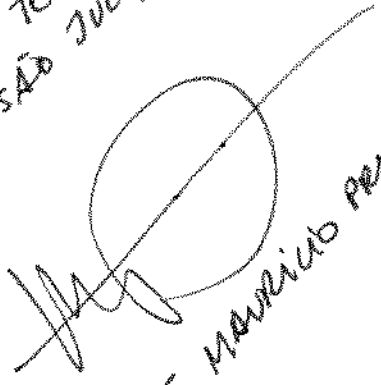


S.P.A.T. SISTEMA DE PROGRAMAÇÃO AUTOMÁTICA
DE TORNOS

Autor: Geraldo Nonato Telles

01/90

ESTE EXEMPLAR CORRESPONDE À
REDACÇÃO FINAL DA TESE DEFENDIDA
POR GERALDO NONATO TELLES E
APROVADO PELA COMISSÃO JULGADORA
EM 30/01/90


PROF. MAURÍCIO PRATES

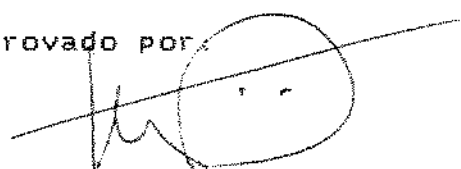
UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE ENGENHARIA DE MATERIAIS

Tese de : Doutorado

Título da Tese: SPAT - Sistema de Programação
Automática de Tornos

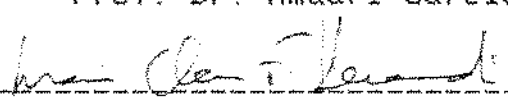
Autor: Geraldo Nonato Telles

Aprovado por:

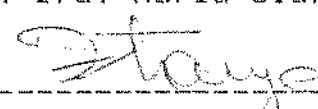


Prof. Dr. Mauricio Prates de Campos Filho

Prof. Dr. Amauri Garcia



Prof. Dra. Maria Clara F. Ierardi



Prof. Dr. Plínio Stange

Prof. Dr. Paulo Roberto Mei

Campinas, 30 de Janeiro de 1990

AGRADECIMENTOS

Aos professores Maurício Prates de Campos Filho e Amauri Garcia, pela orientação e incentivo, sem os quais este trabalho não existiria.

A Maria Clara Ierardi e Lázaro Coimbra, pelas sugestões, pelo interesse e apoio oferecidos durante o trabalho.

Aos funcionários do NAI, Aleksander e Ilma pelo apoio estrutural.

A Aurélio Pires Neto, pelo esmero e eficiência na confecção dos desenhos.

S.P.A.T. - Sistema de Programação Automática de Tornos.

RESUMO

Da mesma maneira que a programação científica em geral, a abordagem de programação para Sistemas Produtivos Automatizados pode assumir duas formas:

- Programação Manual (de baixo nível).
- Programação Automática (de alto nível).

Como ferramenta para programação de MFCN, desenvolveu-se o SPAT, que é composto por 3 módulos: EDITOR, SIMULADOR GRÁFICO e PROCESSADOR GEOMÉTRICO, sendo que o sistema "roda" em micro-computadores do tipo IBM PC Xt.

A linguagem utilizada foi o Turbo Pascal e os 3 módulos requerem, para sua utilização, 640 K de memória RAM e duas unidades de disco flexível (5 1/4" - 360 K).

O Sistema utiliza Macro-Instruções de forma que, ao programador CN, resta a tarefa de programar apenas o perfil final da peça, sendo que o Processador Geométrico "gera" a parte do programa relativa ao detalhamento das operações de desbaste.

A aplicação do SPAT em ambiente acadêmico possibilitou:

- a) Cursos de CN em entidades que não possuem Máquinas-Ferramenta CN, com o Simulador Gráfico fazendo as vezes de "laboratório", viabilizando a verificação de programas escritos pelos alunos;
- b) Caracterização das duas formas de programação: Manual e Automática.

A utilização do SPAT em ambientes industriais evidenciou:

- a) Redução no tempo de programação, uma vez que o programador preocupa-se apenas com o acabamento;
- b) Redução no tempo de preparação das MFCN, devido à depuração prévia dos programas, através da utilização criteriosa do Simulador Gráfico, com eventuais problemas, sendo detectados antes do envio dos mesmos para a Oficina Mecânica.

S.P.A.T. - Sistema de Programação Automática de Tornos.

ABSTRACT

Like scientific programming, there are two ways to programming Manufacturing Systems:

- Manual Programming (low level);
- Automatic Programming (high level).

A programming tool was developed for Numerical Control, composed of 3 pieces: EDITOR, GRAPHICS SIMULATOR and GEOMETRIC PROCESSOR. The system runs in IBM PC Xt (micro-computers).

The language used was Turbo Pascal, and the 3 pieces need 640 K RAM and 2 floppy-disks (5 1/4" - 360 K).

The System uses Macro-Instructions so, that, the NC programmer only: has to specify the finishing operation of a part, and the GEOMETRIC PROCESSOR makes the detailing of the rough operations.

The SPAT's utilization for education makes possible:

- a) NC courses in entities which don't have Numerical Control Machines, with the GRAPHICS SIMULATOR working as "laboratory" for "try-out" in students programming exercises;
- b) Characterization of the 2 ways of NC programming: Manual and Automatic.

The SPAT's utilization in industrial plants shows:

- a) Time reduction in programming, since the programmer works only with the finishing operation;
- b) Time reduction to get the NC machines ready, because the previous optimization of the programs, by the correct using of the GRAPHICS SIMULATOR, with the eventuals problem detected before they are send to the workshop.

GLOSSÁRIO DE SIGLAS

Optou-se, nesta dissertação, por manter-se algumas siglas, no padrão internacionalmente aceito.

APT - Automatically Programmed Tools

CAD - Computer Aided Design

CAM - Computer Aided Manufacturing

CN - Controle Numérico

CNC - Controle Numérico Computadorizado

FORTTRAN - FORMula TRANslation

MF - Máquina-Ferramenta

MFCN - Máquina-Ferramenta a Controle Numérico

PC - Personal Computer

SPAT - Sistema de Programação Automática de Tornos

ÍNDICE

CAPÍTULO 1 - INTRODUÇÃO

1.1 - CONSIDERAÇÕES INICIAIS	1
1.2 - OBJETIVOS DESSE DESENVOLVIMENTO	4
1.3 - MÓDULOS COMPONENTES DO SPAT	6
REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 1)	8-10

CAPÍTULO 2 - LINGUAGENS COMPUTACIONAIS UTILIZADAS EM AMBIENTE DE MANUFATURA INDUSTRIAL

2.1 - GENERALIDADES	11
2.2 - FORMAS BÁSICAS DE PROGRAMAÇÃO EM AMBIENTE INDUSTRIAL	12
2.3 - GÊNESIS	13
2.4 - MODUS-OPERANDI DE UMA MÁQUINA FERRAMENTA CNC	16
2.5 - LINGUAGENS DE PROGRAMAÇÃO ESPECÍFICAS PARA CN	17
2.6 - JUSTIFICATIVA PARA O DESENVOLVIMENTO DE UM PROCESSADOR DE GEOMETRIA	18
2.7 - EQUIPAMENTO E LINGUAGEM UTILIZADOS NO DESENVOLVIMENTO	19
2.8 - COMPARAÇÕES DE CUSTOS ENTRE AS DIFERENTES TÉCNICAS DE PROGRAMAÇÃO	19
2.9 - METAS PRINCIPAIS DO PROJETO	20
2.10 - MÓDULOS E RECURSOS OPERACIONAIS DO SPAT	21
REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 2)	27-28

CAPÍTULO 3 - CICLOS AUTOMÁTICOS DE PROGRAMAÇÃO OFERECIDOS PELO PROCESSADOR DE GEOMETRIA

3.1 - GENERALIDADES	29
3.2 - CICLO DE DESBASTE LONGITUDINAL (G100). DESCRIÇÃO E FORMATO	30
3.2.1 - CICLO DE DESBASTE LONGITUDINAL (G100). ALGORITMOS E	

ENFOQUE COMPUTACIONAL	33
3.3 - CICLO DE DESBASTE TRANSVERSAL (G101). DESCRIÇÃO E FORMATO	35
3.3.1 - CICLO DE DESBASTE TRANSVERSAL (G101). ALGORITMOS E ENFOQUE COMPUTACIONAL	37
3.4 - CICLO DE DESBASTE DE CONTORNOS (G102). DESCRIÇÃO E FORMATO	39
3.4.1 - CICLO DE DESBASTE DE CONTORNOS (G02). ALGORITMOS E ENFOQUE COMPUTACIONAL	41
3.5 - CICLO DE DESBASTE LONGITUDINAL TRAPEZOIDAL (G103). DESCRIÇÃO E FORMATO	43
3.5.1 - CICLO DE DESBASTE LONGITUDINAL TRAPEZOIDAL (G103). ALGORITMOS E ENFOQUE COMPUTACIONAL	45
3.6 - CICLO DE DESBASTE TRANSVERSAL TRAPEZOIDAL (G104). DESCRIÇÃO E FORMATO	46
3.6.1 - CICLO DE DESBASTE TRANSVERSAL TRAPEZOIDAL (G104). ALGORITMOS E ENFOQUE COMPUTACIONAL	48
3.7 - CICLO DE CORTE DE ROSCA (PAT 105). DESCRIÇÃO E FORMATO	49
3.7.1 - CICLO DE CORTE DE ROSCA (PAT 105). ALGORITMOS E ENFOQUE COMPUTACIONAL	51
3.8 - EXEMPLO APLICATIVO DO PROCESSADOR GEOMÉTRICO	53
REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 3)	57
CAPÍTULO 4 - METODOLOGIA DO DESENVOLVIMENTO	
4.1 - GENERALIDADES	58
4.2 - ETAPAS DO DESENVOLVIMENTO	59
4.2.1 - ESPECIFICAÇÃO DOS ATRIBUTOS DO SPAT	59
4.2.2 - ESCOLHA DA LINGUAGEM E DO HARDWARE PARA DESENVOLVIMENTO	61
4.2.3 - MODELAGEM, DEFINIÇÃO DE ALGORITMOS E IMPLEMENTAÇÃO DE MÓDULOS SIMPLES	63
4.2.4 - DESENVOLVIMENTO DE MÓDULOS COMPLEXOS	64
4.2.5 - TESTES EM USUÁRIO TÍPICO E OTIMIZAÇÃO DO SISTEMA	64

4.2.6 - ANÁLISE DE RESULTADOS	65
4.3 - USUÁRIO TÍPICO	67
4.4 - ASPECTOS ORGANIZACIONAIS NECESSÁRIOS PARA IMPLANTAÇÃO DO SPAT	67
4.5 - GERENCIAMENTO DE PROGRAMAS ATRAVÉS DO SPAT	70
REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 4)	71-72

CAPÍTULO 5 - ESTUDO DE CASOS TÍPICOS.

5.1 - GENERALIDADES	73
5.2 - CONSEQUÊNCIAS DA IMPLANTAÇÃO DO SPAT	74
5.3 - APLICAÇÃO TÍPICA NÚMERO 1	74
5.4 - APLICAÇÃO TÍPICA NÚMERO 2	76
5.5 - ANÁLISE DE RESULTADOS	77
REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 5)	78

CAPÍTULO 6 - CONCLUSÕES

6.1 - GENERALIDADES	79
6.2 - METAS ALCANÇADAS	79
6.3 - LINHAS DE PESQUISAS PARA FUTUROS DESENVOLVIMENTOS	80
REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 6)	81

APÊNDICE 1 - PROGRAMAÇÃO MANUAL DE TORNOS CN

A.1.1 - GENERALIDADES	82
A.1.2 - SISTEMA DE COORDENADAS E CÓDIGOS DE EIXOS	82
A.1.3 - SINTAXE DOS BLOCOS DE INFORMAÇÕES EM PROGRAMAS CN	84
A.1.3.1 - PALAVRA PARA NÚMERO DE BLOCO	85
A.1.3.2 - PALAVRA PARA CONDIÇÃO DE PERCURSO: G	85
A.1.3.3 - PALAVRA PARA COORDENADAS	90
A.1.3.4 - PALAVRA PARA VELOCIDADE DE CORTE: S	90
A.1.3.6 - PALAVRA PARA FERRAMENTA: T	91

A.1.3.7 - PALAVRA PARA FUNÇÃO AUXILIAR: M	91
REFERÊNCIAS BIBLIOGRÁFICAS (APÊNDICE 1)	91-92
APÊNDICE 2 - UTILIZAÇÃO DO SPAT	
A.2.1 - GENERALIDADES	93
A.2.2 - MÓDULO EDITOR	94
A.2.2.1 - OPÇÕES PRINCIPAIS	95
A.2.2.2 - OPÇÕES SECUNDÁRIAS	99
A.2.3 - MÓDULO DE SIMULAÇÃO GRÁFICA	103
A.2.4 - MÓDULO PROCESSADOR DE GEOMETRIA	105
REFERÊNCIAS BIBLIOGRÁFICAS (APÊNDICE 2)	106
BIBLIOGRAFIA	108-115

CAPÍTULO 1

INTRODUÇÃO

1.1 - CONSIDERAÇÕES INICIAIS

As máquinas-ferramenta controladas numericamente tiveram o seu lançamento motivado, essencialmente, por questões técnicas: execução de perfis complexos e precisos para a indústria aeronáutica [1, 2].

A notável evolução, que esse conceito de maquinário sofreu, permitiu a sua adoção por diferentes segmentos dos Sistemas de Manufatura, tais como: indústrias têxteis, indústrias de calçados e, naturalmente, um quinhão muito significativo da indústria metal-mecânica [3].

O avanço tecnológico da micro-eletrônica contribuiu, significativamente, para uma drástica redução no investimento requerido para implantação de Máquinas-Ferramenta a Controle Numérico (MFCN), efetivando, assim, a disseminação e popularização desse equipamento [4].

Aplicações iniciais demonstraram que o CN não constitui apenas um complemento à MF, mas, também, (e principalmente !) em nova forma de organização e trabalho, viabilizando o surgimento de máquinas que somam efeitos de universalidade e flexibilidade .

É unânime a assertiva de que o campo economicamente saudável, para aplicação de MFCN, é constituído por lotes pequenos e médios,

sendo que a justificativa para esse fato decorre, principalmente, da "flexibilidade" desse tipo de maquinário, que permite facilmente a mudança do tipo de peça a ser manufaturado [5].

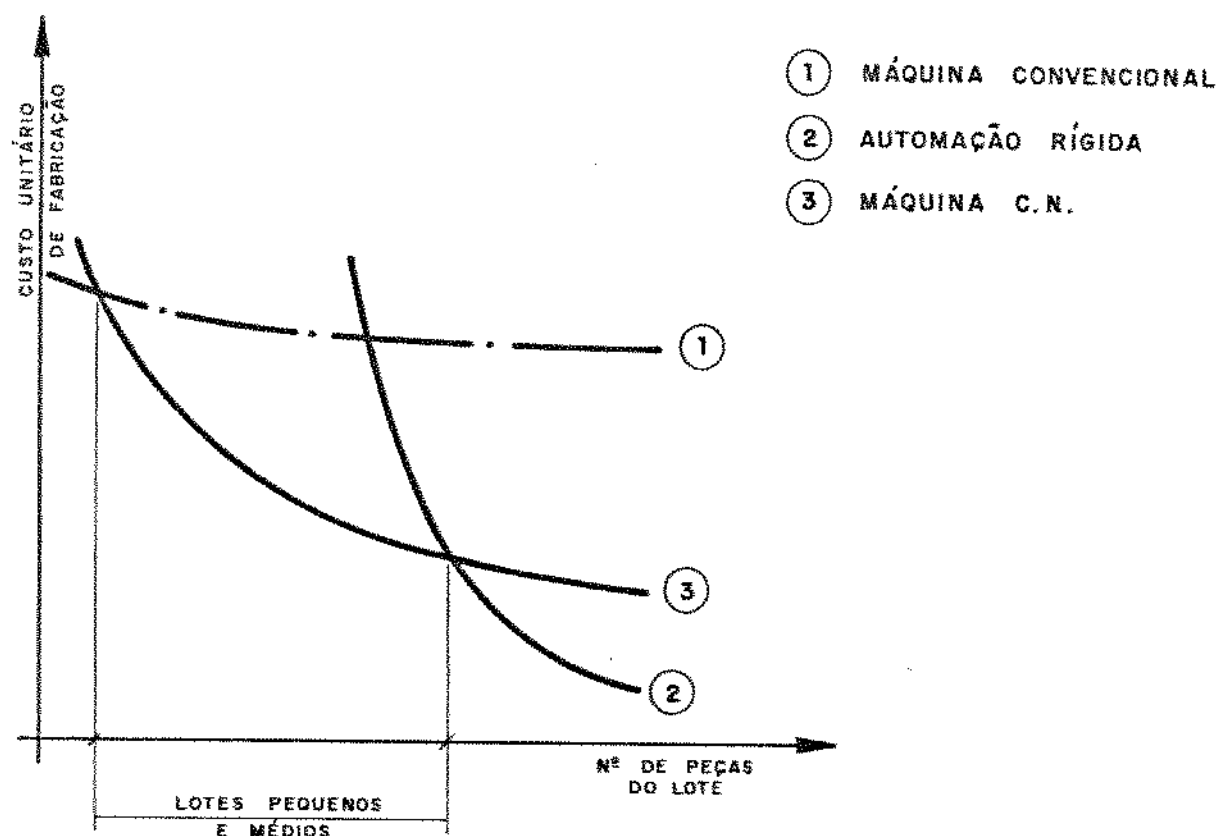


FIGURA 1.1 Custos de Fabricação para diversos Sistemas de Fabricação

Do universo de peças produzidas em países industrializados, prevalece o esforço (e o volume!) dedicado à produção de lotes pequenos e médios, tendência esta que deverá acentuar-se mais ainda, com a implantação das modernas técnicas de produção industrial, tais como "just-in-time" e "KANBAN", que pregam inclusive a viabilização econômica de manufatura com lotes unitários [6, 7].

Outra característica organizacional moderna, indica a situação cada vez mais comum de "horizontalização" , com as grandes empresas desenvolvendo fornecedores diversos, os quais evidentemente vão operar lotes médios e pequenos, propiciando portanto, terreno fértil para utilização de MFCN.

MFCN constituíram-se na "semente" daquilo que hoje denomina-se "Automação Flexível" e têm com sua aplicação, reflexos, não só a nível de Oficina Mecânica, mas, sim, em toda organização empresarial, requerendo o tratamento sistematizado das informações na empresa [8].

"Até recentemente, os serviços de computação (com algumas exceções) davam pouca atenção à fábrica, concentrando-se sobre o mercado mais lucrativo da automação dos escritórios" [9].

A implementação de Sistemas Informatizados para ambientes fabris é bem mais complexa e demorada, tanto assim que os próprios usuários (e não os fabricantes de computadores!) lançaram as bases dos sistemas de normalização do intercambio de informações em ambiente industrial [10,11].

A chamada Fábrica do Futuro deverá distinguir-se basicamente dos sistemas fabris atuais pelo controle sobre a informação, de forma a adequar-se, rápida e eficientemente, às oscilações mercadológicas, onde a vida útil dos produtos tende a ser cada vez menor, caminhando portanto no sentido da automação flexível, onde inovações serão os temas constantes [12, 13].

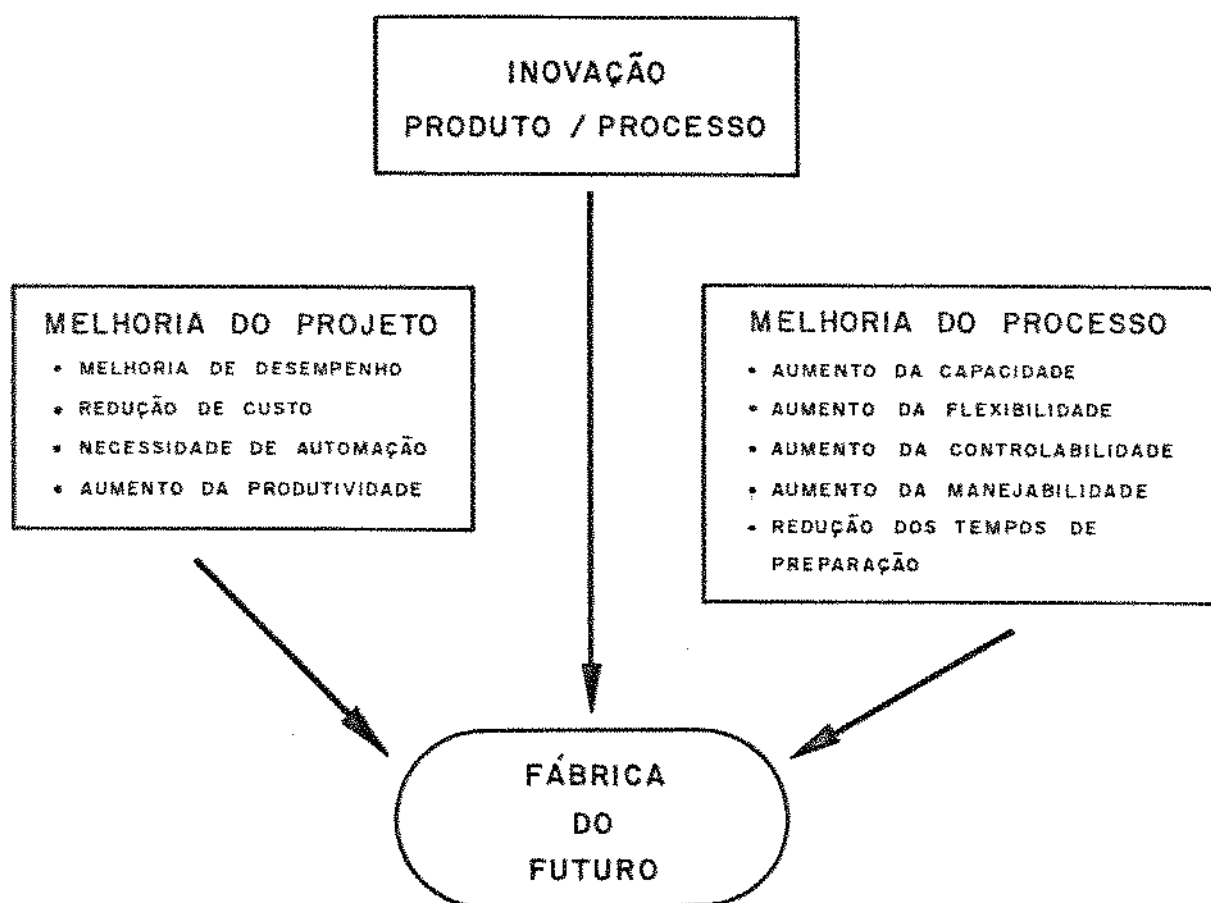


FIGURA 1.2-Necessidades tecnológicas para a Fábrica do Futuro.

1.2 - OBJETIVOS DESTE DESENVOLVIMENTO

MFCN têm seu espaço de trabalho referendado a um sistema de coordenadas cartesianas ortogonais, existindo a necessidade de programas individuais para cada peça do "mix" absorvido pelo maquinário [14].

Conforme o relato do item anterior, os sistemas produtivos deverão adequar-se convenientemente para produção cada vez mais diversificada, e de lotes cada vez menores. Isto implica no desenvolvimento de maquinário e, também, de sistemas de apoio que possibilitem a geração rápida e eficiente de programas para CN [15].

Programadores de C.N. podem atuar em princípio em 2 extremos:

- Programação manual (através de linguagens de baixo nível);
- Programação automática (utilizando linguagens de alto nível)

[16].

A programação manual é específica para a unidade de Controle utilizada, e requer o detalhamento de todas as funções a serem cumpridas pela MF, para efetivação do processo de fabricação.

Na chamada programação automática, o programador gera "programas-fonte" que são gerais e podem, em princípio, serem utilizados por qualquer MF. Este conceito requer a utilização de "processadores", que são responsáveis pela transformação de "programas-fonte" em "programas-objeto", os quais, por sua vez, serão utilizados por controles específicos das MF [17].

O Sistema de Programação Automática de Tornos - SPAT, constitui-se, basicamente, em ferramenta de programação CN com os objetivos:

- a) Prover o mercado nacional de ferramenta de programação, para profissionais programadores de CN;
- b) Viabilizar um pacote de "software" para ser utilizado como "laboratório" de cursos de Automação Industrial.

1.3 - MÓDULOS COMPONENTES DO SPAT

O SPAT é composto de 3 módulos: EDITOR, PROCESSADOR GEOMÉTRICO e SIMULADOR GRÁFICO.

O desenvolvimento do Editor não estava, originalmente previsto, mas essa necessidade tornou-se imperiosa para viabilizar a análise sintática/semântica das informações, à medida que elas são fornecidas.

O Processador Geométrico traduz, detalhadamente, as informações gerais do programa-fonte, gerando o chamado programa-objeto, o qual pode ser encaminhado para a máquina-ferramenta.

O Simulador Gráfico permite a visualização antecipada do processo de fabricação, detectando, antecipadamente, eventuais problemas nos programas CN [18,19].

A aplicação do SPAT em ambientes acadêmicos possibilita:

- a) Viabilização de cursos de CN em entidades que não possuem MFCN, através da aplicação orientada do SIMULADOR GRÁFICO, em testes de exercícios de programação [20];
- b) Caracterização dos dois extremos possíveis de programação CN: programação manual e programação automática.

Como ferramenta de programação em ambiente industrial [21] a aplicação do SPAT, possibilita:

- a) Redução nos tempos de programação;
- b) Elaboração de programas depurados, que reduzem os tempos para testes de programas, na Oficina Mecânica;
- c) Facilidade para desenvolvimento de programas CN de peças de geometria complexa;
- d) Complementação de recursos para Comandos menos evoluídos, podendo mesmo ser utilizado com Comandos CN simples;
- e) Racionalização do gerenciamento da "biblioteca" de programas CN, pela formação de arquivos magnéticos dos mesmos, o que implica, também, na redução do espaço físico reservado para o arquivamento de programas;
- f) Disponibilidade de conjunto de informações aos operadores, tais como trajetória de ferramentas e previsão de tempos;
- g) Redução do índice de refugos, devido a maior confiabilidade nos programas;
- h) Determinação da previsão de custos mais criteriosa, em função da possibilidade da emissão de relatórios da perspectiva da produção horária;
- i) Redução no tempo de "máquina-parada", devido a racionalização do intercâmbio de programas entre o departamento de Programação e a Oficina Mecânica.

REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 1)

- [1] SIMON, W. - The Numerical Control of Machine Tools [Die Numerische Steuerung von Werkzeugmaschinen], Edward Arnold, London, 1973.
- [2] STEMMER, E.C. - Efeitos do CN na Indústria Mecânica e Perspectivas, Máquinas & Ferramentas, Ano 1, n. 7, p. 36, Junho, 1980.
- [3] FUCCI, T.V.; FRANCA, A.L. - A Informática e a Indústria de Bens e Capital, Reunião Aberta do 40º Congresso Anual da ABM, V.1, p.3, São Paulo, 1985.
- [4] MILLER, S.G. - New Directions in Computer-Aided Design/Computer-Aided Manufacturing (CAD/CAM) Systems, NATO ASI SERIES, V. F11, p. 245, Springer-Verlag, Berlin, 1984.
- [5] TELLES, G.N. - Estudos Comparativos entre Máquinas a Controle Numérico e as Convencionais, Máquinas & Ferramentas, Ano 1, n. 11, p. 45, Junho, 1980.
- [6] MIURA, H.M.; GIRAUD, C.O. - Tecnologia de Grupos - O Sistema "SAAP" (Sistema de Apoio a Projetos de Produtividade), Anais do 7º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p. 15.01, 1987.

- [7] LEWIS, W.; BARAS, H.M.M.; SOLBERG, J.J. - Computer Integrated Manufacturing System Control: A Data Flow Approach, Journal of Manufacturing Systems, V. 6, p. 177, 1987.
- [8] OLIVEIRA, J.F.; LIRANI, J. - Algumas Considerações a Respeito do Desenvolvimento de CAM na Indústria Mecânica, Anais do 7º Congresso de Comando Numérico no Brasil - SOBRACON, V.1, p. 14.01, 1987.
- [9] SMITS, L.J. - A Automação e a Robótica Valem Realmente a Pena?, Boletim SOBRACON, ano IV, n. 38/39, p. 5.14, São Paulo, 1988.
- [10] MENDES, M.J. - Redes Locais de Comunicação em Ambiente Industrial, 6º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p. 17.01, 1986.
- [11] MACALONEY B; CAPURRO, J. - GM MAPs Route to Integrated Factory Communications, Blue Book Machine and Tool, V. 80, n 11, p. 45, Nov. 85.
- [12] MONROY, D.; CAULLIRAUX, H.M. - Engenharia Industrial Avançada: Uma Alternativa da Fábrica do Futuro, Anais do 7º Congresso de Comando Numérico no Brasil - SOBRACON, V.1, p. 17.01, 1987.
- [13] INDÚSTRIA ROMI S/A - Programação de Torno a Controle Numérico, Máquinas & Ferramentas, ano 1, n. 5, p. 20, 1979.
- [14] KEREDITH, J.R. - Implementing the Automated Factory, Journal of Manufacturing Systems, V. 6, p. 1, 1987.

[15] KASZA, J. - Organizational Model of NC Machine Installation, Periodica Polytechnica, M30/3-4, p. 359, 1984.

[16] HATSCHEK, R.L. - NC Programming, American Machinist, p. 118, Feb. 80.

[17] ROZENFELD, H. - Desenvolvimento de Uma Linguagem Universal para Programação de Máquinas CNC, 9º Seminário de Comando Numérico -SOBRACON, São Paulo, 1989.

[18] KILLS, R. I. - Computer Simulation - A Feasibility and Planning Tool for FMS, Proceedings of 2nd International Conference on Flexible Manufacturing Systems, London, V. 1, p.185, 1983.

[19] TELLES, G.N. - Desenvolvimento de um EDITOR/SIMULADOR para Tornos a Controle Numérico, Tese de Mestrado (102/85), Faculdade de Engenharia de Campinas FEC/UNICAMP, 1985.

[20] UNBEHAVEN, H. - CAD in Control Engineering Education, IFAC Proceedings Series, V. 5, p. 33, Lyngby, Denmark, 1985.

[21] GENDRE, J.C. - PROMO - An NC System for Small Computer, International Conference on Computer Aided Manufacture and Numerical Control, p. 9-1, Glasgow, 1974.

CAPÍTULO 2

LINGUAGENS COMPUTACIONAIS UTILIZADAS EM AMBIENTE DE MANUFATURA INDUSTRIAL

2.1 - GENERALIDADES

Até alguns anos atrás os sistemas CAD/CAM tinham sua utilização bastante restrita a grandes empresas dos segmentos aéreo-espacial e automobilístico. O advento de "hardwares" mais baratos (particularmente microprocessadores) e o surgimento de pacotes de "software" do tipo "user friendly", possibilitou a empresas de pequeno e médio porte adquirirem e utilizarem sistemas CAD/CAM para desenho, projeto, planejamento, controle da produção industrial e Métodos e Processos. Isto alterou radicalmente a forma de projetar e manufaturar produtos industrializados [1].

Podemos relacionar como benefícios advindos da utilização do CAD/CAM:

Poder de cálculo: Com a notável evolução por que passaram os computadores, foi possível aplicar modelos físico-matemáticos muito mais criteriosos e precisos, de maneira a otimizar os projetos de Engenharia;

Simulação Gráfica: A representação de figuras em 2 dimensões permite a visualização em perspectiva de peças e conjuntos, permitindo a passagem de vistas para perspectivas e vice-versa. É possível, também, a visualização dinâmica de processos e a verificação do desempenho cinemático de conjuntos [2].

De uma forma geral, os sistemas de CAD/CAM possibilitam melhorar a qualidade do projeto e otimizar os ciclos de trabalho,

reduzindo conseqüentemente o "lead-time". A constituição de um banco de dados único possibilita: redução de erros e maior e melhor controle gerencial, o que implica em redução de custos indiretos para a empresa [3].

2.2 - FORMAS BÁSICAS DE PROGRAMAÇÃO EM AMBIENTE INDUSTRIAL

Da mesma maneira que na programação científica em geral, a abordagem de programação para Sistemas Produtivos Automatizados pode ocorrer em diferentes formas:

- Programação Manual, quando a linguagem utilizada é de nível baixo, utilizando-se essencialmente códigos numéricos, sem o emprego de nenhum recurso computacional além do próprio armário de Comando da Máquina. Esse tipo de Linguagem é normalmente referenciado como "linguagem-de-máquina";

- Programação Automática, quando se empregam linguagens de alto nível e recursos computacionais de grande porte, com linguagens de programação que eventualmente empregam "primitivas" do idioma inglês [4].

Com o advento dos micro-computadores, foram lançados alguns pacotes de "software" que poderiam ser classificados em "Programação Semi-Automática". Mas ocorre que, atualmente, o poder computacional dos micro-computadores aumentou tanto que é perfeitamente possível utilizar-se Programação Automática em tais equipamentos.

2.3 - GÊNESIS

A maioria das pessoas envolvidas com Engenharia de Manufatura assume o fato de estarmos atualmente dentro do ciclo evolutivo daquilo que deverá constituir-se na Nova Revolução Industrial [5].

É interessante observar-se que a chamada Revolução Industrial é hoje lembrada, muito mais, por seus impactos sociais, do que pelas inovações tecnológicas dela advindas.

De uma forma simplista podemos afirmar que, tecnicamente com, a Revolução Industrial, o Homem obteve o domínio sobre os Sistemas de Potência (motor a vapor & regulador de Watt). Esses inventos possibilitaram e facilitaram os sistemas de distribuição e controle de energia nas Unidades Fabris.

Uma grande parcela de esforço em Pesquisa e Desenvolvimento nos dias de hoje, pode ser resumida em função da necessidade crescente em busca do controle sobre os Sistemas de Informações.

A chamada Fábrica do Futuro deverá permitir um domínio completo sobre a Informação. De forma abrangente, teremos informações do tipo: "o que fazer" -- descendo aos escalões inferiores e informações do tipo: "o que foi feito" -- subindo aos escalões superiores.

A Sociedade do Consumo tem pressionado cada vez mais os Sistemas Fabris no sentido dos mesmos adequarem-se dentro de um novo conceito de Automação: A Automação Flexível. Isto tudo, devido

à redução do período de exploração comercial dos produtos, ao lado de uma exigente diversificação de modelos.

Podemos assumir como data base da Nova Revolução Industrial o início da década de 50, quando se viabilizou em protótipo uma Máquina-Ferramenta que "entendia números". Hoje, esse tipo de Máquina é denominado "a Comando Numérico" e a sua viabilização marcou o descortínio de um campo amplo de aplicações [6].

A notável evolução no campo dos computadores digitais foi parcialmente motivada pela pressão exercida no sentido de caminhar-se na direção da Automação Flexível.

O desenvolvimento nesse campo possui duas componentes que se complementam: "hardware" e "software".

Um marco importante na área de "software" foi o lançamento, em 1956, do primeiro compilador FORTRAN (FORMula TRANslation). O esforço intelectual dispendido nesse desenvolvimento correspondeu a 40 homens-ano e, quando do lançamento de versões experimentais do compilador, existiam pessoas da própria equipe que desenvolviam o compilador, que eram céticas com relação à viabilização do projeto.

Felizmente, versões posteriores do compilador FORTRAN comprovaram ser bastante confiáveis, constituindo-se, basicamente, na primeira Linguagem Científica de Alto Nível para computadores digitais, permitindo então a pessoas não diretamente familiarizadas com a arquitetura dos computadores, a utilizarem os mesmos em seus campos específicos.

É interessante observar-se que o lançamento de uma nova linguagem de programação proporciona, inevitavelmente, o surgimento de sub-linguagens as quais, por sua vez, foram desenvolvidas a partir daquela.

Logo após o lançamento da linguagem FORTRAN (e utilizando-a !) o MIT desenvolveu a primeira linguagem de Alto Nível para Máquinas tipo CN: o APT (Automatically-Programmed of Tools). Da mesma forma que o FORTRAN, foram desenvolvidas versões mais poderosas do APT: APT II, APT III, etc. proporcionando um campo específico para desenvolvimento de "software".

O início da década de 60 registra como importante a comercialização do primeiro Robô Industrial lançado pela UNIMATION, além dos primeiros pacotes de CAD (Computer Aided Design) [1]

Nos meados dos anos 70, temos a registrar o desenvolvimento do microprocessador, o qual viabilizou o surgimento das Máquinas tipo CNC (Comando Numérico Computadorizado), que corresponde a uma evolução sobre as primeiras máquinas CN.

O final da década de 70 e início dos anos 80, registram o surgimento dos Sistemas Flexíveis de Manufatura, com a integração de Máquinas Ferramentas CNC com Sistemas Robotizados e Sistemas de Movimentação de Materiais.

O microprocessador permitiu, também, uma verdadeira popularização na utilização de computadores através dos micro-computadores do tipo PC (Personal Computer).

2.4 MODUS-OPERANDI DE UMA MÁQUINA-FERRAMENTA CNC

A viabilização das máquinas tipo CNC proporcionou uma verdadeira transferência de responsabilidades do ambiente de Oficina Mecânica para o Departamento de Métodos e Processos [7].

Todo o Processo de Fabricação é planejado e estruturado previamente de maneira que, ao operador, é estipulada a função de supervisor do processo, intervindo apenas em situações impossíveis de serem previstas, tais como: falha de ferramentas, heterogeneidade de material, etc.

As Unidades de Comando Numérico consistem basicamente em micro-computadores de propósito específico que possuem um "software residente" fornecido pelos fabricantes e um "software" desenvolvido pelo usuário final em função de suas necessidades específicas.

Sob o ponto de vista do usuário, um armário de Controle tipo CNC opera com 2 tipos de informações: "informações-de-ligações" e "informações-de-percurso". Informações de ligações, são aquelas reservadas, por exemplo, para energizar uma bomba de fluido refrigerante, o avanço do mangote de um contra-ponto, etc. Informações de trajetória são de densidade bem mais ampla e de uma complexidade bem maior.

Em princípio, as informações de trajetória determinam o posicionamento relativo do par ferramenta-peça em função de seus parâmetros cinemáticos: posição, velocidade e aceleração. Dependendo da situação (acabamento, desbaste, etc.), essa quantidade de informações pode ser muito grande e bastante intensa.

Ao programa cabe indicar o ponto inicial da trajetória, o ponto final e a velocidade. A determinação dos pontos intermediários é executada por um interpolador integrante à Unidade de Comando [8].

Interpoladores com crescente opção de recursos e capacidade de cômputo estão sendo desenvolvidos e, a cada nova geração, ampliam-se os recursos oferecidos pelos mesmos.

De toda forma, interpoladores internos do armário de comando não dispensam a necessidade de um processamento geométrico "off". Esse processamento ocorre tendo em vista o desenho do produto, a máquina-ferramenta escolhida para o trabalho e as condições de usinagem.

2.5-LINGUAGENS DE PROGRAMAÇÃO ESPECÍFICAS PARA CNC

O processamento geométrico indicado no item anterior pode ser feito através da utilização de linguagens específicas, tais como o APT, ou linguagens que utilizam macro-instruções bastante próximas da própria "linguagem-de-máquina" do armário de Comando [4].

A partir do lançamento da linguagem APT foi desenvolvida uma série relativamente grande de linguagens e/ou sub-linguagens para CNC [7]. Todo esse esforço é motivado, naturalmente, para "facilitar" o trabalho de programação de máquinas CNC. Outro motivo é consequência da grande diversificação de modelos de armário CNC e de periféricos de apoio à programação.

Segundo levantamentos por nós efetuados, verifica-se que 80% do pessoal envolvido com programação de MFCN utiliza a linguagem de programação baseada na Norma DIN 66025 [9,10], com o chamado padrão "G" de programação, ou um padrão bastante semelhante a esse [11,12].

2.6 - JUSTIFICATIVA PARA O DESENVOLVIMENTO DE UM PROCESSADOR DE GEOMETRIA

A nossa proposta de trabalho consiste no desenvolvimento específico, para Máquinas-Ferramenta Controladas Numericamente, de um: "Processador de Geometria para Tornos a Controle Numérico".

A função desse processador consiste em "gerar" códigos e funções que farão parte de um programa para máquina CN. Essa geração deverá ser feita em função do perfil final da peça, definido de maneira clara e simples.

Acreditamos que uma definição clara e simples do perfil final da peça pode ser feita pelo próprio padrão da linguagem, na maior parte dos casos: linguagem padrão G, com a qual os programadores já estão bastante habituados.

Este enfoque viabilizará, então, um nível de programação semi-automática para MFCN, de forma que programadores e sistemas, que utilizem a chamada programação manual, possam migrar rapidamente para esse sistema.

A utilização de um Processador de Geometria libera, então, o programador da tarefa enfadonha de cálculos geométricos, necessários para a elaboração de programas, proporcionando uma

redução da ordem de 2/3 no esforço (e no tempo !) requerido para a produção de programas de CN.

2.7 - EQUIPAMENTO E LINGUAGEM UTILIZADA NO DESENVOLVIMENTO

Utilizou-se um microcomputador compatível com o IBM-PC Xt, com 2 unidades de disco flexível (5 1/4") e 640 K de memória RAM.

Como linguagem de desenvolvimento optou-se pelo PASCAL [13] na situação de seu dialeto TURBO-PASCAL da BORLAND [14].

O emprego desse equipamento é justificado pela sua grande disseminação em nosso país, e a opção da linguagem TURBO-PASCAL devido a:

- excelente para desenvolvimentos;
- linguagem estruturada de amplos recursos;
- possibilitar, através de documentação, a implementação modular do "software" desenvolvido;
- possuir "primitivas" para computação gráfica poderosas e de utilização simples.

2.8 - COMPARAÇÕES DE CUSTOS ENTRE AS DIFERENTES TÉCNICAS DE PROGRAMAÇÃO

Partindo da premissa de que o campo ótimo para utilização de máquinas CN é constituído pelo universo abrangido por lotes pequenos e médios, podemos comparar custos de programação em função da complexidade da peça.

Quando da utilização de programação automática, haverá de adicionarem-se os custos indiretos advindos do emprego do computador.

É unânime e intuitivo que, a partir de um certo grau de complexidade da peça, o custo de programação na forma automática é economicamente mais saudável [15].

2.9 - METAS PRINCIPAIS DO PROJETO

Como metas principais nesse projeto podemos relacionar:

- 1) Viabilização de um Sistema de Programação Automática de Tornos - SPAT, com as características principais:
 - a) "rodar" em equipamento de pequeno porte;
 - b) atuação iterativa e "amigável" entre o usuário e os módulos que compõem o Sistema;
 - c) obtenção de um processador geométrico potente, eficaz e de utilização simples.
- 2) Obtenção de um "pacote" da área "CAM" que possa ser utilizado nos ambientes:
 - a) acadêmico, como suporte a cursos, treinamentos, etc;
 - b) industrial, como ferramenta de trabalho para programadores de Tornos CNC [16].
- 3) Tornar disponível ao mercado nacional uma ferramenta de programação que aumente a produtividade e a eficiência em tarefas de programação de MFCN.

Quando da programação manual, os custos são calculados como despesas gerais em termos de ressarcimento da mão de obra envolvida, espaço de trabalho ocupado e equipamento requerido. No

caso da programação automática, devem-se adicionar os custos pela utilização do computador.

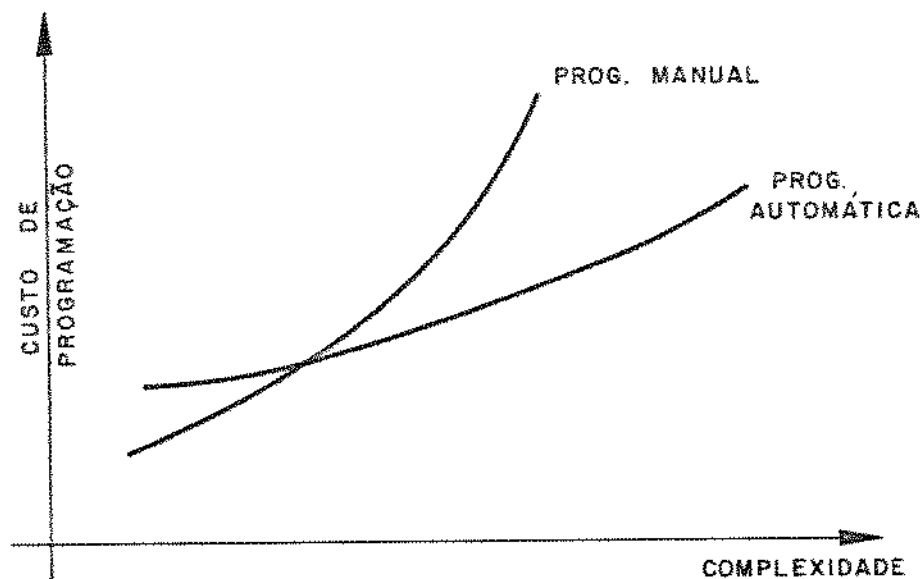


FIGURA 2.1-Curvas de Custos entre os tipos possíveis de Programação.

2.10 - MÓDULOS E RECURSOS OPERACIONAIS DO SPAT

O Sistema de Programação Automática de Tornos, consta de 3 módulos que se integram e se complementam: Módulo de Edição de Programas, Módulo de Simulação Gráfica [14] e Módulo Processador Geométrico.



FIGURA 2.2-Módulos componentes do SPAT.

O EDITOR deve ser dedicado e específico para a Unidade de Controle que deverá receber os programas do SPAT, sendo que editores de propósito múltiplo não são eficazes para esse tipo de trabalho. O editor para CN pode ser elaborado de forma que vá analisando, sintática e analiticamente, o programa, à medida que ocorra a digitação.

A Figura 2.3 ilustra as operações executadas pelo módulo Editor.

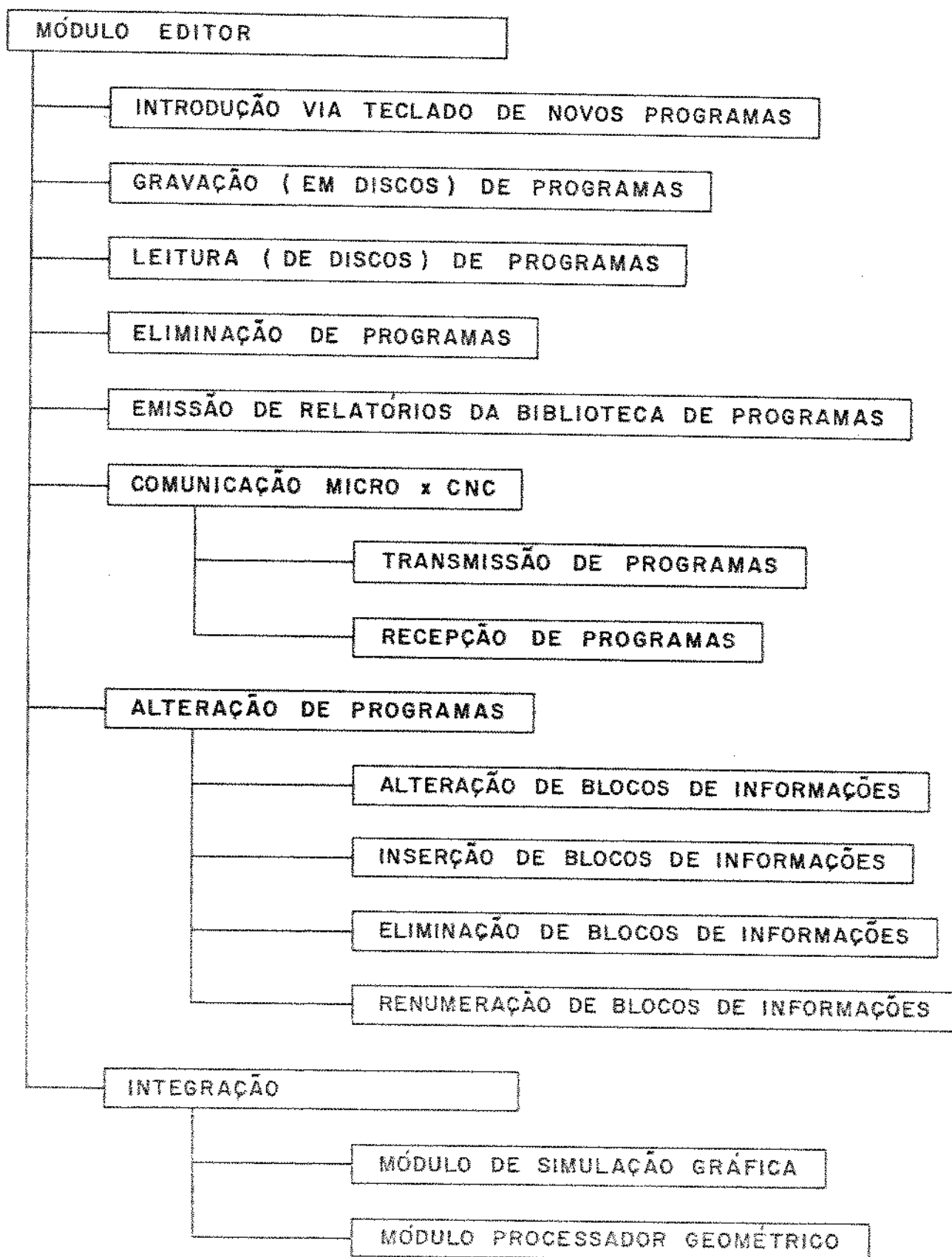


FIGURA 2.3-Operações efetuadas pelo EDITOR.

O Simulador Gráfico possibilita que programas CN sejam testados e analisados, antes do envio dos mesmos para a Oficina Mecânica. A utilização do Simulador detecta eventuais problemas nos programas, o que proporciona a remessa de programas depurados para a Oficina Mecânica.

A Simulação gráfica, permite o rastreamento cinemático da ponta teórica da ferramenta, evidenciando a interação ferramenta x peça (e dispositivos). Essa simulação é feita através da "consulta" do programa CNC, fornecendo ainda no final da mesma, a previsão de tempos, pois todos elementos requeridos para tanto devem constar do corpo do programa.

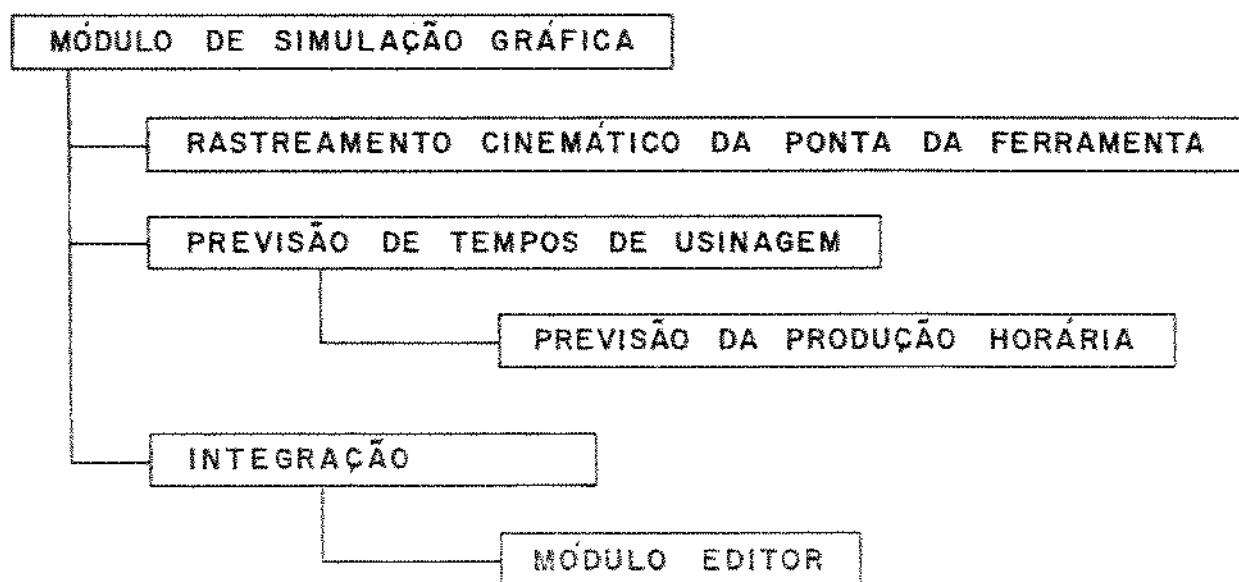


FIGURA 2.4-Operações efetuadas pelo Módulo de Simulação Gráfica.

O desenvolvimento do módulo Processador de Geometria foi precedido pelo levantamento da estrutura funcional dos processadores de geometria disponíveis no mercado .

Concluiu-se que o processador de geometria oferecido pelo TX-8 (DIGICON) oferece uma estrutura bastante simples para o usuário final, sendo, portanto, escolhido como base para a estrutura funcional do PROCESSADOR DE GEOMETRIA.

A diferença fundamental é que o processador de geometria do TX-8 é "residente" no armário de Controle e funciona apenas para esse tipo de controle, sendo que o desenvolvido nesse trabalho opera "off" do armário, podendo ser utilizado para qualquer tipo de controle.

O PROCESSADOR DE GEOMETRIA por nós desenvolvido tem aplicações reais para o controle MACH 3L de fabricação da ROMI e do Controle MCS 210, ambos nacionais.

A idéia consiste em expandir os códigos "G" [6] e trabalhar com macro-instruções mais poderosas do que as previstas originalmente no desenvolvimento do armário de controle.

Basicamente, opera-se com 6 macro-instruções : G100, G101, G102, G103, G104 e PAT105, que permitem o processamento da geometria de peças complexas, sendo que ao programador resta a tarefa de programar apenas as operações de acabamento (que são fáceis de programar !) e ao processador, gerar automaticamente a parte do programa referente ao desbaste (difícil de programar !).

A utilização de micro-computadores e pacotes de "software" adequados agilizam tremendamente a geração, a otimização e o gerenciamento de programas CN. A forma usual do emprego do SPAT é ilustrada na Figura 2.5.

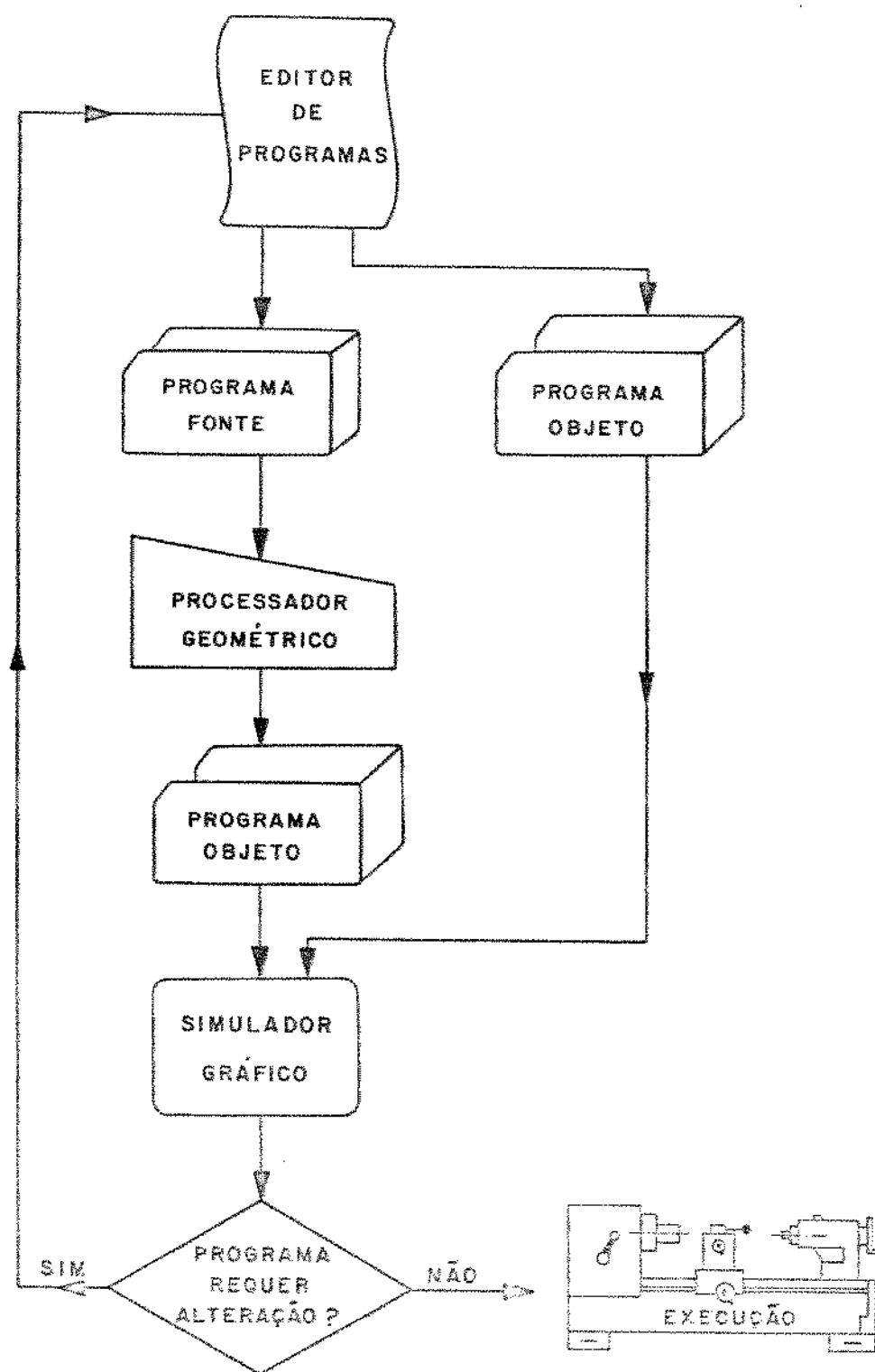


FIGURA 2.5-Fluxo de Trabalho utilizando o S.P.A.T.

REFERÊNCIAS BIBLIOGRÁFICAS (CAP 2)

- [1] TEICHOLZ, E. - CAD/CAM Handbook, Mc Graw Hill Book Co., New York, 1985.
- [2] GOODFELLOW, S.D.; MURNRO, N. - INTEGRA, an Input Translation Facility for Computer Aided Control Systems Engineering, IFAC Proceedings Series, V. 5, p. 87, 1986.
- [3] QUEIROZ, A.A.; BOEHS, L.; COUTINHO, S.A.L. - Seleção de Condições Otimizadas de Corte "Software" Apoiado por Banco de Dados, 6º Seminário de Comando Numérico no Brasil - SOBRACON, V.1, p. 01.01, São Paulo, 1986.
- [4] QUEIROZ, A.A. - Um Sistema de Programação do Tipo "APT" sem Linguagem, 5º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p. 4.1, São Paulo, 1985.
- [5] GOLDBAR, J. D. - The Organizational Impacts of Computer Based Manufacturing, NATO AIS Series, V. F11, p. 567, Springer-Verlag, Berlin, 1984.
- [6] PRESMAN, R.; WILLIAMS, J.E. - Numerical Control and Computer Aided Manufacturing, John Wiley & Sons, New York, 1977.
- [7] SIMON, W. - The Numerical Control of Machine Tools [Die Numerische Steuerung von Werkzeugmaschinen], Edward Arnold, London, 1973.

- [8] MACHADO, A. - O Comando Numérico Aplicado às Máquinas Ferramenta, ícone Editora, São Paulo, 1986.
- [9] DIN 66024: Numerische Steuerung von Arbeitmaschinen - Code fuer 8-Spur-Lochstreifen, Beuth-vertrieb, Berlin, 1967.
- [10] DIN 66025: Programmaufbau fuer numerisch gesteuerte Arbeitsmaschinen, Beuth-vertrieb, Berlin, 1969.
- [11] VDI 3255: Programmieren numerisch gesteuerter Werkzeugmaschinen-Festlegung der Koordinatenachsen und Zuordnung der Bewegungsrichtungen, Duesseldorf, 1968.
- [12] ISO/DIS 2806.2: Numerical Control of Machines - Vocabulary, 1979.
- [13] SCHIMIDT, G.; LAPPUS, G. - Digital Simulation Methods - A tutorial, IFAC Proceedings Series, V. 6, p. 83, 1986.
- [14] Turbo Pascal - Reference Manual, Borland International, 1985.
- [15] WECK, M. - Handbook of Machine Tools [Werkzeugmaschinen, Band 3, Automatisierung und Steuerungstechnik], John Willey & Sons, Chichester, V. 3, 1984.
- [16] YONG, Y.F.; BONNEY, M.C.; KNIGHT, J.A.G. - GRASP a Design Tool for FMS, Proceedings of 2nd International Conference on Flexible Manufacturing Systems, London, V. 1, p. 215, 1983.

CAPÍTULO 3

CICLOS AUTOMÁTICOS DE PROGRAMAÇÃO OFERECIDOS PELO PROCESSADOR DE GEOMETRIA

3.1 - GENERALIDADES

A tarefa de programar tornos CN, utilizando-se linguagens de baixo nível, é bastante trabalhosa, principalmente no aspecto do detalhamento de operações de desbaste, parcela esta do trabalho, bastante enfadonha e demorada.

Esse aspecto evidencia-se mais ainda quando a matéria prima que "dá-a-carga" no torno, constitui-se em "blanque" simplesmente serrado, a partir de vergalhão laminado ou trefilado. Nesta circunstância, uma percentagem elevada de casos requer operações de desbaste acentuado, que precedem as operações de acabamento.

Outro aspecto, a ser ressaltado, refere-se ao corte de roscas em tornos CN com ferramentas mono-corte (um único "VÊ" de corte), onde a rosca é executada através de sucessivas "passadas" e a distribuição das mesmas contribui, significativamente, tanto na qualidade da rosca obtida quanto na vida da ferramenta.

A função principal do Processador de Geometria consiste então, basicamente, em ferramenta de trabalho para programação, onde o programador detalha apenas o perfil final da peça, e o Sistema gera, automaticamente, a parte do programa referente ao desbaste.

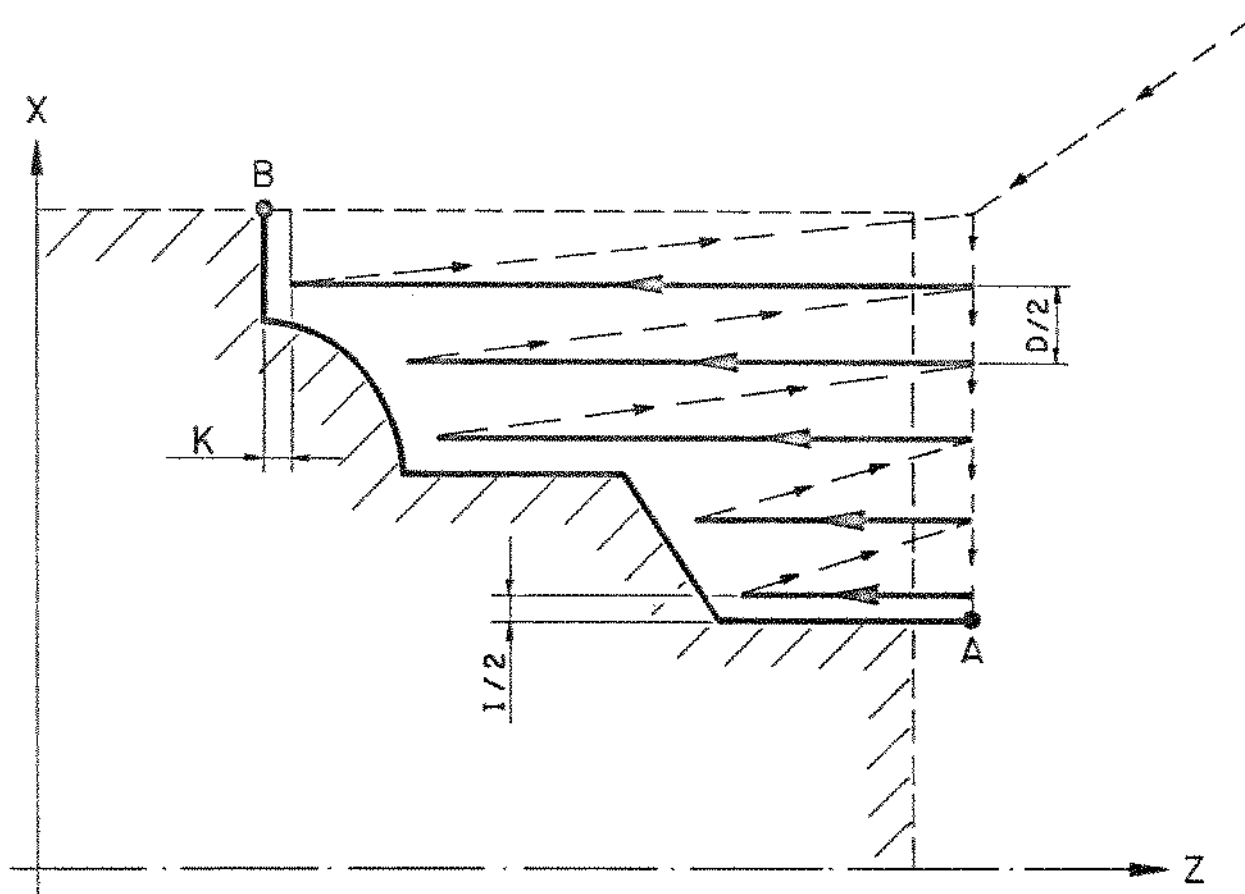
3.2 - CICLO DE DESBASTE LONGITUDINAL (G100): DESCRIÇÃO E FORMATO

O desbaste em torno é, em grande parte dos casos, efetuado através de operações do tipo "cilindrar" [1,2]. Cabe, então, ao programador definir as condições de corte para que o ciclo de desbaste longitudinal seja efetuado. Essas condições estão, evidentemente, vinculadas ao quarteto: máquina-ferramenta, ferramenta, dispositivos de fixação [3] e material.

Conforme demonstra a Figura 3.1, o desbaste deve ser feito trabalhando-se com profundidades de corte máximas, de forma a reduzir-se o número de passadas [4]. Para tanto, o programador deverá cuidar para que as condições de corte estejam dentro do intervalo de máxima eficiência, e que a potência consumida não comprometa a máquina-ferramenta.

O desbaste deve ser feito levando-se em conta o perfil final da peça e o "sobre-material" que deve ser "preservado" para a ferramenta de acabamento. É importante que o processador permita que o usuário estipule "sobre-materiais" diferenciados, de forma que a ferramenta de acabamento trabalhe com um "comprimento-em-corte" uniforme [5]. A especificação do sobre-material será então determinada em função do raio de arredondamento da pastilha e do ângulo de posição da ferramenta de acabamento.

As Figuras a seguir apresentam trajetórias de ferramentas de MFCN onde adotou-se a convenção usual dos programadores CN: tracejado fino representa movimentos rápidos de aproximações ou afastamentos e traço-forte-contínuo indica operações de usinagem.



FORMATO :

$N_{xxx} G100 O_{AAA} Q_{BBB} I_{iii} K_{kkk} D_{ddd} [A] F_{fff}$

ONDE : xxx = Nº DO BLOCO CORRENTE

AAA = Nº DO BLOCO INÍCIO ACABAMENTO

BBB = Nº DO BLOCO FIM DO ACABAMENTO

iii = SOBREMETAL NA DIREÇÃO X [EM Ø]

kkk = SOBREMETAL NA DIREÇÃO Z

ddd = PROFUNDIDADE DE CORTE [NO Ø]

A = ALISAMENTO (FACULTATIVO)

fff = VALOR DO AVANÇO

FIGURA 3.1-Ciclo de Desbaste Longitudinal-G100

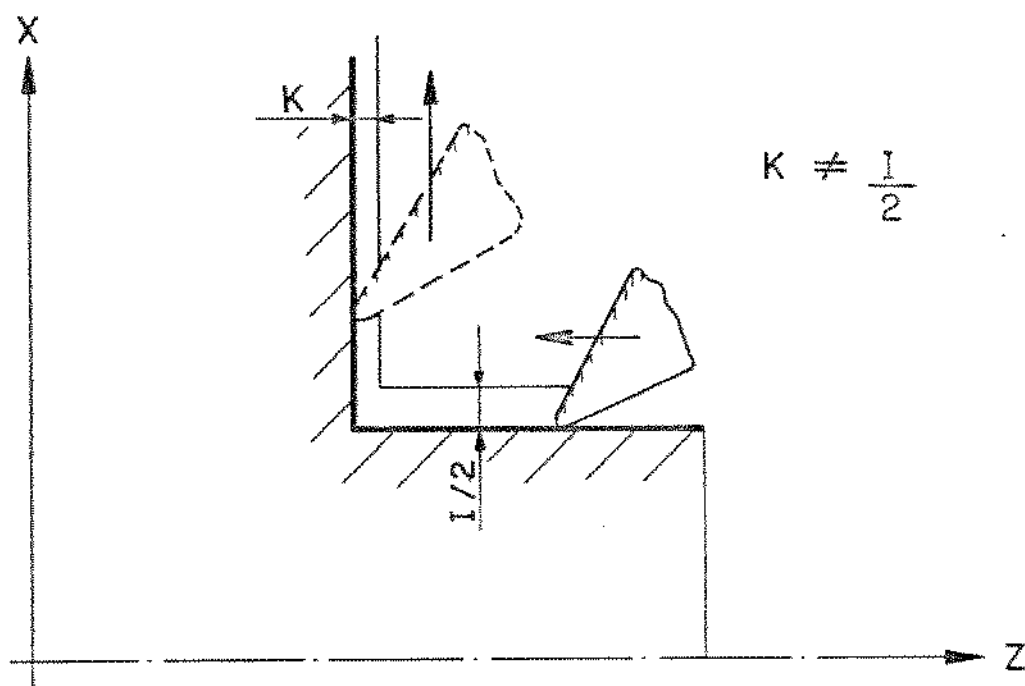


FIGURA 3.2-Sobre-materiais diferenciados em função da geometria da ferramenta.

O processador permite, ainda, que o usuário especifique na "chamada" do Ciclo de Desbaste Longitudinal, um pré-acabamento, obtido através da inclusão do parâmetro A, quando da especificação do ciclo. Esse pré-acabamento é feito, evidentemente, respeitando-se o perfil final da peça e os "sobre-materiais" especificados. Na situação de usinagem de peças com "tolerância-aberta", emprega-se a mesma ferramenta para desbaste e acabamento, o que implica que o usuário não deverá incluir o parâmetro A, quando da especificação do ciclo de Desbaste Longitudinal.

3.2.1 - CICLO DE DESBASTE LONGITUDINAL (G100): ALGORITMOS E ENFOQUE COMPUTACIONAL

A programação do perfil final da peça, proporciona elementos para que o procesador "monte" uma matriz bidimensional denominada MATFUNC (matriz de funções).

Essa matriz terá tantas linhas quantos forem os trechos do perfil final, e constará de 11 colunas de informações numéricas de cada trecho, conforme "lay-out" a seguir.

coluna --->

1	2	3	4	5	6	7	8	9	10	11
T	XC[2]	XC[1]	ZC[2]	ZC[1]	II	KK	RR	ANG1	ANG2	G

onde:

T especifica o tipo do trecho do perfil, e

T = 1	corresponde a um trecho cônico.
T = 2	" " " " cilíndrico.
T = 3	" " " " faceado.
T = 4	" " " " circular.

XC[2]	corresponde à coordenada X do início do trecho.
XC[1]	" " " " X " fim
ZC[2]	" " " " Z " início
ZC[1]	" " " " Z " fim
II	" " " " X " centro do arco.
KK	" " " " Z " " " "
RR	" " ao valor numérico do raio do arco.
ANG1	" " ao ângulo de início do arco.
ANG2	" " " " " fim " "

G especifica o sentido do arco circular, e

G = 2	para interpolação circular horária.
G = 3	" " " " anti-horária.

Resumindo: a matriz MATFUNC, retrata de forma numérica, o perfil final da peça "deslocado" devido aos sobre-materiais especificados.

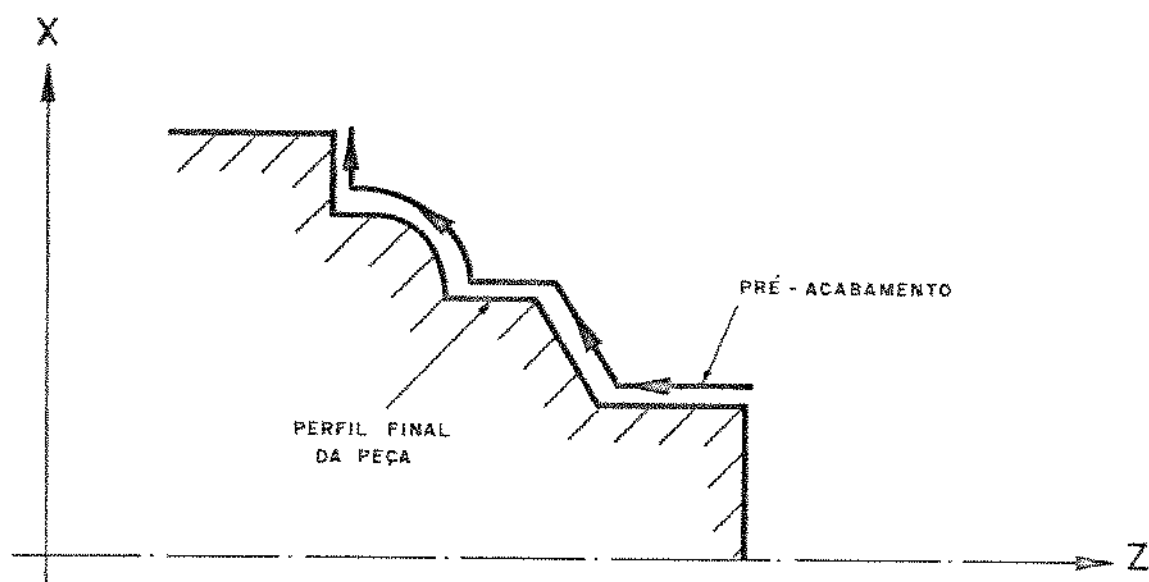


FIGURA 3.3-Perfil Final e Perfil Deslocado.

Montada a matriz MATFUNC, o tratamento computacional foi orientado de forma que sejam calculadas as coordenadas dos pontos de intersecção do "perfil deslocado" com as retas dos diversos "passes".

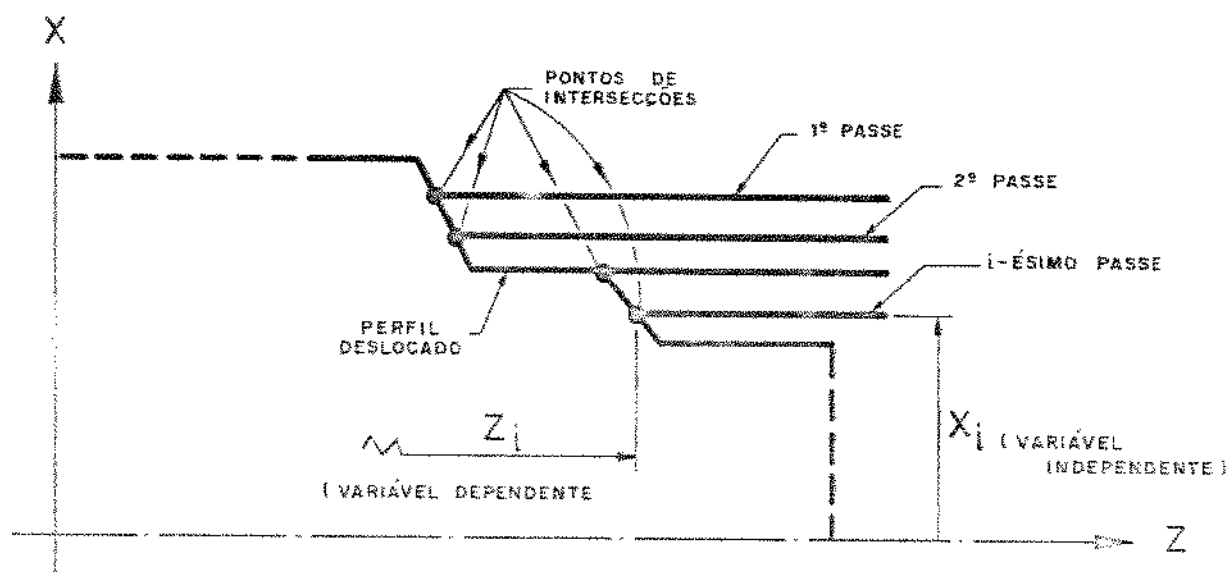


FIGURA 3.4-Pontos de Intersecção a serem calculados para ciclo G100.

O valor da coordenada Z_1 , é determinado em função de X_1 (variável independente) e do perfil deslocado. Atribuído o valor numérico de X_1 , o computador pesquisa as linhas da matriz MATFUNC até localizar a linha que contenha o intervalo $X[2] \text{ ---- } X[1]$ para o qual X_1 é intermediário.

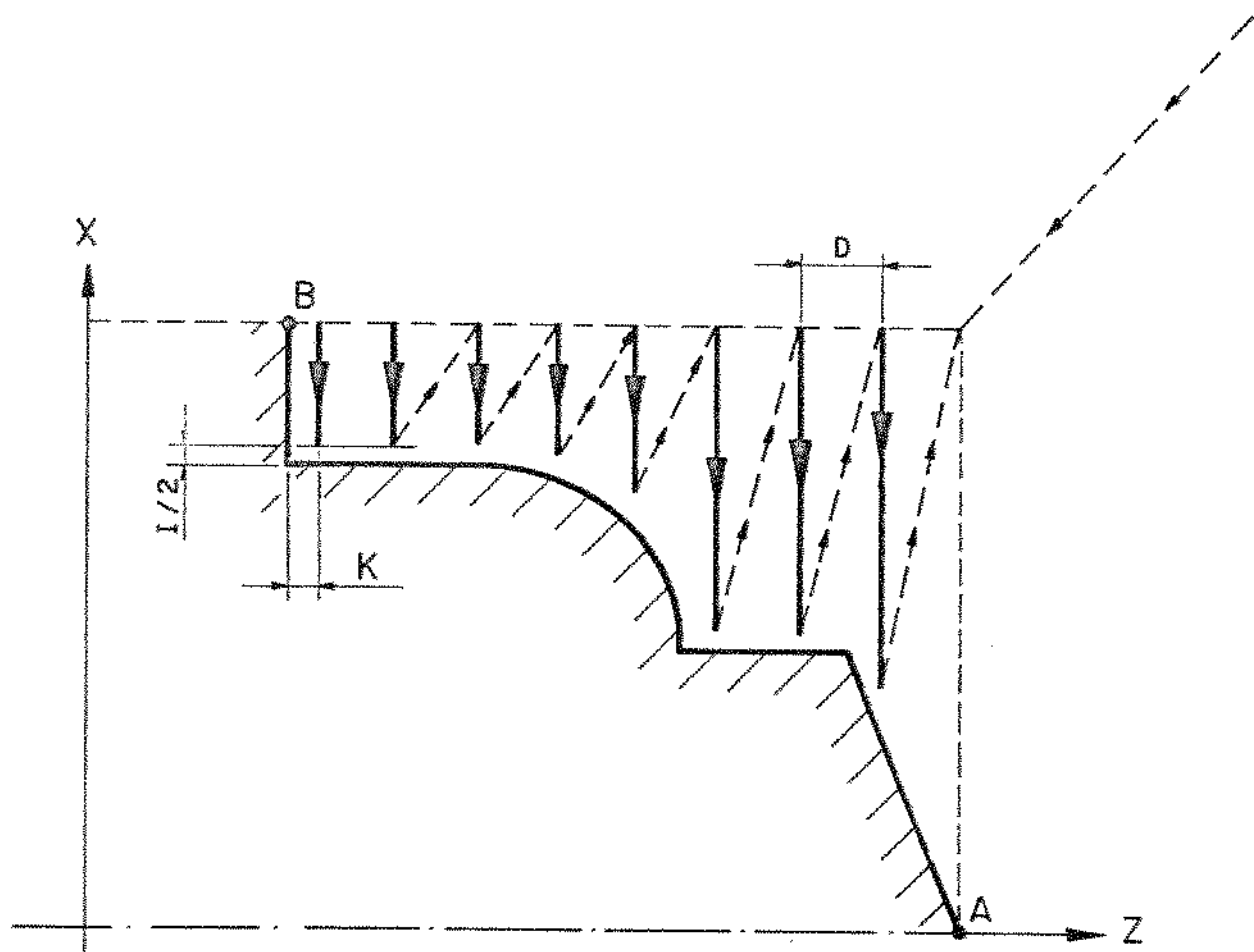
Localizada a linha, pela indicação do tipo do trecho (coluna 1 da matriz), é feito o desvio para a rotina correspondente: cone, cilindro, etc.

Calculadas as coordenadas dos pontos de intersecção, elas são incorporadas como parte do programa-objeto, de acordo com as regras de semântica e sintaxe do Controle em questão. Analogamente, são incluídos, os comandos e as instruções necessários para as aproximações e os afastamentos das ferramentas.

3.3 - CICLO DE DESBASTE TRANSVERSAL (G101): DESCRIÇÃO E FORMATO

Para peças tipicamente qualificadas como : peças de "placa", e matéria prima a partir de "blanques" simplesmente serrados, os programadores lançam mão do Ciclo de Desbaste exposto na Figura 3.5.

Analogamente ao Ciclo de Desbaste Longitudinal, é necessário, nesta situação, que sejam fornecidos os valores dos parâmetros requeridos pelo Processador de Geometria, tais como: avanço, sobre-materiais, profundidade de corte e exclusão/inclusão de pré-acabamento.



FORMATO:

N_{xxx}	G101	O _{AAA}	Q _{BBB}	I _{iii}	K _{kkk}	D _{ddd}	A	F _{fff}
-----------	------	------------------	------------------	------------------	------------------	------------------	---	------------------

ONDE: xxx = Nº DO BLOCO CORRENTE

AAA = Nº DO BLOCO INÍCIO DO ACABAMENTO

BBB = Nº DO BLOCO FIM DO ACABAMENTO

iii = SOBREMÉTAL NA DIREÇÃO X [EM Ø]

kkk = SOBREMÉTAL NA DIREÇÃO Z

ddd = PROFUNDIDADE DE CORTE (DIREÇÃO Z)

A = ALISAMENTO (FACULTATIVO)

fff = VALOR DO AVANÇO

FIGURA 3.5-Ciclo de Desbaste Transversal-G101.

Na situação de ser especificado o pré-acabamento, o Processador cuida para que o mesmo seja efetuado, do ponto B para o ponto A, tendo em vista que, muito provavelmente, uma ferramenta típica de desbaste transversal não teria condições de executar esse trabalho no sentido de A para B.

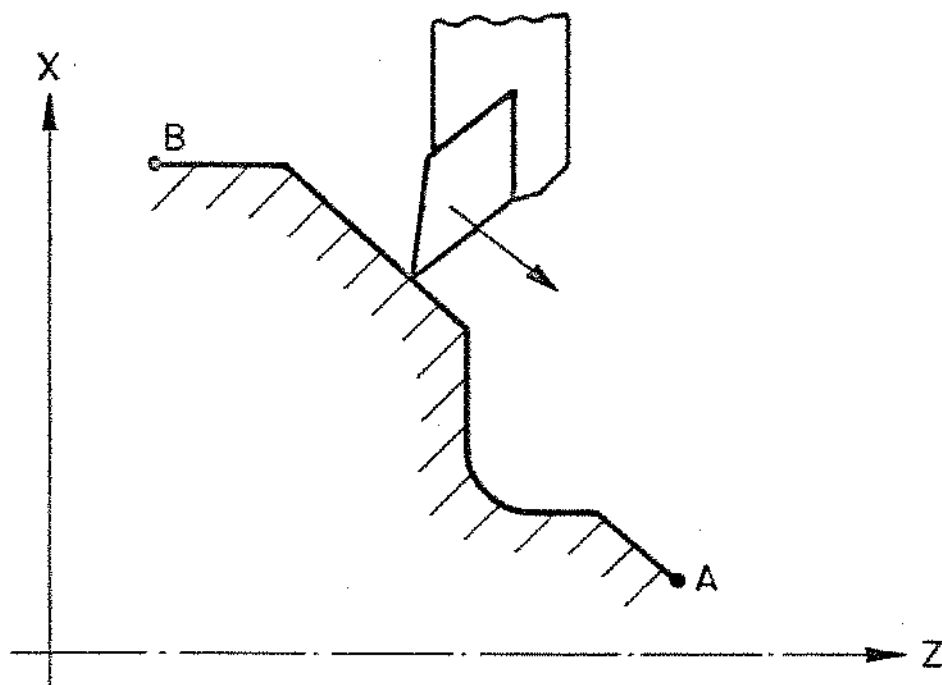


FIGURA 3.6-Ferramenta típica para desbaste transversal.

3.3.1 - CICLO DE DESBASTE TRANSVERSAL (G101): ALGORÍTMOS E ENFOQUE COMPUTACIONAL

A montagem da matriz MATFUNC é feita da mesma forma que o descrito no item 3.2.1.

A situação agora modifica-se, tendo-se como variável independente Z_1 , e X_1 como variável dependente, de acordo com a Figura 3.7

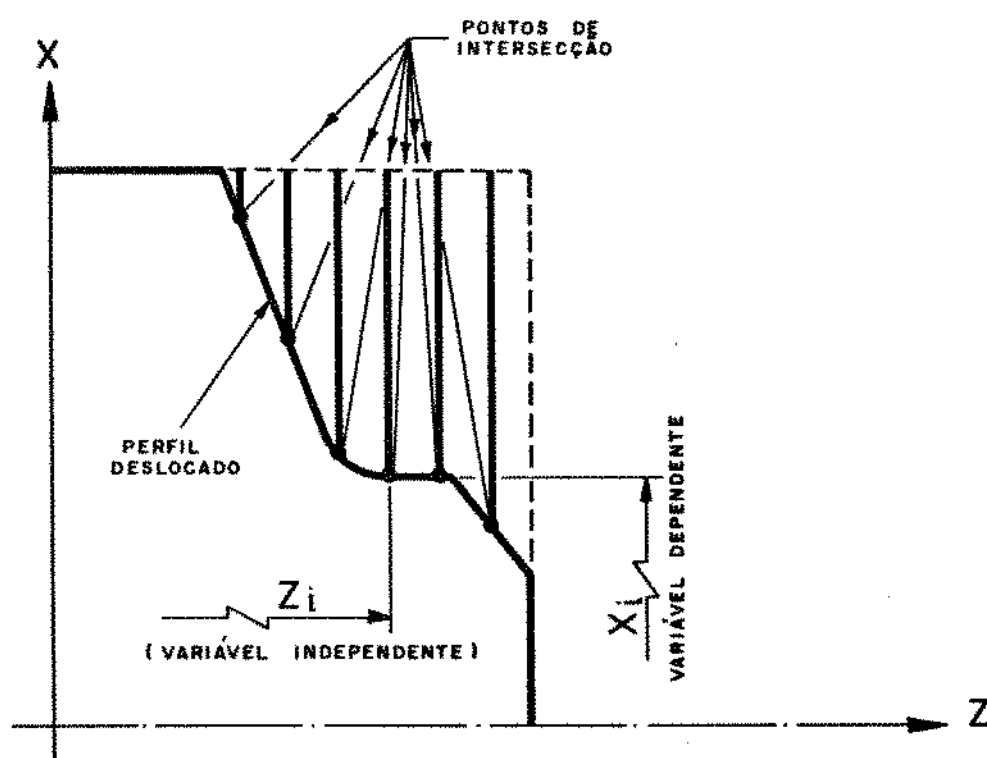


FIGURA 3.7-Pontos de Intersecção - Ciclo G101.

O valor de X_i é determinado em função de Z_i , que, uma vez fixado, obriga o computador determinar a linha da matriz que contenha o intervalo $Z[2] \cdots Z[1]$, para o qual Z_i é intermediário.

Determinada a linha da matriz, pelo tipo do trecho (coluna 1), é feito o desvio para a rotina correspondente: cone, cilindro, etc.

Tendo-se as coordenadas das intersecções, elas são integradas ao programa-objeto, assim como os comandos correspondentes às aproximações e aos afastamentos das ferramentas.

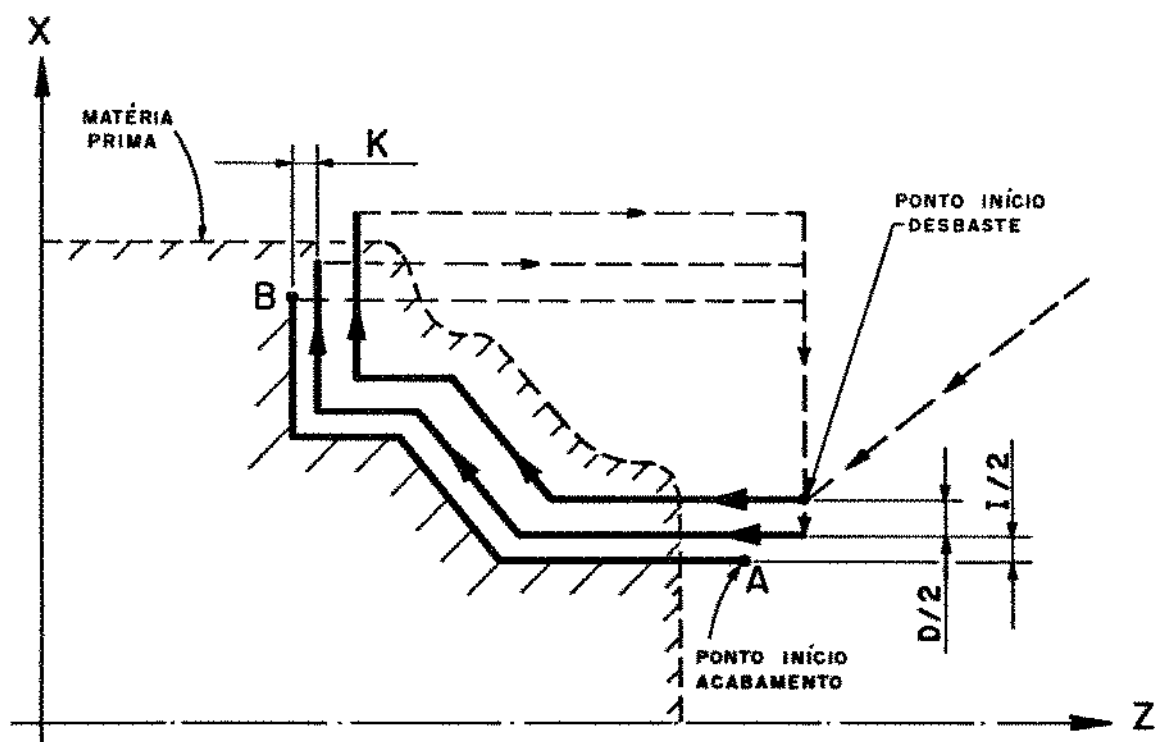
3.4 - CICLO DE DESBASTE DE CONTORNOS (G102): DESCRIÇÃO E FORMATO

São, de certo modo, comuns operações de torneamento, cuja matéria prima constitui-se de fundido ou forjado, com sobre-material relativamente uniforme [1].

Para atender tal necessidade de programação, desenvolveu-se o Ciclo de Desbaste de Contornos, esquematizado na Figura 3.8., que caracteriza-se por sequência de desbastes "paralelos" ao perfil final da peça.

Para utilização desse ciclo, é necessário que a matéria-prima, que alimenta o torno, tenha o sobrematerial convenientemente definido e uniforme, de forma que durante os passes de torneamento não ocorram sobre-cargas em operações, ou passes parcialmente em vazio.

Uma especificação adequada de sobrematerial, seguida de controle dimensional do forjado ou fundido, irá proporcionar no máximo 2 passes em desbaste.



FORMATO:

$N_{xxx} G102 O_{AAA} Q_{BBB} I_{iii} K_{kkk} D_{ddd} F_{fff}$

ONDE : xxx = N° DO BLOCO CORRENTE

AAA = N° DO BLOCO INÍCIO DO ACABAMENTO

BBB = N° DO BLOCO FIM DO ACABAMENTO

iii = SOBREMÉTAL NA DIREÇÃO x [EM Ø]

kkk = SOBREMÉTAL NA DIREÇÃO z

ddd = PROFUNDIDADE DE CORTE [NO Ø]

fff = VALOR DO AVANÇO

FIGURA 3.8-Ciclo de Desbaste de Contornos-G102.

Podemos observar que, para esse ciclo, não existe a necessidade da inclusão do parâmetro A, pois o último passe de desbaste equivale a um pré-acabamento.

É necessário que o programador especifique o ponto de início da "chamada" do ciclo G102, com deslocamentos coerentes (eixo X, e eixo Z), para a entrada da ferramenta ocorrer sem possibilidade de colisão.

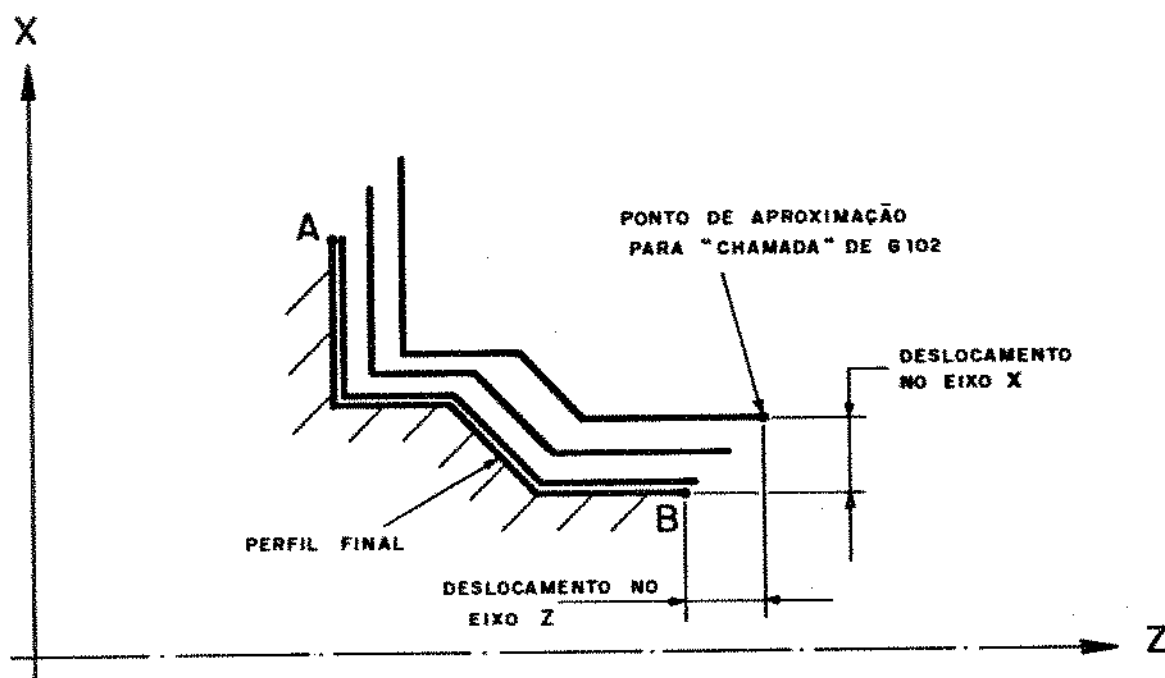


FIGURA 3.9-Deslocamentos requeridos para a utilização do ciclo G102.

3.4.1 - CICLO DE DESBASTE DE CONTORNOS (G102). ALGORITMOS E ENFOQUE COMPUTACIONAL

Após a montagem da matriz MATFUNC, que retrata numericamente o perfil final "deslocado da peça", determina-se o número (N) de passes necessários para o desbaste, em função da posição relativa dos pontos de início do desbaste e do início do acabamento e, evidentemente, da profundidade de corte $D/2$.

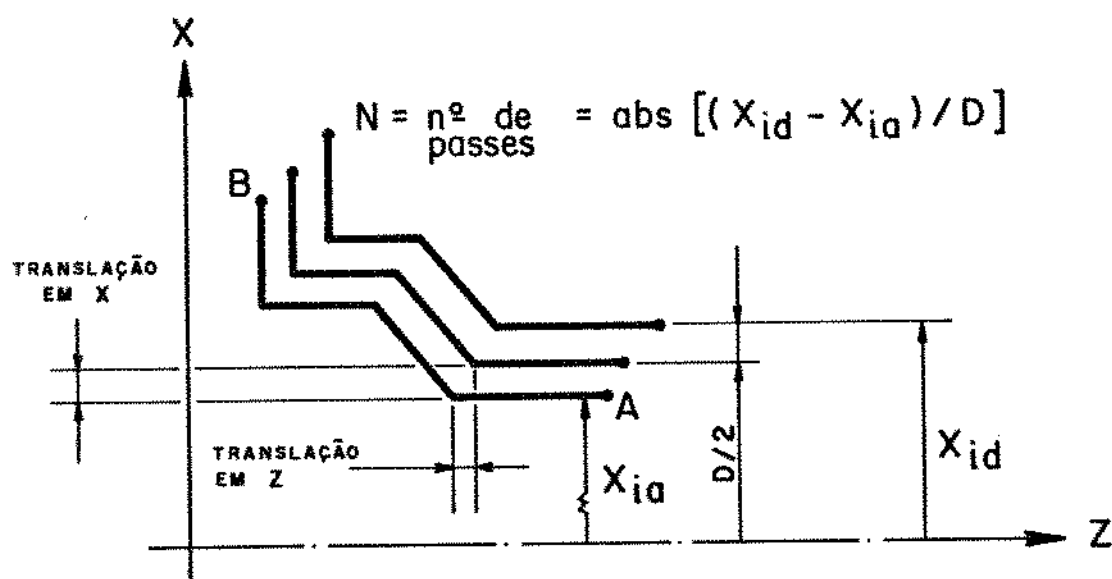


FIGURA 3.10-Determinação do número de passes de desbaste (Ciclo G102).

Calculado o número (N) de passes, o processador gera o "perfil-deslocado" correspondente a cada passe, "consultando" a matriz MATFUNC e transladando convenientemente as coordenadas X e Z de cada trecho em questão.

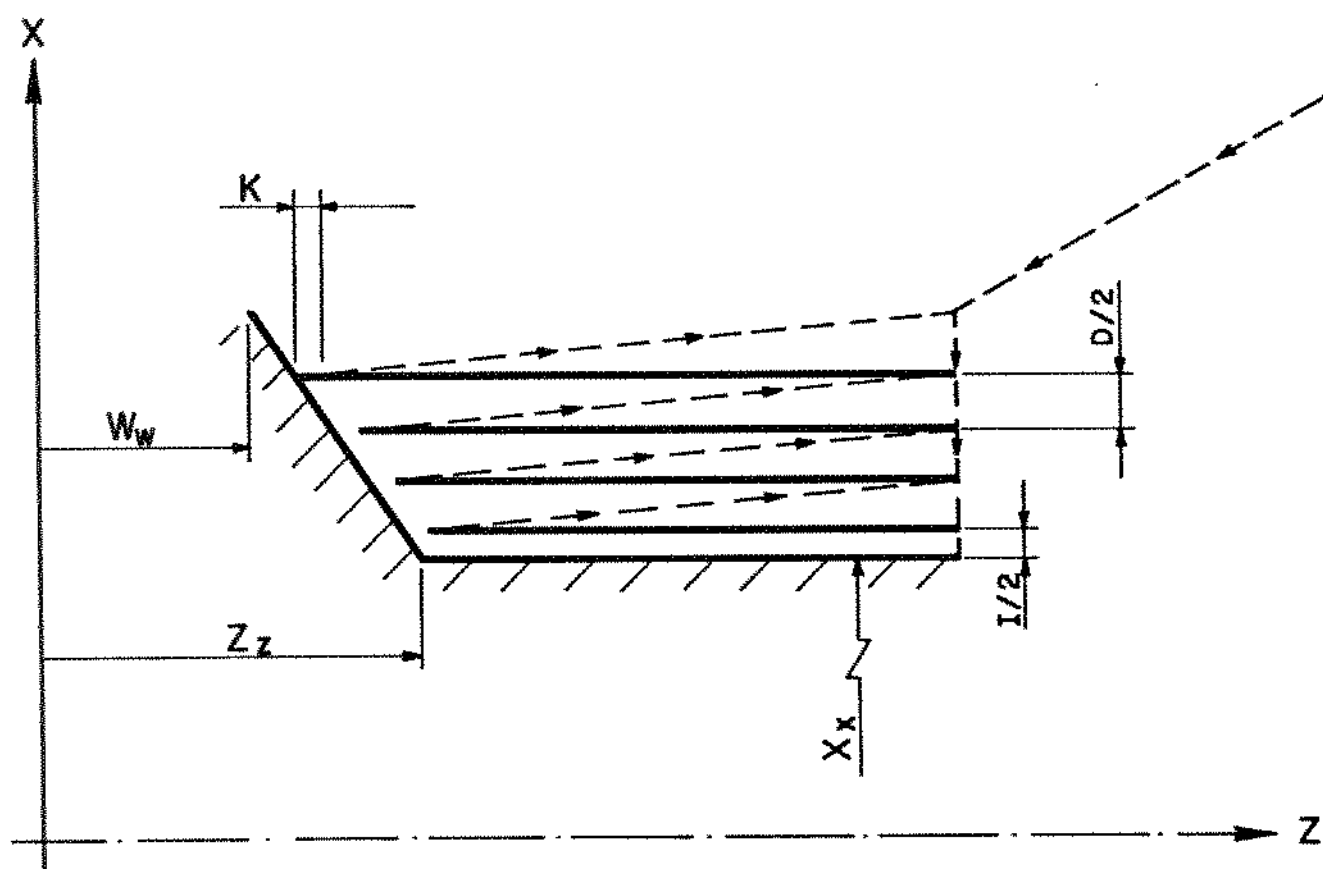
O processador encarrega-se, também, de gerar os comandos referentes às aproximações e afastamentos da ferramenta.

3.5 - CICLO DE DESBASTE LONGITUDINAL TRAPEZOIDAL (G103): DESCRIÇÃO E FORMATO

Nos ciclos anteriores (G100, G101 e G102), o desbaste é executado "respeitando-se" um lugar geométrico que corresponde ao perfil final da peça, o qual é refletido, numericamente, na matriz MATFUNC. O ciclo G103 não requer a constituição da matriz MATFUNC, uma vez que todas as informações necessárias para o tratamento computacional estão disponíveis na própria "chamada" do ciclo.

A utilização do ciclo G103 é, então, feita quando há interesse em desbaste simples do tipo cilíndrico, cuja sucessão proporciona um final cônico.

A necessidade de especificação de um ciclo de desbaste longitudinal retangular é obtida especificando-se: $Ww = Zz$.



FORMATO : N_{xxx} G103 X_x Z_z W_w D_{ddd} I_{iii} K_{kkk} [A] F_{fff}

ONDE : xxx = Nº DO BLOCO CORRENTE

x = DIÂMETRO MENOR

w = DIMENSÃO Z MENOR

z = DIMENSÃO Z MAIOR

iii = SOBREMETAL NA DIREÇÃO x [EM Ø]

kkk = SOBREMETAL NA DIREÇÃO z

A = ALISAMENTO

fff = VALOR DO AVANÇO

ddd = PROFUNDIDADE DE CORTE [NO Ø]

FIGURA 3.11-Ciclo de Desbaste Longitudinal Trapezoidal.

3.5.1 - CICLO DESBASTE LONGITUDINAL TRAPEZOIDAL (G103): ALGORITMOS E ENFOQUE COMPUTACIONAL

A diferença entre o valor da coordenada X inicial e X_n (Figura 3.11) e a profundidade de corte ($D/2$) caracterizam o número requerido de passes.

Para o i -ésimo passe, temos a coordenada X_i que é utilizada para determinar o valor Z_i , que corresponde ao ponto meta da parte final do desbaste.

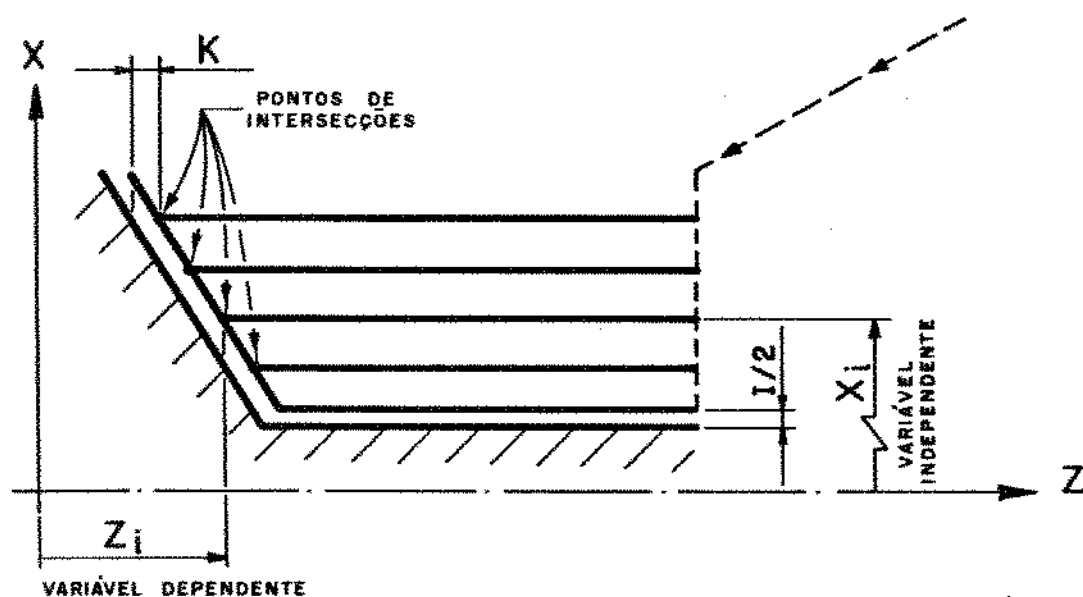


FIGURA 3.12-Pontos de intersecções para o ciclo G103.

O cálculo da coordenada Z_i do i -ésimo passe é feito, também, considerando-se os sobre-materiais na direção X ($I/2$) e na direção $Z(K)$.

Calculadas as coordenadas dos pontos de intersecções, o processador incorpora-as ao programa-objeto, seguindo as regras de semântica e sintaxe do Controle em questão. Também, são

incorporados ao programa objeto, os comandos relativos às aproximações e afastamentos.

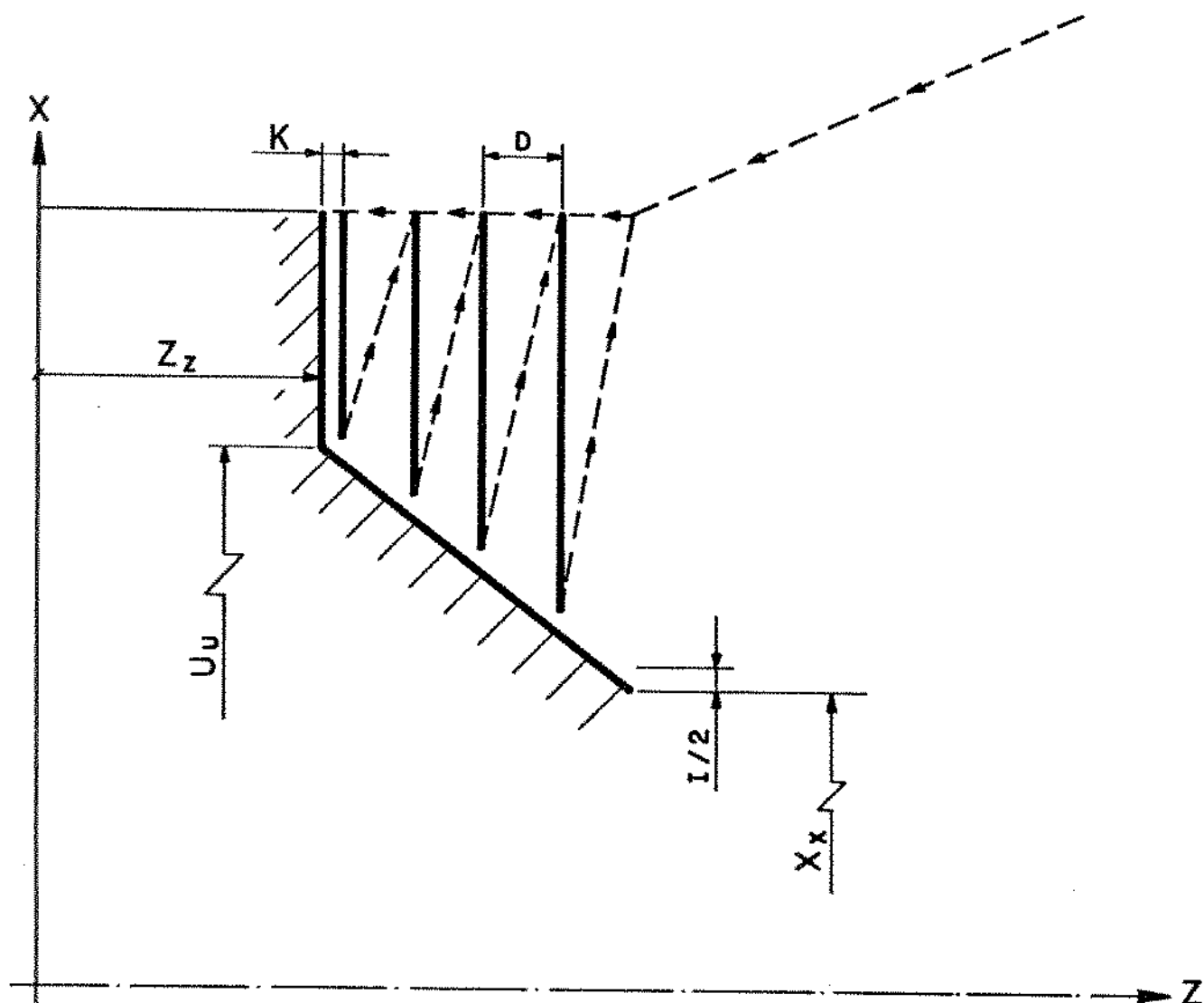
3.6 - CICLO DE DESBASTE TRANSVERSAL TRAPEZOIDAL (G104): DESCRIÇÃO E FORMATO

Quando da usinagem de peças de "placa", é usual a necessidade de um ciclo simples de desbaste transversal com final cônico. Da mesma forma que no ciclo G103, todas as informações requeridas para o tratamento computacional, estão disponíveis na própria "chamada" do ciclo.

Analogamente ao Ciclo G103, esse Ciclo não "monta" a matriz MATFUNC, pois todos os elementos, necessários para o tratamento computacional, são fornecidos quando da "chamada" do ciclo.

Devido a sua maior simplicidade de tratamento computacional, os ciclos G103 e G104, podem ser "chamados", tantas vezes quanto for necessário, de forma distinta dos ciclos G100, G101 e G102, que só podem ser acionados uma única vez, por programa.

A obtenção de um ciclo de desbaste transversal retangular é conseguida especificando-se: $Xx = Uu$.



FORMATO: N_{xxx} G104 X_x Z_z U_u D_{ddd} I_{iii} K_{kkk} [A]

ONDE: xxx = N° DO BLOCO CORRENTE

x = DIÂMETRO MENOR

z = DIMENSÃO Z FINAL

u = DIÂMETRO MAIOR

iii = SOBREMETAL NA DIREÇÃO X [EM Ø]

kkk = SOBREMETAL NA DIREÇÃO Z

ddd = PROFUNDIDADE DE CORTE (DIREÇÃO)

A = ALISAMENTO

FIGURA 3.13-Ciclo de Desbaste Transversal Trapezoidal.

3.6.1 - CICLO DE DESBASTE TRANSVERSAL TRAPEZOIDAL (G104):

ALGORITMOS E ENFOQUE COMPUTACIONAL

O número (N) de passes de desbaste é aqui determinado pela diferença da coordenada Z inicial e a coordenada Z_{m} (Figura 3.13).

Como variável independente temos a coordenada Z_i que, no i -ésimo passe, tem em correspondência um ponto X_i no final do desbaste.

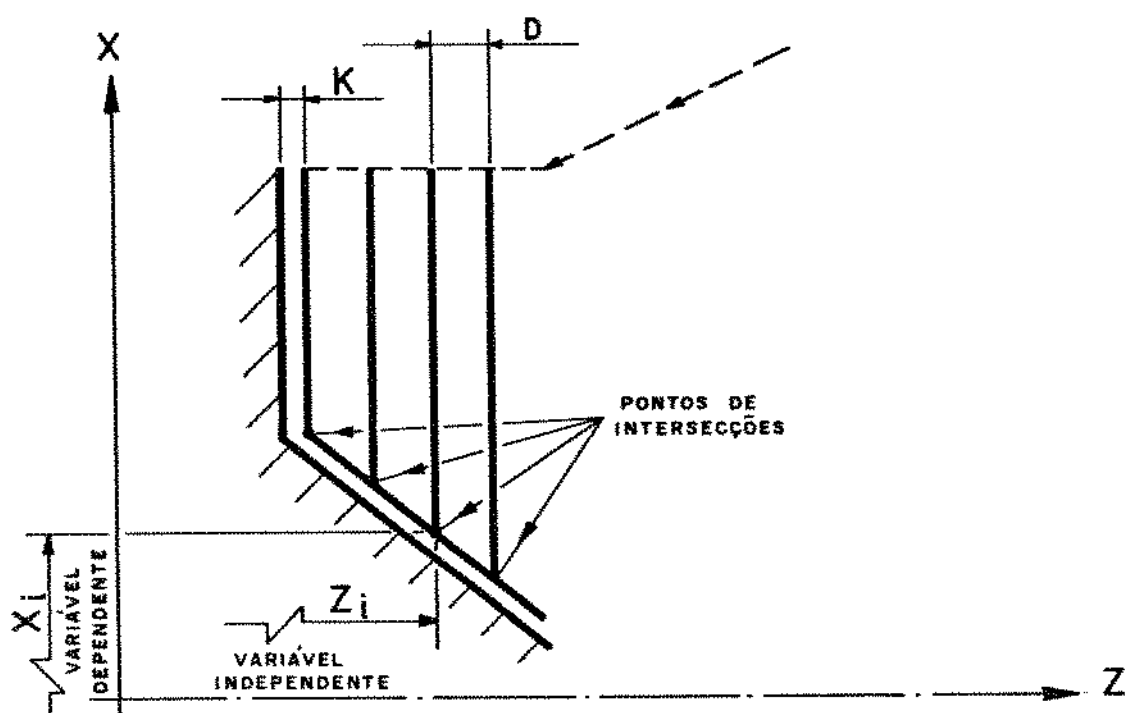


FIGURA 3.14-Pontos de Interseccões para o Ciclo G104.

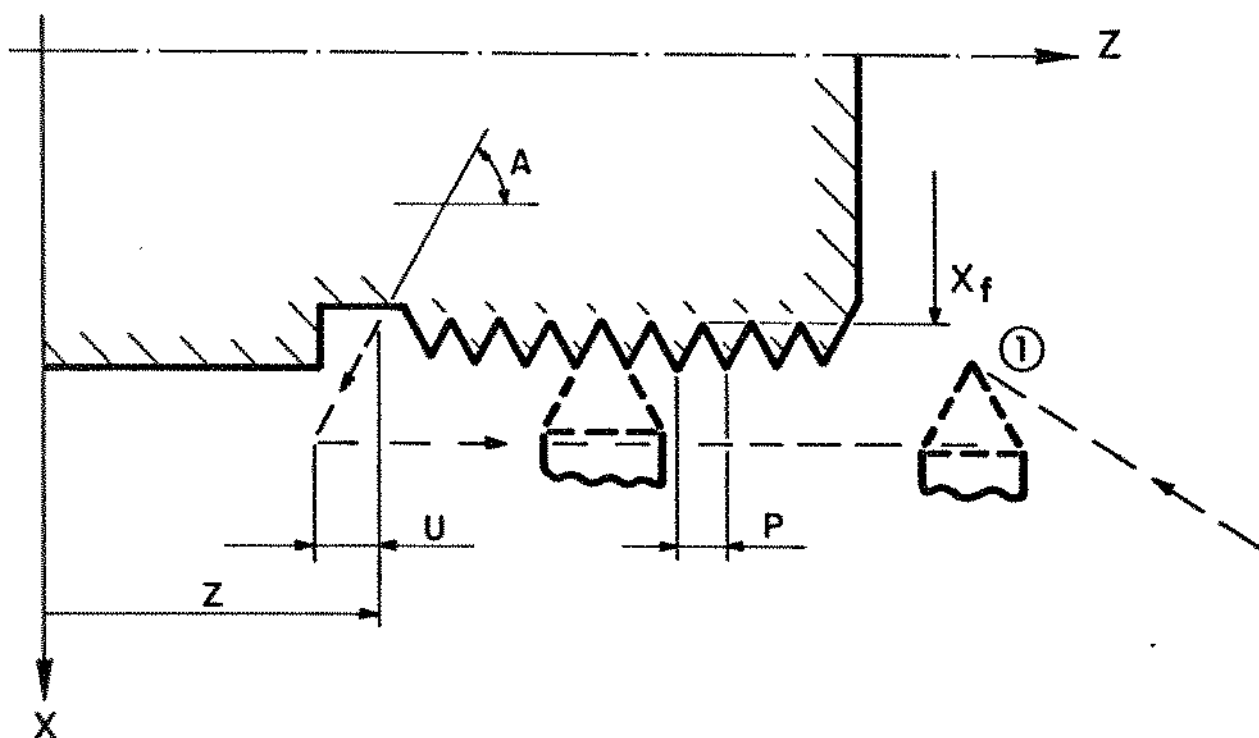
As coordenadas X_1 são determinadas em função da inclinação da parte cônica do desbaste, da profundidade de corte (D) e da especificação de sobre-materiais (I/2 e K).

A montagem do programa-objeto é feita então utilizando-se as coordenadas determinadas, dentro do conjunto de regras de semântica e sintaxe do Controle em questão.

3.7 - CICLO DE CORTE DE ROSCA (PAT 105): DESCRIÇÃO E FORMATO

Os tornos controlados numericamente possibilitam o corte da rosca através de passes sucessivos de ferramenta monocortante. A distribuição criteriosa dos passes possibilita: a obtenção de roscas de qualidade, ciclos curtos de trabalho e otimização da vida da ferramenta [6].

Os recursos de sincronismo eletro/mecânico, com que os tornos CN são equipados, permitem que o Processador de Geometria otimize o corte da rosca, através da "entrada" conveniente da ferramenta para cada passe da rosca (Figura 3.16).



FORMATO :

xxx. PAT 105 X_f Z_f P_p A_a U_u N_n

ONDE : xxx É O BLOCO CORRENTE

① É PONTO DE POSICIONAMENTO PRÉVIO

X_f É O RAIO FINAL DA ROSCA

Z_f É POSIÇÃO Z FINAL DA ROSCA

P É PASSO DA ROSCA

A É ÂNGULO DE SAÍDA EM DESACELERAÇÃO

U É A DESACELERAÇÃO EM Z, DO FINAL DA ROSCA

N É O NÚMERO TOTAL DE PASSADAS

FIGURA 3.15-Ciclo de corte de rosca - PAT 105.

3.7.1 - CICLO DE CORTE DE ROSCA (PAT 105): ALGORITMOS E ENFOQUE COMPUTACIONAL

Em função do par ferramenta-peça, tem-se a recomendação tecnológica para o número total de passadas (n).

A profundidade de corte para a i -ésima passada é determinada pelo critério de mesma secção de corte em cada passada e é obtida, segundo MACHADO [5] por:

$$h_i = h_t \sqrt{i/n} \quad 3.1$$

onde: h_i é a profundidade de corte de cada passada.

h_t é a altura total do filete da rosca.

n é o número total de passadas.

A adoção desse critério otimiza o corte da rosca, pois com a mesma secção de corte em cada passada, têm-se aproximadamente os mesmos esforços atuantes na ferramenta.

Outro aspecto, que o processador de geometria viabilizou, foi o chamado: "corte-de-rosca-composto", obtido deslocando-se convenientemente a ferramenta no "ponto de entrada" de cada passada.

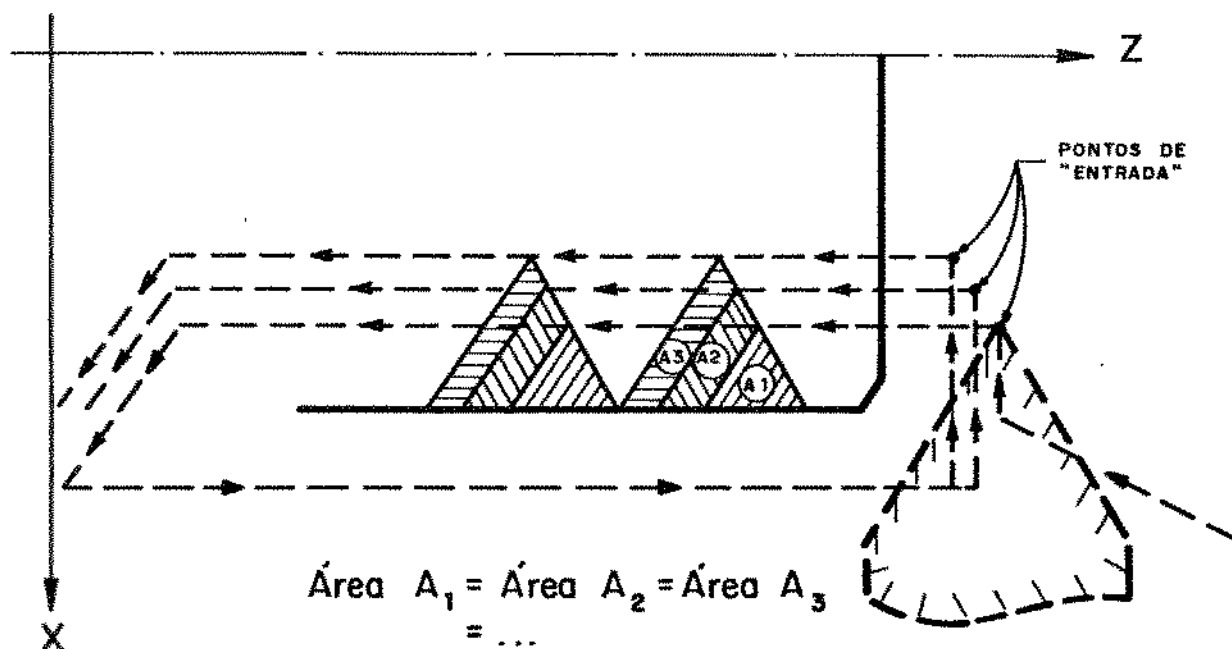


FIGURA 3.16-Corte de rosca composto.

Na realidade, a determinação dos pontos de entrada dos diversos "passes" da rosca é um pouco mais elaborado que o descrito na Figura 3.16, pois, os pontos de entrada deslocam-se alternadamente para a esquerda e para a direita, de maneira a utilizar, ora uma aresta de corte, ora outra aresta de corte, possibilitando, portando, a duplicação da vida útil da ferramenta de corte de rosca. Esse deslocamento possibilita a remoção do efeito de "cunha" pelo encravamento das duas arestas de corte no material. Dessa maneira, a ferramenta tem uma das arestas trabalhando integralmente, enquanto a outra opera com um comprimento-em-corte bastante reduzido.

3.8 - EXEMPLO APLICATIVO DO PROCESSADOR GEOMÉTRICO

As benesses advindas da utilização do Processador Geométrico ficam evidentes através de sua utilização em peça-didática (Figura 3.17).

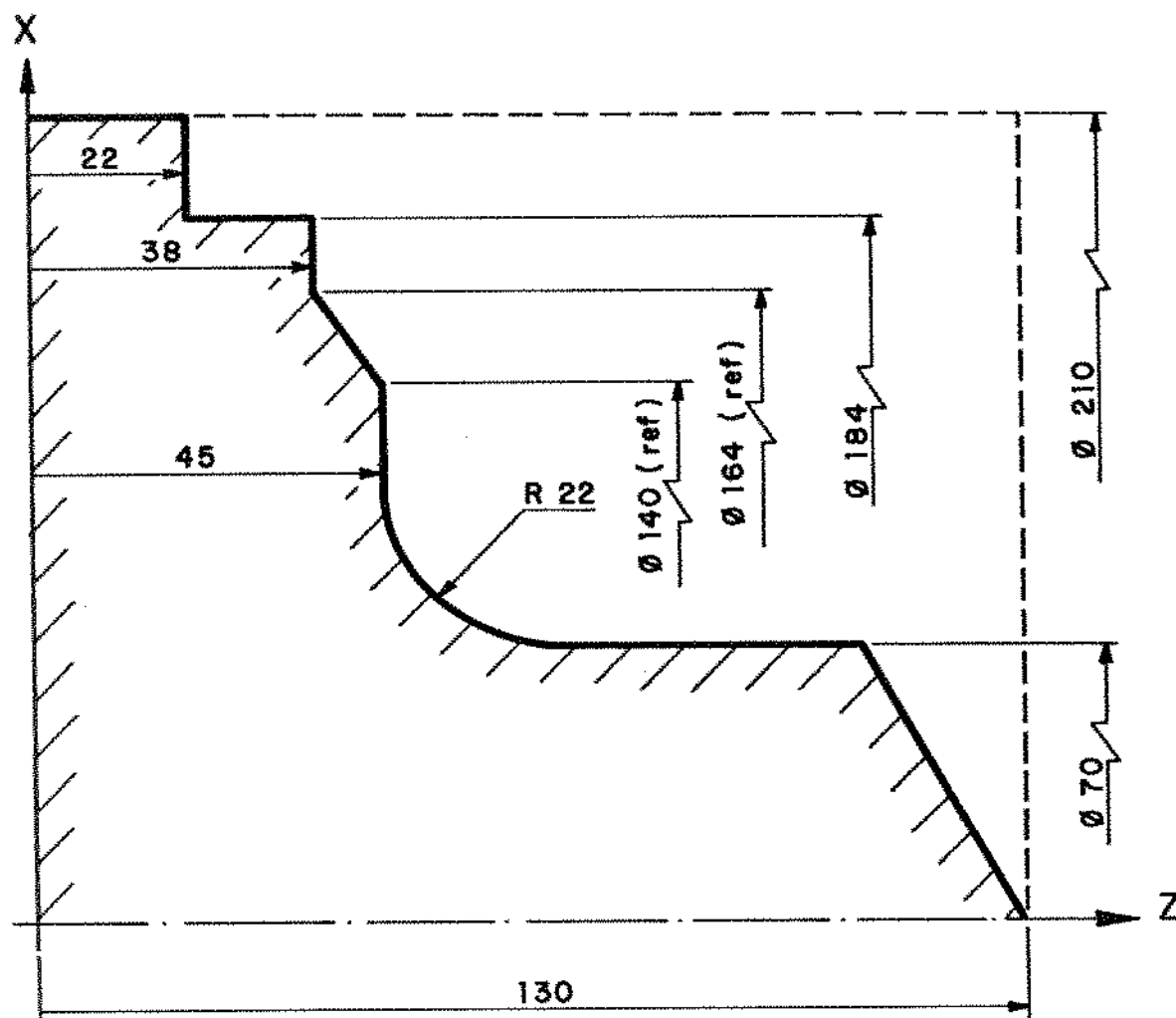


FIGURA 3.17-Peça didática-típica.

O programa-fonte que pode ser utilizado para o torneamento dessa peça em um torno GALAXY, de fabricação da ROMI [7], tem a forma seguinte:

```

N10;EXEMPLO.DE.PROGRAMACAO.AUTOMATICA
N20G99
N30T0101
N40GXZ
N50M6
N60G92X280.Z200.
N70G96
N80S140.
N90G92S3000M4
N100GX210.Z133.
N110G1000210Q290I.5K.1D8.AF.3
N120GX300.Z215.
N130G99
N140T0303
N150GXZ
N160M6
N170G92X320.Z197.
N180G96
N190S180.
N200G92S3000M4
N210GXZ133.
N220G1Z130.F.15
N230X70.Z110.
N240G1Z45.R22.
N250X140.
N260X164.Z38.
N270X184.
N280Z22.
N290X210.
N300X220.
N310GX320.Z197.
N320T0101
N330G99
N340GXZ
N350M6
N360M30

```

O programa-fonte anterior foi escrito no padrão G e consta dos comandos:

Blocos 10 a 90, comandos referentes a definição do "zero-peça", da ferramenta de desbaste e das condições de corte.

Bloco 100 comanda a aproximação da ferramenta de desbaste.

No bloco 110 temos a "chamada" do ciclo G100 (Macro-Instrução a ser geometricamente processada), com as especificações:

0210 indicação do bloco 210 como início do acabamento.
Q290 " " " 290 " fim " "
I.5 " de sobrematerial de 0,25mm na direção X.
K.1 " " " " 0,1mm " " Z.
DB. " " 4mm de profundidade de corte.
A " " necessidade de pré-acabamento.
F.3 " 0,3mm/volta de avanço.

Blocos 120 a 200, comandos referentes ao afastamento da ferramenta de desbaste, a seleção da ferramenta de acabamento, as especificações das condições de corte para o acabamento.

No bloco 210 temos o comando de aproximação da ferramenta de acabamento.

Blocos 220 a 290 comandam a execução do perfil final da peça, pela ferramenta de acabamento.

Blocos 300 a 360 referem-se, da torre porta-ferramentas, ao ponto de troca e encerramento do programa.

Editado o programa-fonte, aciona-se o Processador de Geometria, que substitui no programa-fonte a Macro-Instrução G100, pelo detalhamento do desbaste, fornecendo o programa-objeto:

N10; EXE. PROG. AUTOMATICA
 N20G99
 N30T0101
 N40GXZ
 N50M6
 N60G92X280. Z200.
 N70G96
 N80S140.
 N90G92S3000M4
 N100GX210. Z133.
 N110G00X202. 000Z133. 000
 N120G01Z22. 100F0. 300
 N130G00X210. 000Z133. 000
 N140G00X194. 000
 N150G01Z22. 100
 N160G00X202. 000Z133. 000
 N170G00X186. 000
 N180G01Z22. 100
 N190G00X194. 000Z133. 000
 N200G00X178. 000Z133. 000
 N210G01Z38. 100F0. 300
 N220G00X186. 000Z133. 000
 N230G00X170. 000
 N240G01Z38. 100
 N250G00X178. 000Z133. 000
 N260G00X162. 000Z133. 000
 N270G01Z38. 829F0. 300
 N280G00X170. 000Z133. 000
 N300G01Z41. 162
 N310G00X162. 000Z133. 000
 N320G00X146. 000
 N330G01Z43. 496
 N340G00X154. 000Z133. 000
 N350G00X138. 000Z133. 000
 N360G01Z45. 100F0. 300
 N370G00X146. 000Z133. 000
 N380G00X130. 000
 N390G01Z45. 100
 N400G00X138. 000Z133. 000
 N410G00X122. 000
 N420G01Z45. 100
 N430G00X130. 000Z133. 000
 N440G00X114. 000Z133. 000
 N450G01X114. 000Z45. 101F. 3
 N460G00X122. 000Z133. 000
 N470G00X106. 000
 N480G01X106. 000Z45. 514
 N490G00X114. 000Z133. 000
 N500G00X98. 000
 N510G01X98. 000Z46. 705
 N520G00X106. 000Z133. 000
 N530G00X90. 000
 N540G01X90. 000Z48. 826
 N550G00X98. 000Z133. 000
 N560G00X82. 000
 N570G01X82. 000Z52. 270
 N580G00X90. 000Z133. 000
 N590G00X74. 000
 N600G01X74. 000Z58. 501
 N610G00X82. 000Z133. 000
 N620G00X66. 000Z133. 000
 N630G01Z111. 386F0. 300
 N640G00X74. 000Z133. 000

N650G00X58. 000
 N660G01Z113. 671
 N670G00X66. 000Z133. 000
 N680G00X50. 000
 N690G01Z115. 957
 N700G00X58. 000Z133. 000
 N710G00X42. 000
 N720G01Z118. 243
 N730G00X50. 000Z133. 000
 N740G00X34. 000
 N750G01Z120. 529
 N760G00X42. 000Z133. 000
 N770G00X26. 000
 N780G01Z122. 814
 N790G00X34. 000Z133. 000
 N800G00X18. 000
 N810G01Z125. 100
 N820G00X26. 000Z133. 000
 N830G00X10. 000
 N840G01Z127. 386
 N850G00X18. 000Z133. 000
 N860G00X2. 000
 N870G01Z129. 671
 N880G00X10. 000Z133. 000
 N890G00Z133. 100
 N900G00X0. 500
 N910G01X0. 500Z133. 100
 N920G01X0. 500Z130. 100
 N930G01X70. 500Z110. 100
 N940G01X70. 500Z67. 100
 N950G2X114. 500Z45. 100I114. 500K67. 100
 N960G01X140. 500Z45. 100
 N970G01X164. 500Z38. 100
 N980G01X184. 500Z38. 100
 N990G01X184. 500Z22. 100
 N1000G01X210. 500Z22. 100
 N1010G00Z133. 100
 N1020GX300. Z215.
 N1030G99
 N1040T0303
 N1050GXZ
 N1060M6
 N1070G92X320. Z197.
 N1080G96
 N1090S180.
 N1100G92S3000M4
 N1110GXZ133.
 N1120G1Z130. F. 15
 N1130X70. Z110.
 N1140G01X70. 000Z67. 000
 N1150G2X114. 000Z45. 000R22. 000
 N1160G01X140. 000Z45. 000
 N1170X140.
 N1180X164. Z38.
 N1190X184.
 N1200Z22.
 N1210X210.
 N1220X220.
 N1230GX320. Z197.
 N1240T0101
 N1250G99
 N1260GXZ
 N1270M6
 N1280M30

Caso não tivéssemos disponível um processador geométrico, seria necessário escrevermos diretamente o programa-objeto, calculando-se todo o detalhamento do desbaste.

REFERÊNCIAS BIBLIOGRÁFICAS (CAP. 3)

- [1] Elements of Programming NC, Tolling and Production, p. 102, Apr. 1980.
- [2] IFAD - Institut fuer angewandte Organisationsforschung - Comando Numérico CNC [CNC-Ausbildung fuer die betrieblich Praxis Teil 3: Drehen], Editora Pedagógica Universitária, São Paulo, 1985.
- [3] GRANT, H.E. - Dispositivos em Usinagem [Jigs and Fixtures], Livraria Ciência e Tecnologia, São Paulo, 1982.
- [4] FERRARESI, D. - Fundamentos da Usinagem dos Metais, Editora Edgard Blucher Ltda., São Paulo, 1970.
- [5] MACHADO, A. - O Comando Numérico Aplicado às Máquinas Ferramenta, ícone Editora, São Paulo, 1986.
- [6] FREDERIKS, G. - A Threading Algorithm Suitable for Computer Numerically Controlled Turning Centers, Machine Tool Design, V. 18, p. 29, Great Britain, 1978.
- [7] INDÚSTRIA ROMI S/A - Programação de Torno a Controle Numérico, Máquinas & Ferramentas, Ano 1, N.5, p. 20, 1979.

CAPÍTULO 4

METODOLOGIA DO DESENVOLVIMENTO

4.1 - GENERALIDADES

O desenvolvimento de "softwares" aplicativos é feito utilizando técnicas que lembram, ao mesmo tempo, trabalho artesanal e técnicas da produção seriada [1].

No presente caso, adotou-se a chamada técnica de desenvolvimento "bottom-up" (em-camadas-de-cebola), caminhando-se, progressivamente, de módulos simples (e artesanais) para soluções específicas, na direção de módulos complexos com soluções genéricas (produção seriada) [2,3].

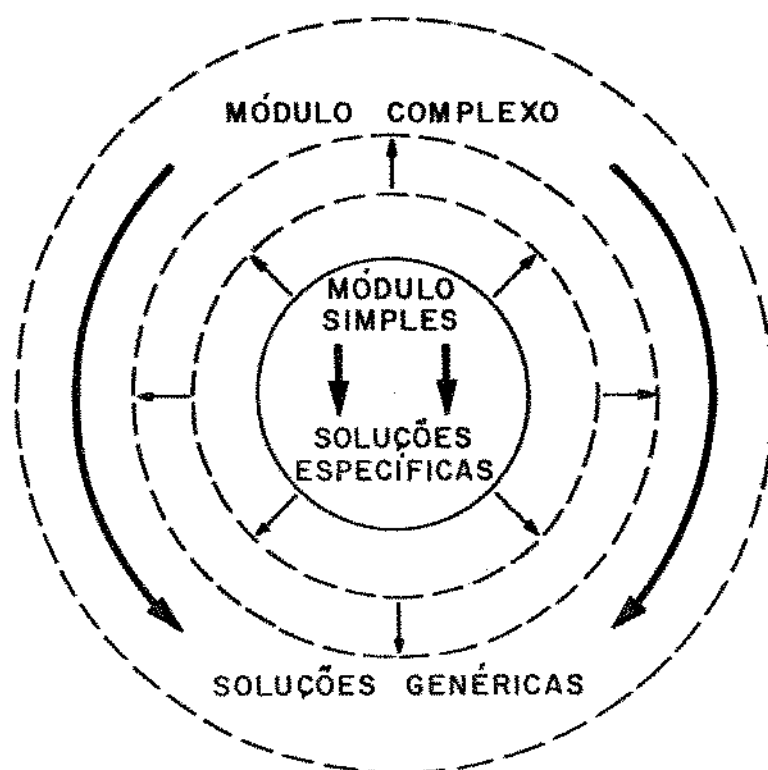


Figura 4.1-Desenvolvimento em camadas-
progressivas.

4.2 - ETAPAS DO DESENVOLVIMENTO

Como o Sistema em questão deve apresentar soluções genéricas, o teste de validação final foi realizado por uma amostragem do usuário final típico. Essa premissa retardou de certa forma, a conclusão do projeto, devido o envolvimento de pessoal não vinculado diretamente ao desenvolvimento do mesmo.

De toda forma, podemos relatar como etapas no desenvolvimento deste trabalho:

- 0) Especificação dos Atributos do SPAT;
- 1) Escolha da Linguagem e do "Hardware" para desenvolvimento;
- 2) Modelagem e Definição de Algoritmos;
- 3) Implementação (Codificação) de módulos simples;
- 4) Testes e depuração de módulos simples;
- 5) Desenvolvimento de módulos complexos;
- 6) Testes de aplicação com usuários típicos;
- 7) Análise de resultados e otimização do Sistema;
- 8) Elaboração de documentação.

4.2.1 - ESPECIFICAÇÃO DOS ATRIBUTOS DO SPAT [2],[5]

Para a especificação das características que o SPAT deveria possuir, consultou-se uma série de programadores CN de forma a ter-se disponível, após o término do projeto, uma ferramenta de programação CN que fornecesse subsídios reais para o dia a dia desses programadores e incorporasse, na sua estrutura, características inovadoras, numa tentativa de contribuição científica efetiva para a comunidade.

A determinação das metas também foi feita considerando-se a dimensão, a complexidade a ser atingida pelo "software" e a análise de pacotes já disponíveis, exigindo, portanto visitas a usuários, feiras e exposições, assim como consulta a manuais e catálogos.

A caracterização do grau de generalidade, portatibilidade e disponibilidade, para manutenção do "software", requereu uma pré-orientação para o universo de possíveis usuários do "software" constituído por empresas de médio porte, motivada por uma demanda real desse segmento. Outro aspecto, relativo ao direcionamento do "software" para empresas de médio porte, deve-se ao grande contingente abrangido por esse segmento da indústria Metal-Mecânica.

Na fase de especificações globais do software, utilizou-se também de macro-blocos para explicitação do tratamento computacional das diversas rotinas. Basicamente, esses blocos indicavam a listagem dos dados de entradas e saídas, e suas possíveis correlações com outras rotinas.

A conclusão principal obtida nesta etapa do trabalho é de que o ponto inicial do trabalho é realmente a caracterização do usuário-alvo e, juntamente com esse, relacionar as tarefas a serem cumpridas pelo "software" a ser projetado. A troca de informações com profissionais da área é então vital para a maturação das metas a serem atingidas.

4.2.2 - ESCOLHA DA LINGUAGEM E DO HARDWARE PARA DESENVOLVIMENTO

A linguagem computacional escolhida para desenvolvimento do SPAT, foi o PASCAL, por tratar-se de linguagem que permite alto grau de estruturação, possibilitando a evolução do Sistema no esquema "bottom-up", onde, em princípio, tudo o que já foi desenvolvido poderá "migrar" como sub-sistema de algo mais abrangente, a ser desenvolvido posteriormente.

Esta metodologia de desenvolvimento permite que etapas do cronograma de trabalho possam ser efetivamente avaliadas, obtendo-se a aprovação de módulos parciais do sistema, o que permitiu o vislumbre antecipado da real possibilidade do cumprimento das metas inicialmente estipuladas.

Outro aspecto que pesou favoravelmente pró PASCAL, refere-se à facilidade de "manutenção" do "software" gerado com essa linguagem, sendo que o termo manutenção aplicado a "software" difere do seu contexto corriqueiro, uma vez que "software não quebra" [2].

A facilidade na manutenção do "software" refere-se aqui muito mais à "manutenção adaptativa", para situações não estipuladas inicialmente, do que "manutenção corretiva". A "manutenção adaptativa" é requerida principalmente para adequação do "software" em ambientes específicos, sendo extremamente difícil o desenvolvimento de pacotes (de média a alta complexidade) que, rigorosamente, independam do ambiente final e particular de utilização.

A opção por micro-computadores do tipo IBM PC Xt deveu-se ao fato de tratar-se do equipamento computacional de baixo custo e de grande aceitação, em nosso país.

A inevitável comparação entre esse desenvolvimento e experiências anteriores [4], onde utilizamos "hardware" e linguagem distintos (computador Apple e linguagem BASIC), revela alguns aspectos interessantes:

- a) 80% do trabalho de desenvolvimento independe do "hardware" hospedeiro, característica esta interessante, porque, em princípio "hardwares" modestos também podem ser utilizados em aplicações industriais. Na maior parte dos casos, a utilização de "hardwares" bem mais onerosos implica apenas em ganho de velocidade computacional, o que nem sempre é vital;
- b) A utilização de linguagens estruturadas constitui-se em notável ferramenta de desenvolvimento de "software", requerendo um tempo inicial maior para implementação, mas extremamente rápida para remoção de "bugs", detectados em testes. Um aspecto importante a ser ressaltado é de que o PASCAL distingue variáveis locais de variáveis globais o que permite "imunizar" o Sistema do perigoso efeito colateral advindo de alterações para sanar erros específicos. Outra característica interessante da linguagem PASCAL reside na necessidade de explicitação prévia da natureza das variáveis a serem empregadas, o que obriga naturalmente a um planejamento inicial mais detalhado e preciso, e que redundará numa versão inicial do "software" com menos chances de problemas.

4.2.3 - MODELAGEM, DEFINIÇÃO DE ALGORITMOS E IMPLEMENTAÇÃO DE MÓDULOS SIMPLES

Para a modelagem e definição dos algoritmos estipulou-se as metas e os dados que os mesmos deveriam tratar. A técnica empregada para implementação foi a denominada "walktought", que consta do procedimento sistemático do tipo : implementação-teste-otimização-teste-...

Testes de desempenho de "sub-sistemas" e testes na etapa final do Sistema completo foram exaustivamente utilizados, com as seguintes conclusões:

- a) Um teste é realmente eficaz, quando consegue provar que o Sistema falhou [2];
- b) Existem "softwares" ruins, bons e ótimos, sendo utópica a pretensão de obtenção de "softwares" perfeitos. O consenso parece concordar na assertiva de considerar-se "software" como algo vivo e dinâmico que possa sempre ser "melhorado", mas que também possa ser considerado concluído em função de sua aprovação em teste de validação para as principais estruturas de dados;
- c) Testes de "software" são muito mais produtivos quando testadores não têm compromisso com o desenvolvimento do mesmo, podendo aplicar testes funcionais do tipo "caixa-preta", onde se verifica que, para entradas corretas, temos ou não saídas também corretas;
- d) O grande problema para desenvolvimento de sistemas computacionais, de complexidade média ou grande, reside no altíssimo número de casos a serem testados, requerendo-se, portanto, critérios de definição de baterias de testes que cubram as situações prioritariamente previstas na fase de estipulação dos atributos a serem cumpridos pelo "software".

4.2.4 - DESENVOLVIMENTO DE MÓDULOS COMPLEXOS

Após a conclusão dos testes de validação dos módulos simples, que funcionavam a contento na situação "stand-alone", buscou-se a integração dos mesmos, o que demonstrou a necessidade de:

- a) Reestruturação parcial de alguns módulos de forma a poderem integrar-se com outros módulos. Essa reestruturação, muitas vezes, consumiu mais tempo do que o inicialmente dispendido para sua conclusão como módulo simples;
- b) Desenvolvimento de "interfaces" de acoplamento entre módulos;
- c) Implementação de novos módulos cuja necessidade não fora inicialmente prevista.

Ao término dessa fase, fechou-se uma versão experimental do Sistema, a qual demonstrou ainda uma grande quantidade de problemas, detectados pelos primeiros usuários.

4.2.5 - TESTES EM USUÁRIO TÍPICO E OTIMIZAÇÃO DO SISTEMA

A estratégia nessa fase consistiu em implantar-se o Sistema em 2 empresas, sendo que, através de relatórios colhidos quinzenalmente (num espaço de 6 meses), procurou-se sanar os problemas detectados, conseguindo-se assim versões evoluídas do Sistema.

A interpretação desses relatórios evidenciou problemas classificados em tipos:

- A) Falha real do Sistema, detectada através de utilização correta;
- B) Falha aparente do Sistema, motivada por utilização incorreta;
- C) Falha por omissão, com o Sistema não suprimindo uma necessidade específica, mas válida.

Erros do tipo A exigiram muitas vezes modificações radicais em determinados conjuntos de rotinas, requerendo a utilização do chamado teste de "caixa-branca" , onde os diversos fluxos computacionais são "rastreados" exaustivamente, de forma a determinar-se os pontos de falha na lógica do Sistema.

Erros do tipo B foram decorrentes muitas vezes de orientação incompleta para o usuário, requerendo apenas melhoria na documentação destinada ao usuário.

Erros do tipo C foram redimidos através da implementação de novas rotinas que cumprissem a necessidade omitida originalmente.

4.2.6 - ANÁLISE DE RESULTADOS

De posse da versão "definitiva" do SPAT elegeu-se mais algumas empresas, onde puderam ser analisados os resultados advindos da utilização do SPAT.

O resumo dessa análise será motivo principal do capítulo seguinte onde, em princípio, tentou-se quantificar a situação anterior e posterior da implantação do Sistema, podendo ser adiantado que uma série de vantagens indiretas não são factíveis de quantificações.

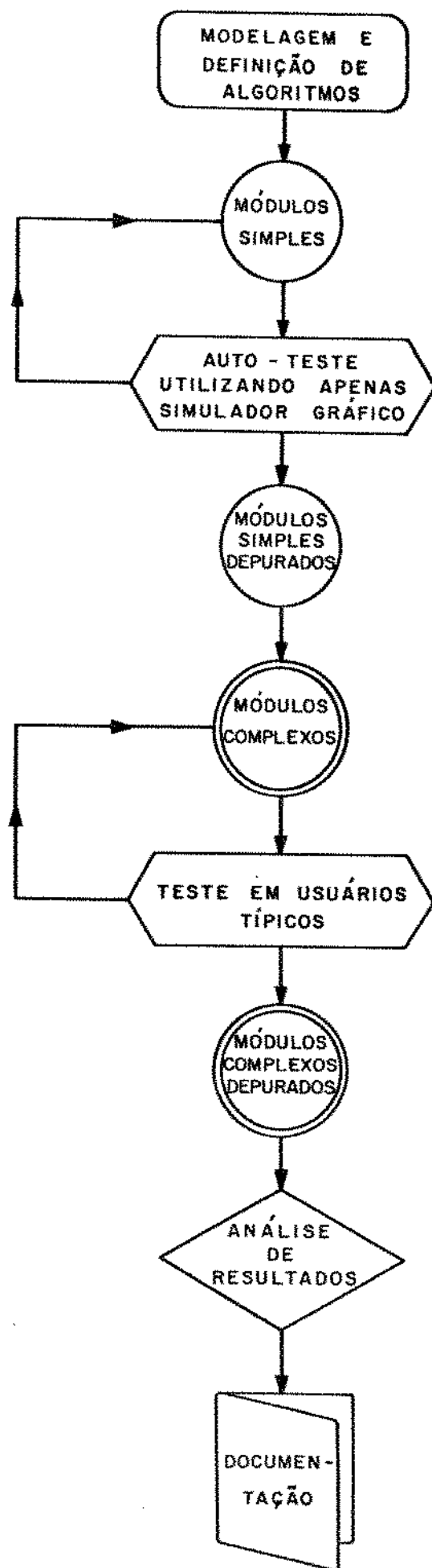


Figura 4.2-Metodologia do desenvolvimento

4.3 - USUÁRIO TÍPICO

A aplicação do SPAT em casos industriais reais, permite que sejam esboçadas as características indicativas do usuário típico:

- a) Pequena e média empresa ou departamento independente de empresa de grande porte;
- b) Número de funcionários, menor que 100;
- c) Número de programas CN, maior que 100;
- d) Dispõe de departamento de Métodos e Processos relativamente organizado, com pessoal já treinado para "programação-manual" de máquinas CN;
- e) Número de máquinas CN, menor que 10;
- f) Programas de complexidade média ou alta.

Além das aplicações industriais, o SPAT também foi aplicado em ambiente acadêmico, quando pôde ter sua validade verificada como "laboratório" para ensino de técnicas de programação CNC.

4.4 - ASPECTOS ORGANIZACIONAIS NECESSÁRIOS PARA IMPLANTAÇÃO DO SPAT

A utilização do SPAT requer muitas vezes que a empresa se reestruture de maneira a conseguir-se um ambiente propício ao mesmo [3].

Essa reestruturação inclui necessidades materiais e de mão-de-obra, sendo importante, muitas vezes, o "aculturamento" do pessoal envolvido com essa nova técnica de trabalho.

Uma implantação intempestiva pode inviabilizar o projeto, pois é natural que o pessoal tradicionalmente envolvido com a parte tecnológica (Mecânica) tenha restrições quanto a automatização na "geração" de programas.

As aplicações que ofereceram melhores resultados apresentam o seguinte perfil:

Localização do Centro de Programação: integrado ao departamento de métodos e processos;

Requisitos básicos do operador: experiência anterior como programador CN e aptidão para trabalhos de digitação;

Documentação Requerida: desenhos de produtos com cotação funcional; arquivo de ferramentas e dispositivos normalizados com desenhos detalhados; informações técnicas sobre a máquina-ferramenta, tais como espaço de trabalho, curvas características, rotações disponíveis, etc;

Informações tecnológicas sobre os materiais e as ferramentas de corte: curvas de vida e características técnicas indicativas, tais como: pressão específica de corte, fluido de corte recomendado, etc;

O trabalho no Centro de Programação deve fluir naturalmente através da utilização integrada dos 3 módulos que compõem o SPAT: EDITOR, PROCESSADOR GEOMÉTRICO E SIMULADOR GRÁFICO. A aplicação racional do SPAT possibilita a redução de testes de programas na

Oficina Mecânica, reduzindo-se, assim, os tempos de máquina parada [3].

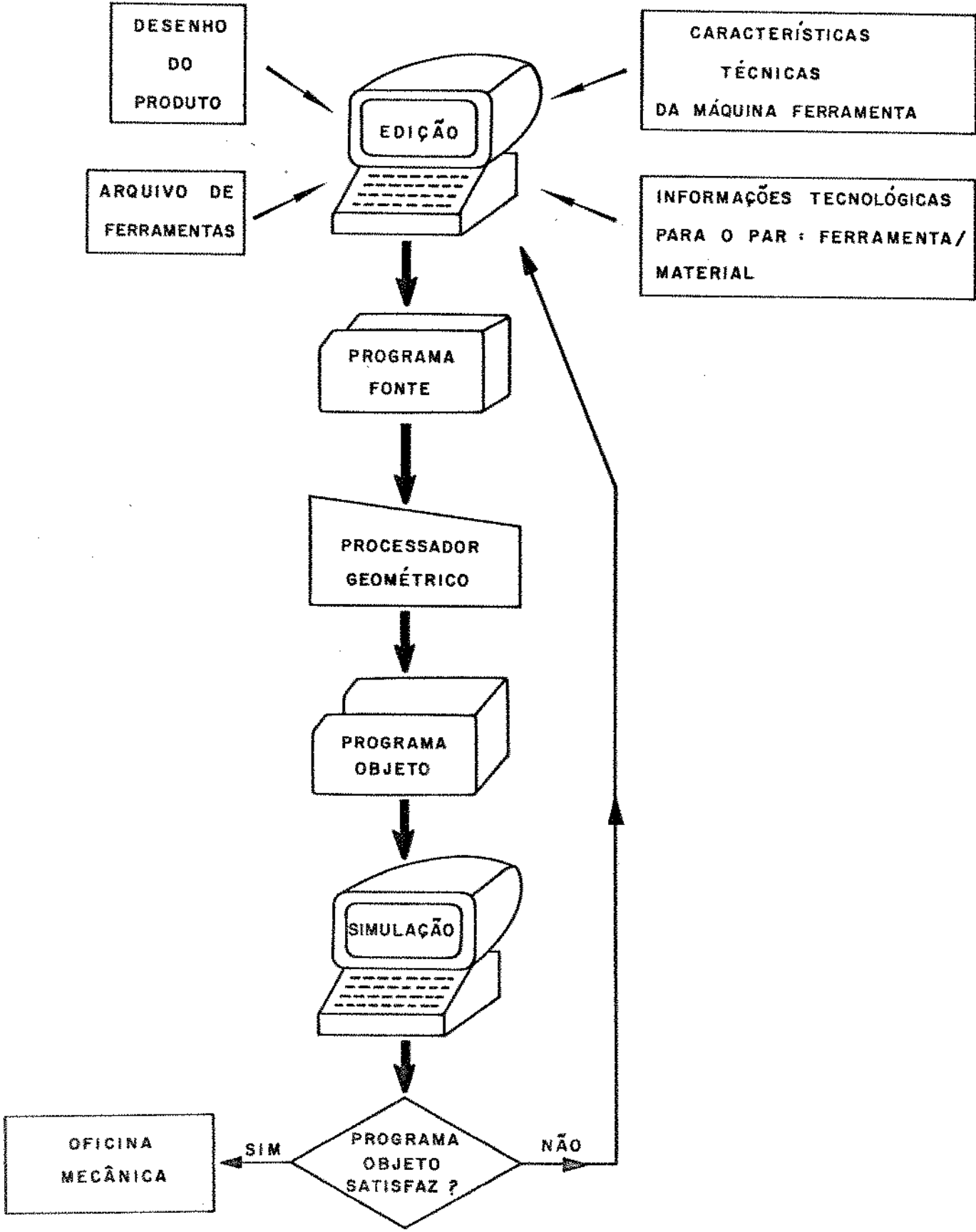


Figura 4.3-Atividades do Centro de Programação.

4.5 - GERENCIAMENTO DE PROGRAMAS CN ATRAVÉS DO SPAT [3].

A implantação do SPAT nas empresas impulsiona-as naturalmente no sentido de racionalização do trabalho, sendo que processos mecânicos de fabricação e os correspondentes programas, são encarados de forma dinâmica, sempre passíveis de otimizações. Inovações e ajustes na forma de fabricar são assimilados naturalmente, e desafios do dia a dia, podem ser resolvidos muito mais racionalmente com a utilização de computadores.

A geração automática de programas permite que as empresas possam, estrategicamente, disputar as flutuações mercadológicas, de forma muito mais competitiva, devido a:

- a) redução (ordem de 2/3) nos tempos de geração de programas;
- b) geração de programas otimizados, o que proporciona redução nos tempos de usinagem;
- c) geração de programas com reduzido índice de problemas, trazendo, como consequência, a redução do tempo de "máquina parada" para acerto do programa, na Oficina Mecânica;
- d) formação da "biblioteca" de programas CN, de maneira racional, possibilitando a localização rápida e eficiente de programas;
- e) documentação dos programas, listagens e relatórios impressos de simulações gráficas, que facilitam eventuais alterações dos programas.

Outro aspecto observado refere-se ao aumento na capacitação técnica dos profissionais envolvidos com MFCN, encaminhando tudo para um "aculturamento" da equipe, na utilização de computadores.

A integração, entre o departamento de Métodos e Processos e a Oficina Mecânica, estreita-se com o aumento do fluxo de informações entre ambos. O nível de trabalho planejado aumenta, evitando-se a ocorrência de modificações de "última-hora". A documentação (listagem de programas, trajetórias de ferramentas, etc.) aumenta a confiabilidade do Sistema Produtivo.

A utilização do SPAT em empresas sem produto próprio (prestadoras de serviço), proporcionou também uma precisão maior nos orçamentos e cotações pois, como "sub-produto" da Simulação Gráfica, têm-se as previsões de tempos, com margem de erro inferior a 5%.

Observou-se, também, a uniformização do "estilo" de programação para máquinas de procedências distintas, o que implica na constituição de equipes com padrão unificado, proporcionando maior facilidade na situação de eventuais alterações de programas, cujos originais foram produzidos por outro programador.

REFERÊNCIAS BIBLIOGRÁFICAS (CAP 4)

[1] SCHMID, C. - A Workstation Concept for Computer Aided Analysis and Design of Control Systems, IFAC Proceedings Series, V.6, p. 565, 1986.

[2] NETO, J.J. - Introdução à Engenharia de Software e Metodologia de Programação, 1º Congresso Nacional de Automação Industrial- CONAI, p. 119, São Paulo, 1983.

[3] MEREDITH, J.R. - Managing Factory Automation Projects, Journal of Manufacturing Systems, V. 6, p. 75, 1987.

[4] TELLES, G.N. - Desenvolvimento de um EDITOR/SIMULADOR para Tornos a Controle Numérico, Tese de Mestrado (102/85), Faculdade de Engenharia de Campinas FEC/UNICAMP, 1985.

[5] EL MIDANY, T.T.; EL LATIF, A.K. - Selection and Evaluation of CAD/CAM Systems for Mechanical Industry, IFAC Proceedings Series, V. 5, p. 387.

CAPÍTULO 5

ESTUDO DE CASOS TÍPICOS

5.1 - GENERALIDADES

A crescente necessidade em busca da Automação Flexível teve como primeira etapa a implantação de MFCN, as quais requerem uma alteração organizacional da empresa no aspecto do gerenciamento de programas, sendo que para tal, até alguns anos atrás, utiliza-se computadores de médio ou grande porte [1].

As primeiras implantações de MFCN, no Brasil, foram efetuadas em grandes empresas: CLARK e FORD, muito provavelmente devido à estrutura realmente onerosa que esse maquinário requeria.

A evolução, que esse equipamento sofreu, possibilitou que empresas de porte médio também passassem a encarar MFCN como solução economicamente viável.

De um total geral de 3600 MFCN disseminadas em 350 empresas, temos, atualmente, no Brasil, um total de 200 empresas de porte médio e pequeno, revertendo um quadro inicial, em que prevaleciam, como usuários desse equipamento, apenas empresas de grande porte [2].

A notável evolução da micro-informática possibilitou a sua introdução em ambiente fabril de forma a melhorar rotinas de trabalho, tais como: controle de estoque, planejamento e controle da produção e, evidentemente, sistemas de apoio à manufatura.

5.2 - CONSEQUÊNCIAS DA IMPLANTAÇÃO DO SPAT

A análise dos resultados advindos da utilização do SPAT, em usuários finais, é difícil de ser quantificada, porque proporciona uma série de benefícios indiretos.

Como benefícios diretos podemos relatar:

- a) redução no tempo requerido para geração de programas novos;
- b) redução nos ciclos de trabalhos, proporcionada por programas otimizados;
- c) documentação mecanizada, através de listagens e esquemas de trabalho;
- d) redução no tempo de testes (try-out) de programas novos na Oficina Mecânica;
- e) redução de refugos devidos a erros de programação;
- f) simplificação no gerenciamento da "biblioteca" de programas;
- g) facilidade para alterações de programas já existentes;
- h) facilidade para desenvolver programas de geometrias complexas;
- i) supre deficiências de comandos menos evoluídos;
- j) redução no tempo de "máquina-parada".

Nos itens seguintes, faremos a análise de resultados pela implantação do SPAT em dois usuários típicos.

5.3 - APLICAÇÃO TÍPICA NÚMERO 1

Como primeiro exemplo de aplicação típica, para comparação de resultados, selecionamos a firma USITEC, localizada em São Carlos -

SP, atuante no ramo de prestação de serviços de usinagem de precisão.

Essa empresa implantou simultaneamente o primeiro torno CNC e o SPAT. Essa simultaneidade, de um lado retardou todo o processo de exploração comercial dessa tecnologia, mas por outro lado permitiu que a empresa iniciasse sua organização de trabalho, assistida por computador.

Abaixo, relatamos em forma tabular alguns aspectos relativos à implantação de MFCN versus MF convencionais:

	convencional	CN
Número de MF	4	1
" " funcionários	4	2
Área ocupada	40 m ²	15 m ²
Investimento	US\$50.000	US\$80.000

A implantação do SPAT nessa empresa proporcionou as seguintes benesses:

- a) redução no "lead-time" para entrega de peça nova no "mix" usual da empresa: 5 dias (convencional: 4 semanas);
- b) redução drástica de peças refugadas: 5% (convencional: 30%);
- c) cotações e orçamentos elaborados com maior precisão, devido a utilização do módulo de Simulação Gráfica o qual fornece a previsão de tempos;
- d) usinagem de lotes de peças de grande complexidade, que seriam inviáveis de serem manufaturados no sistema convencional.

5.4 - APLICAÇÃO TÍPICA NÚMERO 2

A segunda empresa que implantou o SPAT, foi a IRMÃOS CHIEA, localizada em São Caetano do Sul - SP, fabricante de porcas para a indústria automobilística.

Distintamente do caso anterior, essa empresa já utilizava MFCN, quando da implantação do SPAT, trazendo como consequência o rápido discernimento das alterações advindas do emprego do SPAT.

Apresentamos abaixo uma tabela comparativa Antes versus Depois da implantação do SPAT:

	Antes	Depois
Tempo médio para elaboração de programa CN novo:	8,25 horas	2,5 horas
Tempo médio para introdução do programa no CNC:	35 minutos	2,5 minutos
Tempo médio para teste (try-out) de programa novo no CNC:	43 minutos	19 minutos

Além dessas vantagens possíveis de serem quantificadas, podemos citar ainda uma redução média de 22% no tempo de usinagem de roscas, devida à otimização das diversas "passadas", requeridas para a conclusão das roscas.

Uma vantagem adicional observada, nesse usuário, foi o aumento do índice de utilização das ferramentas, correspondente a 17%.

Foi possível, nessa empresa, racionalizar-se a seleção e a transmissão de programas antigos para a MFCN, devido à formação de

banco de dados único (biblioteca de programas armazenados em discos magnéticos), reduzindo assim o tempo de "máquina-parada".

5.5 - ANÁLISE DE RESULTADOS

Podemos resumir as vantagens da aplicação do SPAT através da Figura 5.1, onde são lançados (de forma absoluta) em 4 eixos, a situação média antes e depois da implantação do SPAT.

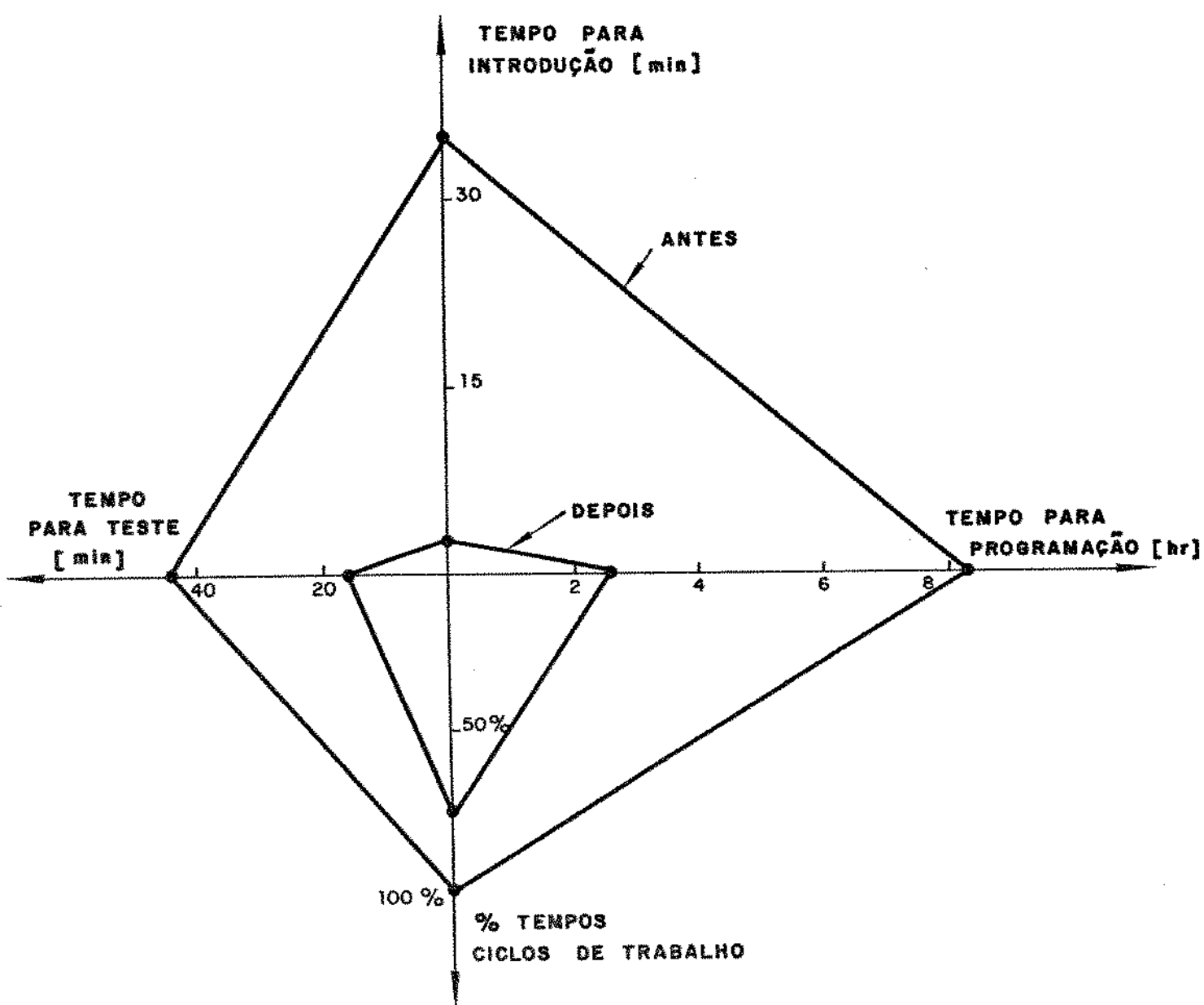


FIGURA 5.1-Situação Antes x Depois da Implantação do SPAT no Caso Típico Número 2.

A "energia" dispendida com o gerenciamento de programas CN é, de alguma forma, proporcional às áreas internas dos polígonos da Figura 5.1, onde podemos então vislumbrar as vantagens de implantação do SPAT.

O discernimento na viabilização da utilização de microcomputadores para auxiliar projeto, planejamento e outros campos técnicos das empresas, motivou o surgimento de uma série de pacotes de "software" para suprir essas áreas. A área das MFCN é intensamente afetada, sendo que no Brasil ocorre uma evolução notável, buscando-se um desenvolvimento acentuado desse segmento.

Sistemas para gerenciamento de MFCN, utilizando microcomputadores constituem, portanto, opção perfeitamente viável, conforme atestam exemplos de implantações efetuadas.

REFERÊNCIAS BIBLIOGRÁFICAS (CAP 5)

[1] HUBBE, J. - Programação Assistida de Máquinas a Comando Numérico e DNC, Simpósio sobre CAD/CAM - SOBRACON, p 2-01-18, 1987, São Paulo.

[2] STEMMER, E. C. - O Comando Numérico no Brasil, situação em 1989, Boletim SOBRACON, 1989, São Paulo.

CAPÍTULO 6

CONCLUSÕES

6.1 - GENERALIDADES

Máquinas-ferramenta constituem segmento importantíssimo da Indústria Metal-Mecânica. Nos países altamente industrializados, 10% do maquinário produzido e 10% da força-de-trabalho utilizada em Manufatura envolvem máquinas-ferramenta [1]. O avanço da micro-eletrônica proporcionou o aumento da parcela de MF que saem da fábrica equipadas com CN. Floresce, inclusive, um novo segmento dedicado à tarefa de equipar com Comando Numérico máquinas-ferramenta convencionais.

MFCN requerem, para bom desempenho, uma retaguarda de trabalho estruturada que inclui: ferramentas de corte padronizadas, desenhos de produtos com cotação funcional e, evidentemente, um sistema racional para gerenciamento de programas CN.

A popularização dos micro-computadores tornou viável a aplicação dos mesmos em tarefas as mais diversas, tendo sido demonstrada a sua viabilidade em aplicações reais, como Sistema de Apoio à Programação de MFCN.

6.2 - METAS ALCANÇADAS

Viabilizou-se, em aplicações reais, prioritariamente duas metas:

A) Dotar o mercado nacional de Sistema de Gerenciamento de Programas, para programadores de MFCN. A utilização do SPAT em

algumas empresas de porte médio permitiu verificar-se a validade da proposta inicial.

B) Viabilização de "software" para apoio ao ensino na programação de MFCN. A aplicação do SPAT para ensino mostrou-se eficaz, constituindo-se em excelente "laboratório" para exercícios de programação.

6.3 - LINHAS DE PESQUISAS PARA FUTUROS DESENVOLVIMENTOS

Como linhas de pesquisas relacionadas com esse desenvolvimento, podemos relacionar:

- a) um Sistema de Programação Automática 3D;
- b) módulo complementar ao SPAT, que utilize formas geométricas definidas por Sistema CAD;
- c) módulo complementar para escolha de ferramentas de corte;
- d) módulo complementar para escolha das condições econômicas de corte;
- e) conjunto de rotinas para soluções de problemas geométricos típicos ao programador CN: concordâncias, intersecções, tangências, etc.

REFERÊNCIAS BIBLIOGRÁFICAS (CAP 6)

[1] WECK, M. - Handbook of Machine Tools [Werkzeugmaschinen, Band 3, Automatisierung und Steuerungstechnik], John Wiley & Sons, Chichester, V.3, 1984.

APÊNDICE-1

PROGRAMAÇÃO MANUAL DE TORNOS CN

A.1.1 - GENERALIDADES

Uma programação padronizada refere-se ao relacionamento internacionalmente aceito para letras, números e símbolos de um lado, e funções das máquinas CN do outro lado. No caso específico de programação de MFCN, apesar de existirem normas internacionais, elas não são integralmente utilizadas, sendo empregadas como orientação geral.

Relataremos, portanto, a seguir, as funções de programação mais utilizadas, tomando como base, as normas VDI 3255 [1] e DIN 66025 [2], e as recomendações da ROMI, fabricante do armário de Comando MACH 3L. [3]

A.1.2 - SISTEMA DE COORDENADAS E CÓDIGOS DE EIXOS

A norma VDI 3255 [1] prevê as seguintes regras:

a) O espaço-de-trabalho (interação ferramenta x peça) é referenciado a um sistema de coordenadas cartesianas ortogonais que seguem as mesmas convenções da Geometria Analítica.

b) O eixo Z é perpendicular ao plano XY e sempre refere-se ao eixo-árvore da MFCN. Para tornos CN, a disposição mais habitual para a torre-porta-ferramentas é do lado oposto ao operador, quando temos então o eixo X perpendicular ao eixo árvore com o sentido indicado na Figura A.1.1.

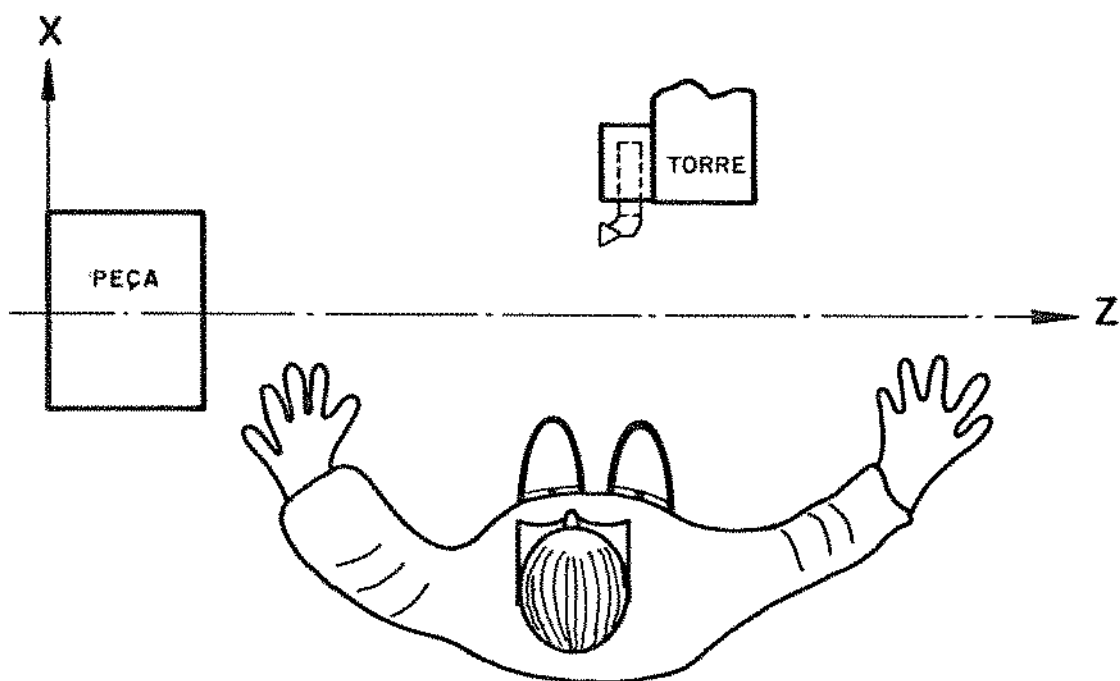


FIGURA A.1.1-Posição relativa operador x torre

c) Para tornos com a torre-porta-ferramentas construída do lado do operador, teríamos o sentido do eixo X, inverso ao indicado na Figura A.1.1.

Os armários de CN modernos, permitem a utilização do "zero-peça" flutuante, definido pelo usuário no corpo do programa CN. Isto permite a definição do "zero-peça" no local mais conveniente para o programador.

A.1.3 - SINTAXE DOS BLOCOS DE INFORMAÇÕES EM PROGRAMAS CN

Um programa CN é constituído de "blocos" (sentenças) de programação. Os blocos por sua vez, são constituídos de palavras, normalmente dispostas na sequência:

1. Palavra para número de bloco: N____;
2. Palavra para condição de percurso: G____;
3. Palavra para coordenadas: X____, Z____, I____, K____, etc;
4. Palavra para avanço: F____;
5. Palavra para velocidade de corte: S____;
6. Palavra para ferramenta: T____;
7. Palavra para função auxiliar: M____.

Não necessariamente um determinado bloco de programa CN contém todas as informações listadas acima, apresentando apenas as palavras e funções de interesse para o bloco. Assim, um bloco de programa CN, que determina o movimento rápido da ferramenta para as coordenadas X = 52,3 mm e Z = 92,6 mm poderia ser escrito por:

N1310 G00 X52.3 Z 92.6 ; APROX. FERRAMENTA

Após a inclusão do ponto-e-vírgula, é possível a inclusão de comentário.

Outra característica refere-se à função do tipo modal, isto é, uma vez especificada, ela continua válida enquanto não for programada outra que tenha capacidade de cancelá-la. Isto libera o programador de ficar reescrevendo a função, repetidamente.

A.1.3.1 - PALAVRA PARA NÚMERO DE BLOCO: N.

É usualmente a primeira informação do bloco. O caractere N é seguido de dígitos numéricos, normalmente apresentados em sequência crescente, de maneira análoga a programas escritos em linguagem BASIC. A função principal dessa informação é o posicionamento correto para eventual operação de retomada de ciclo operacional.

A.1.3.2 - PALAVRA PARA CONDIÇÃO DE PERCURSO: G.

São tão importantes que, muitas vezes, são utilizadas para caracterizar o padrão de programação (padrão G). São utilizadas para definir as condições de percurso. A Norma DIN 66025 especificou-as de G00 a G99. Abaixo, relacionamos as mais interessantes, utilizadas para tornos CN:

G00 movimento rápido (aproximação ou afastamento);
G01 interpolação linear (usinagem em linha reta);
G02 interpolação circular horária (usinagem);
G03 interpolação circular anti-horária (usinagem);
G33 corte de rosca;
G61 definição de áreas de segurança;
G96 programação em velocidade de corte constante.

A seguir, detalharemos cada uma das funções G, relacionadas acima:

G00 - MOVIMENTO RÁPIDO

Conjuntamente com a função G00, especifica-se as coordenadas do ponto-meta, o qual será atingido através de movimento rápido em linha reta, na velocidade máxima permitida pela máquina.

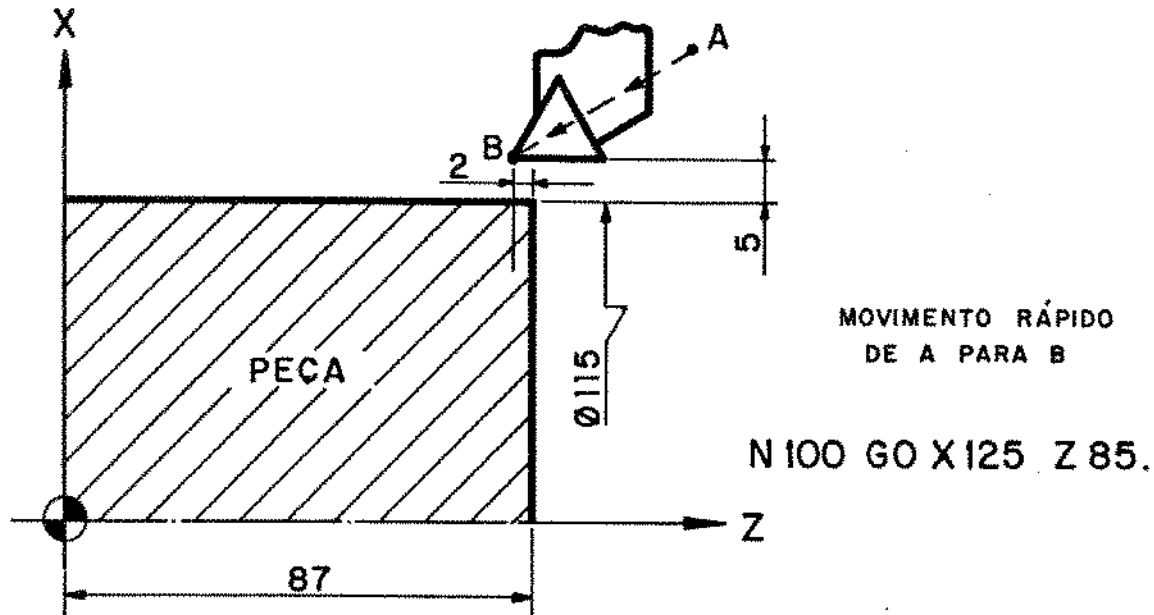


FIGURA A.1.2-Movimento rápido: G00.

G01 - INTERPOLAÇÃO LINEAR

A usinagem de trechos retos é feita programando-se G01. Além da especificação das coordenadas do ponto-meta, é requerido a indicação do avanço de trabalho (função F - vide A.1.3.4).

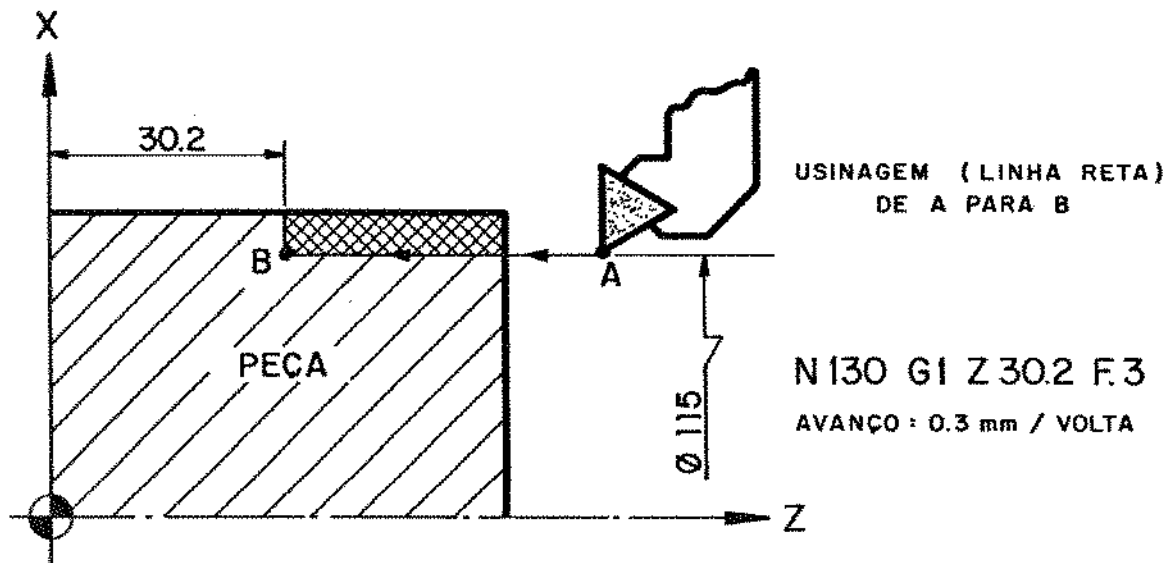


FIGURA A.1.3-Interpolação Linear: G01.

G02 - INTERPOLAÇÃO CIRCULAR HORÁRIA

Além da especificação das coordenadas do ponto-meta, o programador deve indicar as coordenadas do centro do arco-circular (funções I e K) ou o raio (função R).

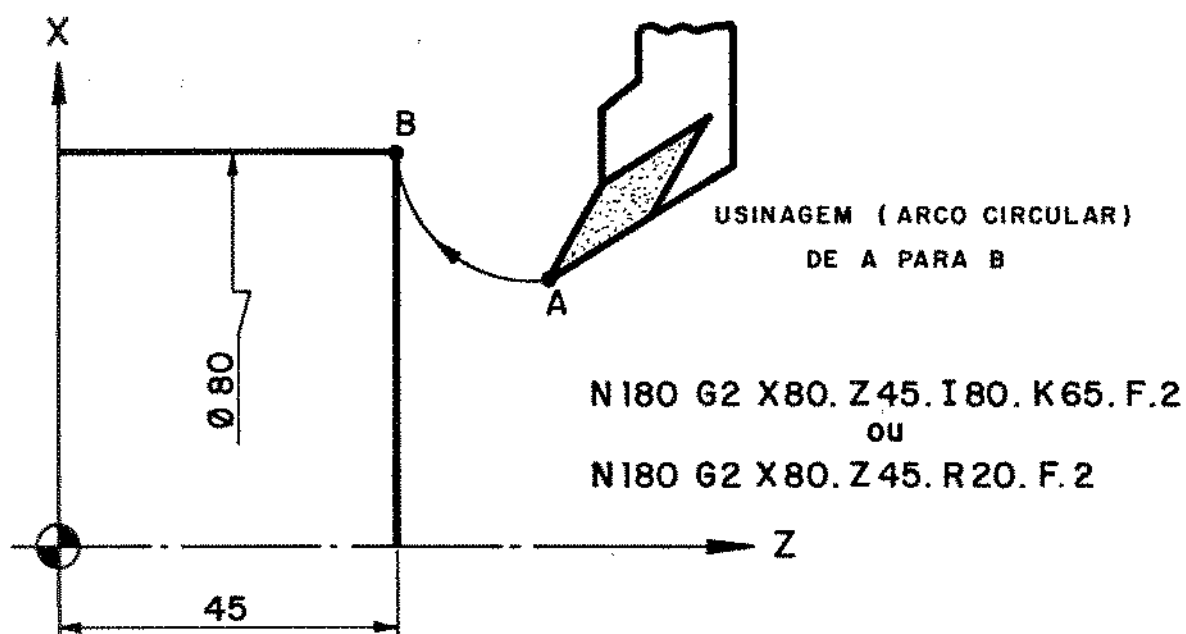


FIGURA A.1.4-Interpolação Circular Horária: G02.

G03 - INTERPOLAÇÃO CIRCULAR ANTI-HORÁRIA

Funcionamento idêntico à G02, mas referente à usinagem de arcos circulares, anti-horários.

G33 - CORTE DE ROSCA

Para corte de rosca o programador deve posicionar a ferramenta no ponto do início do corte da rosca (através de G00) e especificar G33 com as coordenadas do fim da rosca. É necessário também o fornecimento do passo da rosca (funções I e K). A função I refere-se, aqui, à projeção do passo da rosca no eixo X e a função K, refere-se à componente do passo no eixo Z.

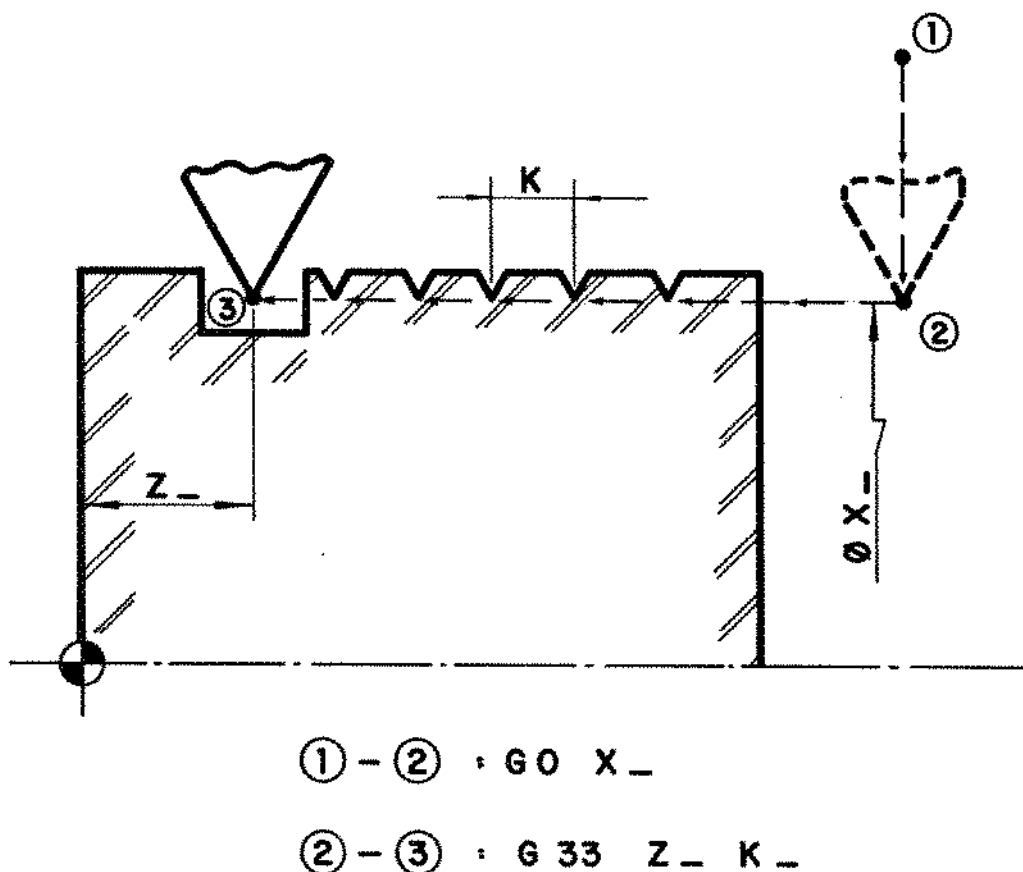


FIGURA A.1.5-Corte de Rosca: G33.

G61 - DEFINIÇÃO DE ÁREAS DE SEGURANÇA

A especificação de áreas de segurança é interessante, porque o emprego consciente das mesmas pode evitar eventuais problemas operacionais (colisões), devido a enganos nos programas.

Podem ser definidos 8 lugares geométricos (retangulares): L1 a L8.

L1, L2, L3 e L4 são utilizados para: placa, castanhas, contra ponto, etc. Para fins de simulação, essas áreas têm "fundo-cheio".

L5, L6 e L7 são empregados para definição de área reservada para matéria prima. Para fins de simulação, essas áreas têm suas fronteiras tracejadas.

L8 também é dedicado para definição de matéria prima, apenas que para fins de simulação, não são "traçados" os seus contornos, sendo utilizado para matéria-prima de geometria muito irregular. (Vide A.2.3)

G96 - PROGRAMAÇÃO EM VELOCIDADE DE CORTE CONSTANTE

Os modernos tornos de CN podem ser programados especificando-se a velocidade de corte, ao invés da rotação. Isto libera o programador da enfadonha função de programar mudanças de marchas ao longo do programa, na tentativa de manter constante a velocidade de corte, uma vez que ocorram variações no diâmetro torneável.

A função G96 é utilizada conjuntamente com a função S, que é seguida do valor numérico da velocidade de corte. Assim, a especificação da velocidade de corte de 150 m/min seria feita da forma:

N140 G96 S150

A.1.3.3 - PALAVRA PARA COORDENADAS

As coordenadas são especificadas através dos parâmetros X,Z e o seu valor numérico. Para tornos, é usual especificar-se os valores de X em diâmetro, apesar dos movimentos ocorrerem nos raios. Os parâmetros I e K, normalmente, referem-se a grandezas paralelas aos eixos X e Z, respectivamente. Assim, o corte de rosca paralela ao eixo Z, com passo 1,5 mm, seria programado por:

N420 G33 Z25. K1.5

A.1.3.4 - PALAVRA PARA AVANÇOS: F.

A programação do avanço é feita pelo parâmetro F. Normalmente em tornos a CN, especifica-se o avanço em mm/volta, ao invés da velocidade de avanço em mm/min. Assim, uma usinagem cônica com 0,15 mm/volta de avanço seria especificada por:

N510 G01 X23.5 Z1.45 F.15

A.1.3.5 - PALAVRA PARA VELOCIDADE DE CORTE: S.

O parâmetro S conjuntamente com G96 especifica numericamente a velocidade de corte constante, usualmente em [m/min]. A especificação da rotação do eixo-árvore seria feita conjuntamente com G97.

A.1.3.6 - PALAVRA PARA FERRAMENTA: T.

A indicação da ferramenta é feita pelo parâmetro T seguido de 4 dígitos numéricos, sendo que os dois primeiros dígitos referem-se à face da torre-porta-ferramentas, e os 2 últimos dígitos referem-se aos "corretores-eletrônicos" de ferramentas. Esses corretores são endereços que podem ser acessados pelo operador de forma a "compensar" eventuais erros de posicionamento de ferramentas, e/ou desgastes, que, inevitavelmente, ocorrem durante os trabalhos.

A.1.3.7 - PALAVRA PARA FUNÇÃO AUXILIAR: M.

Da mesma forma que as funções G, foram especificadas funções auxiliares de M00 a M99. Abaixo relatamos as costumeiramente utilizadas para tornos:

M00	parada incondicional;
M03	sentido de giro do eixo-árvore: horário;
M04	sentido de giro do eixo-árvore: anti-horário;
M06	indexar ferramenta;
M07	ligar fluido de corte;
M09	desligar fluido de corte

REFERÊNCIAS BIBLIOGRÁFICAS (APÊNDICE - 1)

[1] VDI 3255: Programmieren numerisch gesteuerter Werkzeugmaschinen
Festlegung der Koordinatenachsen und Zuordnung der
Bewegungsrichtungen, Duesseldorf, 1968.

[2] DIN 66025: Programmaufbau fuer numerisch gesteuerte Arbeitsmaschinen, Beuth-Vertrieb, Berlin, 1969.

[3] INDÚSTRIA ROMI S/A - Programação de Torno a Controle Numérico, Máquinas & Ferramentas, ano 1, n.5, p.20, 1979.

APÊNDICE-2

UTILIZAÇÃO DO SPAT

A.2.1 - GENERALIDADES

A utilização do SPAT requer 1 micro- computador compatível com IBM PC Xt, com 2 unidades de disco flexível de 5 1/4", 640 K e uma impressora gráfica.

A idéia consiste em reservar-se um dos disquetes para o "software" e outro para arquivamento dos programas CN. Evidentemente, é perfeitamente possível alterar-se essa disposição, quando for disponível um "driver" de disco rígido.

A vantagem da utilização de disquete, para armazenamento do "software", reside na possibilidade de "protegê-lo" contra gravações indevidas, tendo, porém, como desvantagem, uma velocidade de acesso inferior aos discos-rígidos, os quais, por sua vez, não permitem a proteção física contra eventuais acidentes de gravação, quando informações importantes podem ser "apagadas" devido a enganos.

A restrição contra o uso do Sistema, por pessoal não autorizado, é controlada pela utilização de senha de acesso (6 caracteres alfa-numéricos).

O Sistema gravita em torno do módulo EDITOR, sendo que é possível a passagem do EDITOR para o PROCESSADOR GEOMÉTRICO e vice-versa, da mesma forma que do EDITOR para o SIMULADOR GRÁFICO, mas não é possível a passagem direta do SIMULADOR para o PROCESSADOR,

requerendo a passagem através do EDITOR. Da mesma forma, a passagem do PROCESSADOR para o SIMULADOR só é obtida via EDITOR.

O desenvolvimento foi feito de maneira que a utilização interativa do SPAT seja a mais conversacional possível, dispensando-se, inclusive, o manual de utilização após um período inicial de uso.

A.2.2 - MÓDULO EDITOR

A utilização do EDITOR é feita através de 2 "menus", sendo que, logo após o fornecimento da senha, têm-se à disposição as opções previstas para início dos trabalhos.

A sequência dos trabalhos, quando se utiliza o EDITOR, é direcionada pelo próprio "software", sendo possível também a alteração dessa sequência natural, por ação do usuário. Exemplificando: após a "geração" de um programa novo (opção principal <1>) o sistema "cai no menu" de opções secundárias, o qual proporciona, entre outras opções, a possibilidade de imprimir programa, que corresponde à tarefa costumeiramente feita após a inserção de programas novos. Caso o usuário não encontrasse, no "menu" secundário, a opção desejada, ele poderia ter disponível novamente o "menu" principal, através da opção secundária: <9>.

A.2.2.1 - OPÇÕES PRINCIPAIS

O início dos trabalhos com o EDITOR é feito através da seleção de uma das opções principais:

- <1> GERAR PROGRAMA NOVO
- <2> GERAR SUB-PROGRAMA NOVO
- <3> CARREGAR PROGRAMA DO DISCO
- <4> CARREGAR SUB-PROGRAMA DO DISCO
- <5> RELAÇÃO DE PROGRAMAS EM DISCO
- <6> CARREGAR PROGRAMA DO TORNO
- <7> CARREGAR SUB-PROGRAMA DO TORNO
- <8> SIMULAÇÃO
- <9> OPÇÕES SECUNDÁRIAS
- <0> PROCESSADOR DE GEOMETRIA
- <T> RECUPERAÇÃO DE PROGRAMA/SUB-PROGRAMA
- <S> SAIR DO SISTEMA

A seguir, detalharemos cada uma das opções principais:

<1> GERAR PROGRAMA NOVO

Após essa especificação, o sistema "pede" o número com o qual será cadastrado esse novo programa, tendo-se, a seguir um EDITOR DE LINHA, direcionado para CN, com facilidades específicas para o comando em questão. Para aplicações com o comando MACH 3L, o editor dispensa a numeração "linha-a-linha" do programa CNC. Para aplicações com o comando MCS 210, o EDITOR opera de forma que, após o fornecimento do "tri-literal" que caracteriza o bloco (sentença), basta completar os campos que vão sendo "sugeridos" pelo EDITOR.

Apesar de ser "de-linha", o EDITOR desenvolvido é de utilização simples e previne quanto à possibilidade de editar-se programas CN com erros de sintaxe.

(2) GERAR SUB-PROGRAMA NOVO

Para o comando MACH 3L, é feito um arquivamento distinto para programas CN e sub-programas CN. Apesar do comando não distinguir programa de sub-programa, o SPAT "armazena-os" separadamente (veja também a opção principal (5)). Como o comando MCS 210 não admite a possibilidade de estruturação com programas e sub-programas, não há necessidade dessa opção.

(3) CARREGAR PROGRAMA DO DISCO

O acesso a programa CN anteriormente gravado em disco é obtido por essa opção, que requer apenas o fornecimento do número com que o programa foi gravado.

(4) CARREGAR SUB-PROGRAMA DO DISCO

O acesso a sub-programa CN anteriormente gravado em disco é obtido por essa opção, requerendo apenas o fornecimento do número do sub-programa. Como o comando MCS 210 não admite sub-programa, a versão para esse comando não dispõe, evidentemente, dessa opção.

(5) RELAÇÃO DE PROGRAMAS EM DISCO

É possível obter-se a relação dos programas CN anteriormente armazenados em disco, tendo-se as indicações: o "driver" em que o programa foi gravado; o número (crescente) do programa CN e a especificação da peça (30 caracteres). Essa rotina permite ainda a impressão da relação dos programas CN e/ou a relação/impressão (se existir!) dos sub-programas. A exposição no vídeo é feita apresentando-se página por página da relação, sendo que a mudança de páginas é obtida por toque na tecla de espaços do micro-computador. Após a visualização completa da relação de programas, é possível retornar-se à página de OPÇÕES PRINCIPAIS através da digitação da tecla R.

(6) CARREGAR PROGRAMA DO TORNO

Programas CN podem ser transferidos da Unidade de Comando do Torno para o micro-computador utilizando-se o canal serial (RS-232C) [1]. A efetivação dessa transmissão requer, dependendo da situação, a utilização de acopladores ópticos, amplificadores e, evidentemente, cabos elétricos [2].

Dependendo da Unidade de Comando em questão, após a especificação dessa opção, o micro-computador explicita na tela o procedimento para que a transmissão seja efetuada. Durante o período em que efetivamente a transmissão ocorre, é feita a exposição do programa no vídeo, à medida que os impulsos elétricos chegam ao micro-computador. Concluída a transmissão do programa, o computador pede que seja fornecido o número que passará a identificar o mesmo.

<7> CARREGAR SUB-PROGRAMA DO TORNO

Funcionalmente idêntico ao anterior, mas dedicado ao recebimento de sub-programas. A versão relativa ao comando MCS 210 não possui essa opção.

<8> SIMULAÇÃO

Através dessa opção, aciona-se o módulo SIMULADOR GRÁFICO.
(Vide A.2.3)

<9> OPÇÕES SECUNDÁRIAS

Essa opção expõe, e deixa disponível, a relação de opções secundárias.

<0> PROCESSADOR DE GEOMETRIA

O desvio para o processador geométrico é obtido através dessa opção (Vide A.2.4). Através do processador geométrico, temos disponíveis os recursos da Programação Automática.

<T> RECUPERAÇÃO DE PROGRAMA/SUB-PROGRAMA

O último programa (sub-programa) digitado, ou o último programa recebido através do protocolo serial, pode ser "recuperado", quando da "perda" do mesmo, por eventual queda de energia ou operação indevida.

<S> SAIR DO SISTEMA

Concluídos os trabalhos, essa opção deixa disponível o Sistema Operacional do computador (DOS) [3].

A.2.2.2 OPÇÕES SECUNDÁRIAS

Abaixo, relacionamos a lista que compõe as opções secundárias do módulo EDITOR:

- <1> ALTERAR BLOCOS DE PROGRAMA
- <2> INSERIR BLOCOS NO PROGRAMA
- <3> CANCELAR BLOCOS DO PROGRAMA
- <4> IMPRESSÃO DE PROGRAMA
- <5> EXPOSIÇÃO DE PROGRAMA (no vídeo)
- <6> TRANSFERIR PROGRAMA PARA O TORNO
- <7> ARMAZENAR PROGRAMA (disco)
- <8> SIMULAÇÃO
- <9> RETORNO A OPÇÕES PRINCIPAIS
- <0> PROCESSADOR DE GEOMETRIA
- <A> APAGAR PROGRAMA DO DISCO
- <S> SAIR DO SISTEMA

A seguir, apresentamos a descrição, opção por opção.

<1> ALTERAR BLOCOS DE PROGRAMA

Blocos (sentenças) do programa podem ser alterados por essa opção. Após a seleção dessa opção, o sistema solicita a indicação do bloco que se deseja alterar. A forma para efetivar a alteração é indicada na própria tela do micro-computador, requerendo, para tanto, a utilização das teclas: <seta-a-direita> e <seta-a-esquerda>.

<2> INSERIR BLOCOS NO PROGRAMA

Através dessa opção, blocos podem ser estrategicamente inseridos no corpo original do programa. O ponto de inserção do bloco no corpo do programa CN é definida pelo número do bloco que está sendo inserido.

<3> CANCELAR BLOCOS DO PROGRAMA

Após a seleção dessa especificação, é solicitado o número do bloco que se deseja apagar. Uma vez fornecido esse número, o Sistema expõe o bloco (sentença) especificado e pede a confirmação do cancelamento. Essa confirmação é feita para evitar enganos no apagamento de blocos.

(4) IMPRESSÃO DE PROGRAMA

Essa opção dispara a listagem do programa que está na memória do micro-computador. A listagem do programa CN é importante para a "depuração" do programa, ou mesmo para fins de documentação (arquivo).

(5) EXPOSIÇÃO DO PROGRAMA (vídeo)

Programas podem ser expostos no vídeo, 10 blocos (sentenças) por página. A passagem para a página seguinte é obtida acionando-se a tecla (barra-de-espacos). O retorno de 1 bloco na exposição é obtido pela tecla: <T> . Ao término da exposição, é possível renumerar-se o programa.

(6) TRANSFERIR PROGRAMA PARA O TORNO

Através do protocolo serial, podemos transmitir (via cabo-elétrico) programas CN para a unidade de comando. Após essa seleção, o Sistema indica na tela o procedimento que deve ser tomado na Unidade de Comando para efetivar-se a transmissão. Da mesma forma que no recebimento de programas, aqui, conforme a transmissão ocorre, são indicados na tela os caracteres transmitidos.

Essa opção é bastante interessante porque elimina a necessidade da utilização de fita de papel perfurado, veículo portador de informações, tradicionalmente utilizado para MFCN

<7> ARMAZENAR PROGRAMA (disco)

Antes do acionamento dos módulos de Simulação ou Processador Geométrico, é requerida a gravação do programa em disco. No instante da gravação, é vinculado ao programa um comentário de 30 caracteres alfa-numéricos, normalmente utilizados para: especificação da peça (código do produto, nome); fase de fabricação; data e iniciais do programador.

<8> SIMULAÇÃO

A passagem para o módulo de Simulação Gráfica é obtida por essa opção. (Vide A.2.3)

<9> RETORNO AS OPÇÕES PRINCIPAIS

A sequência natural dos trabalhos pode ser alterada, sendo que através dessa opção, têm-se disponíveis, novamente, as Opções Principais.

<0> PROCESSADOR DE GEOMETRIA

Essa opção ativa o módulo PROCESSADOR DE GEOMETRIA, tornando disponíveis os recursos da Programação Automática. (Vide A.2.4).

(A) APAGAR PROGRAMA

Programas anteriormente gravados, podem ser apagados do disco. Após o fornecimento do número de programa que se deseja apagar, o Sistema apresenta o comentário de 30 caracteres que especifica a peça, o programador, etc, e pede confirmação. Esse procedimento é feito de maneira a prevenir-se contra o apagamento de programa errado. Caso não ocorra a confirmação, o programa não é apagado.

(S) SAIR DO SISTEMA

Essa opção é exercida no término dos trabalhos. Tem-se, então, o retorno ao Sistema Operacional do Computador (DOS) [3].

A.2.3 - MÓDULO DE SIMULAÇÃO GRÁFICA

O módulo de SIMULAÇÃO atua de maneira semelhante ao armário de comando da MFCN, sendo que movimentos reais da máquina correspondem ao traçado gráfico na tela do computador, permitindo, portanto a verificação sequencial do trabalho a ser executado pela MFCN. Outras funções tais como "ligar fluido de corte", são indicadas na forma textual.

Na "entrada" do Simulador é solicitado o número do programa que se deseja simular. Caso ocorra um "arrependimento", o retorno ao módulo EDITOR é obtido, digitando-se: -1 (menos um).

A seguir, é solicitado o fornecimento das áreas de segurança as quais são especificadas de acordo com o padrão G61 do comando MACH 3L. Podem ser utilizados 8 Lugares Geométricos (retângulos): L1 a L8, onde:

L1, L2, L3 e L4 são utilizados para: Placa, Castanha, etc.

L5, L6, L7 e L8 são utilizados para: Matéria Prima.

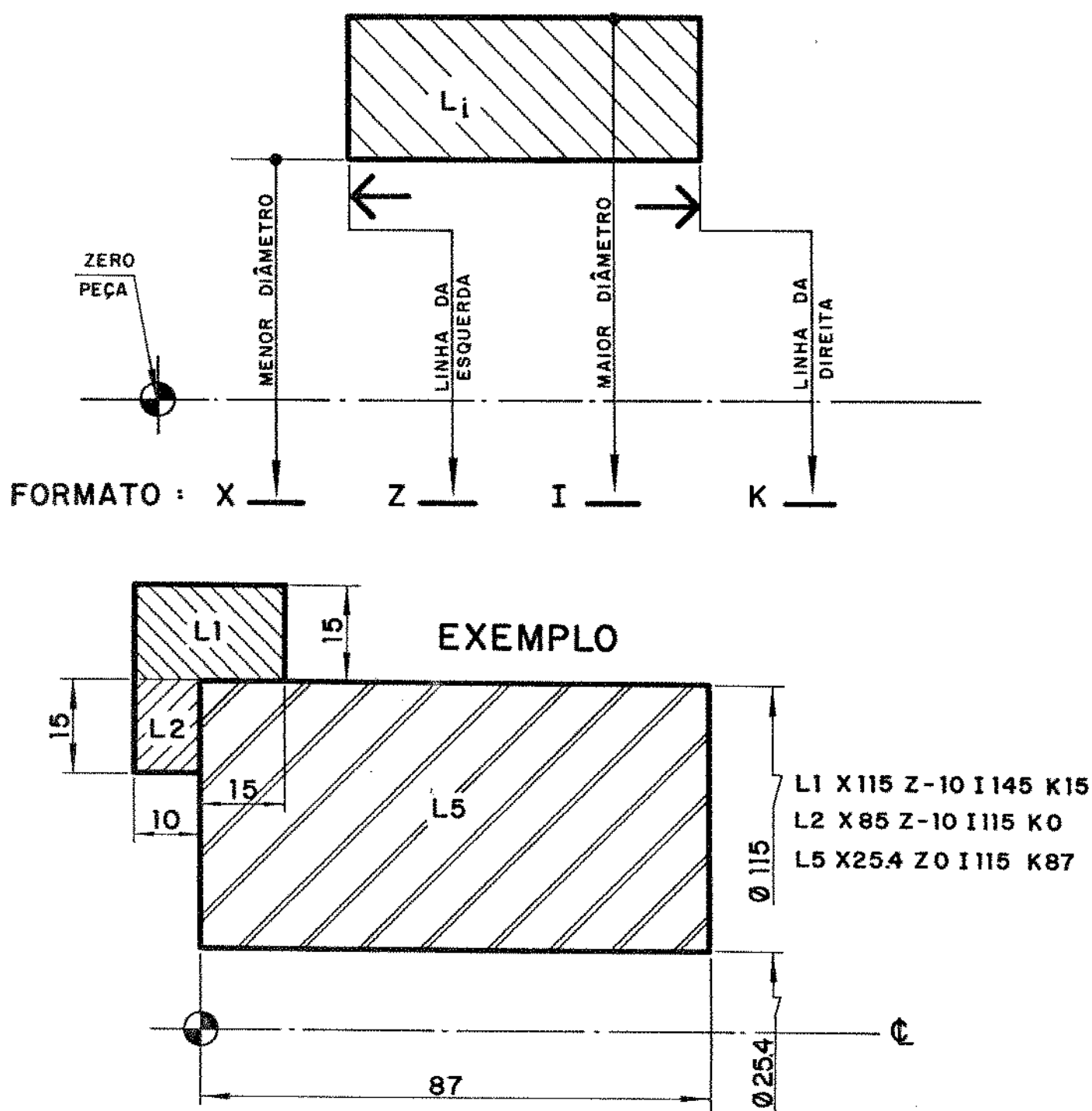


FIGURA A.2.1-Definição das Áreas de Segurança.

A Simulação é feita apresentando-se a matéria prima em "meia-vista", interagindo, com o rastreamento luminoso da ponta teórica da ferramenta, no vídeo. A velocidade de simulação pode ser aumentada ou diminuída, digitando-se seguidamente as teclas +(mais) ou -(menos), respectivamente.

A interrupção da simulação pode ser conseguida digitando-se a tecla <C>, e a retomada da simulação, acionando-se a tecla <D>. O retorno ao módulo EDITOR é obtido digitando-se a tecla <R>.

Além da simulação gráfica, é apresentado na parte inferior da tela o bloco do programa, que está sendo simulado naquele instante, além da interpretação textual do mesmo.

Nos pontos da simulação que correspondem à troca de ferramentas é possível: "limpar" a tela e/ou, descarregar na impressora a imagem gráfica do vídeo.

Ao término da simulação, temos disponível uma tabela de tempos indicativa da previsão de tempos para a usinagem da peça simulada.

A.2.4 - MÓDULO PROCESSADOR DE GEOMETRIA

O acionamento do Processador Geométrico é feito quando vamos trabalhar utilizando a Programação Automática, requerendo, previamente, a edição e a gravação do programa-fonte, através do módulo EDITOR. O Processador Geométrico deverá transformar o programa-fonte em programa-objeto.

A "entrada" inicial consiste, então, no fornecimento do número do programa-fonte, sendo que o fornecimento de - 1 (menos um) corresponde ao retorno ao módulo EDITOR.

Como podem ocorrer enganos na geração do programa-fonte, antes do processamento ocorrer, o Sistema pergunta se há interesse em imprimir o programa-fonte.

O processamento é feito com o Sistema fornecendo ao usuário "satisfações" sobre o tratamento computacional que está evoluindo. Assim sendo, o Sistema indica no vídeo os detalhes dos trechos do acabamento final, à medida que vai utilizando-os no processamento para geração do programa-objeto.

Erros de lógica no programa-fonte, que impossibilitem a constituição do programa objeto, são indicados no vídeo informando também o bloco da ocorrência do problema.

Efetivada a formação do programa-objeto, é possível a impressão do mesmo, antes do retorno ao EDITOR que, obrigatoriamente, corresponde à próxima etapa do trabalho.

Tendo-se disponível o programa-objeto, é possível simulá-lo de forma a verificar-se se o resultado corresponde ao planejamento inicial, codificado no programa-fonte.

Eventuais problemas detectados na fase de Simulação requerem a utilização do EDITOR, para depurações e, novamente, o PROCESSADOR DE GEOMETRIA, até obter-se um programa-objeto que satisfaça o planejamento inicial para execução correta da peça.

REFERÊNCIAS BIBLIOGRÁFICAS (APÊNDICE 2)

[1] CAMPBELL, J. -RS-232 Técnicas de Interface, EBRAS - Editora Brasileira Importação e Comércio de Livros Ltda, São Paulo, 158 p., 1986.

[2] MENDES, M. J. - Redes Locais de Comunicação em Ambiente Industrial, 6 Seminário de Comando Numérico no Brasil - SOBRACON, V.1 , p 17.01, 1986.

[3] SACHS, J. - IBM PC e Compatíveis, McGraw Hill do Brasil, São Paulo, 335 p, 1985.

BIBLIOGRAFIA

CAMPBELL, J. - RS-232 Técnicas de Interface, EBRAS - Editora Brasileira Importação e Comércio de Livros Ltda, São Paulo, 158 p., 1986. [A.2.1]

DIN 66024: Numerische Steuerung von Arbeitsmaschinen - Code fuer 8-Spur-Lochstreifen, Beuth-vertrieb, Berlin, 1967. [2.9]

DIN 66025: Programmaufbau fuer numerisch gesteuerte Arbeitsmaschinen, Beuth-vertrieb, Berlin, 1969. [2.10] [A.1.2]

EL MIDANY, T.T.; EL LATIF, A.K. - Selection and Evaluation of CAD/CAM Systems for Mechanical Industry, IFAC Proceedings Series, V.5, p. 387. [4.5]

Elements of Programing NC, Tooling and Production, p. 102, Apr. 1980. [3.1]

FERRARESI, D. - Fundamentos da Usinagem dos Metais, Editora Edgard Blucher Ltda., São Paulo, 1970. [3.4]

FREDERIKS, G. - A Threading Algorithm Suitable for Computer Numerically Controlled Turning Centers, Machine Tool Design, V. 18, p. 29, Great Britain, 1978. [3.6]

FUCCI, T.V.; FRANÇA, A.L. - A Informática e a Indústria de Bens e Capital, Reunião Aberta do 40º Congresso Anual da ABM, V.1, p.3, São Paulo, 1985. [1.3]

GENDRE, J.C. - PROMO - An NC System for Small Computer, International Conference on Computer Aided Manufacture and Numerical Control, p. 9-1, Glasgow, 1974. [1.21]

GOLDHAR, J.D. - The Organizational Impacts of Computer Based Manufacturing, NATO AIS Series, V. F11, p. 567, Springer-Verlag, Berlin, 1984. [2.5]

GOODFELLOW, S.D.; MURNRO, N. - INTEGRA, an Input Translation Facility for Computer Aided Control Systems Engineering, IFAC Proceedings Series, V. 5, p. 87, 1986. [2.2]

GRANT, H.E. - Dispositivos em Usinagem [Jigs and Fixtures], Livraria Ciência e Tecnologia, São Paulo, 1982. [3.3]

HATSCHEK, J. - NC Programming, American Machinist, p. 118, Feb. 80. [1.16]

HUBBE, J. - Programação Assistida de Máquinas a Comando Numérico e DNC, Simpósio sobre CAD/CAM-SOBRACON, p. 2-01-18, 1987, São Paulo. [5.1]

IFAO - Institut fuer angewandte Organisationsforschung - Comando Numérico CNC [CNC-Ausbildung fuer die betrieblich Praxis Teil 3: Drehen], Editora Pedagógica Universitária, São Paulo, 1985. [3.2]

INDÚSTRIA ROMI S/A - Programação de Torno a Controle Numérico, Máquinas & Ferramentas, ano 1, n. 5, p. 20, 1979. [1.13] [3.7] [A.1.3]

ISO/DIS 2806.2: Numerical Control of Machines - Vocabulary, 1979.
[2.12]

KASZA, J. - Organizational Model of NC Machine Installation,
Periodica Polytechnica, M30/3-4, p. 359, 1984. [1.15]

LEWIS, W. ; BARAS, H.M.M.; SOBERG, J.J. - Computer Integrated
Manufacturing System Control: A Data Flow Approach, Journal of
Manufacturing Systems, V. 6, p. 177, 1987. [1.7]

MACALONEY B; CAPURRO, J. - GM MAPs Route to Integrated Factory
Communications, Blue Book Machine and Tool, V.80, n. 11, p. 45,
Nov. 85. [1.11]

MACHADO, A. - O Comando Numérico Aplicado às Máquinas Ferramenta,
ícone Editora, São Paulo, 1986. [2.8] [3.5]

MENDES, M.J. - Redes Locais de Comunicação em Ambiente Industrial,
6º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p.
17.01, 1986. [1.10] [A.2.2]

MEREDITH, J.R. - Implementing the Automated Factory, Journal of
Manufacturing Systems, V. 6, p. 1, 1987. [1.14]

MEREDITH, J.R. - Managing Factory Automation Projects, Journal of
Manufacturing System, V. 6, p. 75, 1987. [4.3]

MILLER, S.G. - New Directions in Computer-Aided Design/Computer-
Aided Manufacturing (CAD/CAM) Systems, NATO ASI SERIES, V. F11, p.
245, Springer-Verlag, Berlin, 1984. [1.4]

MILLS, R.I. - Computer Simulation - A Feasibility and Planing Tool for FMS, Proceedings of 2nd. International Conference on Flexible Manufacturing Systems, London, V. 1, p. 185, 1983. [1.18]

MIURA, H.M.; GIRAUD, C.O. - Tecnologia de Grupos - O Sistema "SAAP" (Sistema de Apoio a Projetos de Produtividade), Anais do 7º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p. 15.01, 1987. [1.6]

MONROY, D.; CAULLIRAUX, H.M. - Engenharia Industrial Avançada: Uma Alternativa da Fábrica do Futuro, Anais do 7º Congresso de Comando Numérico no Brasil - SOBRACON, V. 1, p. 17.01, 1987. [1.12]

NETO, J.J. - Introdução à Engenharia de Software e Metodologia de Programação, 1º Congresso Nacional de Automação Industrial-CONAI, p. 119, São Paulo, 1983. [4.2]

OLIVEIRA, J.F.; LIRANI, J. - Algumas Considerações a Respeito do Desenvolvimento de CAM na Indústria Mecânica, Anais do 7º Congresso de Comando Numérico no Brasil - SOBRACON, V. 1, p. 14.01, 1987. [1.8]

Pesquisa SOBRACON - VENDAS 88/PREVISÃO 89, BOLETIM SOBRACON, Jan./Fev. 89, n. 42, p. 8-15. [5.3]

PRESMAN, R.; WILLIAMS, J.E. - Numerical Control and Computer Aided Manufacturing, John Wiley & Sons, New York, 1977. [2.6]

QUEIROZ, A.A.; BOEHS, L.; COUTINHO, S.A.L. - Seleção de Condições Otimizadas de Corte "Software" Apoiado por Banco de Dados, 6º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p. 01.01, São Paulo, 1986. [2.3]

QUEIROZ, A.A. - Um Sistema de Programação do Tipo "APT" sem Linguagem, 5º Seminário de Comando Numérico no Brasil - SOBRACON, V. 1, p. 4.1, São Paulo, 1985. [2.4]

ROCHA, J.E.B. - Emprego e Relações Industriais, BOLETIM SOBRACON, Nov./Dez. 88, p. 10-18. [5.4]

ROZENFELD, H. - Desenvolvimento de Uma Linguagem Universal para Programação de Máquinas CNC, 9º Seminário de Comando Numérico - SOBRACON, São Paulo, 1989. [1.17]

SACHS, J. - IBM PC e Compatíveis, McGraw Hill do Brasil, São Paulo, 335 p, 1985. [A.2.3]

SCHIMIDT, G. LAPPUS, G. - Digital Simulation Methods - A tutorial, IFAC Proceedings Series, V. 6, p. 83, 1986. [2.13]

SCHMID, C. - A Workstation Concept for Computer Aided Analysis and Design of Control Systems, IFAC Proceedings Series, V. 6, p. 565, 1986. [4.1]

SIMON, W. - The Numerical Control of Machine Tools [Die Numerische Steuerung von Werkzeugmaschinen], Edward Arnold, London, 1973. [1.1] [2.7]

SMITS, L.J. - A Automação e a Robótica Valem Realmente a Pena ?, Boletim SOBRACON, ano IV, n. 38/39, p. 5.14, São Paulo, 1988. [1.9]

STEMMER, E.C. - Efeitos do CN na Indústria Mecânica e Perspectivas, Máquinas & Ferramentas, Ano 1, n. 7, p. 36, Junho, 1980. [1.2]

STEMMER, E.C. - Os Rumos da Automatização Industrial no Brasil, Máquinas e Metais, Ago., n. 238, p. 28-34, 1985. [5.2]

TEICHOLZ, E. - CAD/CAM Handbook, Mc Graw Hill Book Co., New York, 1985. [2.1]

TELLES, G.N. - Estudos Comparativos entre Máquinas a Controle Numérico e as Convencionais, Máquinas & Ferramentas, Ano 1, n. 11, p. 45, Junho, 1980. [1.5]

TELLES, G.N. - Desenvolvimento de um EDITOR/SIMULADOR para Tornos a Controle Numérico, Tese de Mestrado (102/85), Faculdade de Engenharia de Campinas FEC/UNICAMP, 1985. [1.19] [4.4]

Turbo Pascal - Reference Manual, Borland International, 1985. [2.14]

UNBEHAVEN, H. - CAD in Control Engineering Education, IFAC Proceedings Series, V. 5, p. 33, Lyngby, Denmark, 1985. [1.10]

VDI 3255: Programmieren numerisch gesteuerter Werkzeugmaschinen
Festlegung der Koordinatenachsen und Zuordnung der
Bewegungsrichtungen, Duesseldorf, 1968. [2.11] [A.1.1]

WECK, M. - Handbook of Machine Tools [Werkzeugmaschinen, Band 3, Automatisierung und Steuerungstechnik, John Wiley & Sons, Chichester, V. 3, 1984. [2.15] [6.1]

YONG, Y.F.; BONNEY, M.C.; KNIGHT, J.A.G. - GRASP a Design Tool for FMS, Proceedings of 2nd International Conference on Flexible Manufacturing Systems, London, V. 1, p. 215, 1983. [2.16]