

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

**UMA METODOLOGIA DE INTEGRAÇÃO CAD/CAM  
ATRAVÉS DA APLICAÇÃO DE CICLOS DE  
USINAGEM NA PROGRAMAÇÃO CNC**

**DISSERTAÇÃO DE MESTRADO**

**CLÁUDIO ANDRÉ LOPES DE OLIVEIRA**

**Santa Maria, RS, Brasil**

**2011**

**UMA METODOLOGIA DE INTEGRAÇÃO CAD/CAM  
ATRAVÉS DA APLICAÇÃO DE CICLOS DE USINAGEM NA  
PROGRAMAÇÃO CNC**

**Cláudio André Lopes de Oliveira**

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

**Orientador: Prof. Dr. Alexandre Dias da Silva**

**Santa Maria, RS, Brasil**

**2011**

Ficha catalográfica elaborada através do Programa de Geração Automática da Biblioteca Central da UFSM, com os dados fornecidos pelo(a) autor(a).

oliveira, Cláudio André Lopes de  
Uma Metodologia de Integração CAD/CAM Através da  
Aplicação de Ciclos de Usinagem na Programação CNC /  
Cláudio André Lopes de oliveira.-2011.  
129 p.; 30cm

Orientador: Alexandre Dias da Silva  
Dissertação (mestrado) - Universidade Federal de Santa  
Maria, Centro de Tecnologia, Programa de Pós-Graduação em  
Engenharia de Produção, RS, 2011

1. Programação CNC 2. Ciclos de usinagem 3. Sistema  
CAD/CAM 4. Comando numérico I. Silva, Alexandre Dias da  
II. Título.

---

© 2011

Todos os direitos autorais reservados a Cláudio André Lopes de Oliveira. A reprodução de partes ou do todo deste trabalho só poderá ser feita mediante a citação da fonte.

Endereço: Rua Doze, n. 2010, Bairro da Luz, Santa Maria, RS. CEP: 97110-680

Fone (0xx)55 32225678; Fax (0xx) 32251144; E-mail: ufesme@ct.ufsm.br

---

**Universidade Federal de Santa Maria**

**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

**UMA METODOLOGIA DE INTEGRAÇÃO CAD/CAM ATRAVÉS DA  
APLICAÇÃO DE CICLOS DE USINAGEM NA PROGRAMAÇÃO CNC**

elaborada por  
**Cláudio André Lopes de Oliveira**

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

**COMISSÃO EXAMINADORA**

**Alexandre Dias da Silva, Dr. Eng.**  
(Presidente/Orientador)

**Leandro Costa de Oliveira, Dr. Eng.**

**Moacir Eckhardt, Dr. Eng.**

Santa Maria, Dezembro de 2011.

## **AGRADECIMENTOS**

A Deus por ter me mantido e me dado força para superar as inúmeras dificuldades que surgiram durante essa caminhada e por tornar possível mais essa conquista em minha vida.

À minha esposa Mariester Lira pelo carinho, amor, atenção, compreensão e paciência. A tua presença em minha vida foi muito importante para a realização deste trabalho.

Ao meu pai Alceu e a minha mãe Maria Elizabete (in memoriam), que me deram a base para chegar até aqui.

A minha irmã Débora e ao meu cunhado André, pelo apoio, confiança, alegria e estímulo constante.

Ao professor e orientador Alexandre Dias da Silva pela orientação, dedicação, ensinamentos e pela sua amizade. Muito obrigado pela ajuda.

Aos meus amigos Leandro e Leandra pelos momentos agradáveis de descontração, pelo carinho e atenção. Isso foi importante para recuperar o ânimo e continuar o desenvolvimento das atividades.

Enfim, agradeço a todas as pessoas que, de certa forma, ajudaram na realização deste trabalho e às quais presto minha homenagem.

## RESUMO

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

### **UMA METODOLOGIA DE INTEGRAÇÃO CAD/CAM ATRAVÉS DA APLICAÇÃO DE CICLOS DE USINAGEM NA PROGRAMAÇÃO CNC**

AUTOR: CLÁUDIO ANDRÉ LOPES DE OLIVEIRA

ORIENTADOR: ALEXANDRE DIAS DA SILVA

Data e Local da Defesa: Santa Maria, 23 de dezembro de 2011.

A utilização de recursos computacionais para a programação de máquinas com Comando Numérico Computadorizado (CNC) tem se tornado cada vez mais necessária nas empresas de manufatura. Esses recursos permitem a redução dos tempos improdutivos, a otimização do processo de programação, a diminuição da possibilidade de erro causado pela programação manual e do tempo de máquina parada. Os *softwares* utilizados para a programação de máquinas CNCs, conhecidos pela sigla CAM (Manufatura assistida por computador), são programas que, entre outras funções, possibilitam a geração automática de programas CNC através do cálculo do caminho da ferramenta, a partir da representação geométrica da peça disponível na forma virtual em sistemas computacionais de auxílio ao projeto – CAD (desenho assistido por computador). Da integração desses sistemas, têm-se os sistemas CAD/CAM, que hoje, devido ao avançado grau de desenvolvimento das tecnologias, auxiliam na criação de formas e manipulação de dimensões possibilitando a programação de diversos tipos de equipamentos comandados numericamente. Devido à gama de recursos oferecida por esses programas, seu custo é elevado, o que torna sua aquisição inviável para pequenas empresas, que, muitas vezes, não necessitam de todos esses recursos por trabalharem com geometria de pouca complexidade. Levando em consideração esses aspectos, o objetivo deste trabalho é desenvolver uma metodologia de integração CAD/CAM de baixo custo para aplicação de ciclos fixos para usinagem em fresamento e furação, existentes no comando numérico de máquinas-ferramenta. Para analisar a técnica, foi desenvolvido um sistema utilizando um *software* de CAD, a partir do qual, através de seus recursos de otimização e automatização, foram criadas rotinas para programação dos ciclos fixos de uma fresadora CNC de 3 eixos programáveis. Os resultados demonstraram a viabilidade dessa proposta, pois foi possível obter um sistema CAD/CAM para aplicação específica.

**Palavras chave:** CNC; ciclos de usinagem; usinagem, sistemas CAD/CAM.

## **ABSTRACT**

Master's Dissertation  
Post-Graduation Program in Production Engineering  
Federal University of Santa Maria

### **A CAD/CAM INTEGRATED METHODOLOGY THROUGH THE APPLICATION OF MACHINING CYCLES IN CNC PROGRAMMING**

AUTHOR: CLÁUDIO ANDRÉ LOPES DE OLIVEIRA  
ADVISOR: ALEXANDRE DIAS DA SILVA

Date and Place of defense: Santa Maria, the 23th of December of 2011

The use of computer resources for machinery programming with Computer Numeric Control (CNC) has become increasingly more necessary in manufacturing companies. The objective is to reduce unproductive time, to optimize programming process, to decrease possibility of error caused by manual programming and to decrease the time that the machine is not being used. Softwares employed for the programming of CNC machines, known as CAM, are programs that, among other functions, enable the automatic generation of CNC programs, through the calculus of the tool path, from the geometric representation of the available part in computer systems presented virtually. When their integration occurs, we obtain CAD/CAM systems that currently, due to the great advance of technologies, help the creation of shapes and the handling of dimensions, thus enabling the programming of several kinds of numerically-driven equipment. Due to a great array of resources offered by these programs, its cost is high. Therefore, the purchase of these programs by companies is impracticable. In addition, these companies do not need all the great array of resources to deal with low complexity geometry. By considering these aspects, the objective of this study is to develop a low-cost CAD/CAM integrated methodology to apply in fixed cycles for milling and drilling, existing in the numeric comand of machine tools. In order to analyze the technique, a system using a CAD software was designed. Through its optimization resources, routines for programming fixed cycles of a machine tool (milling) with three-axle programmable CNC were created. The results demonstrated viability of this proposal as it was possible to obtain a CAD/CAM system for specific application.

**Key Words:** CNC; machining Cycles; machining, CAD/CAM systems.

## LISTA DE FIGURAS

Figura 1 – Movimento rápido G00 .....	27
Figura 2 – Movimento linear G01 .....	28
Figura 3 – Movimento circular G02 e G03.....	28
Figura 4 – Geração Automática para Rasgos Radiais para Máquinas CNC .....	37
Figura 5 – Ciclo de usinagem para torneamento simples .....	39
Figura 6 – Ciclo de usinagem para fresamento de alojamento .....	40
Figura 7 – Ciclos de usinagem para centro de usinagem .....	41
Figura 8 – Ciclo fixo do programador .....	42
Figura 9 – Fluxograma da estrutura básica do sistema.....	45
Figura 10 – Extração dos dados de uma entidade no AutoCAD .....	47
Figura 11 – Ciclo de usinagem de círculo de furos.....	51
Figura 12 – Aplicação da função G24 .....	52
Figura 13 – Ciclo de usinagem quadrado ou retangular de furos.....	53
Figura 14 – Aplicação da função G25 .....	54
Figura 15 – Ciclo de fresamento de cavidade retangular ou circular.....	54
Figura 16 – Aplicação da função G26 .....	56
Figura 17 – Ciclo de fresamento de resalto ao redor de uma área especificada .....	56
Figura 18 – Aplicação da função G26 .....	58
Figura 19 – Forma de codificação das <i>layers</i> .....	59
Figura 20 – Modelos de peça bruta.....	60
Figura 21 – Processo de extração entre volumes sólidos do CAD.....	61
Figura 22 – Entidades de armazenamento de dados do programa G24.....	63
Figura 23 – Entidades de armazenamento de dados do programa G25.....	64
Figura 24 – Entidades de armazenamento de dados do programa G26 retangular ..	65
Figura 25 – Entidades de armazenamento de dados do programa G26 circular .....	66
Figura 26 – Entidades de armazenamento de dados do programa G27 retangular ..	67
Figura 27 – Entidades de armazenamento de dados do programa G27 circular .....	67
Figura 28 – Estrutura geral do aplicativo .....	71
Figura 29 – Fluxo de informações entre programa e subprograma.....	73
Figura 30 – Fluxograma do subprograma “Atualiza contador de ciclo” .....	74
Figura 31 – Funcionamento do subprograma “Atualiza Contador de Ciclo” .....	75
Figura 33 – Fluxograma do subprograma “Definição peça bruta” .....	76

Figura 32 – Entidades de armazenamento de dados da peça bruta .....	77
Figura 34 – Fluxograma do subprograma “Novo programa CN” .....	78
Figura 35 – Fluxograma do subprograma “Salvar CN” .....	79
Figura 36 – Fluxograma do subprograma “Ler banco de dados ft” .....	80
Figura 37 – Fluxograma do subprograma “Achar ferramenta” .....	81
Figura 38 – Fluxograma do subprograma “Gera programa CN” .....	83
Figura 39 – Fluxograma do “programa G24” .....	86
Figura 40 – Fluxograma do “programa G25” .....	91
Figura 41 – Fluxograma do “programa G26” .....	94
Figura 42 – Método de construção do desenho para o ciclo G27 .....	95
Figura 43 – Fluxograma do “programa G27” .....	96
Figura 44 – Dados extraídos das entidades do ciclo G24 .....	98
Figura 45 – Dados extraídos das entidades do ciclo G25 .....	100
Figura 46 – Dados extraídos das entidades do ciclo G26 .....	103
Figura 47 – Dados extraídos das entidades do ciclo G27 .....	104
Figura 48 – Modo de acesso aos recursos do aplicativo .....	105
Figura 49 – Dados da peça bruta .....	106
Figura 50 – Dados da ferramenta .....	107
Figura 51 – Geração do programa CN .....	110
Figura 52 – Centro de usinagem utilizado no teste prático. ....	111
Figura 53 – Peça teste 1 .....	112
Figura 54 – Desenho do ciclo 26 retangular .....	113
Figura 55 – Desenho do ciclo 26 circular .....	114
Figura 56 – Desenho do ciclo 24 .....	115
Figura 57 – Programa CN da peça teste 1 .....	116
Figura 58 – Sequência de usinagem da peça teste 1 .....	117
Figura 59 – Peça teste 2 .....	118
Figura 60 – Desenho do ciclo 27 .....	119
Figura 61 – Desenho do ciclo 25 .....	120
Figura 62 – Programa CN da peça teste 2 .....	121
Figura 63 – Sequência de usinagem da peça teste 2 .....	122

## LISTA DE TABELAS

Tabela 1 – Caracteres presentes em um programa CNC .....	24
Tabela 2 – Funções preparatórias padrão ISO 1056 .....	25
Tabela 3 – Funções miscelâneas.....	26
Tabela 4 – Programa CNC para usinagem do perfil da figura 5.....	40
Tabela 5 – Listas de dados da entidade linha .....	48
Tabela 6 – Listas de dados da entidade círculo .....	49
Tabela 7 – Listas de dados da entidade arco.....	49
Tabela 8 – Listas de dados da entidade texto .....	50
Tabela 9 – Parâmetros do banco de dados das ferramentas.....	69
Tabela 10 – Parâmetros para programação do ciclo de usinagem G24 .....	107
Tabela 11 – Parâmetros para programação do ciclo de usinagem G25 .....	108
Tabela 12 – Parâmetros para programação do ciclo de usinagem G26 .....	108
Tabela 13 – Parâmetros para programação do ciclo de usinagem G27 .....	109

## LISTA DE SIGLAS

ANSI	<i>American National Standards Institute</i>
APT	<i>Automatically Programed Tool</i> (Ferramenta Programada Automaticamente)
CAD	<i>Computer Aided Design</i> (Desenho assistido por Computador)
CAE	<i>Computer Aided Engineering</i> (Engenharia assistida por Computador)
CAM	<i>Computer Aided Manufacturing</i> (Manufatura assistida por Computador)
CNC	Comando Numérico Computadorizado
CPU	<i>Central Processing Unit</i> (Unidade Central de Processamento)
ISO	<i>International Organization for Standardization</i> (Organização Internacional para Normalização)
MCU	Unidade de Controle de Máquina
NC	<i>Numeric Control</i> (Comando Numérico)
CLDATA	<i>Cutter Location Data</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>14</b>
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA .....</b>	<b>17</b>
<b>2.1</b>	<b>Comando numérico computadorizado .....</b>	<b>17</b>
<b>2.2</b>	<b>Aplicação do CNC .....</b>	<b>20</b>
<b>2.3</b>	<b>Estrutura do programa CN.....</b>	<b>22</b>
2.3.1	Sistema ISO de programação (linguagem G) .....	23
2.3.2	Movimentos básicos.....	26
<b>2.4</b>	<b>Programação CNC.....</b>	<b>29</b>
2.4.1	Programação manual .....	30
2.4.2	Linguagem de alto nível .....	31
2.4.3	Sistemas gráficos interativos.....	33
2.4.4	Sistemas CAD/CAM .....	33
2.4.5	Integração CAD/CNC .....	35
<b>2.5</b>	<b>Ciclos de usinagem em Comandos Numéricos.....</b>	<b>38</b>
2.5.1	Ciclos com funções do comando.....	39
2.5.2	Ciclos parametrizados.....	41
<b>3</b>	<b>METODOLOGIA .....</b>	<b>44</b>
<b>3.1</b>	<b>Estrutura geral da metodologia proposta .....</b>	<b>44</b>
<b>3.2</b>	<b>Desenvolvimento de aplicativos no AutoCAD .....</b>	<b>45</b>
3.2.1	Banco de dados das entidades do AutoCAD .....	46
<b>3.3</b>	<b>Ciclos de usinagem do sistema .....</b>	<b>50</b>
3.3.1	Ciclo de usinagem de círculo de furos .....	51
3.3.2	Ciclo de usinagem quadrado ou retangular de furos.....	52
3.3.3	Ciclo de usinagem para fresamento de cavidade retangular, quadrado ou circular.....	54
3.3.4	Ciclo de usinagem para fresamento de resalto ao redor de uma área especificada .....	56
<b>3.4</b>	<b>Codificação dos Ciclos de Usinagem.....</b>	<b>58</b>

<b>3.5</b>	<b>Desenho da peça</b> .....	<b>60</b>
<b>3.6</b>	<b>Parâmetros geométricos</b> .....	<b>61</b>
3.6.1	Ciclo de furação circular G24 .....	62
3.6.2	Ciclo de furação linear G25.....	63
3.6.3	Ciclo de usinagem G26 para cavidade retangular.....	64
3.6.4	Ciclo de usinagem G26 para cavidade circular .....	65
3.6.5	Ciclo de usinagem G27 para fresamento de resalto retangular e circular.....	66
<b>3.7</b>	<b>Leitura dos parâmetros geométricos</b> .....	<b>67</b>
<b>3.8</b>	<b>Banco de dados de ferramentas</b> .....	<b>68</b>
<b>4</b>	<b>DESENVOLVIMENTO DA METODOLOGIA PROPOSTA</b> .....	<b>45</b>
<b>4.1</b>	<b>Desenvolvimento dos programas auxiliares</b> .....	<b>72</b>
4.1.2	Subprograma para determinar a codificação do ciclo de usinagem .....	73
4.1.3	Subprograma para desenhar a peça bruta.....	75
4.1.4	Subprograma para atribuir blocos iniciais do programa CN a uma variável.....	77
4.1.5	Subprograma salvar programa CN.....	78
4.1.6	Subprograma para ler o banco de dados das ferramentas .....	79
4.1.7	Seleção de ferramenta .....	81
4.1.8	Programa para identificar o ciclo de usinagem programado .....	82
<b>4.2</b>	<b>Definição geométrica dos ciclos</b> .....	<b>84</b>
4.2.1	Furação em Padrão Circular .....	84
4.2.2	Furação em Padrão Linear.....	89
4.2.3	Usinagem de Cavidade .....	92
4.2.4	Usinagem de resalto ao redor de uma área .....	95
<b>4.3</b>	<b>Geração do programa CN</b> .....	<b>97</b>
4.3.1	Geração do programa do ciclo de usinagem em Padrão Circular .....	97
4.3.2	Geração do programa do ciclo de usinagem em Padrão linear .....	99
4.3.3	Geração do programa do ciclo de usinagem de cavidades.....	102

4.3.3	Geração do programa do ciclo de resalto ao redor de uma área .....	104
<b>4.4</b>	<b>Estrutura funcional do aplicativo .....</b>	<b>105</b>
4.3.1	Inicialização do aplicativo .....	105
4.3.2	Definição da peça bruta .....	105
4.3.3	Definição da ferramenta .....	106
4.3.4	Dados geométricos do ciclo de usinagem .....	107
4.3.5	Geração do programa CN .....	109
<b>5</b>	<b>APLICAÇÃO E ANÁLISE .....</b>	<b>111</b>
<b>5.1</b>	<b>Peça teste 1 .....</b>	<b>112</b>
5.1.1	Etapas de usinagem.....	112
5.1.2	Geração do programa CN .....	116
5.1.3	Usinagem da peça teste.....	116
<b>5.2</b>	<b>Peça teste 2.....</b>	<b>117</b>
5.2.1	Etapas de usinagem.....	118
5.2.2	Geração do programa CN .....	121
5.2.3	Usinagem da peça teste.....	121
<b>5.3</b>	<b>Análise.....</b>	<b>122</b>
<b>6</b>	<b>CONCLUSÃO .....</b>	<b>124</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>126</b>

# 1 INTRODUÇÃO

O surgimento do Comando Numérico contribuiu para uma nova etapa no processo de manufatura. A possibilidade de controlar simultaneamente vários eixos de uma máquina-ferramenta através de um sistema computadorizado permitiu que a usinagem de peças com geometrias complexas fosse realizada de modo relativamente simples, com grande precisão e com mínima intervenção humana.

Essa tecnologia tem se mostrado mais viável na produção de pequenos e médios lotes, devido a sua capacidade de repetibilidade dos movimentos e às altas velocidades de deslocamento dos eixos das modernas máquinas-ferramentas. Tais características, associadas a sua grande precisão de posicionamento, têm contribuído cada vez mais para a redução do tempo de produção, possibilitando assim uma maior flexibilização das linhas de produção (GROOVER, 2011). Essa flexibilização permite uma mudança rápida de tarefa em um equipamento para que determinada tarefa seja completada ou para que outra tarefa seja realizada (BROWNE, 1984).

Paralelo à evolução da tecnologia do Comando Numérico, os sistemas CAD (*Computer Aided Design* - Desenho Auxiliado por Computador) e os sistemas CAM (*Computer Aided Manufacturing* - Fabricação Assistida por Computador) também evoluíram e hoje estão amplamente introduzidos nos meios industriais. Integrados a novas ferramentas como, por exemplo, CAE (*Computer Aided Engineering* - Engenharia Auxiliada por Computador), CAPP (*Computer Aided Process Planning* - Planejamento do Processo Assistido por Computador) e CAI (*Computer Aided Inspection* - Inspeção Assistida por Computador), tornaram-se um poderoso equipamento de gerenciamento de projeto e manufatura, atendendo pela sigla CIM (*Computer Integrated Manufacturing* – Manufatura Integrada por Computador ). O CIM é um sistema que integra todos os aspectos da manufatura, permitindo a automação flexível de todas as atividades e assegurando a coordenação e otimização de todo o sistema (WANG; LI, 1991).

Esses sistemas são ferramentas que podem ser utilizadas separadamente ou em conjunto, possibilitando a integração de profissionais de diferentes áreas a fim de alcançar um único objetivo. Tal integração é facilitada uma vez que esses sistemas não necessitam de conceitos prévios de desenho técnico, como a projeção de vistas

bidimensionais. Além disso, a utilização de geometria tridimensional facilita a visualização dos elementos que compõem o projeto, oferecendo uma visão mais realista do produto.

Apesar de atualmente existirem diversos sistemas computacionais de auxílio à manufatura que possibilitam o desenvolvimento de produtos, o controle, o gerenciamento e o monitoramento fabril, ainda é difícil integrá-los, pois uma empresa de médio ou pequeno porte, por exemplo, além do custo, terá de contratar profissionais preparados para utilizá-los. Por esse motivo, os sistemas CAD e os sistemas CAM são as ferramentas computacionais mais utilizadas nas empresas que atuam na área de manufatura.

Os sistemas CAM têm hoje como função básica a geração das trajetórias das ferramentas sobre uma geometria previamente definida no CAD. Segundo Groover (2011), o termo CAM indica o amplo uso do computador e da tecnologia da informática nos sistemas de produção para auxiliar em todas as fases da manufatura de um produto incluindo:

- processos;
- planejamento de produção;
- usinagem;
- gerenciamento e controle de qualidade.

Desenvolver funções CAD/CAM é uma alternativa ao uso de *software* comercial destinado à programação automática de máquinas CNC. Assim, o presente trabalho tem como objetivo, demonstrar uma metodologia de desenvolvimento de um sistema CAD/CAM para fresamento, aplicada para programação de ciclos de usinagem, de modo a possibilitar o uso dos recursos dos comandos numéricos. Atualmente, a programação deste recurso só é possível diretamente no comando da máquina, o que em muitos equipamentos exige a parada dos mesmos.

A metodologia proposta neste trabalho é desenvolvida em ambiente CAD, através de recursos utilizados para sua automatização. Como aplicação específica, busca-se o desenvolvimento de uma interface que possibilite a fácil entrada de informações tecnológicas de programação CNC. O sistema desenvolvido visa mostrar a viabilidade de implementação da metodologia por pequenas e médias

empresas, devido ao seu baixo custo e a sua conformidade com aplicações específicas em determinados tipos de usinagem.

Para a realização deste trabalho, fez-se uma revisão bibliográfica acerca dos seguintes temas: Comandos Numéricos Computadorizados, suas aplicações e tipos de programação, os sistemas CAD e os sistemas CAM e suas aplicações, seus recursos e sua programação. Essa revisão foi feita em livros, artigos e revistas nacionais, internacionais e disponíveis na internet, entre outros.

## **2 REVISÃO BIBLIOGRÁFICA**

### **2.1 Comando numérico computadorizado**

De acordo com Groover (2011), o comando numérico computadorizado - CNC é uma forma de automação programável em que as ações mecânicas de uma máquina-ferramenta, ou outro equipamento, é controlado por um programa contendo dados alfanuméricos codificados. Estes dados são codificados em um formato apropriado que definem um programa de instruções para a fabricação de uma determinada peça ou execução de uma determinada tarefa.

Segundo Cassaniga (2000) O comando numérico computadorizado, é um dispositivo eletrônico que recebe as informações sobre a forma que o equipamento ou máquina-ferramenta irá realizar a operação por meio de linguagem própria denominada programa CNC, processa essas informações, devolvendo-as ao sistema através de impulsos elétricos. Estes sinais elétricos são responsáveis pelos acionamentos de motores, que darão às máquinas os movimentos desejados com todas as características, possibilitando a realização das operações sequenciais sem intervenção do operador.

A tecnologia CNC engloba diversos processos automáticos de fabricação, podendo ser aplicada desde o desenvolvimento de um produto até a sua concepção. O campo de maior utilização dessa tecnologia é a área de fabricação, sendo aplicada aos processos de fresamento, torneamento, dobramento, corte, entre outros. Sua aplicação no controle de máquinas permite a realização de tarefas repetitivas e de grande complexidade cinemática, possibilitando a reprodutibilidade de produtos de variadas formas geométricas (COSTA; PEREIRA, 2006).

De acordo com Machado (1990), desde tempos remotos o homem procura por técnicas que venham a contribuir com a racionalização de seu trabalho em todas as áreas produtivas com o objetivo de economizar tempo e material, otimizando assim a qualidade dos serviços desenvolvidos e o produto final, tornando-se mais produtivo e competitivo. A automação simplifica todo tipo de trabalho, seja ele físico ou mental. O exemplo mais comum da automação do trabalho mental é o uso da calculadora eletrônica.

No cotidiano observa-se cada vez mais a automação e a racionalização dos trabalhos físicos em geral. Na agricultura, por exemplo, veem-se novos e sofisticados tratores que substituem a enxada e outros meios de produção. A cada geração de novos produtos, observa-se uma evolução que faz com que os esforços físicos e mentais sejam reduzidos. O processo de pesquisa para melhoria dos produtos, aliado ao desenvolvimento dos computadores, permitiu o surgimento das primeiras máquinas controladas numericamente.

O principal fator que impulsionou a indústria a essa busca foi a segunda guerra mundial. Tal período exigiu um ritmo de produção em larga escala e grande precisão de aviões, tanques, barcos, navios, armas, caminhões etc. Grande parte da mão de obra masculina especializada utilizada pelas fábricas foi substituída pela feminina, o que na época implicou a necessidade de treinamento, com reflexos na produtividade e na qualidade. Era o momento certo para se desenvolver máquinas automáticas de grande produção, com peças de precisão e que não dependessem da qualidade da mão de obra aplicada. Diante desse desafio, iniciou-se o processo de pesquisa que resultou na máquina comandada numericamente - CNC.

Machado (1990) e Groover (2011) publicaram um histórico do surgimento e da evolução do Comando Numérico. Resumidamente, pode-se dizer que:

- O primeiro trabalho nesta área é atribuído a John Parsons e Frank Stulen da Parsons Corporation;
- Parsons já havia experimentado o conceito de posicionamento de coordenadas utilizando cartão perfurado durante a década de 40;
- 1949 – A Parsons subcontrata o Instituto de Tecnologia de Massachusetts (MIT) para desenvolver estudos de sistemas sobre controle de máquinas-ferramentas e desenvolver um protótipo baseados em seus estudos;
- 1951 - O MIT e as Forças Aéreas Americanas assinam contrato para completar o trabalho de desenvolvimento;
- 1951 - O nome “Comando Numérico” é adotado após um concurso realizado entre a equipe do MIT que trabalhava no projeto, patrocinado por John Parsons;
- 1952 - Entra em operação com sucesso a primeira máquina comandada numericamente, desenvolvida através do processo de

*retrofitting* de uma fresadora Hydro-tel da Cincinnati; seu sistema controlador era maior do que a própria máquina;

- 1956 - Forças Aéreas dos EUA decidem patrocinar o desenvolvimento de máquinas-ferramentas;
- 1956 - Os avanços tecnológicos dos computadores estimularam mais desenvolvimento e o MIT demonstrou a viabilidade de um sistema de programação assistido por computador;
- 1958 – É desenvolvida a linguagem APT de programação;
- 1960 – É desenvolvido o sistema Controle Numérico Direto (DNC), permitindo que computadores de grande porte controlassem a máquina à distância;
- 1961 - O *Illinois Institute of Technology Research* é escolhido como responsável pela manutenção e atualização a longo prazo da linguagem APT;
- O desenvolvimento do CNC dá-se a partir da década de 70 com o surgimento dos computadores dedicados.
- 1977 - Surgem as máquinas com comandos CNC usando a tecnologia dos microprocessadores, passando a se chamar Comando Numérico Computadorizado - CNC;
- 1980 - Sistemas flexíveis de fabricação são aplicados em larga escala.

O desenvolvimento da tecnologia do comando numérico computadorizado acompanhou a evolução da tecnologia dos computadores em que máquinas ferramentas de CN passaram a incorporar microprocessadores e sistemas eletrônicos em sua estrutura de controle (MENEGHELLO, 2003). Segundo Tauile (1982) e Machado (1990), no Brasil a mais antiga máquina-ferramenta comandada numericamente é uma fresadora Kerney & Tracker, Milwaukee Matic II, com NC GE em 1967, podendo ser considerada a primeira máquina com essa tecnologia no Brasil. A difusão desta tecnologia começou em 1972 com a importação de 16 unidades, sendo que em 1970 apenas duas máquinas haviam sido importadas. Em 1972 foi vendida a primeira máquina-ferramenta CN fabricada no Brasil, um Torno convencional feito pela Romi, adaptado para funcionar com controle numérico DCE

480 (TAUILE, 1982). Somente a partir de 1975 esta empresa produziu outra unidade de CN, um modelo novo e revisado.

Atualmente quase não existem produtos fabris que não estejam de alguma forma relacionada à tecnologia inovadora destas máquinas-ferramentas. Sua aplicação é cada vez mais ampla nas indústrias de transformação, tornando necessário o conhecimento de todo o processo envolvido no uso destes equipamentos por parte dos profissionais.

Porém, as máquinas comandadas numericamente não podem ser consideradas um milagre da tecnologia industrial na solução de todos os problemas, uma vez que as mesmas têm, por suas próprias características, campos de aplicação bem definidos (MACHADO, 1990).

## 2.2 Aplicação do CNC

Dentro do processo de manufatura, são inúmeras as aplicações do comando numérico. Groover (2011) divide-as em duas categorias:

- **Aplicação de usinagem** - é em geral associada às indústrias de metalurgia, onde essa tecnologia está aplicada diretamente às máquinas-ferramenta.
- **Aplicação não destinada à usinagem** - abrange um grupo diversificado de operações de outras indústrias.

A área da usinagem é o campo que apresenta uma maior aplicação deste sistema, voltado ao controle das máquinas-ferramenta. A possibilidade de controlar o caminho que uma ferramenta de corte irá percorrer e, com isso, poder criar a geometria desejada, torna essa tecnologia fundamental nas indústrias metalúrgicas. Hoje diversas máquinas-ferramenta utilizadas nas operações usinagem são comandadas numericamente, tornando assim a taxa de produção relativamente alta para o fornecimento de peças de alta precisão a um custo relativamente baixo (GROOVER, 2011).

A aplicação do comando numérico em outras áreas que não a usinagem tem sido cada vez maior. Exemplo disso é a sua aplicação no processo produtivo com o

objetivo de otimizar a produção e reduzir o risco à saúde do operado, atuando em trabalhos repetitivos ou de grande complexidade e em locais onde uma pessoa estaria exposta a riscos químicos ou ergonômicos.

O comando numérico computadorizado fornece uma série de vantagens quando comparado aos métodos de produção convencionais. Essas vantagens são traduzidas em economias para a empresa usuária. Todavia, o CNC envolve uma tecnologia mais sofisticada do que a utilizada por métodos convencionais de produção, o que torna o custo inicial desta tecnologia muito superior.

Algumas vantagens da aplicação do CNC são:

- Redução do tempo passivo, ou seja, do tempo em que a ferramenta não está promovendo o corte. Essa vantagem é visível em máquinas de usinagem e de corte a *laser* que chegam a uma aceleração acima de 2G e velocidade de deslocamento rápido acima de 200 m/min;
- Maior controle no consumo de ferramenta. Por se tratar de uma usinagem com esforços e velocidades constantes, uniformes e repetitivos, os desgastes estão sob controle;
- Aumenta o volume de produção, reduzindo o custo da mesma;
- Aumenta a precisão da peça produzida;
- Diminui o tempo e desenho da peça e a sua produção;
- Rápida preparação da máquina, principalmente quando o programa CNC já estiver otimizado, e o ferramental de corte e os meios de fixação disponíveis;
- Alta flexibilidade no trabalho. Em função da rápida preparação da máquina, torna-se econômica a produção também de pequenas e médias séries. Máquinas CNC são fundamentais quando se opera em trabalhos *just in time*;
- Fácil e rápida alteração do programa CN. Alterações de dimensões da peça de trabalho e de parâmetros de corte, como avanços e velocidades de corte, são realizadas rapidamente mesmo durante a produção;
- Correção de dimensões da peça e medidas durante o processo. As correções de medidas, devido ao desgaste das ferramentas de corte, são feitas rapidamente pela introdução dos incrementos, deixando as peças dentro das suas tolerâncias;

- Trabalho com parâmetros de corte otimizados. Como os avanços dos carros porta-ferramentas e a rotações do fuso principal e das ferramentas acionadas são programáveis de forma contínua e sem escalonamentos, é possível trabalhar nas condições ideais em função do material que está sendo usinado, das tolerâncias e do grau de acabamento superficial;
- Altas rotações do fuso principal. Esta condição faz com que se atinjam tempos de ciclo mais rápidos e acabamentos superficiais ideais, além de permitir a usinagem de aços endurecidos;
- Altas velocidades de avanço rápido. Esta característica é fundamental para se diminuir os tempos mortos nas aproximações e nos retrocessos das ferramentas de corte;
- Conforto operacional, tanto na preparação da máquina como na sua manutenção;
- Menor necessidade de manutenção mecânica. Como os acionamentos são feitos através de eixo árvore e servomotores de alto rendimento, fica dispensado o uso de caixas de câmbio, trens de engrenagens, embreagens e outros elementos mecânicos de máquinas.

### 2.3 Estrutura do programa CN

Um programa CN é composto por uma sequência de sentenças lógicas que contém informações geométricas e tecnológicas da tarefa a ser executada em cada operação. Esse programa é produzido numa linguagem em que o sistema operacional compreende. A maioria dos Controles Numéricos Computadorizados seguem os códigos normalizados da ISO 1056 (*International Standard Organization*) e da Associação Alemã de Normas Técnicas DIN 66025. Esses códigos, colocados em uma sequência lógica, permitem que a máquina-ferramenta execute os movimentos entre a ferramenta e a peça. Tal movimentação torna possível a usinagem de uma peça (SILVA, 2008).

As linhas do programa CNC são denominadas de blocos. Cada bloco é formado por comandos, também chamados de palavras, que controlam as funções

da máquina. Essas palavras podem indicar movimentos dos elementos da máquina, coordenadas da geometria da peça e funções da máquina. Silva (2008) apresenta a seguinte estrutura do programa CNC:

- **Bloco de dados ou sentença:** é constituído por caracteres, ou seja, letras de endereçamento e algarismos, palavras que juntas têm o objetivo de informar ao comando as operações que devem ser executadas.
- **Caracteres:** é um número letra, espaço, ponto ou qualquer outro símbolo que signifique algo para o comando.  
Exemplo: A, T, Z, -, (, etc.
- **Letras de endereçamento:** são instruções alfabéticas passadas para o comando que, por sua vez, podem executar um movimento ou simplesmente assumir uma nova função.  
Exemplo: G, X, O, B, K, I, etc.
- **Palavra:** é constituída por uma letra seguida de um valor numérico. Dependendo da letra, terá de ser específico.  
Exemplo: G0, G1, G66, X55, X20, K5, etc.

### 2.3.1 Sistema ISO de programação (linguagem G)

Segundo Silva (2008), a linguagem G foi adotada pelo sistema ISO como um padrão a ser usado pelos fabricantes de comandos, no entanto, nem todos os códigos estão sob controle da norma ISO. Por isso, o mesmo código pode ter um significado diferente dependendo do comando ou do tipo de máquina (torno ou fresa). A tabela 1 apresenta uma lista com alguns dos caracteres usados na programação.

Tabela 1 – Caracteres presentes em um programa CNC

PREFIXO DE PALAVRA	FUNÇÃO
O	Número do programa
N	Número da linha (bloco)
G	Função preparatória
X	Dado de coordenada no eixo X
Y	Dado de coordenada no eixo Y
Z	Dado de coordenada no eixo Z
I, J, K	Distância incremental ao longo do eixo X, Y, Z / Valores de coordenadas de centro de arco
R	Raio da trajetória da ferramenta
M	Função miscelânea
T	Número da ferramenta
S	Velocidade de rotação do eixo-árvore
F	Velocidade de avanço
U, W	Dados de coordenadas para movimentos incrementais na direção X e Z no torneamento
P	Número de um sub-programa

Fonte: Groover, p. 154, 2011.

Funções preparatórias são conhecidas como o código G, que tem como função ordenar ou ajustar o sistema de comando para obter uma desejada condição, modo ou estado de operação. A tabela 2 apresenta uma lista com algumas funções preparatórias utilizadas na programação.

Tabela 2 – Funções preparatórias padrão ISO 1056

CÓDIGO G	FUNÇÃO
G00	Interpolação linear com avanço rápido
G01	Interpolação linear com avanço programado
G02	Interpolação circular sentido horário com avanço programado
G03	Interpolação circular sentido anti-horário com avanço programado
G04	Tempo de permanência
G17	Seleção do plano de trabalho XY
G18	Seleção do plano de trabalho ZX
G19	Seleção do plano de trabalho YZ
G28	Retorna a posição zero máquina
G40	Cancela compensação de raio de ferramenta
G41	Compensação de raio de ferramenta à esquerda
G42	Compensação de raio de ferramenta à direita
G43	Compensação de comprimento de ferramenta (positivo)
G44	Compensação de comprimento de ferramenta (negativo)
G53	Coordenada de trabalho em relação ao zero máquina
G54 a G59	Coordenadas de trabalho
G80	Cancelamento dos ciclos fixos
G81 a G89	Ciclos fixos
G90	Posicionamento absoluto
G91	Posicionamento incremental
G92	Estabelece sistema de coordenadas
G94	Avanço dado em milímetros por minutos
G95	Avanço em milímetros por rotação
G96	Rotação do fuso dado em velocidade de corte constante
G97	Rotação do fuso dado em RPM

Fonte: Groover, p. 155, 2011.

O endereço M em um programa CNC identifica uma função miscelânea que executa outras funções da máquina não cobertas pelos códigos G. Isso se dá devido ao fato de muitas funções M serem exclusivas de uma máquina particular, o que torna impossível definir todas as suas atribuições específicas. A tabela 3 apresenta uma lista com as funções miscelâneas mais comuns usadas na programação.

Tabela 3 – Funções miscelâneas

CÓDIGO M	FUNÇÃO
M00	Parada do programa
M01	Parada opcional do programa
M02	Fim de programa – mesma função do M30
M03	Liga eixo árvore sentido horário
M04	Liga eixo árvore sentido anti-horário
M05	Desliga eixo árvore
M06	Troca de ferramenta
M07	Liga fluído de corte alta pressão/ centro do eixo árvore
M08	Liga fluído de corte
M09	Desliga fluído de corte
M19	Orientação do eixo árvore
M30	Fim de programa

Fonte: Groover, p. 156, 2011.

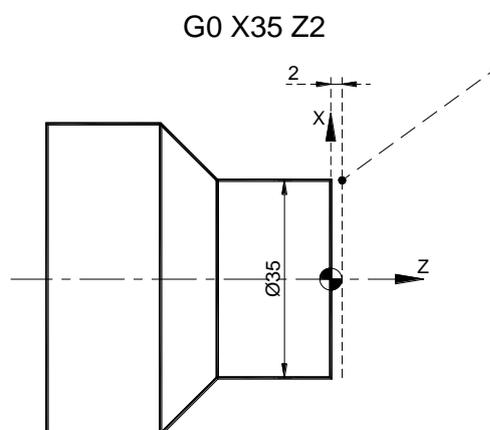
### 2.3.2 Movimentos básicos

Os movimentos básicos são programados pelos códigos G00, G01, G02 e G03.

### 2.3.2.1 Interpolação linear com avanço rápido - G00

Função preparatória usada normalmente para aproximação ou recuo das ferramentas, essa função comanda o movimento com taxa de avanço mais rápida da máquina. É usada para reduzir os tempos improdutivo durante o ciclo de usinagem. Junto com essa função, são programados códigos de posicionamento X, Y, Z que indicam o ponto final do deslocamento (Figura 1) (SILVA, 2008).

Exemplo:



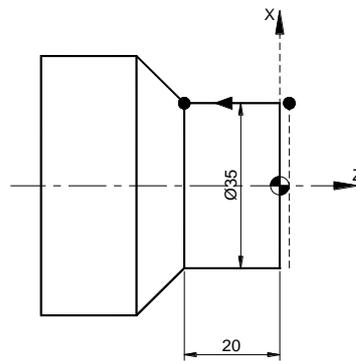
**Figura 1 – Movimento rápido G00**  
(fonte: Silva, 2008)

### 2.3.2.2 Interpolação linear com avanço programado - G01

Realiza movimentos retilíneos com qualquer ângulo, calculados através das coordenadas de posicionamento descritas, utilizando uma velocidade de avanço (F) pré-determinada pelo programador. A ferramenta realizará um movimento em linha reta a partir da posição atual até a posição programada. Junto com essa função, são programados códigos de posicionamento X, Y, Z que indicam o ponto final do deslocamento (Figura 2).

Exemplo:

G0 X35 Z-20 F0.2



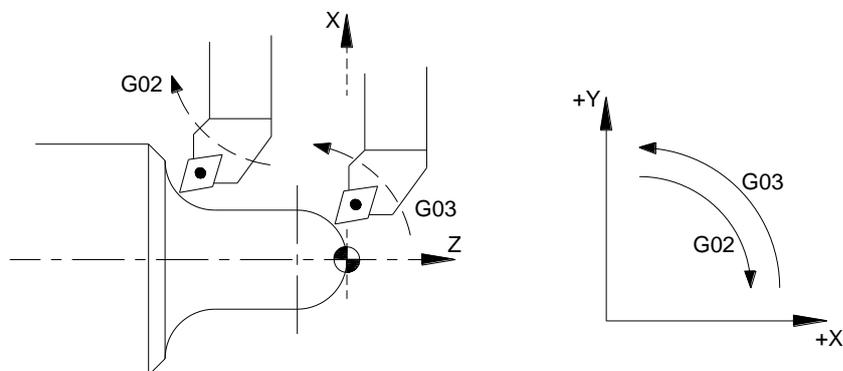
**Figura 2 – Movimento linear G01**  
(fonte: Silva, 2008)

### 2.3.2.3 Interpolação circular no sentido horário e anti-horário - G02/G03

Neste tipo de movimento a ferramenta executa trajetórias na forma de um caminho circular. O código G02 é usado para especificar um movimento circular à direita (sentido horário) e o código G03 à esquerda (sentido anti-horário). A diferença entre sentido horário e anti-horário é estabelecida pela vista frontal do plano. Neste tipo de movimento é necessário definir, além do ponto final do movimento (X, Y, Z) e do avanço, o raio da trajetória dado pela letra (R) ou o centro de arco dados pelas letras (I, J, K) (Figura 3).

Exemplo:

G02/G03 X\_\_ Z\_\_ I\_\_ K\_\_ ou G02/G03 X\_\_ Z\_\_ R\_\_



**Figura 3 – Movimento circular G02 e G03**  
(fonte: Silva, 2008)

## 2.4 Programação CNC

A programação CNC é um termo que se refere aos métodos para geração de instruções que orientam o Comando Numérico Computadorizado (TEICHOLZ, 1985). A programação de CNC pode ser interpretada como um sistema de informação que transforma uma descrição geométrica de uma peça a ser usinada, a partir de uma entrada simbólica da mesma, no controle de posição e velocidade de um ou mais servomotores. Basicamente, o Controle Numérico compreende o comando dos movimentos de aproximação, avanço e recuo de uma ferramenta de corte, de forma similar ao definido na NBR 6162, 1989 (COSTA; PEREIRA, 2006). São várias as técnicas de programação, incluindo uma variedade de procedimentos que vão desde o método manual até o automatizado.

Queiroz e Stemmer (1986) apresentam uma classificação dos métodos de programação CNC, que também é bastante usada por outros autores:

- Programação manual;
- Linguagem de alto nível;
- Sistemas gráfico-interativos;
- Sistemas CAD/CAM.

Nos dois primeiros tipos, o programador elabora o programa, que geralmente é verificado através de sistemas de simulação ou de teste na própria máquina. Nos dois últimos, a elaboração do programa é realizada com o auxílio do computador em situações que apresentam geometria totalmente definida e em que todos os dados necessários são informados. No entanto, a geração final do programa é feita pelo sistema de programação assistida pelo computador. A seguir, apresenta-se algum esclarecimento adicional sobre cada tipo de programação.

### 2.4.1 Programação manual

Segundo Gibbs (1994), a programação manual de máquinas CNC, se refere à preparação de um programa sem a utilização de recursos computacionais para determinar a trajetória de ferramenta, os pontos de intersecção de perfil, os avanços, as velocidades etc. Neste processo, o operador informa manualmente os dados da geometria da peça e os comandos de movimento diretamente na MCU (Unidade de Controle de Máquina) que está executando o serviço.

A programação manual é o método mais comum de programação de máquinas comandadas numericamente. Isso ocorre devido aos modernos recursos computacionais, o que têm tornado a programação manual mais fácil e mais rápida, o que gera economia de tempo. A programação executada diretamente no painel da máquina, conta com diversas rotinas como ciclos de usinagem repetitivos para diversas geometrias, funções de repetição do programa ou de parte dele, programação parametrizada que através do recurso matemático facilita o desenvolvimento de caminhos da ferramenta sendo eles complexos ou não.

Costa e Pereira (2003) destacam que muitas empresas que fazem uso da tecnologia CNC realizam a programação manual, o que se justifica por dois motivos:

- i. O investimento num sistema completo que permita a integração digital entre o modelo dos produtos e os recursos de fabricação é extremamente elevado, se levado em conta o valor da aquisição, da manutenção e do treinamento de pessoal;
- ii. A geometria dos produtos usinados é muito simples e, em vários casos, pode ser tratados como programação em 2 ½ eixos<sup>1</sup>.

Na programação manual, todos os cálculos são feitos manualmente com o auxílio de calculadora e sem o auxílio de sistemas computacionais gráficos. O programa quando muito grande geralmente é digitado em um editor de texto no computador e, após, enviado para máquinas via dispositivos de entrada (cabo ou cartão de memória), sendo esse método o mais rápido e confiável. Programas

---

<sup>1</sup> Processo de fresamento pela qual somente os eixos X e Y movimentam-se simultaneamente, ficando o eixo Z somente com o movimento que determina a profundidade da usinagem.

curtos também podem ser inseridos manualmente, por entrada de teclado, diretamente na máquina.

Gonçalves (2007) explica que esse 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.

Para Pereira (2003) existem algumas desvantagens associadas à programação manual. Segundo ele, talvez a maior seja o tempo necessário para desenvolver um programa completo. Os cálculos, as verificações e as outras atividades exigem muito tempo em programações manuais. Outra grande desvantagem é o alto percentual de erros e a dificuldade de efetuar mudanças no programa.

No entanto, ainda segundo o mesmo autor, a programação manual oferece bons resultados, pois é tão intensa que exige o total envolvimento do programador e ainda oferece uma liberdade infinita de desenvolvimento de estruturas de programa. Programar manualmente não oferece somente desvantagens, mas ensina a disciplina e a organização, forçando o programador a entender as técnicas de programação em detalhes. A programação paramétrica, que mais adiante será definida, é também considerada um tipo de programação manual, pois seu desenvolvimento é feito diretamente através do teclado numérico do controlador (máquina) ou da utilização de um editor de textos em um computador.

#### 2.4.2 Linguagem de alto nível

Em função da extensão dos programas elaborados através da programação manual para peças de geometria mais complexa e da dificuldade de sua verificação, foram desenvolvidas linguagens de programação de alto nível para facilitar o trabalho de programação. Destas linguagens, a mais difundida foi a APT (*Automatically Programmed Tools* - Programação Automática de Ferramenta), sendo que muitas outras linguagens foram criadas a partir desta.

Segundo Groover (2011), o APT é um sistema de programação CNC tridimensional desenvolvido no fim dos anos 1950 e início dos anos 1960. De acordo com Pereira (2003), o APT utiliza palavras do idioma inglês, maiúsculas ou minúsculas, com significados específicos. O vocabulário foi desenvolvido para ser aberto, tanto que novas palavras representam novas funções que foram incorporadas à linguagem. Diferentes versões do APT foram escritas para utilização em diferentes computadores, tanto para grandes quanto para pequenas máquinas. O processador do programa APT e a linguagem APT foram concebidos a fim de possibilitar a máxima flexibilidade para o programador CNC. Como qualquer processador em geral, as funções principais do sistema APT são:

- Definições geométricas;
- Definição de ferramentas e de movimentos;
- Funções da máquina-ferramenta;
- Comandos do sistema computacional.

As funções da linguagem são colocadas juntas em uma sequência que descreve a geometria da peça e os movimentos da ferramenta ao longo de uma superfície, ativam várias funções da máquina e definem as exigências para o sistema. O APT também serviu de base para o desenvolvimento de um sistema mais eficiente de programação CNC, o EXAPT, desenvolvido na Alemanha e que suportava a especificação de áreas de usinagem e a utilização de ciclos fixos (Pereira, 2003 *apud* BUDDE, 1973).

O mesmo autor cita que esse tipo de programação requer o uso de um pós-processador, que utiliza uma base de dados de valores numéricos, baseado nas informações do programador, e converte esta base de dados em um código específico para cada tipo de máquina. Atualmente só é utilizado como ferramenta auxiliar na programação, principalmente para máquinas de 4 e 5 eixos.

### 2.4.3 Sistemas gráficos interativos

Os sistemas gráficos interativos são sistemas computacionais destinados à programação CNC que utilizam a interação homem-máquina para determinar as condições desejadas na elaboração de um programa, sendo essa uma forma de programação manual assistida por computador, em que o trabalho de cálculo requerido na programação é automatizado (GONÇALVES, 2007). Diante da grande diversificação de sistemas existentes, pode-se classificar os sistemas gráficos quanto à entrada de dados, conforme sejam com linguagem ou sem linguagem. O primeiro tipo utiliza linguagens simbólicas, com declarações que expressam a escolha da ferramenta, a definição de geometria por elementos de contorno, a determinação dos movimentos da ferramenta e as informações complementares de representação gráfica e armazenamento. O segundo tipo usa recursos como ícones, teclas funcionais e mouse, que facilitam a manipulação dos dados.

Costa e Pereira (2003) apresentam um exemplo de programação assistida, em que o objetivo é desenvolver uma tecnologia de programação de máquina na qual o programador entra com a geometria da peça descrevendo o caminho que a ferramenta deve seguir através de um editor, com auxílio dos ícones gráficos por ele desenvolvidos. Após a edição o programa é depurado, esse trabalho é realizado pelo computador.

### 2.4.4 Sistemas CAD/CAM

Segundo Groover (2011), um sistema CAD/CAM é definido como um sistema gráfico interativo de computador equipado com *software* para realizar algumas tarefas de projeto e manufatura, de modo a integrar essas funções. Sistemas CAD/CAM permitem uma interpretação da geometria da peça, armazenada em um arquivo gráfico criado no CAD, e a geração dos programas CNC de acordo com algumas informações fornecidas pelo programador usuário do módulo CAM.

Para Meneghello (2003) os sistemas CAD/CAM são métodos de programação auxiliada por computador executados por algoritmos computacionais, em que os dados de entrada do sistema são arquivos de transferência de dados de projeto ou percurso da ferramenta. Esse sistema possibilita a interpretação da geometria das peças através da leitura de arquivos gráficos de transferência de dados e a geração de programas CN. No processo de programação, o programador interage com o computador através de um *software*. Este lhe permite fazer a seleção de parâmetros de operação, podendo escolher a melhor estratégia para o caminho da ferramenta, a ferramenta e seus parâmetros de corte e uma tolerância para o processo de usinagem. Essas informações são armazenadas em um arquivo chamado CLDATA (*Cutter Location Data*) que contém os dados de posicionamento da ferramenta e de comandos de operação da máquina-ferramenta. Como última etapa, é executado o pós-processamento em que os dados do arquivo CLDATA são convertidos em códigos que podem ser lidos pelo controlador do CNC de uma máquina-ferramenta específica.

Um objetivo fundamental do sistema CAD/CAM é a integração das funções de engenharia de projetos e engenharia de manufatura (GROOVER, 2011). As vantagens da programação CNC usando sistemas CAD/CAM incluem:

- Possibilidade de simulação *off-line*;
- Análise de tempo e custo de operação de usinagem pelo sistema;
- Seleção de ferramentas mais apropriadas para o processo pode ser feita de forma automática;
- O sistema pode otimizar o caminho de ferramenta, os parâmetros de corte para o material de trabalho e as operações.

Groover (2001) também cita outra vantagem da utilização deste sistema na programação CNC: quando o desenho de uma peça é desenvolvido em um sistema CAD e o CAM tem acesso ao banco de dados deste mesmo sistema, não há a necessidade de redesenhar a geometria da peça durante a programação. Basta recuperar o modelo geométrico da peça e utilizá-lo para definir o caminho da ferramenta, o que resulta em um ganho de tempo e na redução dos custos do processo de usinagem.

#### 2.4.5 Integração CAD/CNC

A integração CAD/CNC é um método de geração automática de programa CN diretamente do sistema CAD, sem o uso de um sistema CAM e de um pós-processador. Neste método, são desenvolvidos aplicativos nos sistemas CAD, em sua linguagem de programação ou em alguma outra linguagem de programação aceita por esse sistema, voltados para geometrias e máquinas CNC específicas.

De acordo com Cunha (1987), a utilização dos sistemas CAD teve seu início na década de 50. A publicação da tese do Dr. Ivan Sutherland chamou a atenção das indústrias automobilísticas e aeroespaciais americanas. Os conceitos de estruturação de dados bem como o núcleo da noção de Computação Gráfica interativa levaram a General Motors a desenvolver o precursor dos primeiros programas de CAD. Logo em seguida, diversas outras grandes corporações americanas seguiram este exemplo, sendo que no final da década de 60 praticamente toda a indústria automobilística e aeroespacial utilizava *softwares* de CAD.

Uma definição do sistema CAD é dada por Besant (1988), que o define como o processo de projeto que utiliza técnicas gráficas computadorizadas, através do emprego de programas (*softwares*) de apoio, auxiliando na resolução dos problemas associados ao projeto. Para Cunha (1987), o conceito de CAD corresponde à execução da atividade de projetar através da criação, manipulação e representação de modelos, utilizando computador e equipamentos especiais. CAD abrange qualquer atividade que usa o computador para desenvolver, analisar ou modificar o projeto de um produto ou processo.

O Projeto Auxiliado por Computador é uma técnica na qual o homem e a máquina se misturam formando um grupo de resolução de problemas, de modo a agrupar as melhores características de cada um (BESANT, 1988). A aplicação dos sistemas CAD pelas indústrias é bastante ampla. Segundo Groover e Zimmers (1984), hoje esses sistemas são utilizados em diversas áreas da engenharia, como análise de materiais, transferência de calor, simulações gráficas, engenharia de manufatura, engenharia de produção, e ainda nas áreas de geografia de geologia.

Souza e Coelho (2003) explicam que para as indústrias modernas, o CAD representa uma potente e indispensável ferramenta, pois esses sistemas permitem além da geração de desenhos em duas dimensões:

- Modelar produtos tridimensionalmente;
- Aumentar a capacidade do projetista;
- Melhorar a qualidade do projeto;
- Realizar análise de formas geométricas para auxiliar a manufatura;
- Criar banco de dados para manufatura;
- Realizar análise de interferências entre peças de conjuntos montados;
- Definir volumes e centro de massa do produto;
- Comunicação com outros *softwares*, através de interfaces padronizadas.

Os sistemas CAD proporcionam alta eficiência na construção de projetos e, quando associados a métodos de customização (personalização) e automatização, têm seu aproveitamento aumentado consideravelmente, transformando rotinas do dia a dia em formas práticas de utilização. Assim, um usuário ou empresa que desejar realizar uma tarefa específica, pode fazer uso de comandos personalizados e de funções automatizadas.

Segundo McMahon e Browne (1998), a gama de possibilidades de customização de um sistema CAD é muito ampla, uma vez que pode automatizar a execução de tarefas simples a partir de poucos comandos, como, por exemplo:

- Mudar a cor, a espessura ou o tipo de uma *layer*;
- Criar tabelas de conjuntos de peças;
- Desenhar automaticamente componentes de máquina (engrenagem, rolamento, parafuso etc.).

Pode-se também automatizar tarefas complexas nos sistemas CAD, facilitando assim a execução de atividade que levaria muito tempo para ser executada ou que necessitaria do uso de outras ferramentas computacionais. Como exemplo de tarefas mais complexas, pode-se citar:

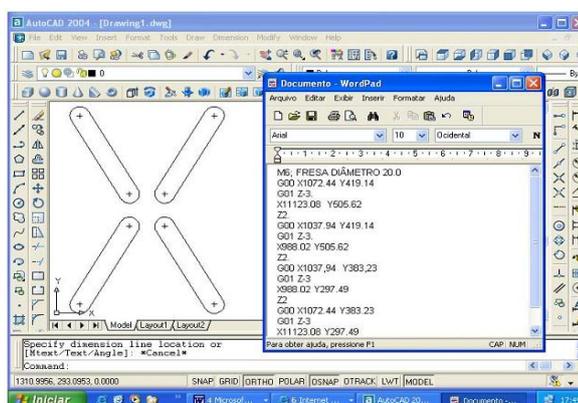
- Análise de projetos de engenharia (simulação de comportamento, resistência, esforço e agressão ao meio ambiente etc.);

- Desenhos de componentes de conjunto com suas identificações e vistas auxiliares;
- Programação CNC de máquinas-ferramentas.

De acordo com McMahon e Browne (1998), os aspectos mais importantes da automatização dos sistemas CAD estão na possibilidade de extração de dados de um modelo para ser usado em algum outro aplicativo ou na modelagem de famílias de peças<sup>2</sup>. A extração de dados normalmente é utilizada para aplicação de produtos, programação e análise de peças.

Um exemplo desse tipo de aplicativo é apresentado por Zindulis et. al. (2005), em que um aplicativo para sistema CAD que gera o programa CN automaticamente é desenvolvido (Figura 4). Muitos desses aplicativos são desenvolvidos nos meios acadêmicos, sendo que poucos chegam às indústrias devido ao fato de serem pouco flexíveis, mesmo que apresentem um custo inferior aos sistemas CAD/CAM comerciais.

Outro exemplo do uso de recurso para automatização dos sistemas CAD é apresentado por Souza (2004) e consiste em uma técnica para desenvolvimento de rotinas avançadas, para uso como sistema CAD/CAM, através de recurso de programação do CAD, no auxílio de geração automática de código CNC. Nesse trabalho, o autor demonstra que através de automatização do CAD pode-se desenvolver uma ferramenta para programação de peças de geometrias simples em máquinas CNC.



**Figura 4 – Geração Automática para Rasgos Radiais para Máquinas CNC (fonte: Zindulis, 2005)**

<sup>2</sup> A expressão “família de peça” é usada para denominar conjunto de peças similares.

Morgan (2008) também apresenta uma ferramenta para extração de características em imagens 2D aplicada à automação industrial, desenvolvida através da automatização do sistema CAD. O método implementado por ele utiliza técnicas de segmentação de imagens em peças modelo, como detecção de bordas, na extração das coordenadas espaciais da imagem e importação em aplicativos CAD para a geração de programas de Comando Numérico Computadorizado (CNC). Neste método, a imagem de algum objeto é copiada e tratada por diversas técnicas de processamento de imagem. Após, um arquivo de texto contendo as coordenadas da imagem (peça) é gerado e importado no CAD. No ambiente CAD, é executado o cálculo da trajetória da ferramenta que identifica a geometria da peça e define o caminho da ferramenta na geração do programa CNC para reprodução da mesma.

## **2.5 Ciclos de usinagem em Comandos Numéricos**

Ciclos de usinagem (ou ciclos fixos) são funções especiais desenvolvidas para facilitar a programação e, principalmente, diminuir o tamanho dos programas, executando em uma única sentença operações de desbastes de perfis complexos, furações com quebra cavaco, abertura de cavidade e roscamento dos mais variados (SILVA, 2008). Os ciclos de usinagem foram desenvolvidos para executar tarefas que exigiam uma grande quantidade de códigos para a sua realização. O Ciclo de usinagem é uma função especializada que está contida no programa do controle da máquina e é utilizada para executar operações específicas e que não podem ser alterados pelo usuário (SMID, 2000; MATTSON, 2009).

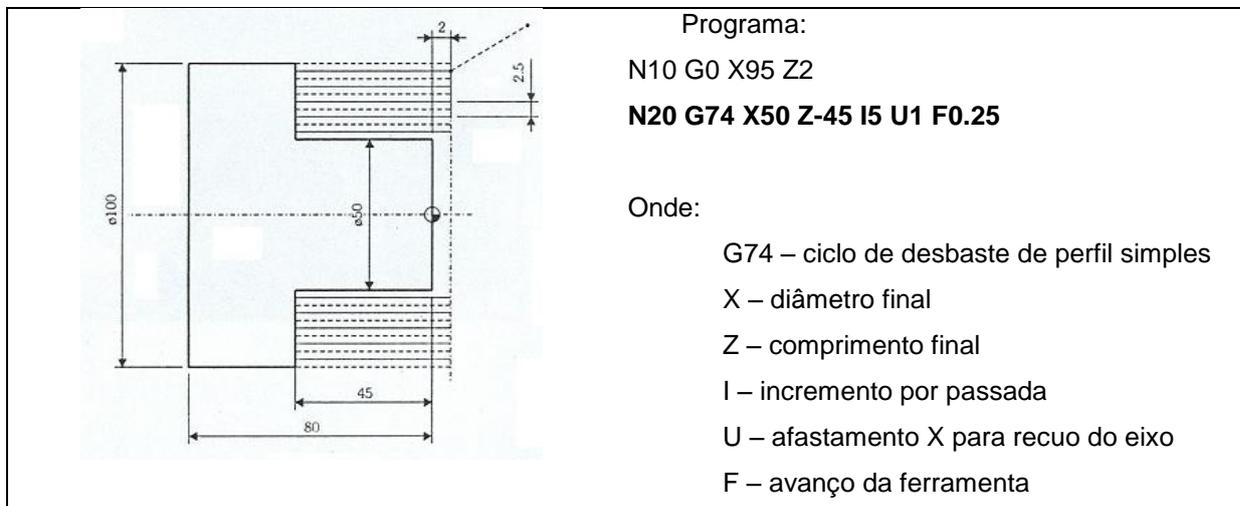
De acordo com Silva (2008), os fabricantes dos Comandos Numéricos desenvolvem os seus próprios ciclos de usinagem, que têm muito em comum, sendo que para o programador essa é uma questão de adaptação. A grande vantagem dos ciclos de usinagem é a redução do tempo de elaboração de um programa. Ao invés de escrever vários blocos de programa para realizar uma determinada operação, basta ao programador, num único bloco, programar a função correspondente ao ciclo e indicar os valores das variáveis envolvidas na operação. A função que define o ciclo, bem como todas as variáveis a ela associadas, são modais. Essa

característica, associada à utilização de sub-rotinas, reduz ainda mais o tempo gasto em programação.

### 2.5.1 Ciclos com funções do comando

São ciclos desenvolvidos pelos fabricantes dos Comandos Numéricos. Eles não podem ser alterados e estão disponíveis para todos os programas, sendo que podem ser chamados a qualquer momento dentro do programa CNC.

A figura 5 apresenta um exemplo de ciclo do fabricante, para o processo de torneamento de um perfil simples em linguagem ISO de um comando modelo FANUC Oi. Neste processo toda a usinagem de desbaste é especificada em somente uma sentença, definida pela função G74 e suas variáveis.



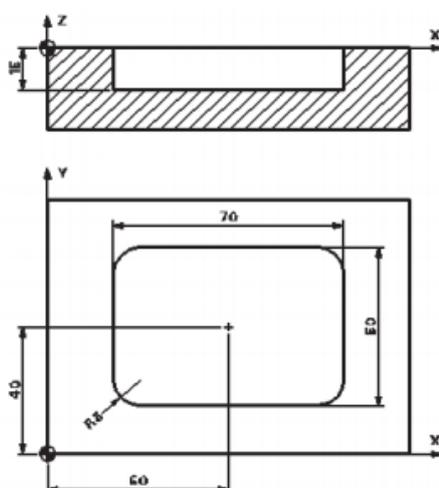
**Figura 5 – Ciclo de usinagem para torneamento simples  
 (fonte: Silva, 2008)**

Se esse mesmo programa CNC fosse escrito sem o uso do ciclo de usinagem, de acordo com a trajetória descrita na figura 5, seriam necessários 42 blocos, conforme apresenta a tabela 4.

Outro exemplo de ciclo do fabricante é apresentado na figura 6, que mostra o programa CNC para usinagem de um cavidade, operação essa típica e comumente realizada no processo de fresamento. Esse tipo de usinagem consiste na retirada de material de uma área delimitada definida por sua parede e fundo, podendo ter a forma quadrada, retangular, circular ou indefinida. A figura 6 representa a utilização de um ciclo de usinagem para fresamento de cavidade retangular em um comando Siemens modelo Sinumerik 810. Neste processo a usinagem de desbaste e acabamento é especificada em somente uma sentença definida pela função POCKET1.

Tabela 4 – Programa CNC para usinagem do perfil da figura 5

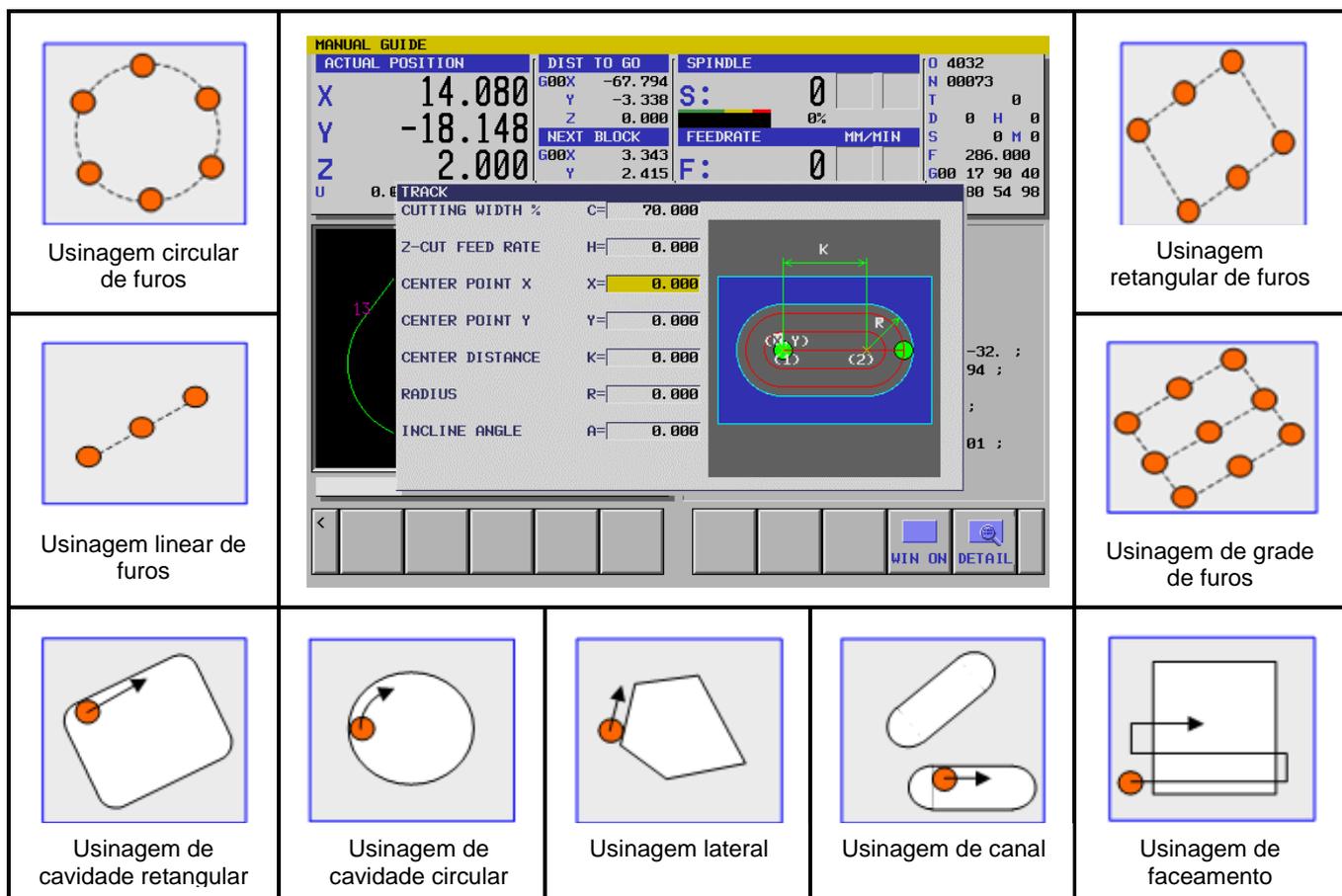
N10 G0 X95 Z2	N150 G1 Z-45	N290 X70
N20 G1 Z-45 F0.25	N160 X85	N300 G0 Z2
N30 X100	N170 G0 Z2	N310 X60
N40 G0 Z2	N180 X75	N330 G1 Z-45
N50 X90	N190 G1 Z-45	N340 X65
N60 G1 Z-45	N200 X80	N350 G0 Z2
N80 X95	N210 G0 Z2	N360 X55
N90 G0 Z2	N230 X70	N370 G1 Z-45
N100 X85	N240 G1 Z-45	N380 X60
N110 G1 Z-45	N250 X75	N390 G0 Z2
N120 X90	N260 G0 Z2	N400 X50
N130 G0 Z2	N270 X65	N410 G1 Z-45
N140 X80	N280 G1 Z-45	N420 X100



G0 X0 Y0 Z10  
 POCKET1 (5, 0, 2, -15, , 70, 50, 8, 40,  
 80, 500, 3, 2, 1, 0, 300, 2000)

Figura 6 – Ciclo de usinagem para fresamento de alojamento  
 (fonte: manual de programação Siemens Sinumerik 810)

Os fabricantes de Comandos Numéricos tornam cada vez mais fáceis a programação dos ciclos fixos diretamente no painel da máquina, pois desenvolvem *menus* interativos que facilitam a entrada de dados e o entendimento do processo. A figura 7 mostra um exemplo de um *menu* interativo do comando FANUC Oi.



**Figura 7 – Ciclos de usinagem para centro de usinagem  
(fonte: adaptação manual comando FANUC – centro de usinagem)**

## 2.5.2 Ciclos parametrizados

Os Comandos numéricos apresentam recursos que possibilitam ao programador desenvolver ciclo de usinagem. Um recurso utilizado para isso é a

programação parametrizada também chamada de macro (DJASSEMI, 2004). Essa técnica torna-se necessária quando os ciclos disponíveis no comando não atendem a uma aplicação específica. Os ciclos desenvolvidos ficam armazenados na memória da máquina e podem ser chamados a qualquer momento pelo operador ou alterados conforme a necessidade.

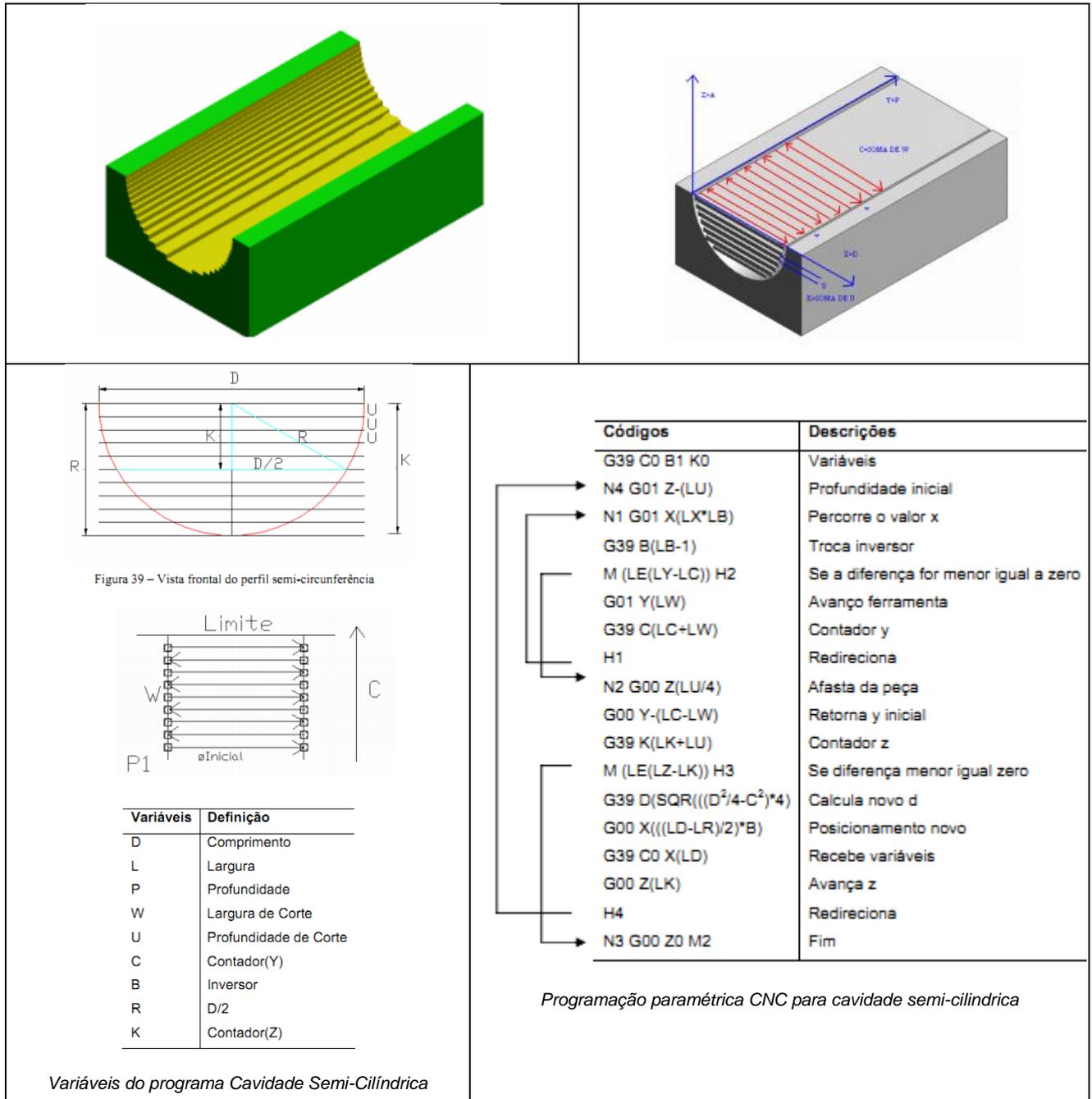


Figura 8 – Ciclo fixo do programador (fonte: Gonçalves, 2007)

A programação parametrizada é uma forma de programação manual avançada que pode ser comparada a qualquer linguagem de programação como PASCAL, BASIC ou C. Porém, ela reside direto no controle do CNC e pode ser acessada ao nível de código G, podendo ser combinada com programação manual. Características relacionadas aos computadores como variáveis, aritmética, declarações de lógica, *loopings* e chamada de sub-rotinas estão disponíveis nesta linguagem (MUNDO CNC, 2011).

Gonçalves (2007) apresenta um estudo sobre a implantação de ciclos de usinagem através de programação parametrizada em que foram desenvolvidos novos ciclos de usinagem para algumas geometrias, aumentando assim a potencialidade de uma máquina CNC. Um dos ciclos por ele desenvolvido destina-se à usinagem no processo de Cavidade Semi-Cilíndrica, apresentado na figura 8. Segundo Souza (2004), apesar do potencial desse dispositivo de programação aumentar a versatilidade do equipamento, seu uso não é amplamente explorado. Empresas de médio e grande porte optam pela aplicação de sistemas de integração CAD/CAM, que apesar de exigir maior investimento são mais eficientes.

### **3 METODOLOGIA**

De modo geral, a programação CNC é gerada e simulada em um ambiente virtual. Com o uso de ferramentas CAD/CAM consegue-se uma maior eficiência no processo produtivo, evitando-se erros de programação e reduzindo o tempo de máquina parada.

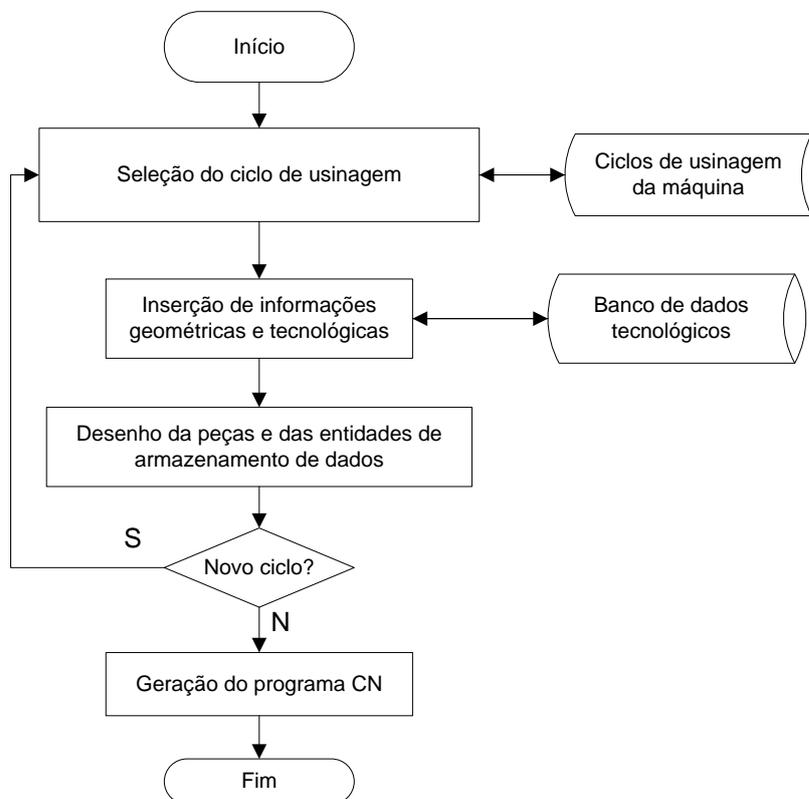
O presente trabalho apresenta uma metodologia de integração CAD/CAM através da aplicação de ciclos usinagem em fresamento e furação de geometrias específicas. O objetivo é mostrar a sua viabilidade através da automatização de um sistema CAD, explorando seus recursos gráficos e de programação para uma aplicação específica. Para a implementação deste trabalho, foi desenvolvido um aplicativo CAD utilizando o *software* AutoCAD® e a linguagem de programação AutoLISP, devido ao fato de esses sistemas trabalharem integrados, permitindo, assim, atender ao objetivo proposto.

Neste capítulo é apresentada a metodologia utilizada no desenvolvimento desta proposta em um ambiente CAD para programação de ciclos de usinagem para fresamento e furação. A linguagem de programação CNC demonstrada neste trabalho é a utilizada no comando MACH9 aplicado em centro de usinagem.

#### **3.1 Estrutura geral da metodologia proposta**

Para o desenvolvimento desta metodologia inicialmente definiu-se um fluxo de informações que o sistema deve seguir o qual é representado na figura 9. Neste sistema, é selecionado no ambiente CAD um dos ciclos de usinagem. Após, é solicitada a entrada de dados referentes à geometria do material bruto e à geometria do ciclo de usinagem selecionado, como furos, cavidade ou rebaixo ao redor de uma área específica. Esses dados fazem com que o sistema crie na área gráfica do CAD um modelo sólido do perfil final da peça. Este serve somente para visualização de como ficará a peça após sua usinagem, já que os dados para criação do programa

CNC são calculados a partir das informações inseridas para criação do desenho e das informações tecnológicas que também são inseridas pelo programador.



**Figura 9 – Fluxograma da estrutura básica do sistema**

### 3.2 Desenvolvimento de aplicativos no AutoCAD

De modo geral, aplicativos são programas de computadores que atuam sobre um sistema operacional, desenvolvidos com o objetivo de facilitar as atividades do programador. Já os aplicativos específicos são programas desenvolvidos para satisfazer uma necessidade não encontrada nos *softwares* existentes. Aplicativos do CAD são programas que complementam um *software* de CAD de uso genérico, adequando-o a uma área específica (DE LUCCA, 1999). Como exemplo, pode-se citar um aplicativo para desenhar componentes de máquinas (rolamentos, parafusos, engrenagens) no AutoCAD. Através dos aplicativos CAD é possível aumentar a produtividade do o processo de desenvolvimento de produtos, porém

poucos conhecem a sua aplicação e o modo como ela é feita. No AutoCAD, o desenvolvimento de aplicativos pode ser feito através da linguagem de programação AutoLISP.

De acordo com Kramer (1995), a linguagem de programação AutoLISP é uma linguagem proveniente do LISP, abreviatura para *LISt Processing*. O LISP é a segunda mais antiga linguagem de programação de alto nível, ainda usada nos computadores modernos. Este tem LISP tem sido usado por cientistas da computação gráfica no campo da inteligência artificial e da implantação de programas de automação. Ainda segundo o mesmo autor, o AutoLISP roda somente dentro do AutoCAD, por isso pode ser considerado o sistema operacional do AutoLISP.

O AutoLISP permite manipular listas, definir novas funções, trabalhar com entrada e saída de dados, efetuar operações matemáticas, expressões condicionais, *looping* e processamento externo de arquivos, acessar e modificar o banco de dados de desenho do AutoCAD, executar qualquer comando do CAD, chamar quadro de diálogo etc. (SOUZA, 2004).

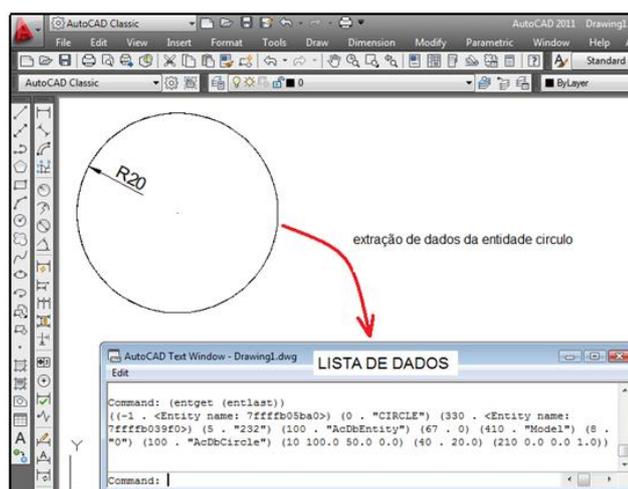
Segundo Kramer (1995), a linguagem LISP trabalha com processamento de lista e, por esse motivo, ela se encaixa quase naturalmente dentro de um sistema CAD, uma vez que esse sistema é baseado em pontos e vetores. Pontos podem ser realmente considerados listas, já que dois ou três números reais são combinados para formar um ponto de referência (X, Y, Z) a fim de definir sua posição no espaço. Então ela é uma lista contendo duas listas que descrevem os valores de coordenadas do ponto (SOUZA, 2004).

### 3.2.1 Banco de dados das entidades do AutoCAD

O AutoCAD possui uma estrutura de banco de dados interna que contém informações sobre objetos e entidades (objetos criados com representação gráfica), necessárias à manipulação das características geométricas e à construção de sua representação gráfica. Um desenho do AutoCAD é formado por um conjunto de

objetos armazenados neste banco de dados. O conhecimento dessa estrutura é fundamental para a programação do AutoCAD (KRAMER, 1995).

O AutoLISP possui funções que permitem acessar, consultar e editar entidades do banco de dados gráficos do AutoCAD. Quando o desenho é carregado na memória, as informações sobre cada entidade individual são armazenadas em um determinado local na memória. Cada entidade possui um nome que é formado por um conjunto de números hexadecimais, os quais apontam para o local onde esta está armazenada. Esse conjunto de números é chamado de *Entity Name* (nome da entidade). Segundo Kramer (1995), um dos recursos mais poderosos do AutoLISP é o de acessar informações específicas dentro da base de dados de desenho do AutoCAD (Figura 10).



**Figura 10 – Extração dos dados de uma entidade no AutoCAD**

Segundo Ruschel (2000), no AutoLISP a lista de dados de uma entidade é obtida a partir de seu nome utilizando uma função chamada ENTGET. Essa lista de dados é uma lista de associações, sendo que algumas destas associações são representadas por listas (ou sublistas) denominadas par pontuado. Uma lista par pontuado é formada por dois átomos (código de grupo. valor) separados por um ponto. Esse ponto não conta como átomo da lista. Cada código de grupo representa um tipo de valor, por exemplo:

- -1 está associado ao nome da entidade.

- 0 está associado ao tipo da entidade "LINE", "ARC"...
- 8 está associado ao nome da *layer* onde a entidade foi desenhada.

Como exemplo, são apresentados os principais elementos em uma lista de dados de uma linha desenhada no AutoCAD a partir das coordenadas (2,3) até (8,5):

**((-1 . <Entity name: xxx.....xxxx>) (0 . "LINE") (8 . "0") (10 2.0 3.0 0.0) (11 8.0 5.0 0.0))**

Onde:

- O primeiro elemento da sublista é um número inteiro que indica o tipo de informação. Este número é considerado como CÓDIGO DE GRUPO.
- O segundo elemento da sublista é o próprio dado.

A tabela 5 apresenta uma lista de dados de uma entidade tipo linha, seus principais códigos e o seu significado.

Tabela 5 – Listas de dados da entidade linha

Código de grupo	Descrição
-1	Nome da entidade
0	Tipo de entidade "linha"
6	Tipo de linha
8	Nome do layer
10	Ponto inicial (X, Y, Z)
11	Ponto final (X, Y, Z)
38	Elevação
39	Altura
62	Cor da linha
210	Direção da extrusão

A tabela 6 apresenta uma lista de dados de uma entidade tipo círculo, seus principais códigos e o seu significado.

Tabela 6 – Listas de dados da entidade círculo

Código de grupo	Descrição
-1	Nome da entidade
0	Tipo de entidade "círculo"
6	Tipo de linha
8	Nome do layer
10	Centro do círculo
38	Elevação
39	Altura
40	Raio do círculo
62	Cor do círculo
210	Direção da extrusão

A tabela 7 apresenta uma lista de dados de uma entidade tipo arco, seus principais códigos e o seu significado.

Tabela 7 – Listas de dados da entidade arco

Código de grupo	Descrição
-1	Nome da entidade
0	Tipo de entidade "arco"
6	Tipo de linha
8	Nome do layer
10	Centro do arco
38	Elevação da entidade
39	Altura da entidade
40	Raio do arco
50	Ângulo inicial do arco
51	Ângulo final do arco
62	Cor do círculo
210	Direção da extrusão

A tabela 8 apresenta uma lista de dados de uma entidade tipo texto, seus principais códigos e o seu significado.

Tabela 8 – Listas de dados da entidade texto

Código de grupo	Descrição
-1	Nome da entidade
0	Tipo de entidade "texto"
1	Texto
6	Nome do tipo de linha
7	Estilo de fonte do texto
8	Nome do layer
10	Ponto de inserção
11	Ponto de alinhamento
38	Elevação da entidade
39	Altura da entidade
40	Altura do texto
50	Ângulo de rotação
62	Cor do texto
210	Direção da extrusão

### 3.3 Ciclos de usinagem do sistema

Segundo Silva (2008), ciclos de usinagem são funções especiais desenvolvidas para facilitar a programação e, principalmente, diminuir o tamanho dos programas, executando em uma única sentença operações de desbastes de perfis complexos, furações com quebra cavaco, abertura de cavidade e roscamentos dos mais variados.

O desenvolvimento deste trabalho visa demonstrar uma metodologia CAD/CAM em um ambiente CAD. Para isso, foram selecionados quatro ciclos de usinagem existentes no comando MACH9:

- 1 - Ciclo de usinagem de círculo de furos;
- 2 - Ciclo de usinagem quadrado ou retangular de furos;
- 3 - Ciclo de usinagem para fresamento de cavidades retangulares, quadrados ou circulares;
- 4 - Ciclo de usinagem para fresamento de resalto ao redor de uma área especificada.

### 3.3.1 Ciclo de usinagem de círculo de furos

Esta função é ativada pelo código G24 e é aplicada para a execução automática dos ciclos fixos de furação e rosqueamento (G79-G86), podendo também ser aplicada à utilização de subprogramas ou sub-rotinas. Este ciclo de usinagem desenvolve uma sequência de furos em um padrão circular de pontos igualmente espaçados. A figura 11 apresenta um modelo de aplicação.

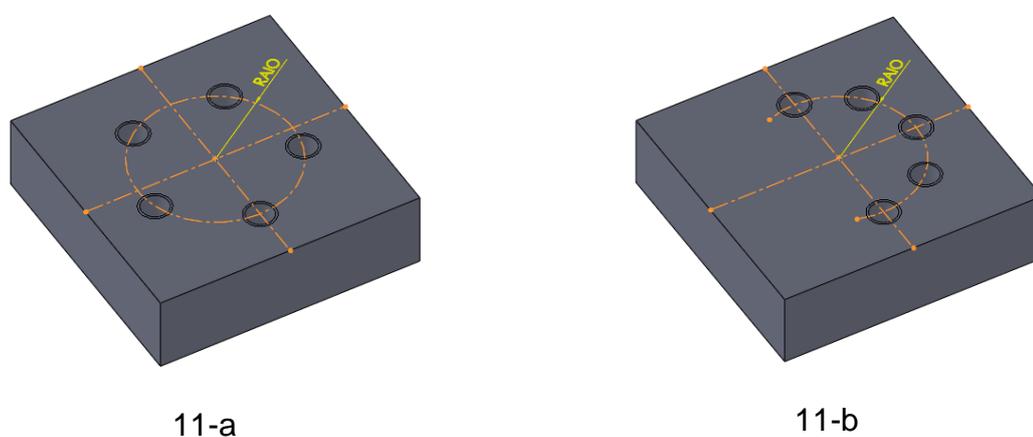


Figura 11 – Ciclo de usinagem de círculo de furos.

O ciclo G24 é definido pela seguinte sentença:

**G24 R\_\_ C\_\_ I\_\_ J\_\_ L\_\_ B\_\_**

Onde:

R – define o raio do círculo;

C – define o ângulo de posição inicial do círculo;

I – é a coordenada do centro do círculo no eixo X, em absoluto;

J – é a coordenada do centro do círculo no eixo Y, em absoluto;

L – é o número total de pontos em que os autociclos, subprogramas, ou sub-rotinas serão executados;

B – define o espaçamento angular igual para os pontos no círculo.

A figura 12 apresenta um exemplo de aplicação deste ciclo, em que sua programação segue a seguinte sequência:

- a- Aproximação da ferramenta da peça;
- b- Programação do ciclo fixo de furação;
- c- Programação do ciclo de repetição do ciclo fixo;
- d- Cancelamento do ciclo fixo de furação.

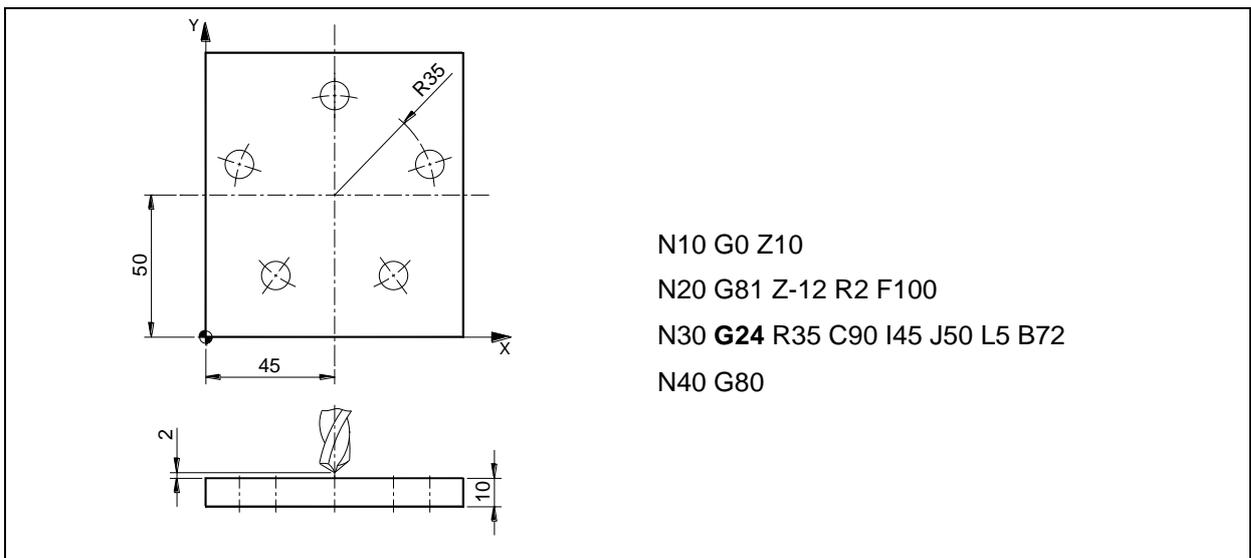
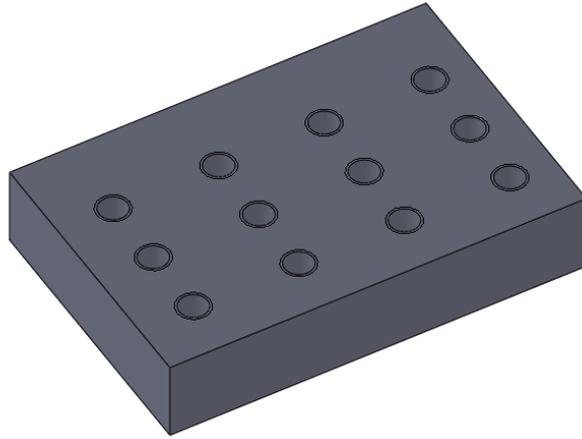


Figura 12 – Aplicação da função G24

### 3.3.2 Ciclo de usinagem quadrado ou retangular de furos

Esta função é ativada pelo código G25 e é aplicada para a execução automática dos ciclos fixos de furação e rosqueamento (G79-G86), podendo também ser aplicada à utilização de subprogramas ou sub-rotinas. Este ciclo de usinagem desenvolve uma sequência de furos em um padrão quadrado ou retangular de pontos igualmente espaçados. A figura 13 mostra um modelo de aplicação.



**Figura 13 – Ciclo de usinagem quadrado ou retangular de furos.**

O ciclo G25 é definido pela seguinte sentença:

**G25 X\_\_ Y\_\_ I\_\_ J\_\_**

Onde:

- X – especifica a distância incremental em X e a direção entre os pontos a serem executados;
- Y – especifica a distância incremental em Y e a direção entre os pontos a serem executados;
- I – define o número de pontos ao longo do eixo X;
- J – define o número de pontos ao longo do eixo Y.

A figura 14 apresenta um exemplo de aplicação deste ciclo, em que sua programação segue a seguinte sequência:

- a- Aproximação da ferramenta da peça;
- b- Programação do ciclo fixo de furação;
- c- Programação da auto-rotina para repetição do ciclo fixo;
- d- Cancelamento do ciclo fixo de furação.

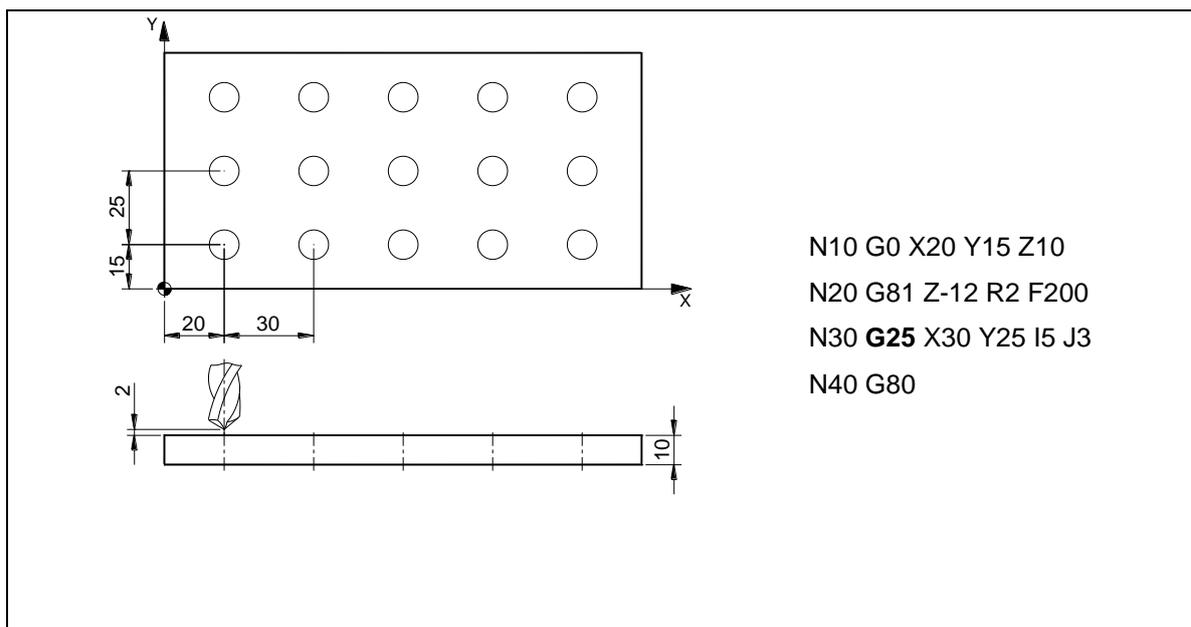


Figura 14 – Aplicação da função G25

### 3.3.3 Ciclo de usinagem para fresamento de cavidade retangular, quadrado ou circular

Esta função é ativada pelo código G26 e é aplicada para a execução automática da operação de desbaste e/ou acabamento em cavidades. No caso de cavidades retangulares e quadrados, também é possível especificar um raio de canto. A figura 15 apresenta modelos de aplicação.

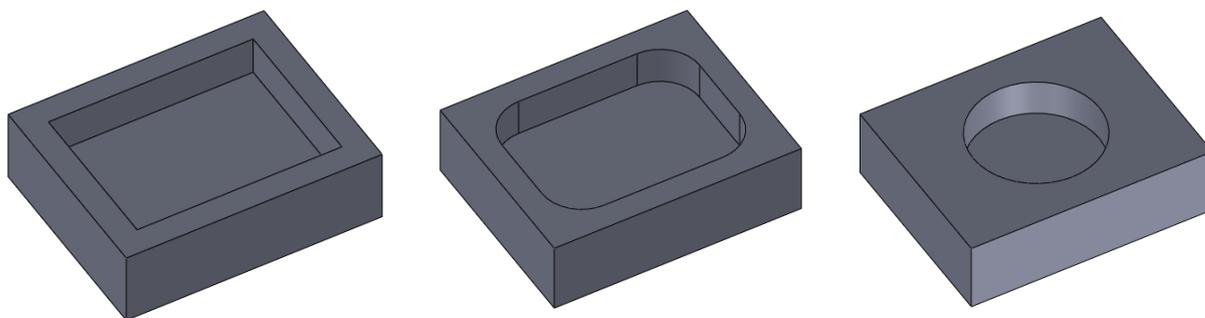


Figura 15 – Ciclo de fresamento de cavidade retangular ou circular.

O ciclo G26 é definido pela seguinte sentença:

**G26 X\_\_ Y\_\_ Z\_\_ I\_\_ J\_\_ K\_\_ U\_\_ W\_\_ F\_\_ H\_\_ V\_\_ R\_\_**

Onde:

X – coordenada do canto oposto da cavidade no eixo X no sistema absoluto em relação ao zero da peça;

Y – coordenada do canto oposto da cavidade no eixo Y no sistema absoluto em relação ao zero da peça;

Z - nível Z do fundo da cavidade (profundidade final);

I – sobremetal para acabamento ao logo do eixo X;

J – sobremetal para acabamento ao logo do eixo Y;

K – sobremetal para acabamento no fundo da cavidade;

U – profundidade de corte (desbaste) para o eixo Z;

W – determina a largura de corte ao longo dos eixos X e Y;

F – define o avanço de desbaste;

H – define o avanço de acabamento;

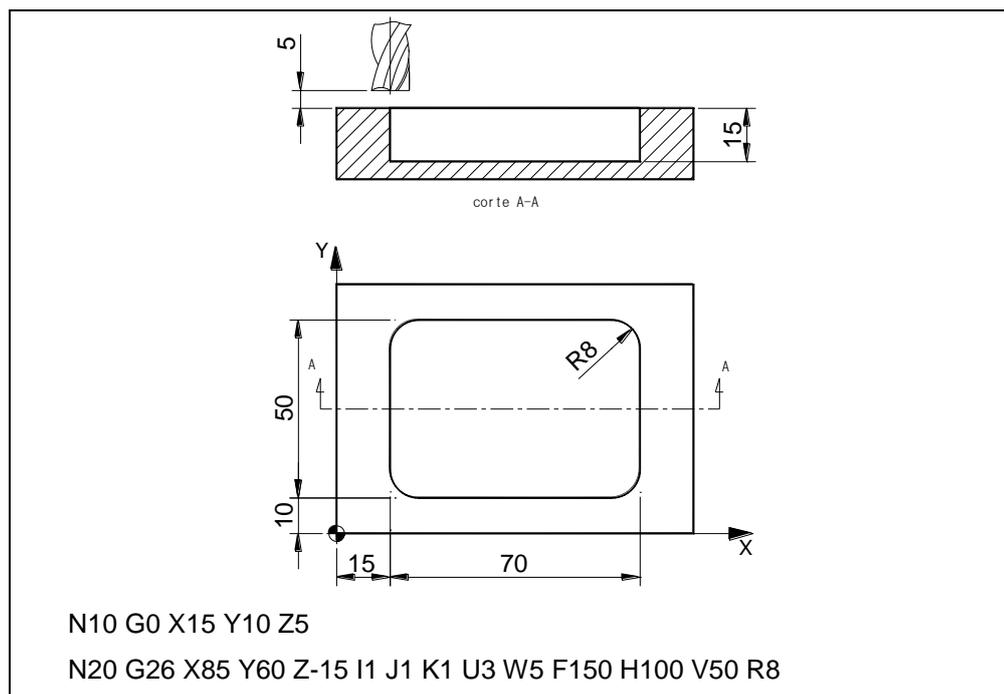
V – define o avanço de penetração para o eixo Z;

R – determina o raio de canto da cavidade. Se X e Y não forem programados, R será o raio da cavidade circular.

A figura 16 apresenta um exemplo de aplicação deste ciclo, em que sua programação segue a seguinte sequência:

Bloco N10 - Aproximação da ferramenta da peça;

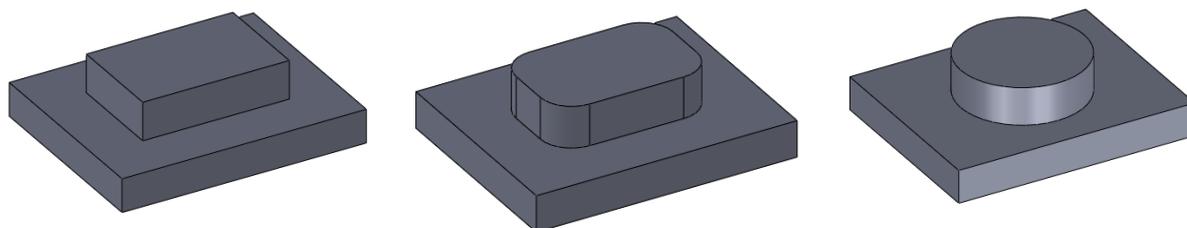
Bloco N20 - Programação do ciclo para fresamento da cavidade.



**Figura 16 – Aplicação da função G26**

### 3.3.4 Ciclo de usinagem para fresamento de resalto ao redor de uma área especificada

Esta função é ativada pelo código G27 e é aplicada para a execução automática da operação de desbaste e/ou acabamento em resaltos ao redor de uma área. No caso de resaltos retangulares e quadrados, também é possível especificar um raio de canto. A figura 17 apresenta modelos de aplicação.



**Figura 17 – Ciclo de fresamento de resalto ao redor de uma área especificada**

O ciclo G27 é definido pela seguinte sentença:

**G27 X\_\_ Y\_\_ Z\_\_ I\_\_ J\_\_ K\_\_ U\_\_ W\_\_ F\_\_ H\_\_ V\_\_ R\_\_**

Onde:

- X – coordenada do canto oposto do resalto no eixo X no sistema absoluto em relação ao zero da peça;
- Y – coordenada do canto oposto do resalto no eixo Y no sistema absoluto em relação ao zero da peça;
- Z – nível Z do fundo do resalto (profundidade final);
- I – sobremetal para acabamento ao logo do eixo X;
- J – sobremetal para acabamento ao logo do eixo Y;
- K – sobremetal para acabamento no fundo do resalto;
- U – profundidade de corte (desbaste) para o eixo Z;
- W – determina a largura de corte ao longo dos eixos X e Y;
- F – define o avanço de desbaste;
- H – define o avanço de acabamento;
- V – define o avanço de penetração para o eixo Z;
- R – determina o raio de canto do resalto. Se X e Y não forem programados, R será o raio do resalto circular.

A figura 18 apresenta um exemplo de aplicação deste ciclo, em que sua programação segue a seguinte sequência:

Bloco N10 - Aproximação da ferramenta da peça;

Bloco N20 - Programação da auto-rotina para fresamento de resalto.

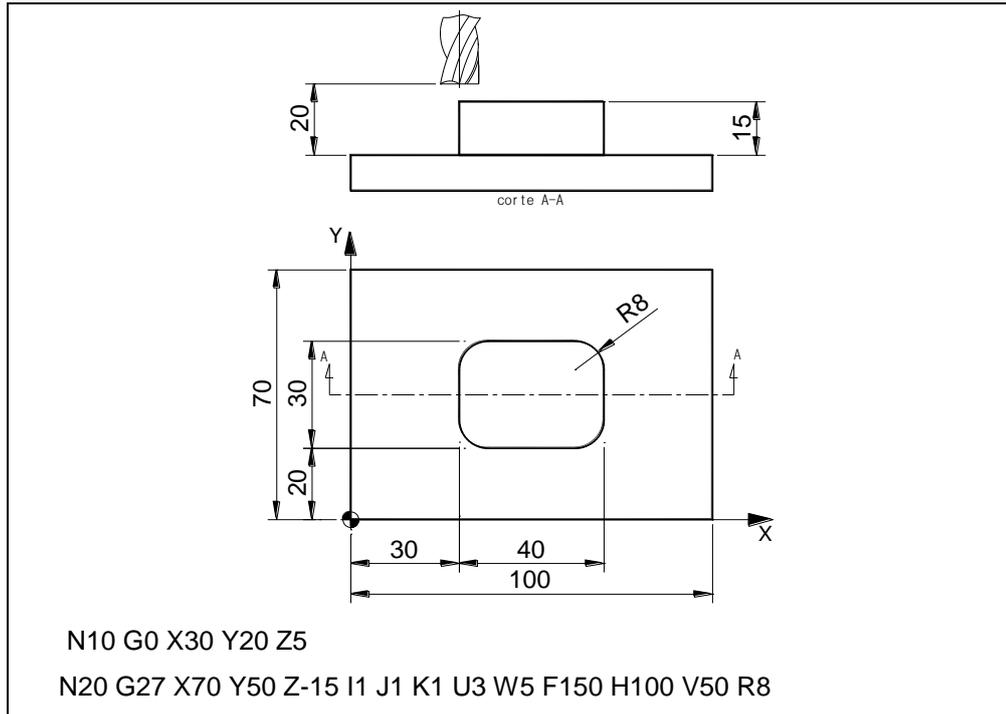


Figura 18 – Aplicação da função G26

### 3.4 Codificação dos Ciclos de Usinagem

Para codificar os ciclos no CAD, é utilizado o recurso *layer*, que, em português, significa camada. É um recurso existente em todos os *softwares* profissionais de CAD que permite organizar as informações por categorias, além de disponibilizar o gerenciamento visual dos dados de um arquivo. Esses permitem aplicar características diferentes a cada entidade do desenho, como, por exemplo: nome, cor, espessura da linha, tipo de linha etc.

Dentro do sistema CAD, pode-se criar a quantidade de *layers* necessárias e dar-lhes nomes de acordo com a necessidade. Elas podem ser criadas tanto manualmente como através de funções específicas, ou ainda de modo automático através da programação do CAD. A grande vantagem da utilização desse recurso é que ao abrir um arquivo elas são inicializadas automaticamente, pois permanecem vinculadas a esses. Outra vantagem das *layers* é que elas podem ser ocultadas, chaveadas e congeladas, evitando assim qualquer alteração na propriedade de uma entidade de modo intencional ou acidental.

Neste trabalho as *layers* têm as seguintes aplicações:

- Especificação do ciclo de usinagem;
- Especificação da sequência de usinagem;
- Especificação das geometrias.

A especificação da geometria identifica se no arquivo existem as entidades peça bruta e peça, sendo que cada uma dessas é atribuída a uma *layer*. A especificação do ciclo de usinagem e da sequência de usinagem é dada através do nome da *layer*. Para isso, um sistema de codificação do nome da *layer* foi desenvolvido com o objetivo de ordenar a sequência de criação dos ciclos de usinagem e identificar os ciclos criados. Essa codificação tem seu formato apresentado na figura 19:



**Figura 19 – Forma de codificação das *layers***

Onde:

- 1 - Número da sequência de criação dos ciclos de usinagem;
- 2 - Espaçador (hífen);
- 3 - Código G da função;
- 4 e 5 - Nome da função do ciclo de usinagem;
- 6 - Espaçador (hífen);
- 7 - Número da sequência do ciclo.

Exemplo:

**1-G25-1**

O primeiro elemento do código significa que esse ciclo de usinagem é o primeiro a ser criado; o terceiro, quarto e quinto elementos do código significam que o nome da função do ciclo de usinagem é G25, e o sétimo elemento do código

significa que é a primeira função G25 programada. A cada novo ciclo de usinagem criado, um novo código de identificação é gerado.

### 3.5 Desenho da peça

O desenho da peça bruta<sup>3</sup> é um modelo sólido tridimensional utilizado por todos os *softwares* de geração de programa CNC (CAM) para representar o material bruto que originará o produto final. Alguns *softwares* também utilizam o modelo sólido para visualizar a retirada de material durante o processo de simulação. A peça bruta pode ter uma geometria simples (ex.: caixa, cilindro ou um perfil do contorno ao qual é dado uma profundidade), podendo ser criada diretamente no *software* CAM, ou uma geometria complexa (ex.: formato final de peça com várias curvas), criada através da seleção de um modelo sólido tridimensional já desenhado.

Na área gráfica do CAD, é apresentado o desenho de um modelo sólido da geometria final da peça após a usinagem. Inicialmente, o sistema realiza o desenho da peça bruta. Para isso, define-se ao sistema a sua forma, podendo ser um cubo ou um cilindro, e suas dimensões, conforme apresentado na figura 20. O ponto de origem é fixado pelo sistema e está localizado no ponto de origem do sistema de coordenada do CAD.

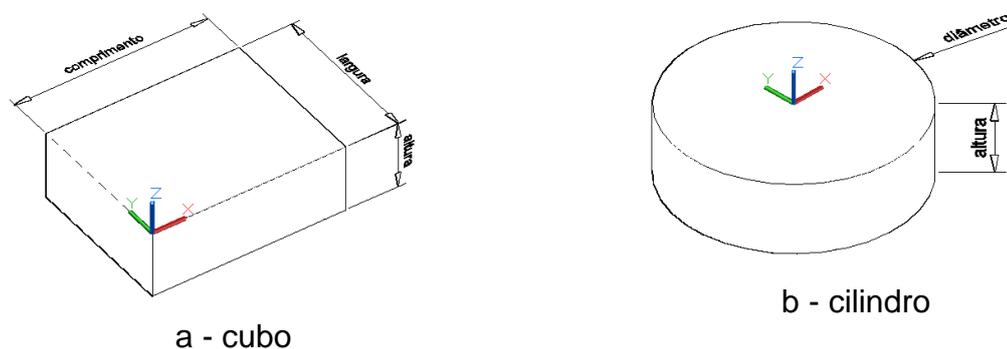
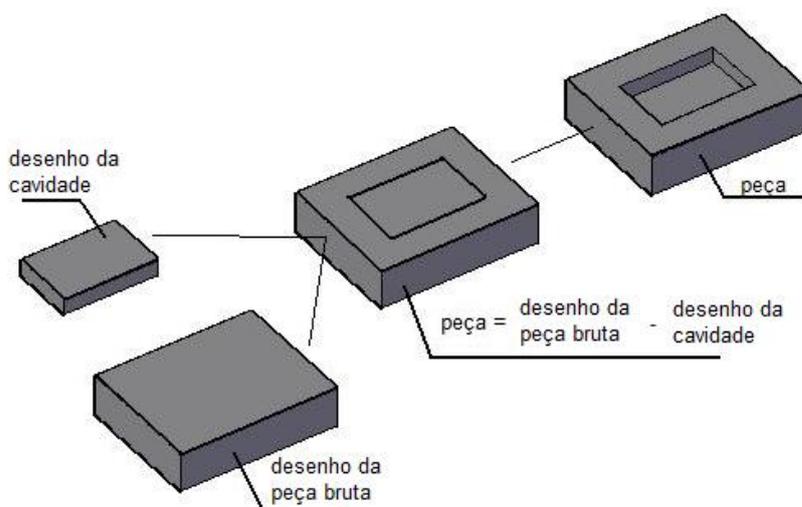


Figura 20 – Modelos de peça bruta

<sup>3</sup> Peça Bruta é a definição dada ao estado em que se encontra uma peça antes de sofrer algum processo de manufatura.

Após, é realizado sobre o desenho da peça bruta o desenho da geometria do ciclo de usinagem solicitado pelo programador. Utilizando o comando de extração entre volumes do CAD, é retirado o desenho da peça bruta e o desenho da geometria do ciclo de usinagem, ficando como resultado o peça acabada. A figura 21 apresenta um exemplo deste processo para o ciclo de usinagem G26.



**Figura 21 – Processo de extração entre volumes sólidos do CAD**

### 3.6 Parâmetros geométricos

Para armazenar os dados geométricos da peça e os dados tecnológicos do processo, são usadas entidades de desenho do CAD, que são linhas, arcos, círculos e texto. Essas armazenam em seu banco de dados informações que podem ser lidas pelo sistema e transformadas em dados do programa CN. Para desenhar essas entidades, são utilizadas informações fornecidas durante a criação da peça.

Cada tipo de ciclo de usinagem tem um código de identificação conforme visto no item 3.4. As entidades de cada ciclo são desenhadas utilizando o mesmo código, tornando possível identificá-las durante a geração do programa CN. A seguir, será apresentado o desenho das entidades de cada um dos ciclos de usinagem.

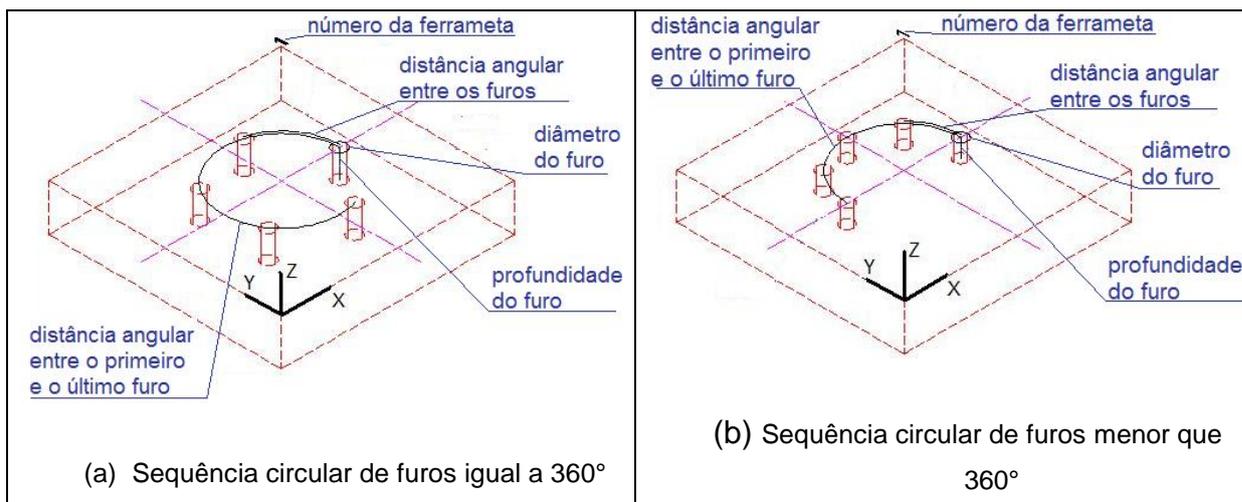
### 3.6.1 Ciclo de furação circular G24

A figura 22 apresenta as entidades que são desenhadas para armazenar os dados de construção do ciclo de furação circular G24. Durante a programação deste ciclo, o sistema solicita que seja inserido o número da ferramenta, que é a sua posição no magazine da máquina. Esse número é gravado junto ao desenho da peça, usando o comando para criação de texto do CAD. No banco de dados da entidade texto, esse número está associado ao código de grupo número 1, conforme apresentado na tabela 8.

Para o armazenamento do diâmetro da ferramenta, é utilizada a entidade círculo. O sistema solicita qual o diâmetro da ferramenta a ser utilizada no programa e, após, desenha um círculo nesse diâmetro sobre o primeiro furo a ser criado pelo sistema. No banco de dados da entidade círculo, o valor do raio está associado ao código de grupo número 40, conforme apresentado na tabela 6.

O valor da profundidade do furo é armazenado, utilizando o desenho de uma entidade linha. Quando indicado esse valor, o sistema desenha uma linha que tem como ponto inicial o plano superior do furo e como ponto final esse valor inserido em relação ao plano superior. No banco de dados da entidade linha, a coordenada Z do ponto inicial e final esta associada ao código de grupo número 10 e 11, respectivamente, conforme apresentado na tabela 5.

Para armazenamento da distância angular entre os furos e o número de furos do ciclo de furação circular, é utilizado o desenho de duas entidades arco. A entidade que armazena a distância angular entre os furos tem como ponto inicial a posição do primeiro furo e como ponto final a posição do segundo furo. Para facilitar a sua localização pelo sistema, ela é desenhada na coordenada Z1 em relação à origem do sistema CAD. A entidade que armazena o número de furos tem como ponto inicial a posição do primeiro furo e como ponto final a posição do último furo. No banco de dados da entidade arco, as coordenadas X e Y do ponto inicial e final estão associados ao código de grupo número 50 e 51, respectivamente, conforme apresentado na tabela 7.

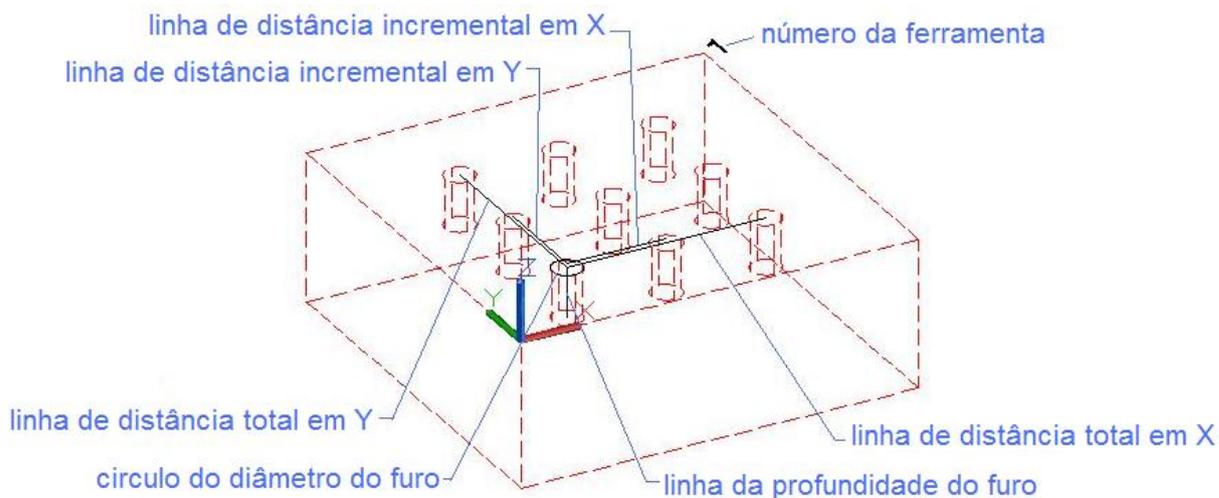


**Figura 22 – Entidades de armazenamento de dados do programa G24**

### 3.6.2 Ciclo de furação linear G25

A figura 23 apresenta as entidades que são desenhadas para armazenar os dados de construção do ciclo de furação quadrada/retangular G25. O armazenamento do número da ferramenta, do diâmetro da ferramenta e da profundidade do furo ocorre da mesma maneira que no ciclo de usinagem G24.

Para armazenamento da distância linear entre os furos e o número de furos do ciclo de furação quadrado ou retangular, é utilizado o desenho de duas entidades linha, tanto na direção X quanto na direção Y. A entidade que armazena a distância linear entre os furos tem como ponto inicial a posição do primeiro furo e como ponto final a posição do segundo furo. Para facilitar a sua localização pelo sistema, ela é desenhada na coordenada Z1 em relação à origem do sistema CAD. A entidade que armazena o número de furos tem como ponto inicial a posição do primeiro furo e como ponto final a posição do último furo. No banco de dados da entidade linha, as coordenadas X e Y do ponto inicial e final estão associados ao código de grupo número 10 e 11, respectivamente, conforme apresentado na tabela 5.



**Figura 23 – Entidades de armazenamento de dados do programa G25**

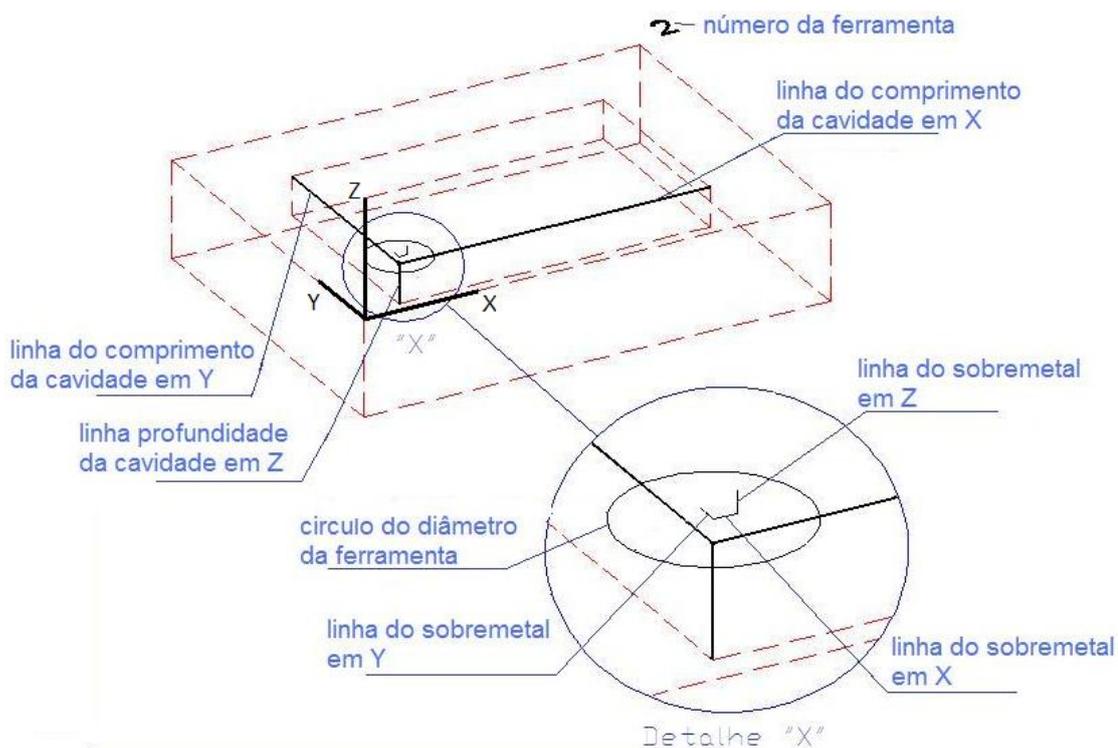
### 3.6.3 Ciclo de usinagem G26 para cavidade retangular

A figura 24 apresenta as entidades que são desenhadas para armazenar os dados de construção do ciclo de usinagem de cavidade retangular G26. O armazenamento do número da ferramenta e do diâmetro da ferramenta ocorre da mesma maneira que no ciclo de usinagem G24.

O valor do comprimento, da largura e da profundidade da cavidade é armazenado utilizando o desenho de uma entidade linha. Para que o sistema realize o desenho da cavidade, insere-se o valor da coordenada X e Y do vértice inicial, as coordenadas X e Y do vértice oposto e o valor da profundidade em Z. Esses mesmos dados são usados pelo sistema para desenhar as linhas de armazenamento de dados, as quais ficaram sobrepostas às quinas do desenho da cavidade. No banco de dados da entidade linha, a coordenada X, Y e Z do ponto inicial e final de cada linha estão associados ao código de grupo número 10 e 11, respectivamente, conforme apresentado na tabela 5.

Para armazenar o valor do sobremetal, também é utilizada a entidade linha. O sistema solicita que seja especificado o valor do sobremetal nas paredes da cavidade (direção X e Y) e no fundo da cavidade (direção Z). O sistema usa esse valor como comprimento da linha. As coordenadas iniciais X e Y das linhas sobremetal em X e sobremetal em Y é a mesma coordenada X e Y do vértice inicial

da cavidade. Já a coordenada inicial Z das mesmas linhas é 1. A linha que define o sobremetal no fundo da cavidade tem sua coordenada inicial X, Y e Z no ponto final da linha sobremetal X. As coordenadas finais de cada uma das linhas são dadas pelo valor do sobremetal transformado em comprimento de linha.

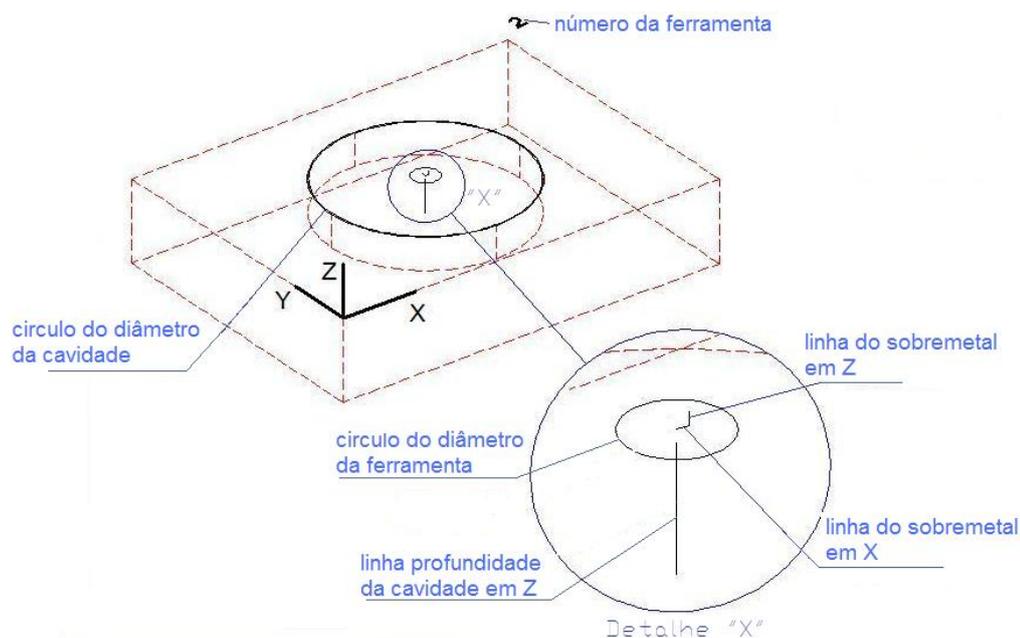


**Figura 24 – Entidades de armazenamento de dados do programa G26 retangular**

### 3.6.4 Ciclo de usinagem G26 para cavidade circular

A figura 25 apresenta as entidades que são desenhadas para armazenar os dados de construção do ciclo de usinagem de cavidade circular G26. O processo de desenvolvimento do desenho destas entidades é praticamente o mesmo aplicado ao desenvolvimento do ciclo G26 retangular. O que muda neste caso é que a entidade que armazena os dados de diâmetro da cavidade e as coordenadas X, Y e Z do centro da mesma é um círculo. A entidade linha que armazena o valor da profundidade da cavidade tem sua coordenada X e Y inicial e final no centro da

mesma. Outra diferença diz respeito às entidades que armazenam o valor do sobremetal. Pelo fato de ser uma cavidade circular, o valor do sobremetal é o mesmo nas direções X e Y. Por esse motivo, somente a linha ao longo de X é desenhada. Para o desenho da linha do sobremetal em Z, o procedimento é o mesmo do ciclo G26 retangular.



**Figura 25 – Entidades de armazenamento de dados do programa G26 circular**

### 3.6.5 Ciclo de usinagem G27 para fresamento de resalto retangular e circular

Os mesmos parâmetros utilizados para o desenho das entidades de armazenamento de dados do ciclo de usinagem G26 retangular e circular são utilizados para o ciclo de usinagem G27. As figuras 26 e 27 representam o posicionamento das entidades junto ao desenho.

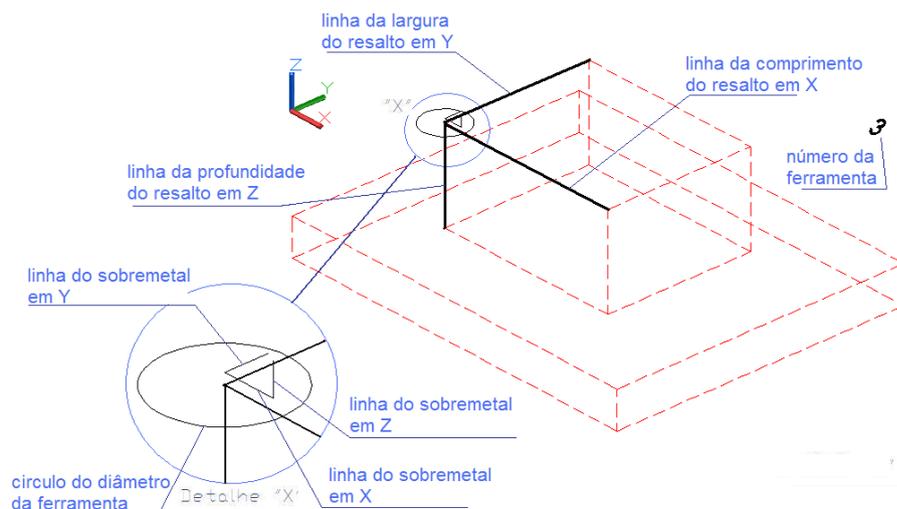


Figura 26 – Entidades de armazenamento de dados do programa G27 retangular

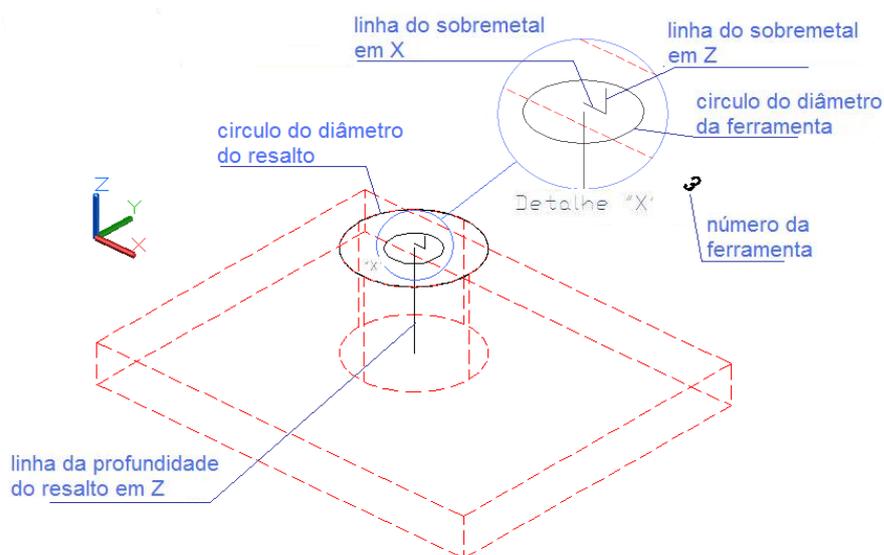


Figura 27 – Entidades de armazenamento de dados do programa G27 circular

### 3.7 Leitura dos parâmetros geométricos

Para extrair os dados do banco de dados da entidade, inicialmente, o sistema procura no arquivo CAD um grupo de entidade que está desenhada com o mesmo "nome *layer*". Para isso, uma lista de dados com característica igual a que se

pretende encontrar é construída através da união de *strings*<sup>4</sup>. Os *strings* juntados são o número do código de grupo e o valor do contido neste código, por exemplo:

$$8 + 1-G24-1 \rightarrow ((8. 1-G24-1))$$

Cada entidade encontrada tem o seu nome gravado em uma variável. Esse nome é o dado contido no código de grupo número -1. Esses nomes são gravados em forma de lista. Após a seleção de todas as entidades que possuem o mesmo nome *layer*, uma nova busca é realizada, mas somente nas entidades que têm seu nome *layer* gravados na variável. Essa busca classifica todas as entidades que são do mesmo tipo. Para isso, uma nova lista de dados com características iguais à procurada é construída através da união dos *strings* 8 e do tipo de entidade, que pode ser linha, arco ou círculo. O nome da entidade encontrada é atribuído a uma variável.

Para caso de desenho que tem mais de uma entidade do mesmo tipo, é necessário fazer outra procura utilizando mais alguma característica conhecida da entidade. Encontrado o banco de dados correto, os seguintes dados são extraídos:

- Linhas – coordenadas X, Y, Z inicial e final, esses valores estão contidos nos códigos de grupo 10 e 11, respectivamente;
- Círculo – coordenada de centro de círculo, (cód. de grupo 10) raio do círculo (cód. de grupo 40);
- Arco – ângulo inicial e final, esses valores então contidos nos códigos de grupo 50 e 51, respectivamente;
- Texto – *string* texto, esse dado está contido no código de grupo 1.

### 3.8 Banco de dados de ferramentas

A grande parte dos *softwares* CAD/CAM comerciais possui um banco de dados de ferramentas integrado, o que auxilia a otimizar o processo de programação. Isso se dá uma vez que esse banco de dados contém os parâmetros

---

<sup>4</sup> A função de unir ou juntar *strings* é conhecida na linguagem da informática como CONCATENAR.

de cortes e geométricos das ferramentas, de modo que, ao ser solicitada uma ferramenta, tem os parâmetros automaticamente carregados no processo. Esses bancos de dados também dão ao programador a possibilidade de editar informações, adicionar ou excluir ferramentas.

Como parte da metodologia, neste estudo é desenvolvido um banco de dados de ferramentas em que o sistema busca esses parâmetros para aplicá-los no programa CN. Para arquivar esses dados, é utilizado um editor de texto que é acessado pelo sistema quando for solicitada uma ferramenta. A tabela 9 apresenta os parâmetros contidos no banco de dados para cada ferramenta, os quais podem ser manipulados através deste subprograma, que também permite a criação ou exclusão de ferramentas.

Tabela 9 – Parâmetros do banco de dados das ferramentas

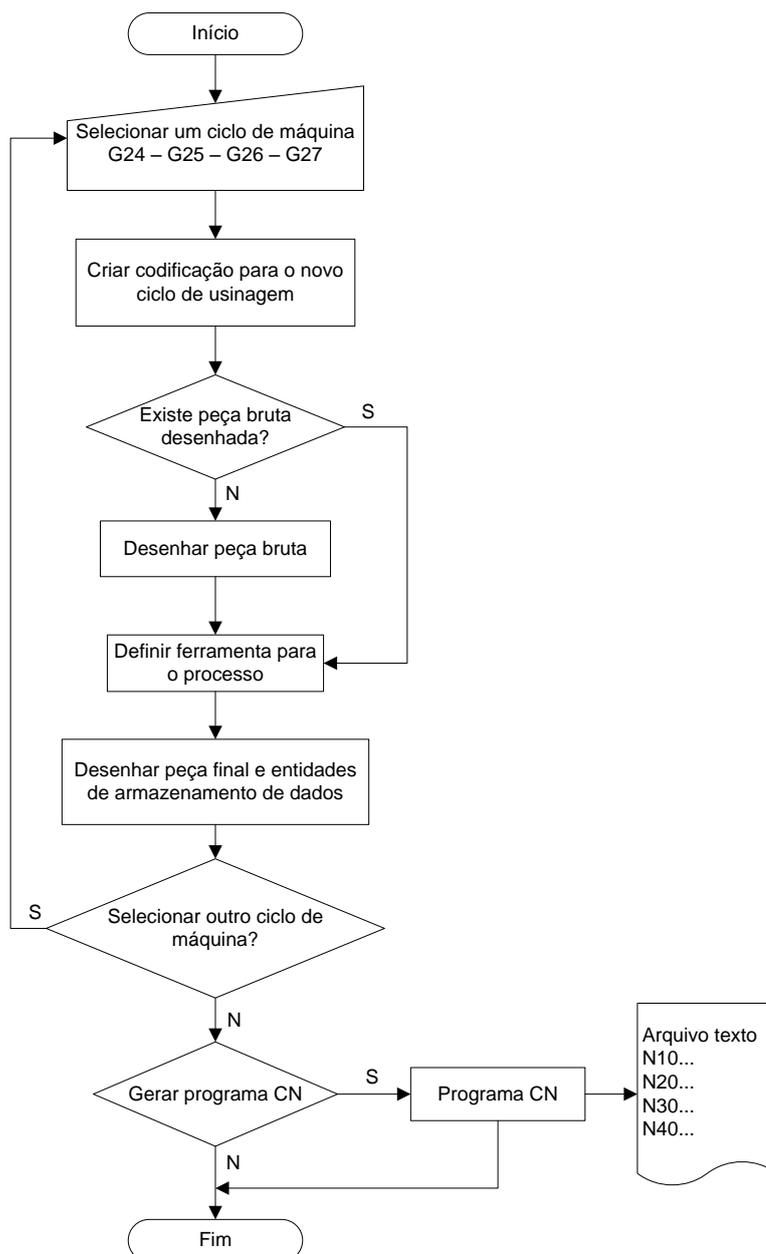
<b>Nome</b>	<b>Diâmetro (mm)</b>	<b>Velocidade de corte (m/min)</b>	<b>Velocidade de avanço (mm/min)</b>	<b>Profundidade de corte (mm)</b>	<b>Largura de corte (mm)</b>
Broca diâmetro 3	3	100	80	0	0
Fresa diâmetro 5	5	130	300	2	3
Fresa diâmetro 8	8	140	420	4	5

## 4 DESENVOLVIMENTO DA METODOLOGIA PROPOSTA

Este capítulo apresenta o desenvolvimento da metodologia proposta. Conforme já visto em capítulos anteriores, existem diversos métodos para geração de programa CN auxiliados por computador de modo automático. A proposta apresentada caracteriza-se por ser um sistema interativo de programação que utiliza o método *off-line* para programação de máquinas CNC utilizando os ciclos de usinagem da máquina. Neste capítulo é descrito com detalhe o desenvolvimento deste aplicativo através do *software* AutoCAD e da linguagem de programação AutoLISP.

Para o desenvolvimento desta metodologia, quatro ciclos de usinagem do comando MACH9 para centro de usinagem foram selecionados, conforme apresentado no capítulo 3.3. Para cada um dos ciclos, um programa em linguagem AutoLISP foi desenvolvido, com o objetivo de demonstrar o funcionamento desta proposta. O fluxograma da figura 28 apresenta uma visão geral do funcionamento deste aplicativo, que será descrito no algoritmo da seção seguinte.

No ambiente CAD, é selecionado um dos ciclos de usinagem da máquina, de modo que uma codificação para esse ciclo é criada, conforme apresentado no item 3.3. A seguir, o sistema verifica se já existe o desenho de uma peça bruta. Se esse desenho não for encontrado, um desenho é criado. Na sequência, são definidos o tipo e os parâmetros de corte da ferramenta a serem utilizados por esse ciclo. Esses dados são arquivados em um arquivo externo. Na próxima etapa, é desenhada a geometria do perfil a ser usinado e as entidades do armazenamento de dados, a partir de dados fornecidos que informam o tipo e as dimensões dessa geometria. Ao final desta etapa, as informações necessárias para geração do programa CN estão completas e o ciclo de usinagem selecionado é finalizado.



**Figura 28 – Estrutura geral do aplicativo**

A etapa seguinte apresenta três alternativas:

- 1º. Criar um novo ciclo de usinagem → Um novo ciclo de usinagem é selecionado, e o processo inicia-se novamente.
- 2º. Gerar o programa CNC → selecionada essa opção, o sistema faz a interpretação das entidades desenhadas para armazenamento de dados, extrai e trabalha esses dados, gerando um arquivo com o programa CN.

3º. Finalizar arquivo → pode-se finalizar o arquivo salvando-o de modo que qualquer uma das duas alternativas anteriores possa ser executada em outro momento.

#### 4.1 Desenvolvimento dos programas auxiliares

Para cada um dos ciclos apresentados neste trabalho, um programa em linguagem AutoLISP foi desenvolvido, com o objetivo de demonstrar o funcionamento desta proposta. Entretanto, alguns programas desenvolvidos para a execução de determinados procedimentos são idênticos nos quatro ciclos:

- Programa para criar a codificação do ciclo de usinagem;
- Programa para criar o desenho da peça bruta;
- Programa para ler banco de dados de ferramenta;
- Programa para achar ferramenta no banco de dados;
- Programa para criar novo programa CN;
- Programa para gerar programa CN;
- Programa para salvar programa CN.

Por esse motivo, optou-se por torná-los programas independentes, o que os caracteriza como SUBPROGRAMAS. As vantagens de aplicar esta técnica de subprograma consistem em:

- Permitir o seu uso quantas vezes forem necessárias durante a execução do programa principal;
- Após a sua execução, pode-se retornar ao programa principal com o resultado por ele produzido;
- Economizar tempo e memória da máquina;
- Facilitar a leitura do programa.

As características fundamentais dos subprogramas são (Figura 29):

- Cada subprograma possui um único ponto de entrada;

- O programa principal tem a sua execução suspensa durante a execução do subprograma;
- O fluxo de controle sempre retorna ao ponto de chamada quando termina a execução do subprograma.

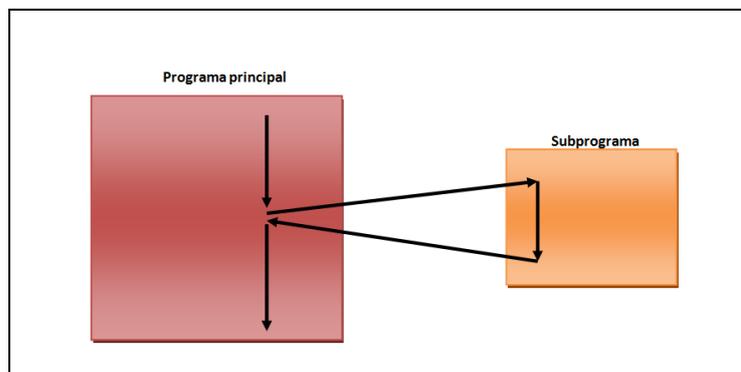


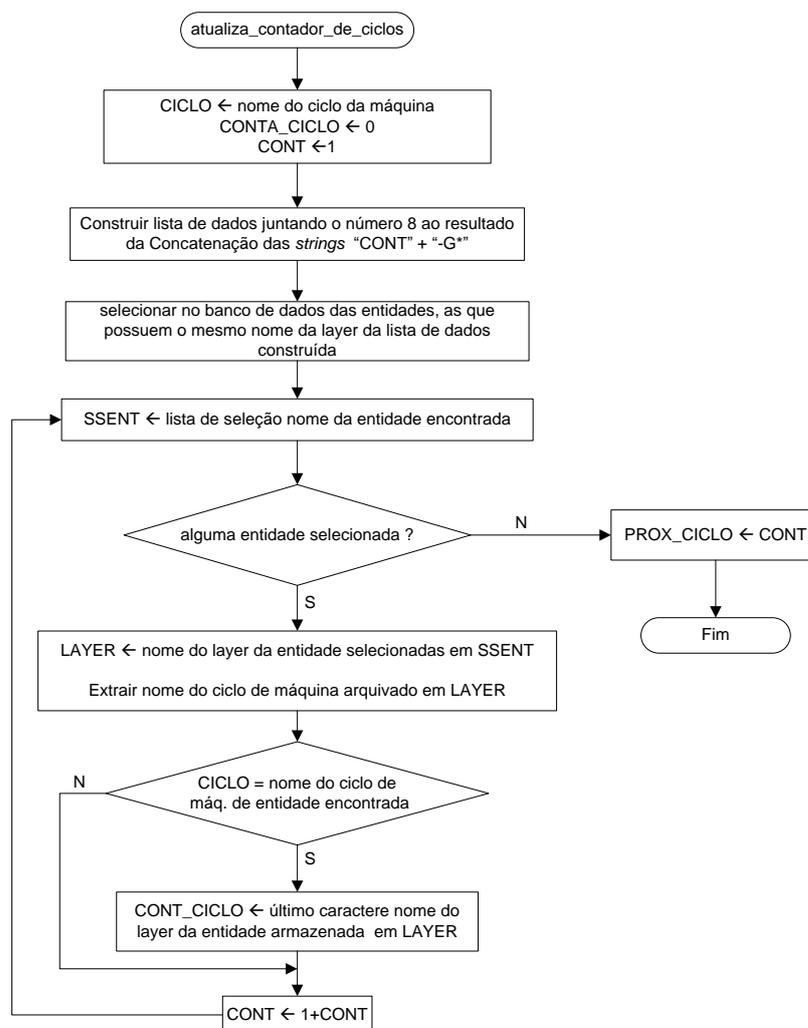
Figura 29 – Fluxo de informações entre programa e subprograma

A seguir são apresentados os algoritmos dos subprogramas.

#### 4.1.2 Subprograma para determinar a codificação do ciclo de usinagem

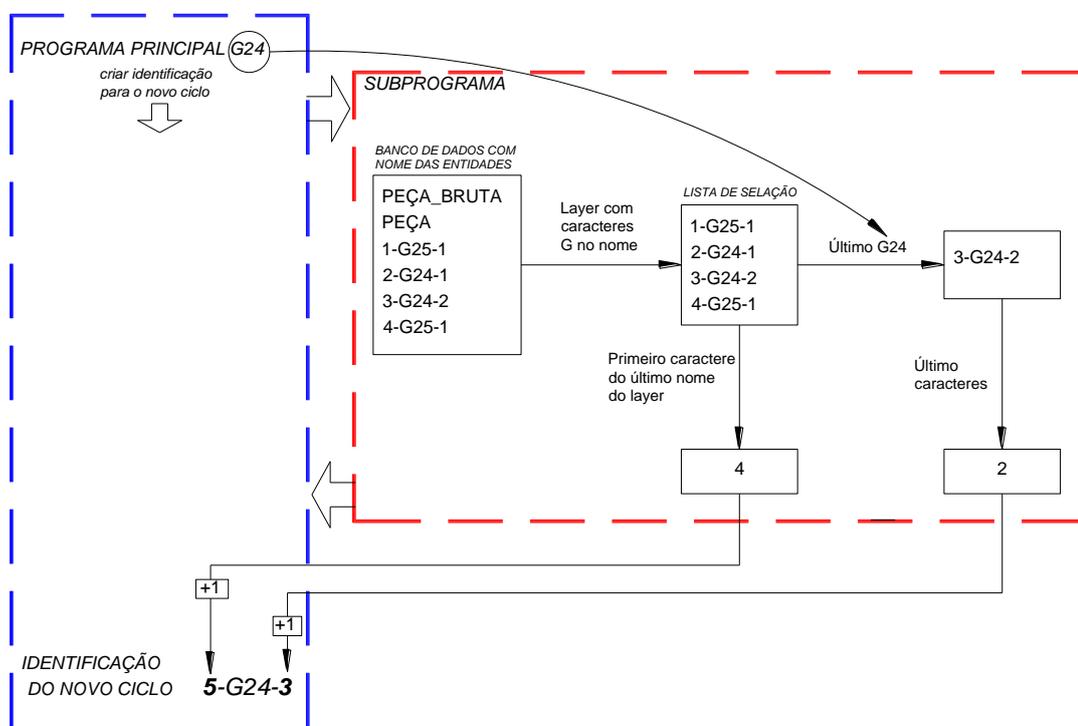
O objetivo deste subprograma é determinar a codificação que será aplicada a um novo ciclo de usinagem. Essa codificação define o nome do ciclo e a sequência de usinagem (conforme apresentado no item 3.4). No fluxograma da figura 30 é apresentado o funcionamento deste subprograma. O programa principal passa como parâmetro para essa rotina o nome do ciclo de usinagem. Uma vez que, para cada ciclo selecionado, o sistema cria uma entidade no CAD com a geometria correspondente, o processamento consiste em realizar uma consulta ao banco de dados das entidades a fim de identificar quais estão em *layer* cujo nome identifica algum ciclo de usinagem. Cada entidade encontrada tem a sua identificação inserida em uma lista de seleção. Ao varrer esta lista, identifica-se no banco de dados da entidade correspondente se há coincidência entre o nome do ciclo de usinagem codificado no nome do *layer* e o ciclo usado como parâmetro na chamada do subprograma. Nos casos em que essa condição é satisfeita, o processamento

armazena o último caractere do nome do *layer*, que é uma variável tipo *string*. Conforme descrito no item 3.4, este caractere corresponde ao número sequencial do respectivo ciclo. Como o algoritmo analisa todos os ciclos na ordem em que foram inseridos, a rotina retorna um valor que corresponde ao número de vezes que o respectivo ciclo já foi utilizado. Na chamada dessa rotina, esse número é incrementado de uma unidade, assim como a variável que identifica a sequência geral de usinagem. Finalmente, um novo nome de *layer* é criado, contendo a sequência geral de usinagem, o nome e o número sequencial desse ciclo no processo. O desenho da figura 31 apresenta um exemplo do funcionamento deste subprograma.



**Figura 30 – Fluxograma do subprograma “Atualiza contador de ciclo”**

Neste exemplo, o programa principal “G24” chama o subprograma que faz a identificação da codificação. A rotina então classifica dentro do banco de dados todas as entidades que contêm a letra G em seu nome do *layer*, pois essa letra identifica os ciclos de usinagem. Para criar a identificação do novo ciclo de usinagem G24, o sistema verifica qual o primeiro e o último caractere do último ciclo G24 criado. Esses valores identificados são levados para o programa principal, os quais são incrementados de mais uma unidade e, juntados ao nome do ciclo, criam a nova identificação.



**Figura 31 – Funcionamento do subprograma “Atualiza Contador de Ciclo”**

#### 4.1.3 Subprograma para desenhar a peça bruta

O objetivo deste subprograma é desenhar um modelo sólido tridimensional que representa a peça bruta. Esse desenho pode ter a geometria retangular ou cilíndrica, conforme mostrado na figura 20. Neste desenho, é formada a geometria final da peça.

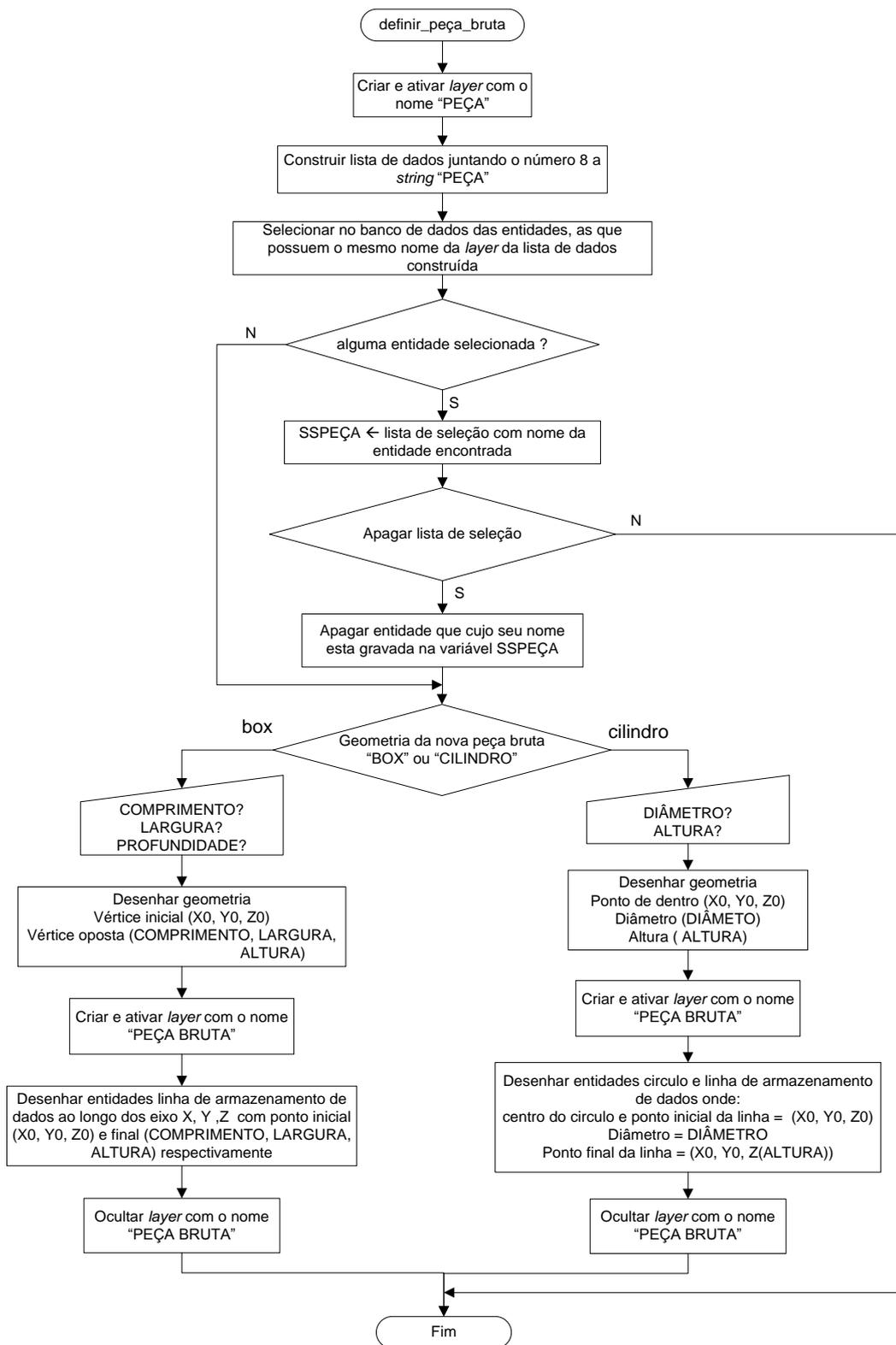
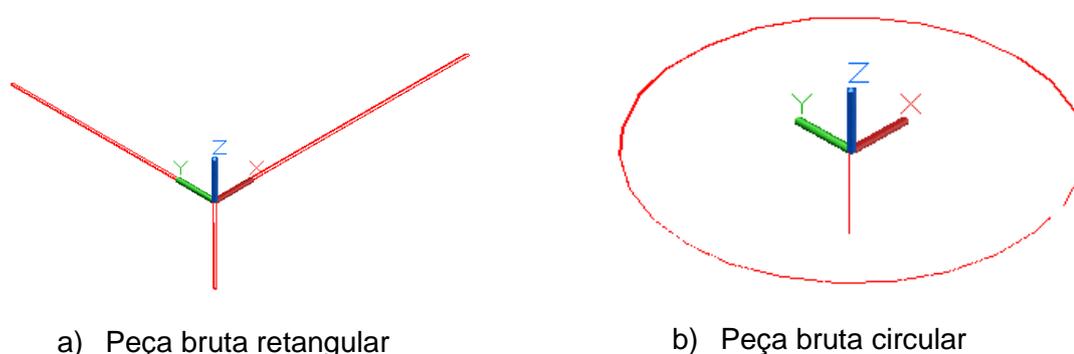


Figura 32 – Fluxograma do subprograma “Definição peça bruta”

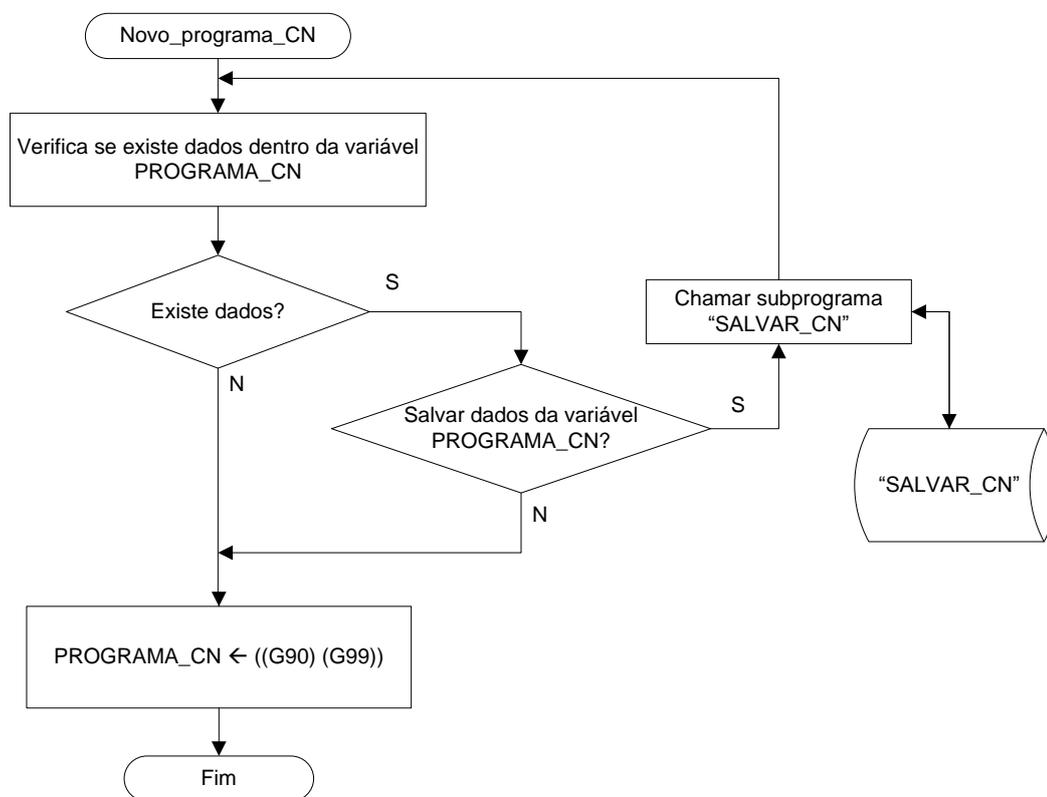
O fluxograma da figura 33 apresenta o funcionamento deste subprograma. Quando o programa principal não identifica o desenho de uma entidade que define uma peça no sistema CAD, é chamado o subprograma que define a peça bruta. Para isso, é necessária a inserção do tipo de geometria e de suas dimensões. Com esses dados e sobre uma *layer* específica, a rotina constrói o modelo sólido da peça bruta. Aproveitando os dados inseridos, e sobre outra *layer* específica, são construídas as entidades que armazenam os dados da geometria da peça bruta, conforme apresenta a figura 32.



**Figura 33 – Entidades de armazenamento de dados da peça bruta**

#### 4.1.4 Subprograma para atribuir blocos iniciais do programa CN a uma variável

O objetivo deste subprograma é atribuir à variável uma lista de dados com os blocos de inicialização do programa CN. O fluxograma da figura 34 apresenta o funcionamento desta rotina. O subprograma que define a peça bruta, ao detectar que está sendo criado um novo arquivo, chama esta rotina para atribuir a uma variável os blocos de inicialização do programa CN. Antes de atribuir esses dados, o sistema realiza uma consulta a esta variável para identificar se há algo a ela atribuído. No caso de algo ser encontrado, esses dados podem ser extraídos para um arquivo texto. Para isso, a rotina chama um subprograma que realiza essa tarefa, de modo que a variável é esvaziada e são inseridos em seu lugar blocos com os dados de inicialização do programa CN. Caso os dados encontrados não sejam extraídos, os novos dados são juntados aos dados já existentes na variável.



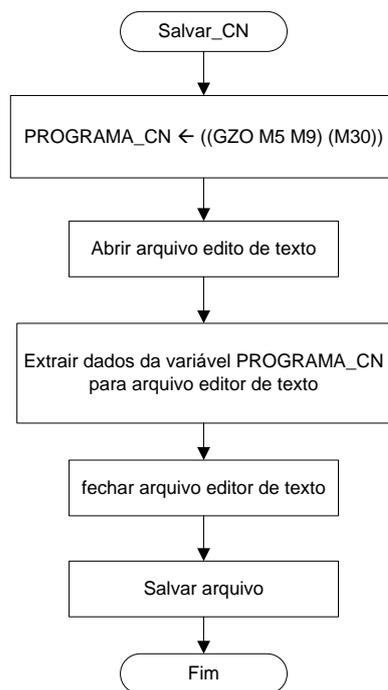
**Figura 34 – Fluxograma do subprograma “Novo programa CN”**

#### 4.1.5 Subprograma salvar programa CN

Este subprograma tem os seguintes objetivos:

- Atribuir a uma variável uma lista dados, contendo os blocos finais do programa CN;
- Extrair os dados da variável PROGRAMA\_CN para um arquivo editor de texto;
- Salvar esse arquivo.

A figura 35 apresenta o fluxograma do funcionamento deste subprograma.



**Figura 35 – Fluxograma do subprograma “Salvar CN”**

#### 4.1.6 Subprograma para ler o banco de dados das ferramentas

O objetivo deste subprograma é inserir em uma variável os dados das ferramentas contidos em um arquivo de dados. É desta variável que o sistema irá extrair as informações geométricas e tecnológicas das ferramentas a serem utilizadas pelos ciclos de usinagem e para completar o programa CN. O fluxograma da figura 36 apresenta o funcionamento desta rotina. São definidas inicialmente as variáveis que receberão as informações do arquivo de dados. Os conteúdos que possam existir nelas são excluídos para evitar que sejam confundidos com os novos que serão atribuídos. Dentro do arquivo de dados, as ferramentas então organizadas na seguinte ordem: 1º) nome da ferramenta, 2º) diâmetro, 3º) velocidade de corte, 4º) profundidade de corte e 5º) largura de corte.

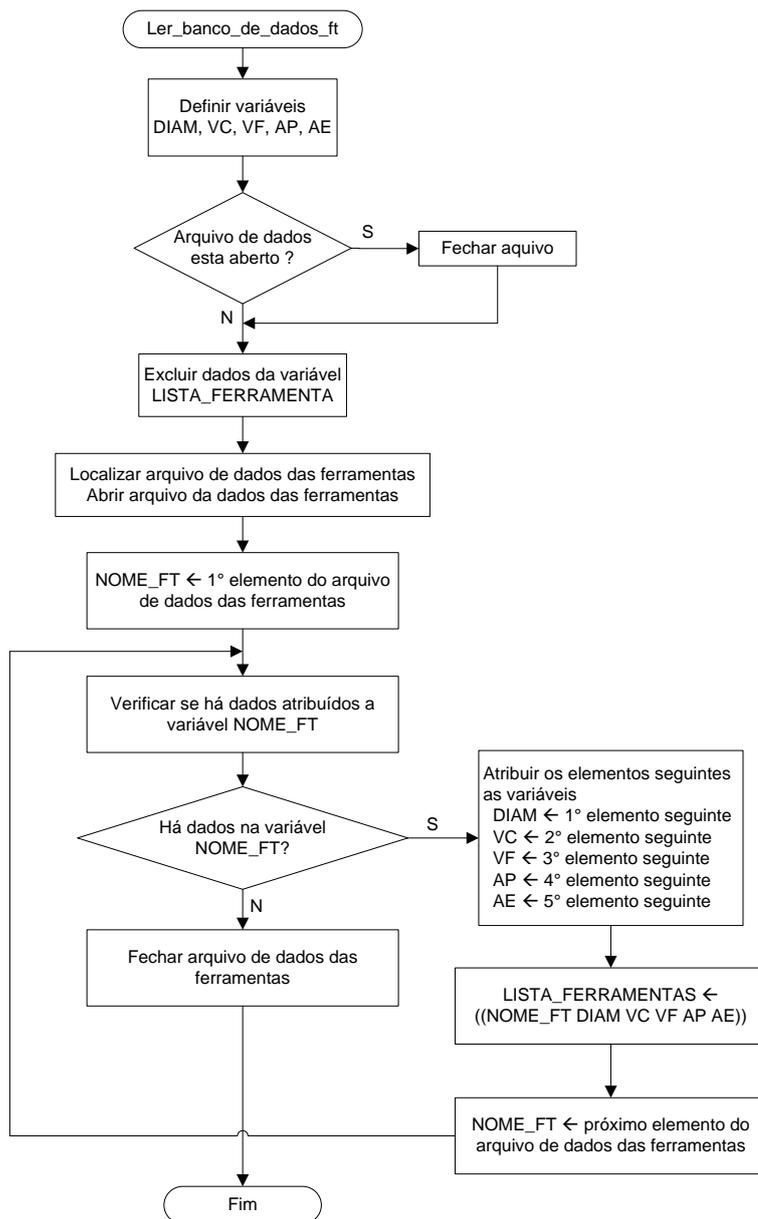


Figura 36 – Fluxograma do subprograma “Ler banco de dados ft”

O sistema inicia uma varredura no arquivo de dados, de modo que o primeiro elemento encontrado é inserido na variável definida no início desta rotina, e a testa para saber se há alguma informação inserida. Se a resposta retornar positiva, os próximos cinco elementos do arquivo de dados de ferramentas são inseridos, cada um em sua respectiva variável. Ao finalizar o sexto elemento, esses dados são inseridos na variável que as armazenaram. Novamente são excluídos os conteúdos das variáveis que recebem os dados do arquivo de dados; o próximo elemento é atribuído à primeira variável e o ciclo é feito novamente. Tal processo terá término

apenas quando o teste que verifica se há informação na variável que armazena o primeiro elemento de cada lista de ferramenta apresentar resultado negativo.

#### 4.1.7 Seleção de ferramenta

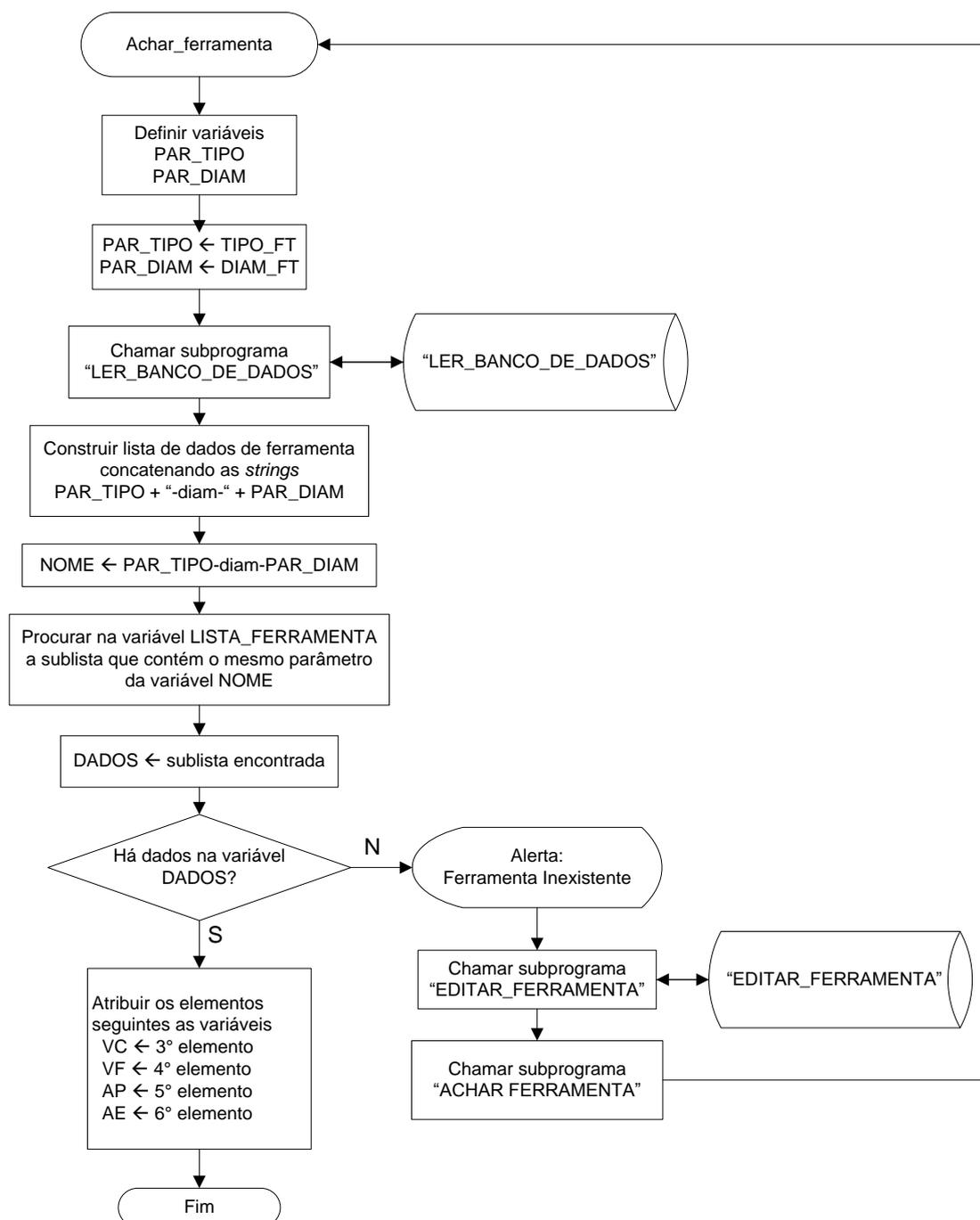
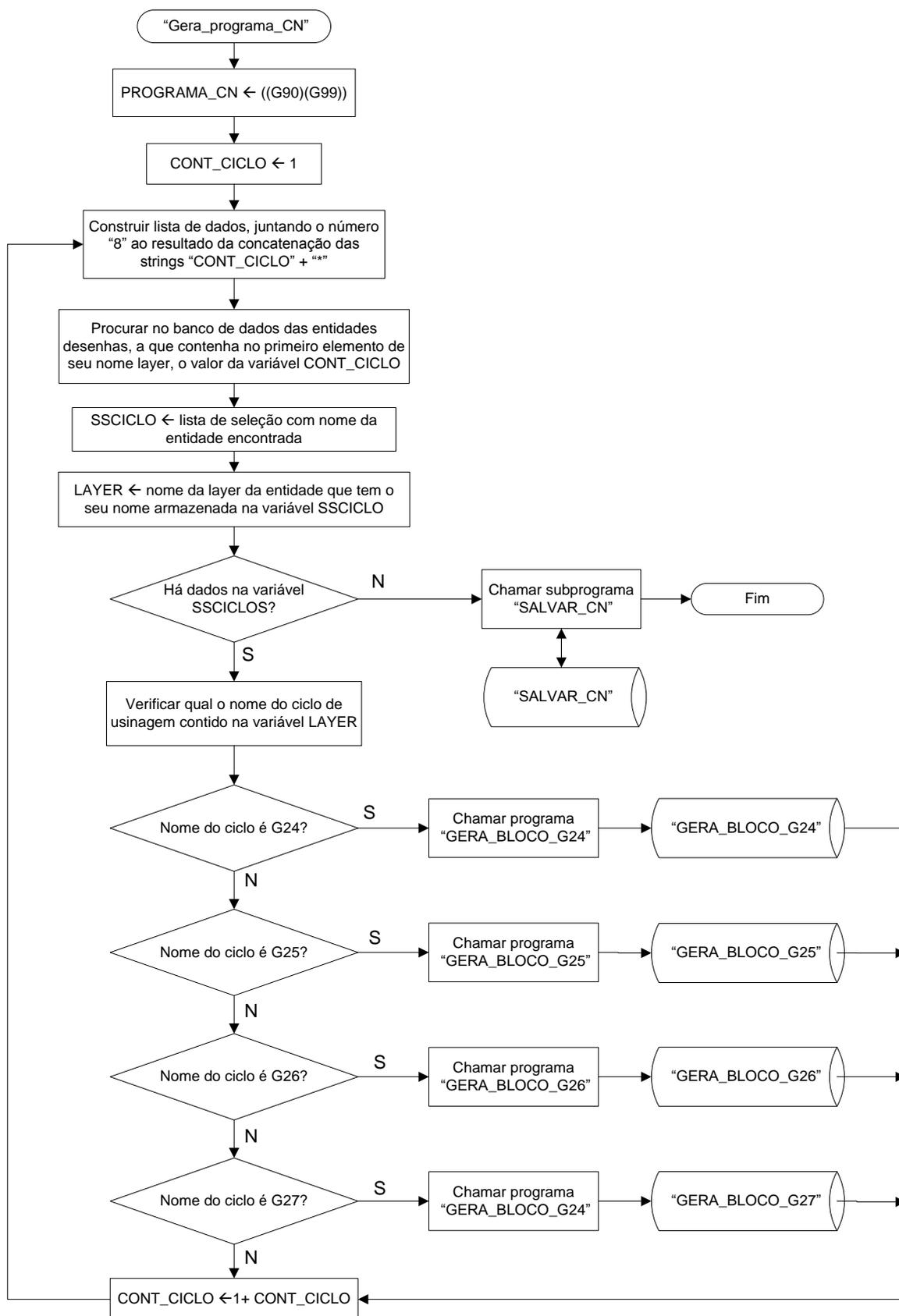


Figura 37 – Fluxograma do subprograma “Achar ferramenta”

O objetivo deste programa é encontrar e extrair os parâmetros da ferramenta que estão armazenados em uma variável. Esses parâmetros são usados pelo sistema para preencher as informações no programa CN. O fluxograma da figura 37 apresenta o funcionamento desta rotina. O programa principal passa para a rotina como parâmetro o nome e o diâmetro da ferramenta usada no ciclo de usinagem. Uma vez que uma ferramenta é definida para cada ciclo de usinagem inserido, o processamento consiste em realizar uma consulta à variável que armazena esses parâmetros a fim de identificar a ferramenta cujo nome e diâmetro coincidem com aquilo que é aplicado ao processo. Ao encontrar o parâmetro correspondente, esse e os cinco elementos seguintes são atribuídos a uma variável. A seguir, ela é testada para saber se há alguma informação inserida, e, se a resposta retornar positiva, os quatro últimos elementos contidos nesta variável são inseridos, cada um em sua respectiva variável. Caso o teste retorne negativo, o subprograma para editar ferramenta é chamado. Após a execução desse subprograma, essa rotina é reiniciada.

#### 4.1.8 Programa para identificar o ciclo de usinagem programado

Ao finalizar o processo de criação do ciclo de usinagem no sistema CAD, tem-se a opção de inserir no processo outro ciclo ou gerar o programa CN, conforme apresentado no fluxograma da figura 28. Quando a opção de gerar o programa CN é selecionada, o sistema executa esse subprograma, que tem por objetivo identificar na codificação dos ciclos a ordem em que foram inseridos e o nome do ciclo de usinagem e chamar a rotina correspondente que irá gerar os blocos do programa CN. O fluxograma da figura 38 apresenta o funcionamento desse subprograma. A variável que armazena os blocos gerados pelos subprogramas é inicializada, sendo nela inseridas as funções de inicialização do programa CN. Como já apresentado no item 4.1.1, cada ciclo possui uma codificação - um número que identifica a sequência em que foi criado e um nome, que é armazenada no nome da *layer* de cada entidade. Essa rotina realiza uma varredura no banco de dados das entidades a fim de identificar no nome da *layer* das entidades aquela que possui o número um em seu primeiro caractere.



**Figura 38 – Fluxograma do subprograma “Gera programa CN”**

Ao localizar essa entidade, o processamento armazena o nome do ciclo contido no nome da *layer* em uma variável. O sistema a testa para verificar se está vazia. Retornado a resposta verdadeira, uma rotina que finaliza o programa CN é chamada e o sistema é finalizado. Retornando a resposta falsa, o processamento identifica qual o nome do ciclo de usinagem está armazenado na variável e chama o subprograma que gera os blocos de programa CN do respectivo ciclo. Ao finalizar o sistema, o número do contador é incrementado de mais uma unidade e uma nova varredura é realizada no banco de dados das entidades. Esse processo se repete até que a resposta do teste na variável retorne verdadeira.

## 4.2 Definição geométrica dos ciclos

A seguir é apresentado o desenvolvimento dos programas principais que geram o desenho da peça e as entidades de armazenamento de dados de cada ciclo de usinagem selecionados para essa metodologia.

### 4.2.1 Furação em Padrão Circular

O ciclo para usinagem de furos distribuídos em arranjo circular igualmente espaçados, ou ciclo G24 no Comando Mach9, como já apresentado no item 3.3.1, é aplicado em operações de roqueamentos, execução de subprogramas ou sub-rotinas CN. O objetivo desse programa principal é criar o desenho que representa este ciclo de usinagem e as entidades que armazenam suas informações geométricas, conforme mostrado anteriormente na figura 24. Essas entidades são linhas, círculos, arcos e texto, desenhadas em uma *layer* específica que as identificam. A sequência circular de furos é distribuída uniformemente, podendo ser de 0° a 360° em relação ao centro definido. Os dados utilizados para a realização do desenho dos furos e das entidades que armazenam as informações geométricas sobre a peça bruta são: diâmetro da ferramenta, posição da ferramenta no magazine, ponto de referência na peça, coordenadas em X e Y do centro da

sequência de furos, tipo de entrada de dado para definição do primeiro furo (por coordenada X, Y ou por centro e raio), número total de furos, ângulo entre os furos, profundidade dos furos e altura de aproximação. Esses dados são calculados e inseridos no desenho pela rotina que, posteriormente, os utilizará na elaboração dos blocos do programa CN a ser gerado. O fluxograma da figura 39 apresenta o funcionamento desse subprograma. Para o melhor entendimento do desenvolvimento do algoritmo, a explicação será dividida em etapas.

#### 1ª etapa – Codificação de identificação do novo ciclo G24.

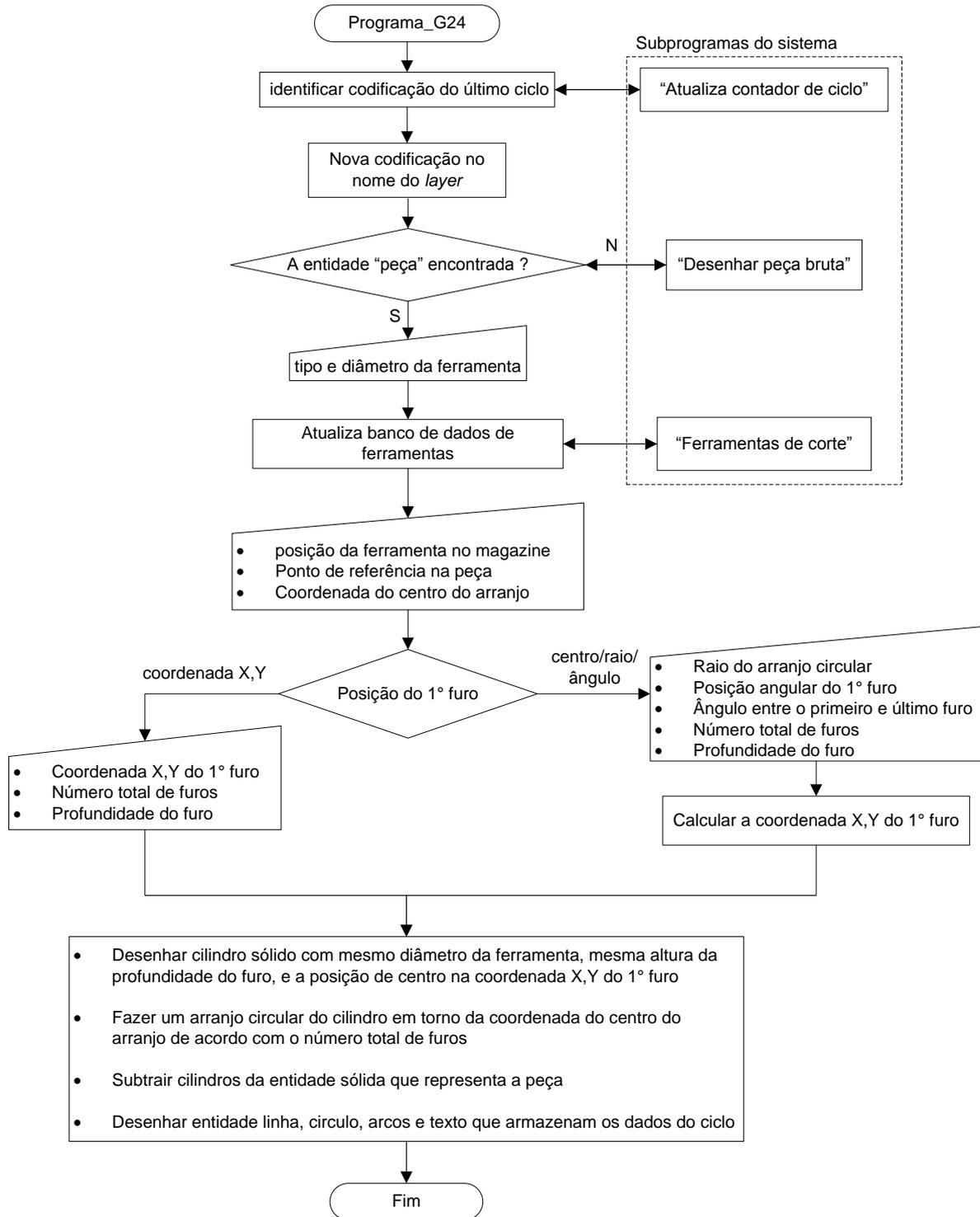
Inserir-se nessa etapa a codificação de identificação para o novo ciclo de usinagem. O sistema, através do subprograma descrito no item 4.1.1 (Figura 31), verifica a última codificação inserida no processo e gera a nova identificação para esse ciclo.

#### 2ª etapa – Desenho da geometria da peça bruta

Nessa etapa, o processamento consulta no banco de dados das entidades se há no nome do *layer* algum dado que caracterize a existência de ciclo já desenvolvido. Se o sistema o encontrar, significa que a peça bruta já foi definida. Neste caso, existe a possibilidade de apagar essas entidades e recomeçar o processo para um novo programa CN. Se o sistema não encontrar, são apagadas todas as entidades desenhadas e um novo arquivo é inicializado. No caso de ser o primeiro ciclo, o desenho da peça bruta é gerado através de subprograma específico.

#### 3ª etapa – Definição de ferramenta

Nessa etapa define-se a ferramenta a ser usada na usinagem do ciclo. Os parâmetros usados são o tipo e o diâmetro da ferramenta. Esses parâmetros são passados para o subprograma descrito no item 4.1.6, que faz a seleção dos dados geométricos e tecnológicos no banco de dados de ferramentas. Essas informações são inseridas em uma variável, que é usada pelo programa na criação do desenho da peça.



**Figura 39 – Fluxograma do “programa G24”**

#### 4ª etapa – Determinar número da ferramenta

Nesta etapa, é informado ao programa o número da posição da ferramenta no magazine da máquina CNC. O sistema insere essa informação em forma de entidade de texto no mesmo *layer* referente a esse ciclo, fazendo com que essa

informação possa ser visualizada na área gráfica do CAD, conforme mostrado anteriormente na figura 22.

#### 5ª etapa – Definição das coordenadas do centro do círculo

Para que o arranjo seja construído, a localização das coordenadas do centro do círculo nas direções X e Y é informada ao sistema. Esses valores podem ter como referência um ponto selecionado na superfície da peça ou a origem do sistema de coordenadas do CAD. Essas coordenadas são armazenadas em uma variável juntamente com o valor de Z do ponto de referência para serem usadas posteriormente na geração do desenho.

#### 6ª etapa– definição das coordenadas do primeiro furo

O objetivo desta etapa é definir ao sistema as coordenadas X e Y do primeiro furo. Há duas opções de entrada desses dados: a localização do primeiro furo é dada a partir das coordenadas X e Y em relação ao ponto de referência da peça ou, a partir do centro do círculo dos furos, informam-se a posição angular do primeiro furo e o raio do círculo. Sendo selecionada a primeira opção, os dados inseridos são somados com as coordenadas do ponto de referência, obtendo-se assim a posição do primeiro furo dada em relação à origem do sistema de coordenada do CAD. Essa opção é usada para desenhar um círculo de furos em um ângulo de 360°, conforme apresentado na figura 11a.

Se selecionada a segunda opção, os dados são transformados em coordenadas X e Y em relação ao centro do círculo, através das equações 1 e 2, e, depois, são somados à coordenada do ponto de referência, obtendo-se assim a posição do primeiro furo, dada em relação à origem do sistema de coordenada do CAD. Essa opção é usada para desenhar um círculo de furos em um ângulo menor que 360°, conforme apresentado na figura 11b. Outros parâmetros também definidos nesta etapa são o número de furos, a profundidade e a coordenada angular do último furo.

$$coord.X = raio\ do\ circulo * \coseno\left(\frac{\pi * \hat{a}ngulo\ do\ 1^o\ furo}{180}\right) \quad (1)$$

$$\text{coord. } Y = \text{raio do círculo} * \text{seno} \left( \frac{\pi * \text{ângulo do } 1^{\circ} \text{ furo}}{180} \right) \quad (2)$$

### 7ª etapa – Desenho da peça

Gera-se nessa etapa o desenho da peça. Inicialmente, um cilindro sólido é desenhado com os dados de diâmetro da ferramenta, profundidade e coordenadas do primeiro furo, determinados nas etapas anteriores. Após, um arranjo circular é executado em relação ao centro do círculo, definido na quinta etapa, e o número de furos, informado na sexta etapa. A distância angular entre os furos é executada em função da opção de entrada de dados selecionada na etapa anterior. As equações 3 e 4 definem essa distância para a primeira e segunda opção, respectivamente.

$$\text{distância angular} = \frac{360}{\text{número de furos}} \quad (3)$$

$$\text{distância angular} = \frac{\text{ângulo final}}{\text{número de furos} - 1} \quad (4)$$

Na sequência, subtraem-se os elementos sólidos que representam os furos do elemento sólido que representa a peça, formando-se assim o desenho da peça final.

### 8ª etapa – Desenho das entidades de armazenamento de dados

Os dados para gerar o desenho dessas entidades são definidos nas etapas anteriores. Inicialmente, uma entidade círculo é desenhada na mesma posição do primeiro furo, e com o mesmo diâmetro da ferramenta. Na sequência, uma linha é desenhada paralelamente ao centro do primeiro furo, sendo o seu comprimento coincidente com a profundidade. A seguir, os arcos que determinam o valor do ângulo entre os furos e o ângulo final dos furos são desenhados. Esses têm como

ponto inicial o centro do primeiro furo e final o centro do segundo e do último furo, respectivamente.

#### 4.2.2 Furação em Padrão Linear

O ciclo para usinagem de furos distribuídos em arranjo linear igualmente espaçados, ou ciclo G25 no Comando Mach9, como já apresentado no item 3.3.2, é aplicado em operações de roqueamentos, execução de subprogramas ou subrotinas CN. O objetivo desse programa é criar o desenho que representa este ciclo de usinagem e as entidades que armazenam suas informações geométricas, conforme mostrado anteriormente na figura 25. Essas entidades são linhas, círculos e texto, que são desenhadas em uma *layer* específica que as identificam.

A sequência linear de furos é distribuída uniformemente na direção X e Y dos eixos da máquina. Os dados utilizados para a realização do desenho dos furos e das entidades que armazenam as informações geométricas sobre a peça são: diâmetro da ferramenta, posição da ferramenta no magazine, ponto de referência, coordenadas em X e Y do primeiro furo em relação ponto de referência, distância entre eles nas direções X e Y, número e profundidade dos furos e altura de aproximação. Esse dados são calculados e inseridos no desenho pela rotina que posteriormente os utilizará na elaboração dos blocos do programa CN a ser gerado. O fluxograma da figura 40 apresenta o funcionamento desse subprograma. Para o melhor entendimento do desenvolvimento do algoritmo deste programa, a explicação será dividida em etapas. As seguintes etapas são idênticas às descritas no item 4.2.1 (exceto a aplicação do nome do ciclo, que neste caso é G25).

- 1ª etapa – Codificação de identificação do novo ciclo G25;
- 2ª etapa – Desenho da geometria da peça bruta;
- 3ª etapa – Definição de ferramenta;
- 4ª etapa – Determinação do número da ferramenta.

As etapas descritas a seguir são específicas para esse ciclo de usinagem.

#### 5ª etapa – Definição da coordenada do primeiro furo

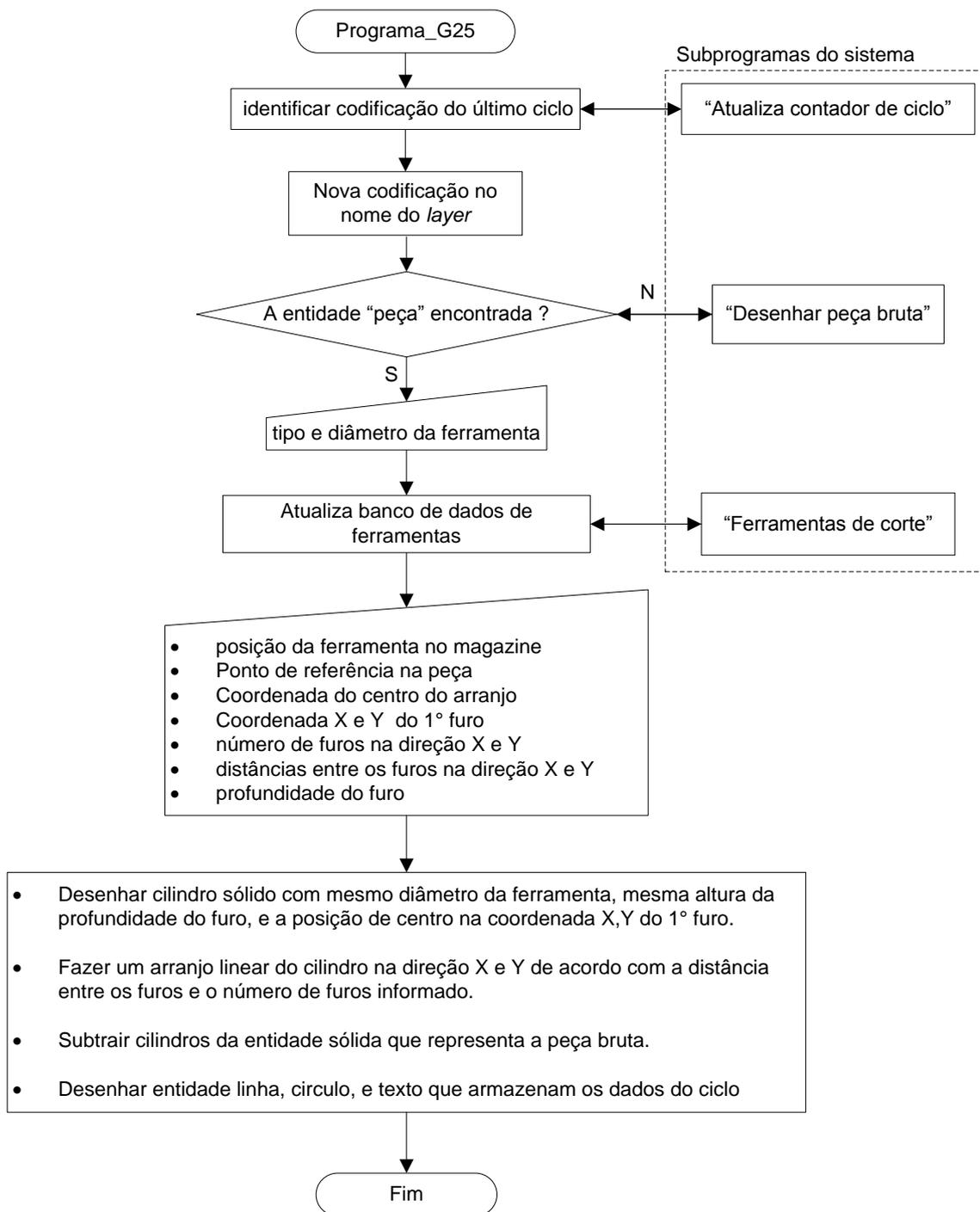
Para que o arranjo seja construído, a localização das coordenadas do primeiro furo na direção X e Y é informada ao sistema. Esses valores podem ter como referência um ponto selecionado na superfície da peça ou a origem do sistema de coordenadas do CAD. Essas coordenadas são armazenadas em uma variável juntamente com o valor de Z do ponto de referência para serem usadas posteriormente na geração do desenho. Outros parâmetros também definidos nesta etapa são o número de furos, sua profundidade e as distâncias X e Y entre eles.

#### 6ª etapa – Desenho da peça

Gera-se nessa etapa o desenho da peça. Inicialmente, um cilindro sólido é desenhado com os dados de diâmetro da ferramenta, profundidade e coordenadas do primeiro furo, determinados nas etapas anteriores. Após, um arranjo linear é executado nas direções X e Y tendo como parâmetros a distância entre os furos e o número de furos. Na sequência, subtraem-se os elementos sólidos que representam os furos do elemento sólido que representa a peça, formando-se assim ao desenho da peça final.

#### 7ª etapa – Desenho das entidades de armazenamento de dados

Os dados para gerar o desenho dessas entidades são definidos nas etapas anteriores. Inicialmente, uma entidade círculo é desenhada na mesma posição do primeiro furo, e com o mesmo diâmetro da ferramenta. Na sequência, uma linha é desenhada paralelamente ao centro do primeiro furo, sendo o seu comprimento coincidente com a profundidade. As linhas que determinam as distâncias entre os furos e o número total de furos iniciam na mesma posição da anterior e finalizam no furo seguinte e no último furo do arranjo, respectivamente, em ambas as direções.



**Figura 40 – Fluxograma do “programa G25”**

### 4.2.3 Usinagem de Cavidade

O ciclo de usinagem de cavidade, ou ciclo G26, no comando Mach9, como já apresentado no item 3.3.3, é aplicado para a operação de desbaste e/ou acabamento em cavidade retangular ou circular. O objetivo deste programa é criar o desenho que representa este ciclo de usinagem e as entidades que armazenam suas informações geométricas, conforme mostrado anteriormente na figura 26.

Essas entidades são linhas, círculos e texto, que são desenhadas em uma *layer* específica que as identificam. Os dados utilizados para a realização do desenho da cavidade e das entidades que armazenam as informações geométricas são: diâmetro da ferramenta, posição da ferramenta no magazine, ponto de referência na peça, profundidade da cavidade, coordenadas em X e Y da primeira e da segunda vértice no caso de cavidade retangular, ou coordenada de centro e diâmetro da no caso de cavidade circular. Esses dados são trabalhados e inseridos no desenho pela rotina que posteriormente os utilizará na elaboração dos blocos do programa CN a ser gerado. O fluxograma da figura 41 apresenta o funcionamento desse programa. Para o melhor entendimento do desenvolvimento do algoritmo deste programa, a sua explicação será dividida em etapas. As seguintes etapas são idênticas às descritas no item 4.2.1, sendo somente diferente aplicação do nome do ciclo, que neste caso é G26.

1ª etapa – codificação de identificação do novo ciclo G26.

2ª etapa– Desenho da geometria da peça bruta

3ª etapa– Definição de ferramenta

4ª etapa– Determinar número da ferramenta

As etapas seguintes são específicas para esse ciclo de usinagem.

5ª etapa– determinação do tipo de geometria da cavidade.

Inserir-se nesta etapa se o tipo de cavidade a ser desenhada será retangular ou circular. A resposta é avaliada pelo sistema, direcionando-o para que os parâmetros necessários sejam inseridos na rotina, e a geometria da peça e as entidades de armazenam os dados sejam construídas.

#### 6ª etapa– Desenho da geometria retangular

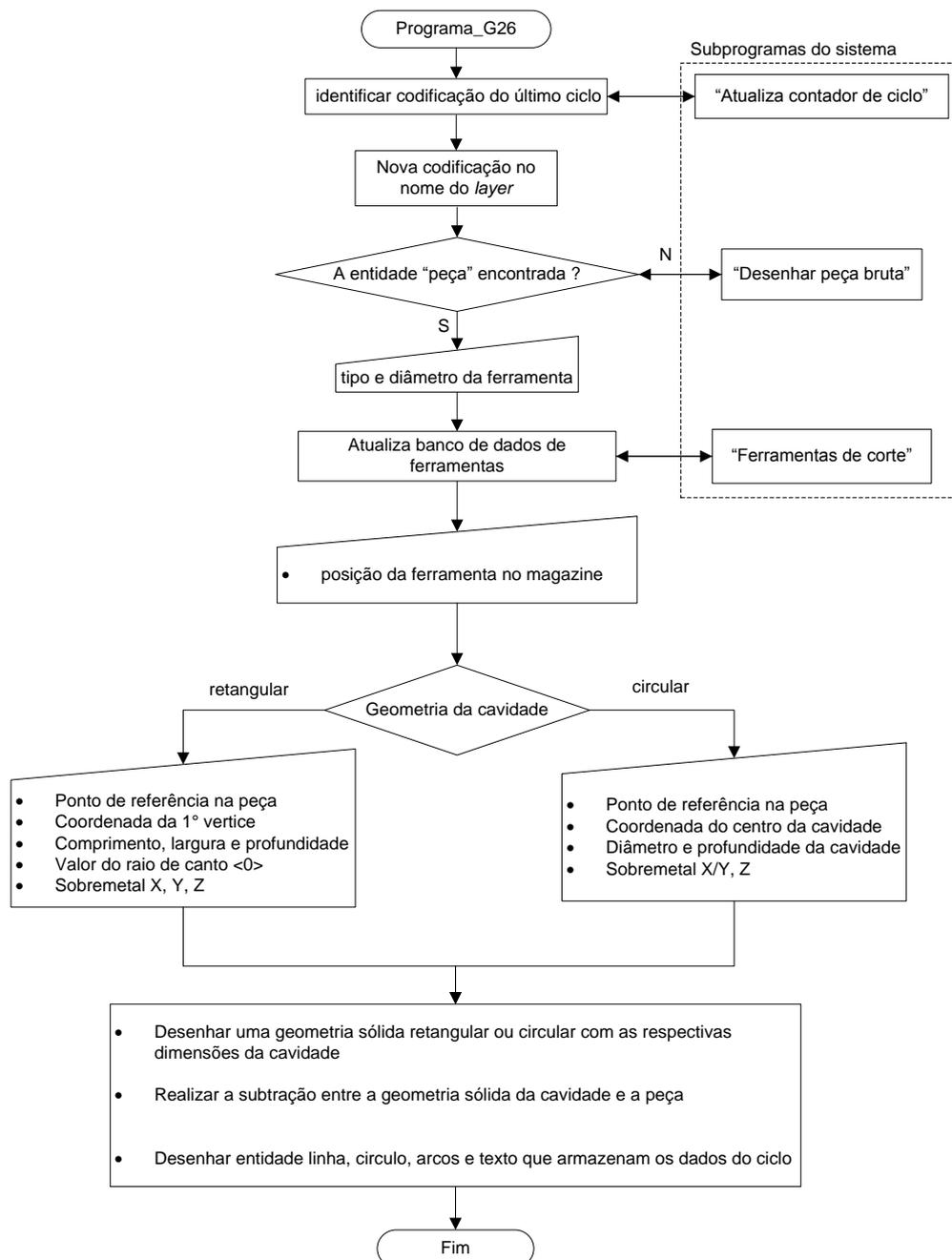
Nesta etapa inicialmente são inseridos todos os parâmetros que definem a geometria da cavidade e os dados referentes ao processo de usinagem. Um ponto de referências para a origem das coordenadas primeiramente é determinado. A seguir, é informada ao sistema as coordena X e Y da primeira vértice em relação a esse ponto. A vértice oposta é definida através do comprimento, largura e profundidade da cavidade. Se nessa geometria, houver raio nos cantos conforme apresentado anteriormente na figura 16, e sobremetal nas paredes e no fundo, esse valor também é informado nessa etapa. Com esses dados, uma geometria sólida em forma de caixa é construída pela rotina. Essa é extraída da geometria da peça, ficando como resultado uma cavidade conforme mostrado anteriormente na figura 21. Para finalizar essa etapa, o sistema desenha as entidades que armazenam os dados geométricos e tecnológicos desse processo. Linhas retas que tem como ponto inicial as coordenadas da primeira vértice, e como ponto final, os dados inseridos para o comprimento, largura e profundidade, e um circulo com o valor do raio de canto se caso houver, armazenam os dados geométricos da cavidade. Outras linhas retas que possuem as mesmas coordenadas X e Y iniciais da anterior, porém no plano Z1, também são desenhadas. Os seus comprimentos são relacionados com os valores inseridos para o sobremetal

#### 7ª etapa– Desenho da geometria retangular ou circular

Nesta etapa inicialmente são inseridos todos os parâmetros que definem a geometria da cavidade selecionada. Um ponto de referências para a origem das coordenadas primeiramente é determinado. No caso da geometria retangular, são informados ao sistema, as coordena X e Y da primeira vértice em relação a esse ponto. A vértice oposta é definida através do comprimento, largura e profundidade da cavidade. Se nessa geometria, houver raio nos cantos conforme apresentado anteriormente na figura 16, e sobremetal nas paredes e no fundo, esse valor também é informado nessa etapa.

Se a geometria for a circular, são informados ao sistema as coordenas X e Y do centro dessa cavidade e o valor do seu diâmetro. Se houver sobremetal na parede e no fundo, esse valor também é informado nessa etapa.

Com esses dados, uma geometria sólida em forma de caixa ou cilindro é construída pela rotina. Essa é extraída da geometria da peça, ficando como resultado uma cavidade conforme mostrado anteriormente na figura 21.



**Figura 41 – Fluxograma do “programa G26”**

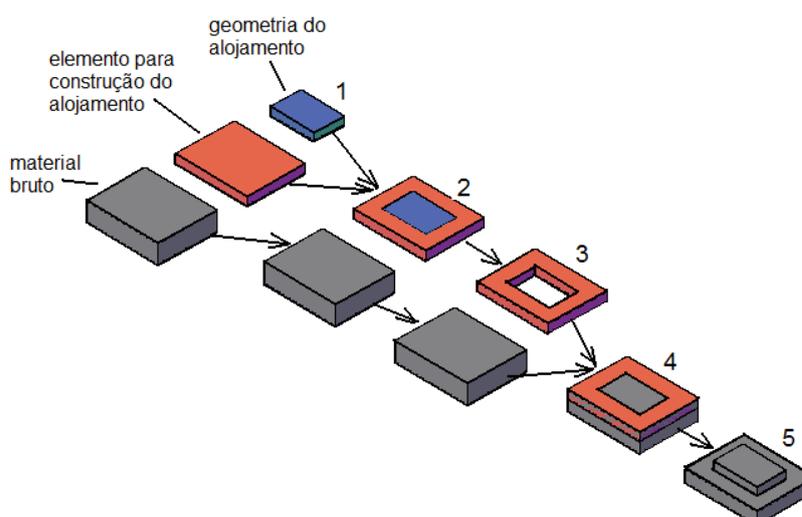
### 8ª etapa– Desenho das entidades que armazenam dados

Para a cavidade retangular, são desenhadas linhas retas que se iniciam junto a sua primeira vértice. Os dados, comprimento, largura e profundidade inseridos no sistema na etapa anterior, definem também dimensões dessas linhas.

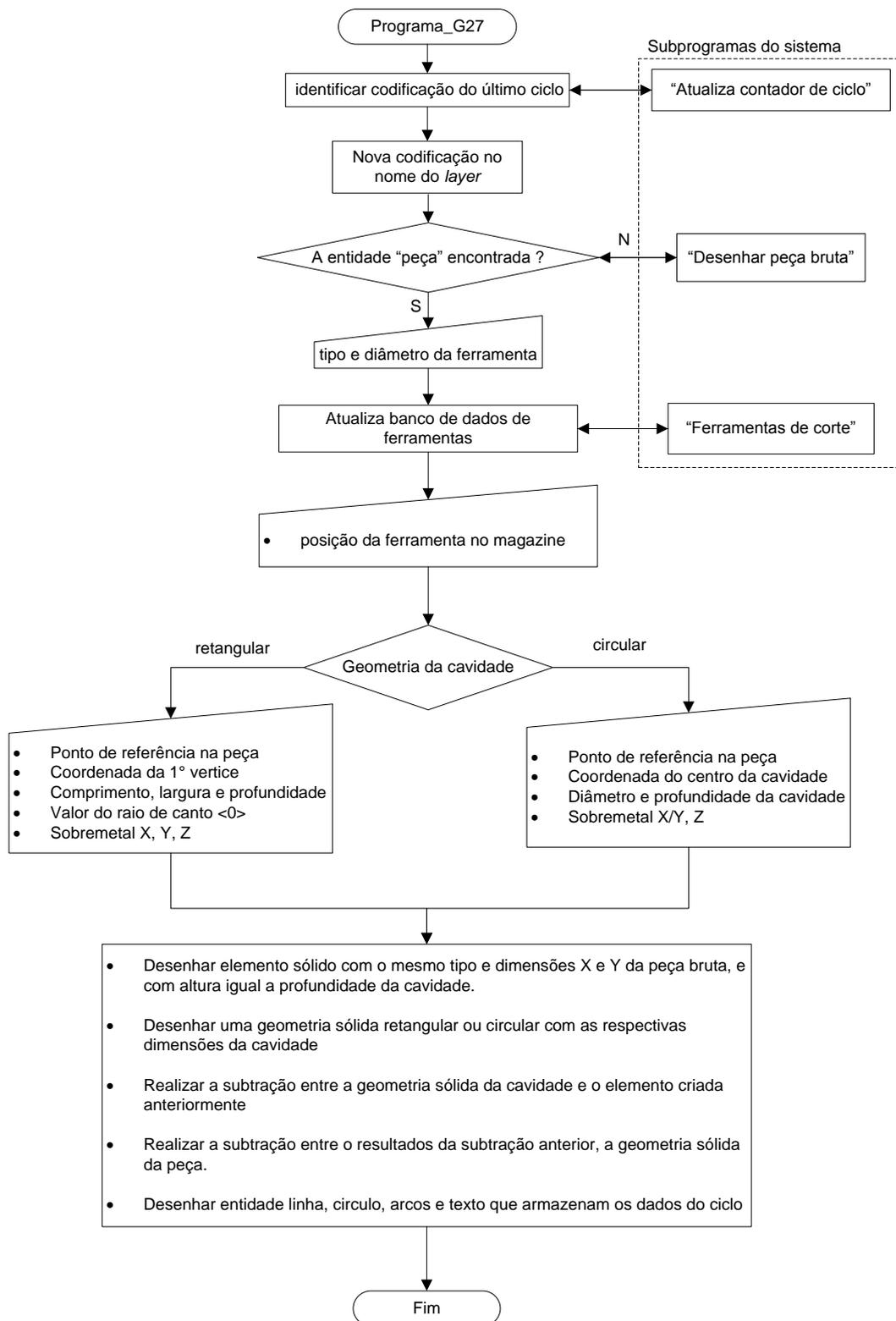
Na cavidade circular, um círculo é desenhado coincidente à borda superior da cavidade, armazenando em seu banco de dados o diâmetro da mesma. Para armazenar a profundidade da cavidade, uma linha reta paralela ao seu centro é gerada. Essa inicia-se no plano superior da cavidade e finaliza-se no fundo da mesma. Também são desenhadas as linhas para armazenar os valores do sobremetal e o círculo para armazenar o valor do diâmetro da ferramenta, conforme apresentado anteriormente nas figuras 24 e 25.

#### 4.2.4 Usinagem de resalto ao redor de uma área

O ciclo de usinagem de resalto ao redor de uma área específica, ou ciclo G27, no comando Mach9, como já apresentado no item 3.3.4, é aplicado para a operação de desbaste e/ou acabamento em resalto retangular ou circular ao redor de uma determinada área. O objetivo desse programa é criar o desenho que representa este ciclo de usinagem e as entidades que armazenam suas informações geométricas, conforme mostrado anteriormente na figura 28. O desenvolvimento deste programa é igual do programa do ciclo G26, sendo diferente somente no método de construção do desenho que representa o ciclo, conforme apresenta a figura 42. O fluxograma da figura 43 apresenta o funcionamento deste programa.



**Figura 42 – Método de construção do desenho para o ciclo G27**



**Figura 43 – Fluxograma do “programa G27”**

### 4.3 Geração do programa CN

Gerar o programa CN a partir da leitura dos dados das entidades é o objetivo final dessa proposta. Neste item é apresentada a metodologia implementada no aplicativo que extrai do banco de dados das entidades as informações que são inseridas nos blocos formadores do programa CN. Para cada um dos ciclos inseridos no processo durante a programação da peça, o subprograma descrito no item 4.1.8 identifica no conjunto de *layer* a sua codificação. O número da sequência e o nome do ciclo contido nessa codificação representam a ordem em que foram inseridos e a rotina que deverá ser chamada para gerar o programa CN, respectivamente. Essas rotinas são apresentadas a seguir.

A variável existente após cada palavra que define a sequência dos blocos CN recebe um valor extraído do banco de dados de uma entidade. Inicialmente, o sistema faz uma varredura em todos os bancos de dados das entidades desenhadas para localizar as que possuem o mesmo nome da *layer*. As entidades localizadas possuem o seu nome inserido em uma variável, pois fazem parte de um mesmo ciclo de usinagem. Após, o banco de dados dessas entidades passa por uma nova varredura. Porém, o objetivo agora é localizar uma entidade específica através do dado contido no código chave número zero, que é o tipo da entidade. A seguir, é apresentado o desenvolvimento da rotina que gera o programa CN para cada um dos ciclos de usinagem.

#### 4.3.1 Geração do programa do ciclo de usinagem em Padrão Circular

A função desta rotina é gerar o programa CN do ciclo de usinagem G24, que é definido pelos seguintes blocos:

**T**(n° da posição da ferramenta no magazine) **M6**; (nome ferramenta)

**O**(n° do corretor da ferramenta) **S**(rotação) **M3**

**G0 X**(coordenada X do primeiro furo) **Y**(coordenada Y do primeiro furo)

**Z2.**

**G81 Z**(profundidade do furo) **R2. F**(avanço da ferramenta)

**G24 R**(raio do círculo) **C**(ângulo inicial do primeiro furo) **I**(coord. do centro do círculo em X) **J**(coord. do centro do círculo em Y) **L**(número de furos) **B**(ângulo entre os furos)

**G80**

**GZO M5**

As entidades arco, círculo, linha e texto desenhadas para armazenar os dados desse ciclo são identificados no banco de dados por seu código de grupo número zero. Do banco de dados do arco, são extraídos as coordenadas de centro, o ângulo inicial e final e o valor do raio. Do banco de dados do círculo, são extraídos a coordenada de centro e o raio. Já da linha, extrai-se o valor Z do ponto final; e, da entidade texto, extrai-se o texto. De posse desses dados, é só utilizá-los para o que convém. Esses dados juntamente com seu código são exemplificados pela figura 44.

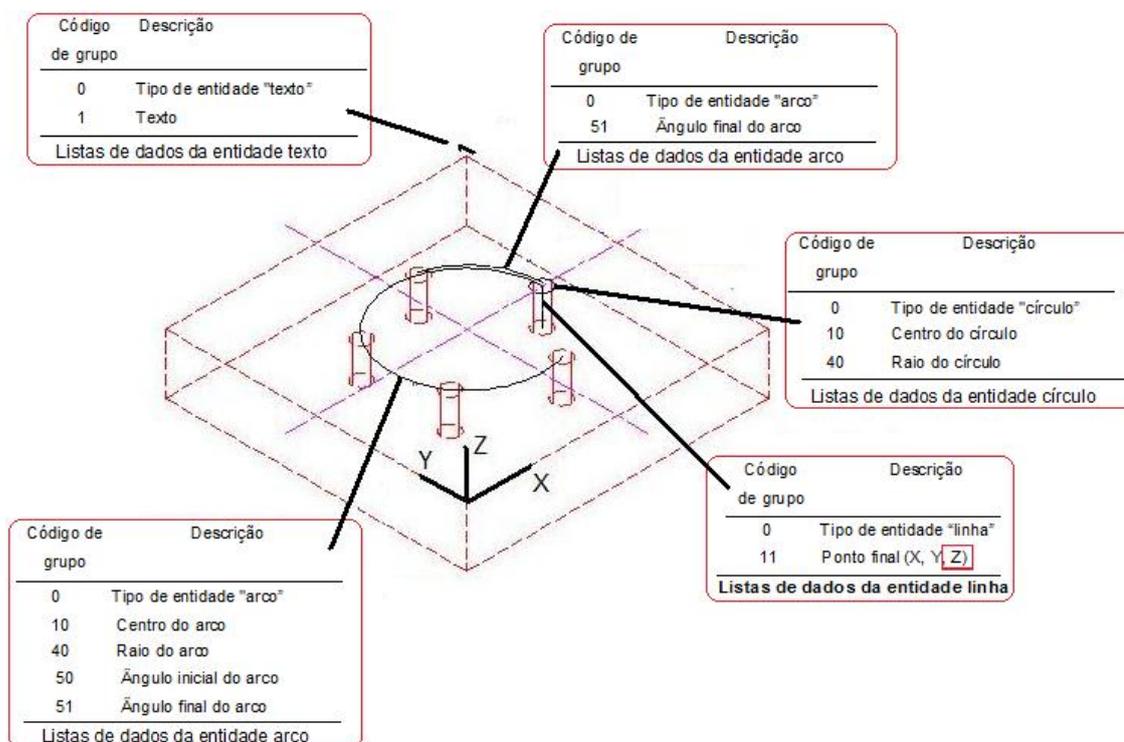


Figura 44 – Dados extraídos das entidades do ciclo G24

A primeira entidade específica a ser procurada é a do tipo texto. Dessa, extrai-se o valor contido no código chave número um. Esse valor identifica o número da posição da ferramenta no magazine da máquina e o número do corretor da ferramenta. O nome da ferramenta é determinado pelo seu processo e diâmetro. No processo de furação, usa-se a broca, que será o primeiro dado do nome. O segundo dado é o diâmetro da ferramenta. Esse valor encontra-se no código chave número 40 do banco de dados em uma entidade tipo círculo, desenhada no plano Z1. Para determinar a rotação e o avanço, busca-se no arquivo de dados das ferramentas o valor da velocidade de corte e do avanço. Para localizar esse dado, usa-se o nome da ferramenta obtido anteriormente. Através da fórmula da velocidade de corte e do diâmetro da ferramenta, é obtida a rotação.

As coordenadas X e Y do primeiro furo são extraídas da mesma entidade tipo círculo, que contém o diâmetro da ferramenta, pois essa foi desenhada na mesma posição do primeiro furo. Esses valores da posição de centro de círculo localizam-se no código de grupo número 10. O valor do afastamento de segurança foi fixado em 2 milímetros acima da superfície de referência da peça. A profundidade do furo está armazenada no valor de Z do ponto final da linha, que está no código de grupo número 11. O raio do círculo, o ângulo inicial do primeiro furo e as coordenadas de centro do círculo são obtidos através do banco de dados do arco que está desenhado no plano Z zero, nos códigos de grupo 40, 50 e 10, respectivamente. O número de furo é calculado através da equação 5, e o número de furos é o próprio valor do ângulo do arco menor, dado em graus.

$$\text{número de furos} = \left( \frac{\text{ângulo final} - \text{ângulo inicial do arco maior}}{\text{ângulo final} - \text{ângulo inicial do arco menor}} \right) + 1 \quad (5)$$

#### 4.3.2 Geração do programa do ciclo de usinagem em Padrão linear

A função desta rotina é gerar o programa CN do ciclo de usinagem G25, que é definido pelos seguintes blocos:

**T**(n° da posição da ferramenta no magazine) **M6**; (nome ferramenta)

**O**(n° do corretor da ferramenta) **S**(rotação) **M3**

**G0 X**(coordenada X do primeiro furo) **Y**(coordenada Y do primeiro furo)

**Z2.**

**G81 Z**(profundidade do furo) **R2. F**(avanço da ferramenta)

**G25 X**(distância entre furos em X) **Y**(distância entre furos em Y) **I**(n° de furos em X) **J**(n° de furos em Y)

**G80**

**GZO M5**

As entidades círculo, linha e texto, desenhadas para armazenar os dados desse ciclo, são identificadas no banco de dados por seu código de grupo número zero. Do banco de dados do círculo, são extraídos a coordenada de centro e o raio. Já da linha, extrai-se o valor Z do ponto final; e, da entidade texto, extrai-se o texto. De posse desses dados, é só utilizá-los para o que for necessário. Esses dados juntamente com seus códigos são exemplificados pela figura 45.

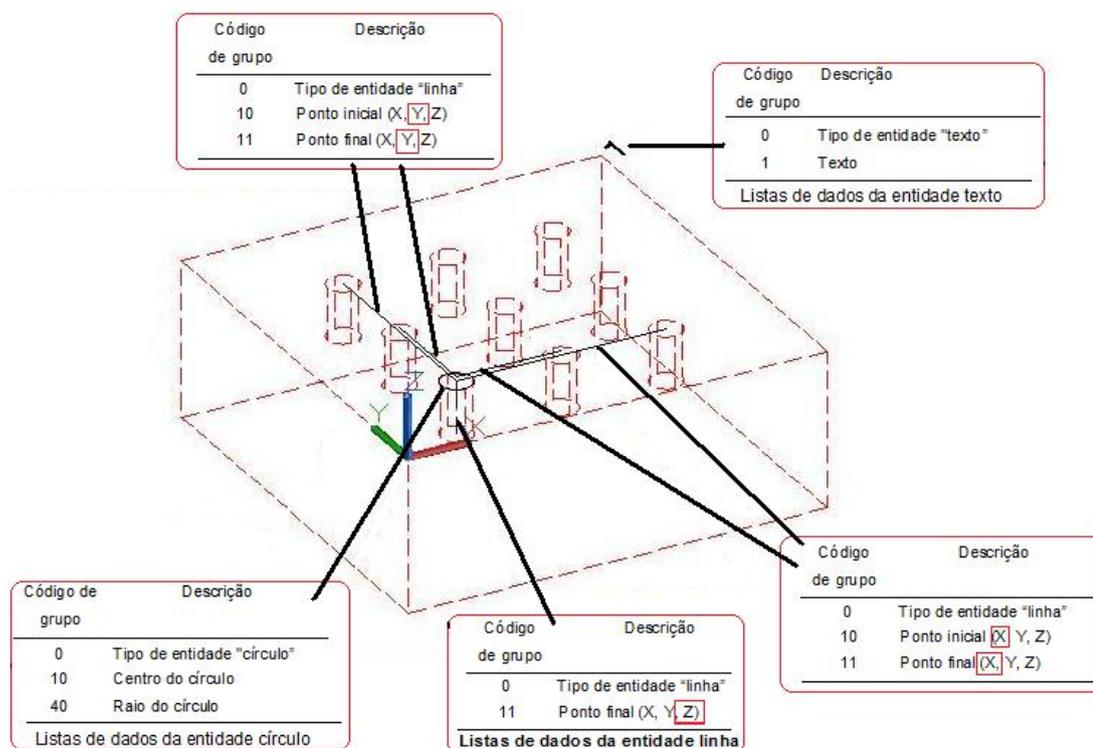


Figura 45 – Dados extraídos das entidades do ciclo G25

O modo de identificação das entidades que definem esse ciclo é semelhante ao do ciclo G23, sendo diferente somente no bloco que define o ciclo G24. A distância entre os furos e o número de furos é obtida através das linhas desenhadas nas direções X e Y. Como se têm duas linhas no mesmo plano, antes de extrair seus dados, o sistema verifica a qual direção pertence a linha selecionada. Para a linha da direção X, é verificado se os valores de Y do ponto inicial e final são idênticos. Se forem idênticos, a linha X pertence a essa direção; se não forem, ela pertence à direção Y.

A distância entre os furos é dada pelo comprimento da linha menor, que está desenhada no plano Z1. Após localizar a linha correta em relação à direção, a distância é calculada através da subtração dos valores entre os pontos finais e iniciais, contidos no banco de dados de cada entidade. As equações 6 e 7 são aplicadas a esse procedimento.

$$\text{distância em X} = \text{valor do ponto final em X} - \text{valor do ponto inicial em X} \quad (6)$$

$$\text{distância em Y} = \text{valor do ponto final em Y} - \text{valor do ponto inicial em Y} \quad (7)$$

O número de furos na direção X e Y é calculado através das equações 8 e 9, respectivamente.

$$n^{\circ} \text{ de furos em X} = \left( \frac{\text{comprimento da linha maior em X}}{\text{comprimento da linha menor em X}} \right) + 1 \quad (8)$$

$$n^{\circ} \text{ de furos em Y} = \left( \frac{\text{comprimento da linha maior em Y}}{\text{comprimento da linha menor em Y}} \right) + 1 \quad (9)$$

### 4.3.3 Geração do programa do ciclo de usinagem de cavidades

A função desta rotina é gerar o programa CN do ciclo de usinagem G26, que é definido pelos seguintes blocos:

**T**(n° da posição da ferramenta no magazine) **M6**; (nome ferramenta)  
**O**(n° do corretor da ferramenta) **S**(rotação) **M3**  
**G0 X**(coordenada X do ponto inicial da cavidade) **Y**(coordenada Y do ponto inicial da cavidade)  
**Z2**.  
**G26 X**(coord. canto oposto da cavidade em X) **Y**(coord. canto oposto da cavidade em Y) **Z**(profundidade final da cavidade) **I**(sobremetal para acabamento em X) **J**(sobremetal para acabamento em Y) **K**(sobremetal para acabamento em Z) **U**(profundidade de corte) **W**(largura de corte ao longo dos eixos X e Y) **R**(raio de canto da cavidade) **F**(avanço de corte no desbaste) **H**(avanço de corte no acabamento) **V**(avanço de penetração da ferramenta)  
**GZO M5**

As entidades círculo, linha e texto, desenhadas para armazenar os dados desse ciclo, são identificadas no banco de dados por seu código de grupo número zero. Do banco de dados do círculo, são extraídos a coordenada de centro e o raio. Já da linha, extrai-se o valor X, Y e Z do ponto inicial e final; e, entidade texto, extrai-se o texto. De posse desses dados, é só utilizá-los para o que for conveniente. Esses dados juntamente com seus códigos são exemplificados pela figura 46.

O método de identificação das entidades que definem esse ciclo é semelhante ao do ciclo G24, sendo diferente somente no bloco que define as coordenadas X e Y do ponto inicial da cavidade e no bloco que define o ciclo G26. Inicialmente, o sistema seleciona as linhas que estão desenhadas no plano superior da cavidade da peça. Essas linhas determinam o ponto inicial, o comprimento e a largura da cavidade. Antes de extrair a coordenada X e Y do ponto inicial, é necessário determinar a qual direção pertence à linha selecionada. Para isso, o sistema verifica se os valores de Y inicial e final dessa linha são idênticos.

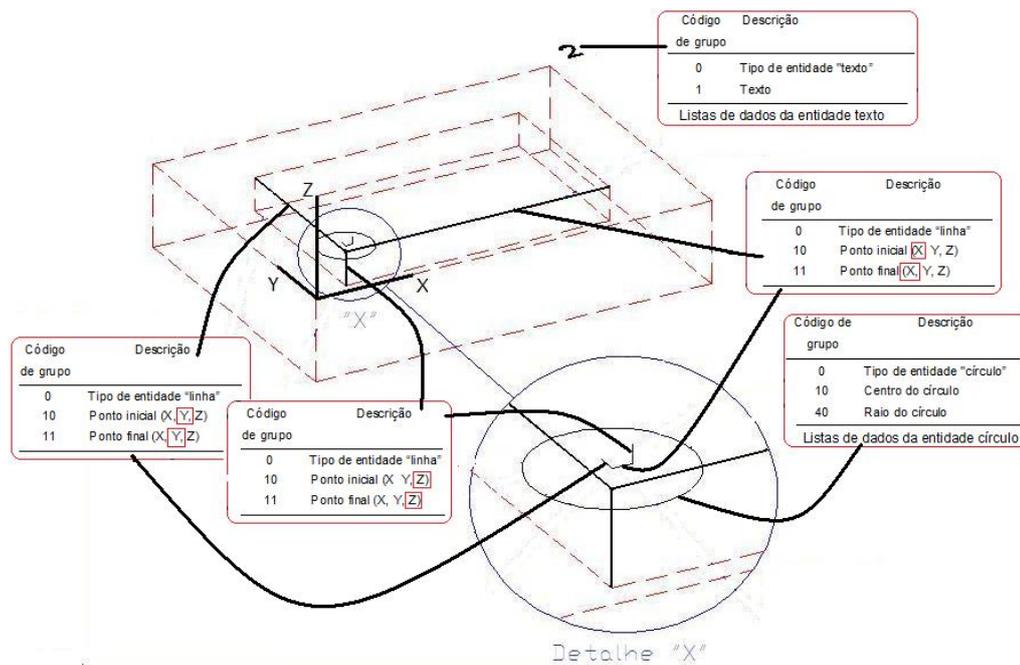


Figura 46 – Dados extraídos das entidades do ciclo G26

Se forem idênticos, essa entidade pertence a essa direção X, se não, pertence à direção Y. Após, é extraída a coordenada do ponto inicial e do ponto final de cada uma delas. As coordenadas do ponto inicial e final em relação ao ponto de referência são os valores inseridos no programa CN diretamente. A profundidade da cavidade extraída da linha possui os mesmos valores iniciais e finais na direção X e Y e tem seu valor Z inicial na superfície da cavidade, sendo o seu valor Z final inserido diretamente no programa. Para definir o sobremetal, extrai-se o comprimento das linhas que as definem nas respectivas direções. Essas linhas são desenhadas em um plano  $x,y$  diferente das linhas anteriores, facilitado assim a sua identificação. No caso dessa cavidade possuir um raio de canto, o sistema localiza um círculo desenhado no plano Z2 e, do banco de dados desse, extrai o valor do raio do círculo que é o mesmo da cavidade.

Os parâmetros largura e profundidade de corte são fixados em 50% e 25% do diâmetro da ferramenta, respectivamente. Os valores de avanço de corte no acabamento e de penetração da ferramenta são fixados em 80% e 20% do avanço de corte da ferramenta, respectivamente. Esses parâmetros podem ser modificados durante o processo de geração do programa CN desse ciclo, pois o sistema solicita a confirmação ou a inserção de novos valores.

### 4.3.3 Geração do programa do ciclo de resalto ao redor de uma área

A função desta rotina é gerar o programa CN do ciclo de usinagem G27, que é definido pelos seguintes blocos:

**T**(nº da posição da ferramenta no magazine) **M6**; (nome ferramenta)

**O**(nº do corretor da ferramenta) **S**(rotação) **M3**

**G0 X**(coordenada X do ponto inicial da cavidade) **Y**(coordenada Y do ponto inicial da cavidade)

**Z2.**

**G27 X**(coord. canto oposto da cavidade em X) **Y**(coord. canto oposto da cavidade em Y) **Z**(profundidade final da cavidade) **I**(sobremetal para acabamento em X) **J**(sobremetal para acabamento em Y) **K**(sobremetal para acabamento em Z) **U**(profundidade de corte) **W**(largura de corte ao longo dos eixos X e Y) **R**(raio de canto da cavidade) **F**(avanço de corte no desbaste) **H**(avanço de corte no acabamento) **V**(avanço de penetração da ferramenta)

**GZO M5**

As mesmas definições apresentados no ciclo G26 são aplicadas para esse ciclo, sendo que somente muda o nome do ciclo. A figura 47 apresenta quais dados são extraídos de cada entidade desenhada na peça e seus respectivos códigos.

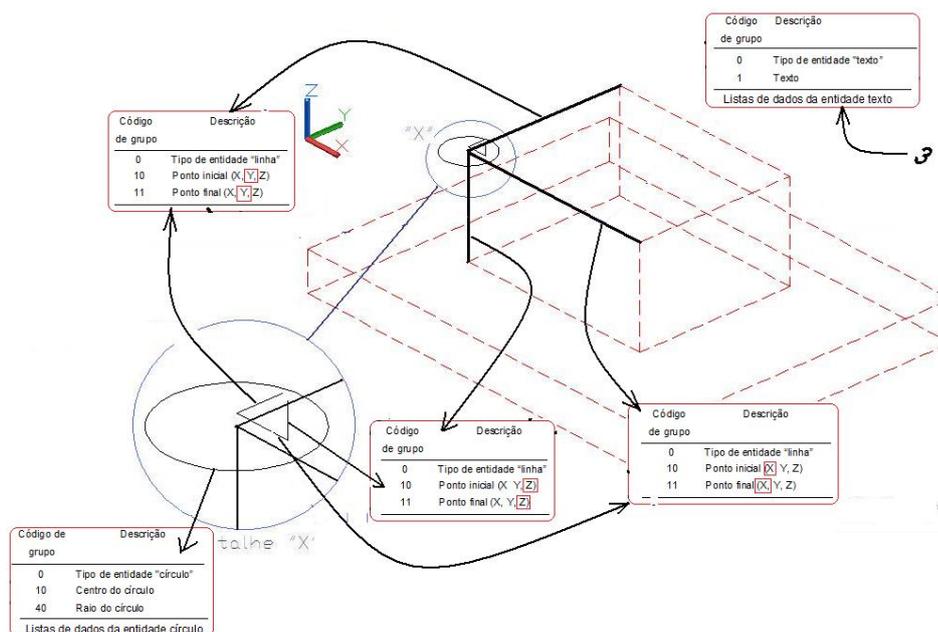


Figura 47 – Dados extraídos das entidades do ciclo G27

## 4.4 Estrutura funcional do aplicativo

### 4.3.1 Inicialização do aplicativo

O aplicativo proposto apresenta quatro ciclos de usinagem, conforme apresentado no item 3.3.1. A sua inicialização se dá através do menu do CAD, o qual contém um submenu composto por seis itens, ou através da linha de comando do CAD, bastando digitar o nome da função desejada.

O primeiro item do menu dá acesso ao banco de dados das ferramentas. Esse abre um arquivo editor de texto contendo uma lista com todas as ferramentas. Os quatro itens seguintes executam a programação dos ciclos de usinagem, conforme já apresentado. O último item executa a geração do programa CN (Figura 48).

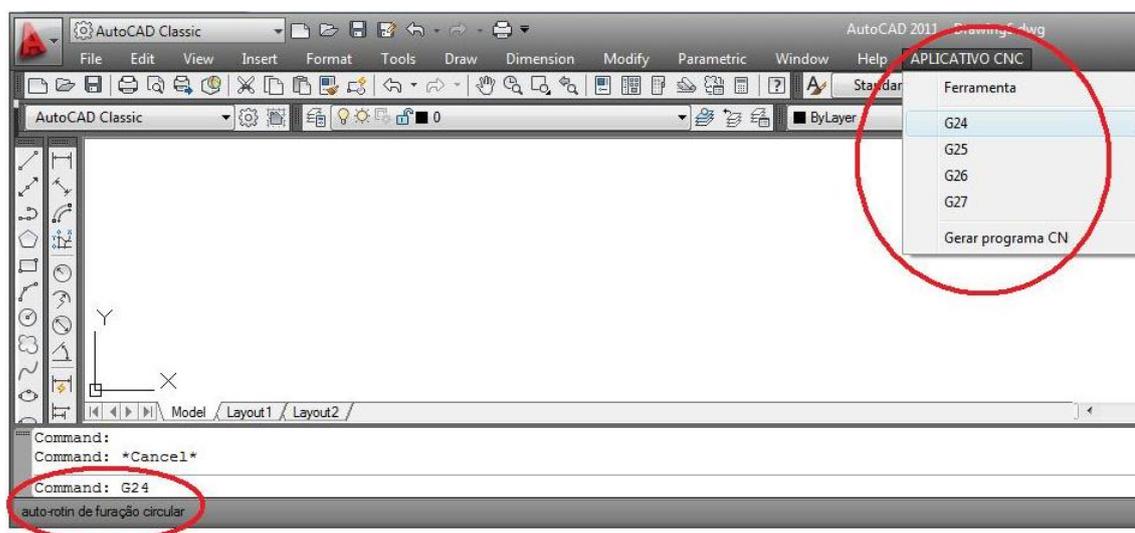
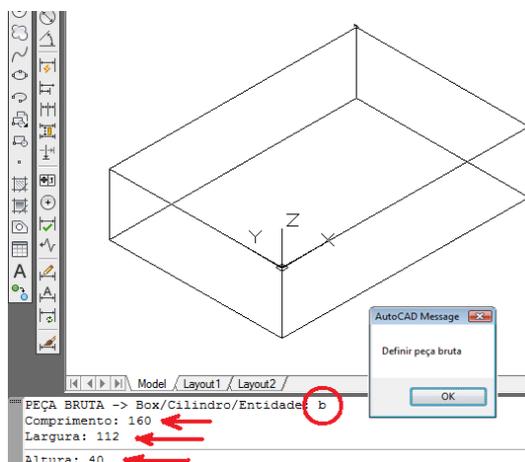


Figura 48 – Modo de acesso aos recursos do aplicativo

### 4.3.2 Definição da peça bruta

No caso de um novo programa criado, o sistema envia uma mensagem informando que será definida uma peça bruta. Para isso, uma das opções do tipo de

peça bruta deverá ser selecionada e a seguir as informações dimensionais da peça são informadas ao aplicativo, que fará o seu desenho (Figura 49).



**Figura 49 – Dados da peça bruta**

#### 4.3.3 Definição da ferramenta

Para cada tipo de operação de usinagem, tem-se uma ferramenta apropriada. Como exemplos podem-se citar a utilização de brocas na furação e a utilização de fresas no fresamento. Cada tipo de ferramenta tem suas características e promove um tipo de movimento específico durante a usinagem. Para a definição da ferramenta, o sistema envia uma mensagem de alerta deste procedimento. O programador deverá informar o tipo, o diâmetro e a posição da ferramenta no magazine da máquina. Os parâmetros “tipo” e “diâmetro” são usados para que o sistema possa localizar a existência dessa ferramenta no banco de dados do aplicativo. Se a ferramenta não for encontrada, outra mensagem é enviada informando que esta é inexistente e que deverá ser criada (figura 50-a). Os dados da nova ferramenta são solicitados pelo aplicativo, que depois de inseridos são salvos no banco de dados (Figura 50-b).

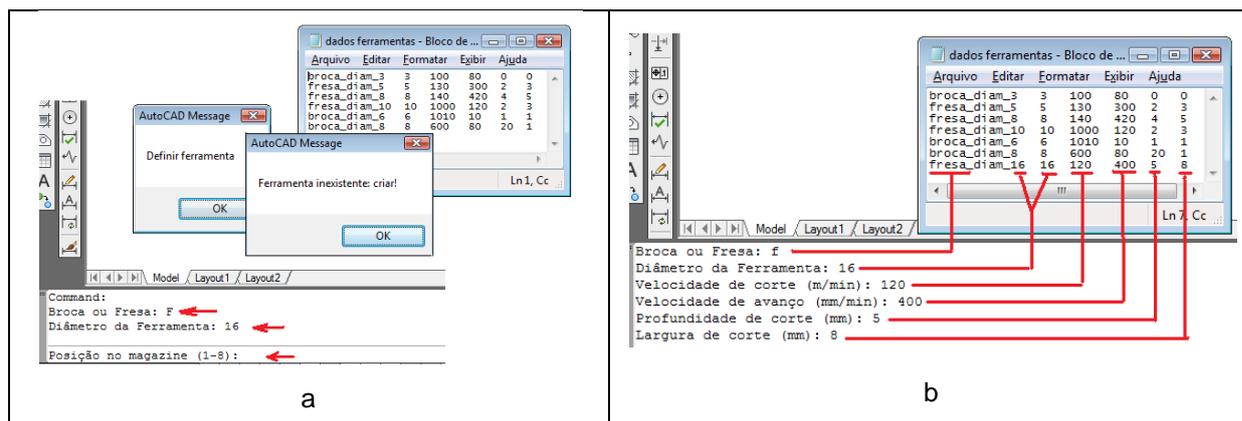


Figura 50 – Dados da ferramenta

#### 4.3.4 Dados geométricos do ciclo de usinagem

Os parâmetros de entrada no aplicativo de cada um dos ciclos são mostrados nas tabelas seguintes. Esses parâmetros referem-se à construção do desenho da geometria do ciclo de usinagem sobre a peça bruta, sendo também utilizados para a construção do desenho das entidades de armazenamento de dados. Esses são inseridos através da linha de comando do CAD. As tabelas a seguir apresentam quais são essas informações para cada ciclo.

Tabela 10 – Parâmetros para programação do ciclo de usinagem G24

<b>Ciclo de usinagem G24 – furação circular</b>	
Ponto de referência no plano superior da cavidade:	
Distância do centro do círculo na direção I:	
Distância do centro do círculo na direção J:	
Definir Ponto inicial por (1)-posição X,Y ou (2)-Cento e Raio ?-> 1/2:	
<i><b>Opção 1</b></i>	<i><b>Opção 2</b></i>
Definir a coordenada do ponto inicial no eixo X:	Definir o raio do círculo do ponto inicial R:
Definir a coordenada do ponto inicial no eixo Y:	Definir o ângulo do ponto inicial C:
Definir número total de furos L:	Definir número de furos L:
	Ângulo entre os furos inicial e final B:
Profundidade do furo Z:	

Tabela 11 – Parâmetros para programação do ciclo de usinagem G25

<b>Ciclo de usinagem G25 – furação linear</b>	
Ponto de referência no plano superior da cavidade:	
Distância do primeiro furo na direção X:	
Distância do primeiro furo na direção Y:	
Definir o número de furos ao longo do eixo X:	
Distância incremental ao entre os furos em X:	
Definir o número de furos ao longo do eixo Y:	
Distância incremental ao entre os furos em Y:	
Profundidade do furo Z:	

Tabela 12 – Parâmetros para programação do ciclo de usinagem G26

<b>Ciclo de usinagem G26 – cavidade retangular ou circular</b>	
Ponto de referência no plano superior da cavidade:	
CAVIDADE -> Retangular/Circular:	
<b>Opção R</b>	<b>Opção C</b>
Afastamento de referência na direção X:	Distância entre ponto de referência e centro da cavidade na direção X:
Afastamento de referência na direção Y:	Distância entre ponto de referência e centro da cavidade na direção Y:
Comprimento da cavidade (direção X):	Diâmetro da cavidade R:
Largura da cavidade (direção Y):	Profundidade Z:
Profundidade Z:	Sobremetal? <S>
Raio do canto <0>:	Sobremetal em X/Y:
Sobremetal? <S>	Sobremetal em Z:
Sobremetal em X:	
Sobremetal em Y:	
Sobremetal em Z:	

Tabela 13 – Parâmetros para programação do ciclo de usinagem G27

<b>Ciclo de usinagem G27 – resalto retangular ou circular ao redor de uma área específica</b>	
Ponto de referência no plano superior da cavidade:	
RESALTO -> Retangular/Circular:	
<b>Opção R</b>	<b>Opção C</b>
Afastamento de referência na direção X:	Distância entre ponto de referência e centro do resalto na direção X:
Afastamento de referência na direção Y:	Distância entre ponto de referência e centro do resalto na direção Y:
Comprimento do resalto (direção X):	Diâmetro do resalto R:
Largura do resalto (direção Y):	Profundidade Z:
Profundidade Z:	Sobremetal? <S>
Raio do canto <0>:	Sobremetal em X/Y:
Sobremetal? <S>	Sobremetal em Z
Sobremetal em X:	
Sobremetal em Y:	
Sobremetal em Z:	

#### 4.3.5 Geração do programa CN

A função de gerar o programa CN é o objetivo final do aplicativo. Ao escolher essa função no menu, o sistema identifica no sistema de *layer* do CAD a codificação criada para cada um dos ciclos de usinagem, busca no desenho as informações referente a cada um deles e gera o programa CN. O arquivo gerado é salvo em um arquivo editor de texto (Figura 51).

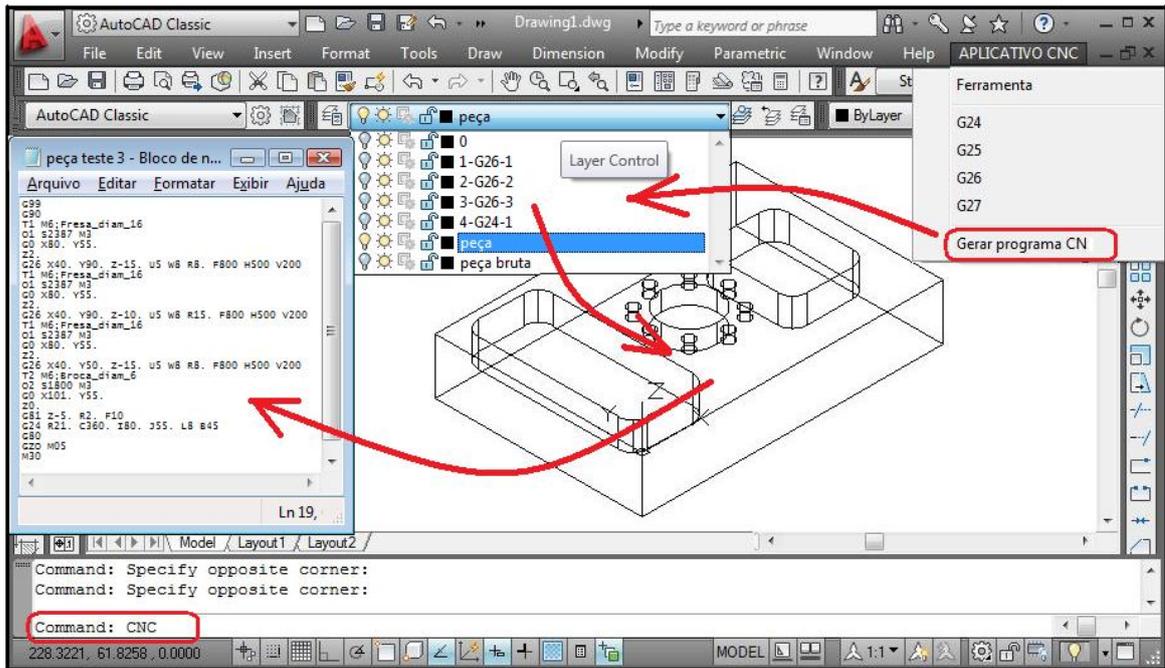


Figura 51 – Geração do programa CN

## 5 APLICAÇÃO E ANÁLISE

Para analisar a viabilidade da metodologia proposta, apresentar-se-á a elaboração do programa de usinagem para duas peças com o auxílio do aplicativo desenvolvido. São aplicados a elas os ciclos apresentados no desenvolvimento deste trabalho. As peças foram usinadas em um centro de usinagem equipado com comando MACH9, cedido por uma empresa local (Figura 52).



Figura 52 – Centro de usinagem utilizado no teste prático.

## 5.1 Peça teste 1

O desenho da peça teste 1 (Figura 53) apresenta uma cavidade retangular e outra circular, que serão programadas utilizando o ciclo de usinagem G26, e uma sequência circular de furos, para a qual se deve utilizar o ciclo G24.

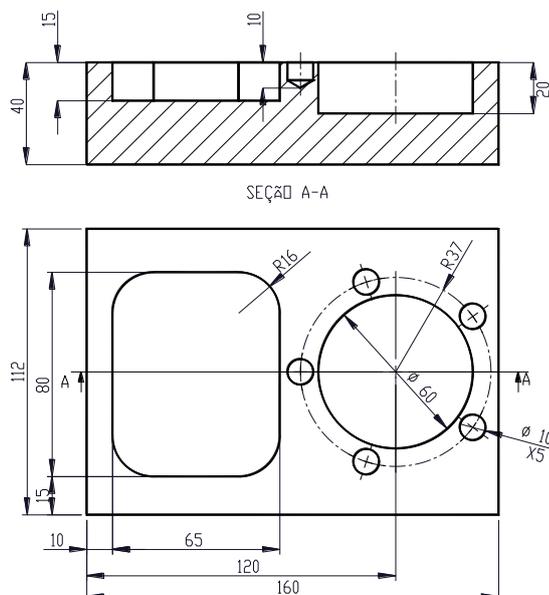


Figura 53 – Peça teste 1

### 5.1.1 Etapas de usinagem

Para a usinagem da peça 1, foram definidas as seguintes etapas:

- 1<sup>a</sup>. Usinagem em processo de desbaste e acabamento da cavidade retangular;
- 2<sup>a</sup>. Usinagem em processo de desbaste e acabamento da cavidade circular;
- 3<sup>a</sup>. Usinagem em processo de furação circular dos 5 furos.

A seguir são apresentados os parâmetros preenchidos na linha de comando do aplicativo que definiu a geometria da peça e o processo de usinagem.

## 1 - Usinagem em processo de desbaste e acabamento da cavidade retangular:

**Ciclo de usinagem:** G26

**Dados da Peça Bruta**

- Tipo de peça bruta (Box/Cilindro): **B**
- Comprimento: **160**
- Largura: **112**
- Altura: **40**

**Dados da Ferramenta**

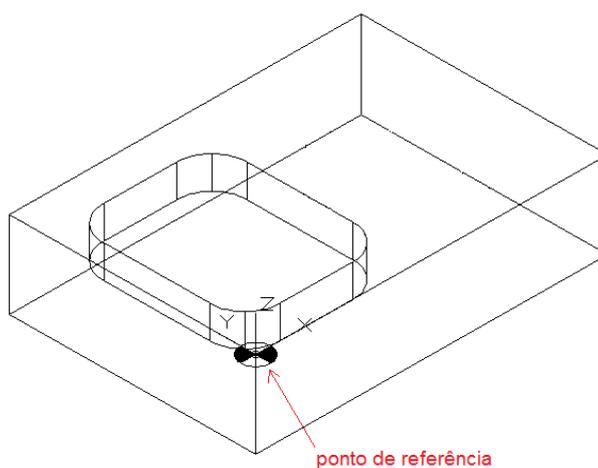
- Tipo de ferramenta (Broca/Fresa): **F**
- Diâmetro da ferramenta: **16**
- Posição da ferramenta no magazine (1-8): **1**

**Dados da cavidade**

- Ponto de referência no plano superior da cavidade: **ver figura 54**
- CAVIDADE -> Retangular/Circular: **R**
- Afastamento de referência na direção X: **10**
- Afastamento de referência na direção Y: **15**
- Comprimento da cavidade (direção X): **65**
- Largura da cavidade (direção Y): **80**
- Profundidade Z: **15**
- Raio do canto <0>: **16**

**Dados do processo**

- Sobremetal? <S>: **S**
- Sobremetal em X: **1**
- Sobremetal em Y: **1**
- Sobremetal em Z: **0**



**Figura 54 – Desenho do ciclo 26 retangular**

## 2 - Usinagem em processo de desbaste e acabamento da cavidade circular:

**Ciclo de usinagem:** G26

**Dados da Ferramenta**

- Tipo de ferramenta (**B**roca/**F**resa): **F**
- Diâmetro da ferramenta: **16**
- Posição da ferramenta no magazine (1-8): **1**

**Dados da cavidade**

- Ponto de referência no plano superior da cavidade: **ver figura 55**
- CAVIDADE -> Retangular/Circular: **C**
- Distância entre ponto de referência e centro da cavidade na direção X: **160**
- Distância entre ponto de referência e centro da cavidade na direção Y: **56**
- Diâmetro da cavidade R: **60**
- Profundidade Z: **20**

**Dados do processo**

- Sobremetal? <S>: **S**
- Sobremetal em X: **1**
- Sobremetal em Y: **1**
- Sobremetal em Z: **0**

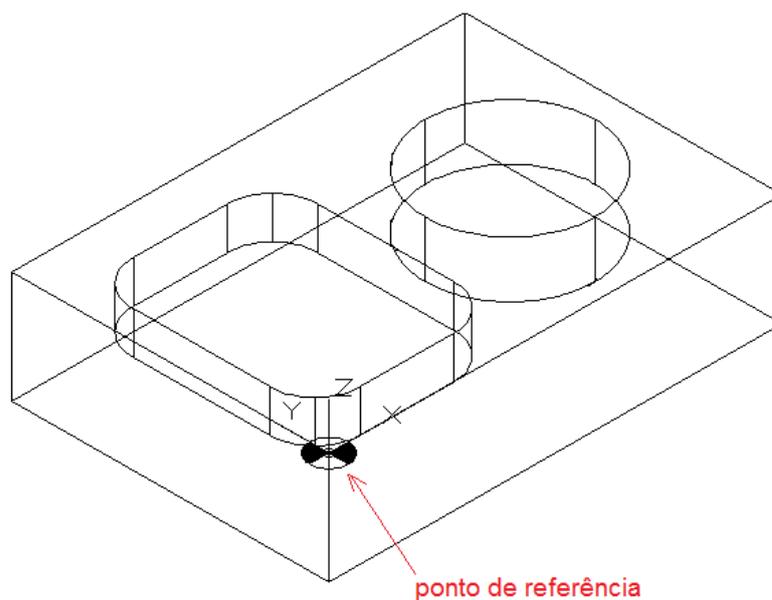


Figura 55 – Desenho do ciclo 26 circular

### 3 - Usinagem em processo de furação:

**Ciclo de usinagem:** G24

**Dados da Ferramenta**

- Tipo de ferramenta (**B**roca/**F**resa): **B**
- Diâmetro da ferramenta: **10**
- Posição da ferramenta no magazine (1-8): **2**

**Dados da cavidade**

- Ponto de referência no plano superior da cavidade: **ver figura 56**
- Distância do centro do círculo na direção I: **120**
- Distância do centro do círculo na direção J: **56**
- Definir Ponto inicial por (1)-posição X,Y ou (2)-Cento e Raio ?-> 1/2: **1**
- Definir a coordenada do ponto inicial no eixo X: **83**
- Definir a coordenada do ponto inicial no eixo Y: **56**
- Definir número total de furos L: **5**

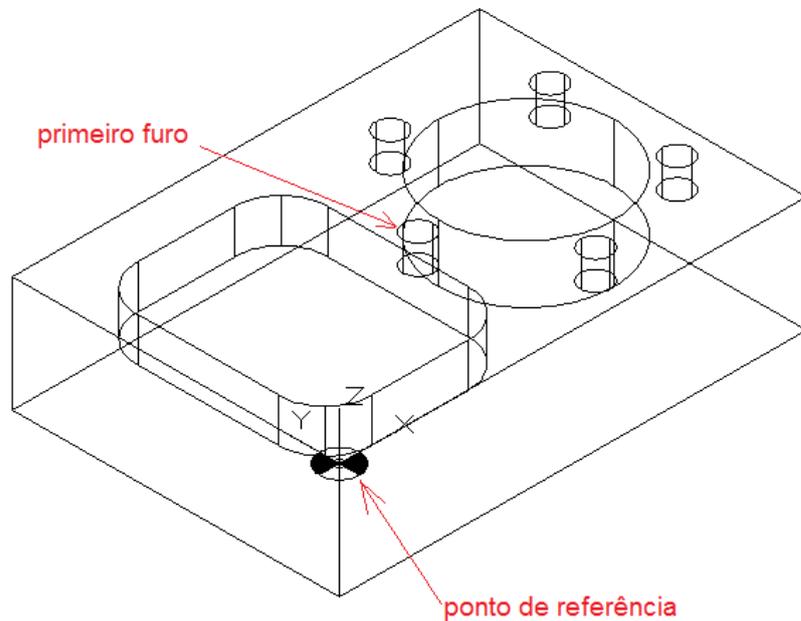


Figura 56 – Desenho do ciclo 24

### 5.1.2 Geração do programa CN

A figura 57 apresenta o programa CN gerado para a peça teste 1.

```
peça teste 1 - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda

G99
G90

T1 M6; Fresa_diam_16
O1 S2387 M3
G0 X10. Y15.
Z2.
G26 X65. Y80. Z-15. I1. J1. K0. U5. W8. R16. F600 H500 V5
Z2.

T1 M6; Fresa_diam_16
O1 S2387 M3
G0 X120. Y56.
Z2.
G26 Z-20. I1. J1. K0. U8. W8. R30. F400 H20 V10
Z2.

T2 M6; Broca_diam_10
O2 S955 M3
G0 X83. Y56.
Z5.
G81 Z-10. R2. F80
G24 R37. C180. I120. J56. L5. B72.
G80
Z5.

GZO M05
M30

Ln 7, Col 44
```

**Figura 57 – Programa CN da peça teste 1**

### 5.1.3 Usinagem da peça teste

Para a comprovação do resultado do processamento para o exemplo proposto, o programa CN gerado pelo aplicativo foi aplicado para a usinagem da peça teste 1. O resultado da usinagem para cada um dos processos é mostrado na figura 58.

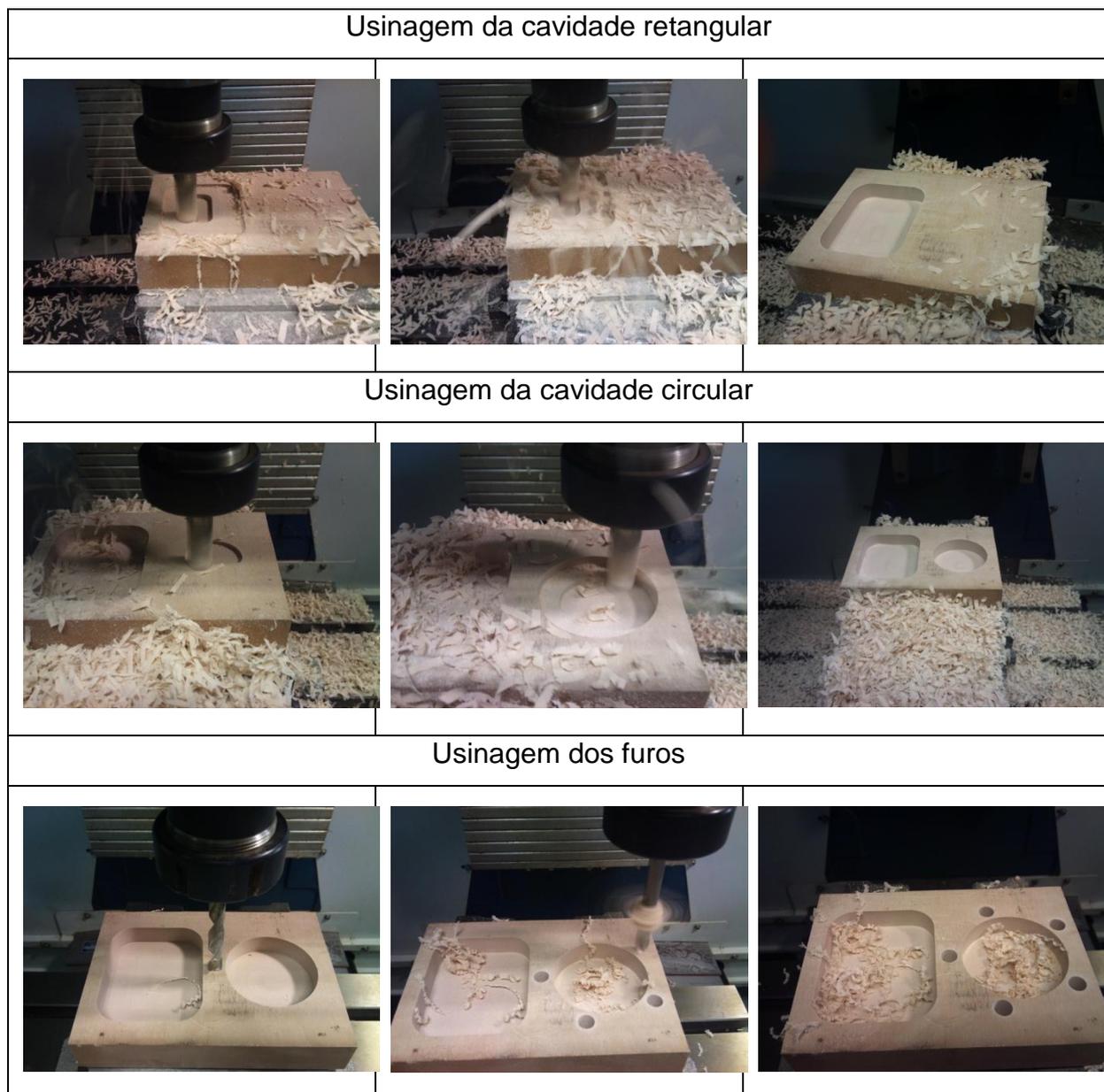
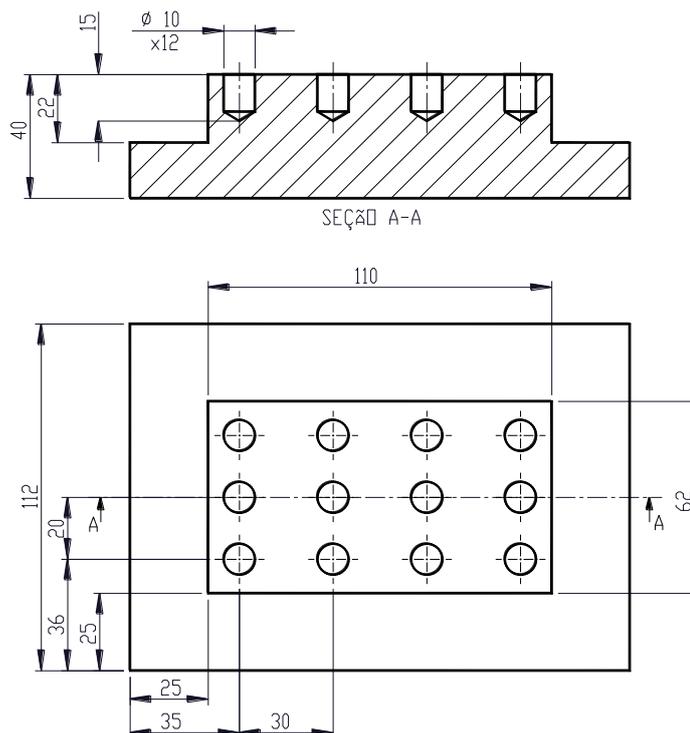


Figura 58 – Sequência de usinagem da peça teste 1

## 5.2 Peça teste 2

O desenho da peça teste 2 (Figura 59) apresenta um resalto retangular que será programado utilizando o ciclo de usinagem G27, e uma sequência linear de furos, na qual deve ser utilizado o ciclo G25.



**Figura 59 – Peça teste 2**

### 5.2.1 Etapas de usinagem

Para a usinagem da peça 2 foram definidas as seguintes etapas:

- 1<sup>a</sup>. Usinagem em processo de desbaste e acabamento do resalto retangular ao redor de uma área;
- 2<sup>a</sup>. Usinagem em processo de furação linear.

A seguir são apresentados os parâmetros preenchidos na linha de comando do aplicativo que definiu a geometria da peça e o processo de usinagem.

#### 1 - Usinagem em processo de desbaste e acabamento do resalto retangular ao redor de uma área:

**Ciclo de usinagem:** G27

**Dados da Peça Bruta**

- Tipo de peça bruta (**B**ox/**C**ilindro): **B**
- Comprimento: **160**

- Largura: **112**
- Altura: **40**

#### Dados da Ferramenta

- Tipo de ferramenta (Broca/Fresa): **F**
- Diâmetro da ferramenta: **16**
- Posição da ferramenta no magazine (1-8): **1**

#### Dados da cavidade

- Ponto de referência no plano superior da cavidade: **ver figura 60**
- RESALTO -> Retangular/Circular: **R**
- Afastamento de referência na direção X: **25**
- Afastamento de referência na direção Y: **25**
- Comprimento da cavidade (direção X): **110**
- Largura da cavidade (direção Y): **62**
- Profundidade Z: **22**
- Raio do canto <0>: **0**

#### Dados do processo

- Sobremetal? <S>: **S**
- Sobremetal em X: **1**
- Sobremetal em Y: **1**
- Sobremetal em Z: **0**

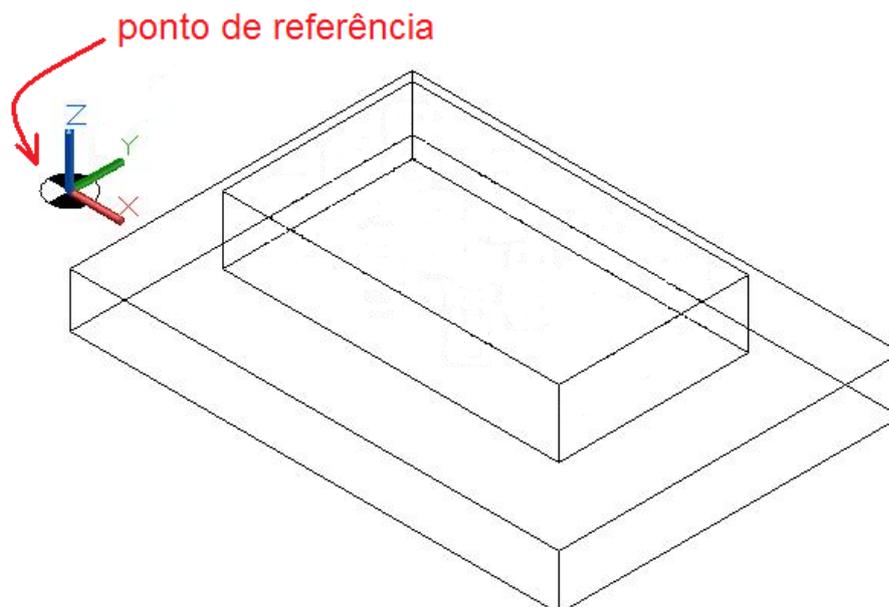


Figura 60 – Desenho do ciclo 27

## 2 - Usinagem em processo de furação:

**Ciclo de usinagem:** G24

**Dados da Ferramenta**

- Tipo de ferramenta (**B**roca/**F**resa): **B**
- Diâmetro da ferramenta: **10**
- Posição da ferramenta no magazine (1-8): **2**

**Dados da cavidade**

- Ponto de referência no plano superior da cavidade: **ver figura 61**
- Distância do primeiro furo na direção X: **10**
- Distância do primeiro furo na direção Y: **11**
- Definir o número de furos ao longo do eixo X: **30**
- Distância incremental ao entre os furos em X: **4**
- Definir o número de furos ao longo do eixo Y: **20**
- Distância incremental ao entre os furos em X: **3**
- Profundidade do furo Z: **15**

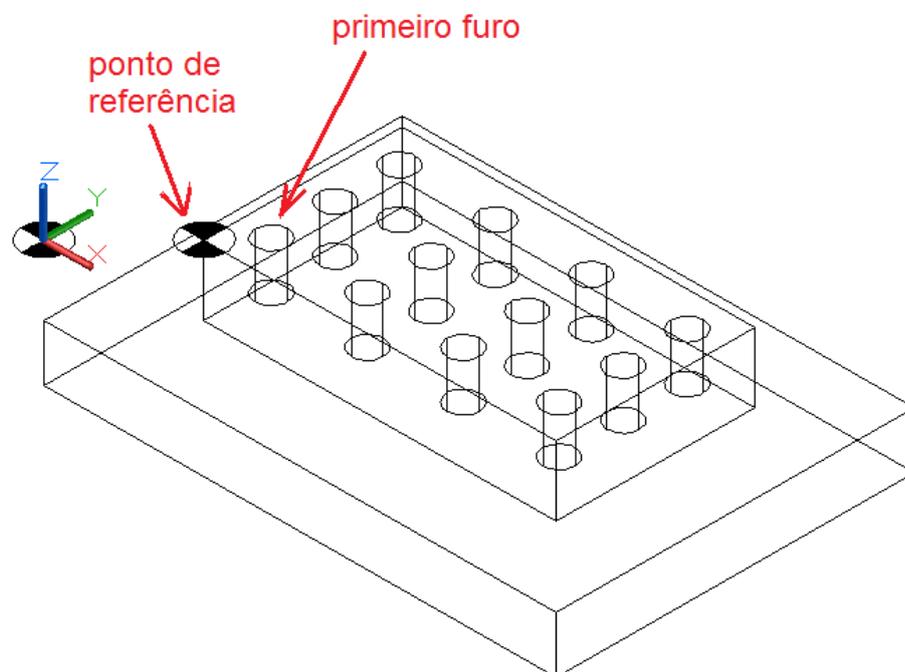


Figura 61 – Desenho do ciclo 25

### 5.2.2 Geração do programa CN

A figura 62 apresenta o programa CN gerado para a peça teste 2.

```
peça teste 2 - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
G99
G90
T1 M6;Fresa_diam_16
O1 S2387 M3
G0 X25. Y25.
Z2.
G27 X110. Y62. Z-22. I1. J1. K0. U5. W8. F-600 H-500 V200
GZO M05
T2 M6;Broca_diam_10
O2 S955 M3
G0 X35. Y36.
Z2.
G81 Z-15. R2. F80
G25 X30. Y20. I4 J3
G80
GZO M05
M30
Ln 7, Col 44
```

**Figura 62 – Programa CN da peça teste 2**

### 5.2.3 Usinagem da peça teste.

Para a comprovação do resultado do processamento para o exemplo proposto, o programa CN gerado pelo aplicativo foi aplicado para a usinagem da peça teste 2. O resultado da usinagem para cada um dos processos é mostrado na figura 63.



Figura 63 – Sequência de usinagem da peça teste 2

### 5.3 Análise

A aplicação prática da metodologia proposta permitiu avaliar a eficiência e a viabilidade do seu uso no chão de fábrica. O resultado mostrou-se satisfatório, atendendo ao objetivo esperado. Os benefícios constatados com a aplicação desse aplicativo foram:

- Possibilidade de gerar programa CN sem usar o painel de comando da máquina;
- Visualização gráfica de como ficará a peça após sua usinagem;
- O fato de que o programa CN, após ser carregado no comando da máquina, não precisa ser simulado;
- Programas gerados contêm poucos blocos pelo fato de usarem os ciclos disponíveis na máquina;
- Geração de um sistema interativo de fácil utilização, possibilitando o rápido aprendizado por parte do programador.

Não foram aplicados a este sistema todos os recursos disponíveis nos ciclos de máquina estudados, uma vez que o objetivo era somente demonstrar a sua viabilidade como alternativa para um sistema de programação assistida por computador. Portanto, como o sistema CAD no qual foi desenvolvida essa metodologia é um sistema aberto, há a possibilidade de aplicação desses recursos e de outros ciclos e rotinas de usinagens existentes nas mais diversas máquinas CNC.

## 6 CONCLUSÃO

A metodologia proposta nesse trabalho foi analisada através da implementação de um aplicativo em sistema CAD, com o objetivo de desenvolver um sistema dedicado à integração CAD/CAM. Essa aplicação prática permitiu uma melhor avaliação da técnica adotada, pois puderam ser identificadas as vantagens e desvantagens do sistema em comparação com sistemas CAD/CAM comerciais.

Como vantagens, podem-se citar: fácil e rápida operação do aplicativo, independência do painel do comando da máquina para a geração do programa CN, visualização gráfica em modelo tridimensional da peça final e uso direto dos ciclos de usinagem da máquina. Como desvantagens, têm-se: gamas de geometrias limitadas aos ciclos disponíveis na máquina, dependência de um sistema CAD e necessidade de domínio da técnica de desenvolvimento de aplicativos para automatizar procedimentos em ambiente CAD. No entanto, a possibilidade de associar essa técnica ao uso de ferramentas disponíveis em Comandos CNC modernos, como a programação parametrizada, para o desenvolvimento de novos ciclos de usinagem, permite aumentar ainda mais a potencialidade e flexibilidade do sistema produtivo.

É possível aplicar esta metodologia a outras linguagens de programação CNC como, por exemplo, aos comandos Siemens, Fanuc, Fagor etc. Nesse caso, é necessário adaptar o modo de interpretação das informações requeridas para a criação dos Ciclos de usinagem.

De modo geral, pode-se comprovar que o uso dos recursos de programação dos sistemas CAD traz benefícios às diversas áreas da manufatura. A falta de conhecimento desses recursos torna os sistemas subutilizados, o que impede o uso de toda a sua gama de possibilidades, que auxiliariam na redução de tempos e custos nas empresas, principalmente na área de programação de máquinas CNC. O investimento em sistemas próprios para a geração automática de programas CN pode ser uma alternativa viável ao uso de *softwares* comerciais. Isso é especialmente válido para pequenas e médias empresas, pois a aquisição e manutenção desses sistemas podem representar um custo significativamente alto.

Como sugestões para estudos futuros, relacionados a essa metodologia, têm-se:

- Análise da viabilidade da introdução de novos ciclos para a aplicação da técnica em outros comandos de máquina CNC;
- Implantação dessa metodologia em sistemas CAD livre;
- Desenvolvimento de um simulador de trajetória de ferramenta que interprete os ciclos de usinagem;
- Criação de um método que torne possível a interpretação de desenho digital para identificação de ciclos de usinagem.

## REFERÊNCIAS BIBLIOGRÁFICAS

ÁLVARES, A. J.; FERREIRA, J. C. E. **Uma Nova Abordagem de CAPP Para Peças Baseada em Mapeamento de Features**. In: 3º CONGRESSO NACIONAL DE ENGENHARIA (CONEM). Anais. Belém, PA, 2004, 13p.

BESANT, C. B. **CAD/CAM - Projeto e Fabricação com o Auxílio de Computador**. 3ª ed. Rio de Janeiro: Campus, 1988. 249p

BROWNE, J.; et al. **The development of a FMS design procedure**. In: STEEKE, K.; SURI, R.(Ed.). Proceedings of the First ORSA/TIMS Special Interest Conference on FMS. Michigan: Ann Harbor, 1984.

CASSANIGA, F. A. **Fácil Programação do Controle Numérico**. São Paulo: F.A.C , 2000, 310p.

COSTA, D. D.; PEREIRA, A. G. **Programação CNC Baseada em Features**. In: 3º CONGRESSO BRASILEIRO DE ENGENHARIA DE FABRICAÇÃO, 2003, Uberlândia, SP. Anais do II COBEF, 2003. v1

CUNHA, J. G.; et al. **Computação Gráfica e sua Aplicação em CAD. Introdução e Padronização**. São Paulo: Atlas 1987, 199p.

DJASSEMI, M. A. **A Parametric Programming Technique For Efficient CNC Machining Operations**. Computers and Industrial Engineering, v35, n.1-2, 1998, p33-36.

DOSSAN, 2010, **Manual de Programação Comando Fanuc – DNM\_400**.

FERREIRA, J. C. E.; VIVIAN, D. **Reconhecimento de Features em Peças Rotacionais Modeladas por Sólidos num Sistema CAD/CAM**. 2006 Revista Máquina e Metais, Aranda São Paulo, v42, n° 480, p.52-73, 2006

FILHO, E. R. **Sistema Integrado de Manufatura**. Apostila do curso de engenharia de produção - UFMG, 2004.

GIACAGLIA, M. E. **A organização da informação em sistemas CAD: análise crítica de esquemas existentes e proposta para o caso brasileiro**. Disponível em:

<[http://www.usp.br/fau/docentes/deptecnologia/m\\_giacaglia/nlayers.pdf](http://www.usp.br/fau/docentes/deptecnologia/m_giacaglia/nlayers.pdf)>. Acesso em: 05 dez. 2011.

GIBBS, D. **CNC Part Programming: A Practical Guide**. London: Cassell Publishers Limited, 1994, 186p.

GONÇALVES, M. A. F. **Um estudo sobre implementação de ciclos de usinagem através de programação parametrizada em máquinas de comando numérico computadorizado**. . 2007. 91p. Dissertação de Mestrado em Engenharia de Produção - Universidade Federal de Santa Maria - UFSM, Santa Maria, 2007.

GROOVER, M. P., ZIMMERS JR., E. W. **CAD/CAM Computer Aided Design and Manufacturing**. New Jersey: Prentice-Hall, Englewood Cliffs, 1984, 489p.

GROOVER, M. P. **Automação Industrial e Sistemas de Manufatura**. 3ª ed. São Paulo: Pearson Education, 2011, 581p.

KANITA, F. P. **Análise do Desenvolvimento dos Sistemas CAD/CAE/CAM no Brasil nos Diversos Setores do Conhecimento Sob a Ótica da Propriedade Industrial**. 2005. 178p. Dissertação de Mestrado em Tecnologia - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, Rio de Janeiro, 2005.

KRAMER, W.; KRAMER, D. **Programando em AutoLISP**. São Paulo: Makron books, 1995, 274p.

DE LUCCA, R. T. **Avaliação de programas CADD no Setor de projeto Arquitetônico: Etapas Legais de Projeto**. 1999. 166p. Dissertação de Mestrado em arquitetura - Universidade de São Paulo, São Carlos, 1999.

MACHADO, A. **Comando Numérico Aplicado às Máquinas-Ferramentas**. 4ª ed. São Paulo: Ícone, 1990, 461p.

MAGALHÃES, J. S. F. **Sistema Variante para Seleção de Máquinas-ferramenta**. 2008. 71p. Dissertação de Mestrado em Engenharia Mecânica -Universidade Federal do Paraná, Curitiba, 2008.

MATTSON, M. **CNC Programming - Principles and Applications**. 2ª ed. New York: Cengage learning, 2009, 405 p.

MCCMAHON, C., BROWNE, J. **CAD/CAM Principles Practice and Manufacturing Management**. 2ª ed. Boston: Addison-Wesley Longman Publishing Co. 1998, 93p.

MENEGHELLO, G. P. **Aplicação de um Sistema Robótico Utilizando Recursos de sistema CAD/CAM para o Processo de Fresamento**. 2003. 121p. Dissertação de Mestrado em Engenharia Mecânica - Universidade Federal do Rio Grande do Sul - UFRGS, Porto Alegre, 2003.

MORGAN J. **Técnicas de Segmentação de Imagens na geração de Programas para Máquina de Comando Numérico**. 2008. 93p. Dissertação de Mestrado em Engenharia da Produção - Universidade Federal de Santa Maria, Santa Maria, 2008.

MUNDO CNC. **Conceitos Avançados. Programação Parametrizada**. Disponível em: <http://www.mundocnc.com.br/avan2.php>>. Acesso em: 25 nov. 2011.

PEREIRA, A. G. **Desenvolvimento e Avaliação de um Editor para Programação CN em Centros de Usinagem**. 2003. 122p. Dissertação de Mestrado em Engenharia Mecânica - Universidade Federal do Paraná – UFPR, Paraná, 2003.

QUEIROZ, A. A.; STEMMER, C. **Cenário da Programação CN - Da Programação Manual ao CAD/CAM**. In: SIMPÓSIO SOBRE CAD/CAM - REVISÃO DA SITUAÇÃO BRASILEIRA, 1986. São Paulo. *Anais*. SOBRACOM, 1986.

ROMI, 2006, **Manual de Programação comando Siemens 810D – Discovery 560**.

ROMI, 1999, **Manual de Programação comando Mach9 – Discovery 4022**.

RUSCHEL, R.C. **Programando em AutoLISP** – Notas de aula. Disponível em: <<http://www.fec.unicamp.br/~regina/alisp0.html>>. Acesso em: 29 nov. 2011.

SILVA, S. D. **CNC Programação de Comando Numérico Computadorizado – Torneamento**. 8ª ed. São Paulo: Érica, 2008, 312p.

SMID, P. **CNC Programming Handbook**. 2ª ed. New York: Industrial Press, 2003, 529p.

SOUZA, D. J. A. **Técnicas de Implementação de Ciclos Expandidos e Avançados de Usinagem para Máquinas com Comando Numérico**. 2004. 113p. -

Dissertação de Mestrado em Engenharia da Produção - Universidade Federal de Santa Maria, Santa Maria: 2004.

SOUZA, A. F.; COELHO, R. T. **Tecnologia CAD/CAM - Definições e Estado da Arte Visando Auxiliar sua Implantação em um Ambiente Fabril.** In: XXIII Encontro Nacional de Engenharia de Produção (ENECEP). Anais... Ouro Preto, MG, 2003, 8p.

TAUILE, J. R. **A Difusão de Máquinas-Ferramenta com Controle Numérico no Brasil e Algumas Implicações para o Desenvolvimento Econômico.** Ciência, Tecnologia e Desenvolvimento, Brasília, n.2, p681-704, 1985.

TEICHOLZ, E. **CAD/CAM HANDBOOK.** New York: McGraw-hill, 1985, 400p.

Wang, H.P.; Li, J.K., **Computer-Aided Process Planning,** Advances in Industrial Engineering, Vol. 13, Elsevier, 1991

ZINDULIS, Y.; TRINDADE, I. M.; BITTENCOURT, W. S.; SILVA, A. D. **Geração Automática de Rasgos Radiais para Máquinas CNC.** In: XII Congresso Nacional de Engenharia Mecânica (CREEM). Anais... Ilha Solteira, SP, 2005, 2p.