma da elipse

3.4 Cavidades Elípticas Múltiplas

Este estudo é uma ampliação do módulo da elipse, para desenvolver uma série de desenhos num sistema de cópias múltiplas circulares conforme mostrado na Figura 29. Este estudo tem como finalidade testar a rotina feita para o item 3.4.

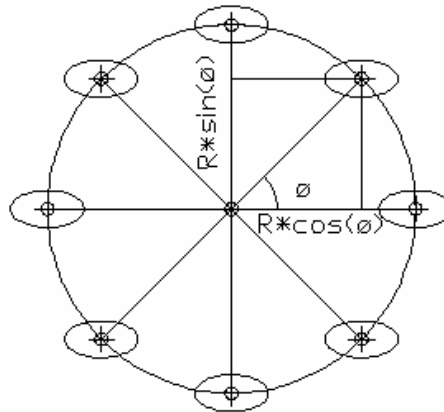


Figura 29 – Esquema para o cálculo da cópia múltipla de elipse

O ponto central de cada elipse é calculado em função do ângulo e o centro da circunferência maior, fórmulas 6 e 7. A Figura 30, mostra o traçado geométrico da ferramenta e as variáveis necessárias para o algoritmo são mostradas na Tabela 19.

$$X = R * \cos(\phi) \quad (6)$$

$$Y = R * \sin(\phi) \quad (7)$$

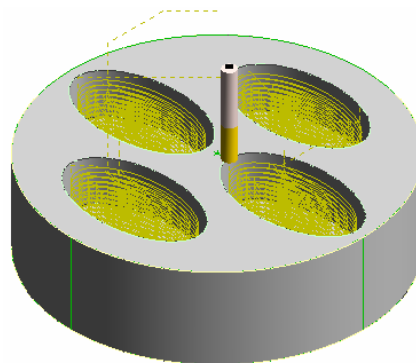


Figura 30 – Trajetória da ferramenta para gerar cópias múltiplas de elipse

Tabela 19 - Variáveis para o programa copia múltiplas de elipse

Variáveis	Definições
W	Raio
V	Incremento angular
H	Ângulo inicial
Y	Ângulo final
S	Incremento Z
J	Contador Z
M	Raio maior
N	Raio menor
R	Variável raio maior
C	Variável raio menor
A	Ângulo inicial
B	Ângulo final
I	Incremento angular
D	Decremento raios
Z	Altura Z

Algoritmo

Esta rotina tem como finalidade, calcular sobre uma circunferência, pontos com distâncias angulares iguais, para serem as origens de elipses geradas pela chamada da rotina cavidade elíptica item 3.4. Na seqüência a ferramenta posicionar-se no centro da circunferência base, é realizado o cálculo para determinar os valores de (X) e (Y), primeiramente utilizando o ângulo inicial (H). Avança a ferramenta no valor de (X), é feita a chamada da rotina da elipse. Ocorre o incremento do ângulo inicial (H). Após incrementar a posição angular, a condição desse valor é analisada quanto ao ângulo final. Se a resposta da condição do ângulo inicial (H) ser maior ou igual ao ângulo final (Y), for negativa, são recalculados novamente os valores de (X) e (Y) para o novo ângulo e chamada a rotina da elipse. Sendo a resposta para a condição anterior positiva, o programa é finalizado.

O fluxograma da cavidade elíptica múltiplas é mostrado na Figura 31.

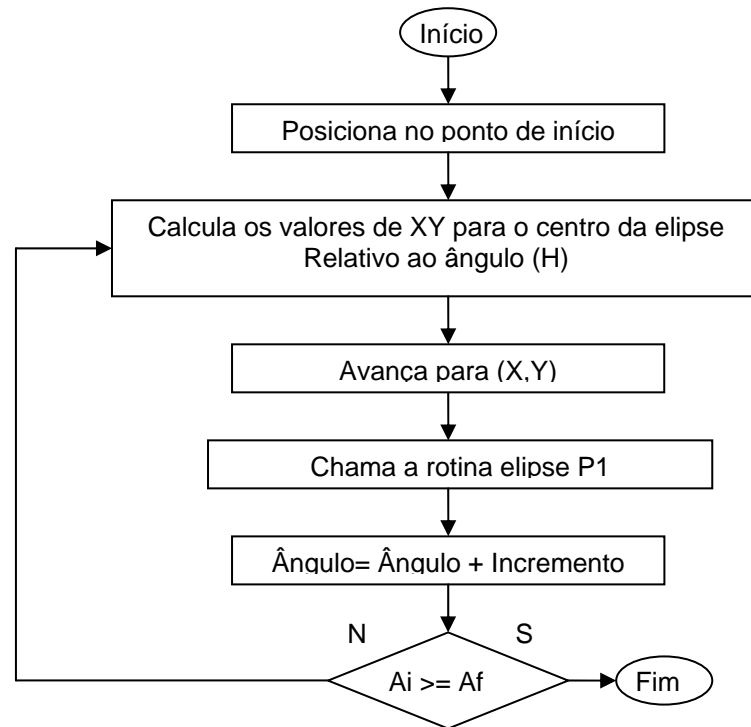


Figura 31 – Fluxograma cavidade elípticas múltiplas

3.5 Cavidade Cônica

Esta rotina é para usinagem de uma cavidade cônica Figura 32, utilizando o movimento espiral. A altura do cone também é uma variável lançada no programa.

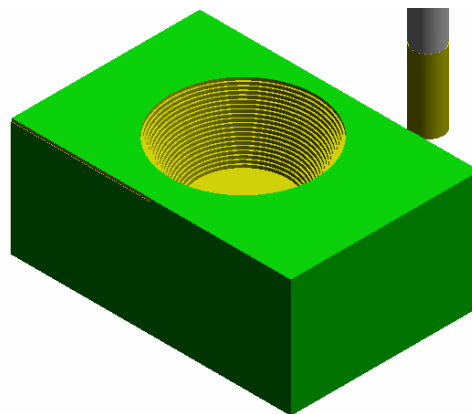


Figura 32 – Concavidade Cônica

A Figura 33 mostra o movimento em espiral realizado pela ferramenta. O ponto P1 é o início do movimento da ferramenta e R é o raio inicial, que receberá um decremento no decorrer do movimento. A Figura 34 mostra a profundidade do cone.

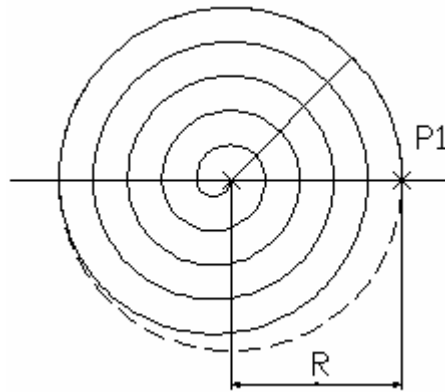


Figura 33 – Movimento em espiral da ferramenta

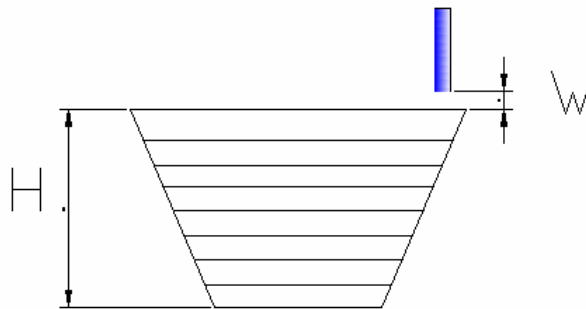


Figura 34 – Vista lateral da Cavidade Cônica

A Figura 35 mostra a trajetória da ferramenta. Na Tabela 20, as variáveis necessárias para a rotina da cavidade cônica.

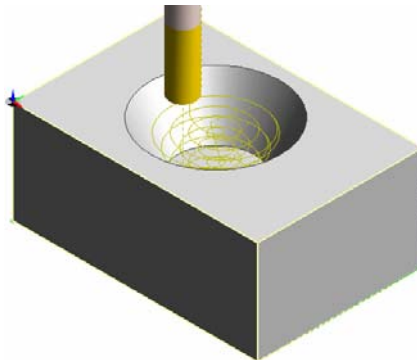


Figura 35 – Trajetória da ferramenta para a Cavidade Cônica

Tabela 20 - Variáveis para o programa cavidade cônica

Variáveis	Definições
A	Ângulo inicial
I	Incremento angular
Q	Decremento para o raio
T	Fator número de pontos
S	Número pontos
B	Coordenada X
C	Coordenada Y
H	Profundidade cone
W	Profundidade corte

Algoritmo

Para a usinagem da cavidade cônica, o movimento espiral foi escolhido para este caso. No início a ferramenta é posicionada no ponto de partida P1, de onde avança valor negativo em (Z), correspondente a variável (W). Temos que obter o número de pontos a serem calculados para desenhar o espiral, representado pela variável (S), que é determinado conforme a fórmula 8. O decremento para o raio é representado por (Q) e corresponde a fórmula 9, onde se reduz 10% do raio e o divide por (S). Em cada cálculo o ângulo é incrementado de (I) e o raio tem decremento de (Q). Os pontos são calculados pelas fórmulas usadas nos estudos anteriores, para (X) o raio é multiplicado pelo COS (A) e para o (Y) o raio é multiplicado pelo SEN (A) (fórmulas 6 e 7). Depois que a ferramenta fizer esta trajetória uma condição tem que ser satisfeita, se o raio não for menor ou igual a zero o programa retorna para a linha de incremento do ângulo e decremento do raio e calcula novos pontos para a ferramenta. Quando esta condição for aceita como verdadeira, o programa abastece o contador (J) com a soma da variável (J) com (W), e outra condição tem que ser verificada. Se a profundidade da cavidade (H) subtraída do contador (J) for menor ou igual à zero, o programa termina, caso contrario o raio é decrescido de 10% e a ferramenta movimentada para a nova posição do raio, e o ciclo começa novamente.

$$S = \frac{360}{LI} \quad (8)$$

$$Q = LR - \frac{(LR * .9)}{LS} \quad (9)$$

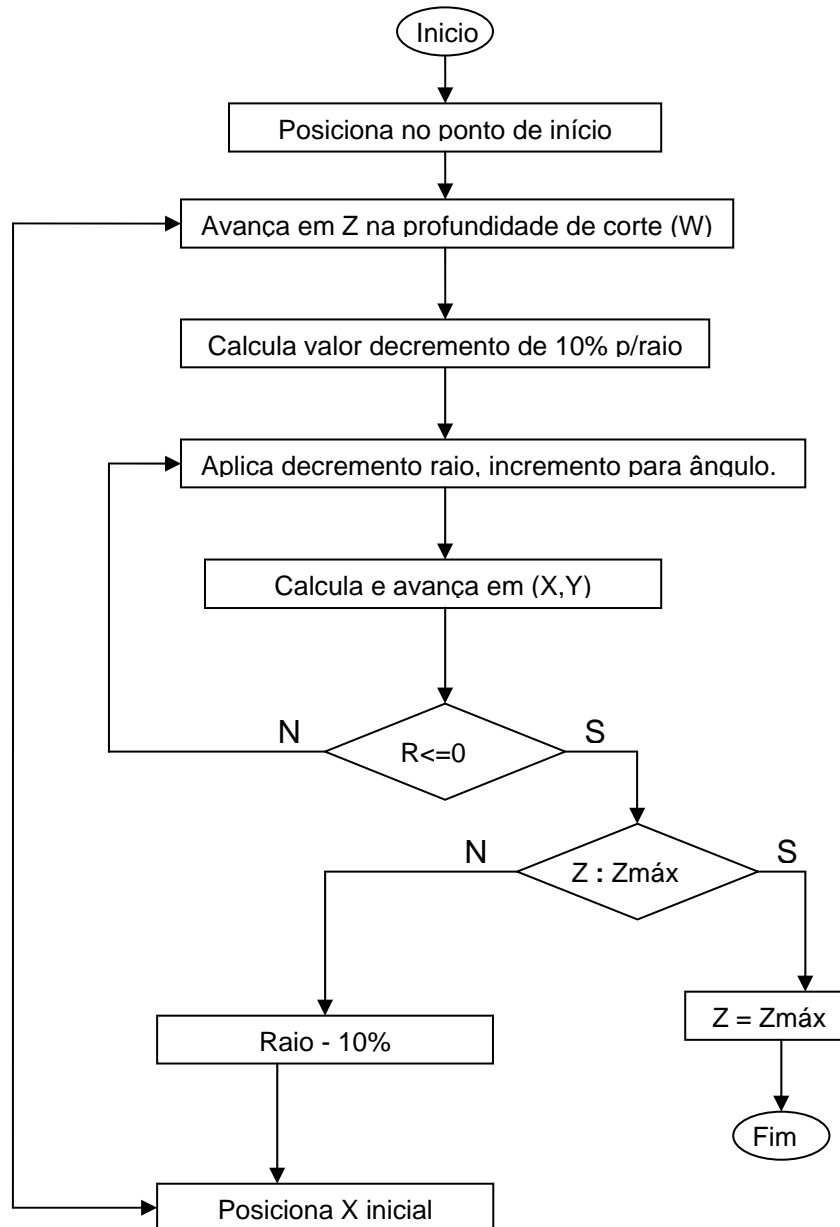


Figura 36 – Fluxograma para cavidade cônica

3.6 Cavidade Semi-Cilíndrica

Este ciclo de usinagem tem por objetivo formar um rasgo circular. Diferentes técnicas podem ser usadas. Neste caso, é utilizado o ciclo já estudado da cavidade retangular, uma sub-rotina já estudada, que é a base para realizar o perfil. Esse ciclo é realizado em camadas, com redução de área em cada passada na direção de (Z). Como é representado nas Figuras 37 e 38.

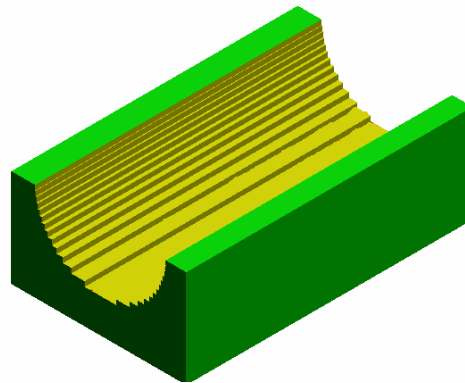


Figura 37 – Rasgo semi-circular

A cada incremento da profundidade recalcula-se o valor do comprimento, e aplica-se o movimento do ciclo cavidade retangular, proposto anteriormente. Este processo é utilizado até o limite da altura da semi-circunferência. Nas Figuras 38, 39 e 40, vemos geometrias para o desenvolvimento matemático.

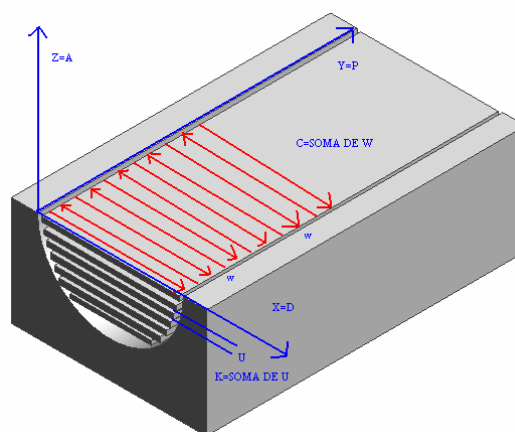


Figura 38 – Esquema para o desenvolvimento do perfil semi-circular

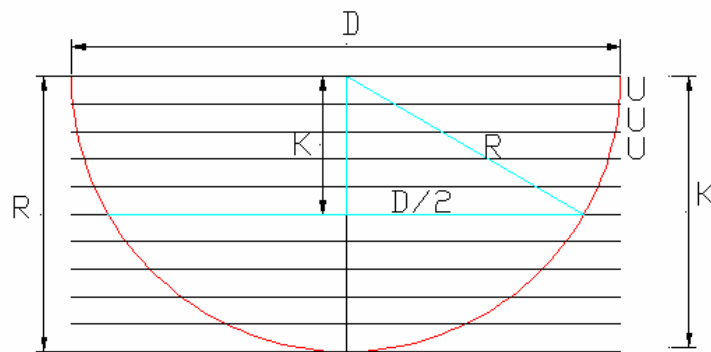


Figura 39 – Vista frontal do perfil semi-circunferência

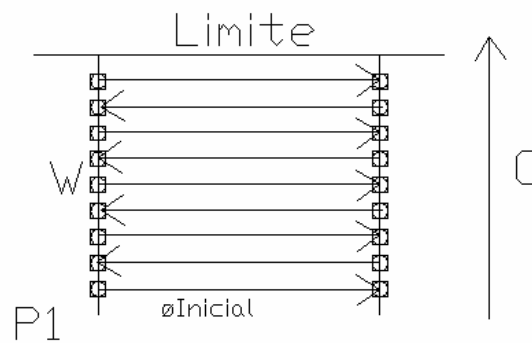


Figura 40 – Vista superior do movimento da ferramenta

O triângulo representado na Figura 41, é a relação geométrica utilizada para determinar as variáveis do algoritmo a ser implementado..

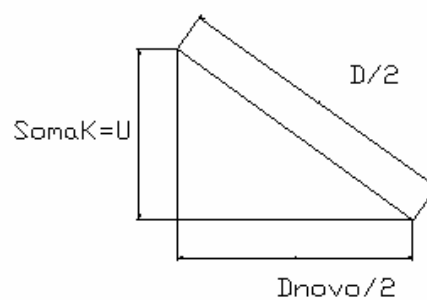


Figura 41 – Triângulo retirado da Figura 39

A fórmula 10, representa o valor recalculado para a metade do comprimento de cada nova área retangular. Este novo diâmetro será subtraído do original para se obter o novo ponto em (X), como mostra a fórmula 11.

$$\left(\frac{D}{2}\right)^2 = \left(\frac{Dn}{2}\right)^2 + K^2$$

$$D_{novo} = \sqrt{\left(\left(\frac{D^2}{4}\right) - C^2\right) * 4} \quad (10)$$

$$X_{novo} = (((D_{novo} - R) / 2) * B) \quad (11)$$

A Figura 42 mostra o traçado da ferramenta. A Tabela 21 relaciona as variáveis necessárias para programar a rotina de usinagem da cavidade semi-circular.

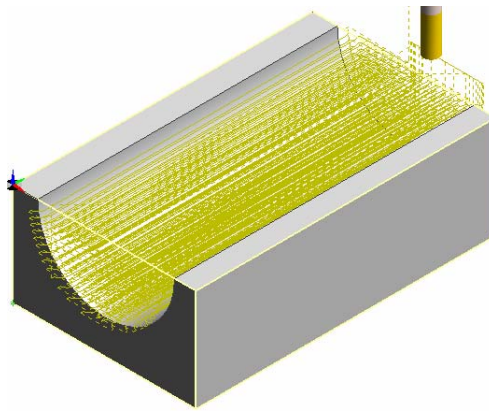


Figura 42 – Trajetória tradicional da ferramenta para usinar a peça

Tabela 21 - Variáveis para o programa rasgo cilíndrico

Variáveis	Definição
D	Comprimento
L	Largura
P	Profundidade
W	Largura de Corte
U	Profundidade de Corte
C	Contador(Y)
B	Inversor
R	D/2
K	Contador(Z)

Algoritmo

Para usar esse ciclo de usinagem, o programador necessita posicionar a ferramenta no ponto P1. Iniciando o ciclo a ferramenta avança o valor de (X) que é o comprimento representado pelo diâmetro (D) representado no fluxograma da Figura 43. No valor de (X) é aplicado o inversor. Se o valor de (Y) não for o máximo, avança (W) que é a largura de corte. Quando alcançado o valor de (Y) máximo, ocorre um último movimento em (Y) máximo. Se o (Z) máximo não foi atingido, retorna ao plano X,Y inicial e calcula-se o novo valor para (D) pela fórmula 10, e o novo valor para o ponto inicial (X) pela fórmula 11. Avança para o novo valor de (X) e redireciona para o início do módulo. Caso o (Z) máximo for alcançado é o fim do programa.

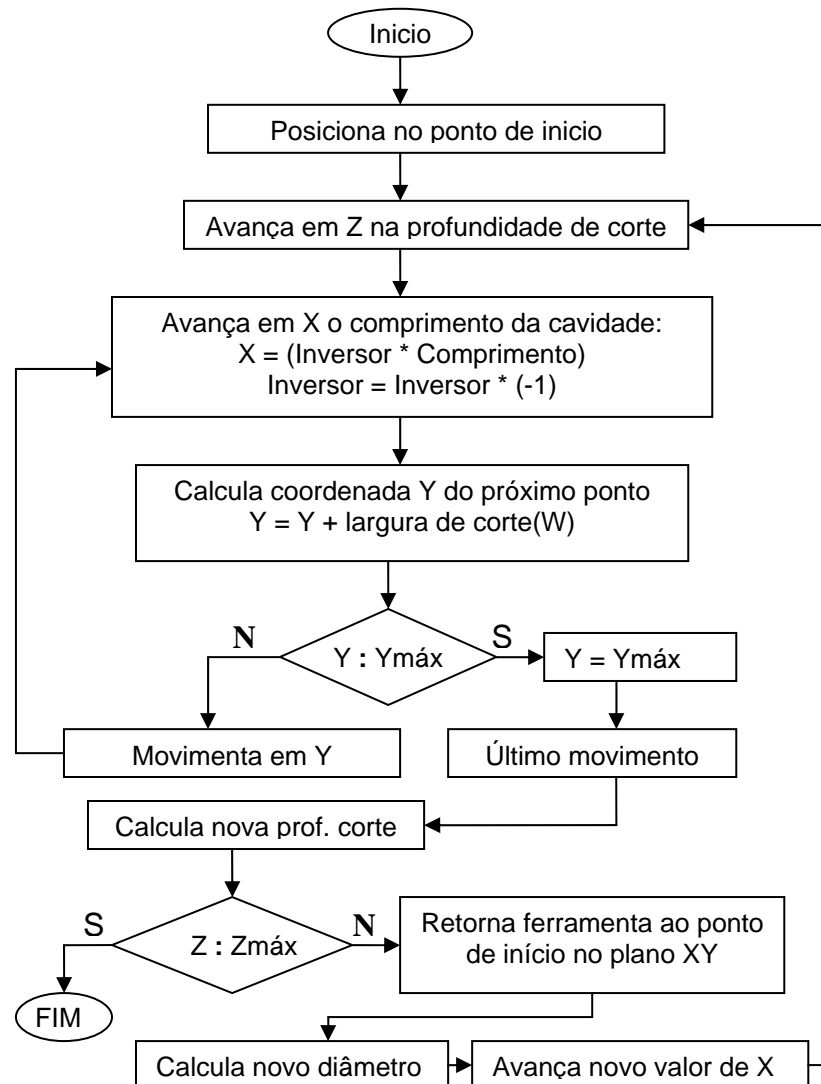


Figura 43 – Fluxograma cavidade semi-cilíndrica

4 – RESULTADOS E DISCUSSÃO

As rotinas estudadas foram carregadas na máquina por um editor de programação CNC, cada ciclo de usinagem foi registrado com um número de registro de memória do comando mostrado na Figura 44. O trabalho inicial foi no comando CNC da máquina para edição, entrada de valores nas variáveis e corrigir erros executando o programa. Para o uso dos ciclos é necessário a chamada de um programa principal, utiliza-se a função G39 e o número de armazenamento na memória dos ciclos. Foi gerada a simulação da trajetória da ferramenta no modo gráfico do comando CNC, e a seguir foi usinado um bloco de madeira. As rotinas trabalhadas se propõem ao estudo da programação parametrizada do CNC para o desenvolvimento de ciclos de usinagem. Foram programadas as equações para a ferramenta realizar geometrias esperadas.



Figura 44 Comando da fresadora Romi

Além da análise das rotinas, também é apresentado nesse capítulo, a título de comparação com o método proposto, a aplicação de sistema CAD/CAM para gerar automaticamente a programação CNC.

O sistema adotado foi o CAD/CAM mostrado na Figura 45, que usa uma característica de modelagem denominada "paramétrica", baseada em modelar e montar. No ambiente do CAD a criação de um sólido ou superfície tipicamente começa com a definição de topologia em um esboço 2D ou 3D. A topologia define a conectividade e certos relacionamentos geométricos entre vértices e curvas, ambos no esboço e externos ao esboço.

As geometrias estudadas foram analisadas no software de geração de sólidos CAD, onde foi possível visualizar a peça e prepará-la para um processo CAD/CAM.

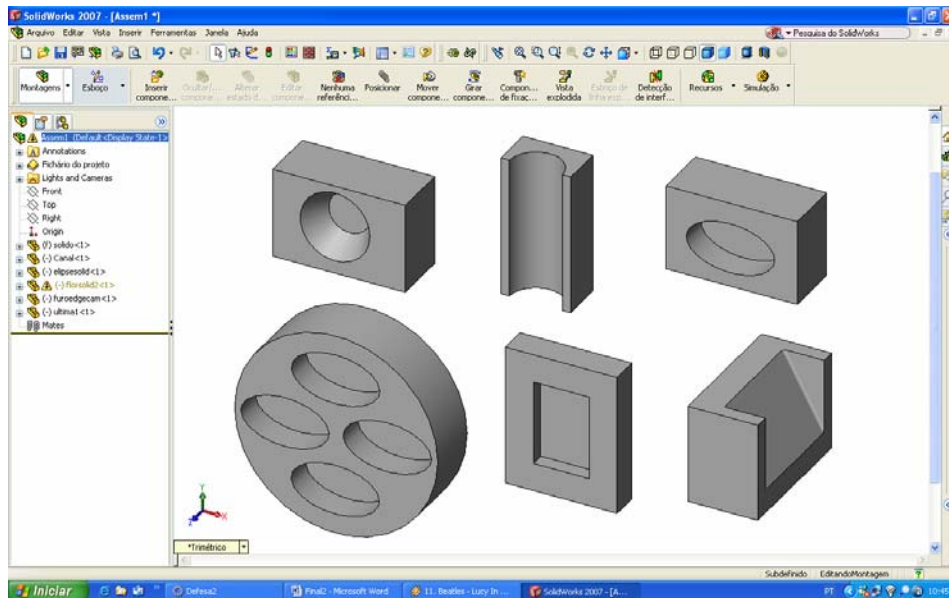


Figura 45 - - Software CAD utilizado para analisar os projetos

Para geração dos programas automaticamente foi utilizado o CAD/CAM mostrado na Figura 46, é um CAM para usinagem de produção e moldes e matrizes. Os programas foram gerados pelo processo CAD/CAM, com finalidade de comparar com a programação parametrizada.

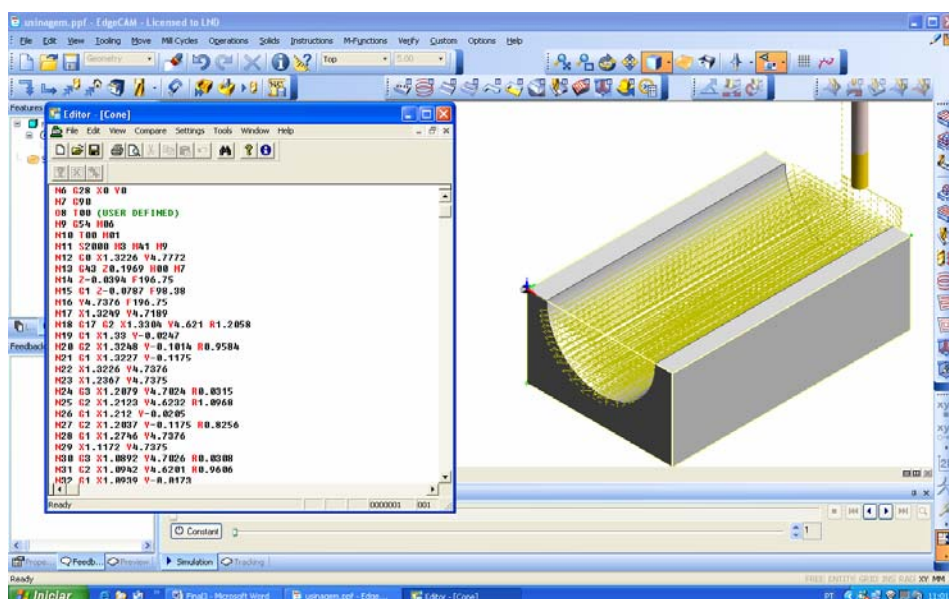


Figura 46 – Software CAD/CAM, utilizado para gerar o código CNC

4.1 Cavidade Retangular

O programa proposto em linguagem parametrizada do CNC para realizar a cavidade retangular esta descrito na Tabela 22.

Tabela 22 - Programa parametrizado CNC para rebaixo retangular

Códigos	Descrição
G39 C0 B1 K0	Contador/Inversor
G91	Posicionamento incremental
N4 G01 Z -(LU)	Avanço de corte
N1 G01 X(LX*LB)	Percorre o valor de (X)
G39 B(LB*-1)	Inverte o valor (B)
M (LE (LY-LC)) H2	Desvio condicional $y \leq c$
G01 Y(LW)	Largura de corte
G39 C(LC+LW)	Contador
H1	Volta em N1
N2 G00 Z(LU/4)	Recuo
G00 Y -(LC-LW)	Valor do recuo
G39 K (LK+LU)	Contador
M (LE(LZ-LK)) H3	Desvio condicional $z \leq k$
G39 C0	Zera contador
H4	Volta em N4
N3 G00 Z0	Volta Z0
M2	Final

Conforme a descrição do algoritmo dessa função a trajetória da ferramenta foi verificada no comando em simulação gráfica. A Figuras 47 mostra esta simulação no plano XY. A ferramenta percorre o valor de (X), partindo do canto inferior esquerdo avança em (Y), retorna em (X) negativo, e segue no movimento previsto até a condição de (Y) máximo ser verdadeira.



Figura 47 – Traçado para cavidade retangular (X,Y)

Na Figura 48 é verificado o avanço em (Z) no plano YZ, previsto no programa, limitado quando for verdadeira a condição de (Z) máximo.



Figura 48 – Traçado da cavidade retangular (Y,Z)

Esta rotina foi elaborada para o estudo do movimento da ferramenta, não possui ciclos de acabamento e compensação da ferramenta. A Figura 49 mostra a usinagem Cavidade Retangular. A Figura 50 mostra o programa gerado pelo CAD/CAM.

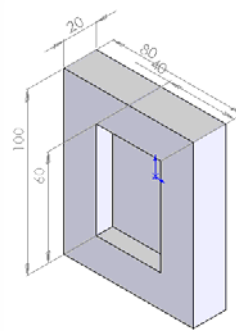


Figura 49 – Seqüência de usinagem da cavidade retangular

```

%
N1 G20 G90 G40
N2 G10 P1 Z0.0 R0.2756 T00
N3 G10 P2 Z0.0 R0.2756 T00
N4 G10 L2 P1 X0.0 Y0.0 Z0.0 (Top)
N5 G28 G91 Z0
N6 G28 X0 Y0
N7 G90
O8 T00 (USER DEFINED)
N9 G54 M06
N10 T00 M01
N11 S2000 M3 M41 M7
N12 (ROUGHING OPERATION)
O13 T00 (USER DEFINED)
N14 G54 M06
N15 T00 M01
N16 S2000 M3 M41 M9
N17 G0 X0.0004 Y-0.5461
N18 G43 Z0.9843 H00 M7
N19 G1 X0.0732 Y-0.5922 Z0.9719 F98.82
N20 X0.1567 Y-0.6135 Z0.9595
N21 X0.2426 Y-0.6081 Z0.9472
N22 X0.3227 Y-0.5763 Z0.9348
N23 X0.3891 Y-0.5213 Z0.9225
-
-
N170 G00 Z20.0 M09
N171 Z0 H00 M19
N172 M30
%

```



```

%
N1 G99
N2 G71
N3 T1 M6
N4 O1
N5 G00 X0.Y0.Z0.
N6G39 X40.Y60.Z20.W2.U2 P1
N7 M2
%

```

Figura 50 – Programas da cavidade retangular para comparação

A Figura 50 mostra uma listagem à esquerda de um programa gerado por CAD/CAM do modelo de cavidade retangular que está cotada a direita, e um trecho de uma chamada para o ciclo de usinagem da cavidade retangular. Este programa carrega as variáveis e chama o programa na memória por P1. É verificada uma diferença na quantidade de linhas de programação. No caso de mudarmos as dimensões da peça, no processo CAD/CAM é necessário refazer o desenho, transferi-lo para a máquina, refazer as configurações de ferramenta, por exemplo, e gerar o programa novamente. No processo com a linguagem parametrizada basta apenas alimentar com as dimensões atualizadas as variáveis. As próximas rotinas apresentada terão o mesmo sistema de comparação.

4.2 Rampa

O programa parametrizado do CNC para realizar o ciclo de rampa está descrito na Tabela 23.

Tabela 23 - Programação paramétrica CNC para perfil de rampa

Códigos	Descrição
G39 J0 T(LZ/LY) I0 B1	Variáveis local
G39 K(LU)	Variáveis local
N30 G39 L(LK/LT)	Calcula L
G01 Z-(LU)	Posicionamento Z
N10 G01 X(LX*LB)	Inversor
G39 B(LB*(-1))	Troca inversor
M (LE(LY-LJ-LL) H20	Limite em Y
G01 Y(LW)	Profundidade Y
G39 J(LJ+LW)	Contador Y
H10	Retorna
N20 G00 Z(LU/4)	Movimento em Z
G00 Y-(LJ-LW)	Retorna Y inicial
G39 K(LK+U)	Contador Z
M (LE(LZ-LK)) H40	Limite Z
G39 J0	Zera contador
H30	Redireciona
N40 M2	Final

A ferramenta percorre o valor de (X) Figura 51, mostrando o deslocamento à direita, avança o valor de (Y), retorna em (X) negativo, e segue no movimento previsto até a condição de (Y) máximo ser verdadeira.



Figura 51 – Traçado da rampa (X,Y)

Na Figura 52 é verificado a inclinação da rampa, previsto no programa, e a profundidade da rampa, limitado quando for verdadeira a condição de (Z) máximo.



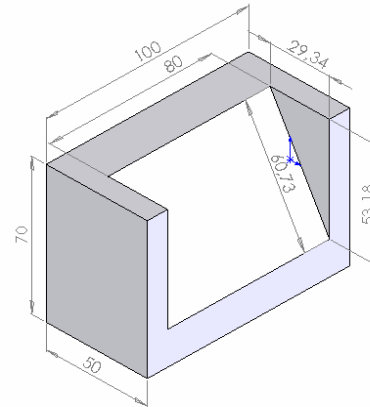
Figura 52 – Traçado da rampa (Y,Z)

Esta rotina foi elaborada para o estudo do movimento da ferramenta. A Figura 53 programa gerado por CAD/CAM.

```

%
N1 G20 G90 G40
N2 G10 P1 Z0.0 R0.2756 T00
N3 G10 L2 P1 X0.0 Y0.0 Z0.0 (Top)
N4 (ROUGHING OPERATION)
N5 G28 G91 Z0
N6 G28 X0 Y0
N7 G90
O8 T00 (USER DEFINED)
N9 G54 M06
N10 T00 M01
N11 S2000 M3 M41 M9
N12 G0 X2.2831 Y2.4677
N13 G43 Z0.1969 H00 M7
N14 Z-0.0394 F98.82
N15 G1 Z-0.0787 F49.41
N16 X2.2609 Y2.4676 F98.82
N17 G17 G3 X2.2037 Y2.4577 R0.8411
N18 G2 X2.075 Y2.4381 R1.3965
N19 G1 X2.0423 Y2.4349
N20 G3 X1.9687 Y2.3502 R0.0856
N21 G1 X1.9688 Y1.5829
N22 G3 X2.0513 Y1.5011 R0.086
N23 G2 X2.2079 Y1.478 R1.1223
-
-
N1479 G00 Z20.0 M09
N1480 Z0 H00 M19
N1481 M30
%

```



```

%
N1 G99
N2 G71
N3 T1 M6
N4 O1
N5 G00 X0.Y0.Z0.
N6 G39 X80.Y30.W2.U5 P1
N7 M2
%

```

Figura 53 – Programa gerado pelo CAD/CAM para rampa

4.3 Cavidade Elíptica

O programa proposto em linguagem parametrizada do CNC para realizar a cavidade elíptica está descrito na Tabela 24.

Tabela 24 - Programação paramétrica CNC para elipse

Códigos	Definições
N50 G39 R(LM) C(LN)	Variáveis R e C
N40 G01 X(LR)	Posicionamento X
G01 Z-(LS)	Profundidade Z
N10 G01 X(LR*(COS(LA))) Y(LC*(SEN(LA)))	Determina X e Y
G39 A(LA*LI)	Incremento ângulo
M (LE(LB-LA)) H20	Se diferença for menor e igual zero
H10	Redireciona
N20 G01 X(LR*(COS(LB))) Y(LC*(SEM(LB)))	Determina ultimo ponto
G39 R(LR-LD)	Decremento R
G39 C(LC-LD)	Decremento C
G39 A0	Zera o ângulo
M (LE(LR)) H30	Se menor e igual a zero
H40	Redireciona
N30 G39 J(LJ+LS)	Contador
M (LE(LZ-LJ)) H40	Se diferença for menor e igual a zero
H50	Redireciona
N40 Z0 M5	
M2	

A ferramenta percorre o valor de (X,Y) calculados ponto a ponto sobre o raio maior Figura 54, mostrando a trajetória elíptica da ferramenta, e segue no movimento previsto pelo programa. A Figura 55 mostra o traçado com redução do raio, como previsto.



Figura 54 – Traçado da elipse de raio maior



Figura 55 – Traçado elíptico com redução do raio

Esta rotina foi elaborada para o estudo do movimento da ferramenta, na usinagem o comportamento foi de acordo com o programado. A Figura 56 mostra a usinagem da elipse.

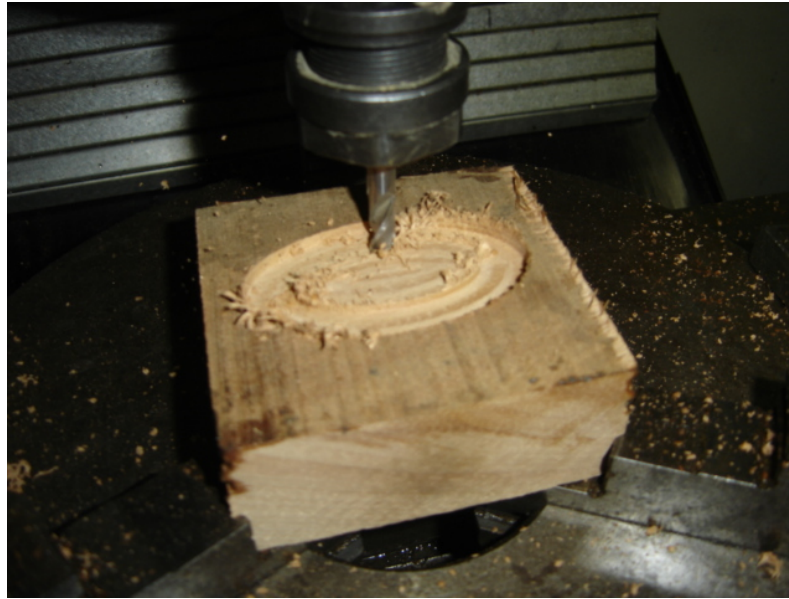


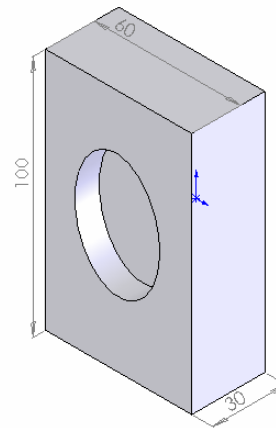
Figura 56 – Sequência de usinagem da elipse

A Figura 57 mostra o programa gerado pelo CAD/CAM para a elipse.

```

%
N1 G20 G90 G40
N2 G10 P1 Z0.0 R0.2756 T00
N3 G10 L2 P1 X0.0 Y0.0 Z0.0 (Top)
N4 (ROUGHING OPERATION)
N5 G28 G91 Z0
N6 G28 X0 Y0
N7 G90
O8 T00 (USER DEFINED)
N9 G54 M06
N10 T00 M01
N11 S2000 M3 M41 M9
N12 G0 X1.3763 Y2.0521
N13 G43 Z0.1969 H00 M7
N14 G1 X1.4443 Y2.105 Z0.1845 F98.82
N15 X1.4927 Y2.1763 Z0.1721
N16 X1.5166 Y2.2591 Z0.1598
N17 X1.5138 Y2.3452 Z0.1474
N18 X1.4845 Y2.4262 Z0.1351
N19 X1.4316 Y2.4942 Z0.1227
N20 X1.3602 Y2.5426 Z0.1103
N21 X1.2774 Y2.5665 Z0.098
N22 X1.1913 Y2.5637 Z0.0856
N23 X1.1103 Y2.5344 Z0.0733
N24 X1.0423 Y2.4814 Z0.0609
-
-
N643 G00 Z20.0 M09
N644 Z0 H00 M19
N645 M30
%

```



```

%
N1 G99
N2 G71
N3 T1 M6
N4 O1
N5 G00 X0.Y0.Z0.
N6 G39 R80. C30. D5. S5. A0. B360. P1
N7 M2
%

```

Figura 57 – Programa gerado pelo CAD/CAM para a cavidade elíptica

Foi gerada uma simulação em planilha eletrônica para verificação das fórmulas utilizadas para o programa da elipse, mostrada na Figura 58. Esta planilha eletrônica foi elaborada com as fórmulas (4) e (5) citadas na metodologia, que tem a finalidade de calcular as coordenadas dos pontos para ângulos de 0° a 360° com incremento de 10, e os resultados das coordenadas são plotadas em um gráfico de dispersão.

A planilha permite alterações dos valores de raio maior e raio menor da elipse, e o incremento usado no ângulo.

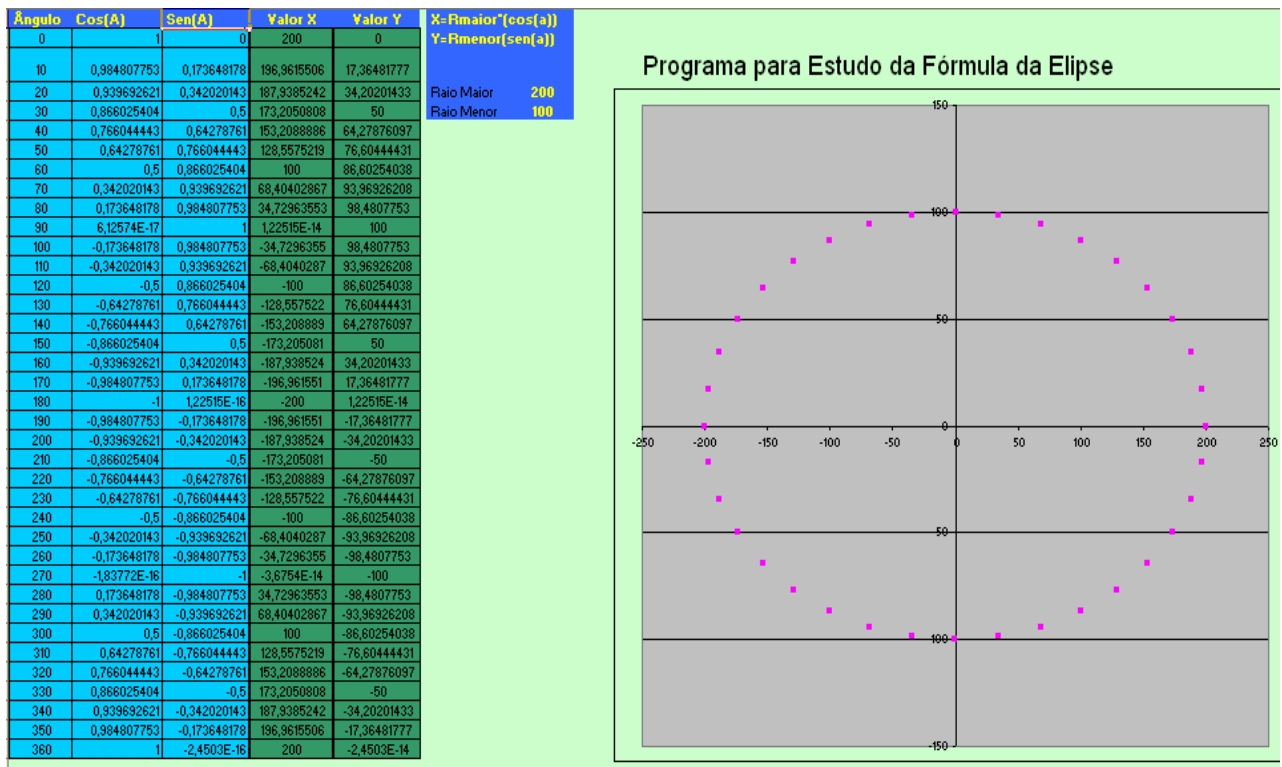


Figura 58 – Programa com Raio maior 200 e Raio menor 100

4.4 Cavidades elípticas múltiplas

O programa proposto em linguagem parametrizada do CNC para realizar as cavidades elípticas múltiplas está descrito na Tabela 25.

Tabela 25 - Programação paramétrica CNC para cópia múltiplas de elipse

Códigos	Definições
N70 G00 X(LW*(COS(LH))) Y(LW*(SEN(LH)))	Posicionamento
G39 P1	Chama programa elipse P1
N40 H(H+V)	Incrementa ângulo
M (LE(LY-LH) H60	Se diferença dos ângulos menor igual
H70	Redireciona
N60 Z0 M5	Fim
M1	

A Figura 59 mostra o comando simulando o programa com um incremento de 90° gerando quatro cópias de elipses distribuídas igualmente. Os valores calculados ponto a ponto obedecem ao previsto na programação.



Figura 59 – Traçado cópias múltiplas de elipse

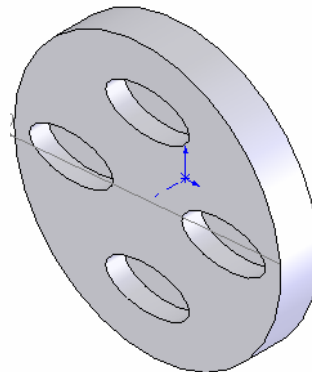
A Figura 60 mostra a usinagem da cavidade elíptica de cópias múltiplas. A Figura 61 mostra o programa gerado pelo CAD/CAM.



Figura 60 – Sequência de usinagem da cavidade elípticas múltiplas

```

%
N1 G20 G90 G40
N2 G10 P1 Z0.0 R0.2362 T00
N3 G10 L2 P1 X0.0 Y0.0 Z0.0 (Top)
N4 (ROUGHING OPERATION)
N5 G28 G91 Z0
N6 G28 X0 Y0
N7 G90
N8 T00 (USER DEFINED)
N9 G54 M06
N10 T00 M01
N11 S2000 M3 M41 M9
N12 G0 X0.1894 Y1.5676
N13 G43 Z0.9843 H00 M7
N14 G1 X0.6144 Z0.9134 F15.75
N15 X-0.6141 Z0.7087
N16 X-0.6075 Y1.5597
N17 X-0.5958 Y1.548
N18 X-0.5879 Y1.5407
N19 G17 G3 X-0.5487 Y1.5126 R0.3904
N20 X-0.5095 Y1.4902 R0.3825
N21 X-0.4653 Y1.4696 R0.4867
N22 G1 X-0.4507 Y1.4633
N23 G3 X-0.3135 Y1.4207 R1.0286
N24 X-0.2351 Y1.4051 R1.1367
-
-
N991 G00 Z20.0 M09
N992 Z0 H00 M19
N993 M30
%
```



```

%
N1 G99
N2 G71
N3 T1 M6
N4 O1
N5 G00 X0.Y0.Z0.
N6 G39 R80. C30. D5. S5. A0. B360. H0. V90. P1
N7 M2
0%
```

Figura 61 – Programa gerado pelo CAD/CAM para a cavidades elípticas múltiplas

4.5 Cavidade Cônica

O programa proposto em linguagem parametrizada do CNC para realizar a cavidade cônica está descrito na Tabela 26.

Tabela 26 - Programação paramétrica CNC para cavidade cônica

Códigos	Definições
G39 R_A_I_T_H_W_T360	Variáveis
N3 G01 Z - (LW)	Profundidade de corte
G39 S(LT/LI)	Número pontos
G39 (Q((LR-(LR*.9))/LS)	Calculo decremento do raio 10%
N1 G39 A(LA+LI) R(LR-LQ)	Redefinindo A e R
G39 B(LR*COS(LA) C(LC*SEN(LA))	Calculando X e Y
G01 X(LB) Y(LC)	Posicionamento
M (LE(LR)) H2	Se raio for menor ou igual a zero
H1	Redirecionamento
N2 G39 J(LJ+LW)	Contador
M (LE(LH-LJ)) H4	Se a diferença for menor ou igual a zero
G39 R(LR*.9)	Menos 10% do raio
G01 X(LR) Y(0)	Posicionamento novo
H3	Redireciona
N4 M2 M5	

A Figura 62 mostra o comando simulando o movimento em espiral no plano XY previsto no programa. A Figura 63 mostra no plano (Y,Z) a profundidade do cone gerado graficamente.



Figura 62 – Movimento espiral gerado pelo comando (X,Y)



Figura 63 – Comando gerando profundidade cônica (Y,Z)

A Figura 64 mostra a usinagem da cavidade cônica.

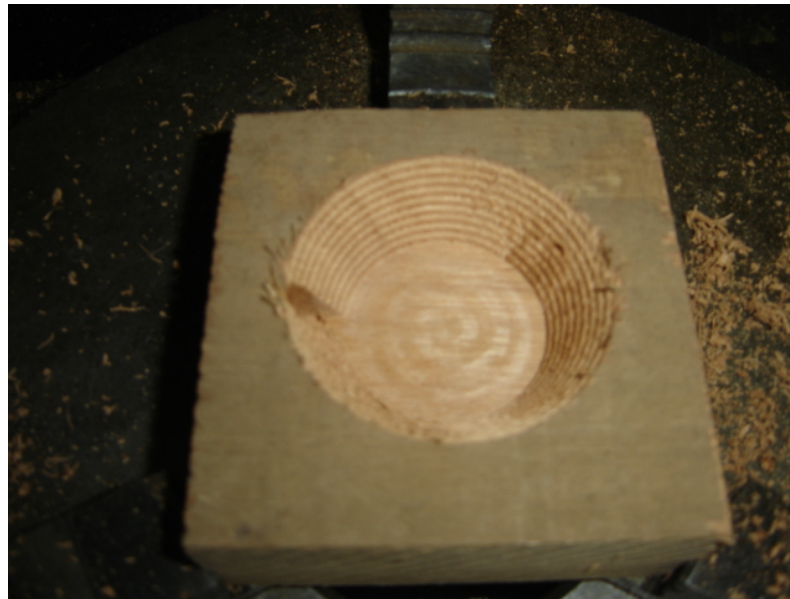


Figura 64 – Usinagem da cavidade cônica

Foi gerada uma simulação em planilha eletrônica para validar das fórmulas utilizadas para o programa da cavidade cônica. Esta planilha eletrônica foi elaborada com as fórmulas (8) e (9) citadas na metodologia, os resultados das coordenadas são plotadas em um gráfico de dispersão, como mostrada na Figura 65.

A planilha permite alterar os valores de raio da espiral, e o decremento do raio (Q). Figura 66 mostra programa gerado pelo CAD/CAM para a cavidade cônica.

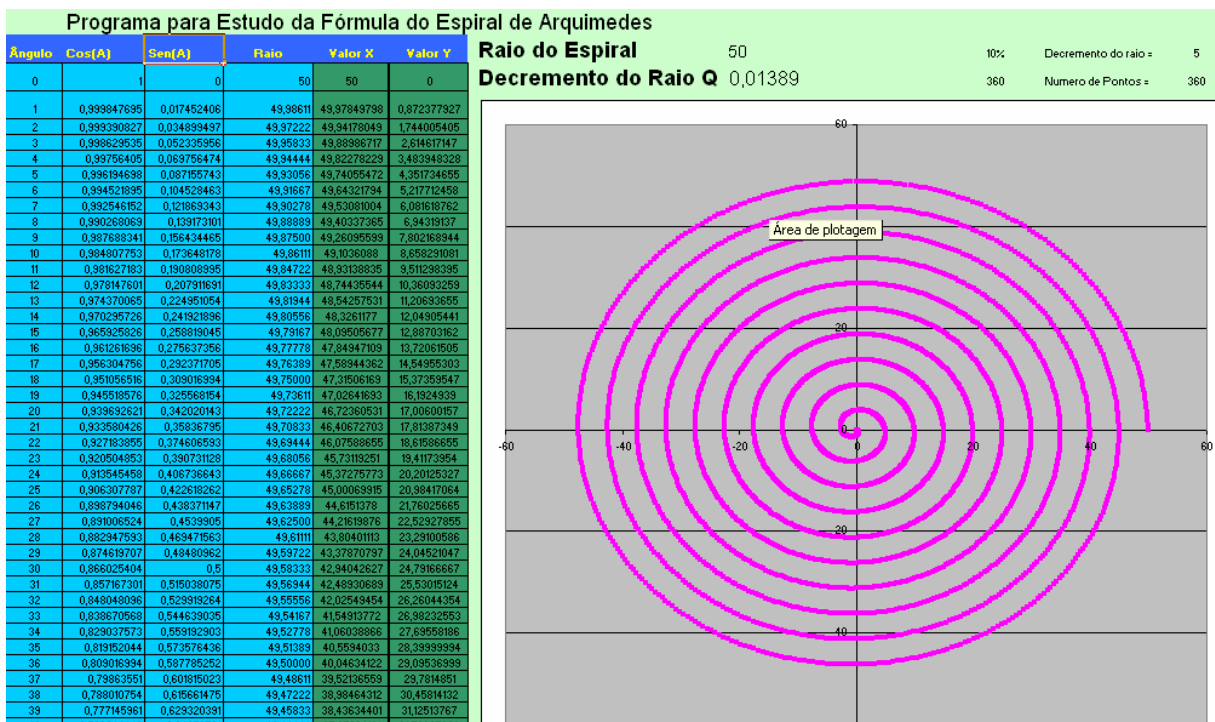
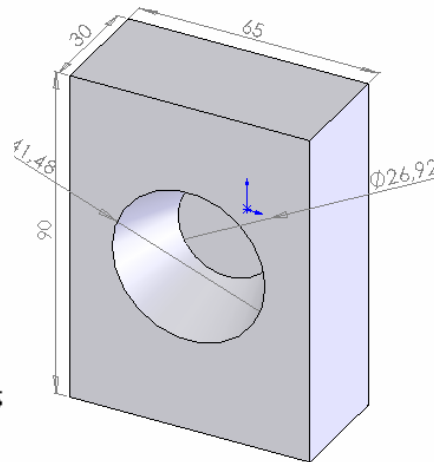


Figura 65 – Planilha de simulação para gerar a espiral

```

%
N1 G20 G90 G40
N2 G10 P1 Z0.0 R0.2362 T00
N3 G10 L2 P1 X0.0 Y0.0 Z0.0 (Top)
N4 (ROUGHING OPERATION)
N5 G28 G91 Z0
N6 G28 X0 Y0
N7 G90
O8 T00 (USER DEFINED)
N9 G54 M06
N10 T00 M01
N11 S2000 M3 M41 M9
N12 G0 X1.8242 Y1.1237
N13 G43 Z0.1969 H00 M7
N14 G1 X1.89 Y1.1573 Z0.1845 F15.75
N15 X1.9421 Y1.2095 Z0.1721
N16 X1.9756 Y1.2754 Z0.1598
N17 X1.9871 Y1.3483 Z0.1474
N18 X1.9754 Y1.4212 Z0.1351
N19 X1.9419 Y1.487 Z0.1227
N20 X1.8896 Y1.5392 Z0.1103
N21 X1.8238 Y1.5726 Z0.098
N22 X1.7508 Y1.5841 Z0.0856
N23 X1.6779 Y1.5725 Z0.0733

```



-

-

```

N431 G00 Z20.0 M09
N432 Z0 H00 M19
N433 M30
%
```

```

%
N1 G99
N2 G71
N3 T1 M6
N4 O1
N5 G00 X0.Y0.Z0.
N6 G39 R50. A0. H30. B360. P1
N7 M2
o%
```

Figura 66 – Programa gerado pelo CAD/CAM para a cavidade cônica

4.6 Cavidade Semi-Cilíndrica

O programa proposto em linguagem parametrizada do CNC para realizar o ciclo cavidade cônica está descrito na Tabela 25.

Tabela 27 - Programação paramétrica CNC para rasgo simi circular

Códigos	Descrições
G39 C0 B1 K0	Variáveis
N4 G01 Z-(LU)	Profundidade inicial
N1 G01 X(LX*LB)	Percorre o valor x
G39 B(LB-1)	Troca inversor
M (LE(LY-LC)) H2	Se a diferença for menor igual a zero
G01 Y(LW)	Avanço ferramenta
G39 C(LC+LW)	Contador y
H1	Redireciona
N2 G00 Z(LU/4)	Afasta da peça
G00 Y-(LC-LW)	Retorna y inicial
G39 K(LK+LU)	Contador z
M (LE(LZ-LK)) H3	Se diferença menor igual zero
G39 D(SQR(((D ² /4-C ²)*4)	Calcula novo d
G00 X(((LD-LR)/2)*B)	Posicionamento novo
G39 C0 X(LD)	Recebe variáveis
G00 Z(LK)	Avança z
H4	Redireciona
N3 G00 Z0 M2	Fim

A Figura 67 e 68 mostram o comando simulando o programa da cavidade cilíndrica conforme o previsto na programação. A Figura 69 mostra o programa gerado pelo CAD/CAM para a cavidade cilíndrica.



Figura 67 – Simulação gráfica da cavidade circular (X,Y)

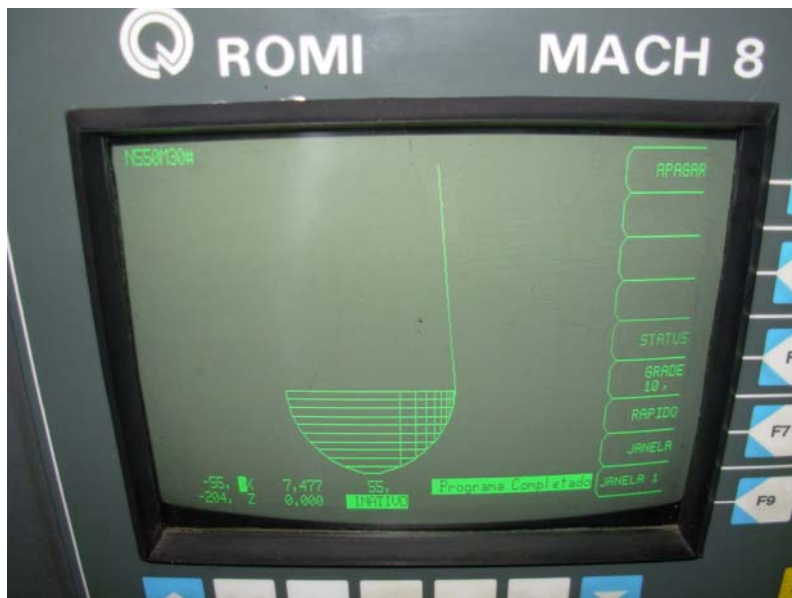
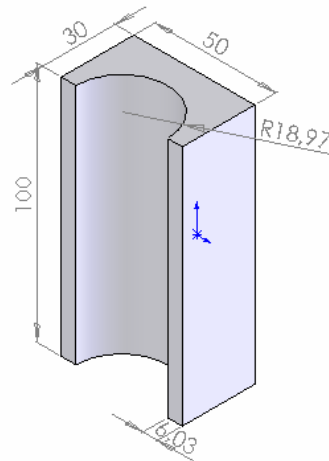


Figura 68 - Simulação gráfica da cavidade circular (X,Z)

```

%
N1 G20 G90 G40
N2 G10 P1 Z0.0 R0.1181 T00
N3 G10 L2 P1 X0.0 Y0.0 Z0.0 (Top)
N4 (ROUGHING OPERATION)
N5 G28 G91 Z0
N6 G28 X0 Y0
N7 G90
N8 T00 (USER DEFINED)
N9 G54 M06
N10 T00 M01
N11 S2000 M3 M41 M9
N12 G0 X1.3226 Y4.7772
N13 G43 Z0.1969 H00 M7
N14 Z-0.0394 F196.75
N15 G1 Z-0.0787 F98.38
N16 Y4.7376 F196.75
N17 X1.3249 Y4.7189
N18 G17 G2 X1.3304 Y4.621 R1.2058
N19 G1 X1.33 Y-0.0247
N20 G2 X1.3248 Y-0.1014 R0.9584
N21 G1 X1.3227 Y-0.1175
N22 X1.3226 Y4.7376
N23 X1.2367 Y4.7375
N24 G3 X1.2079 Y4.7024 R0.0315
N25 G2 X1.2123 Y4.6232 R1.0968

N1916 G00 Z20.0 M09
N1917 Z0 H00 M19
N1918 M30
%
```



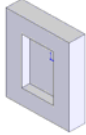





```

%
N1 G99
N2 G71
N3 T1 M6
N4 O1
N5 G00 X0.Y0.Z0.
N6 G39 R20. Y100. Z30. W5. U5. P1
N7 M2
o/
```

Figura 69 – Programa gerado pelo CAD/CAM para a cavidade cilíndrica

A Tabela 28 mostra o número de linhas geradas pelo sistema CAD/CAM e o sistema de programação parametrizado. Usando uma ferramenta de corte de 14 mm e profundidade de corte diferentes com 2mm e 0.5mm.

Tabela 28 – Comparação do número de linhas gerado pelos dois processos.

GEOMETRIAS	CAD/CAM	PARAMETRIZADO
	<p>Dimensões: X40.Y60.Z20. Para profundidade de corte U=2. Gerado 172 linhas</p> <p>Para profundidade de corte U=0.5 Gerado 607 linhas</p>	<p>Dimensões: Qualquer</p> <p>Programado com 13 linhas</p>
	<p>Dimensões: X80.Y30.Z53. Para profundidade de corte U=2. Gerado 1481 linhas</p> <p>Para profundidade de corte U=0.5 Gerado 6052 linhas</p>	<p>Dimensões: Qualquer</p> <p>Programado com 15 linhas</p>
	<p>Dimensões: R50.C25.Z10. Para profundidade de corte U=2. Gerado 645 linhas</p> <p>Para profundidade de corte U=0.5 Gerado 2462 linhas</p>	<p>Dimensões: Qualquer</p> <p>Programado com 17 linhas</p>
	<p>Dimensões: R200.C30.A0.B360. Para profundidade de corte U=2. Gerado 993 linhas</p> <p>Para profundidade de corte U=0.5 Gerado 4130 linhas</p>	<p>Dimensões: Qualquer</p> <p>Programado com 23 linhas</p>
	<p>Dimensões: R50.A0.B360.H30. Para profundidade de corte U=2. Gerado 433 linhas</p> <p>Para profundidade de corte U=0.5 Gerado 1778 linhas</p>	<p>Dimensões: Qualquer</p> <p>Programado com 13 linhas</p>
	<p>Dimensões: D40.Y100.Z30. Para profundidade de corte U=2. Gerado 1918 linhas</p> <p>Para profundidade de corte U=0.5 Gerado 7747 linhas</p>	<p>Dimensões: Qualquer</p> <p>Programado com 17 linhas</p>

4.7 Restrições dos ciclos

Embora não tenha sido previsto parâmetros para acabamento, condições reduzidas de largura de corte e profundidade de corte pode ser gerada no último passe de desbaste, quando definidos apropriadamente os parâmetros. A compensação da ferramenta pode ser feita dentro do ciclo junto com a entrada em rampa dependendo do tipo de ferramenta usada.

5 - CONCLUSÃO

A metodologia demonstrou que a programação parametrizada em equipamentos CNC é uma técnica útil para aumentar a flexibilidade e a potencialidade de sistemas produtivos. Em comparação com a geração de programas CNC através de sistemas CAD/CAM, o método proposto apresenta vantagens quanto à fabricação de geometrias semelhantes. Quando altera-se as dimensões de uma geometria, no sistema CAD/CAM é necessário reinicializar todo o processo, ou seja, redesenhar a peça no CAD, carregar o arquivo no CAD/CAM, definir parâmetros de usinagem e gerar novo programa. No sistema proposto, essas etapas são desnecessárias porque o usuário especifica a nova geometria do corte apenas pela redefinição dos parâmetros do ciclo.

Comparando-se o tamanho dos programas CNC gerados pelos dois processos através da observação do número de blocos. A significativa diferença no tamanho dos programas implica em menor necessidade de memória no comando para programas gerados pelos ciclos de usinagem parametrizados. Em comandos com limitação na capacidade de memória, essa característica significa maior capacidade de armazenamento do número de programas.

Outra vantagem que se pode observar é que os ciclos disponíveis em máquinas CNC são rígidos quanto à seqüência de usinagem. A técnica proposta permite gerar novos ciclos de usinagem para a mesma geometria, com liberdade de configuração. Certos ciclos disponíveis no comando, por exemplo, podem executar a usinagem apenas no modo concordante ou discordante. Para inverter a condição disponível, bastaria definir um novo ciclo com seqüência de usinagem invertida em relação ao original.

Embora as geometrias analisadas sejam de relativa simplicidade, a técnica de combinação dessas rotinas, como no ciclo de cavidades elípticas múltiplas, mostrou ser uma possibilidade de desenvolvimento de rotinas mais elaboradas. Porém, a limitação para programar geometrias mais complexas e a complexidade de programação, são desvantagens verificadas no sistema de programação parametrizada.

Algumas sugestões de trabalhos futuros relacionadas a esse sistema são:

- Aperfeiçoar os ciclos, com descida em rampa, acabamento e compensação da ferramenta.

- Criar uma estrutura de banco de dados de ciclos de usinagem para simplificar a disponibilidade do sistema para o usuário, através de uma biblioteca de modelos.
- Desenvolver um simulador para PC da linguagem parametrizada do CNC.
- Desenvolver um compilador para uma linguagem de programação parametrizada.

REFERÊNCIAS BIBLIOGRÁFICAS

ABIMAQ - Associação Brasileira da Indústria de Máquinas e Equipamentos. Disponível em <<http://www.abimaq.org.br/>>

AGUIAR, M. M.; **Um exemplo de programação parametrizada.** Disponível em: <<http://www.mundocnc.com.br/basico12.htm>>

ANDERL, R.; CLAASSEN, E. **Virtual product development based on product data technology.** In: Seminário de Alta Tecnologia, 3., 1998, Santa Bárbara Doeste. Anais.. Santa Bárbara Doeste, UNIMEP, 1998. p.19-34.

BALDUINO FILHO, Jovelino. **Desenvolvimento de um Protótipo de uma Mesa X-Y para Corte e Solda, Aplicando Técnicas da Qualidade Total.** Dissertação de Mestrado da Universidade Federal de Uberlândia, Curso de Engenharia Mecânica, 2001.

BOLWIJN, P. T.; KUMPE, T. **Manufacturing in the 1990's: productivity, flexibility and innovation.** Long Range Planning. V.23, n.4, 1990.

CASSANIGA, F. A. **Fácil Programação do Controle Numérico.** São Paulo, 2000.

CIMM – Centro de Informação Metal Mecânica. Disponível em <<http://www.cimm.com.br/>>

COSTA, E. S.; Santos, D.J. **Processos de usinagem.** Apostila do curso de Eletromecânica do CEFET-MG unidade de Divinópolis, 2006.

DEMEC/EE/UFMG. Disponível em: <<http://www.demec.ufmg.br/>>

FERREIRA, Erwin Rommel. **Planejamento de Trajetória e Simulação para Robôs Manipuladores.** Dissertação de Mestrado da Universidade Federal de Pernambuco, Curso de Engenharia Mecânica, 2003.

FIGUEIRA, Ricardo Jorge Costa de Moraes. **CAD/CAE/CAM/CIM.** Instituto Politécnico do Porto, Instituto Superior de Engenharia do Porto, Departamento de Engenharia Informática, 2002/2003.

GIBBS, D. **CNC part programming: a practical guide.** London: Cassell Publishers Limited, 1994. 186p.

GROOVER, M.P. **Automation, production systems and computer integrated manufacturing.** Englewood Cliffs: Prentice-Hall, 1987.Vol1. 357p.

HEIDENHAN; **User's Manual Conversational Programming,** Agosto 1998.

IATAGAN, Francisco. **Programação Parametrizada.** Disponível em:

<<http://www.angelfire.com/va3/aco100/paramacro/index.htm>>

KANITAR, Fátima Passos. **Análise do Desenvolvimento dos Sistemas CAD/CAE/CAM no Brasil nos Diversos Setores do Conhecimento sob a Ótica da Propriedade Industrial.** Dissertação de Mestrado do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca – CEFET/RJ, 2005.

LYNCH, M. **Computer numerical control: accessory devices.** New York: McGraw-Hill, 1994. 262p.

LYNCH, M. **The key concepts of CNC.** *Modern Machine Shop*, Cincinnati, vol 69, n11A, pp81-144, April 1997.

MOREIRA, M. E.; “**Mercado de Moldes Abre suas Portas para CAD/CAM**”, *Revista CADesign*; Ano 8, n. 81, Agosto 2002.

MOTA, A.V.C.S. **O que é Controle Numérico.** PGT/USP-Política e Gestão Tecnológica da Universidade de São Paulo, 2002.

Mundo CNC: Disponível em: <<http://www.mundocnc.com.br/avan2.htm>> .

OLIVEIRA, Flávio Zica de & RUBIO, Juan Carlos Campos. **Cenário de Manufatura Integrada para Produção de Moldes e Matrizes.** Universidade Federal de Minas Gerais, Laboratório de Usinagem e Automação – DEMEC.

ROMI; Manual de Programação Parametrizada.

SANTOS, Ricardo Adriano dos. **Sistemas CNC.** Disciplina de Usinagem, Curso Técnico em Mecânica, Colégio Técnico Industrial – FUNDACEN. Disponível em <<http://www.netpar.com.br/rsantos/cncz.htm>>

SHULZ, H.; FECHTER, T.A . **Defizite derheutigen werkstück-Programmierung.** *Werkstatt und Betrieb*, nr127, p18-21, 1994.

SILVA, Sidnei Dominques. **CNC Programação de Comando Numérico Computadorizado Torneamento.** Editora Érica ,2004. pg. 104-105.

SIMON, A.; AGOSTINHO, O .L. **Condições de utilização da tecnologia CNC: um estudo para máquinas-ferramenta de usinagem na indústria brasileira.** Campinas, SP, 2001., Dissertação de Mestrado. 135p. Campinas, UNICAMP, 2001.

SIMON, Alexandre Tadeu; MAESTRELLI, Nelson Carvalho ; AGOSTINHO, Oswaldo Luiz . **Considerações sobre as Técnicas de Programação CNC Utilizadas no Parque Industrial Brasileiro.** Disponível em <<http://www.simpep.feb.unesp.br/anais10/gestaoqualidprodutividade/>>

SOUZA,A.F. **Conceitos Avançados Mundo CNC.** Disponível em: <<http://www.mundocnc.com.br/avan2.htm>> .

SPECK, Henderson José. **Proposta de Método para Facilitar a Mudança das Técnicas de**

Projetos: da Prancheta à Modelagem Sólida (Cad) para Empresas de Engenharia de Pequeno e Médio Porte. Tese (Doutorado em Engenharia de Produção) – Universidade Federal de Santa Catarina, Florianópolis, 2005.

Telecurso 2000. Disponível em <<http://www.bibvirt.futuro.usp.br/>>

Usinagem Brasil: Disponível em: <<http://www.usinagem-brasil.com.br/index2.jsp>>

WILSON, F. **Numerical control in manufacturing.** USA: Mc-Graw Hill Book Company, 1980.

ZEILMANN, R. P. **Comando Numérico Computadorizado CNC.** Disponível em: <<http://hermes.ucs.br/ccet/demc/rpzeilma/Docs/>>