

SIEMENS

SINUMERIK 840D/840Di/810D/FM-NC

Manual de programação

Edição 04.2000

Avançada

SIEMENS

SINUMERIK 840D/840Di/810D/FM-NC Avançada

Manual de programação

Válido para

| <i>Controles</i> | <i>Versão de software</i> |
|-----------------------------------|---------------------------|
| SINUMERIK 840D | 5 |
| SINUMERIK 840Di | 5 |
| SINUMERIK 840DE (v. p/exportação) | 5 |
| SINUMERIK 810D | 3 |
| SINUMERIK 810DE (v. p/exportação) | 3 |
| SINUMERIK FM-NC | 3 |

Edição 04.2000

| | |
|--|----|
| Programação NC flexível | 1 |
| Subprogramas, Macros | 2 |
| Gerenciamento de programas e arquivos | 3 |
| Zonas de proteção | 4 |
| Comandos especiais de movimento | 5 |
| Frames | 6 |
| Transformações | 7 |
| Corretores de ferramenta | 8 |
| Comportamento dos movimentos na trajetória | 9 |
| Movimentos com ações síncronas | 10 |
| Oscilação | 11 |
| Puncionar e riscar | 12 |
| Funções adicionais | 13 |
| Programas de usuário para desbaste | 14 |
| Tabelas | 15 |
| Apêndice | A |

SINUMERIK® - Documentação

Histórico

Detalhes superficiais desta edição e das anteriores encontram-se listados abaixo.

O estado de cada edição é exibido através do código na coluna "Comentário".

Códigos utilizados na coluna "Comentário"

A Nova documentação.

B Reimpressão não revisada com novo número de encomenda

C Edição revisada com novo estado

A alteração de fatos técnicos mencionados em uma página, em comparação com a versão anterior, é indicada pela versão de edição atualizada no cabeçalho da respectiva página.

| Edição | N.º de pedido | Comentário |
|--------|--------------------|------------|
| 02.95 | 6FC5298-2AB00-0BP0 | A |
| 04.95 | 6FC5298-2AB00-0BP1 | C |
| 12.95 | 6FC5298-3AB10-0BP0 | C |
| 03.96 | 6FC5298-3AB10-0BP1 | C |
| 08.97 | 6FC5298-4AB10-0BP0 | C |
| 12.97 | 6FC5298-4AB10-0BP1 | C |
| 12.98 | 6FC5298-5AB10-0BP0 | C |
| 08.99 | 6FC5298-5AB10-0BP1 | C |
| 04.00 | 6FC5298-5AB10-0BP2 | C |

Este livro forma parte integrante da documentação no disco CD-ROM (**DOCONCD**)

| Edição | N.º de pedido | Comentário |
|--------|---------------------|------------|
| 04.00 | 6FC5 298-5CA00-0BG2 | C |

Marcas registradas

SIMATIC®, SIMATIC HMI®, SIMATIC NET®, SIROTEC®, SINUMERIK® e SIMODRIVE® são todas marcas registradas Siemens. Outros nomes utilizados nesta publicação podem ser também marcas registradas, cujo uso por uma terceira parte em seu benefício poderá violar os direitos de seu proprietário.

Para mais informações veja-se no Internet sob:
<http://www.aut.siemens.de/sinumerik>

Esta documentação foi criada mediante WinWord V 7.0
e Designer V 4.0.

Esta publicação não pode ser reproduzida nem transmitida, é proibido de utilizar
ou participar o conteúdo da mesma sem prévia autorização expressa.
Contravenções obrigam à indenização por perdas e danos. Reservados todos os
direitos, em particular para o caso da concessão de patente ou do registro de
modelos de utilidade.

No controle podem encontrar-se outras funções operacionais não descritas nesta
documentação. Porém, não há qualquer direito a estas funções no caso de um
fornecimento novo ou do serviço de assistência pós-venda.

Controlamos o conteúdo deste livro quanto à conformidade com o hardware e
software descritos. No entanto, não é possível excluir diferenças, i.é., não
assumimos a garantia para a conformidade total. As indicações neste livro
verificamos em intervalos regulares e, se necessário, efetuamos correções
necessárias.
Muito agradeceríamos sugestões e propostas de melhoramento.

© Siemens AG 1996 - 1997. All Rights Reserved.

Reservadas alterações técnicas.

Nr.de encomenda. 6FC5298-5AB10-0BP2
Impresso na República federal da Alemanha

Siemens Aktiengesellschaft

Conteúdo

| | |
|---|-------------|
| Prefácio | 0-13 |
| Programação NC flexível | 1-21 |
| 1.1 Variáveis e parâmetros aritméticos | 1-22 |
| 1.2 Definição de variáveis..... | 1-25 |
| 1.3 Definição de array (matrizes) | 1-30 |
| 1.4 Programação indireta | 1-36 |
| 1.5 Carga de valores | 1-38 |
| 1.6 Operações aritméticas/funções..... | 1-39 |
| 1.7 Comparação e operadores lógicos..... | 1-41 |
| 1.8 Prioridade dos operadores | 1-44 |
| 1.9 Possibilidades de conversão | 1-45 |
| 1.10 Operações com string | 1-46 |
| 1.10.1 Conversão de tipo | 1-47 |
| 1.10.2 Encadeando seqüências de caracteres..... | 1-49 |
| 1.10.3 Conversão de caracteres maiúsculas/minúsculas | 1-50 |
| 1.10.4 Comprimento da string | 1-51 |
| 1.10.5 Procura por caracter/seqüência de caracteres na string | 1-51 |
| 1.10.6 Seleção de uma substring | 1-53 |
| 1.10.7 Selecionando um único caracter..... | 1-54 |
| 1.11 Instrução CASE | 1-56 |
| 1.12 Estruturas de controle | 1-58 |
| 1.13 Coordenação de programas..... | 1-63 |
| 1.14 Rotina de interrupção | 1-68 |
| 1.15 Transferência de eixo, transferência de fuso | 1-76 |
| 1.16 NEWCONF: Ativando dados de máquina (A partir da SW 4.3) | 1-80 |
| 1.17 WRITE: Escrita de arquivo (a partir da SW 4.3) | 1-81 |
| 1.18 DELETE: Apagar arquivo (a partir da SW 4.3)..... | 1-83 |
| 1.19 READ: Lê linhas de arquivos (a partir da SW 5.2) | 1-84 |
| 1.20 ISFILE: Arquivo disponível na memória de usuário NCK (a partir da SW5.2) | 1-87 |
| 1.21 CHECKSUM: Criar o checksum em um array (> SW 5.2) | 1-88 |
| Subprogramas, Macros | 2-91 |

| | | |
|--|--|--------------|
| 2.1 | Utilizando subprogramas | 2-92 |
| 2.2 | Subprograma com o mecanismo SAVE | 2-94 |
| 2.3 | Subprogramas com transferência de parâmetros | 2-95 |
| 2.4 | Chamando subprogramas | 2-99 |
| 2.5 | Repetir subprograma | 2-103 |
| 2.6 | Subprograma modal, MCALL | 2-104 |
| 2.7 | Chamada indireta de subprograma | 2-105 |
| 2.8 | Chamando subprogramas com diretórios e parâmetros, PCALL..... | 2-106 |
| 2.9 | Suprimindo a exibição do bloco atual, DISPLOF..... | 2-107 |
| 2.10 | Supressão de bloco a bloco, SBLOF, SBLON (SW 4.3 em diante) | 2-108 |
| 2.11 | Execução de subprogramas externos (SW 4.2 em diante)..... | 2-111 |
| 2.12 | Ciclos: Carregando parâmetros em ciclos de usuário..... | 2-113 |
| 2.13 | Macros | 2-118 |
| Gerenciamento de programas e arquivos | | 3-121 |
| 3.1 | Generalidades..... | 3-122 |
| 3.2 | Memória de programa | 3-123 |
| 3.3 | Memória de usuário | 3-128 |
| 3.4 | Definindo dados de usuário | 3-131 |
| 3.5 | Definindo níveis de proteção para dados de usuário (GUD)..... | 3-135 |
| 3.6 | Ativação automática das GUDs e MACs (SW 4.4 em diante)..... | 3-137 |
| Zonas de proteção | | 4-139 |
| 4.1 | Definindo zonas de proteção CPROTDEF, NPROTDEF | 4-140 |
| 4.2 | Ativando/desativando as zonas de proteção: CPROT, NPROT | 4-144 |
| Comandos especiais de movimento | | 5-149 |
| 5.1 | Posicionando em pontos codificados, CAC, CIC, CDC, CACP, CACN..... | 5-150 |
| 5.2 | Interpolação Spline | 5-151 |
| 5.3 | Compressor COMPON/COMPCURV | 5-160 |
| 5.4 | Interpolação polinomial, POLY | 5-163 |
| 5.5 | Referência de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante)..... | 5-169 |
| 5.6 | Medições através de pontas de prova, MEAS, MEAW | 5-174 |

| | | |
|------|--|-------|
| 5.7 | Funções adicionais de medição MEASA, MEAWA, MEAC (SW 4 em diante, opção) | 5-177 |
| 5.8 | Funções especiais para usuários OEM..... | 5-187 |
| 5.9 | Critérios programáveis para fim de movimentação (SW 5.1 em diante) | 5-188 |
| 5.10 | Bloco de parâmetros servo programável (SW 5.1 em diante) | 5-189 |

Frames 6-191

| | | |
|-------|---|-------|
| 6.1 | Transformação de coordenadas através de variáveis frame | 6-192 |
| 6.2 | Variáveis frame/carga de valores em variáveis frame | 6-197 |
| 6.3 | Deslocamento grosso/fino (coarse/fine)..... | 6-204 |
| 6.4 | Deslocamento DRF | 6-205 |
| 6.5 | Deslocamento de origem externo | 6-206 |
| 6.6 | Programando deslocamentos preset, PRESETON..... | 6-207 |
| 6.7 | Desativando frames..... | 6-208 |
| 6.8 | Cálculo do frame a partir de três pontos de medição na área, MEAFRAME | 6-209 |
| 6.9 | Frames globais NCU (SW 5 em diante) | 6-212 |
| 6.9.1 | Frames específicos do canal | 6-213 |
| 6.9.2 | Frames ativos no canal..... | 6-215 |

Transformações 7-219

| | | |
|-------|---|-------|
| 7.1 | Transformações em três, quatro e cinco eixos: TRAORI..... | 7-220 |
| 7.1.1 | Programando a orientação da ferramenta..... | 7-223 |
| 7.1.2 | Eixos de orientação, ORIWCS, ORIMCS..... | 7-228 |
| 7.1.3 | Posições singulares e como elas são manuseadas | 7-229 |
| 7.1.4 | Eixos de orientação (SW 5.2 em diante) | 7-230 |
| 7.1.5 | Movimento cartesiano PTP (SW 5.2 em diante) | 7-233 |
| 7.2 | Fresando em peças torneadas: TRANSMIT | 7-238 |
| 7.3 | Transformação de superfície cilíndrica: TRACYL..... | 7-241 |
| 7.4 | Eixo inclinado: TRAANG | 7-247 |
| 7.5 | Condições suplementares durante a seleção de uma transformação..... | 7-251 |
| 7.6 | Desativando transformações: TRAFOOF | 7-253 |
| 7.7 | Encadeamento de transformações | 7-254 |
| 7.8 | Eixos geométricos comutáveis, GEOAX..... | 7-257 |

Corretores de ferramenta 8-263

| | | |
|-----|-----------------------------|-------|
| 8.1 | Memória de corretores | 8-264 |
|-----|-----------------------------|-------|

| | |
|---|---------------|
| 8.2 Comandos para o gerenciamento de ferramentas..... | 8-266 |
| 8.3 Corretor de ferramenta “online” PUTFTOCF, PUTFTOC, FTOCON, FTOCOF | 8-269 |
| 8.4 Mantendo a correção de raio em nível constante, CUTCONON (SW 4 em diante) | 8-276 |
| 8.5 Ativando corretores de ferramenta 3D | 8-279 |
| 8.6 Orientação da ferramenta | 8-287 |
| 8.7 Utilização livre de números D, número do incerto CE (A partir da SW5)..... | 8-291 |
| 8.7.1 Checar números D (CHKDNO)..... | 8-292 |
| 8.7.2 Renomeando números D (GETDNO, SETDNO)..... | 8-293 |
| 8.7.3 Números T para os números D especificados (GETACTTD) | 8-294 |
| 8.7.4 Definir números D restantes como inválidos | 8-295 |
| 8.8 Cinemática do porta ferramenta..... | 8-296 |
| Comportamento dos movimentos na trajetória | 9-301 |
| 9.1 Controle tangencial TANG, TANGON, TANGOF | 9-302 |
| 9.2 Movimento acoplado TRAILON, TRAILOF | 9-307 |
| 9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV | 9-311 |
| 9.4 Valor de acoplamento axial principal, LEADON, LEADOF | 9-319 |
| 9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO | 9-325 |
| 9.6 Programa c/memória p/pré processamento, STARTFIFO, STOPFIFO, STOPRE..... | 9-330 |
| 9.7 Reposicionamento no contorno, REPOSA, REPOSL, REPOSQ, REPOSH | 9-332 |
| Movimentos com ações síncronas | 10-337 |
| 10.1 Estrutura, informações básicas | 10-339 |
| 10.1.1 Programação e elementos de comando..... | 10-341 |
| 10.1.2 Faixa válida: Número de identificação ID | 10-342 |
| 10.1.3 Instruções..... | 10-343 |
| 10.1.4 Ações | 10-346 |
| 10.1.5 Relação das ações síncronas | 10-348 |
| 10.2 Módulos básicos para condições e ações | 10-350 |
| 10.3 Variáveis em tempo real especiais para ações síncronas | 10-353 |
| 10.3.1 Memórias/contadores \$AC_MARKER[n]..... | 10-353 |
| 10.3.2 Variável tipo temporizador \$AC_TIMER[n], a partir da SW 4 | 10-353 |
| 10.3.3 Parâmetros em ações síncronas \$AC_PARAM[n] | 10-354 |
| 10.3.4 Acesso a parâmetros R \$Rxx | 10-355 |
| 10.3.5 Leitura/escrita em dados de máquina e dados setting , a partir da SW4 | 10-356 |
| 10.3.6 Variáveis FIFO \$AC_FIFO1[n] ... \$AC_FIFO10[n], SW 4 em diante | 10-357 |
| 10.4 Comandos nas ações síncronas | 10-359 |

| | |
|--|---------------|
| 10.4.1 Funções auxiliares..... | 10-359 |
| 10.4.2 Bloqueio da execução do programa RDISABLE | 10-360 |
| 10.4.3 Cancelamento da parada no pré processamento STOPREOF..... | 10-361 |
| 10.4.4 Cancelamento da distância a percorrer..... | 10-362 |
| 10.4.5 Cancela distância a percorrer c/preparação, DELDTG, DELTG (eixo1,...) | 10-362 |
| 10.4.7 Definição de polinômio, FCTDEF, sincronizada com o bloco | 10-364 |
| 10.4.8 Controle de potência laser | 10-366 |
| 10.4.9 Função avaliação SYNFACT | 10-367 |
| 10.4.10 Controle adaptável (aditivo) | 10-368 |
| 10.4.11 Controle adaptável (multiplicativo)..... | 10-369 |
| 10.4.12 Controle de tolerância com compensação limitada..... | 10-370 |
| 10.4.13 Corretores de ferramenta online FTOC..... | 10-372 |
| 10.4.14 Movimentos de posicionamento | 10-374 |
| 10.4.15 Posicionando eixo POS | 10-376 |
| 10.4.16 Inicia/para movimento MOV | 10-376 |
| 10.4.17 Avanço axial: FA..... | 10-377 |
| 10.4.18 Limite de software..... | 10-377 |
| 10.4.19 Coordenação de eixos | 10-378 |
| 10.4.20 Carregando o valor atual | 10-379 |
| 10.4.21 Movimentos com o fuso..... | 10-380 |
| 10.4.22 Movimento de eixos acoplados: TRAILON, TRAILOF | 10-381 |
| 10.4.23 Valor principal de acoplamento LEADON, LEADOF | 10-382 |
| 10.4.24 Medição | 10-384 |
| 10.4.25 Marcas de espera (WAIT) ligar/cancelar: SETM, CLEARM..... | 10-384 |
| 10.4.26 Resposta a erros | 10-385 |
| 10.5 Ciclos tecnológicos..... | 10-386 |
| 10.5.1 Travar, destravar, resetar: LOCK, UNLOCK, RESET | 10-388 |
| 10.6 Cancelamento de uma ação síncrona: CANCEL | 10-390 |
| 10.7 Condições adicionais..... | 10-391 |
| Oscilações | 11-395 |
| 11.1 Oscilação assíncrona | 11-396 |
| 11.2 Oscilação controlada através de ações síncronas | 11-403 |
| Puncionar e riscar | 12-415 |
| 12.1 Ativação, desativação..... | 12-416 |
| 12.1.1 Instruções | 12-416 |
| 12.1.2 Uso de comandos M..... | 12-419 |
| 12.2 Segmentação automática de trajetória..... | 12-420 |
| 12.2.1 Segmentação da trajetória para eixos de trajetória | 12-421 |
| 12.2.2 Segmentação de trajetória para um único eixo | 12-422 |

| | |
|------------------------------------|--------|
| 12.2.3 Exemplo de programação..... | 12-424 |
|------------------------------------|--------|

Funções adicionais **13-427**

| | |
|---|--------|
| 13.1 Funções de eixo AXNAME, SPI, ISAXIS | 13-428 |
| 13.2 Aprendendo as características de compensação: QECLRNON, QECLRNOF | 13-429 |
| 13.3 Fuso síncrono | 13-431 |
| 13.4 EG: Câmbio eletrônico (SW 5 em diante) | 13-441 |
| 13.4.1 Definir o câmbio eletrônico: EGDEF | 13-441 |
| 13.4.2 Ativando o câmbio eletrônico | 13-443 |
| 13.4.3 Desativando o câmbio eletrônico | 13-445 |
| 13.4.4 Apaga a definição de um câmbio eletrônico | 13-446 |
| 13.4.5 Avanço por rotação (G95)/câmbio eletrônico (SW 5.2) | 13-446 |
| 13.4.6 EG no Power ON, RESET, troca de modo e pesquisa de bloco | 13-447 |
| 13.4.7 Variáveis de sistema para câmbios eletrônicos | 13-447 |
| 13.5 Expansão de parada e recuo (a partir da SW 5) | 13-447 |
| 13.5.1 Reações independentes do acionamento | 13-448 |
| 13.5.2 Fontes possíveis de disparo | 13-449 |
| 13.5.3 Funções lógicas de disparo: Operação fonte/reação | 13-450 |
| 13.5.4 Ativação | 13-450 |
| 13.5.5 Operação como gerador/Backup de tensão DC | 13-451 |
| 13.5.6 Parada independente do acionamento | 13-451 |
| 13.5.7 Recuo independente do acionamento | 13-452 |
| 13.5.8 Exemplo: Utilizando uma reação independente do acionamento | 13-453 |
| 13.6 Conexão de comunicação (SW 5.2 em diante) | 13-454 |
| 13.7 Recipiente de eixos (SW 5.2 em diante) | 13-457 |
| 13.8 Tempo de execução de programa/contagem de peças (a partir da SW 5.2) | 13-459 |
| 13.8.1 Tempo de execução do programa | 13-459 |
| 13.8.2 Contagem de peças | 13-460 |

Programas de usuário para desbaste **14-463**

| | |
|--|--------|
| 14.1 Funções de apoio para desbaste..... | 14-464 |
| 14.2 Preparação do contorno: CONTPRON..... | 14-465 |
| 14.3 Decodificação do contorno: CONTDCON (a partir da SW 5.2)..... | 14-472 |
| 14.4 Interseção de dois elementos de contorno: INTERSEC..... | 14-476 |
| 14.5 Executando movimentos de elementos de contorno de uma tabela: EXECTAB | 14-478 |
| 14.6 Calcular dados circulares: CALCDAT | 14-479 |

Tabelas **15-481**

| | | |
|---------|--|--------|
| 15.1 | Lista de instruções..... | 15-484 |
| 15.2 | Lista das variáveis de sistema..... | 15-509 |
| 15.2.1 | Parâmetros R..... | 15-509 |
| 15.2.2 | Frames 1..... | 15-509 |
| 15.2.3 | Dados do porta ferramenta..... | 15-510 |
| 15.2.4 | Zonas de proteção específicas do canal | 15-513 |
| 15.2.5 | Parâmetros de ferramenta..... | 15-514 |
| 15.2.6 | Dados de monitoração para o gerenciamento de ferramentas | 15-526 |
| 15.2.7 | Dados de monitoração para usuários OEM..... | 15-527 |
| 15.2.8 | Dados relacionados à ferramenta..... | 15-527 |
| 15.2.9 | Dados de ferramenta relativos à retíficas | 15-529 |
| 15.2.10 | Dados de alojamento do magazine | 15-530 |
| 15.2.11 | Dados de magazine para usuários OEM | 15-531 |
| 15.2.12 | Dados de descrição do magazine para o gerenciamento de ferramentas | 15-532 |
| 15.2.13 | Descrição do magazine de ferramentas para usuários OEM..... | 15-533 |
| 15.2.14 | Parâmetro de módulo do magazine..... | 15-534 |
| 15.2.15 | Valores de compensação do sistema de medição | 15-534 |
| 15.2.16 | Compensação de erro de quadrante | 15-535 |
| 15.2.17 | Compensação de interpolação | 15-536 |
| 15.2.18 | Zonas específicas de proteção NCK | 15-537 |
| 15.2.19 | Dados de sistema | 15-538 |
| 15.2.20 | Frames 2..... | 15-539 |
| 15.2.21 | Dados de ferramentas | 15-539 |
| 15.2.22 | Valores programados | 15-541 |
| 15.2.23 | Grupos G | 15-541 |
| 15.2.24 | Estados do canal | 15-543 |
| 15.2.25 | Ações síncronas | 15-546 |
| 15.2.26 | I/Os | 15-548 |
| 15.2.27 | Leitura e escrita de variáveis PLC | 15-549 |
| 15.2.28 | Conexão NCU..... | 15-549 |
| 15.2.29 | I/O PLC direto | 15-550 |
| 15.2.30 | Gerenciamento de ferramenta..... | 15-551 |
| 15.2.31 | Temporizadores..... | 15-552 |
| 15.2.32 | Movimento da peça | 15-553 |
| 15.2.33 | Velocidades | 15-554 |
| 15.2.34 | Fusos | 15-555 |
| 15.2.35 | Valores de polinômios para ações síncronas..... | 15-557 |
| 15.2.36 | Estado do canal | 15-558 |
| 15.2.37 | Posições | 15-558 |
| 15.2.38 | Eixos indexados..... | 15-559 |
| 15.2.39 | Limite de frequência do encoder | 15-559 |
| 15.2.40 | Valores do encoder..... | 15-560 |
| 15.2.41 | Medição axial | 15-561 |

| | |
|---|--------|
| 15.2.42 Deslocamentos | 15-561 |
| 15.2.43 Distâncias axiais | 15-562 |
| 15.2.44 Oscilação | 15-563 |
| 15.2.45 Velocidades dos eixos | 15-564 |
| 15.2.46 Dados de acionamento | 15-565 |
| 15.2.47 Estado do eixo | 15-566 |
| 15.2.48 Câmbio eletrônico 1 | 15-567 |
| 15.2.49 Valor principal de acoplamento..... | 15-568 |
| 15.2.50 Fuso sincronizado | 15-569 |
| 15.2.51 Safety Integrated 1 | 15-569 |
| 15.2.52 Parada prolongada e recuo..... | 15-570 |
| 15.2.53 Recipiente de eixos..... | 15-571 |
| 15.2.54 Câmbio eletrônico 2 | 15-571 |
| 15.2.55 Safety Integrated 2..... | 15-572 |

Apêndice

A-575

- A Índice..... **Erro! Indicador não definido.-Erro! Indicador não definido.**
- B Comandos, Identificadores..... **Erro! Indicador não definido.-Erro! Indicador não definido.**

Prefácio

Estrutura do manual

A documentação SINUMERIK encontra-se dividida em 3 níveis:

- Documentação geral
- Documentação para os usuários
- Documentação de fabricante/de serviço

Grupo alvo

A presente documentação dirige-se aos programadores. Fornece informações detalhadas para a programação dos controles SINUMERIK 840D/810D e SINUMERIK FM-NC.

Estado de fornecimento

Este manual de programação descreve o funcionamento do comando em seu estado de fornecimento (standard). Ampliações ou alterações implementadas pelo fabricante da máquina são documentadas pelo fabricante da máquina.

Para maiores detalhes relativos a outros manuais dos SINUMERIK 840D/840Di/810D e SINUMERIK FM-NC e outras publicações a respeito de todos os controles SINUMERIK (p.e. interface universal, ciclos de medição....) favor contatar seu representante Siemens local.

No controle podem ser encontradas outras funções que não foram explicadas nesta documentação. Porém, não há qualquer direito a estas funções no caso de um fornecimento novo ou em caso assistência técnica.

Validade

Este manual de programação é válido para os seguintes controles:

| | |
|---------------------------------------|-----|
| SINUMERIK 840D | SW5 |
| SINUMERIK 840Di | SW5 |
| SINUMERIK 840DE (Versão p/exportação) | SW5 |
| SINUMERIK 810D | SW3 |
| SINUMERIK 810DE (Versão p/exportação) | SW3 |
| SINUMERIK FM-NC | SW3 |

Versão de exportação

As seguintes funções não fazem parte da versão de exportação:

| Função | FM-NC | 810DE | 840DE |
|---|-------|-------|-----------------|
| Pacote de usinagem com 5 eixos | – | – | – |
| Pacote transformação Handling (5 eixos) | – | – | – |
| Interpolação de eixos múltiplos (> 4 eixos) | – | – | – |
| Interpolação helicoidal 2D+6 | – | – | – |
| Ações síncronas nível 2 | – | – | O ¹⁾ |
| Medição nível 2 | – | – | O ¹⁾ |
| Controle adaptável | – | – | O ¹⁾ |
| Dressamento contínuo | – | – | O ¹⁾ |
| Utilização de ciclos compilados (OEM) | – | – | – |
| Compensação de flecha multidimensional | – | – | O ¹⁾ |

– Função não possível

1) Funcionalidade limitada

Estrutura das descrições

Todos os ciclos e todas as possibilidades de programação foram descritos – quando apropriado e possível - segundo a mesma estrutura interna. A divisão em vários níveis de informação permite um acesso rápido às informações que você procura.

1. Visualização rápida



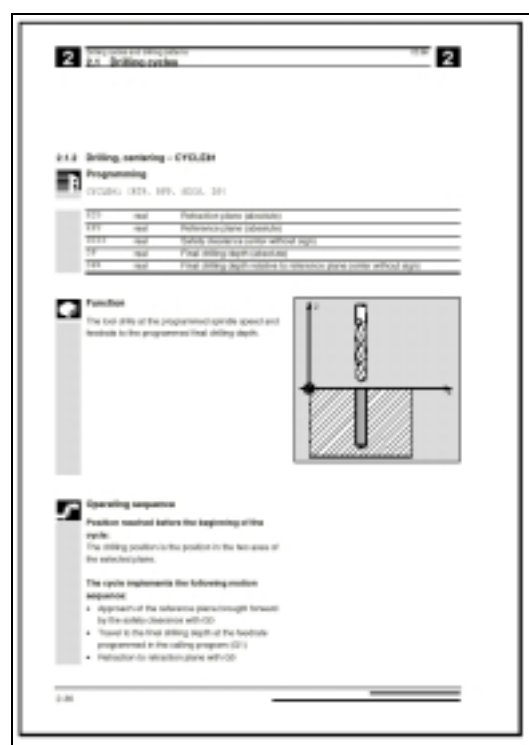
Caso você procure um comando raramente utilizado ou o significado de um parâmetro, você pode localizar de forma rápida a forma de programação em conjunto com explicações dos comandos e parâmetros.



Esta informação é sempre apresentada no início da página.

Nota:

Para manter esta documentação o mais compacta possível, não é sempre possível listar todos os tipos de representação disponíveis para os comandos e parâmetros. Entretanto, são representados da forma mais usual ao chão de fábrica.



2. Explicações detalhadas

Na parte teórica encontra-se descrito de forma detalhada o seguinte:

Para que preciso deste comando?

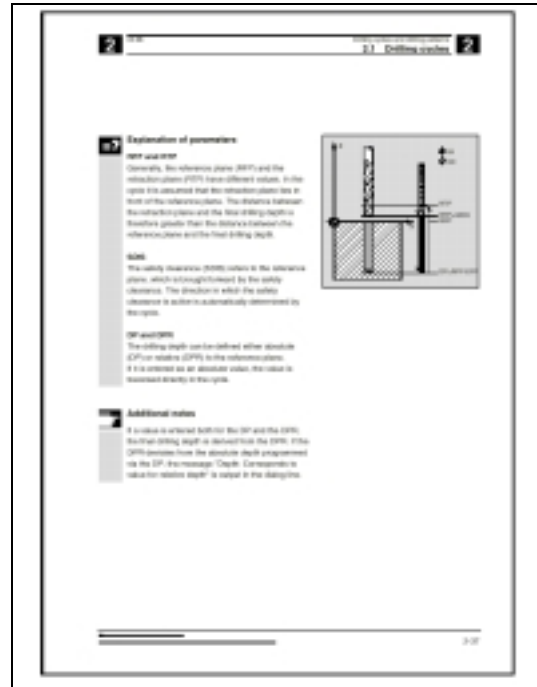
Que provoca o comando?

Qual é o decurso?

Que provocam os parâmetros?

Que deve ser especialmente observado?

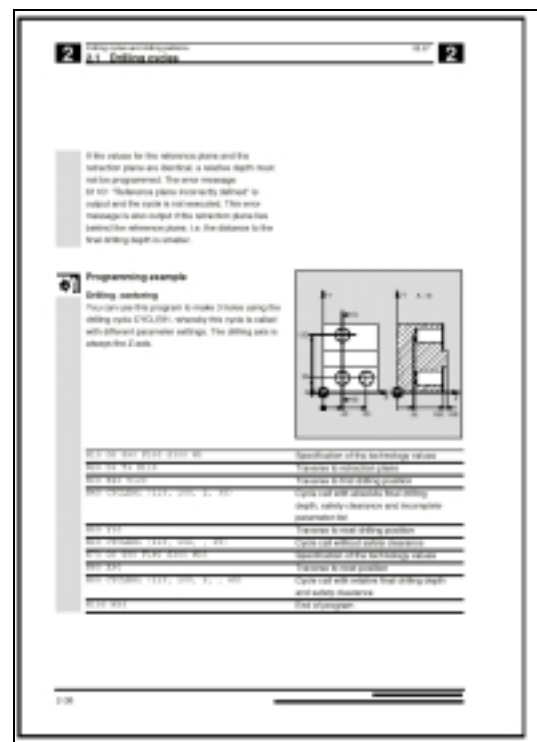
As partes teóricas servem de base da aprendizagem especialmente para principiantes que entram na matéria de CN. Por favor, leia o manual pelo menos uma vez a fim de formar-se uma idéia do volume e da potência do Seu controle SINUMERIK.













3. Da teoria para a prática

A forma da utilização dos comandos encontra-se descrita no exemplo de programação.

Para todos os comandos há um exemplo de utilização após a parte teórica.



| | |
|---|--|
| | Explicação dos símbolos |
|  | Seqüência de operações |
|  | Explicação |
|  | Função |
|  | Parâmetros |
|  | Exemplo de programação |
|  | Programação |
|  | Informações adicionais |
|  | Referências cruzadas a outras documentações e capítulos |
|  | Informações importantes e notas de sobre seguranças |
|  | |

Para sua informação

O seu SIEMENS 840D/810D ou FM-NC está construído conforme as mais recentes tecnologias e obedece reconhecidas normas, especificações e regras de segurança.

Equipamentos adicionais

Equipamentos adicionais, módulos de ampliação e níveis de configuração especiais oferecidos por SIEMENS permitem a ampliação apropriada do campo de aplicação dos controles SIEMENS.

Pessoal

Somente **pessoal especialmente treinado, autorizado e experiente** deve trabalhar com o controle. É proibido de trabalhar no controle, mesmo por um curto período, sem ter a qualificação necessária.

As competências correspondentes do pessoal que se ocupa da preparação, do manejo e da manutenção têm de ser claramente especificadas e a sua observação tem de ser supervisionada.

Ações

Antes de colocar o controle em funcionamento, tem de ser garantido que as instruções de serviço tenham sido lidas e compreendidas pelo pessoal competente. Para isso a empresa é **obrigada a controlar** permanentemente o estado técnico total do controle (defeitos e danos aparentes, assim como alterações do comportamento funcional).

Assistência técnica

Só **pessoas qualificadas e com formação especializada** podem efetuar reparações conforme as indicações nas Instruções de manutenção.

Têm de ser observadas todas as prescrições de segurança correspondentes.

**Nota**

Uma aplicação **não conforme as disposições** que excluirá **toda responsabilidade do fabricante** é:

Toda aplicação que difere dos pontos atrás mencionados ou ultrapassa as disposições.

Se for trabalhado com o controle no **estado técnico não impecável**, sem ter consciência da segurança e de perigos e sem observar todos as instruções no Manual de serviço.

Se falhas que podem reduzir a segurança não forem eliminadas **antes de** colocar o controle em funcionamento.

Cada **alteração, comutação em ponte** ou **colocação fora do funcionamento** de dispositivos no controle que servem para o funcionamento impecável, a utilização não limitada assim como para a segurança ativa e passiva.



Podem surgir **perigos imprevisíveis** para:

- a saúde e a vida de pessoas,
- o controle, a máquina e outros bens da empresa e do usuário.

[illegible]

A

SIEMENS AG

A&D MC IS

P.O. Box 3180

D-91050 Erlangen

Germany

(Tel. 0180/525–8008/5009 [Hotline])

Fax +49(0)9131/98–1145

email: motioncontrol.docu@.siemens.de)

Propostas

Correções

Para a Publicação/Manual:

SINUMERIK 840D/840Di/810D/FM-NC
Manual de programação
Avançado

Documentação do usuário

Remetente

Nome

Empresa/departamento

Endereço:

Fone: /

Fax: /

N.º de encomenda: 6FC5298-5AB10-0BP2

Edição: 04.00

Caso você venha a perceber quaisquer erros de impressão durante a leitura deste manual, favor nos informar através deste formulário. Agradecemos também por sugestões para melhorias.

Sugestões e/ou correções

Publicada pela Siemens S/A
Departamento Técnica de automatização
Campo de ação Sistemas de automatização
para máquinas-ferramenta, robôs e
máquinas especiais
Caixa postal 3180, D - 91050 Erlangen
Federal Republic of Germany

Qualidade examinada de Siemens para Software e
treinamento segundo DIN ISO 9000, N^o. Reg. 2160-01.
O conteúdo desta documentação foi imprimido em
papel branqueado sem cloro, que não prejudica o meio
ambiente.
Copyright Siemens AG 2000 Todos os direitos
reservados. Tópicos estão sujeitos à alterações sem
aviso prévio

Progresso
em Automação.
Siemens

Programação NC flexível

| | | |
|--------|--|------|
| 1.1 | Variáveis e parâmetros aritméticos | 1-22 |
| 1.2 | Definição de variáveis..... | 1-25 |
| 1.3 | Definição de array (matrizes) | 1-30 |
| 1.4 | Programação indireta | 1-36 |
| 1.5 | Carga de valores | 1-38 |
| 1.6 | Operações aritméticas/funções | 1-39 |
| 1.7 | Comparação e operadores lógicos..... | 1-41 |
| 1.8 | Prioridade dos operadores | 1-44 |
| 1.9 | Possibilidades de conversão | 1-45 |
| 1.10 | Operações com string | 1-46 |
| 1.10.1 | Conversão de tipo | 1-47 |
| 1.10.2 | Encadeando seqüências de caracteres | 1-49 |
| 1.10.3 | Conversão de caracteres maiúsculas/minúsculas | 1-50 |
| 1.10.4 | Comprimento da string..... | 1-51 |
| 1.10.5 | Procura por caracter/seqüência de caracteres na string..... | 1-51 |
| 1.10.6 | Seleção de uma substring..... | 1-53 |
| 1.10.7 | Selecionando um único caracter | 1-54 |
| 1.11 | Instrução CASE | 1-56 |
| 1.12 | Estruturas de controle | 1-58 |
| 1.13 | Coordenação de programas..... | 1-63 |
| 1.14 | Rotina de interrupção | 1-68 |
| 1.15 | Transferência de eixo, transferência de fuso | 1-76 |
| 1.16 | NEWCONF: Ativando dados de máquina (A partir da SW 4.3) | 1-80 |
| 1.17 | WRITE: Escrita de arquivo (a partir da SW 4.3) | 1-81 |
| 1.18 | DELETE: Apagar arquivo (a partir da SW 4.3)..... | 1-83 |
| 1.19 | READ: Lê linhas de arquivos (a partir da SW 5.2) | 1-84 |
| 1.20 | ISFILE: Arquivo disponível na memória de usuário NCK (a partir da SW5.2) | 1-87 |
| 1.21 | CHECKSUM: Criar o checksum em um array (> SW 5.2) | 1-88 |

1.1 Variáveis e parâmetros aritméticos



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.1 Variáveis e parâmetros aritméticos



Função

Variáveis podem ser utilizados ao invés de valores fixos para aumentar a flexibilidade de um programa. A você pode gravar valores como valores de medição, posições, etc. O mesmo programa pode ser utilizado para diferentes geometrias.

Um programador qualificado pode utilizar cálculos com variáveis e instruções de salto para criar programas altamente flexíveis, reduzindo consideravelmente o trabalho de programação necessário.



Tipos de variáveis

O controle faz a distinção entre três tipos de variáveis:

| | |
|------------------------|--|
| Variáveis de usuário | Variáveis cujos nomes e tipos são definidas pelo usuário, p.e. parâmetros aritméticos. |
| Parâmetros aritméticos | Variáveis aritméticas especiais para as quais o endereço R, seguido de um número, encontram-se disponíveis. Estas variáveis aritméticas pré-definidas são do tipo REAL. |
| Variáveis de sistema | Variáveis que o controle disponibiliza que podem ser processadas (lidas/escritas) em um programa. Com as variáveis de sistema é possível ler ou escrever nos deslocamentos de origem, corretores de ferramentas, valores atuais de posição, valores medidos nos eixos, estados de controle, etc. (Em anexo, as variáveis de sistema encontram-se descritas). |

1.1 Variáveis e parâmetros aritméticos



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Tipos de variáveis

| Tipo | Significado | Faixa de valores |
|--------|--|---|
| INT | Inteiro com sinal | $\pm(2^{31} - 1)$ |
| REAL | Números reais (frações com ponto decimal, LONG REAL na IEEE) | $\pm(10^{-300} \dots 10^{+300})$ |
| BOOL | Valores booleanos: TRUE (1) e FALSE (0) | 1, 0 |
| CHAR | 1 caractere ASCII especificado por código | 0 ... 255 |
| STRING | Seqüência de caracteres, quantidade de caracteres em [...], máximo de 200 caracteres | Seqüência de valores, entre 0 ... 255 |
| AXIS | Somente nomes de eixos (endereços de eixos) | Todos os identificadores de eixos e fusos de um canal |
| FRAME | Parâmetros geométricos para translação, rotação, escala, espelhamento, etc. Vide capítulo 4. | |



Variáveis aritméticas

100 variáveis aritméticas do tipo REAL encontram-se disponíveis sob o endereço R sem definições adicionais.



O número exato destas variáveis aritméticas (até 1000) é definido em dados de máquina.

Exemplo: R10=5

Variáveis de sistema

1.1 Variáveis e parâmetros aritméticos



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



O nome de uma variável de sistema é sempre identificado pelo carácter “\$” seguido do nome específico.

Relação dos tipos de variáveis de sistema

| Primeira letra | Significado |
|----------------|--------------------------------------|
| \$M | Dado de máquina |
| \$S | Dado setting |
| \$T | Dado do gerenciamento de ferramentas |
| \$P | Valores programados |
| \$A | Valores atuais |
| \$V | Valores de serviço |
| | |
| Segunda letra | Significado |
| N | Global NCK |
| C | Específica de canal |
| A | Específica de eixo |

Exemplo: \$AA_IM

Significado: Valor de posição atual do eixo no sistema de coordenadas da máquina

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

1.2 Definição de variáveis



Variáveis de usuário

Adicionalmente às variáveis pré-definidas, os programadores podem definir suas próprias variáveis e carregar nelas valores.

Variáveis locais (LUD) são válidas somente no programa onde são definidas.

Variáveis globais (GUD) são válidas em todos os programas.

SW 4.4 em diante:

As variáveis locais (LUD) definidas em um programa principal podem ser redefinidas como variáveis globais de programa (PUD) através de dados de máquina.



Fabricante da máquina

Vide as especificações do fabricante da máquina.

Caso estas variáveis sejam definidas no programa principal, podem continuar válidas em todos os níveis de subrotinas chamadas através deste programa. São criadas no início do programa e apagadas com o fim do programa ou reset.

Exemplo:

```
$MN_LUD_EXTENDED_SCOPE=1
PROC MAIN                ;programa principal
DEF INT VAR1             ;definição PUD
...
SUB2                     ;chamada de
...                     subrotina
M30
PROC SUB2                ;Chamada de
SUB2                     subrotina
  DEF INT VAR2           ;definição LUD
  ...
  IF (VAR1==1)           ;leitura PUD
    VAR1=VAR1+1          ;leitura e escrita
                        ;PUD
    VAR2=1               ;escrita LUD
  ENDIF
SUB3                     ;chamada de
...                     subprograma
M17
PROC SUB3                ;subprograma SUB3
...
```

1.2 Definição de variáveis



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

```
IF (VAR1==1)      ;leitura PUD
  VAR1=VAR1+1     ;leitura e escrita
                  ;PUD
  VAR2=1          ;erro: LUD de SUB2
                  ;não reconhecida
```

IF

...

M17

Caso o dado de máquina

\$MN_LUD_EXTENDED_SCOPE esteja ligado, não será mais permitida a definição de variáveis de mesmo nome em programas e subprogramas.

Nomes de variáveis

O nome de uma variável pode conter até 30 caracteres. Os dois primeiros devem ser letras ou o caracter sublinhado.

O caractere "\$" não pode ser utilizado na definição de variáveis de usuário, pois é reservado para as variáveis de sistema

Programação

DEF INT nome

OU DEF INT nome=Value

DEF REAL nome

OU DEF REAL nome1,nome2=3,nome4

OU DEF REAL nome[array índice1,array
índice2]

DEF BOOL nome

DEF CHAR nome

OU DEF CHAR nome[array índice]=("A","B",...)

DEF STRING[comprimento] nome

DEF AXIS nome

OU DEF AXIS nome[array índice]

DEF FRAME nome

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Caso não seja atribuído um valor à variável durante a definição, o sistema a inicializa com zero.

Variáveis devem ser definidas antes de serem utilizadas no programa. A definição deve ser feita em um bloco separado, somente um tipo de variável pode ser definido em um bloco.



Explicação

| | |
|--------|---|
| INT | Variável tipo Integer, ou seja, número inteiro |
| REAL | Variável tipo Real, ou seja, fração com ponto decimal |
| BOOL | Variável tipo Bool, ou seja, 1 ou 0 (TRUE ou FALSE) |
| CHAR | Variável tipo Char, ou seja, caracter ASCII especificado por código (0 a 225) |
| STRING | Variável tipo String, ou seja, sequência de caracteres |
| AXIS | Variável tipo Axis, ou seja, endereço de eixos e fusos |
| FRAME | Variável tipo Frame, ou seja, parâmetros geométricos |
| Nome | Nome de variável |



Exemplos de programação

Variável tipo INT

| | |
|------------------|---|
| DEF INT NUMBER | Uma variável tipo INTEGER é criada com o nome NUMBER. O sistema inicializa a variável com zero. |
| DEF INT NUMBER=7 | Uma variável tipo INTEGER é criada com o nome NUMBER. O sistema inicializa a variável com o valor 7. |

Variável tipo REAL

| | |
|--|--|
| DEF REAL DEPTH | Uma variável tipo REAL é criada com o nome DEPTH. O sistema inicializa a variável com zero (0.0). |
| DEF REAL DEPTH=6.25 | Uma variável tipo REAL é criada com o nome DEPTH. O sistema inicializa a variável com 6.25. |
| DEF REAL DEPTH=3.1, LENGTH=2, QUANTITY | É possível inicializar diversas variáveis em |

1.2 Definição de variáveis



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

| | |
|--|--|
| | uma única linha. |
| Variável tipo BOOL | |
| DEF BOOL IF_TOOMUCH | Uma variável tipo BOOL é criada com o nome IF_TOOMUCH. O sistema inicializa a variável com o valor zero (FALSE). |
| DEF BOOL IF_TOOMUCH=1 ou DEF BOOL IF_TOOMUCH=TRUE ou DEF BOOL IF_TOOMUCH=FALSE | Uma variável tipo BOOL é criada com o nome IF_TOOMUCH. |
| Variável tipo CHAR | |
| DEF CHAR GUSTAV_1=65 | Você pode atribuir um código para o caracter ASCII em uma variável tipo CHAR ou |
| DEF CHAR GUSTAV_1="A" | Atribuir o caracter ASCII diretamente (65 é o código para a letra "A"). |
| Variável tipo STRING | |
| DEF STRING[6] SAMPLE_1="START" | Variáveis tipo STRING podem gravar uma seqüência de caracteres. A quantidade máxima de caracteres é especificada entre colchetes após o tipo de variável. |
| Variável tipo AXIS | |
| DEF AXIS AXISNAME=(X1) | A variável do tipo AXIS de nome AXISNAME e contém um identificador de eixo pertencente a um canal – aqui X1 (nomes de eixos com endereços expandidos devem ser colocados entre parênteses) |
| Variável tipo FRAME | |
| DEF FRAME INCLINE_1 | A variável tipo FRAME de nome INCLINE_1. |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Informações adicionais

Em uma variável tipo AXIS são gravados os identificadores de eixos e fusos de um canal.

Nota:

Os nomes de eixos com expansões devem estar entre parênteses



Exemplo de programação com variáveis locais

```
DEF INT COUNT
LOOP: G0 X... ;Loop
COUNT=COUNT+1
IF COUNTER<50 GOTOB LOOP
M30
```



Exemplo de programação

Procura eixos geométricos existentes

```
DEF AXIS ABSCISSA; ;primeiro eixo geométrico
IF ISAXIS(1)==FALSE GOTOF CONTINUE
ABSCISSA = $P_AXN1
...
CONTINUE:
```

Programação indireta do fuso

```
DEF AXIS SPINDLE
SPINDLE=(S1)
OVRA[SPINDLE]=80 ;Override do fuso= 80%
SPINDLE=(S3)
...
```

1.3 Definição de array (matrizes)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.3 Definição de array (matrizes)



Programação

```
DEF CHAR NAME[n,m]
DEF INT NAME[n,m]
DEF REAL NAME[n,m]
DEF AXIS NAME[n,m]
DEF FRAME NAME[n,m]
DEF STRING[compr.da string] NAME[m]
DEF BOOL[n,m]
```



Explicação

| | |
|-------------------------------------|---|
| INT NOME[n,m] | Variáveis do tipo (CHAR, INTEGER, REAL, AXIS, FRAME, BOOL) n = tamanho do array (matriz) primeira dimensão m = tamanho do array (matriz) Segunda dimensão |
| REAL NOME[n,m] | |
| DEF STRING[compr.da string] NAME[m] | Os dados tipo STRING podem ser definidos somente em arrays de uma dimensão |
| NOME | Nome da variável |

As variáveis tipo BOOL e CHAR ocupam o mesmo espaço de memória.

SW 3 em diante:

O tamanho máximo dos arrays é definido em dados de máquina.



Fabricante de máquinas

Vide as especificações do fabricante da máquina.

| Tipo | Memória utilizada para cada elemento de um array (matriz) |
|--------|---|
| BOOL | 1 byte |
| CHAR | 1 byte |
| INT | 4 bytes |
| REAL | 8 bytes |
| STRING | Comprimento da string + 1 |
| FRAME | ~ 400 bytes, dependendo da quantidade de eixos |
| AXIS | 4 bytes |

O tamanho máximo de um array determina o tamanho de blocos de

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

memória a serem gerenciados. Não devem ser maiores que o necessário.

Standard 812 bytes.

Caso não utilizados arrays maiores, favor selecionar 256 bytes.

SW 4 em diante:

Um array pode ser maior que um bloco de memória.

O dado de máquina para tamanho de blocos de memória deve ser programado de forma que arrays sejam fragmentados somente em casos excepcionais.

Standard: 256 bytes

Vide acima para necessidade de memória por elemento.

Exemplo:

Dados de usuário globais devem conter dados de máquina de PLC para ligar e desligar o controle (definição de arrays BOOL).

Informações adicionais

Arrays de no máximo 2 dimensões podem ser definidos.

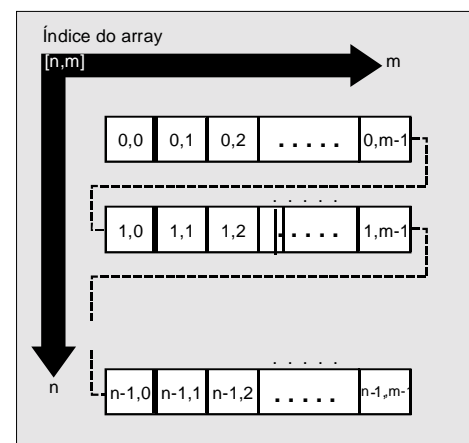
Arrays com variáveis STRING podem Ter somente uma dimensão. O comprimento da string é especificado após o tipo de dado string (seqüência de caracteres).



Índice do array

Os elementos de um array podem ser acessados através do índice do array, podendo-se ler ou escrever valores nestes elementos

O primeiro elemento começa com o índice [0,0]. No caso de um array de dimensão [3,4], por exemplo, o índice máximo é [2,3].



1.3 Definição de array (matrizes)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



No exemplo acima, os valores de inicialização combinam com o índice do elemento do array para ilustrar a sequência dos elementos individuais do array.



Carga de arrays

Valores iniciais podem ser carregados durante a execução do programa ou quando o array é definido.

O índice direito é o inicialmente incrementado em um array bidimensional.



Carga através de lista de valores, SET

1. Opções durante a definição de arrays

```
DEF Type VARIABLE = SET(VALOR)
DEF Type ARRAY[n,m] = SET(VALOR,
valor, ...)
```

Ou:

```
DEF Type VARIABLE = valor
DEF Type ARRAY[n,m] = (valor, valor,
...)
```

- A quantidade de elementos carregados corresponde à quantidade de valores programados.
- Elementos de array sem valores (lacunas na lista de valores) são automaticamente carregados com o valor "0".
- Não devem existir lacunas para variáveis do tipo AXIS.
- Caso programados mais elementos que os existentes no array, o sistema emitirá um alarme.

Exemplo:

```
DEF REAL ARRAY[2,3]=(10, 20, 30, 40)
```



Você pode especificar SET de durante a criação do array.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2. Opções durante a execução do programa

```
ARRAY[n,m]= SET(valor, valor, valor,...)
```

```
ARRAY[n,m]= SET(expressão, expressão,  
expressão,...)
```

- Os elementos são carregados com os valores conforme descrito na definição de array acima.
- Expressões podem ser usadas para carga de valores.
- A carga dos valores tem início no índice programado. Os valores podem ser carregados de forma seletiva em subarrays.

Exemplo:

Carga de valores com expressões

```
DEF INT ARRAY[5, 5]  
ARRAY[0,0] = SET(1, 2, 3, 4, 5)  
ARRAY[2,3] = SET(VARIAVEL, 4*5.6)
```

O índice tipo eixo não é processado da mesma forma.

Exemplo:

Inicialização em uma linha

```
$MA_AX_VELO_LIMIT[1, AX1] = SET(1.1, 2.2, 3.3)
```

Corresponde à:

```
$MA_AX_VELO_LIMIT[1,AX1] = 1.1  
$MA_AX_VELO_LIMIT[2,AX1] = 2.2  
$MA_AX_VELO_LIMIT[3,AX1] = 3.3
```

1.3 Definição de array (matrizes)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Inicialização com valores idênticos, REP

1. Opções durante a definição de arrays

```
DEF Type ARRAY[n,m] = REP(valor)
```

Todos os elementos do array são carregados com o mesmo valor (constante).



Variáveis do tipo FRAME não podem ser inicializadas.

Exemplo:

```
DEF REAL ARRAY5[10,3] = REP(9.9)
```

2. Opções durante a execução do programa

```
ARRAY[n,m] = REP(valor)
```

```
ARRAY[n,m] = REP(expressão)
```

- Expressões podem ser utilizadas para a carga de valores.
- Todos os elementos do array são carregados com o mesmo valor.
- A carga de valores é iniciada a partir do índice programado. Valores podem ser carregados em subarrays.



São permitidas variáveis tipo FRAME, e elas podem ser inicializadas de forma simples através do método a seguir.

Exemplo:

Inicialização de todos os elementos com um só valor

```
DEF FRAME FRM[10]
```

```
FRM[5] = REP(CTrans (X,5))
```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Exemplo de programação

A inicialização completa dos arrays de variáveis.

A figura mostra os valores atuais.

```
N10 DEF REAL ARRAY1[10,3] = SET(0, 0, 0, 10, 11, 12, 20, 20, 20, 30,
30, 30, 40, 40, 40,)
```

```
N20 ARRAY1[0,0] = REP(100)
```

```
N30 ARRAY1[5,0] = REP(-100)
```

```
N40 ARRAY1[0,0] = SET(0, 1, 2, -10, -11, -12, -20, -20, -20, -30, , , ,
-40, -40, -50, -60, -70)
```

```
N50 ARRAY1[8,1] = SET(8.1, 8.2, 9.0, 9.1, 9.2)
```

| Índice do array | | | | | | | | | | | | | | | | | 2 |
|-----------------|-------|---|----|----|--|------|------|---|------|------|------|------|------|------|------|------|---|
| 1 | [1,2] | N10: Inicialização com definição | | | N20/N30: Inicialização com valores idênticos | | | N40/N50: Inicialização com valores diferentes | | | | | | | | | |
| | | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | |
| | 0 | 0 | 0 | 0 | 100 | 100 | 100 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | |
| | 1 | 10 | 11 | 12 | 100 | 100 | 100 | -10 | -11 | -12 | -10 | -11 | -12 | -10 | -11 | -12 | |
| | 2 | 20 | 20 | 20 | 100 | 100 | 100 | -20 | -20 | -20 | -20 | -20 | -20 | -20 | -20 | -20 | |
| | 3 | 30 | 30 | 30 | 100 | 100 | 100 | -30 | 0 | 0 | -30 | 0 | 0 | -30 | 0 | 0 | |
| | 4 | 40 | 40 | 40 | 100 | 100 | 100 | 0 | -40 | -40 | 0 | -40 | -40 | 0 | -40 | -40 | |
| | 5 | 0 | 0 | 0 | -100 | -100 | -100 | -50 | -60 | -70 | -50 | -60 | -70 | -50 | -60 | -70 | |
| | 6 | 0 | 0 | 0 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | |
| | 7 | 0 | 0 | 0 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | -100 | |
| | 8 | 0 | 0 | 0 | -100 | -100 | -100 | -100 | 8.1 | 8.2 | -100 | 8.1 | 8.2 | -100 | 8.1 | 8.2 | |
| | 9 | 0 | 0 | 0 | -100 | -100 | -100 | 9.0 | 9.1 | 9.2 | 9.0 | 9.1 | 9.2 | 9.0 | 9.1 | 9.2 | |
| | | Os elementos do array [5,0] a [9,2] foram inicializados com o valor default (0.0) | | | | | | Os elementos do array [3,1] a [4,0] foram inicializados com o valor default (0.0). Os elementos [6,0] a [8,0] não foram alterados | | | | | | | | | |

1.4 Programação indireta



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.4 Programação indireta



A programação indireta permite que programas sejam utilizados de forma universal. Os endereços estendidos (índices) são substituídos por variáveis de tipo adequado.



Todos os endereços podem ser configurados, exceto para:

- N – Número de bloco
- G – Comando G
- L – Subrotina

A programação indireta não pode ser utilizada com endereços ajustáveis.

Exemplo: X[1] não pode ser utilizado ao invés de X1.



Programação

ADDRESS [INDEX]



Exemplo de programação

Fuso

S1=300

Programação direta

DEF INT SPINU=1
S[SPINU]=300

Programação indireta:

Rotação de 300 rpm para o fuso cujo número encontra-se gravado na variável SPINU (1 neste exemplo).

Avanço

FA[U]=300

Programação direta

DEF AXIS AXVAR2=U
FA[AXVAR2]=300

Programação indireta:

Avanço para o eixo de posicionamento cujo endereço está gravado na variável de tipo AXIS de nome AXVAR2.

Valor medido

\$AA_MM[X]

Programação direta

DEF AXIS AXVAR3=X
\$AA_MM[AXVAR3]

Programação indireta:

Valor medido em coordenadas de máquina para o eixo cujo nome encontra-se gravado

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

na variável AXVAR3.

Elemento de array

```
DEF INT ARRAY1[4,5]
```

Programação direta

```
DEFINE DIM1 AS 4
```

Programação indireta:

```
DEFINE DIM2 AS 5
```

Os tamanhos dos campos devem sempre ser especificados como valores fixos nas dimensões do array.

```
DEF INT ARRAY[DIM1,DIM2]
```

```
ARRAY[DIM1-1,DIM2-1]=5
```

Instruções de eixos com variáveis tipo eixos

```
X1=100 X2=200
```

Programação direta

```
DEF AXIS AXVAR1 AXVAR2
```

Programação indireta:

```
AXVAR1=(X1) AXVAR2=(X2)
```

Carga de variáveis

```
AX[AXVAR1]=100 AX[AXVAR2]=200
```

Carga do nome do eixo. Movimento dos eixos gravados nas variáveis para as posições 100 e 200.

Parâmetros de interpolação com variáveis de eixos

```
G2 X100 I20
```

Programação direta

```
DEF AXIS AXVAR1=X
```

Programação indireta:

```
G2 X100 IP[AXVAR1]=20
```

Define e carrega o nome do eixo

Programação indireta do ponto central

Chamada indireta de subprograma

```
CALL "L" << R10
```

Chama o programa cujo número é especificado em R10



Informações adicionais

Os parâmetros R podem também ser interpretados como arrays de uma única dimensão com notação abreviada (R10 corresponde à R[10]).

1.5 Carga de valores



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.5 Carga de valores

Valores de tipos combinados podem ser carregados em variáveis/parâmetros aritméticos no programa.



A carga é sempre feita em um bloco separado, e até duas cargas podem ser feitas por bloco. Carga de endereços de eixos (para instruções de movimentação) sempre necessitam de um bloco separado.



Exemplo de programação

```
R1=10.518 R2=4 VARI1=45
```

Carga de valor numérico

```
X=47.11 Y=R2
```

```
R1=R3 VARI1=R4
```

Carga de valor de mesmo tipo

```
R4=-R5 R7=-VARI8
```

Carga com sinal oposto (permitida somente com os tipos INT e REAL)



Carga em variáveis tipo string

Existe distinção entre caracteres maiúsculos e minúsculos nas variáveis tipo CHAR ou STRING. Caso os caracteres ' ou " existam no texto a ser carregado, estes caracteres devem ser colocados entre '...' (aspas).

Exemplo:

```
MSG("Viene lavorata l' 'ultima  
figura")
```

O texto exibido será 'Viene lavorata l'ultima figura'.

Caracteres que não são exibidos podem ser gravados como constantes binárias ou hexadecimais.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

1.6 Operações aritméticas/funções



Funções aritméticas são utilizadas de forma predominante com parâmetros R e variáveis (ou ainda constantes e funções) do tipo REAL. Os tipos INT e CHAR são também permitidos.



Notações matemáticas padrão são utilizadas em operações aritméticas. A prioridade de execução é indicada entre parênteses. Ângulos são especificados para funções trigonométricas e suas funções inversas (ângulo direito = 90°).



Operadores/funções aritméticas

| | |
|------------|---|
| + | Soma |
| - | Subtração |
| * | Multiplicação |
| / | Divisão |
| | Atenção: (Tipo INT)/(Tipo INT)=(Tipo REAL); Exemplo: 3/4 = 0.75 |
| DIV | Divisão, para variável tipo INT e REAL |
| | Atenção: (Tipo INT)DIV(Tipo INT)=(Tipo INT); Exemplo: 3 DIV 4 = 0 |
| MOD | Divisão de módulo (INT ou REAL) produz o resto de uma divisão INT, e.g. 3 MOD 4=3 |
| :: | Cadeira de operador (para variáveis FRAME) |
| Sin() | Seno |
| COS() | Coseno |
| TAN() | Tangente |
| ASIN() | Arco seno |
| ACOS() | Arco coseno |
| ATAN2(,) | Arco tangente2 |
| SQRT() | Raiz quadrada |
| ABS() | Número absoluto |
| POT() | Quadrado |
| TRUNC() | Truncar para inteiro |
| ROUND() | Arredondar para inteiro |
| LN() | Logaritmo neperiano |
| EXP() | Função exponencial |
| CTrans() | Translação |
| CROT() | Rotação |
| CSCALE() | Alterar escala |
| CMIRROR() | Espelhamento |

1.6 Operações aritméticas/funções



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



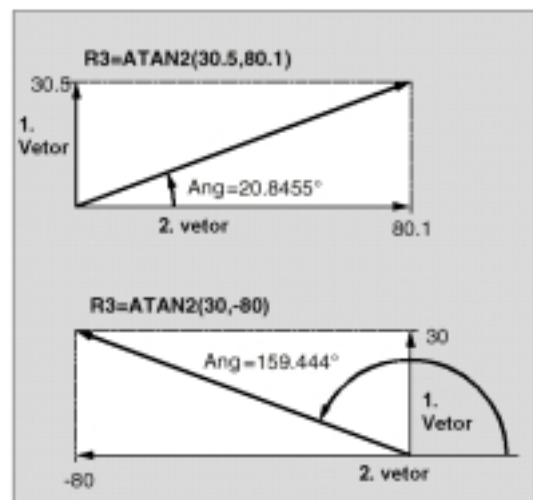
Exemplo de programação

| | |
|---|--|
| $R1=R1+1$ | novo R1 = antigo R1 +1 |
| $R1=R2+R3 \quad R4=R5-R6 \quad R7=R8 \cdot R9$ | |
| $R10=R11/R12 \quad R13=\text{SIN}(25.3)$ | |
| $R14=R1 \cdot R2+R3$ | Multiplicação e divisão tem prioridade sobre adição e subtração |
| $R14=(R1+R2) \cdot R3$ | Parênteses tem prioridade |
| $R15=\text{SQRT}(\text{POT}(R1)+\text{POT}(R2))$ | O conteúdo dos parênteses é calculado primeiro $R15 = \text{raiz quadrada de } (R1^2+R2^2)$ |
| $\text{RESFRAME} = \text{FRAME1}:\text{FRAME2}$ | A cadeia de operadores combina os valores dos frames em um frame resultante ou |
| $\text{FRAME3}=\text{CTRANS}(\dots):\text{CROT}(\dots)$ | carrega valores às componentes do frame |



Função aritmética ATAN2(,)

A função calcula o ângulo do vetor resultante de dois vetores com um ângulo reto entre eles. O resultado estará em um dos quatro quadrantes ($-180 < 0 < +180^\circ$). A referência angular é sempre baseada no segundo valor na direção positiva.



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

1.7 Comparação e operadores lógicos



Comparado operadores

Operadores de comparação podem ser utilizados para variáveis dos tipos CHAR, INT, REAL e BOOL.

Para os tipos STRING, AXIS e FRAME, as comparações possíveis são: == e <>.

O resultado de uma comparação é sempre tipo BOOL.

Comparação de operadores podem ser utilizadas, por exemplo, para determinar condições de salto. Expressões complexas podem também ser comparadas.



Significado dos operadores de comparação

| | |
|----|-------------------------|
| == | Igual a |
| <> | Diferente de |
| > | Maior que |
| < | Menor que |
| >= | Maior ou igual à |
| <= | Menor ou igual à |
| << | Encadeamento de strings |



Exemplo de programação

```
IF R10>=100 GOTOF DEST
```

Ou

```
R11=R10>=100
```

```
IF R11 GOTOF DEST
```

O resultado da comparação R10>=100 é memorizado em R11.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Operadores lógicos

Operadores lógicos são utilizados para combinar valores binários.

AND, OR, NOT e XOR podem, de forma genérica serem utilizados somente para variáveis tipo BOOL. Entretanto, podem também serem utilizados com os tipos de dados CHAR, INT e REAL através de conversões implícitas de tipo.

Espaços devem ser inseridos entre os operadores booleanos.

Nas operações lógicas (booleanas), a seguinte definição é aplicada aos tipos de dados BOOL, CHAR, INT e REAL:

Valor 0 equivale a FALSE

Valores diferentes de 0 equivalem à TRUE



Significado das operações lógicas

| | |
|-----|--------------|
| AND | AND |
| OR | OR |
| NOT | NOT |
| XOR | OR exclusivo |

Parênteses podem ser utilizados em expressões aritméticas para definir a ordem de execução para todos os operadores e com isto conduzi-las às prioridades necessárias.



Exemplo de programação

```
IF (R10<50) AND ($AA_IM[X]>=17.5) GOTOF DEST
```

```
IF NOT R10 GOTOB START
```

NOT se refere somente à um operando.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Operações lógicas bit a bit**

Operações lógicas bit a bit podem também serem executadas em variáveis do tipo CHAR e INT. A conversão do tipo é feita automaticamente.

**Significado das operações lógicas bit a bit**

| | |
|-------|---------------------|
| B_AND | AND bit |
| B_OR | OR bit |
| B_NOT | NOT bit |
| B_XOR | OR exclusivo do bit |



A operação B_NOT se refere somente à um operando.

**Exemplo de programação**

```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTOF ACT_PLANE
```

1.8 Prioridade dos operadores



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.8 Prioridade dos operadores



Prioridade dos operadores

A operador é atribuída uma prioridade. Quando uma expressão é executada, os operadores de maior prioridade são executados inicialmente. Quando os operadores possuírem a mesma prioridade, são executados da esquerda para a direita.

Parênteses podem ser utilizados em expressões aritméticas para definir a ordem de execução das operações.

Sequência de operações (maior para menor)

| | | |
|-----|----------------------|--|
| 1. | NOT, B_NOT | Inversão, inversão de bit |
| 2. | *, /, DIV, MOD | Multiplicação, divisão |
| 3. | +, - | Adição, subtração |
| 4. | B_AND | AND bit |
| 5. | B_XOR | OR exclusivo de bit |
| 6. | B_OR | OR bit |
| 7. | AND | AND |
| 8. | XOR | OR exclusivo |
| 9. | OR | OR |
| 10. | << | Encadeamento de strings, resultado tipo STRING |
| 11. | ==, <>, >, <, >=, <= | Comparações |

Exemplo para uma instrução IF:

If (otto==10) and (anna==20) gotof end



O operador de encadeamento ":" para frames não pode ser combinado com outros operadores em uma expressão. Portanto não existe nível de prioridade para este operador.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

1.9 Possibilidades de conversão



Tipos de conversão na carga

O valor da constante numérica, variável ou expressão deve ser compatível com o tipo de variável. Nestes casos, o tipo sofre uma conversão automática ao ser carregado.

Conversões possíveis

| de | REAL | INT | BOOL | CHAR | STRING | AXIS | FRAME |
|-----------|------|------|-------------------|-------------------|--------|------|-------|
| para REAL | Sim | sim* | sim ¹⁾ | sim* | – | – | – |
| INT | Sim | sim | sim ¹⁾ | sim ²⁾ | – | – | – |
| BOOL | sim | sim | sim | sim | sim | – | – |
| CHAR | sim | sim | sim ¹⁾ | sim | sim | – | – |
| STRING | – | – | sim ⁴⁾ | sim ³⁾ | sim | – | – |
| AXIS | – | – | – | – | – | sim | – |
| FRAME | – | – | – | – | – | – | sim |

* nas conversões de REAL para INT, uma fração $>+0.5$ é arredondada para cima, caso contrário para baixo (mesmo efeito da instrução ROUND)

¹⁾ Valores $<> 0$ são TRUE, valores $== 0$ são FALSE

²⁾ Caso o valor esteja na faixa permitida

³⁾ Caso somente 1 carácter

⁴⁾ Comprimento de string 0 = FALSE, caso contrário TRUE



Caso um valor seja maior que a faixa permitida de conversão, uma mensagem será exibida.



Informações adicionais

Caso existam vários tipos em uma expressão, a conversão é realizada automaticamente.

1.10 Operações com string



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.10 Operações com string



Generalidades

Adicionalmente às operações de “carga” e “comparação” descritas nesta seção, as seguintes manipulações de strings são possíveis:



Explicação

Conversão de tipo para STRING:

| | |
|---------------------------------------|---------------------------|
| STRING_ERG = <<bel._Typ ¹⁾ | Tipo do resultado: STRING |
|---------------------------------------|---------------------------|

| | |
|------------------------------|---------------------------|
| STRING_ERG = AXSTRING (AXIS) | Tipo do resultado: STRING |
|------------------------------|---------------------------|

Conversão de tipo de STRING:

| | |
|------------------------------|-------------------------|
| BOOL_ERG = ISNUMBER (STRING) | Tipo do resultado: BOOL |
|------------------------------|-------------------------|

| | |
|----------------------------|-------------------------|
| REAL_ERG = NUMBER (STRING) | Tipo do resultado: REAL |
|----------------------------|-------------------------|

| | |
|----------------------------|-------------------------|
| AXIS_ERG = AXNAME (STRING) | Tipo do resultado: AXIS |
|----------------------------|-------------------------|

Encadeamento de strings:

| | |
|--|---------------------------|
| bel._Typ ¹⁾ << bel._Typ ¹⁾ | Tipo do resultado: STRING |
|--|---------------------------|

Conversão maiúsculas/minúsculas:

| | |
|-------------------------------|---------------------------|
| STRING_ERG = TOUPPER (STRING) | Tipo do resultado: STRING |
|-------------------------------|---------------------------|

| | |
|-------------------------------|---------------------------|
| STRING_ERG = TOLOWER (STRING) | Tipo do resultado: STRING |
|-------------------------------|---------------------------|

Comprimento de string:

| | |
|---------------------------|------------------------|
| INT_ERG = STRLEN (STRING) | Tipo do resultado: INT |
|---------------------------|------------------------|

Pesquisa por caracter/string em string:

| | |
|--------------------------------|------------------------|
| INT_ERG = INDEX (STRING, CHAR) | Tipo do resultado: INT |
|--------------------------------|------------------------|

| | |
|---------------------------------|------------------------|
| INT_ERG = RINDEX (STRING, CHAR) | Tipo do resultado: INT |
|---------------------------------|------------------------|

| | |
|-----------------------------------|------------------------|
| INT_ERG = MINDEX (STRING, STRING) | Tipo do resultado: INT |
|-----------------------------------|------------------------|

| | |
|----------------------------------|------------------------|
| INT_ERG = MATCH (STRING, STRING) | Tipo do resultado: INT |
|----------------------------------|------------------------|

Seleção de substring:

| | |
|-----------------------------------|------------------------|
| STRING_ERG = SUBSTR (STRING, INT) | Tipo do resultado: INT |
|-----------------------------------|------------------------|

| | |
|--|------------------------|
| STRING_ERG = SUBSTR (STRING, INT, INT) | Tipo do resultado: INT |
|--|------------------------|

Seleção de um caracter:

| | |
|----------------------------|-------------------------|
| CHAR_ERG = STRINGVAR [IDX] | Tipo do resultado: CHAR |
|----------------------------|-------------------------|

| | |
|--|-------------------------|
| CHAR_ERG = STRINGFIELD [IDX_FIELD, IDX_CHAR] | Tipo do resultado: CHAR |
|--|-------------------------|

¹⁾ "bel._Typ" está disponível para os tipos INT, REAL, CHAR, STRING e BOOL.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Significado especial do caracter 0

O caracter 0 é interpretado internamente como caracter identificador de fim de string.

Caso um caracter seja trocado pelo caracter 0, a string será reduzida.

Exemplo:

```
DEF STRING[20] STRG = "Eixo . parado"
STRG[6] = "X"
```

```
MSG(STRG)
```

```
STRG[6] = 0
```

```
MSG(STRG)
```

```
;fornece a mensagem "Eixo X
parado" "
```

```
;fornece a mensagem "Eixo"
```

1.10.1 Conversão de tipo



A conversão de tipo permite que variáveis de diferentes tipos sejam utilizadas como parte integrante de mensagens (MSG).

Conversão para STRING

É o resultado do uso do operador << quando utilizado com variáveis de tipo INT, REAL, CHAR e BOOL (vide "Encadeamento de strings").

Um valor INT é convertido para uma forma legível.

Até 10 casas depois da vírgula estão disponíveis para valores tipo REAL.

Variáveis tipo AXIS podem ser convertidas para STRING através da instrução AXSTRING.

Variáveis FRAME não podem ser convertidas.

Exemplo:

```
MSG("Position:" << $AA_IM[X])
```

1.10 Operações com string



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Conversão de STRING

A função NUMBE converte STRING para REAL.

Caso ISNUMBER retorne um valor FALSE, um alarme será emitido quando NUMBER for chamada com o mesmo parâmetro.

Uma string pode ser convertida em dados tipo AXIS, através da função AXNAME. Será emitido um alarme caso a sequência de caracteres não combine com os eixos configurados.

Sintaxe

| | |
|------------------------------|---------------------------|
| BOOL_ERG = ISNUMBER (STRING) | Tipo de resultado: BOOL |
| REAL_ERG = NUMBER (STRING) | Tipo de resultado: REAL |
| STRING_ERG = AXSTRING (AXIS) | Tipo de resultado: STRING |
| AXIS_ERG = AXNAME (STRING) | Tipo de resultado: AXIS |

Significado:

ISNUMBER (STRING) retorna TRUE caso a sequência de caracteres represente um número REAL válido. Com isto é possível checar quando a string (sequência de caracteres) pode ser convertida em um número válido.

NUMBER (STRING) retorna o valor representado pela string como um número REAL.

AXSTRING (AXIS) fornece o nome do eixo como uma sequência de caracteres.

AXNAME (STRING) converte a sequência de caracteres especificada em um identificador de eixo.

Exemplos

| | |
|---------------------------------------|----------------------------|
| DEF BOOL BOOL_ERG | |
| DEF REAL REAL_ERG | |
| DEF AXIS AXIS_ERG | |
| DEF STRING[32] STRING_ERG | |
| BOOL_ERG = ISNUMBER ("1234.9876Ex-7") | ;BOOL_ERG == TRUE |
| BOOL_ERG = ISNUMBER ("1234XYZ") | ;BOOL_ERG == FALSE |
| REAL_ERG = NUMBER ("1234.9876Ex-7") | ;REAL_ERG == 1234.9876Ex-7 |
| STRING_ERG = AXSTRING(X) | ;STRING_ERG == "X" |
| AXIS_ERG = AXNAME("X") | ;AXIS_ERG == X |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

1.10.2 Encadeando seqüências de caracteres



Esta função torna possível compilar uma string com comentos individuais. A função de encadeamento é realizada através do operador:<<. Este operador possui STRING como tipo de dado fonte para todas as combinações dos tipos básicos CHAR, BOOL, INT, REAL e STRING. Quaisquer conversões que possam vir a ser necessárias são feitas de acordo com as regras existentes. Os tipos de dados FRAME e AXIS não podem ser utilizados com este operador.

Sintaxe:

```
bel._Typ << bel._Typ
```

Tipo de resultado: STRING

Significado:

As strings especificadas (a conversão de outros tipos é feita de forma implícita) são encadeadas.

Este operador encontra-se também disponível com variantes. Desta forma é possível a conversão de tipos explícitos para STING (não para FRAME e AXIS)

Sintaxe

```
<< bel._Typ
```

Tipo de resultado: STRING

Explicação:

O tipo especificado é convertido em STRING.

Por exemplo, esta função pode ser utilizada para compilar uma mensagem ou um comando a partir de listas de texto e para inserir parâmetros (como nome de módulos):

```
MSG ( STRG_TAB [ LOAD_IDX ] <<MODULE_NAME )
```

1.10 Operações com string



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



O resultado intermediário do encadeamento de uma string não deve exceder o comprimento máximo da string.



Exemplo de programação

```
DEF INT IDX = 2
DEF REAL VALOR = 9.654
DEF STRING[20]STRG = "INDICE:2"
IF STRG == "Indice:" <<IDX GOTOF
NO_MSG
MSG ("Indice:" <<IDX <<"/Valor:" ;Será exibido: "Indice: 2/valor: 9.654"
<<VALOR)
NO_MSG:
```

1.10.3 Conversão de caracteres maiúsculas/minúsculas



Esta função pode ser utilizada para converter todas as letras de uma string para um único tamanho de caracteres.

Sintaxe:

| | | |
|----------------------|----------|---------------------------|
| STRING_ERG = TOUPPER | (STRING) | Tipo de resultado: STRING |
| STRING_ERG = TOLOWER | (STRING) | Tipo de resultado: STRING |

Significado:

Todas as letras minúsculas são convertidas tanto para letras maiúsculas ou para minúsculas.

Exemplo:

Como o usuário pode carregar textos a partir da MMC, é possível arquivar os textos de forma uniforme (isto é, com caracteres maiúsculos ou minúsculos):

```
DEF STRING [29] STRG
...
IF "LEARN.CNC" == TOUPPER (STRG) GOTOF LOAD_LEARN
```

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

1.10.4 Comprimento da string



Permite determinar a quantidade de caracteres de um string.

Sintaxe:

`INT_ERG = STRLEN (STRING)`

Tipo de resultado: INT

Significado:

A quantidade de caracteres é determinada – a partir do início da string – desde que existam caracteres.

Exemplo:

Esta função pode ser utilizada para determinar o fim da string, por exemplo:

```
IF (STRLEN (MODULE_NAME) > 10) GOTOF ERROR
```

1.10.5 Procura por caracter/seqüência de caracteres na string



Esta função pode ser utilizada para procurar um caracter ou um conjunto de caracteres em uma string. A função retorna a posição específica onde o caracter/conjunto de caracteres encontra-se posicionado na string onde foi feita a pesquisa.

`INT_ERG = INDEX (STRING, CHAR)` Tipo de resultado: INT

`INT_ERG = RINDEX (STRING, CHAR)` Tipo de resultado: INT

`INT_ERG = MINDEX (STRING, STRING)` Tipo de resultado: INT

`INT_ERG = MATCH (STRING, STRING)` Tipo de resultado: INT

Significado:

Funções de pesquisa: Retornam a posição (primeiro parâmetro) do caracter ou conjunto de caracteres na string onde realizada com sucesso a pesquisa. Caso o caracter/caracteres não tenham sido encontrados, será retornado o valor -1. O primeiro caracter da string é o caracter 0.

1.10 Operações com string



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

| | |
|--------|---|
| INDEX | Procura a partir do início do primeiro parâmetro pelo caracter especificado no segundo parâmetro. |
| RINDEX | Procura a partir do final do primeiro parâmetro o caracter especificado no segundo parâmetro. |
| MINDEX | Corresponde à função INDEX, com a diferença que uma lista de caracteres é transferida (como string). O índice do primeiro caracter encontrado nesta lista será o retorno da função. |
| MATCH | Procura por uma string dentro de uma string. |

Entretanto, as strings podem ser quebradas de acordo com certos critérios, ou seja, em posições com espaços ou separadores de diretórios ("/").



Exemplo de programação

Exemplo de como quebrar um valor digitado no nome do arquivo e do diretório.

| | |
|---|---|
| DEF INT PATHIDX, PROGIDX | |
| DEF STRING[26] INPUT | |
| DEF INT LISTIDX | |
| INPUT = "/_N_MPF_DIR/_N_EXECUTE_MPF" | |
| LISTIDX = MINDEX (INPUT, "M,N,O,P") + 1 | 3 é o valor de retorno em LISTIDX, pois "N" é o primeiro caracter do parâmetro INPUT (início da lista de seleção). |
| PATHIDX = INDEX (INPUT, "/") +1 | ; o valor de PATHIDX é 1 |
| PROGIDX = RINDEX (INPUT, "/") +1 | ; o valor de PROGIDX é 12 |
| | ; Utilizando a função SUBSTR, explicada na próxima seção, a variável INPUT pode ser quebrada em "diretório e "arquivo": |
| VARIABLE = SUBSTR (INPUT, PATHIDX, PROGIDX-PATHIDX-1) | Fornece "_N_MPF_DIR" |
| VARIABLE = SUBSTR (INPUT, PROGIDX) | Fornece "_N_EXECUTE_MPF" |

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

1.10.6 Seleção de uma substring



Esta função torna possível separar uma string em substrings. Para este propósito, o índice do primeiro carácter e o comprimento desejado da substring (se aplicável) são especificados. Caso não exista a informação de comprimento, então o comprimento será todo o restante da string.

| | |
|---------------------|------------------------|
| STRING_ERG = SUBSTR | Tipo de resultado: INT |
|---------------------|------------------------|

| | | |
|---------------------|--------------------|------------------------|
| STRING_ERG = SUBSTR | (STRING, INT, INT) | Tipo de resultado: INT |
|---------------------|--------------------|------------------------|

Significado:

No primeiro caso, a substring definida a partir da posição determinada pelo primeiro parâmetro até o final da string principal será o retorno da função.

No segundo caso, o resultado, a string resultante é limitada ao comprimento máximo definido pelo terceiro parâmetro.

Caso a posição inicial esteja definida após o término da string, o retorno da função será uma string vazia ("").

Caso a posição inicial ou o comprimento sejam negativos, um alarme será emitido.

Exemplo:

```
DEF STRING [29] ERG
```

```
ERG = SUBSTR ("ACKNOWLEDGMENT: 10 to 99", ; Portanto ERG == "10"
10, 2)
```

1.10 Operações com string



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.10.7 Selecionando um único caracter



Caracteres individuais de uma string podem ser selecionados através desta função. Isto se aplica tanto para operações de leitura quanto de escrita.

Sintaxe:

| | |
|---|-------------------------|
| CHAR_ERG = STRINGVAR [IDX] | Tipo de resultado: CHAR |
| CHAR_ERG = STRINGARRAY [IDX_ARRAY, IDX_CHAR] | Tipo de resultado: CHAR |

Significado:

O caracter localizado na posição específica dentro da string será lido/escrito. Caso o parâmetro de posição seja negativo ou maior que o comprimento da string, um alarme será emitido.

Exemplo das mensagens:

Inserção de um identificador de eixo em uma string pré-definida.

```
DEF STRING [50] MESSAGE = "Eixo n
atingiu a posição"
MESSAGE [6] = "X"
MSG (MESSAGE)
```

;Retorna a mensagem "Eixo X atingiu a posição"

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Somente é possível acessar caracteres individuais em variáveis que tenham sido definidas pelo usuário (LUD, GUD e PUD).

Adicionalmente, este modo de acesso de dados em chamada de subrotinas pode somente ser utilizado para parâmetros do tipo “chamada pelo valor”.

Exemplos:

Acessando um único caracter em um dado de sistema, dado de máquina, ...

```
DEF STRING [50] STRG
```

```
DEF CHAR ACKNOWLEDGMENT
```

```
...
```

```
STRG = $P_MMCA
```

```
ACKNOWLEDGMENT = STRG [0]
```

```
;Avaliação do componente  
acknowledgment
```

Acessando um caracter com parâmetro “chamada por referência”:

```
DEF STRING [50] STRG
```

```
DEF CHAR CHR1
```

```
EXTERN UP_CALL (VAR CHAR1)
```

```
;Parâmetro “Chamada por referência”!
```

```
...
```

```
CHR = STRG [5]
```

```
UP_CALL (CHR1)
```

```
;Chamada por referência
```

```
STRG [5] = CHR1
```

1.11 Instrução CASE840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

1.11 Instrução CASE**Programação**

CASE (expressão) OF constante1 GOTOF LABEL1 ... DEFAULT GOTOF LABELn

CASE (expressão) OF constante1 GOTOB LABEL1 ... DEFAULT GOTOB LABELn

**Explicação dos comandos**

| | |
|-----------|---|
| CASE | Instrução de salto |
| GOTOF | Instrução de salto com destino para frente (no sentido do fim do programa) |
| GOTOB | Instrução de salto com destino para trás (no sentido do início do programa) |
| LABEL | Destino (label dentro do programa); |
| LABEL: | O nome do destino do salto é seguido de “:” |
| Expressão | Expressão aritmética |
| Constante | Constante do tipo INT |
| DEFAULT | Destino caso nenhum destino seja encontrado. |

**Função**

A instrução CASE habilita a execução de várias ramificações de acordo com um valor de tipo INT.

**Seqüência**

O programa salta para o ponto especificado como destino, dependendo do valor da constante existente na instrução CASE.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Em casos onde a constante não combina com nenhum dos valores pré definidos, a instrução **DEFAULT** pode ser utilizada para definir o destino do salto.

Caso a instrução não seja programada, o destino do salto será a instrução seguinte à **CASE**.



Exemplo de programação

Exemplo 1

```
CASE(expressão) OF 1 GOTOF LABEL1 2 GOTOF LABEL2 ... DEFAULT GOTOF LABELn
```

"1" e "2" são as constantes possíveis.

Caso o valor da expressão = 1 (INT constante), saltar para o bloco com LABEL1

Caso o valor da expressão = 2 (INT constante), saltar para o bloco com LABEL2

...

Caso contrário saltar para o bloco LABELn

Exemplo 2

```
DEF INT VAR1 VAR2 VAR3
```

```
CASE(VAR1+VAR2-VAR3) OF 7 GOTOF LABEL1 9 GOTOF LABEL2 DEFAULT GOTOF LABEL3
```

```
LABEL1: G0 X1 Y1
```

```
LABEL2: G0 X2 Y2
```

```
LABEL3: G0 X3 Y3
```

1.12 Estruturas de controle



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.12 Estruturas de controle



Explicação

| | |
|-----------------|------------------------------|
| IF-ELSE-IFENDIF | Seleção entre 2 alternativas |
|-----------------|------------------------------|

| | |
|--------------|--------------|
| LOOP-ENDLOOP | Loop sem fim |
|--------------|--------------|

| | |
|------------|-------------------|
| FOR-ENDFOR | Loop com contador |
|------------|-------------------|

| | |
|----------------|--------------------------------|
| WHILE-ENDWHILE | Loop com coordenação no início |
|----------------|--------------------------------|

| | |
|--------------|-------------------------------|
| REPEAT-UNTIL | Loop com coordenação no final |
|--------------|-------------------------------|



Função

O controle processa os blocos NC na seqüência programada.

Adicionalmente às ramificações de programa descritas neste capítulo, estes comandos podem ser utilizados para definir alternativas de programação em loops.

Estes comandos habilitam o usuário a produzir programas bem estruturados e de fácil compreensão.



Seqüência

1. IF-ELSE-ENDIF

Um bloco IF-ELSE-ENDIF é utilizado para seleção de uma alternativa entre duas.

IF (expressão)

Blocos NC

ELSE

Blocos NC

ENDIF

Caso o valor da expressão seja TRUE, ou seja, a condição satisfeita, o próximo bloco de programa será executado. Caso a condição não seja satisfeita, então a ramificação especificada em ELSE será executada.

ELSE pode ser omitido.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2. LOOP sem fim

Loops sem fim são utilizados em programas sem fim. No fim da do programa sempre existe uma instrução de retorno para o início.

LOOP

Blocos NC

ENDLOOP

3. Contador de loop FOR

O loop FOR é utilizado caso seja necessário repetir as operações um número específico de vezes. Neste caso, a variável de contagem será incrementada de um valor inicial até um valor final. O valor inicial deve ser menor que o final. A variável deve ser tipo INT.

FOR Variável = valor inicial TO valor final

Blocos NC

ENDFOR

4. Programa em loop com condições iniciais e finais WHILE

O loop de programa WHILE é executado enquanto a condição existir.

WHILE expressão

Blocos NC

ENDWHILE

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

5. Loop de programa com condições no fim do loop REPEAT

O loop REPEAT é executado uma vez e repetido continuamente até que a condição seja satisfeita.

REPEAT

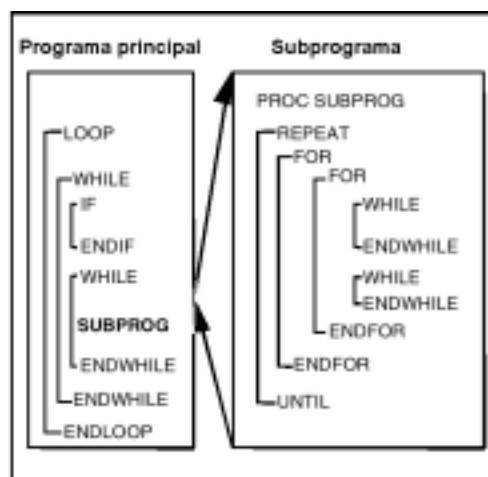
Blocos NC

UNTIL (expressão)



Nível de encadeamento

As estruturas de checagem são aplicadas de forma local dentro dos programas. Um nível de encadeamento de até 8 estruturas de checagem podem ser utilizadas em cada nível de sub programa.



Tempo de processamento

No modo interpretador (condição standard de fornecimento), é possível reduzir os tempos de processamento de forma mais efetiva através do uso de ramificações do que com estruturas de checagem.

Não há diferenças entre ramificações de programas e estruturas de checagem em ciclos pré compilados.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Condições suplementares

Blocos com elementos de checagem de estrutura não podem ser suprimidos. Labels não devem ser utilizados em blocos deste tipo.

As estruturas de checagem são processadas de forma interpretativa. Quando o fim de um loop é detectado, uma busca pelo início do loop é realizada, permitindo à estrutura de checagem retornar ao processo.

Por esta razão, a estrutura de blocos de um programa não é checada completamente no modo interpretador.

Não é geralmente aconselhável misturar estruturas de checagem com ramificações de programa.

Uma verificação pode ser realizada para garantir que as estruturas de checagem estão ramificadas de forma correta durante o pré processamento.



Estruturas de checagem podem ser incluídas somente na parte relativa às instruções do programa. As definições contidas nos cabeçalhos não devem ser executadas de forma condicional ou repetida.

Não é possível sobrepor macros por instruções nas estruturas de checagem ou em destinos de ramificações. Não são realizadas checagens durante a definição de macros.

1.12 Estruturas de controle840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Exemplo de programação****1. Programa sem fim**

```

%_N_LOOP_MPF
LOOP
    IF NOT $P_SEARCH                                ;Sem pesquisa de bloco
        G01 G90 X0 Z10 F1000
        WHILE $AA_IM[X] <= 100
            G1 G91 X10 F500                        ;Modelo de furação
            Z-5 F100
            Z5
        ENDWHILE
        Z10
    ELSE                                              ;Com pesquisa de bloco
        MSG("Sem furação durante a pesquisa de blocos")
    ENDIF
    $A_OUT[1]=1                                      ;Próxima furação
    G4 F2
ENDLOOP
M30

```

2. Produção de uma quantidade fixa de peças

```

%_N_WKPCCOUNT_MPF

DEF INT WKPCCOUNT
FOR WKPCCOUNT = 0 TO 100
    G01 ...
ENDFOR
M30

```




840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

1.13 Coordenação de programas

Canais

Um canal pode processar seu programa de forma independente dos outros canais. Ele pode controlar temporariamente os eixos e fusos a ele definidos através do programa. Dois ou mais canais podem ser definidos para o controle durante a colocação em funcionamento.

Coordenação do programa

Caso vários canais estejam envolvidos na usinagem de uma peça, pode ser necessário sincronizar os programas. Instruções especiais (comandos) encontram-se disponíveis para a coordenação de programas. Cada instrução deve ser programada separadamente em um bloco.

1.13 Coordenação de programas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Instruções para coordenação de programas

• Especificação absoluta de diretório

INIT (n, " /_HUGO_DIR/_N_name_MPF") ou

INIT (n, " /_N_MPF_DIR/_N_name_MPF")

Exemplo:

INIT(2, " /_N_WCS_DIR/_DRESSING_MPF")

G01 F0.1

START

INIT (2, " /_N_WCS_DIR/_N_UNDER_1_SPF")

A especificação absoluta do diretório é realizada de acordo com as seguintes regras:

- *Diretório atual/_N_name_MPF*
"diretório atual" especifica o diretório de peças selecionado ou o diretório standard */_N_MPF_DIR*.
- Seleciona um programa particular para a execução em um canal específico:
n: Número do canal, valor conforme a configuração do controle
- Nome completo do programa

SW 3 e anteriores:

Ao menos um bloco executável deve ser programado entre o comando **init** (sem sincronismo) e o **NC start**.

Para chamada de subrotinas, "_SPF" deve ser acrescentadas à especificação do diretório.

• Especificação relativa do diretório

Exemplo:

INIT(2, "DRESS")

INIT(3, "UNDER_1_SPF")

As mesmas regras são aplicadas para a definição relativa dos diretórios na chamada dos programas.

Com as chamadas de subrotinas "_SPF" deve ser acrescentada ao nome do programa.

START (n,n)

Inicia os programas selecionados em outros canais.

n,n: Número do canal: valor depende da configuração do controle.

WAITM (Marca Nr.,n,n,...)

Define o número da marca "Marca Nr." No mesmo canal. O bloco anterior é encerrado com parada exata. Espera até que as marcas com o mesmo "Marca Nr." nos canais especificados "n" (o canal atual não deve ser especificado). A marca é apagada após o sincronismo. 10 marcas podem ser programadas simultaneamente em um canal.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

WAITMC(Marca Nr., n, n, ...)

Define a marca "Marca Nr." no mesmo canal . Uma parada exata é iniciada somente se os outros canais ainda não houverem atingido a marca. Espera até que as marcas com o mesmo "Marca Nr." nos canais especificados "n" (o canal atual não deve ser especificado). Quando a marca "Marca Nr" no canal específico for atingida, o movimento continua sem parada exata.

WAITE (n,n)

Espera o fim do programa nos canais especificados (não especifica o canal atual)

SETM(Marca Nr., Marca Nr., ...)

Define a marca "Marca Nr." No mesmo canal sem afetar o processamento atual. SETM() permanece válido após RESET e NC START. SETM() pode também ser programado independentemente em uma ação síncrona.

CLEARM(Marca Nr., Marca Nr., ...)

Apaga as marcas "Marca Nr." No mesmo canal sem afetar o processamento. Todas as marcas podem ser apagadas com CLEARM(). CLEARM (0) apaga a marca "0". CLEARM() permanece válido após RESET e NC START. CLEARM() pode também ser programado de forma independentemente de uma ação síncrona.



Nota

Todos os comandos mencionados acima devem ser programados em blocos separados.

Nomes dos canais

Os nomes dos canais devem ser convertidos em números através de variáveis (vide seção 10 "Variáveis e parâmetros aritméticos).



Reserve uma quantidade de números, de forma que não sejam alterados de forma accidental.

1.13 Coordenação de programas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Exemplo:

O canal de nome "MACHINE" deve conter o canal 1,

O canal de nome "LOADER" deve conter o canal de numero 2,

```
DEF INT MACHINE=1, LOADER=2
```

Os nomes das variáveis são os mesmos nomes dos canais.

Portanto, a instrução START será programada do seguinte modo:

```
START ( MACHINE )
```

Exemplo de coordenação de um programa

Canal 1:

```
%_N_MPF100_MPF
```

```
N10 INIT(2, "MPF200")
```

```
N11 START (2)
```

Execução do programa no canal 2

```
.
```

```
N80 WAITM(1,1,2)
```

Espera por WAIT marca 1 nos canais 1 e 2 e a execução continua no canal 1

```
.
```

```
N180 WAITM(2,1,2)
```

Espera por WAIT marca 2 nos canais 1 e 2 e a execução continua no canal 1

```
.
```

```
N200 WAITE(2)
```

Espera pelo fim do programa no canal 2

```
N201 M30
```

Fim do programa no canal 1

```
...
```

Canal 2:

```
%_N_MPF200_MPF
```

```
; $PATH=/_N_MPF_DIR
```

Execução do programa no canal 2

```
N70 WAITM(1,1,2)
```

Espera por WAIT marca 1 nos canais 1 e 2 e a execução continua no canal 2

```
.
```

```
N270 WAITM(2,1,2)
```

Espera por WAIT marca 2 nos canais 1 e 2 e a execução continua no canal 2

```
.
```

```
N400 M30
```

Fim do programa no canal 2

840D
NCU 571840D
NCU 572
NCU 573

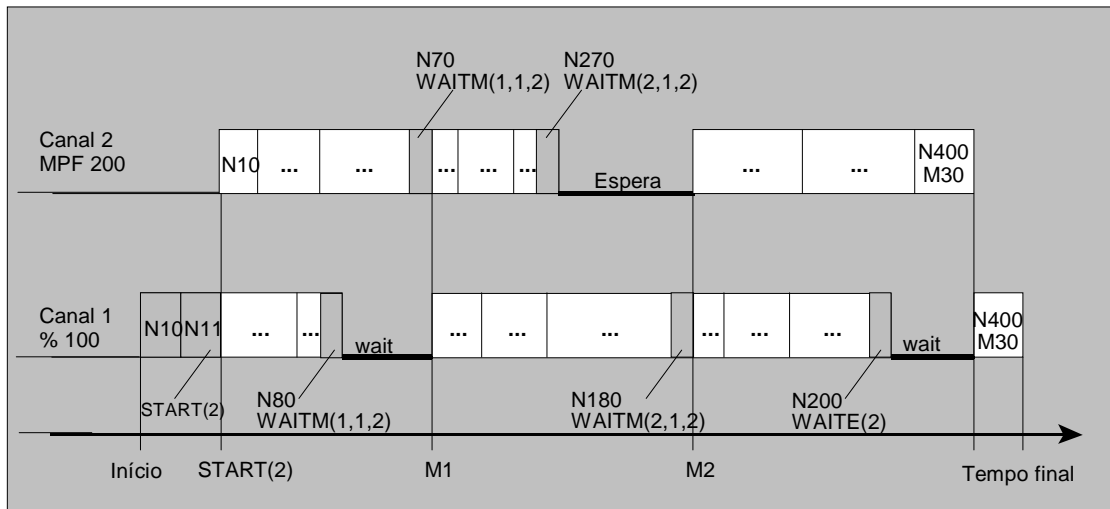
FM-NC



810D



840Di



Exemplo de programa para peça

```
N10 INIT(2, "/_N_WCS_DIR/_N_SHAFT1_WPD/_N_STOKREM1_MPF")
```

Exemplo do comando INIT com definição de diretório relativa

```
; Programa /_N_MPF_DIR/_N_MAIN_MPF encontra-se selecionado no canal 1
N10 INIT(2, "MYPROG") ; Seleciona o programa /_N_MPF_DIR/_N_MYPROG_MPF
                        no canal 2.
```

Informações adicionais

As variáveis que todos os canais podem acessar (variáveis globais NCK) podem ser utilizadas para a troca de dados entre os programas. Por outro lado, programas devem ser escritos separadamente para cada canal.



SW 3 e anteriores:

WAITE não deve ser processada imediatamente após o comando START. Caso seja programado, o fim do programa será interpretado mesmo antes do programa ter sido iniciado.

Solução: Programar um tempo de espera

Exemplo:

```
N30 START (2)
N31 G4 F0.01
N40 WAITE(2)
```

1.14 Rotina de interrupção840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

1.14 Rotina de interrupção**Programação**

```

SETINT(3) PRIO=1 NAME
SETINT(3) PRIO=1 LIFTFAST
SETINT(3) PRIO=1 NAME LIFTFAST
G... X... Y... ALF=...
DISABLE(3)
ENABLE(3)
CLRINT(3)

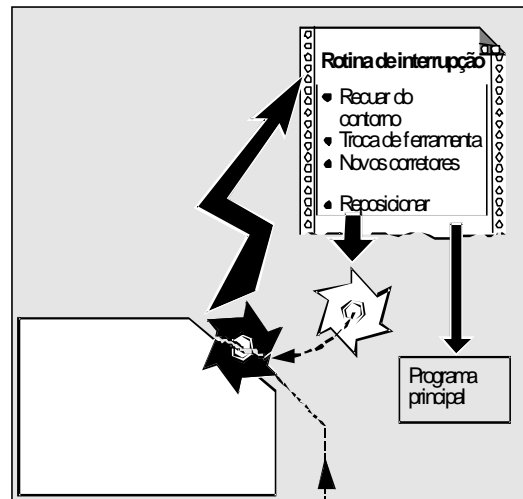
```

**Explicação dos comandos**

| | |
|------------|---|
| SETINT(n) | Inicia a rotina de interrupção caso a entrada n estiver habilitada, n(1...8) representa o número da entrada |
| PRIO=1 | Define a prioridade de 1 a 128 (1 tem a maior prioridade) |
| LIFTFAST | Recurso rápido do contorno |
| NOME | Nome do sub programa a ser executado |
| ALF=... | Direção programável de movimento (no bloco de movimento) |
| DISABLE(n) | Desativa a rotina de interrupção número n |
| ENABLE(n) | Reativa a rotina de interrupção número n |
| CLRINT(n) | Limpa as definições de rotinas de interrupção n |

**Função**

Exemplo: A ferramenta quebra durante a usinagem. Isto dispara um sinal que interrompe o processo de usinagem e simultaneamente inicia um sub programa. Este sub programa é chamado de rotina de interrupção. A rotina de interrupção contém todas as instruções que devem ser executadas neste caso. Quando uma rotina de interrupção foi finalizada e a máquina encontra-se pronta para continuar a operação, o controle retorna ao programa principal e continua a usinagem do ponto de interrupção – dependendo do comando REPOS.



Para informações adicionais de REPOS, vide o capítulo 9, Comportamento do movimento na trajetória, Reposicionando.

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Seqüência

Criar uma rotina de interrupção como um sub programa

A rotina de interrupção é identificada como um subprograma na definição.

Exemplo:

```
PROC LIFT_Z  
N10...  
N50 M17
```

O nome do programa LIFT_Z, seguido por blocos NC, é finalizado com M17 e retorna para o programa principal.



Nota:

As instruções SETINT podem ser programadas dentro de rotinas de interrupção e utilizadas para ativar rotinas adicionais de interrupção. São disparadas através de entradas.



Você encontrará maiores informações sobre o como criar subprogramas no capítulo 2.

Gravar posição de interrupção, SAVE

A rotina de interrupção pode ser identificada com SAVE em sua definição.

Exemplo:

```
PROC LIFT_Z SAVE  
N10...  
N50 M17
```

No fim de uma subrotina de interrupção as funções G modais são colocadas no mesmo estado que estavam antes da interrupção através do atributo SAVE.

Os deslocamentos de origem programados, o deslocamento básico e os deslocamentos ajustáveis (grupo de funções G 8 –modais) são restabelecidos. Caso o grupo de funções G 15 (avanço) tenha sido alterado, p.e. de G94 para G95, o valor apropriado

1.14 Rotina de interrupção



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

de F é também restabelecido.

A usinagem pode ser retomada posteriormente a partir do ponto de interrupção.

Definindo e iniciando uma rotina de interrupção, SETINT

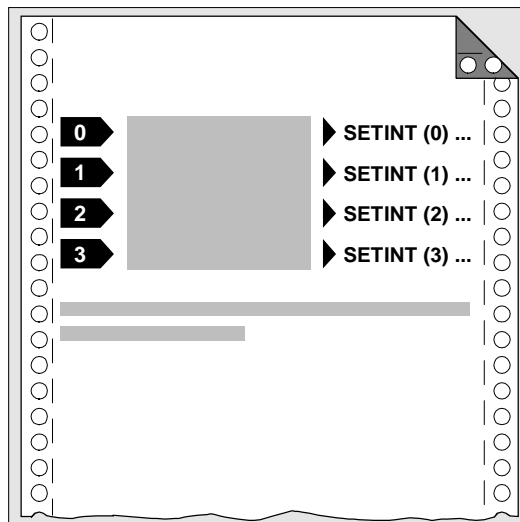
O controle possui oito sinais (entradas de 1 à 8) para interromper o programa e iniciar a subrotina de interrupção correspondente.

A relação entre a entrada e o programa é feita no programa principal.

Exemplo:

```
N10 SETINT(3) PRIO=1 LIFT_Z
```

Quando a entrada 3 for ativada, a rotina LIFT_Z será imediatamente iniciada.



Iniciando diversas rotinas de interrupção, definindo as prioridades , PRIO=

Caso várias instruções SETINT sejam programadas NC e vários sinais de ativação possam ocorrer simultaneamente, você deve atribuir prioridades para as rotinas de interrupção, de forma a determinar sua ordem de execução.

Níveis de prioridade PRIO de 1 à 128 encontram-se disponíveis, o nível 1 tem maior prioridade.

Exemplo:

```
N10 SETINT(3) PRIO=1 LIFT_Z
```

```
N20 SETINT(2) PRIO=2 LIFT_X
```

As rotinas são executadas sucessivamente conforme suas prioridades caso as entradas sejam ativadas ao mesmo tempo. Inicialmente a SETTING(3), e após concluída a SETTING(2). Caso novos sinais sejam recebidos durante a execução de uma rotina de interrupção, a rotina atual é também interrompida caso ativadas entradas com maior prioridade.

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Desativando/reactivando rotinas de interrupção, **DISABLE, ENABLE**

Você pode desativar rotinas de interrupção no programa NC com o comando **DISABLE(n)** e reativá-las com **ENABLE(n)** (n indica o número da entrada).



A relação entre entrada/rotina é mantida com **DISABLE** e restabelecida com **ENABLE**.

Redefinindo rotinas de interrupção

Caso uma nova subrotina será atribuída a uma entrada, a relação entre rotina e entrada anterior é automaticamente cancelada.

Exemplo:

```
N20 SETINT(3) PRIO=2 LIFT_Z
...
...
N120 SETINT(3) PRIO=1 LIFT_X
```

Apagando as definições, **CLRINT**

As definições podem ser canceladas através do comando **CLRINT(n)**.

Exemplo:

```
N20 SETINT(3) PRIO=2 LIFT_Z
N50 CLRINT(3)
```

A relação entre a entrada 3 e a rotina **LIFT_Z** é cancelada.

1.14 Rotina de interrupção



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Recuo rápido do contorno, LIFTFAST

Quando a entrada for comutada, LIFTFAST recua a ferramenta rapidamente do contorno da peça.

Caso a instrução SETINT inclua uma rotina de interrupção como a LIFTFAST, esta será executada antes da rotina de interrupção.

Exemplo:

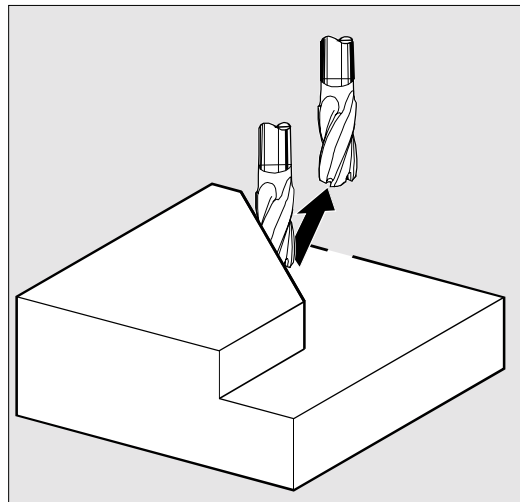
```
N10 SETINT(2) PRIO=1 LIFTFAST
```

Ou

```
N30 SETINT(2) PRIO=1 LIFT_Z LIFTFAST
```

Em ambos os casos, LIFTFAST será executada quando a entrada 2 for habilitada.

- Com N10, a execução será interrompida com o alarme 16010 (pois não existe subprograma assíncrono, ASUP, especificado).
- O programa síncrono "LIFT-Z" é executado em N30.



Seqüência de movimentos com recuo rápido

A distância de recuo do contorno com LIFTFAST pode ser definida através de dados de máquina.

Direção de movimento programável, ALF=...

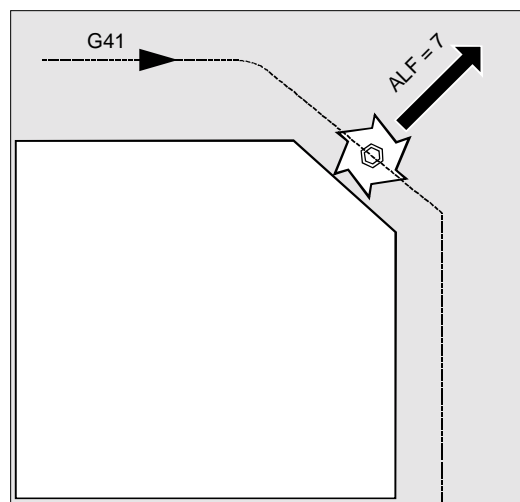
Você pode especificar a direção com que a ferramenta será recuada no programa NC.

As direções possíveis são gravadas em códigos especiais através de números e podem ser chamadas através destes números.

Exemplo:

```
N10 SETINT(2) PRIO=1 LIFT_Z LIFTFAST  
ALF=7
```

Os movimentos da ferramenta – ativados com G41 (direção de usinagem à esquerda do contorno) – é recuado de forma perpendicular, conforme ilustrado ao lado.



840D
NCU 571840D
NCU 572
NCU 573

810D

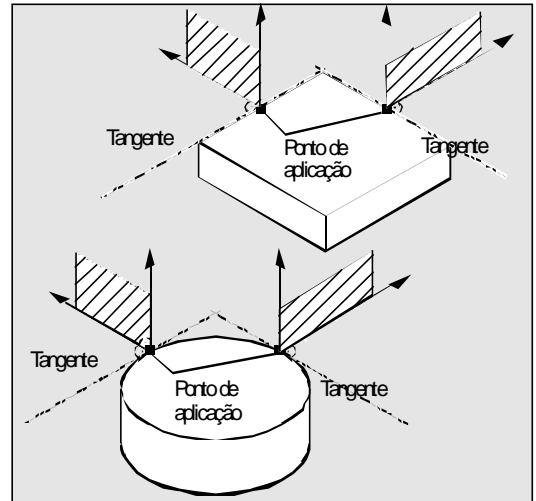


840Di

Plano de referência para descrição das direções de movimentos

No ponto de aplicação da ferramenta ao contorno programado, a ferramenta é relacionada a um plano que é utilizado como referência para especificar o movimento de recuo com o número de código correspondente.

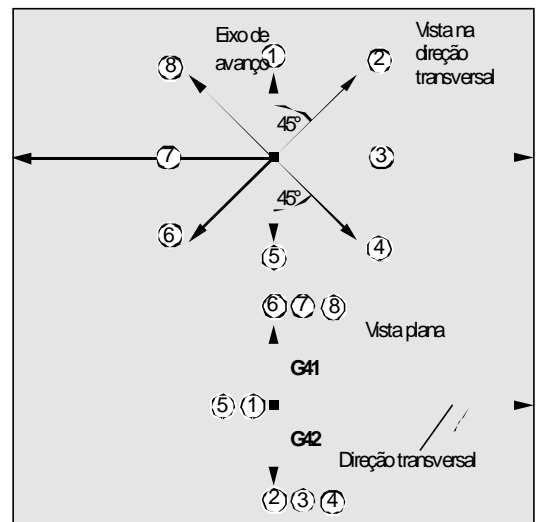
O plano referência é derivado do eixo longitudinal da ferramenta (direção entrada) e um vetor posicionado perpendicularmente a este eixo e perpendicular à tangente no ponto de aplicação da ferramenta.



Relação do número código com as direções de movimentos.

Os números código em relação ao plano referência são mostrados no diagrama à direita.

ALF=0 desativa a função recuo rápido.



Favor observar:

Os seguintes códigos **não** devem ser utilizados quando a correção de raio da ferramenta estiver ativa:

Códigos 2, 3, 4 com G41

Códigos 6, 7, 8 com G42.

Nestes casos, a ferramenta irá se aproximar do contorno e colidir com a peça.

1.14 Rotina de interrupção



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Movimento de recuo na SW4.3 em diante

A direção do movimento de recuo é programada através do código G **LFTXT** ou **LFWP** com a variável **ALF**.

- **LFTXT**

O plano do movimento de recuo é determinado a partir da tangente da trajetória e a direção da ferramenta. Este código G (estado de fornecimento) é utilizado para programar o comportamento para o recuo rápido.

- **LFWP**

O plano para o movimento de recuo é o plano de trabalho ativo selecionado através dos códigos G G17, G18 ou G19. A direção do movimento de recuo não depende da tangente da trajetória. Desta forma é possível a programação de um recuo rápido paralelo ao eixo.

No plano do movimento de recuo, **ALF** é utilizado para programar a direção em passos discretos de 45 graus como no caso anterior. ALF=1 define o recuo **LFTXT** na direção da ferramenta.

Com **LFWP** a direção do plano de trabalho é definida de acordo com:

- **G17:** X/Y plano ALF=1 Recuo na direção X
 ALF=3 Recuo na direção Y
- **G18:** Z/X plano ALF=1 Recuo na direção Z
 ALF=3 Recuo na direção X
- **G19:** Y/Z plano ALF=1 Recuo na direção Y
 ALF=3 Recuo na direção Z

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Exemplo de programação

Neste exemplo, uma ferramenta quebrada deve ser automaticamente trocada por outra. A usinagem deve continuar com a nova ferramenta.

Programa principal

```
N10 SETINT(1) PRIO=1 C_CHANGE ->
-> LIFTFAST
```

Quando a entrada 1 for habilitada, a ferramenta é automaticamente recuada do contorno através de LIFTFAST (código 7 para correção de raio G41). A rotina de interrupção C_CHANGE é executada posteriormente.

```
N20 G0 Z100 G17 T1 ALF=7 D1
```

```
N30 G0 X-5 Y-22 Z2 M3 S300
```

```
N40 Z-7
```

```
N50 G41 G1 X16 Y16 F200
```

```
N60 Y35
```

```
N70 X53 Y65
```

```
N90 X71.5 Y16
```

```
N100 X16
```

```
N110 G40 G0 Z100 M30
```

Subprograma

```
PROC C_CHANGE SAVE
```

Subprograma com memorização do estado atual de operação

```
N10 G0 Z100 M5
```

Posição de troca de ferramenta, parada do fuso

```
N20 T11 M6 D1 G41
```

Troca de ferramenta

```
N30 REPOS L RMB M3
```

Reposiciona e retorna ao programa principal

-> Programado em um único bloco.



Caso você não programe o comando REPOS, o eixo será posicionado no fim do bloco seguinte ao bloco no qual ocorreu a interrupção..

1.15 Transferência de eixo, transferência de fuso



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

1.15 Transferência de eixo, transferência de fuso



Explicação dos comandos

| | |
|--|--|
| RELEASE(nome do eixo, nome do eixo, ...) | Habilita o eixo |
| GET(nome do eixo, nome do eixo, ...) | Aceita o eixo |
| GETD (nome do eixo, nome do eixo, ...) | Aceita diretamente o eixo |
| Nome do eixo | Define o eixo no sistema: AX1, AX2, ... ou especifica o nome do eixo de usinagem |
| RELEASE(S1) | Habilita fusos S1, S2, ... |
| GET(S2) | Aceita fusos S1, S2, ... |
| GETD(S3) | Aceita diretamente fusos S1, S2, ... |



Função

Um ou mais eixos ou fusos podem ser sempre utilizados em um canal. Caso um eixo deva ser alternado entre dois canais (p.e. trocador de paletes) ele deve ser primeiramente habilitado no canal atual e então transferido para o outro canal. O eixo é transferido de canal para canal.



Seqüência

Pré condições para o transferência de eixo

- O eixo deve ser definido em todos os canais através de dados de máquina.
- O canal para o qual o eixo pertence após a ligação do comando é definido através de dados de máquina.

1.15 Transferência de eixo, transferência de fuso



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Libera eixo: RELEASE

Durante a habilitação de um eixo, favor notar:

1. O eixo não deve fazer parte de uma transformação.
2. Todos os eixos envolvidos em alguma conexão de eixo (controle tangencial, movimento acoplado) devem estar habilitados.
3. Um eixo de posicionamento A coexistente não deve ser transferido.
4. Todos os eixos que acompanham um eixo tipo gantry serão transferidos com o eixo mestre.

Transferência de eixo: buscar eixo : GET

A transferência atual do eixo é executada com este comando. O canal no qual o comando foi programado fica totalmente responsável pelo eixo.

Efeitos de GET:

Eixo transferido com sincronismo:

Um eixo deve sempre ser sincronizado caso tenha sido transferido para outro canal ou para o PLC e não tenha sido sincronizado com "WAIT", G74 ou cancelar percurso restante antes da instrução GET.

- Uma parada de pré processamento é executado (da mesma forma que para STOPRE)
- **A execução é interrompida até que a transferência tenha sido completada.**

Eixos sem sincronismo:

Caso o eixo não tenha sido sincronizado, não ocorrerá parada de pré processamento com GET.

Exemplo:

```
N01 G0 X0
N02 RELEASE (AX5)
N03 G64 X10
N04 X20
N05 GET (AX5)

N06 G01 F5000
N07 X20

N08 X30
N09 ...
```

Caso não seja necessário o sincronismo, este não é um bloco executável
Não é um bloco executável
Não é um bloco executável pois a posição de X é a mesma de N04.
Primeiro bloco executável após N05.

1.15 Transferência de eixo, transferência de fuso



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

“GET” automático

Caso um eixo esteja a princípio disponível em um canal mas não esteja atualmente definido como um “eixo de canal”, GET será executado automaticamente. Caso os eixos (ou eixo) estejam sincronizados não será gerada parada de pré processamento.



Um eixo aceito com GET permanece relacionado ao canal mesmo após um reset via botões ou fim de programa. Quando um programa é iniciado, os eixos e fusos transferidos devem ser redefinidos através do programa caso estes sejam necessários nos canais de origem.

O eixo retornará a seu canal de origem (definido em dados de máquina) após a ligação do comando (power on

Transferência direta do eixo: GETD

Um eixo é tomado diretamente de outro canal com a instrução GETD (GET direto). Isto significa que não é necessária a programação do comando RELEASE no outro canal para a instrução GETD. Isto implica que a comunicação com o outro canal deve ter sido estabelecida.

(p.e. espera (wait) por marcadores).

1.15 Transferência de eixo, transferência de fuso



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di



Exemplo de programação

De 6 eixos, os seguintes são utilizados para usinagem no canal 1: 1., 2., 3. e 4.

Os eixos 5 e 6 pertencentes ao canal 2 são utilizados na troca de peças.

O eixo 2 deve ser transferido entre os dois canais e ser definido para o canal 1 após a ligação do comando.

Programa "MAIN" no canal 1

%_N_MAIN_MPF

| | |
|-----------------------|--|
| INIT (2, "TRANSFER2") | Seleção do programa TRANSFER2 no canal 2 |
| N... START (2) | Inicia o programa no canal 2 |
| N... GET (AX2) | Aceita o eixo AX2 |
| ... | |
| ... | |
| N... RELEASE (AX2) | Habilita o eixo AX2 |
| N... WAITM (1,1,2) | Espera pela marca no canal 1 e 2 para sincronizar os dois canais |
| N... | Restante do programa após a transferência |
| N... M30 | do eixo. |

Programa "Replace2" no canal 2

%_N_TRANSFER2_MPF

| | |
|--------------------|--|
| N... RELEASE (AX2) | |
| N160 WAITM (1,1,2) | Espera pela marca WAIT no canal 1 e 2, para sincronização dos canais |
| N150 GET (AX2) | Aceita eixo 2 |
| N... | Restante do programa após a transferência |
| N...M30 | |

1.16 NEWCONF: Ativando dados de máquina (A partir da SW 4.3)840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

1.16 NEWCONF: Ativando dados de máquina (A partir da SW 4.3)**Função**

Todos os dados de máquina do nível "NEW_CONFIG" serão ativados através do comando NEWCONF. A função corresponde à softkey "Ativar MD".

Quando a função NEWCONF é executada ocorre uma parada de pré processamento implícita, ou seja, o movimento é interrompido.

**Explicação**

NEWCONF

Todos os dados de máquina do tipo "NEW_CONFIG" são ativados

**Exemplo de programação**

Fresagem: Posicionamento para furação com diferentes tecnologias.

N10 \$MA_CONTOUR_TOL[AX]=1.0

; Alteração em dados de máquina

N20 NEWCONF

; Ativa a alteração dos dados de máquina

1.17 WRITE: Escrita de arquivo (a partir da SW 4.3)

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

1.17 WRITE: Escrita de arquivo (a partir da SW 4.3)



Programação

```
WRITE(var int ERROR, char[160] nomearquivo, char[200] string)
```

O comando WRITE anexa um bloco ao fim do arquivo especificado.



Explicação dos parâmetros

| | |
|-------------|---|
| Error | Variável de retorno de erro |
| 0 | Sem erro |
| 1 | Diretório não permitido |
| 2 | Diretório não encontrado |
| 3 | Arquivo não encontrado |
| 4 | Tipo de arquivo incorreto |
| 10 | Arquivo cheio |
| 11 | Arquivo sendo utilizado |
| 12 | Não há recursos livres |
| 13 | Faltam direitos de acesso |
| 20 | Outros erros |
| Nomearquivo | <p>Nome do arquivo na qual a string será escrita.</p> <p>O nome do arquivo pode ser especificado através do diretório e do nome do arquivo. Os diretórios devem ser representados de forma absoluta, ou seja, iniciam com "/". Caso o nome do arquivo não contenha o identificador de domínio (_N_), será realizado o acréscimo. Caso não seja especificado identificador (_MPF, SPF ou _CYC), _MPF será automaticamente adicionado. Caso o diretório não seja especificado, o arquivo será gravado no diretório atual (= diretório do programa selecionado). O nome do arquivo pode possuir até 32 bytes de comprimento, o nome do arquivo 128 bytes.</p> <p>Exemplo: PROFILE _N_PROFILE _N_PROFILE_MPF /_N_MPF_DIR/_N_PROFILE_MPF/</p> |
| String | Texto a ser gravado. LF é acrescentado internamente, ou seja, um caracter é acrescentado. |

1.17 WRITE: Escrita de arquivo (a partir da SW 4.3)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di



Função

Através do comando WRITE, dados (p.e., resultados de medições) podem ser anexados ao fim de um arquivo específico.

O comprimento máximo em KB do arquivo de protocolo é ajustado no MD 11420 LEN_PROTOCOL_FILE. Este comprimento é aplicado para todos os arquivos criados através do comando WRITE.

Uma vez que o arquivo atinja o comprimento especificado, uma mensagem de erro será emitida e a string não será gravada. Caso exista memória suficiente, um novo arquivo será criado.

Os arquivos criados podem ser

- Lidos, editados e apagados por qualquer usuário,
- Escritos através de programas atualmente em andamento.

Os blocos são inseridos após o fim do arquivo, ou seja, após M30.



Exemplo de programação

```

N10 DEF INT ERROR ;
N20 WRITE(ERROR,"TEST1","LOG FROM 7.2.97") ; Escreve o texto de LOG FROM 7.2.97 no arquivo TEST1
N30 IF ERROR ;
N40 MSG ("Erro na escrita com comando WRITE: "<<ERROR) ;
N50 M0 ;
N60 ENDIF ;
...
WRITE(ERROR, ; Diretório absoluto
"/_N_WCS_DIR/_N_PROT_WPD/_N_PROT_MPF", "LOG
FROM 7.2.97")

```



Informações adicionais

- Caso não exista o arquivo no NC, ele será criado e poderá ser escrito através do comando WRITE.
- Caso um arquivo de mesmo nome exista no disco rígido, este será sobrescrito após o arquivo ser fechado (no NC).
Remédio: Altere o nome no NC através da área de operação Serviços com auxílio da softkey "Propriedades".



Fabricante da máquina

Blocos do programa de usinagem podem ser gravados através do comando WRITE. O tamanho do arquivo (KB) é especificado em dados de máquina.

1.18 DELETE: Apagar arquivo (a partir da SW 4.3)

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

1.18 DELETE: Apagar arquivo (a partir da SW 4.3)



Programação

```
DELETE(var int error, char[160] nomearquivo)
```

O comando DELETE apaga o arquivo especificado.



Explicação dos parâmetros

| | |
|-------------|---|
| Error | Variável de retorno de erro |
| 0 | Sem erro |
| 1 | Diretório não permitido |
| 2 | Diretório não encontrado |
| 3 | Arquivo não encontrado |
| 4 | Tipo de arquivo incorreto |
| 11 | Arquivo sendo utilizado |
| 12 | Não há recursos livres |
| 20 | Outros erros |
| Nomearquivo | <p>Nome do arquivo a ser apagado.</p> <p>O nome do arquivo pode ser especificado através do diretório e do nome do arquivo. Os diretórios devem ser representados de forma absoluta, ou seja, iniciam com "/". Caso o nome do arquivo não contenha o identificador de domínio (_N_), será realizado o acréscimo. Caso não seja especificado identificador (_MPF, SPF ou _CYC), _MPF será automaticamente adicionado. Caso o diretório não seja especificado, o arquivo será gravado no diretório atual (= diretório do programa selecionado). O nome do arquivo pode possuir até 32 bytes de comprimento, o nome do arquivo 128 bytes.</p> <p>Exemplo: PROFILE _N_PROFILE _N_PROFILE_MPF /_N_MPF_DIR/_N_PROFILE_MPF/</p> |

1.19 READ: Lê linhas de arquivos (a partir da SW 5.2)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di



Função

Todos os arquivos podem ser apagados através do comando DELETE. Mesmo arquivos criados com direitos de acesso superiores podem também ser apagados com DELETE.



Exemplo de programação

```

N10 DEF INT ERROR ;
N20 DELETE(ERROR,"TEST1") ; Apagar o arquivo TEST1
N30 IF ERROR ;
N40 MSG ("Erro no comando DELETE:" ;
        <<ERROR)
N50 M0 ;
N60 ENDIF ;
...

```

1.19 READ: Lê linhas de arquivos (a partir da SW 5.2)



Programação

```

READ(var int error, string[160] file, int line, int number, var
string[255] result[])

```

O comando READ lê uma ou várias linhas no arquivo especificado e grava a informação lida num array tipo STRING. Neste array, cada linha lida irá ocupar um elemento do array.



Explicação dos parâmetros

| Error | Variável de retorno de erro (chamado através de parâmetro, tipo INT) |
|-------|--|
| 0 | Sem erro |
| 1 | Diretório não permitido |
| 2 | Diretório não encontrado |
| 3 | Arquivo não encontrado |
| 4 | Tipo de arquivo incorreto |
| 13 | Faltam direitos de acesso |
| 21 | Linha não disponível (parâmetro "linha" ou "número" maior que quantidade de linhas do arquivo) |
| 22 | Comprimento da variável do array de destino muito pequeno |
| 23 | Gama de linhas muito grande (parâmetro "número" selecionado) |

1.19 READ: Lê linhas de arquivos (a partir da SW 5.2)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

| | |
|---------------------|--|
| | muito grande, a leitura irá além do fim do arquivo). |
| <code>file</code> | Nome/diretório do arquivo a ser lido (chamado através do valor do parâmetro de tipo STRING com um comprimento máximo de 160 bytes). O arquivo deve estar gravado na memória de usuário da NCK (sistema de arquivos passivo). O nome do arquivo pode conter o identificador de domínio (<code>_N_</code>). Caso não possua, será realizado o acréscimo adequado. Caso não seja especificado identificador, <code>_MPF</code> será automaticamente adicionado. Caso o diretório não seja especificado, o arquivo será procurado no diretório atual (= diretório do programa selecionado). Caso especificado o diretório no campo "file", este deve ser iniciado com um "/" (identificação absoluta). |
| <code>line</code> | Indicação da posição da faixa de linhas a serem lidas (especificadas através do parâmetro de tipo INT). 0 Será lida a quantidade de linhas antes do fim do arquivo especificado através do parâmetro "number". 1 a n Número da primeira linha a ser lida. |
| <code>number</code> | Quantidade de linhas a serem lidas (especificadas pelo parâmetro tipo INT). |
| <code>result</code> | Array do tipo STRING, onde o texto lido será gravado (chamado através do parâmetro de referência com o comprimento de 255). |



Função

Uma ou mais linhas podem ser lidas a partir de um arquivo através do comando READ. As linhas lidas serão gravadas em um elemento de um array. A informação é disponibilizada em cadeias de caracteres (string).



Informações adicionais

- Arquivos binários não podem ser lidos. A mensagem de erro =4: tipo de arquivo errado será emitida. Os seguintes tipos de arquivo não podem ser lidos: `_BIN`, `_EXE`, `_OBJ`, `_LIB`, `_BOT`, `_TRC`, `_ACC`, `_CYC`, `_NCK`.
- O nível de proteção atual deve ser igual ou superior ao do arquivo a ser lido. Caso contrário o acesso será negado com o erro =13.
- Caso a quantidade de linhas especificado no parâmetro "number" seja menor que o comprimento do array "result", os outros elementos do array não serão alterados.
- Os caracteres de controle de fim de linha "LF" (Line Feed) ou "CR

1.19 READ: Lê linhas de arquivos (a partir da SW 5.2)840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

LF" (Carriage Return Line Feed) não são gravados na variável "result". A linha lida será cortada, caso a linha seja mais longa que o comprimento da variável de destino "result". Não serão emitidas mensagens de erro.

**Exemplo de programação**

```

N10 DEF INT ERROR                                ; variável de erro
N20 STRING[255] RESULT[5]                        ; variável resultado
...
N30 READ(ERROR, "TESTFILE", 1, 5,                ; nome do arquivo sem identificação de
          RESULT)                                domínio e arquivo
...
N30 READ (ERROR, "TESTFILE_MPF", 1, 5,            ; nome do arquivo sem identificação de
          RESULT)                                domínio e com identificador de arquivo
...
N30 READ(ERROR, "_N_TESTFILE_MPF", 1, 5,          ; nome do arquivo com identificação de
          RESULT)                                domínio e arquivo
...
N30 READ(ERROR, "/_N_CST_DIR/N_TESTFILE          ; nome do arquivo com identificação de
          _MPF", 1, 5 RESULT)                    domínio, arquivo e diretório
^...
N40 IF ERROR <> 0                                ; checagem de erros
N50 MSG("ERRO"<<ERROR<<" com o comando de
leitura")
N60 M0
N70 ENDIF
...

```


1.20 ISFILE: Arquivo disponível na memória(a partir da SW5.2)

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

1.20 ISFILE: Arquivo disponível na memória de usuário NCK (a partir da SW5.2)



Programação

```
result=isfile(string[160]file)
```

Com o comando ISFILE você pode checar se um arquivo existe na memória de usuário da NCK (sistema de arquivos passivo). Como resultado, os estados TRUE (arquivo existe) ou FALSE (arquivo não existe) serão emitidos.



Explicação dos parâmetros

| | |
|------|---|
| File | Nome/diretório do arquivo a ser lido (chamado através do valor do parâmetro de tipo STRING com um comprimento máximo de 160 bytes). O arquivo deve estar gravado na memória de usuário da NCK (sistema de arquivos passivo). O nome do arquivo pode conter o identificador de domínio (_N_). Caso não possua, será realizado o acréscimo adequado. O identificador de arquivo ("_" mais três caracteres, p.e. SPF) é opcional. Caso não seja especificado identificador, _MPF será automaticamente adicionado. Caso o diretório não seja especificado, o arquivo será procurado no diretório atual (= diretório do programa selecionado). Caso especificado o diretório no campo "file", este deve ser iniciado com um "/" (identificação absoluta). |
|------|---|

| | |
|--------|--|
| Result | Variável para gravação do resultado de tipo BOOL (TRUE ou FALSE) |
|--------|--|



Exemplo de programação

```
N10 DEF BOOL RESULT
N20 RESULT=ISFILE("TESTFILE")
N30 IF (RESULT==FALSE)
N40   MSG("FILE DOES NOT EXIST")
N50   M0
N60 ENDIF
...
ou:
N30 IF (NOT ISFILE("TESTFILE"))
N40   MSG("FILE DOES NOT EXIST")
N50   M0
N60 ENDIF
...
```

1.21 CHECKSUM: Criar um checksum em um array (> SW 5.2)840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

1.21 CHECKSUM: Criar o checksum em um array (> SW 5.2)**Programação**

```
error=CHECKSUM(var string[16] chksum,string[32]array, int first, int
last)
```

A função CHECKSUM cria o checksum de um array.

**Explicação dos parâmetros**

| | | |
|--------|---|--------------------------|
| Error | Variável de retorno de erro | |
| | 0 | Sem erro |
| | 1 | Símbolo não encontrado |
| | 2 | não existe o array |
| | 3 | índice 1 muito grande |
| | 4 | índice 2 muito grande |
| | 5 | tipo de arquivo inválido |
| chksum | 10 | estouro de checksum |
| | Checksum de um array de seqüências de caracteres (chamado através do parâmetro referência de tipo string, com um comprimento de 16). O checksum será indicado como sendo uma seqüência de caracteres de 16 números hexadecimais. Entretanto, nenhum formato de caracteres será indicado. Exemplo: em MY_CHECKSUM | |
| array | Número do array a partir do qual a checksum será formada. (chamado por um parâmetro de tipo string com comprimento máximo de 32). | |
| | Arrays permitidos: 1 ou 2 dimensões dos tipos BOOL, CHAR, INT, REAL, STRING | |
| | Arrays contendo dados da máquina não são permitidos. | |
| first | Número da coluna de início (opcional) | |
| last | Número da última coluna (opcional) | |

**Função**

Com CHECKSUM você cria o checksum de um array.

Aplicável na aplicação desbaste:

Verifica se o contorno inicial foi alterado.

1.21 CHECKSUM: Criar um checksum em um array (> SW 5.2)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di



Informações adicionais

Os parâmetros *first* e *last* são opcionais. Caso não sejam indicados os índices de colunas, o checksum será executado para todo o array.

O resultado do checksum é único. Caso algum elemento do array seja alterado, a string resultante também será alterada.



Exemplo de programação

```
N10 DEF INT ERROR
N20 DEF STRING[16] MY_CHECKSUM
N30 DEF INT MY_VAR[4,4]
N40 MY_VAR=...
N50 ERROR=CHECKSUM
    (CHECKSUM;"MY_VAR", 0, 2)
...
```

Retorna em MY_CHECKSUM o valor
"A6FC3404E534047C"



Notas

Subprogramas, Macros

| | | |
|------|---|-------|
| 2.1 | Utilizando subprogramas | 2-92 |
| 2.2 | Subprograma com o mecanismo SAVE | 2-94 |
| 2.3 | Subprogramas com transferência de parâmetros | 2-95 |
| 2.4 | Chamando subprogramas | 2-99 |
| 2.5 | Repetir subprograma | 2-103 |
| 2.6 | Subprograma modal, MCALL | 2-104 |
| 2.7 | Chamada indireta de subprograma | 2-105 |
| 2.8 | Chamando subprogramas com diretórios e parâmetros, PCALL | 2-106 |
| 2.9 | Suprimindo a exibição do bloco atual, DISPLOF | 2-107 |
| 2.10 | Supressão de bloco a bloco, SBLOF, SBLON (SW 4.3 em diante) | 2-108 |
| 2.11 | Execução de subprogramas externos (SW 4.2 em diante) | 2-111 |
| 2.12 | Ciclos: Carregando parâmetros em ciclos de usuário | 2-114 |
| 2.13 | Macros | 2-120 |

2.1 Utilizando subprogramas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

2.1 Utilizando subprogramas



O que é um subprograma?

Em princípio, um subprograma possui a mesma estrutura de um programa principal. É composto de blocos NC com comandos de movimento e ligar/desligar funções.

A princípio, não existem diferenças entre um programa e um subprograma. O subprograma contém também ciclos de usinagem ou seções de usinagem que devem ser executadas mais de uma vez.

Uso de subprogramas

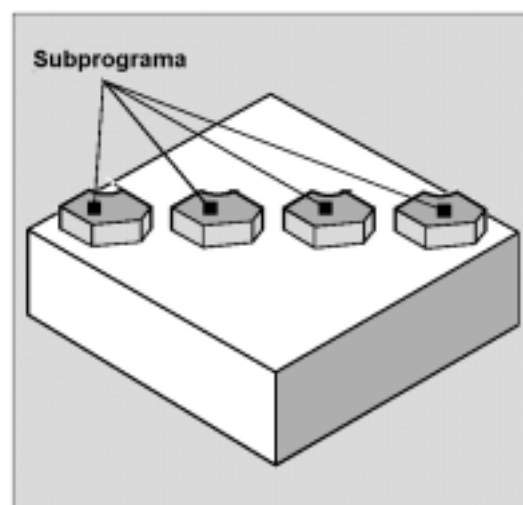
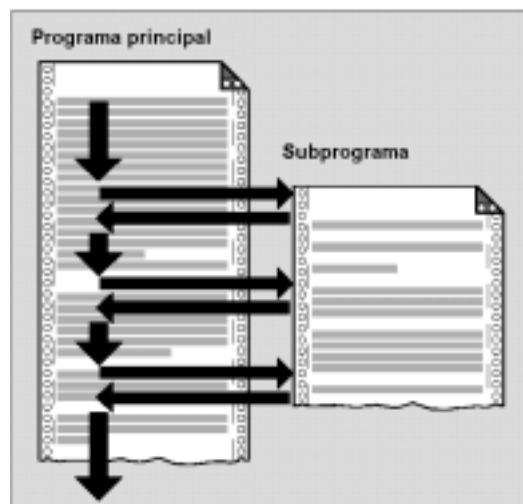
Seqüências de usinagem que repetidas podem ser programadas somente uma vez em um subprograma. Por exemplo, certos perfis de contorno que ocorrem mais de uma vez ou ciclos de usinagem.

Este subprograma pode ser chamado e executado em qualquer programa principal.

Estrutura do subprograma

A estrutura de um subprograma é idêntica à de um programa principal.

Em um subprograma é também possível programar um cabeçalho com definição de parâmetros.



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Profundidade de encadeamento

Encadeamento de subprogramas

Um subprograma pode conter outras chamadas de subprogramas. Os subprogramas chamados podem também conter outras chamadas de subprogramas, etc.

A quantidade máxima de níveis de chamada de subprogramas é de 12 níveis.

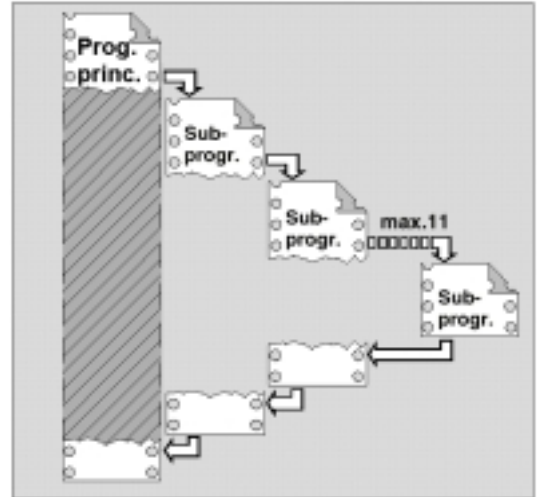
Isto significa:

Um programa principal pode conter 11 níveis de chamada de subprogramas.

Restrições

É também possível chamar subprogramas em rotinas de interrupção. Para trabalhar com subprogramas você deve manter livres quatro níveis, ou trabalhar com no máximo sete níveis de chamadas de subprogramas.

Os ciclos de usinagem e de medição SIEMENS necessitam de 3 níveis. Caso um ciclo seja chamado de uma subrotina, esta não deve estar em um nível superior à 5 (caso quatro níveis sejam reservados para as rotinas de interrupção).



2.2 Subprograma com o mecanismo SAVE



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

2.2 Subprograma com o mecanismo SAVE

Para isto, especifique o comando adicional SAVE com a instrução PROC.

Com o atributo SAVE, no fim da rotina de interrupção as funções G modais ativas no instante da ativação da interrupção serão reativadas. Os deslocamentos de origem e o deslocamento básico são restabelecidos, assim como os deslocamentos ajustáveis (funções G modais do grupo 8). Caso o grupo de funções G 15 (tipo de avanço) estivessem alterados, p.e. de G94 para G95, O valor apropriado de F é também restabelecido.

Exemplo:

Definição de subprograma

```
PROC CONTOUR SAVE
N10 G91 ...
N100 M17
```

Programa principal

```
%123
N10 G0 X... Y... G90
N20...
N50 CONTOUR
N60 X... Y...
```

No subprograma CONTOUR G91 define dimensões incrementais. Após retornar ao programa principal, são ativadas novamente as dimensões absolutas, pois as funções modais do programas principal foram gravadas com SAVE.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2.3 Subprogramas com transferência de parâmetros



Início de programa, PROC

Um subprograma que deve utilizar parâmetros do programa que o chamou é identificado com a instrução PROC.

Fim de programa M17, RET

O comando M17 identifica o fim de um subprograma e é também uma instrução de retorno para o programa principal que o chamou.

Como alternativa à M17, a instrução RET finaliza um subprograma sem interrupção do modo “passagens contínuas” e sem envio de sinais ao PLC



RET deve ser programada em um bloco NC separado.

Exemplo:

```
PROC CONTOUR
N10...
...
N100 M17
```

Transferência de parâmetros entre um programa principal e um subprograma

Caso você esteja trabalhando com parâmetros no programa principal, você pode usar valores calculados ou definidos em subprogramas.

Para este propósito, os valores dos **parâmetros atuais** do programa principal são transferidos para **parâmetros formais** do subprograma na sua chamada, e processados durante a execução do subprograma.

2.3 Subprogramas com transferência de parâmetros



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

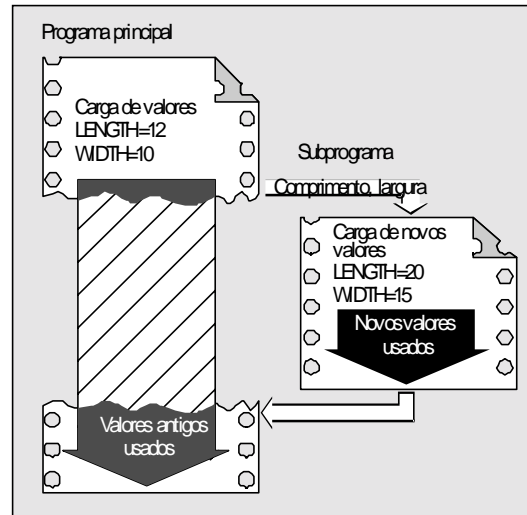
Exemplo:

```
N10 DEF REAL LENGTH,WIDTH
N20 LENGTH=12 WIDTH=10
N30 BORDER (LENGTH,WIDTH)
```

Os valores definidos na N20 no programa principal são passados em N30 durante a chamada do subprograma.

Os parâmetros são passados na sequência declarada.

Os nomes dos parâmetros não necessitam ser idênticos no programa principal e no subprograma.



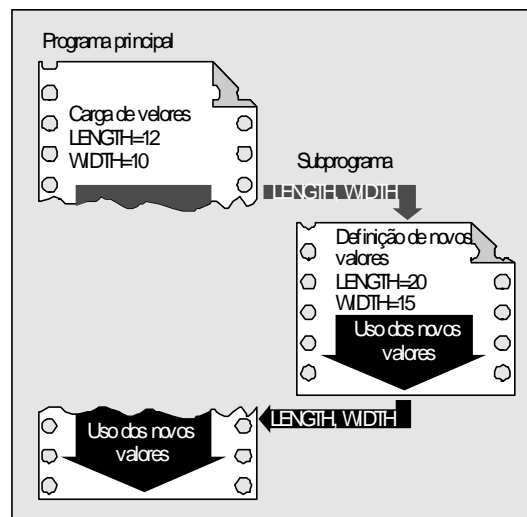
Duas formas de transferir parâmetros

Os valores são somente passados (chamada por valor)

Caso os parâmetros transferidos sejam alterados no decorrer do subprograma, estas alterações não terão efeito no programa principal. Os parâmetros permanecem inalterados (vide figura)

Transferência de parâmetros com troca de dados (chamada por referência)

Quaisquer alterações nos parâmetros ocorridas no subprograma também causam alterações no programa principal (vide figura).



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Programação

Os parâmetros relevantes para a transferência devem ser listados no início do subprograma com seus nomes e tipos.

Parâmetros transferidos – valores apenas passados para a subrotina (chamada por valor)

```
PROC PROGRAM_NAME(VARIABLE_TYPE1 VARIABLE1,VARIABLE_TYPE2 VARIABLE2,...)
```

Exemplo:

```
PROC CONTOUR(REAL LENGTH, REAL WIDTH)
```

Parâmetros transferidos – valores retornam da subrotina (chamada por referência), identificados através da instrução VAR

```
PROC PROGRAM_NAME(VARIABLE_TYPE1 VARIABLE1,VARIABLE_TYPE2 VARIABLE2, ...)
```

Exemplo:

```
PROC CONTOUR(VAR REAL LENGTH, VAR REAL WIDTH)
```

Transferência de array com retorno de parâmetros da subrotina, identificados através da instrução VAR

```
PROC PROGRAM_NAME(VAR VARIABLE_TYPE1 ARRAY_NAME1[tamanho do array],
VAR VARIABLE_TYPE2 ARRAY_NAME2[Tamanho do array], VAR VARIABLE_TYPE3
ARRAY_NAME3[array tamanho1, array tamanho2], VAR VARIABLE_TYPE4
ARRAY_NAME4[ ],
VAR VARIABLE_TYPE5 ARRAY_NAME5 [,tamanho do array ])
```

Exemplo:

```
PROC PALLET (VAR INT ARRAY[,10])
```



Informações adicionais

As definições com PROC devem ser programadas em um bloco NC separado. Um México de 127 parâmetros podem ser declarados para transferência de parâmetros.

2.3 Subprogramas com transferência de parâmetros



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Definição de arrays

As seguintes regras são aplicadas quando definidos os parâmetros formais:

Com arrays de duas dimensões o número do campo da primeira dimensão não tem a necessidade de ser especificado, mas a vírgula deve ser escrita.

Exemplo:

```
VAR REAL ARRAY[ ,5]
```

Com certas dimensões de arrays é possível processar subprogramas com arrays de comprimento variável. Entretanto quando definindo as variáveis, você deve definir quantos elementos elas devem conter.

Vide as explicações sobre a definição de arrays neste manual.



Exemplo de programação

Programação com comprimento de array variável

| | |
|--|------------------------|
| %_N_DRILLING_PLATE_MPF | Programa principal |
| DEF REAL TABLE[100,2] | Define posição |
| EXTERN DRILLING_PATTERN (VAR REAL[,2],INT) | |
| TABLE[0.0]=-17.5 | Define posições |
| ... | |
| TABLE[99.1]=45 | |
| DRILLING_PATTERN(TABLE,100) | Chamada do subprograma |
| M30 | |

Criando um padrão p/furação com a tabela de posições variáveis passada na chamada da subrotina

| | |
|---|---|
| %_N_DRILLING_PATTERN_SPF | Subprograma |
| PROC DRILLING_PATTERN(VAR REAL ARRAY[,2],-> -> INT NUMBER) | Parâmetros passados pelo programa principal |
| DEF INT COUNT | |
| STEP: G1 X=ARRAY[COUNT,0]-> -> Y=ARRAY[COUNT,1] F100 | Seqüência de usinagem |
| Z=IC(-5) | |
| Z=IC(5) | |
| COUNT=COUNT+1 | |
| IF COUNT<NUMBER GOTOB STEP | |
| RET | Fim do subprograma |

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

2.4 Chamando subprogramas

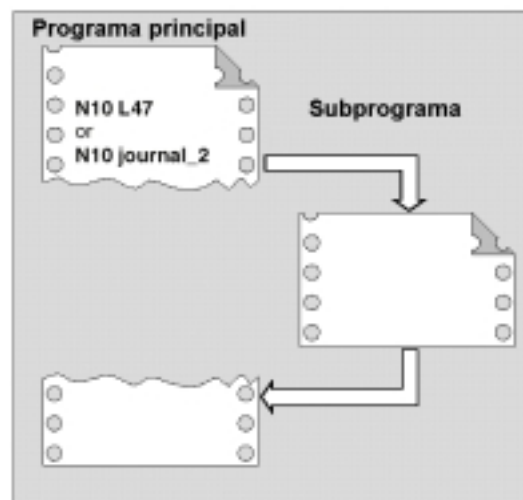
Chamada de subprograma sem transferência de parâmetros

No programa principal você chama o subprograma tanto com o endereço L e o número do subprograma ou através do nome do subprograma.

Exemplo:

N10 L47 OU

N10 SPIGOT_2



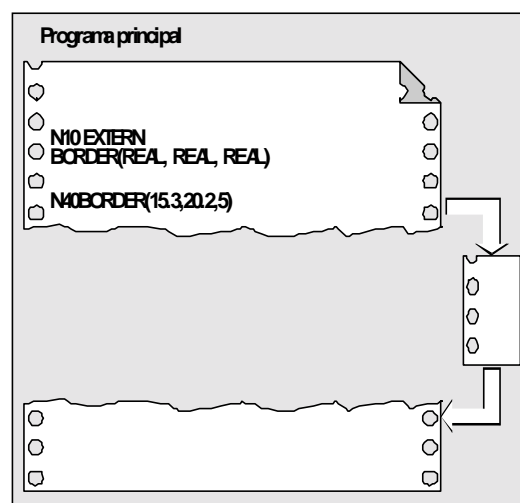
Subprograma com transferência de parâmetros, declarações com EXTERN

Subprogramas com transferência de parâmetros devem ser listados com EXTERN no programa principal antes de sua chamada, por exemplo, no início do programa.

O nome do subprograma e os tipos de variáveis são declarados na sequência que devem ser transferidos.

Você somente deve especificar EXTERN caso o subprograma esteja no diretório da peça ou no diretório global de subprogramas.

Você não deve declarar ciclos como EXTERN.



EXTERN instrução

EXTERN NAME (TYP1, TYP2, TYP3, ...) OU

EXTERN NAME (VAR TYP1, VAR TYP2, ...)

Exemplo:

N10 EXTERN BORDER (REAL, REAL, REAL)

...

N40 BORDER (15.3, 20.2, 5)

N10 Declaração do subprograma, N40 chamada do subprograma com transferência de parâmetros.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Chamada de subprograma com transferência de parâmetros

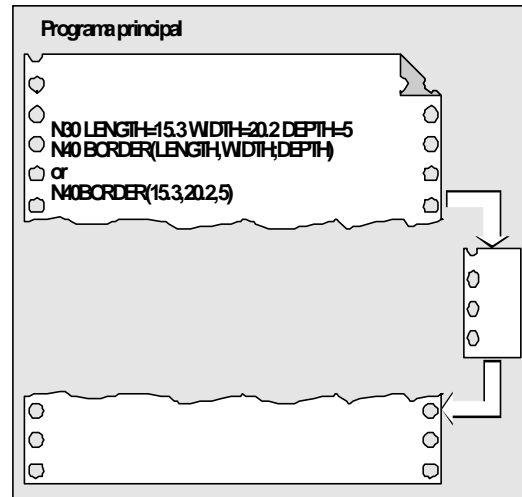
No programa principal você chama o subprograma especificando seu nome e parâmetros de transferência. Ao transferir parâmetros você pode transferir variáveis ou valores diretamente (não para parâmetros VAR).

Exemplo:

```
N10 DEF REAL LENGTH,WIDTH,DEPTH
N20 ...
N30 LENGTH=15.3 WIDTH=20.2 DEPTH=5
N40 BORDER(LENGTH,WIDTH,DEPTH)
```

Ou

```
N40 BORDER(15.3,20.2,5)
```



A definição do subprograma deve combinar com a sua chamada.

Os tipos das variáveis e a seqüência de transferência devem combinar com as declarações de PROC no nome do subprograma os nomes dos parâmetros podem ser diferentes no programa principal e no subprograma.

Exemplo:

Definição no subprograma:

```
PROC BORDER(REAL LENGTH, REAL WIDTH, REAL DEPTH)
```

Chamada no programa principal:

```
N30 BORDER(LENGTH, WIDTH, DEPTH)
```



840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

Transferência de parâmetros incompleta

Em uma chamada de subprograma, os valores dos parâmetros podem ser omitidos. Neste caso, o parâmetro em questão é carregado com o valor **zero** no subprograma.

As vírgulas devem ser sempre escritas para indicar a sequência. Caso os parâmetros omitidos sejam os últimos da sequência, as vírgulas podem também ser omitidas.

Retornando ao exemplo anterior:

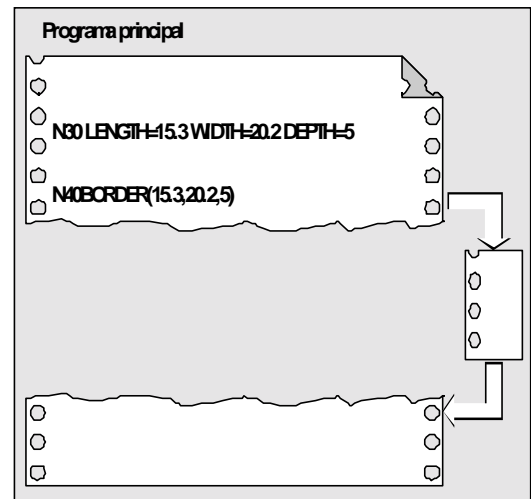
```
N40 BORDER(15.3, , 5)
```

O valor 20.2 foi aqui omitido.

Nota



*O parâmetro de tipo AXIS não pode ser omitido.
Os parâmetros VAR devem ser completamente transmitidos.*



SW 4.4 em diante:

Com a transferência incompleta de parâmetros, é possível informar ao sistema através da variável `$P_SUBPAR[i]` quando a transferência de parâmetros foi programada ou não na chamada da subrotina.

A variável de sistema contém no argumento (i) a quantidade de parâmetros transferidos.

A variável de sistema `$P_SUBPAR` retorna

- TRUE, caso a transferência de parâmetros tenha sido programada
- FALSE, caso nenhum valor tenha sido carregado como parâmetro de transferência.

Caso uma quantidade não permitida de parâmetros tenha sido especificada, o processamento do programa é abortado e um alarme é emitido.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

Exemplo:**Subprograma**

```

PROC SUB1 (INT VAR1, DOUBLE VAR2)

IF $P_SUBPAR[1]==TRUE
    ;Parâmetro VAR1 não estava presente
    ;na chamada do subprograma
ELSE
    ;Parâmetro VAR1 não programado
    ;na chamada do subprograma
    ;com valor 0
ENDIF

IF $P_SUBPAR[2]==TRUE
    ;Parâmetro VAR2 não estava presente
    ;na chamada do subprograma
ELSE
    ;Parâmetro VAR2 não programando
    ;na chamada do subprograma
    ;e nele foi carregado o valor 0.0
    ;pelo sistema
ENDIF

;Parâmetro e não definido
IF $P_SUBPAR[3]==TRUE -> Alarme 17020
M17

```

Chamada de um programa principal como subrotina

Um programa principal pode também ser chamado como se fosse um subprograma. O fim de programa M2 ou M30 disparados no final do programa são considerados como M17 neste caso (fim de programa com retorno ao programa que fez a chamada).

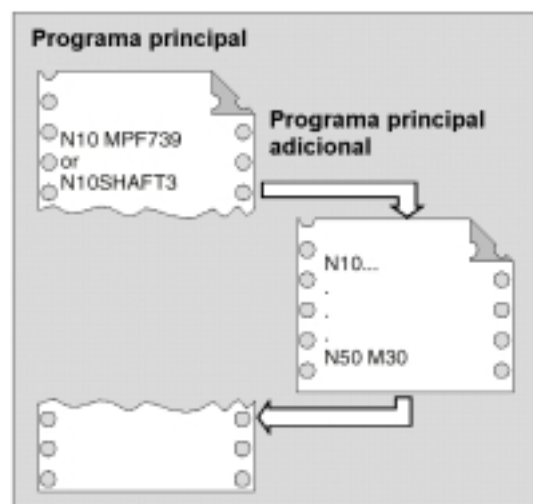
A chamada do programa é feita através de seu nome.

Exemplo:

```

N10 MPF739 ou
N10 SHAFT3

```



Um subprograma pode também ser iniciado como se fosse um programa principal.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2.5 Repetir subprograma

Repetição de programa, P

Caso você queira executar um subprograma várias vezes em seqüência, você pode programar a quantidade desejada de repetições no bloco onde ocorre a chamada do subprograma através do endereço P.

Exemplo:

N40 BORDER P3

O subprograma Border deve ser executado 3 vezes em seguida.

Faixa de valores:

P: 1...9999

A seguinte regra é aplicada em toda chamada de subprogramas:



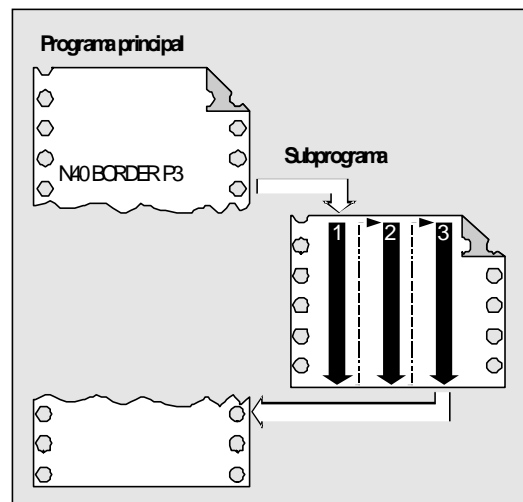
A chamada de subprogramas deve ocorrer em um bloco de NC separado (que contém somente a instrução de chamada)

Chamada de subprograma com repetição e transferência de parâmetros



Os parâmetros são transferidos somente durante a chamada do programa ou na primeira passagem. Os parâmetros permanecem inalterados para as demais repetições.

Caso você queira alterar os parâmetros nas repetições do programa você deve redefinir as declarações em subprogramas.



2.6 Subprograma modal, MCALL

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2.6 Subprograma modal, MCALL

Chamada modal de subprograma, MCALL

Com esta função o subprograma é automaticamente chamado e executado após todo bloco com movimentos de trajetória. Desta forma, você pode automatizar a chamada de subprogramas que devem ser executados em diferentes posições da peça. Por exemplo, para padrões de furação.

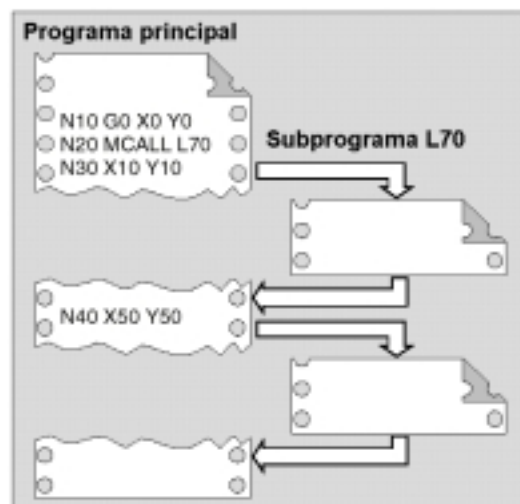
Exemplos:

```
N10 G0 X0 Y0
N20 MCALL L70
N30 X10 Y10
N40 X50 Y50
```

Nos blocos N30 a N40, a posição programada é atingida e a subrotina L70 é executada.

```
N10 G0 X0 Y0
N20 MCALL L70
N30 L80
```

Neste exemplo, os blocos de NC programados com eixos de trajetória estão gravados em L80. L70 será chamada por L80.



Durante a execução do programa, somente um MCALL pode ser programado por vez. Os parâmetros são passados somente uma vez, na linha da instrução MCALL.

Desativando a chamada modal de subprograma

Com MCALL sem a especificação de subprograma ou com um novo MCALL para outro subprograma.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

2.7 Chamada indireta de subprograma

Chamada indireta de subprograma, CALL

Dependendo das condições de um ponto em particular do programa, diferentes subprogramas podem ser chamados.

O nome do subprograma é gravado em uma variável tipo STRING. A chamada do subprograma é realizada através da instrução CALL e do nome da variável.



A chamada indireta de subprogramas é permitida somente para subprogramas sem transferência de parâmetros.



Para chamar um subprograma de forma direta, grave o nome em uma string constante.

Exemplo:

Chamada direta de subprograma com string constante:

```
CALL "/_N_WCS_DIR/_N_SUBPROG_WPD/_N_PART1_SPF"
```

Chamada indireta via variável:

```
DEF STRING[100] PROGNAME
PROGNAME = "/_N_WCS_DIR/_N_SUBPROG_WPD/_N_PART1_SPF"
CALL PROGNAME
```

O nome do subprograma PART1 é carregado na variável PROGNAME. Com CALL e o nome especificado na variável, o subprograma é chamado de forma indireta.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2.8 Chamando subprogramas com diretórios e parâmetros, PCALL

Com PCALL você pode chamar subprogramas especificando-se de forma absoluta o diretório e os parâmetros:

PCALL diretório/nome (parâmetro 1, ..., parâmetro n)



Explicação

PCALL

Instrução para chamada absoluta de subprograma com diretório

Diretório/nome

Nome absoluto do diretório, começando com "/", e incluindo o nome do subprograma.

Caso não seja programado o diretório do subprograma, PCALL se comporta como uma chamada comum de subprograma. O nome do subprograma deve ser escrito sem os caracteres guias `_N_` e sem extensão.

Caso você queira que o nome do programa contenha o guia `_N_` e a extensão, você deve declarar isto de forma explícita com o guia `_N_` e a extensão como externos.

Parâmetros 1 a n

Os parâmetros atuais de acordo com as declarações em PROC do subprograma

Exemplo:

PCALL/_N_WCS_DIR/_N_SHAFT_WPD/SHAFT(parâmetro1, parâmetro2, ...>)

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2.9 Suprimindo a exibição do bloco atual, DISPLOF



Programação

PROC ... DISPLOF



Função

Com DISPLOF a exibição dos blocos atuais é suprimida no subprograma. DISPLOF deve ser programado no final instrução PROC. Ao invés do bloco atual, a chamada do ciclo ou subprograma será exibida.

Por definição, a exibição de blocos de programa encontra-se ativada. A desativação com DISPLOF é aplicada até que ocorra o retorno daquela subrotina ou fim do programa. Caso outros subprogramas sejam chamados a partir daquele em que foi programado DISPLOF, a exibição destes também será suprimida. Caso um subprograma com supressão de exibição de blocos seja interrompido por um subprograma assíncrono, os blocos do programa atual serão exibidos.



Exemplo de programação

Suprimindo a exibição dos blocos em um ciclo

```
%_N_CYCLE_SPF
```

```
; $PATH=/_N_CUS_DIR
```

```
PROC CYCLE (AXIS TOMOV, REAL POSITION) SAVE DISPLOF
```

```
;Suprime a exibição do bloco atual
```

```
;A partir de então a chamada do ciclo  
será exibida como sendo o bloco atual
```

```
;p.e.: CYCLE(X, 100.0)
```

```
DEF REAL DIFF
```

```
;Conteúdo do ciclo
```

```
G01 ...
```

```
;
```

```
...
```

```
RET
```

```
;Retorno do subprograma, o bloco  
seguinte à chamada do programa será  
novamente exibido.
```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2.10 Supressão de bloco a bloco, SBLOF, SBLON (SW 4.3 em diante)



Programação

PROC ... SBLOF ; O comando pode ser programado em um bloco PROC ou em bloco separado
 SBLON ; O comando deve ser programado em bloco separado



Explicação

SBLOF
 SBLON

Desativa bloco a bloco
 Reativa bloco a bloco



Função

Programação específica de supressão de bloco a bloco

Todos os blocos a partir da programação de SBLOF são executados como se fossem apenas um bloco. Caso SBLOF seja programado na linha de PROC, será válido durante toda a subrotina ou até que esta seja abortada.

SBLOF é válido também nos subprogramas chamados.

Exemplo:

```
PROC EXAMPLE SBLOF
G1 X10
RET
```

Suprimindo bloco a bloco no programa

SBLOF pode estar sozinho em um bloco. Deste bloco em diante, o modo bloco a bloco estará desativado até

- O próximo SBLON ou
- Até o fim do subprograma ativo.

Exemplo:

```
N10 G1 X100 F1000
N20 SBLOF
N30 Y20
N40 M100
N50 R10=90
```

Desativa bloco a bloco

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

N60 SBLON

N70 M110

N80 . . .

Reativa bloco a bloco



Em bloco a bloco, as linhas de N20 a N60 são executadas como se fossem somente um bloco.

Desabilitando bloco a bloco para subprogramas assíncronos

Os subprogramas ASUP1.SYF e ASUP2.SYF rodam internamente no sistema com REPOS e devem ser executados passo a passo. Nas versões de software 4.3 e superiores, as ASUP de sistema podem ser executadas em um passo através da programação de SBLOF.

Exemplo:

N10 SBLOF

N20 IF \$AC_ASUP== 'H200 '

N30 RET

N40 ELSE

N50 REPOSA

N60 ENDIF

N70 RET

Sem REPOS na troca de modo

REPOS em todos os outros casos

Condições adicionais

- A exibição do bloco atual pode ser suprimida através da instrução DISPLOF.
- Caso DISPOF seja programado em conjunto com SBLOF, a chamada do ciclo continuará a ser exibida em bloco a bloco durante o ciclo.
- A definição inicial feita em MD20117: IGNORE_SINGLEBLOCK_ASUP para o comportamento dos programas assíncronos em bloco a bloco pode ser alterada através de SBLOF.
- Para testes, é possível suprimir SBLOF através de uma variável OPI (vide documentação OEM).

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Exemplo de programação 1****O ciclo deve atuar como um comando para o programador**

Programa principal

N10 G1 X10 G90 F200

N20 X-4 Y6

N30 CYCLE1

N40 G1 X0

N50 M30

Ciclo de programa 1

N100 PROC CYCLE1 DISPLOF SBLOF

Suprime bloco a bloco

N110 R10=3*SIN(R20)+5

N120 IF (R11 <= 0)

N130 SETAL(61000)

N140 ENDIF

N150 G1 G91 Z=R10 F=R11

N160 RET



O ciclo CYCLE1 é executado como um único passo quando bloco a bloco for ativado.

**Exemplo de programação 2**

Uma ASUP que roda a partir do PLC para ativação de deslocamentos de origem e corretores de ferramentas não deve ser visível.

N100 PROC NV SBLOF DISPLOF

N110 CASE \$P_UIFRNUM OF 0 GOTOF _G500

-->1 GOTOF _G54 2 GOTOF _G55 3

-->GOTOF _G56 4 GOTOF _G57

-->DEFAULT GOTOF END

N120 _G54: G54 D=\$P_TOOL T=\$P_TOOLNO

N130 RET

N140 _G54: G55 D=\$P_TOOL T=\$P_TOOLNO

N150 RET

N160 _G56: G56 D=\$P_TOOL T=\$P_TOOLNO

N170 RET

N180 _G57: G57 D=\$P_TOOL T=\$P_TOOLNO

N190 RET

N200 END: D=\$P_TOOL T=\$P_TOOLNO

N210 RET



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

2.11 Execução de subprogramas externos (SW 4.2 em diante)



Esta função encontra-se disponível na MMC102/103

Você pode utilizar EXTCALL para recarregar um programa da MMC102/103 no modo “Execução externa”.

EXTCALL diretório/nome do programa



Explicação

EXTCALL
Diretório

Nome do programa

Exemplo:

EXTCALL "SHAFT" **p.e.**

EXTCALL " / _N_WCS_DIR / _N_SHAFT_WPD / SHAFT "

Instrução para chamada de subprograma
Opcional, não necessário
Constante/variável tipo STRING. Nome do diretório especificado de forma absoluta começando por "/",
O identificador do programa é escrito com/sem o identificador `_N_` e sem extensão. Uma extensão pode ser anexada utilizando-se o caracter `<">`.



Função

Durante a usinagem de peças complexas, seqüências de programa podem ser geradas para estágios de usinagem separados que não podem ser gravados na memória principal devido às limitações de memória do sistema. Você pode utilizar EXTCALL para recarregar um programa da MMC 102/103 no modo “Execução externa”.

Todos os programas que podem ser acessados através da estrutura de diretório da MMC102 podem ser recarregados.

SD 42700 EXT_PROG_PATH

O dado setting específico de canal

SD 42700 EXT_PROG_PATH encontra-se disponível para especificar o diretório de chamada da subrotina.

SD 42700 contém a definição de diretório a partir do qual o programa especificado será chamado.

2.11 Execução de subprogramas externos (SW 4.2 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Caso o subprograma externo seja chamado sem um diretório específico, a mesma pesquisa de diretório feita na memória de usuário é executada na MMC.

Memória de carga ajustável (buffer FIFO)

Uma memória de carga é necessária na NCK de forma a processar o programa no modo “Execução externa” (programa principal ou subprograma).

A definição standard desta memória é de 30 Kbytes.

O tamanho da memória pode ser ajustada através do MD18360 EXT_PROG_BUFFER_SIZE.

POWER ON, RESET

O Reset e POWER ON interrompem a chamada externa de programas e a memória de carga associada é apagada.

Informações adicionais

Em subprogramas externos não é permitido o uso de comandos de salto como GOTO, GOTOB, CASE, IF - ELSE, FOR, LOOP, WHILE ou REPEAT. A chamada de subprogramas é possível.

Exemplo de programação

O dado setting \$SC_EXT_PROG_PATH contém a especificação do seguinte diretório:

"_N_WCS_DIR/_N_WPC1".

O subprograma principal _N_MAIN_MPF encontra-se na memória do usuário e selecionado.

```
%_N_MACHINE1_MPF
```

```
N10 PROC MAIN
```

```
N20 ...
```

```
N30 EXTCALL ROUGHING_SPF ; Chamada do programa externo
                           ROUGHING_SPF
```

```
N40 ...
```

```
N50 M30
```

2.12 Ciclos: Carregando parâmetros em ciclos de usuário



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Subprograma ROUGHING_SPF (localizado no subdiretório de peça da MMC ->WST1):

N10 PROC ROUGHING

N20 G1 F1000

N30 X=... Y=... Z=...

N40 ...

N90 M17

2.12 Ciclos: Carregando parâmetros em ciclos de usuário

Arquivos e diretórios



Explicação

conv.com Relação de ciclos

uc.com Descrição da chamada dos ciclos



Função

Ciclos definidos pelo usuário podem ser parametrizados através destes arquivos.



Seqüência

O arquivo conv.com é fornecido com os ciclos standard e pode ser ampliado de acordo com a necessidade. O arquivo uc.com deve ser criado pelo usuário.

Ambos arquivos devem ser carregados no sistema de arquivos passivo no diretório "Ciclos de usuário" (ou o diretório deve ser especificado no programa:

; \$PATH=/_N_CST_DIR).

2.12 Ciclos: Carregando parâmetros em ciclos de usuário840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Adaptando cov.com – Relação de ciclos

| | |
|-------------------------------------|----------------------------|
| %_N_COV_COM | Nome do arquivo |
| ;\$PATH=/_N_CUS_DIR | Especificação do diretório |
| ;Vxxx 11.12.95 Descrição do ciclo | Linha de comentário |
| C1(CYCLE81) Furação, centragem | Chamada do 1.ciclo |
| C2(CYCLE82) Furação, contra furação | Chamada do 2.ciclo |
| ... | |
| C24(CYCLE98) Encadeamento de roscas | Chamada do último ciclo |
| M17 | Fim de arquivo |

Para cada novo ciclo adicionado uma linha deve ser acrescentada com a seguinte sintaxe:

C<Número> (<Nome do ciclo>) comentário

Número: Qualquer número inteiro, desde que não utilizado anteriormente no arquivo;

Nome do ciclo: O nome do programa do ciclo deve ser incluído

Texto de comentário: texto opcional de comentário para o ciclo

Exemplo:

```
C25 (MY_CYCLE_1) ciclo_de_usuario_1
C26 (SPECIALCYCLE)
```

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Exemplo de arquivo uc.com –**Descrição do ciclo de usuário**

A explicação é baseada na continuação do exemplo:

Exemplo:

Para os ciclos deve ser criada a seguinte parametrização:

```
PROC MY_CYCLE_1 (REAL PAR1, INT PAR2, CHAR PAR3, STRING[10] PAR4)
```

```
;O ciclo possui os seguintes parâmetros de transferência:
```

```
;
```

```
;PAR1: Valor real na faixa -1000.001 <= PAR2 <= 123.456, default 100
```

```
;PAR2: Inteiro positivo com valor entre 0 <= PAR3 <= 999999,  
Default 0
```

```
;PAR3: 1 caracter ASCII
```

```
;PAR4: String de comprimento 10 para o nome de um subprograma
```

```
;
```

```
...
```

```
M17
```

```
PROC SPECIALCYCLE (REAL VALUE1, INT VALUE2)
```

```
;O ciclo possui os seguintes parâmetros de transferência:
```

```
;
```

```
;VALUE1: Valor real sem limitação ou default
```

```
;VALUE2: Valor inteiro sem limitação ou default
```

```
...
```

```
M17
```

2.12 Ciclos: Carregando parâmetros em ciclos de usuário840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Arquivo uc.com associado

%_N_UC_COM

; \$PATH=/_N_CUS_DIR

//C25(MY_CYCLE_1) ciclo_usuario_1

(R/-1000.001 123.456 / 100 /Parâmetro_2 do ciclo)

(I/0 999999 / 1 / valor inteiro)

(C/"A" / Parâmetro caracter)

(S///Nome do subprograma)

//C26(SPECIALCYCLE)

(R///Comprimento total)

(I/*123456/3/Tipo de usinagem)

M17

**Descrição da sintaxe do arquivo uc.com –
Descrição do ciclo de usuário****Cabeçalho para cada ciclo:**

Da mesma forma que no arquivo cov.com precedido por "///"

//C<Número> (<Nome do ciclo>) texto de comentário

Exemplo:

//C25(MY_CYCLE_1) ciclo_de_usuario_

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Linha para descrição de cada parâmetro

(<Identificação do tipo de dado> / <Valor mínimo> <Valor máximo>
(<valor default> / <Comentário>

Identificador do tipo de dado:

| | |
|---|----------------------------|
| R | Para real |
| I | Para inteiro |
| C | Para caracter (1 caracter) |
| S | Para string |

Valor mínimo. Valor máximo (podem ser omitidos)

Limitações nos valores digitados podem ser cheçadas; valores fora de faixa não podem ser carregados.

É possível especificar uma relação de valores a serem preenchidos através da tecla toggle; serão listados precedidos de “*”. Neste caso, outros valores quaisquer serão bloqueados.

Exemplo:

(I/*123456/1/tipo de usinagem)

Não existem limites para strings e tipos de caracteres;

Valor default (pode ser omitido)

Valor com o qual a tela será preenchida quando o ciclo for chamado; pode ser alterado pelo operador.

Comentário

Texto com até 50 caracteres que é exibido em frente do campo de parametrização na tela de chamada do ciclo.

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Exemplo de visualização para ambos ciclos

Tela de exibição para o ciclo MY_CYCLE_1

| | |
|----------------------|-----|
| Parâmetro 2 do ciclo | 100 |
| Valor inteiro | 1 |
| Parâmetro caracter | |
| Subprogramas | |

Tela de exibição para o ciclo SPECIAL CYCLE

| | |
|-------------------|-----|
| Comprimento total | 100 |
| Tipo de usinagem | 1 |

2.13 Macros

O que é uma macro?

Macro é uma seqüência de instruções individuais a qual é atribuída um nome próprio. Funções G, M, H ou nomes de subprogramas L podem também ser utilizados como macros.

Quando uma macro é chamada durante a execução de um programa, as instruções programadas sob o nome da macro são executadas uma após a outra.

Uso de macros

Seqüências de instruções que serão repetidas são programadas uma vez separadamente em um bloco macro no início do programa.

A macro então pode ser chamada no programa principal ou subrotina e executada

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Programação:

As macros são identificadas com a instrução

DEFINE . . . AS

A definição de uma macro é feita da seguinte forma:

DEFINE NAME AS <Instrução>

Exemplo:

Definição de macro:

DEFINE LINE AS G1 G94 F300

Chamada no programa NC

N20 LINE X10 Y20

Ativando macros

- Até a **SW 4**
Macros são ativadas após POWER ON.
- **SW 5** em diante
As macros são ativadas quando carregadas no NC (softkey “Carga”).

Funções M/G com três dígitos (a partir da SW 5)

- Até **SW 4**
Caso programada uma função M com 3 dígitos é disparado o alarme 12530.
- **SW 5** em diante
Suporta a programação de funções G e M com três dígitos.
Exemplo:
N20 DEFINE M100 AS M6
N80 DEFINE M999 AS M6



Informações adicionais

Não é permitido o encadeamento de macros.

Funções H e L com dois dígitos podem ser programadas.



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di



Exemplo de programação

| | |
|-------------------------|---|
| DEFINE M6 AS L6 | Na troca de ferramentas, uma subrotina é chamada para manuseio de dados. A função atual de troca de ferramentas será enviada internamente no subprograma (p.e. M106). |
| DEFINE G81 AS DRILL(81) | Emulação de funções G DIN |
| DEFINE G33 AS M333 G333 | Durante rosqueamento é solicitado um sincronismo com o PLC. A função G original G33 é renomeada para G333 através dos dados de máquina, de forma que a programação não é alterada para o usuário. |

Exemplo de um arquivo macro global:

Após a leitura do arquivo pelo controle, as macros são ativadas (vide acima). As macros podem então serem utilizadas pelo programa.

```
%_N_UMAC_DEF
; $PATH=/_N_DEF_DIR; macros do usuário
DEFINE PI AS 3.14
DEFINE TC1 AS M3 S1000
DEFINE M13 AS M3 M7 ;Fuso à direita, liga refrigeração
DEFINE M14 AS M4 M7 ;Fuso à esquerda, liga refrigeração
DEFINE M15 AS M5 M9 ;Desliga fuso, desliga refrigeração
DEFINE M6 AS L6 ;Chamada do programa para troca de ferramentas
DEFINE G80 AS MCALL ;Desliga ciclo de furação
M30 ;
```



- Os nomes das instruções são reservados e não devem ser redefinidos como macros.
- O uso de macros pode alterar significativamente a linguagem de programação do controle!
Portanto, é necessário cuidado no uso de macros.
- Macros podem também ser declaradas no programa NC. Para nomear macros é permitido o uso apenas de identificadores. Macros com funções G podem ser definidas somente no bloco de macros para todo o controle.
- Com as macros você pode definir quaisquer identificadores, tais como G, M, funções H e nomes de programa L.
- Identificadores de macro com 1 letra e 1 número são permitidos (**somente no FM-NC**).

Gerenciamento de programas e arquivos

| | | |
|-----|--|-------|
| 3.1 | Generalidades | 3-122 |
| 3.2 | Memória de programa | 3-123 |
| 3.3 | Memória de usuário | 3-128 |
| 3.4 | Definindo dados de usuário | 3-131 |
| 3.5 | Definindo níveis de proteção para dados de usuário (GUD) | 3-135 |
| 3.6 | Ativação automática das GUDs e MACs (SW 4.4 em diante) | 3-137 |

3.1 Generalidades



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

3.1 Generalidades



Estrutura de memória

A estrutura de memória disponível para o usuário encontra-se organizada em duas áreas.

1. Memória do usuário

A memória do usuário contém os dados de sistema e de usuário com os quais o controle trabalha (sistema de arquivos ativos).

Exemplo:

Dados de máquina ativos, deslocamentos de origem, corretores de ferramentas.

2. Memória de programa

Os arquivos e programas são gravados na memória de programa e lá permanecem gravados (sistema de arquivos passivos).

Exemplo:

Programas principais e subrotinas, definições de macros.

840D
NCU 571840D
NCU 572

FM-NC



810D



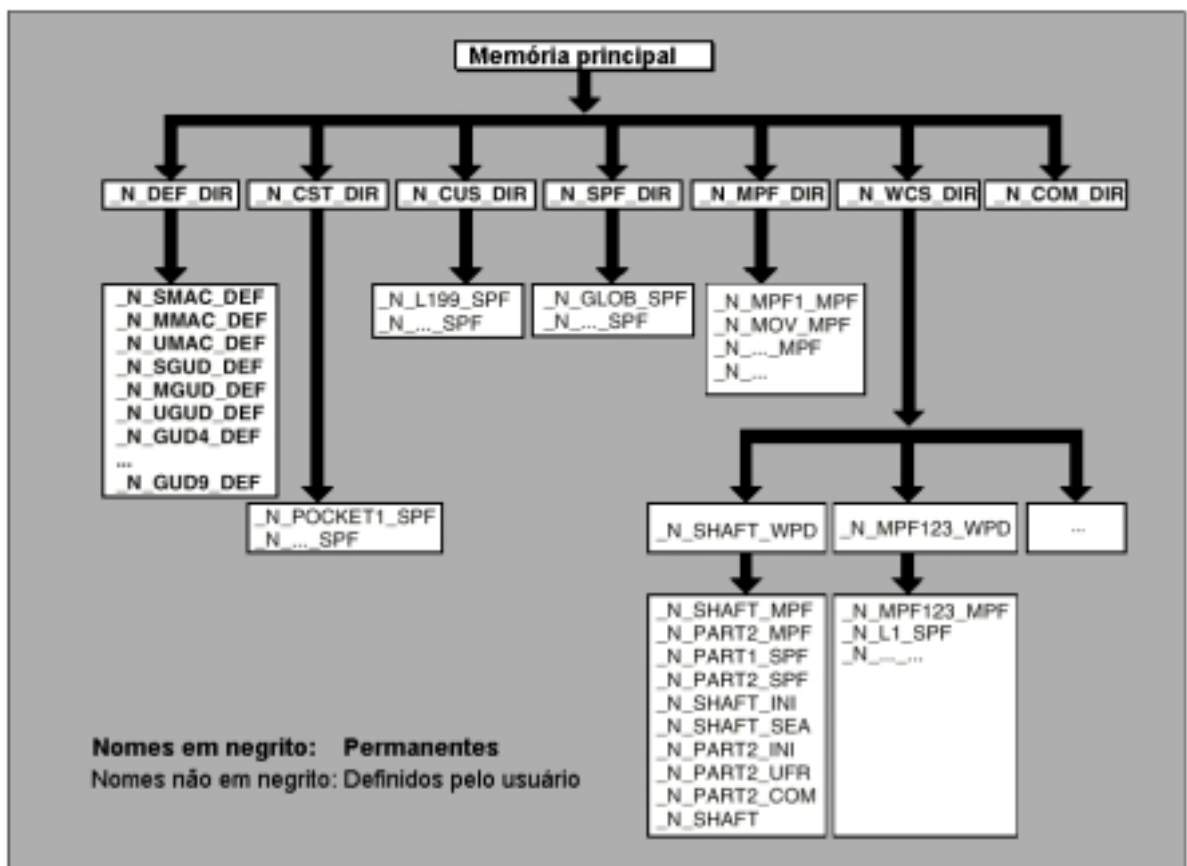
840Di

NCU 573

3.2 Memória de programa

Generalidades

Programas e subprogramas são gravados na memória principal. Outros tipos de arquivos são também gravados temporariamente na memória principal, podendo ser transferidos para a memória de trabalho quando necessário (p.e., para inicialização da máquina para uma peça específica).



3.2 Memória de programa



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Diretórios

Os seguintes diretórios são fornecidos quando um painel de operação estiver conectado:

| | |
|---------------|---|
| 1. _N_DEF_DIR | Módulos de dados e macros (definidos na colocação em funcionamento) |
| 2. _N_CST_DIR | Ciclos standard (definidos na colocação em funcionamento) |
| 3. _N_CUS_DIR | Ciclos do usuário (definidos na colocação em funcionamento) |
| 4. _N_WCS_DIR | Diretórios de peças |
| 5. _N_SPF_DIR | Subprogramas globais |
| 6. _N_MPF_DIR | Diretório standard para programas principais |
| 7. _N_COM_DIR | Diretório standard para comentários |

Tipos de arquivos

Os seguintes tipos de arquivos podem ser gravados na memória principal:

| | |
|------------------|---|
| <i>nome</i> _MPF | Programa principal |
| <i>nome</i> _SPF | Subprograma |
| <i>nome</i> _TEA | Dados de máquina |
| <i>nome</i> _SEA | Dados de ajuste (setting) |
| <i>nome</i> _TOA | Corretores de ferramenta |
| <i>nome</i> _UFR | Deslocamento de origem |
| <i>nome</i> _INI | Arquivo de inicialização |
| <i>nome</i> _GUD | Dados globais de usuário |
| <i>nome</i> _RPA | Parâmetros R |
| <i>nome</i> _COM | Comentários |
| <i>nome</i> _DEF | Definições para dados globais de usuário e macros |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Diretório de peças, `_N_WCS_DIR`

O diretório de peças é definido de forma standard sob o nome

`_N_WCS_DIR.`

O diretório de peças principal contém todos os outros diretórios de peças para as peças programadas.

Diretório de peças, identificador WPD

Para tornar o manuseio de dados e programas mais flexíveis podemos fazer agrupamentos ou gravá-los em diretórios de peça individuais.

Um diretório de peças contém todos os arquivos necessários para a usinagem de uma peça

Isto pode incluir programas principais, subprogramas, quaisquer programas de inicialização e arquivos de comentários.

Exemplo:

O diretório de peças `_N_SHAFT_WPD`, criado para a peça SHAFT contém os seguintes arquivos:

| | |
|---------------------------|---|
| <code>_N_SHAFT_MPF</code> | Programa principal |
| <code>_N_PART2_MPF</code> | Programa principal |
| <code>_N_PART1_SPF</code> | Subprograma |
| <code>_N_PART2_SPF</code> | Subprograma |
| <code>_N_SHAFT_INI</code> | Programa de inicialização geral para os dados da peça |
| <code>_N_SHAFT_SEA</code> | Programa de inicialização dos dados de ajuste (Setting) |
| <code>_N_PART2_INI</code> | Programa de inicialização geral para os dados do programa PART2 |
| <code>_N_PART2_UFR</code> | Programa de inicialização para os dados frame do programa PART2 |
| <code>_N_SHAFT_COM</code> | Arquivo de comentários |

3.2 Memória de programa



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Criando diretórios de peças em um PC externo

Os passos descritos abaixo são executados em uma estação de dados externa.

Favor consultar o Manual de operação para detalhes sobre o gerenciamento de programas e arquivos (de um PC para o controle) no controle.

Instrução ;\$PATH

O diretório de destino \$PATH= . . . é especificado na segunda linha do arquivo.

Exemplo:

```
; $PATH=/_N_WCS_DIR/_N_SHAFT_WPD
```

O arquivo será gravado no diretório específico.

Importante

Caso o diretório não tenha sido especificado, os arquivos do tipo SPF são gravados em /_N_SPF_DIR, arquivos com extensão _INI na memória de trabalho e outros arquivos em /_N_MPF_DIR.

Exemplo com diretório especificado para o exemplo anterior SHAFT:

```
%_N_SHAFT_MPF
; $PATH=/_N_WCS_DIR/_N_SHAFT_WPD
N10 G0 X... Z...
•
M2
•
•
%_N_SHAFT_SPF
```


840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

Seleção de uma peça para usinagem

Um diretório de peças pode ser selecionado para execução em um canal.

Caso um programa com o **mesmo nome do diretório** esteja nele gravado, este será automaticamente selecionado para execução.

Exemplo:

O diretório de peças

/ _N_WCS_DIR/ _N_SHAFT_WPD contém os arquivos

_N_SHAFT_SPF e _N_SHAFT_MPF.

SW 5 em diante (somente MMC 102/103):

Vide o "Manual de operação " /BA/ seção Lista de tarefas e Seleção de programas.

Pesquisa de diretório na chamada de subprogramas

Caso o diretório não seja especificado de forma explícita no programa de usinagem quando uma subrotina ou arquivo de inicialização forem chamados, o programa irá procurá-los em um diretório específico.

Exemplo da chamada de um subprograma com especificação absoluta de diretório:

CALL " / _N_CST_DIR/ _N_CYCLE1_SPF "

Os programas são geralmente chamados sem a especificação do diretório:

Exemplo:

CYCLE1

Seqüência de pesquisa de diretório

1. Diretório atual / *nome*

Diretório de peças diretório
standard

_N_MPF_DIR

2. Diretório atual / *nome* _SPF

3. Diretório atual / *nome* _MPF

4. / _N_SPF_DIR / *nome* _SPF

Subprogramas globais

5. / _N_CUS_DIR / *nome* _SPF

Ciclos de usuário

6. / _N_CST_DIR / *nome* _SPF

Ciclos standard

3.3 Memória de usuário



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

3.3 Memória de usuário

Programas de inicialização

Estes são os programas com os quais os dados da memória de trabalho são inicializados).

Os seguintes tipos de arquivo podem ser utilizados:

| | |
|------------------|---|
| <i>Nome</i> _TEA | Dados de máquina |
| <i>Nome</i> _SEA | Dados de ajuste (Setting) |
| <i>Nome</i> _TOA | Corretores de ferramentas |
| <i>Nome</i> _UFR | Deslocamentos de origem/frames |
| <i>Nome</i> _INI | Arquivos de inicialização |
| <i>Nome</i> _GUD | Dados globais de usuário (Global user data) |
| <i>Nome</i> _RPA | Parâmetros R |

Áreas de dados

Os dados podem ser organizados em áreas diferentes daquelas que serão utilizados. Por exemplo, um controle pode utilizar diversos canais (não o SINUMERIK FM-NC, 810D CCU1, 840D NCU 571) e vários eixos. As seguintes áreas encontram-se disponíveis

| Identificador | Área de dados |
|---------------|---|
| NCK | Dados específicos NCK |
| CH<n> | Dados específicos de canal (n especifica o número do canal) |
| AX<n> | Dados específicos do eixo (n especifica o número do eixo de usinagem) |
| TO | Dados de ferramenta |
| COMPLETE | Todos os dados |

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

Gerando um programa de inicialização em um PC externo PC

Os identificadores de área e tipo de dados podem ser utilizados para determinar as áreas as quais devem ser tratadas como uma unidade quando os dados forem gravados.

Exemplo:

| | |
|----------------------------------|--------------------------------|
| <code>_N_AX5_TEA_INI</code> | Dados de máquina para o eixo 5 |
| <code>_N_CH2_UFR_INI</code> | Frames do canal 2 |
| <code>_N_COMPLETE_TEA_INI</code> | Todos os dados de máquina |

Quando o controle é inicialmente configurado, um conjunto de dados é automaticamente carregado para garantir a operação correta do controle.

Gravando programas de inicialização

Os arquivos da memória de trabalho podem ser gravados em um PC externo (back up) e recarregados quando necessário.

- Os arquivos são gravados com `COMPLETE`.
- um arquivo INI : `INITIAL` pode ser criado para todas áreas com `_N_INITIAL_INI`.

Carregando programas de inicialização

Programas INI também podem ser chamados como programas de usinagem caso utilizem dados de somente um canal. Com isto torna-se possível inicializar dados controlados pelo programa.



Informações sobre os tipos de arquivos são fornecidas no Manual de operação.

3.3 Memória de usuário



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Procedimento para controles multi canais

CHANDATA (Número do canal) para vários canais é somente permitido no arquivo N_INITIAL_INI. N_INITIAL_INI é o arquivo de instalação com o qual todos os dados do controle são inicializados.

Exemplo:

```
%_N_INITIAL_INI
CHANDATA(1)
;Definição dos eixos do canal 1
$MC_AXCONF_MACHAX_USED[0]=1
$MC_AXCONF_MACHAX_USED[1]=2
$MC_AXCONF_MACHAX_USED[2]=3
CHANDATA(2)
;Definição dos eixos do canal 2
$MC_AXCONF_MACHAX_USED[0]=4
$MC_AXCONF_MACHAX_USED[1]=5
CHANDATA(1)
;Dados de máquina dos eixos
;Janela de parada exata - grossa:
$MA_STOP_LIMIT_COARSE[AX1]=0.2 ;Eixo 1
$MA_STOP_LIMIT_COARSE[AX2]=0.2 ;Eixo 2
;Janela de parada exata - fina:
$MA_STOP_LIMIT_COARSE[AX1]=0.01 ;Eixo 1
$MA_STOP_LIMIT_COARSE[AX1]=0.01 ;Eixo 2
```



*No programa de usinagem, a instrução CHANDATA pode somente ser utilizada para os dados do canal onde o programa encontra-se rodando, ou seja, a instrução pode proteger programas de serem utilizados em canais errados acidentalmente.
O processamento do programa é interrompido caso ocorram erros..*



Informação

Arquivos INI em listas de tarefas (job lists) não podem conter instruções CHANDATA.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

3.4 Definindo dados de usuário



Função



Os dados de usuário (GUD) são definidos durante o procedimento de colocação em funcionamento do comando .

Os dados de máquina necessários são devem ser preenchidos de forma adequada.

A memória de usuário deve estar configurada, e esta configuração deve ser definida no arquivo %_N_INITIAL_INI o qual é carregado após o arquivo de definições. Todos os dados de máquina relacionados possuem em seus nomes o componente GUD .

- **SW 5 em diante (01.99):**

Os dados de usuário (GUD) podem ser definidos na área de operação Serviços. Desta forma, procedimentos de reimportação demorados de arquivos de back up (%_N_INITIAL_INI) não são necessários.

Os seguintes princípios são aplicados:

- Arquivos de definição existentes no disco rígido não se encontram ativos.
- Arquivos de definição existentes na memória de NC estão sempre ativos.

Nomes reservados

Os seguintes módulos podem ser gravados no diretório / _N_DEF_DIR:

| | |
|-------------|--|
| _N_SMAC_DEF | Contém definição de macros (Siemens, nível de proteção 0) |
| _N_MMAC_DEF | Contém definição de macros (fabricante da máquina, nível de proteção 2) |
| _N_UMAC_DEF | Contém definição de macros (usuário, nível de proteção 3) |
| _N_SGUD_DEF | Contém definições p/dados globais (Siemens, nível de proteção 0) |
| _N_MGUD_DEF | Contém definições p/dados globais (fabricante da máquina, nível de proteção 2) |
| _N_UGUD_DEF | Contém definições p/dados globais (usuário nível de proteção 3) |
| _N_GUD4_DEF | Livre definição |
| _N_GUD5_DEF | Contém definições para ciclos de medição (Siemens, nível de proteção 0) |
| _N_GUD6_DEF | Contém definições para ciclos de medição (Siemens, nível de proteção 0) |

3.4 Definindo dados de usuário



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

| | |
|--------------------------|--|
| <code>_N_GUD7_DEF</code> | Contém definições para os ciclos standard (Siemens, nível de proteção 0) ou livre definição caso não utilizados os ciclos standard |
| <code>_N_GUD8_DEF</code> | Livre definição |
| <code>_N_GUD9_DEF</code> | Livre definição |

O nível de acesso necessário é determinado no arquivo de definição através do comando APR ou APW.

Quando um arquivo de definições GUD é ativado pela primeira vez, qualquer nível de acesso nele definido é analisado e automaticamente transferido para as características de leitura/escrita do próprio arquivo de definições.



Informação

Níveis de acesso no arquivo de definições GUD podem restringir e não ampliar o nível de acesso do arquivo de definições.

Exemplo

O arquivo de definições `_N_GUD7_DEF` contém:
APW2

- Caso o arquivo `_N_GUD7_DEF` possuir valor 3 como proteção de escrita. O valor 3 é então substituído pelo valor 2.
- Caso o arquivo `_N_GUD7_DEF` possuir valor 0 como nível de proteção de escrita, não serão realizadas alterações.

A instrução APW faz uma alteração retrospectiva no nível de acesso de escrita do arquivo.



Informação

Caso você tenha tentado acessar de forma errada um arquivo de definição GUD (com um nível de proteção superior ao que você possui), o arquivo de back up deve ser novamente carregado.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

Definindo dados de usuário (GUD)

1. Gravar o arquivo `_N_INITIAL.INI`.
2. Criar um arquivo de definições para os dados de usuário:

- Em computador externo PC (até **SW 4**)
- Na área de operação services (**SW 5** em diante)

Existem nomes pré definidos (vide página anterior):

```
_N_SGUD_DEF
_N_MGUD_DEF
_N_UGUD_DEF
_N_GUD4_DEF ... _N_GUD9_DEF
```

Arquivos com estes nomes podem conter definições para as variáveis GUDs.

Um atributo adicional é necessário para identificar a variável como sendo uma GUD e para definir a área em que esta GUD será válida:

```
NCK, CHAN.
```

Uma parada de pré processamento implícita pode também ser definida para quando as variáveis forem lidas/escritas em um estágio posterior:

```
SYNR: Parada de pré processamento durante
      leitura
SYNRW: Parada de pré processamento
       durante leitura/escrita.
```

3. Carregar o arquivo de definições na memória do controle.

O controle sempre cria o diretório `_N_DEF_DIR`.

O nome é carregado do mesmo modo que a especificação de diretório no cabeçalho do arquivo de definição GUD e considerado quando lido através da interface V.24.

Exemplo de um arquivo de definições, dados globais (Siemens):

```
%_N_SGUD_DEF
; $PATH=/_N_DEF_DIR
DEF NCK REAL RTP                                ;Plano de recuo
DEF CHAN INT SDIS                               ;Folga de segurança
M30
```

3.4 Definindo dados de usuário



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

4. Ativando arquivos de definição

- Até a **SW4**

O arquivo de definições somente era ativado após a carga de `_N_INITIAL_INI`.

- **SW 5** em diante

Quando um arquivo GUD é carregado no NC, é tornado ativo imediatamente.



Grave todos os programas, Frames e dados de máquina antes da carga de `_N_INITIAL_INI` pois a memória estática do comando será formatada.

5. Gravando dados

Quando o arquivo `_N_COMPLETE_GUD` for criado a partir da memória de trabalho, somente os dados contidos no arquivo são gravados. Os arquivos de definição criados para as variáveis de usuário global devem ser gravados separadamente.

O valor das variáveis gravadas nos dados de usuário globais são também gravadas em `_N_INITIAL_INI`, os nomes devem ser idênticos aos nomes nos arquivos de definição.

Exemplo de um arquivo de definição para dados globais (fabricante da máquina):

```
%_N_MGUD_DEF
; $PATH=/_N_DEF_DIR
; Definição dos dados globais do fabricante
DEF NCK SYN RW INT QUANTITY ; Parada de pré processamento implícita durante
                               leitura/escrita
                               ; Dado especificado disponível no controle
                               ; Acesso para todos os canais
DEF CHAN INT TOOLTABLE[100] ; Tabela de ferramentas específica para o canal
                              ; Imagem dos números das ferramentas do magazine
M30 ; Para cada canal é gerada uma tabela própria
```


3.5 Definindo níveis de proteção para dados de usuário (GUD)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

3.5 Definindo níveis de proteção para dados de usuário (GUD)



Explicação

| | |
|-------|---|
| APR n | Proteção para acesso de leitura |
| APW n | Proteção para acesso de escrita |
| n | Nível de proteção de 0/10 (mais alto) até 7/17 (mais baixo) |

APW 0-7, APR 0-7:

O módulo de variáveis não pode ser escrito ou lido através do programa NC ou no modo MDA.

APW 10-17, APR 10-17:

O módulo de variáveis pode ser escrito ou lido através do programa NC ou no modo MDA.

Níveis de proteção

0/10 = SIEMENS

1/11 = OEM_ HIGH

2/12 = OEM _LOW

3/13 = usuário final

4/14 to 7/17 = chave na posição 3 à 0



Nota

A seqüência de digitação do comando é a seguinte
APR.. APW..

Qualquer outra seqüência representa um erro de sintaxe.

Para que todo o arquivo seja protegido, o comando deve ser programado na primeira linha do arquivo!

3.5 Definindo níveis de proteção para dados de usuário (GUD)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Função

Critérios de acesso podem ser definidos para módulos GUD. Com o uso dos critérios é possível, por exemplo, inibir alterações em ciclos que o fabricante da máquina tenha definido como módulos GUDs.

A proteção de acesso é aplicada para todas as variáveis definidas neste módulo.

Caso feita tentativa de acesso à área protegida, o controle emite um alarme apropriado.



Seqüência

O nível de proteção de acesso é programado no módulo antes da definição de qualquer variável. As instruções devem ser programadas em blocos separados.

Exemplo: definição de nível de proteção de acesso (fabricante da máquina).

```
%_N_GUD6_DEF
```

```
; $PATH=/_N_DEF_DIR
```

```
APR 5 APW 2
```

```
;Exibição/leitura somente com chave na  
posição 2
```

```
;Escrita com nível de proteção  
OEM_LOW
```

```
;Cuidado! Com este valor programado o  
arquivo/programa pode ele mesmo  
alterar seu nível de acesso (vide acima)
```

```
DEF CHAN REAL_CORRVAL
```

```
...
```

```
M30
```

```
;
```

3.6 Ativação automática das GUDs e MACs (SW 4.4 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

3.6 Ativação automática das GUDs e MACs (SW 4.4 em diante)



Função

Os arquivos para definições de macros e GUDs são editados na

- Área de operação Serviços da MMC 102/103.

Quando editado um arquivo de definições, no instante em que o editor estiver sendo abandonado será emitida uma mensagem perguntando se as definições daquele arquivo devem ser ativadas.

Exemplo:

"Você deseja ativar as definições do arquivo GUD7.DEF?"

"OK" → Um outro aviso será emitido, perguntando se os dados atuais do arquivo de definições devem ser salvos:

"Você deseja salvar os dados anteriormente existentes nas definições?"

"OK" → Os blocos GUD do arquivo de definições editado serão salvos, as novas definições ativadas e os dados salvos serão novamente importados.

"Cancela" → As novas definições são ativadas e os antigos dados zerados.

"Cancela" → As alterações no arquivo de definições serão descartadas, o bloco de dados associado não é alterado.

Descarregar

Caso um arquivo de definições seja descarregado, o bloco de dados associado é apagado após a exibição da pergunta.

3.6 Ativação automática das GUDs e MACs (SW 4.4 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Carga

Caso um arquivo de definições seja carregado, um aviso será emitido, para confirmar se o arquivo deve ser ativado e os dados mantidos. Caso não seja ativado, o arquivo não será carregado.

Caso o cursor esteja posicionado em um arquivo de definições já carregado, a identificação das softkeys é comutada de “Carregar” para “Ativar”, permitindo que as definições sejam ativadas. Caso você selecione “Ativar”, outro aviso será emitido perguntando se você deseja manter os dados.



Os dados podem ser salvos somente para os arquivos de definição de variáveis, e não para macros.



Informações adicionais (MMC 103)

Caso não exista capacidade de memória suficiente para a ativação do arquivo de definições, pois o tamanho de memória foi alterado, o arquivo deve ser transferido do NC para a MMC e novamente para o NC para que seja ativado.

Zonas de proteção

- 4.1 Definindo zonas de proteção CPROTDEF, NPROTDEF4-140
- 4.2 Ativando/desativando as zonas de proteção: CPROT, NPROT.....4-144

4.1 Definindo zonas de proteção CPROTDEF, NPROTDEF



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

4.1 Definindo zonas de proteção CPROTDEF, NPROTDEF



Programação

```
DEF INT NOT_USED
CPROTDEF(n,t,applim,applus,appminus)
NPROTDEF(n,t,applim,applus,appminus)
EXECUTE (NOT_USED)
```



Explicação dos comandos

| | |
|------------------|---|
| DEF INT NOT_USED | Aloca variável, define tipo de dado como sendo inteiro (vide capítulo 10) |
| CPROTDEF | Define zona de proteção do canal (somente para NCU 572/573) |
| NPROTDEF | Define a zona de proteção da máquina |
| EXECUTE | Fim da definição |



Explicação dos parâmetros

| | |
|----------|--|
| n | Quantidade de zonas de proteção |
| t | TRUE = Zona de proteção de acordo com a ferramenta FALSE = Zona de proteção de acordo com a peça |
| applim | Tipo do limite na terceira dimensão 0 = Sem limite 1 = Limite na direção positiva 2 = Limite na direção negativa 3 = Limite na direção positiva e negativa |
| applus | Valor limite na direção positiva na terceira dimensão |
| appminus | Valor limite na direção negativa na terceira dimensão |
| NOT_USED | Variável de erro não possui efeito nas zonas de proteção com EXECUTE |

4.1 Definindo zonas de proteção CPROTDEF, NPROTDEF



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Função

Você pode utilizar as zonas de proteção para proteger vários elementos da máquina bem como a peça, contra movimentos incorretos.

Zonas de proteção de acordo com a **ferramenta**:

Para os componentes pertencentes às ferramentas (p.e. ferramenta, magazine de ferramentas).

Zonas de proteção de acordo com a **peça**

Para partes da máquina relacionadas à peça (p.e. partes da peça, mesa de fixação, dispositivo de fixação, fuso).



Seqüência

Definindo zonas de proteção

A definição das zonas de proteção inclui o seguinte:

- CPROTDEF para zonas de proteção específicas do canal
- NPROTDEF para zonas de proteção específicas da máquina
- Descrição do contorno da zona de proteção
- Encerramento da definição com EXECUTE



Você pode especificar um deslocamento de origem para o ponto de referência de uma zona de proteção quando esta for ativada no programa de NC.

4.1 Definindo zonas de proteção CPROTDEF, NPROTDEF



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Ponto de referência para a descrição do contorno

As zonas de proteção relacionadas à peça são definidas a partir do sistema de coordenadas básico. As zonas de proteção relacionadas às ferramentas são definidas com relação ao ponto F do porta ferramentas.

Definição do contorno das zonas de proteção

O contorno das zonas de proteção é especificado por até 11 movimentos no plano selecionado. O primeiro movimento é o de aproximação do contorno. Os comandos de movimento programados entre CPROTDEF ou NPROTDEF e EXECUTE não serão executados, servem somente para definir a zona de proteção.

Plano de trabalho

O plano de trabalho é selecionado antes de CPROTDEF e NPROTDEF através de G17, G18 ou G19 e não deve ser alterado antes da instrução EXECUTE. A aplicação não deve ser programada entre CPROTDEF ou NPROTDEF e EXECUTE.

Elementos de contorno

É permitido o seguinte:

- G0, G1 para elementos de contorno lineares
- G2 arcos de circunferências no sentido horário (somente para zonas de proteção relacionadas às ferramentas)
- G3 para arcos de circunferência no sentido anti horário

Um máximo de 4 elementos de contorno estão disponíveis para definir uma zona de proteção (máximo de 4 zonas) com o SINUMERIK FM-NC. Com o 810D, um máximo de 4 elementos de contorno são disponíveis para definir uma zona de proteção (máximo de 4 específicos para canal e 4 para NCK).

4.1 Definindo zonas de proteção CPROTDEF, NPROTDEF



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Caso um círculo completo especifique uma zona, ele deve ser dividido em dois semicírculos. A ordem de G2, G3 ou G3, G2 não é permitida. Um pequeno bloco em G1 deve ser inserido, se necessário.

O último ponto da descrição de contorno deve coincidir com o primeiro.

Zonas de proteção externas (possíveis somente para zonas de proteção relacionadas à peça) devem ser definidas no **sentido horário**.

Para zonas de proteção balanceadas dinamicamente (p.e. fusos) você deve descrever o contorno completo (e não somente o centro da rotação!).

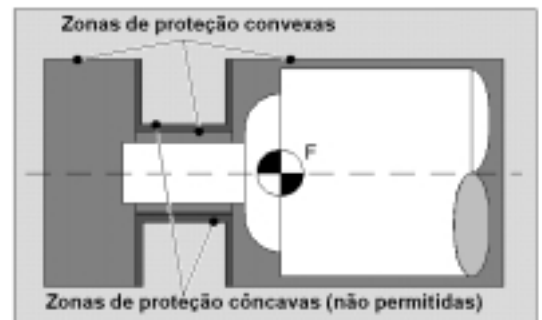
As zonas de proteção relacionadas à ferramenta devem sempre ser convexas. Caso uma zona de proteção **côncava** seja desejada, esta deve ser subdividida em várias zonas de proteção convexas.



Não devem ser ativados os seguintes comandos durante a definição das zonas de proteção:

- Raio de corte ou a correção do raio da ferramenta,
- Transformações,
- Frame.

Não devem ser também ativados o referenciamento (G74), aproximação à ponto fixo (G75), parada na pesquisa de bloco ou fim de programa.



4. Erro! Apenas o documento principal. Erro! Apenas o documento principal.840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

4.2 Ativando/desativando as zonas de proteção: CPROT, NPROT**Programação**CPROT *n, state, xMov, yMov, zMov*NPROT (*n, state, xMov, yMov, zMov*)**Explicação dos comandos e parâmetros**

| | |
|-------------------------|--|
| CPROT | Chamada das zonas de proteção específicas do canal (somente para NCU 572/573) |
| NPROT | Chamada de zona de proteção específica da máquina |
| <i>n</i> | Número da zona de proteção |
| <i>state</i> | Parâmetro de estado 0 = Zona de proteção desativada 1 = Zona de proteção pré ativa 2 = Zona de proteção ativada |
| <i>xMov, yMov, zMov</i> | Movimenta a zona de proteção ativa com os eixos geométricos |

**Função**

Ativando, desativando as zonas de proteção para monitoração de colisão.

O número máximo de zonas de proteção que podem ser ativadas simultaneamente em um mesmo canal é definida através de dados de máquina.

Caso não existam zonas de proteção relativas às ferramentas, o percurso da ferramenta é checado de acordo com as zonas de proteção específicas da peça.



Caso não existam zonas de proteção relativas à peça ativa, a monitoração não será ativada.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573



Seqüência

Estado de ativação

Uma zona de proteção é geralmente ativada no programa com estado=2.

O estado é sempre específico do canal, mesmo para as zonas de proteção da máquina.

Caso um programa de usuário de PLC ative uma zona de proteção, esta ativação pode ocorrer somente com o estado =1.

As zonas de proteção são desativadas e portanto desabilitadas caso o estado seja 0. Não são necessários corretores.

Ativando deslocamentos de origem em zonas de proteção (pré) ativação

Os deslocamentos podem ocorrer em 1, 2 ou 3 dimensões.

O deslocamento de origem se refere à:

- Zero máquina em zonas específicas da peça,
- Referência do porta ferramenta (F) na zona de proteção relativa à ferramenta.



Informações adicionais

As zonas de proteção podem ser ativadas logo após a ligação do sistema (boot) mais o referenciamento dos eixos. A variável de sistema \$SN_PA_ACTIV_IMMED [n] ou \$SN_PA_ACTIV_IMMED[n] = TRUE deve ser ligada para este propósito. São sempre ativadas com estado =2 e não possuem deslocamentos de origem.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Ativação de várias zonas de proteção

Uma zona de proteção pode ser ativada simultaneamente em vários canais (p.e. dispositivo onde existam dois lados opostos).

As zonas de proteção são monitoradas somente após os eixos geométricos terem sido referenciados. As seguintes regras são aplicadas:

- A zona de proteção não pode ser ativada simultaneamente com diferentes deslocamentos em um único canal.
- As zonas de proteção relativas à máquina devem possuir a mesma orientação em ambos os canais.



Exemplo de programação

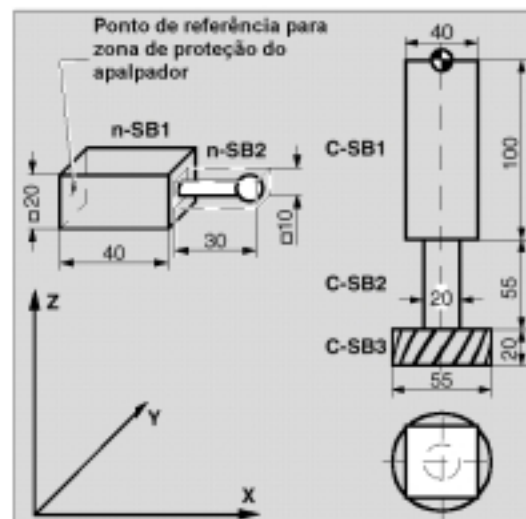
A possível colisão de uma fresa com a ponta de medição deve ser monitorada em uma fresadora. A posição da ponta de medição deve ser definida por um deslocamento de origem quando a função for ativada.

As seguintes zonas de proteção são definidas para este propósito:

- Uma zona específica para a máquina e relativa à peça tanto para o suporte do apalpador (n-SB1) quanto para o apalpador (n-SB2) em si.
- Uma zona de proteção específica do canal e orientada de acordo com a ferramenta para o suporte da fresa (c-SB1), para a haste (c-SB2) e para a fresa (c-SB3).

Todas as zonas de proteção são orientadas no sentido Z.

A posição do ponto de referência da ponta de medição para a ativação da função deve ser $X = -120$, $Y = 60$ e $Z = 80$.



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

DEF INT PROTECTB

Cria a variável auxiliar

Definindo a zona de proteção

Define a orientação

G17

NPROTDEF(1,FALSE,3,10,-10)

Zona de proteção n-SB1

G01 X0 Y-10

X40

Y10

X0

Y-10

EXECUTE(PROTECTB)

NPROTDEF(2,FALSE,3,5,-5)

Zona de proteção n-SB2

G01 X40 Y-5

X70

Y5

X40

Y-5

EXECUTE(PROTECTB)

CPROTDEF(1,TRUE,3,0,-100)

Zona de proteção c-SB1

G01 X-20 Y-20

X20

Y20

X-20

Y-20

EXECUTE(PROTECTB)

CPROTDEF(2,TRUE,3,-100,-150)

Zona de proteção c-SB2

G01 X0 Y-10

G03 X0 Y10 J10

X0 Y-10 J-10

EXECUTE(PROTECTB)

CPROTDEF(3,TRUE,3,-150,-170)

Zona de proteção c-SB3

G01 X0 Y-27,5

G03 X0 Y27,5 J27,5

X0 Y27,5 J-27,5

EXECUTE(PROTECTB)

Ativando as zonas de proteção:

NPROT(1,2,-120,60,80)

Ativa a zona de proteção n-SB1 c/desl.de origem

NPROT(2,2,-120,60,80)

Ativa a zona de proteção n-SB2 c/desl.de origem

CPROT(1,2,0,0,0)

Ativa a zona de proteção c-SB1 c/desl.de origem

CPROT(2,2,0,0,0)

Ativa a zona de proteção c-SB2 c/desl.de origem

CPROT(3,2,0,0,0)

Ativa a zona de proteção c-SB3 c/desl.de origem



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Notas

Comandos especiais de movimento

| | | |
|------|--|-------|
| 5.1 | Posicionando em pontos codificados, CAC, CIC, CDC, CACP, CACN | 5-150 |
| 5.2 | Interpolação Spline | 5-151 |
| 5.3 | Compressor COMPON/COMPCURV | 5-160 |
| 5.4 | Interpolação polinomial, POLY | 5-163 |
| 5.5 | Referência de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante) | 5-169 |
| 5.6 | Medições através de pontas de prova, MEAS, MEAW | 5-174 |
| 5.7 | Funções adicionais de medição MEASA, MEAWA, MEAC (SW 4 em diante, opção) | 5-177 |
| 5.8 | Funções especiais para usuários OEM | 5-187 |
| 5.9 | Critérios programáveis para fim de movimentação (SW 5.1 em diante) | 5-188 |
| 5.10 | Bloco de parâmetros servo programável (SW 5.1 em diante) | 5-189 |

5.1 Posicionando em pontos cod., CAC, CIC, CDC, CACP, CACN



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

5.1 Posicionando em pontos codificados, CAC, CIC, CDC, CACP, CACN



Explicação dos comandos

| | |
|----------|---|
| CAC (n) | Posiciona em posições codificadas de forma absoluta |
| CIC (n) | Posiciona em pontos codificados de forma incremental, por n espaços no sentido positivo ou negativo |
| CDC (n) | Posiciona no ponto codificado através do menor percurso (somente para eixos rotativos) |
| CACP (n) | Posiciona o eixo codificado de forma absoluta no sentido positivo (somente para eixos rotativos) |
| CACN (n) | Posiciona o eixo codificado de forma absoluta no sentido negativo (somente para eixos rotativos) |
| (n) | Número das posições: 1, 2, ... máximo de 60 posições por eixo |



Seqüência

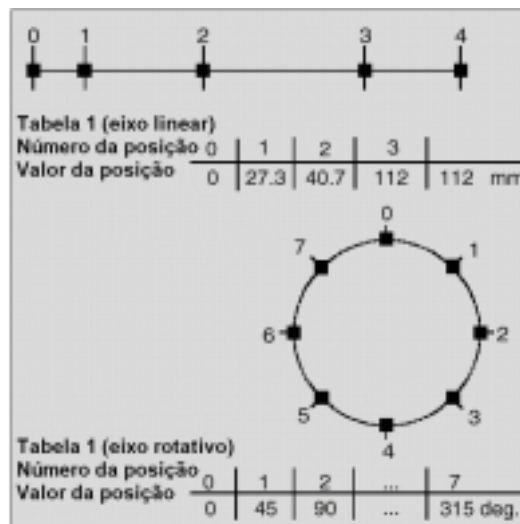
Em dados de máquina você pode especificar um máximo de 60 (0 à 59) posições em tabelas especiais para 2 eixos.

Para um exemplo típico de tabela de posição vide o diagrama.



Detalhes adicionais

Caso um eixo esteja situado entre duas posições, ele não se movimentará em resposta a um comando de posição incremental com CIC (...). É sempre aconselhável a programação de do primeiro comando de movimento com um valor absoluto de posição.



Exemplo de programação

| | |
|----------------------|---|
| N10 FA[B]= 300 | Velocidade para posicionar o eixo B |
| N20 POS[B]= CAC (10) | Movimenta até a posição codificada 10 (de forma absoluta) |
| N30 POS[B]= CIC (-4) | Movimenta 4 espaços para trás da posição atual |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

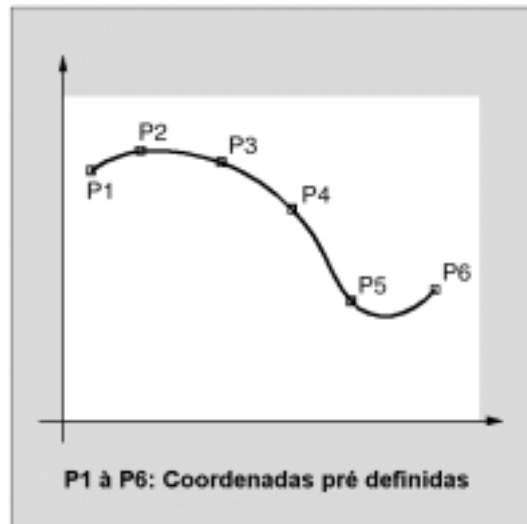
5.2 Interpolação Spline



Introdução

A interpolação spline pode ser utilizada para conectar uma série de pontos ao longo de uma curva. Splines podem ser aplicados, p.e., para criar curvas a partir de pontos digitalizados.

Existem vários tipos de spline com diferentes características, cada uma produzindo diferentes efeitos de interpolação. Além de selecionar o tipo de spline, o usuário pode manipular uma faixa de parâmetros. Normalmente são necessárias várias tentativas para se obter o resultado desejado.



Programação

```
ASPLINEX Y Z A B C
ou
BSPLINE X Y Z A B C
ou
CSPLINE X Y Z A B C
```



Função

Com a programação de um spline, você irá conectar vários pontos ao longo de uma curva.

Você pode selecionar um entre 3 tipos de spline:

- Spline A (spline akima)
- Spline B (spline não uniforme, de base relativa, NURBS)
- Spline C (spline cúbico)

5.2 Interpolação Spline



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Informações adicionais

Os spline A, B e C são modais e pertencem ao grupo de comando de movimentos. A compensação de raio da ferramenta pode ser utilizada. A monitoração de colisão é realizada na projeção do plano.

Os eixos a serem interpolados no agrupamento spline são selecionados com o comando SPLINEPATH (detalhes adicionais nas próximas páginas).



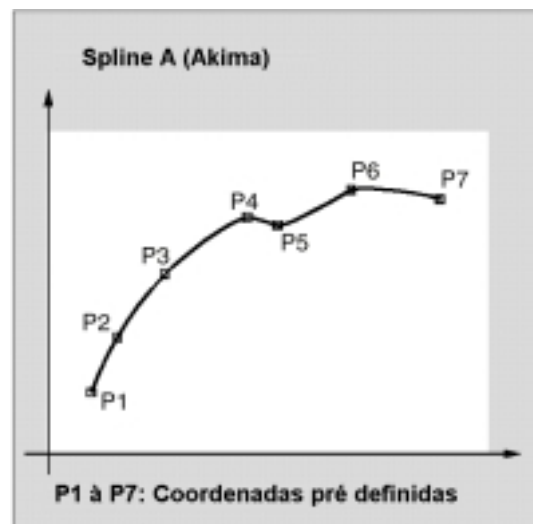
Seqüência

Spline A

O spline A (spline Akima) passa exatamente através dos pontos intermediários. Devido ao fato de poder produzir oscilações, ele não cria uma curva contínua nos pontos de interpolação.

O spline akima é local, isto é, uma alteração em um ponto de interpolação afeta somente até 6 pontos adjacentes.

A aplicação primária para este spline é portanto a interpolação de pontos digitalizados. Condições suplementares podem ser programadas para splines akima (vide abaixo para mais informações). Um polinômio de grau 3 é utilizado para a interpolação.



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

spline B

Com um spline B, as posições programadas não são pontos de interpolação, mas apenas pontos de controle do spline, ou seja, a curva é traçada “através” dos pontos, mas não passa diretamente sobre eles.

As linhas conectadas aos pontos formam o polígono de controle do spline. Splines B são a forma otimizada para definição da trajetória da ferramenta em faces esculpidas. O seu principal propósito é o de servir de interface aos sistemas de CAD. Um spline B de grau 3 não produz oscilações apesar de suas contínuas transições curvas.

Condições suplementares programadas não afetam o spline B (favor vide próximas páginas para maiores informações). O spline B é sempre tangencial ao polígono de controle em seus pontos inicial e final.

Peso do ponto:

Um peso pode ser programado para cada ponto de interpolação..

Programação:

PW = n

Faixa de valores:

$0 \leq n \leq 3$; em passos de 0.0001

Efeito:

$n > 1$ O ponto de controle exerce mais “força” na curva.

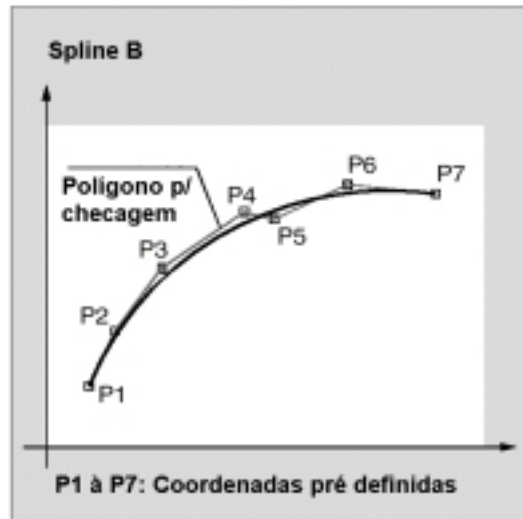
$n < 1$ O ponto de controle exerce menos “força” na curva.

Grau do spline:

Um polinômio de grau 3 é utilizado como standard, mas é possível também a utilização de polinômios de grau 2.

Programação:

SD = 2



5.2 Interpolação Spline



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



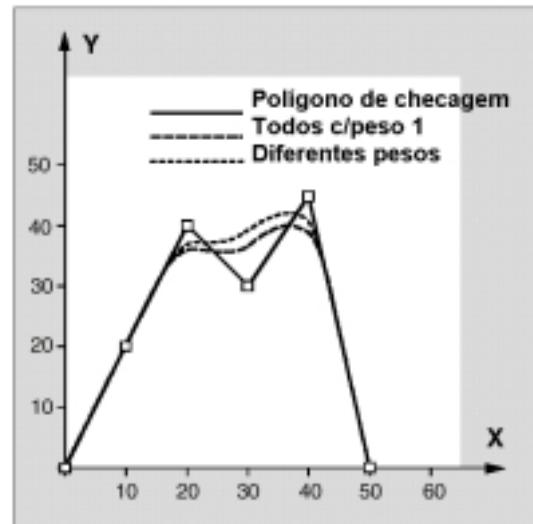
840Di

Distância entre os nós:

O controle calcula a distância entre nós é calculada de forma adequada, mas o sistema é também capaz de utilizar distâncias definidas pelo usuário.

Programação:

PL = faixa de valores para dimensionar a trajetória



Exemplo de spline B:

Todos pesos 1

| | | | | | |
|-----|---------|-----|----|------|-----|
| N10 | G1 | X0 | Y0 | F300 | G64 |
| N20 | BSPLINE | | | | |
| N30 | X10 | Y20 | | | |
| N40 | X20 | Y40 | | | |
| N50 | X30 | Y30 | | | |
| N60 | X40 | Y45 | | | |
| N70 | X50 | Y0 | | | |

Pesos diferentes

| | | | | | |
|-----|---------|-----|--------|------|-----|
| N10 | G1 | X0 | Y0 | F300 | G64 |
| N20 | BSPLINE | | | | |
| N30 | X10 | Y20 | PW=2 | | |
| N40 | X20 | Y40 | | | |
| N50 | X30 | Y30 | PW=0.5 | | |
| N60 | X40 | Y45 | | | |
| N70 | X50 | Y0 | | | |

Polígono de controle

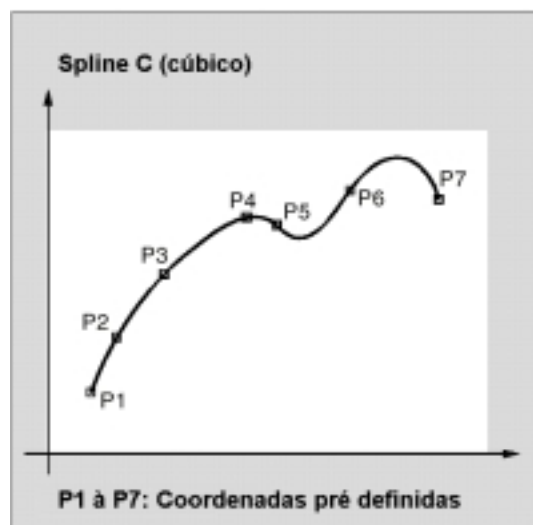
| | | | | | |
|-----|----------|-----|----|------|-----|
| N10 | G1 | X0 | Y0 | F300 | G64 |
| N20 | ;omitida | | | | |
| N30 | X10 | Y20 | | | |
| N40 | X20 | Y40 | | | |
| N50 | X30 | Y30 | | | |
| N60 | X40 | Y45 | | | |
| N70 | X50 | Y0 | | | |

Spline C

Diferente do spline akima, o spline cúbico é curvo de forma contínua nos pontos intermediários.

Entretanto, ele tende a produzir flutuações inesperadas. Ele pode ser utilizado em casos onde os pontos de interpolação encontram-se dispostos em uma curva calculada de forma analítica. Splines C utilizam polinômios de grau 3.

O spline não é local, ou seja, a alteração de um ponto de interpolação pode influenciar uma grande quantidade de blocos (com efeito decrescente gradativo).



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Condições adicionais

As seguintes condições adicionais são aplicadas somente aos splines akima e cúbicos.

A reação à transições (inicial e final destas curvas podem ser definidas em dois grupos de instruções, com três comandos cada.



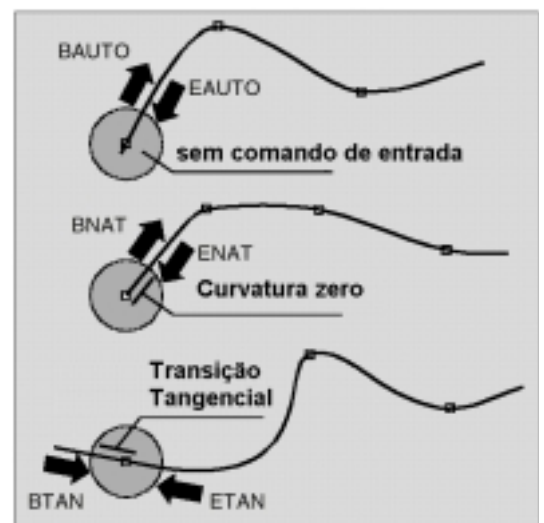
Explicação dos comandos

Início da curva spline

| | |
|-------|--|
| BAUTO | Sem comando de entrada, o início é determinado pela posição do primeiro ponto. |
| BNAT | Curvatura zero |
| BTAN | Transição tangencial com o bloco anterior (estado inicial) |

Fim da curva spline:

| | |
|-------|--|
| EAUTO | Não há comando, o fim é determinado pela posição do último ponto |
| ENAT | Curvatura zero |
| ETAN | Transição tangencial ao próximo bloco (estado inicial) |



5.2 Interpolação Spline



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D

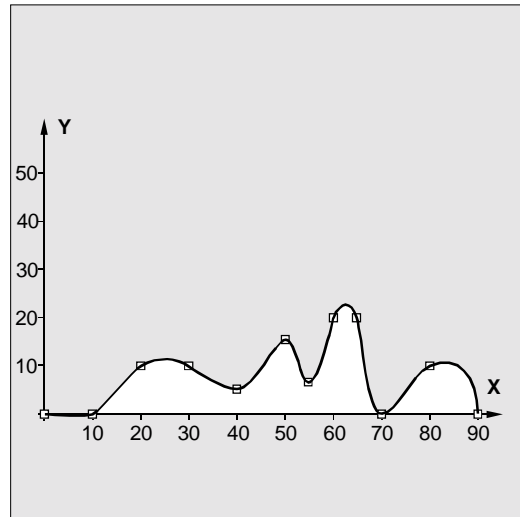


840Di



Exemplo

Spline C, curvatura zero no início e fim



N10 G1 X0 Y0 F300

N15 X10

N20 BNAT ENAT

Spline C spline, no início e fim curvatura zero

N30 CSPLINE X20 Y10

N40 X30

N50 X40 Y5

N60 X50 Y15

N70 X55 Y7

N80 X60 Y20

N90 X65 Y20

N100 X70 Y0

N110 X80 Y10

N120 X90 Y0

N130 M30



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



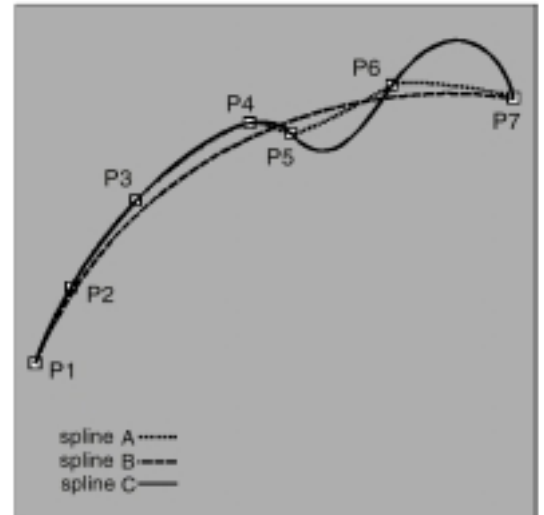
O que um spline faz?

Comparação dos 3 splines com os mesmos pontos de interpolação:

Spline A (akima)

Spline B (Bezier)

Spline C (cúbico)



Agrupamento spline

Até oito eixos de trajetória podem ser envolvidos em uma agrupamento de interpolação spline. A instrução `SPLINEPATH` define quais eixos serão envolvidos no spline. A instrução é programada em um bloco separado. Caso `SPLINEPATH` não seja programado de forma explícita, os primeiros 3 eixos do canal serão utilizados no agrupamento spline.

5.2 Interpolação Spline



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Programação

`SPLINEPATH(n,X,Y,Z,...)`



Explicação

`SPLINEPATH(n,X,Y,Z,...)`

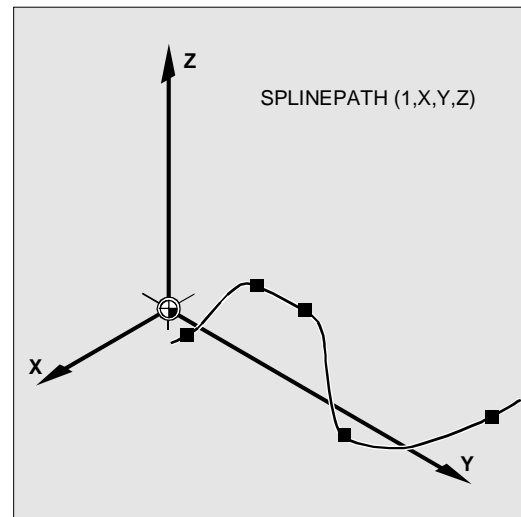
$n = 1$, valor fixo

X,Y,Z,... nome dos eixos de trajetória



Exemplo

Agrupamento spline com três eixos de trajetória



N10 G1 X10 Y20 Z30 A40 B50 F350

N11 SPLINEPATH(1,X,Y,Z)

Agrupamento spline

N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60 Spline C

N14 X30 Y40 Z50 A60 B70

Pontos de interpolação

...

N100 G1 X... Y...

Cancela seleção da interpolação spline

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Ajustes para splines

Os códigos G ASPLINE, BSPLINE e CSPLINE conectam os pontos finais do bloco com os splines.

Para este propósito, uma série de blocos (pontos finais) devem ser calculados simultaneamente.

O tamanho padrão do buffer para os cálculos é de 10 blocos.

Nem toda a informação do bloco é um ponto final de spline. Entretanto, o controle necessita de uma certa quantidade de blocos com pontos finais de spline nos 10 blocos.

Esta quantidade é a seguinte para:

| | |
|-----------|--|
| Spline A: | Ao menos 4 blocos dos 10 devem ser blocos spline. Isto não inclui blocos de comentário ou cálculos de parâmetros |
| Spline B: | Ao menos 6 blocos dos 10 devem ser blocos spline. Isto não inclui blocos de comentário ou cálculos de parâmetros |
| Spline C: | Para cada 10 blocos ao menos o conteúdo do dado de máquina \$MC_CUBIC_SPLINE_BLOCKS+1 devem ser blocos spline (também no caso standard 9) A quantidade de pontos deve ser carregada no dado de máquina \$MC_CUBIC_SPLINE_BLOCKS (valor standard 8) os quais são utilizados para o cálculo do segmento spline. |



Um alarme será emitido caso o valor tolerado seja excedido. O mesmo ocorre quando um dos eixos envolvidos no spline for eixo de posicionamento.

5.3 Compressor COMPON/COMPCURV



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

5.3 Compressor COMPON/COMPCURV



De forma geral, os sistema de CAD/CAM fornecem blocos lineares de forma a garantir a precisão programada.

Em contornos complexos isto gera uma quantidade considerável de dados para pequenas partes da trajetória, Estas pequenas partes da trajetória restringem a velocidade de execução.

Com o compressor, um certo número (máximo 10) destas pequenas trajetórias podem ser colocadas em conjunto em uma seção de trajetória.

O código G modal COMPON ou COMPCURV ativa um "compressor de blocos NC".

Com interpolação linear, esta agrupa uma quantidade de blocos com linhas retas (quantidade restrita a 10) e os aproxima através de um polinômio de grau 3 (COMPON), ou de grau 5 (COMPCURV), com uma faixa de tolerância especificada através de dados de máquina. Desta forma, o NC processa um grande bloco de movimento ao invés de uma grande quantidade de pequenos blocos.

Esta operação pode somente ser executada em blocos lineares (G1). É interrompida por qualquer outro tipo de instrução NC, tais como funções auxiliares, mas não com cálculo de parâmetros.

Os blocos a serem comprimidos podem conter apenas o número do bloco, G1, o endereço do eixo, avanço e comentário. Esta seqüência é obrigatória. Variáveis não podem ser utilizadas.

840 D
NCU 571840 D
NCU 572
NCU 573

FM-NC



840Di

Com o código G COMPON as transições de blocos são realizadas com velocidade constante, enquanto a aceleração dos eixos que participam podem estar em saltos nas transições do bloco. Isto pode aumentar a oscilação da máquina.

SW 4.4 em diante:

Com o código G COMPCURV, as transições de blocos são realizadas com aceleração constante. Isto garante transições suaves de blocos para todos os eixos do bloco.



Programação

| | |
|-------------------|---------------------|
| COMPON / COMPCURV | Ativa compressor |
| COMPOF | Desativa compressor |



Fabricante da máquina

Existem 3 dados de máquina disponíveis para a função compressor:

- \$MC_COMPRESS_BLOCK_PATH_LIMIT
O comprimento máximo da trajetória é aqui definido. Todos os blocos ao longo desta trajetória estão sujeitos à compressão. Blocos maiores não serão comprimidos.
- \$MA_COMPRESS_POS_TOL
Uma tolerância pode ser programada para cada eixo. A curva spline gerada não irá desviar mais que este valor em seus pontos finais programados. Quanto maior estes valores, mais os blocos podem ser comprimidos.
- \$MC_COMPRESS_VELO_TOL
O desvio máximo da velocidade de trajetória com o compressor ativo pode ser definido em conjunto com FLIN e FCUB.

5.3 Compressor COMPON/COMPCURV



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Exemplo

| | |
|------------------------|--|
| N10 COMPON | ou COMPCURV, compressor ON (liga) |
| N11 G1 X0.37 Y2.9 F600 | G1 deve ser programado antes do ponto final e avanço |
| N12 X16.87 Y-4.698 | |
| N13 X16.865 Y-4.72 | |
| N14 X16.91 Y-4.799 | |
| ... | |
| N1037 COMPOF | Compressor OFF (desliga) |
| ... | |



Todos os blocos cuja sintaxe seja simples serão comprimidos.

p.e.

N19 X0.103 Y0. Z0.

N20 X0.102 Y-0.018

N21 X0.097 Y-0.036

N22 X0.089 Y-0.052

N23 X0.078 Y-0.067

Não comprimidos são, p.e.

Endereços expandidos, tais como C=100 ou A=ACNC.

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

5.4 Interpolação polinomial, POLY



O sistema é capaz de executar movimentos em curvas (trajetórias) nas quais cada eixo esteja sendo operado de acordo com uma função (polinômio de grau 3, no máximo).

A equação utilizada para expressar a função polinomial é geralmente a seguinte:

$$F(p) = a_0 + a_1p + a_2p^2 + a_3p^3$$

As letras possuem os seguintes significados:

a_n : Coeficientes constantes

p : Parâmetros

Ao atribuir valores concretos para estes coeficientes, é possível gerar uma ampla variedade de curvas, linhas, parábolas e funções de potência.

Ao atribuir aos coeficientes $a_2 = a_3 = 0$, é possível criar, p.e. uma linha reta:

$$f(p) = a_0 + a_1p$$

Significado:

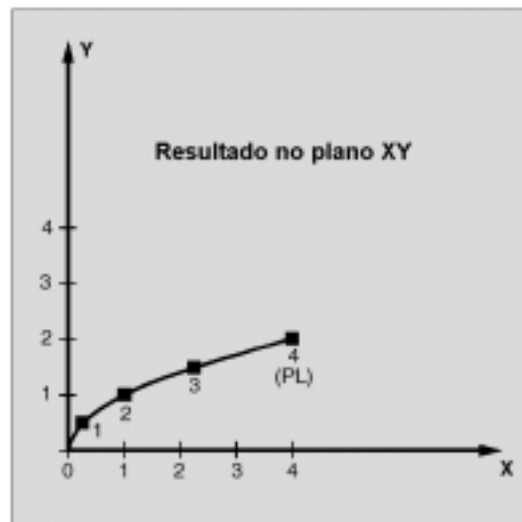
a_0 = Posição do eixo no final do bloco anterior

a_1 = Posição do eixo no fim da área de definição (PL)

Definição

A interpolação polinomial (POLY) não é um tipo de interpolação spline real. Seu principal propósito é agir como interface para a programação gerada através de curvas spline, onde partes da spline podem ser diretamente programadas.

Este modo de interpolação ajuda o NC na tarefa de calcular os coeficientes do polinômio. Pode ser aplicada para otimização em casos onde os coeficientes são fornecidos diretamente por um sistema CAD ou pós processador.



A interpolação polinomial pertence ao primeiro grupo de funções G juntamente com G0, G1, G2, G3, spline A, spline B e spline C. Caso esteja ativo,

5.4 Interpolação polinomial, POLY



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

não existe a necessidade de se programar a sintaxe polinomial: Os eixos são programados com seus nomes e pontos finais são movimentados linearmente. Caso todos os eixos sejam programados desta forma, o sistema responde como se estivesse sendo programada a função G1.

A interpolação polinomial é desativada por outro comando do mesmo grupo G (p.e. G0, G1).



Coeficiente polinomial

O valor de PO (PO[]=) especifica todos os coeficientes do polinômio para um eixo. Vários valores, separados por vírgulas, são especificados de acordo com o grau do polinômio. Diferentes graus podem ser programados para diferentes eixos dentro de um mesmo bloco.

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Programação

POLY PO[X]=(x_{e1}, a₂, a₃) PO[Y]=(y_{e1}, b₂, b₃) PO[Z]=(z_{e1}, c₂, c₃) PL=n



Explicação

| | |
|--|--|
| POLY | Ativa a interpolação polinomial |
| PO[]=(... , ... , ...) | Pontos finais e coeficientes do polinômio |
| x _e , y _e , z _e | Especificação da posição final para os eixos relevantes, faixa de valores programados da mesma forma como dimensões de trajetórias |
| a ₂ , a ₃ | Os coeficientes a ₂ e a ₃ são programados com seus valores, a faixa de valores da mesma forma como na programação de trajetória. O último coeficiente pode ser omitido caso seja igual a zero. |
| PL | Comprimento do intervalo de parâmetros sobre os quais os polinômios são definidos (faixa de definição da função f(p)). O intervalo sempre começa em 0. p pode ser programado com valores enter 0 e PL. Teoricamente a faixa de valores para PL é: 0.0001 ... 99 999.9999. Os valores de PL são utilizados no bloco onde PL é programado. PL=1 é utilizado caso PL não tenha sido programado. |



Exemplo

| | |
|--|-------------------------------------|
| N10 G1 X... Y... Z... F600 | |
| N11 POLY PO[X]=(1,2.5,0.7) -> -> PO[Y]=(0.3,1,3.2) PL=1.5 | Liga interpolação polinomial |
| N12 PO[X]=(0,2.5,1.7) PO[Y]=(2.3,1.7) PL=3 | |
| ... | |
| N20 M8 H126 ... | |
| N25 X70 PO[Y]=(9.3,1,7.67) PL=5 | Vários eixos programados |
| N27 PO[X]=(10.2.5) PO[Y]=(2.3) | Sem PL programado, é utilizado PL=1 |
| N30 G1 X... Y... Z. | Desliga interpolação polinomial |
| ... | |

5.4 Interpolação polinomial, POLY



840D
NCU 571



840D
NCU 572
NCU 573



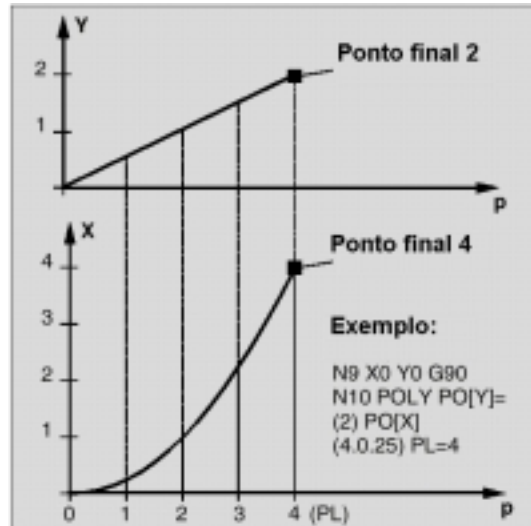
810D



840Di

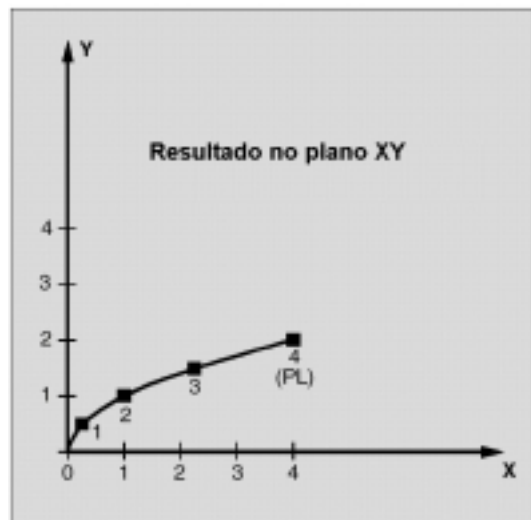


Exemplo de uma curva no plano X/Y



```
N9 X0 Y0 G90 F100
```

```
N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4
```



840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Caso especial – denominador polinomial

O comando `PO[]=(. . .)` pode ser utilizado para programar um denominador polinomial comum para os eixos geométricos (sem a especificação dos nomes dos eixos), ou seja, o movimento dos eixos geométricos são então interpolados como quociente de dois polinômios.

Com esta opção de programação, é possível representar formas cônicas (círculos, elipses, parábolas, hipérboles) de forma exata.



Exemplo

`POLY G90 X10 Y0 F100`

Os eixos geométricos são posicionados linearmente na posição X10, Y0

`PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)`

Os eixos geométricos movimentam-se através do quadrante para X0, Y10

O coeficiente constante (a_0) do denominador polinomial é sempre colocado com valor 1, o ponto final não depende de G90/G91.

O resultado final obtido do exemplo acima é o seguinte:

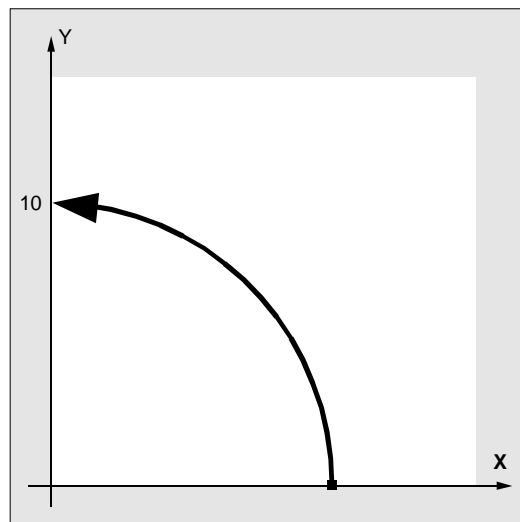
$X(p)=10(1-p^2)/(1+p^2)$ e $Y(p)=20p/(1+p^2)$
onde $0 \leq p \leq 1$

Como resultado dos pontos iniciais, finais, coeficiente a_2 e $PL=1$, os valores intermediários são os seguintes

Numerador (X)= $10+0 \cdot p-10p^2$

Numerador (Y)= $0+20 \cdot p+0 \cdot p^2$

Denominador = $1+2 \cdot p+1 \cdot p^2$



Um alarme será emitido caso um denominador polinomial com zeros seja programado dentro do intervalo $[0, PL]$ onde a interpolação polinomial esteja ativa. Denominadores polinomiais não tem efeito no

5.4 Interpolação polinomial, POLY



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

movimento de eixos especiais.



Informações adicionais

A compensação de raio da ferramenta pode ser ativada com G41, G42 em conjunto com a interpolação polinomial e pode ser aplicada da mesma forma como nos modos de interpolação linear ou circular.

5.5 Ref.de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

5.5 Referência de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante)



Programação

SPATH Referência de trajetória para os eixos do FGROUP é o comprimento do arco
UPATH O parâmetro da curva é a referência de trajetória para os eixos do FGROUP



Introdução

Durante a interpolação polinomial, o usuário pode necessitar de duas relações entre a velocidade – determinada para os eixos do FGROUP e dos outros eixos de trajetória: Os eixos de trajetória devem ser controlados

- tanto sincronizados com a trajetória dos eixos do FGROUP
- ou sincronizados com os parâmetros da curvar.

Anteriormente, somente a primeira variante de controle de movimento era implementada; a partir da SW 4.3 foram disponibilizados os códigos G (SPATH, UPATH) para seleção e programação da resposta desejada.



Função

Durante a interpolação polinomial - e aqui nos referimos a interpolação polinomial no seu sentido restrito (POLY), todas os tipos de interpolação spline (ASPLINE, BSPLINE, CSPLINE) e interpolação linear com compressão (COMPON, COMPCURV) – As posições de todos os eixos de trajetória são determinadas através de polinômios $p_i(U)$. O parâmetro de curva U é comutado de 0 para 1 no bloco NC, conforme normalização.

Os eixos para os quais a velocidade de trajetória será considerada podem ser selecionados a partir dos eixos de trajetória, através da instrução FGROUP. Entretanto, durante a interpolação polinomial, uma interpolação com velocidade constante na trajetória S destes eixos normalmente significa uma alteração não constante para o curva U.

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Portanto, para os eixos não presentes no FGROUP existem duas formas de prosseguir na trajetória:

1. Tanto seus movimentos podem ser sincronizados com a trajetória S (SPATH)
2. Ou sincronizados com o parâmetro de curva U dos eixos do FGROUP (UPATH).

Ambos os tipos de interpolação de trajetória são usados em diferentes aplicações e podem ser comutados através dos códigos G SPATH e UPATH.

UPATH e SPATH também determinam a relação entre a palavra polinomial F (FPOLY, FCUB, FLIN) com o movimento de trajetória.



Expansão para arredondamento

Caso todos os eixos de trajetória não estejam contidos no FGROUP, os eixos não incluídos freqüentemente sofrem alterações súbitas de velocidade durante as transições dos blocos

Para redução da velocidade na transição do bloco, o controle pode limitar a extensão desta mudança para um valor permissível programado em MD 32300:

MAX_AX_ACCEL e MD 32310:

_MAX_ACCEL_OVL_FACTOR. Esta desaceleração pode ser evitada através do arredondamento da posição específica com relação aos eixos de trajetória.

- Arredondamento com G641
O arredondamento é ativado de forma modal através de G641 e da especificação do raio de arredondamento ADIS (ou ADISPOS em movimento rápido) para as funções de trajetória. O controle encontra-se livre para não obedecer a trajetória dentro deste raio em torno do ponto de mudança de bloco e a substitui por uma trajetória otimizada.
Desvantagem: Somente um valor ADIS é válido para todos os eixos.
Vide também: Referências /PG/, manual de programação – Fundamentos, capítulo 5, Trajetória

5.5 Ref.de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

- Arredondamento com G642
Arredondamento com tolerância axial é ativado através de G642. O arredondamento não é realizado em uma área específica ADIS, mas sim garante a tolerância axial definida em MD 33100: COMPRESS_POS_TOL é adicionada.
O restante do funcionamento é idêntico à G641.



Referências: /FB/, B1, Modo trajetória contínua, Parada exata, e LookAhead



Condições adicionais

A transferência de trajetória especificada não tem influência para

- interpolações lineares e circulares,
- rosqueamento
- caso todos os eixos de trajetória estejam incluídos no FGROUP,.



Ativação

A referência de trajetória para os eixos que não estão contidos no FGROUP é definida através dos comandos SPATH e UPATH contidas no grupo G de nr.45. Os comandos são modais. Caso SPATH esteja ativo, os eixos são movimentados de forma síncrona com a trajetória; e caso UPATH ativo, de forma transversal com o parâmetro de curva.

5.5 Ref.de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante)840D
NCU 571840D
NCU 572
NCU 573

810D

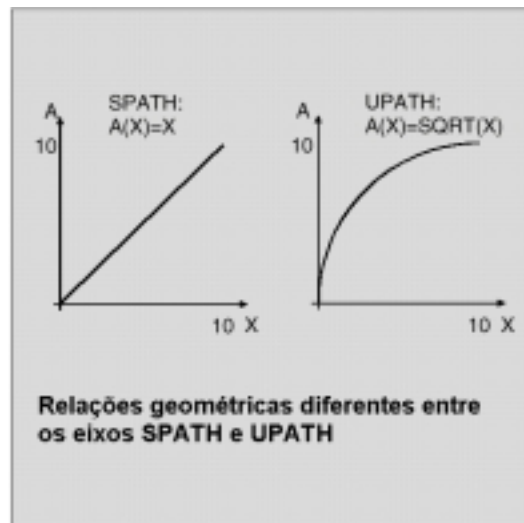


840Di

**Exemplo de programação**

O seguinte exemplo de programa mostra a diferença entre ambos os tipos de controle de movimento.

Ambas as vezes as definições iniciais de FGROUP(X,Y,A) encontram-se ativas.



```
N10 G1 X0 A0 F1000 SPATH
```

```
N20 POLY PO[X]=(10, 10) A10
```

ou

```
N10 G1 X0 F1000 UPATH
```

```
N20 POLY PO[X]=(10, 10) A10
```

No bloco N20, a trajetória S dos eixos do FGROUP depende do quadrado do parâmetro de curva U.

Entretanto, diferentes posições são atingidas para o eixo síncrono A ao longo da trajetória de X, dependendo da ativação de SPATH ou de UPATH:

Resposta do controle durante a ligação, troca de modo, reset, pesquisa de bloco e REPOS

Após um reset o código G definido pelo MD 20150: GCODE_RESET_VALUES [44] será ativado (grupo G 45).

O valor base para os tipos de arredondamentos é definido no MD 20150: GCODE_RESET_VALUES [9] (Grupo G 10).

5.5 Ref.de trajetória ajustável, SPATH, UPATH (SW 4.3 em diante)840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Dados de máquina/opções

O grupo G a ser ativado após um reset é determinado através dos dados de máquina MD 20150:

GCODE_RESET_VALUES [44].

Para manter a compatibilidade com os comandos anteriormente produzidos, SPATH é o valor de fornecimento.

A definição tomada como base para o tipo de arredondamento encontra-se no MD 20150:

GCODE_RESET_VALUES [9] (Grupo G 10).

O dado de máquina de eixo MD 33100:

COMPRESS_POS_TOL foi expandido a partir da SW4.3. Ele contém as tolerâncias para a função de compressão e para o arredondamento com G642.

5.6 Medições através de pontas de prova, MEAS, MEAW

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di

5.6 Medições através de pontas de prova, MEAS, MEAW



Programação

| | | |
|---------|---------------------|--|
| MEAS=±1 | G... X... Y... Z... | (+1/+2 medição com a borda de subida e cancelamento do percurso restante) |
| MEAS=±2 | G... X... Y... Z... | (-1/-2 medição com a borda de descida e cancelamento do percurso restante) |
| MEAW=±1 | G... X... Y... Z... | (+1/+2 medição com a borda de subida sem cancelamento do percurso restante) |
| MEAW=±2 | G... X... Y... Z... | (-1/-2 medição com a borda de descida sem cancelamento do percurso restante) |



Explicação dos comandos

| | |
|-----------|---|
| MEAS=±1 | Medição com o apalpador 1 na entrada de medição 1 |
| MEAS=±2 * | Medição com o apalpador 2 na entrada de medição 2 |
| MEAW=±1 | Medição com o apalpador 1 na entrada de medição 1 |
| MEAW=±2 * | Medição com o apalpador 2 na entrada de medição 2 |

*Máximo de duas entradas dependendo da configuração.



Seqüência

As posições coincidem com a borda de comutação do apalpador e são consideradas para todos os eixos programados no bloco NC. O valor da posição de cada eixo é escrito em uma posição de memória apropriada. Um máximo de dois apalpadores podem ser instalados.

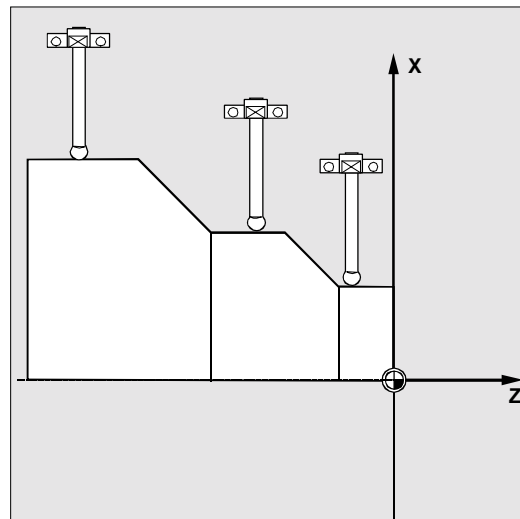
Resultado da medição

O resultado da medição encontra-se disponível na seguinte variável para estes eixos:

- Em \$AA_MM[Eixo] no sistema de coordenadas da máquina
- Em \$AA_MW[Eixo] no sistema de coordenadas da peça

Não é gerada parada de pré-processamento interna quando estas variáveis são lidas.

Uma parada de pré-processamento deve ser programada com STOPRE na posição de programa adequada. Caso contrário o sistema irá ler valores



5.6 Medições através de pontas de prova, MEAS, MEAW



840D
NCU 571



840D
NCU 572
NCU 573



810D
CCU 2



840Di

falsos.

Estado da medição

A variável de estado `$AC_MEA[n]` (n= número do apalpador) pode ser escaneada caso o estado do apalpador deva ser considerado pelo programa:

- 0 Medição não realizada
- 1 Medição realizada com sucesso
(apalpador comutou seu estado)



Caso o apalpador seja defletido durante a execução do programa, esta variável é colocada em 1. No início de um bloco de medição, ela é automaticamente colocada no estado correspondente do apalpador.

Programando blocos de medição, MEAS, MEAW

Quando o comando MEAS for programado em conjunto com um modo de interpolação, as posições atuais são aproximadas e os valores medidos são gravados simultaneamente. A distância a ser percorrida entre o ponto atual e a posição desejada é cancelada.

A função MEAW é empregada em caso de tarefas especiais de medição em que uma posição programada deve ser sempre atingida.

MEAS e MEAW são programadas com comandos de movimento. Os avanços e tipos de interpolação (G0, G1, ...) devem ser selecionados de forma adequada à tarefa de medição; isto também se aplica à quantidade de eixos.

Exemplo:

```
N10 MEAS=1 G1 F1000 X100 Y730 Z40
```

Bloco de medição com apalpador na primeira entrada de medição e interpolação linear. Uma parada de pré processamento é gerada automaticamente.

5.6 Medições através de pontas de prova, MEAS, MEAW



840D
NCU 571



840D
NCU 572
NCU 573



810D
CCU 2



840Di

Gravando a medição

As posições de todos os eixos de trajetória e posicionamento (a quantidade máxima de eixos depende da configuração do controle) no bloco de movimentação são armazenadas.

Em caso de MEAS, o movimento é interrompido de modo definitivo após o apalpador Ter sido acionado.

Comentário

Caso um eixo geométrico (GEO) tenha sido programado em um bloco de medição, os valores das posições de todos os eixos geométricos serão gravados.

Caso um eixo que participe de uma transformação seja programado em um bloco de medição, os valores medidos para todos os eixos que participam desta transformação são gravados.



Informações adicionais

As funções MEAS e MEAW não são modais.

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di

5.7 Funções adicionais de medição MEASA, MEAWA, MEAC (SW 4 em diante, opção)



Programação

| | |
|--|--|
| MEASA[eixo]=(modo, TE1,..., TE 4) | Medição com cancelamento da distância a percorrer |
| MEASA[eixo]=(modo, TE1,..., TE 4) | Medição sem cancelamento da distância a percorrer |
| MEAC[eixo]=(modo, memória de medição, TE 1,...TE4) | Medição contínua sem cancelamento da distância a percorrer |



Explicação

| | |
|--------------------|--|
| Eixo | Nome do eixo do canal usado na medição |
| Modo | Dois dígitos definem o modo de operação, consistindo de Modo de medição (unidade) 0 cancela a medição 1 Modo 1: Até 4 eventos de disparo podem ser ativados simultaneamente 2 Modo 2: Até 4 eventos de disparo podem ser ativados em seqüência 3 Modo 3: Até 4 eventos de disparo podem ser ativados em seqüência Entretanto, sem monitoração do evento de disparo 1 no START (alarmes 21700/21703 suprimidos) Nota: Modo 3 não possível com MEAC Sistema de medição (dezena) 0 ou sem definição: Ativa sistema de medição 1 Sistema de medição 1 2 Sistema de medição 2 3 Ambos sistemas de medição |
| TE 1...4 | Evento de disparo 1 Borda de subida apalpador 1 -1 Borda de descida apalpador 1 2 Borda de subida apalpador 2 -2 Borda de descida apalpador 2 |
| Memória de medição | Quantidade de FIFO (Memória circular) |

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di



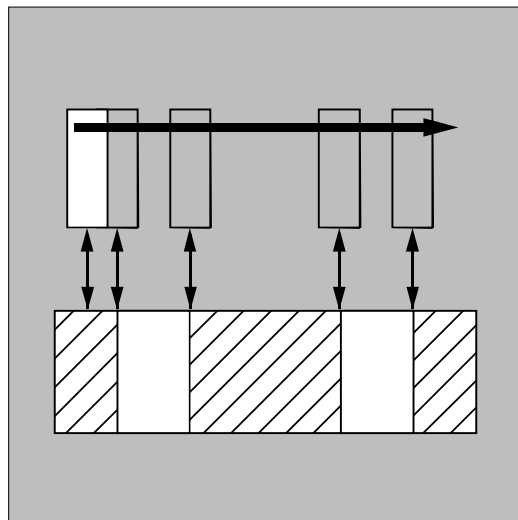
Função

Medições axiais estão disponíveis a partir da SW 4. Com este sistema as medições podem ser feitas de forma axial com vários apalpadores e vários sistemas de medição.

Quando MEASA ou MEAWA forem programados, até quatro valores são obtidos para o eixo programado em cada etapa de medição. Os valores medidos são gravados em variáveis de sistema de acordo com o evento de disparo.

MEASA e MEAWA são comandos não modais.

Operações contínuas de medição podem ser enxutas com MEAC. Neste caso, os resultados de medição são gravados em variáveis FIFO. A quantidade máxima de medições é também de 4 para MEAC.



Seqüência

As medições podem ser programadas no programa de usinagem **ou** a partir de uma ação síncrona (capítulo 10). Favor notar que somente uma tarefa de medição pode ser ativada a qualquer instante para cada eixo.



Informações adicionais

- O avanço deve ser ajustado adequadamente para a tarefa de medição.
- No caso de **MEASA** e **MEAWA**, os resultados corretos podem ser garantidos somente com velocidades nas quais não mais que um evento de disparo de mesmo tipo e não mais que 4 eventos ocorram em uma posição do ciclo de controle.
- No caso de medições contínuas com **MEAC**, A relação entre o ciclo de interpolação e o ciclo de controle de posição não deve ser superior à 8:1.

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di

Evento de disparo

Até 4 eventos de disparo de um apalpador selecionado podem ser processados em cada medição, ou seja, até 2 apalpadores com dois sinais de medição cada. A seqüência de processamento e a quantidade máxima de eventos de disparo dependem do modo selecionado.



Em uma tarefa de medição o mesmo evento de disparo pode ser utilizado somente uma vez (somente para o modo 1)!

Modo de operação

O primeiro dígito da seleção de modo seleciona o sistema de medição desejado. Caso somente um sistema de medição esteja instalado, mas um segundo seja programado, o sistema instalado será automaticamente selecionado.

Com o segundo dígito, ou sejam com o **modo de medição**, o processo de medição é adaptado às capacidades do sistema de controle conectado:

- **Modo 1:** Os eventos de disparo são analisados em seqüência **cronológica**.
Ao selecionar este modo, somente um evento de disparo pode ser programado nos módulos para até seis eixos. Caso mais de um evento de disparo seja especificado, o modo é automaticamente comutado para o modo 2 (sem mensagens).
- **Modo 2:** Os eventos de disparo são analisados na seqüência **programada**.
- **Modo 3:** Os disparos de eventos devem ser considerados na seqüência programada, entretanto não será realizada a monitoração do evento 1 no START.



Informações adicionais

Caso dois sistemas de medição estejam sendo utilizados, não mais que 2 eventos podem ser programados.

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)



840D
NCU 571



840D
NCU 572
NCU 573



810D
CCU 2



840Di

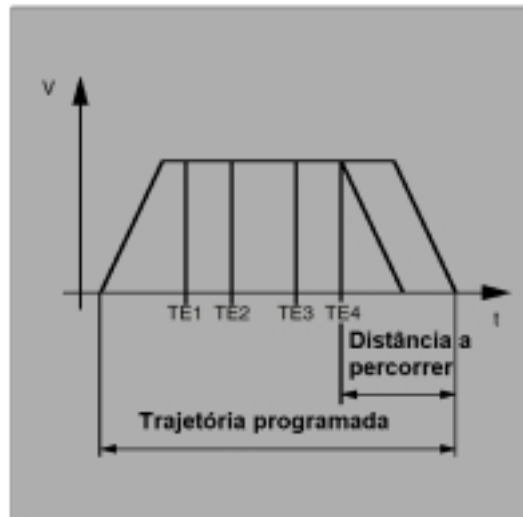
Medição com e sem o cancelamento da distância a ser percorrida

Quando o comando MEASA for programado, a distância a ser percorrida não é cancelada até que os valores de medição tenham sido gravados.

A função MEAWA é empregada em casos especiais de medição onde a posição programada deva sempre ser alcançada.

MEASA e MEAWA podem ser programados no mesmo bloco.

Caso MEASA/MEAWA sejam programados com MEAS/MEAW no mesmo bloco, uma mensagem de erro será emitida.



- MEASA não pode ser programada em ações síncronas.
Como uma alternativa, MEAWA mais o comando para apagar o percurso restante podem ser programados em ações síncronas.
- Caso a tarefa de medição MEAWA seja iniciada a partir de ações síncronas, os valores medidos estão presentes nas coordenadas da máquina.

Resultados para MESAS, MEAWA

Os resultados das medições encontram-se disponíveis nas seguintes variáveis de sistema:

- No sistema de coordenadas da máquina:

\$AA_MM1[Eixo] Valor medido do sistema de medição programado no evento de disparo 1

... ..

\$AA_MM4[Eixo] Valor medido do sistema de medição programado no evento de disparo 4

- No sistema de coordenadas da peça:

\$AA_MW1[Eixo] Valor medido do sistema de medição programado no evento de disparo 1

... ..

\$AA_MW4[Eixo] Valor medido do sistema de medição programado no evento de disparo 4

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di



Informações adicionais

Não é gerada nenhuma parada de pré processamento quando as variáveis são lidas.

Uma parada de pré processamento deve ser programada na posição apropriada com STOPRE (seção 15.1). Caso contrário serão lidos valores errados.

Caso uma medição deva ser realizada em um eixo geométrico, a mesma instrução de medição deve ser programada para os demais eixos geométricos. O mesmo deve ser feito para os eixos envolvidos em transformações.

Exemplo:

```
N10 MEASA[Z]=(1,1) MEASA[Y]=(1,1)
MEASA[X]=(1,1) GO Z100;
```

ou

```
N10 MEASA[Z]=(1,1) POS[Z]=100
```



Medição com dois sistemas

Caso uma medição seja executada por dois sistemas, cada uma das duas possibilidades de eventos de disparo de ambos sistemas do eixo relevante é obtida. O preenchimento das variáveis reservadas é então realizado:

| | | | |
|----------------|----|----------------|---|
| \$AA_MM1[Eixo] | Ou | \$AA_MW1[Eixo] | Medir do sistema 1 com disparo no evento 1 |
| \$AA_MM2[Eixo] | Ou | \$AA_MW2[Eixo] | Medir do sistema 2 com disparo no evento 1 |
| \$AA_MM3[Eixo] | Ou | \$AA_MW3[Eixo] | Medir do sistema 2 com disparo no evento 1 |
| \$AA_MM4[Eixo] | Ou | \$AA_MW4[Eixo] | Valor medido do sistema 2 com evento de disparo |

O estado da ponta de prova pode ser lida com

\$A_PROBE[n]

n= Apalpador

1==Apalpador acionado

0==Apalpador não acionado

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)



840D
NCU 571



840D
NCU 572
NCU 573



810D
CCU 2



840Di

Estado do apalpador para MEASA, MEAWA

Caso o estado de comutação do apalpador necessite ser considerado no programa, o estado da tarefa de medição pode ser amostrado através de **\$AC_MEA[n]**, onde n é o número do apalpador

Uma vez que todos os eventos de disparo do apalpador "n" programado em um bloco tenham ocorrido, estas variáveis são comutadas para o estágio "1". Caso contrário este valor é zero.

Caso a medição tenha sido iniciada com ações síncronas, **\$AC_MEA** não é atualizada. Para estes casos, novos sinais do PLC DB(31-40) no dbb62 bit 3 ou a variável equivalente **\$AA_MEA**ACT["eixo"] deve ter sido testada

Significado **\$AA_MEA**ACT==1: medição ativa

\$AA_MEAACT==0: Medição não ativa

Referências: /FB/ M5, Medições

Medição contínua MEAC

Os valores medidos para MEAC estão disponíveis nas coordenadas de sistema da máquina e armazenados na memória FIFO[n] especificada (memória de circulação). Caso existam 2 apalpadores configurados para a medição, os valores de medição os valores medidos pelo segundo apalpador são gravados separadamente na memória FIFO[n+1] especialmente configurada para este propósito (definida através de dados de máquina).

A memória FIFO é uma memória de circulação na qual os valores medidos são escritos em variáveis **\$AC_FIFO** de acordo com o princípio de circulação

Referências: /PGA/ Capítulo 10, Ações síncronas

Informações adicionais

- O conteúdo da FIFO pode ser lido somente uma vez a partir do armazenamento circular. Caso estes dados devam ser usados mais vezes, devem ser gravados transferidos para dados de usuário.
- Caso a quantidade de valores medidos exceda o comprimento máximo da FIFO especificado em dados de máquina, as medições são automaticamente encerradas.
- Um processo de medição sem fim pode ser implementado através da leitura cíclica de valores. Neste caso, os dados devem ser lidos na mesma frequência com que as novas medições são carregadas.

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di



Exemplo de programação

(Avaliação e, sequência cronológica)

a) com 1 sistema de medição

| | |
|--|---|
| ... | |
| N100 MEASA[X] = (1,1,-1) G01 X100 F100 | Medição no modo 1 com o sistema de medição ativo. Espera pela borda de subida/descida do apalpador 1 na trajetória X=100. |
| N110 STOPRE | Parada de pré-processamento |
| N120 IF \$AC_MEA[1] == FALSE gotof END | Checa se medição foi bem sucedida. |
| N130 R10 = \$AA_MM1[X] | Grava o valor medido obtido pelo primeiro evento de disparo programado (borda de subida) |
| N140 R11 = \$AA_MM2[X] | Grava o valor medido obtido pelo segundo evento de disparo programado (borda de descida). |
| N150 END: | |



Exemplo de programação

b) Com 2 sistemas de medição

| | |
|--|--|
| ... | |
| N200 MEASA[X] = (31,1-1) G01 X100 F100 | Medições no modo 1 com ambos sistemas de medição, Espera pelo sinal de medição, ou seja, pela borda de subida ou decida do apalpador 1 no percurso X = 1==100. |
| N210 STOPRE | Parada de pré-processamento |
| N220 IF \$AC_MEA[1] == FALSE gotof END | Checa se medição foi bem sucedida. |
| N230 R10 = \$AA_MM1[X] | Grava valor do sistema de medição 1 na borda de subida |
| N240 R11 = \$AA_MM2[X] | Grava valor do sistema de medição 2 na borda de subida |
| N250 R12 = \$AA_MM3[X] | Grava valor do sistema de medição 1 na borda de descida |
| N260 R13 = \$AA_MM4[X] | Grava valor do sistema de medição 2 na borda de descida |
| N270 END: | |

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di

**Medição com o cancelamento do caminho restante
no modo 2**

(avaliação na sequência programada)

...

```
N100 MEASA[X] = (2,1,-1,2,-2) G01 X100
F100
```

Medição no modo 2 utilizando o sistema de medição ativo. Espera pelo sinal de medição na seguinte ordem: borda de subida do apalpador 1, borda de descida do apalpador 1, borda de subida do apalpador 2 e borda de descida do apalpador 2 no percurso X=100.

```
N110 STOPRE
```

Parada de pré-processamento

```
N120 IF $AC_MEA[1] == FALSE gotof
```

Checa o sucesso da medição com o apalpador 1

```
PROBE2
```

```
N130 R10 = $AA_MM1[X]
```

Grava o valor obtido pelo primeiro evento de disparo programado (borda de subida do apalpador 1)

```
N140 R11 = $AA_MM2[X]
```

Grava o valor obtido pelo segundo evento de disparo programado (borda de subida do apalpador 1)

```
N150 PROBE2:
```

```
N160 IF $AC_MEA[2] == FALSE gotof END
```

Checa o sucesso da medição com o apalpador 2

```
N170 R12 = $AA_MM3[X]
```

Grava o valor obtido pelo terceiro evento de disparo programado (borda de subida do apalpador 2)

```
N180 R13 = $AA_MM4[X]
```

Grava o valor obtido pelo quarto evento de disparo programado (borda de subida do apalpador 2)

```
N190 END:
```

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)



840D
NCU 571



840D
NCU 572
NCU 573



810D
CCU 2



840Di



Exemplo de programação

Modo de medição contínua no modo 1

(Em seqüência cronológica)

Medição de até 100 valores

...

```
N110 DEF REAL MEASVALUE[100]
```

```
N120 DEF INT INDEX = 0
```

```
N130 MEAC[X] = (1,1,-1) G01 X1000 F100
```

Medição no modo 1 utilizando o sistema de medição ativo, grava os valores medidos em \$AC_FIFO1, espera pela borda de descida do apalpador 1 na trajetória para o ponto X=1000.

```
N135 STOPRE
```

```
N140 MEAC[X] = (0)
```

Encerra a medição quando a posição do eixo for atingida.

```
N150 R1 = $AC_FIFO1[4]
```

Grava a quantidade de medições acumuladas no parâmetro R1.

```
N160 FOR INDEX = 0 TO R1-1
```

```
N170 MEASVALUE[INDEX] = $AC_FIFO1[0]
```

Lê os valores medidos a partir de \$AC_FIFO1 e os armazena.

```
N180 ENDFOR
```

Medição com o cancelamento da distância a percorrer após 10 valores medidos

...

```
(x)
```

Cancela distância a percorrer

```
N20 MEAC[x]=(1,1,1,-1) G01 X100 F500
```

```
N30 MEAC[X]=(0)
```

```
N40 R1 = $AC_FIFO1[4]
```

Quantidade de valores medidos

...

5.7 Funções ad.medição MEASA, MEAWA, MEAC (SW 4, opção)840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di



Os seguintes erros de programação podem ser detectados e são exibidos de forma apropriada:

- MEASA/MEAWA programado com MEAS/MEAW no mesmo bloco

Exemplo:

```
N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100
```

- MEASA/MEAWA com quantidade de parâmetros <2 ou >5

Exemplo:

```
N01 MEAWA[X]=(1) G01 F100 POS[X]=100
```

- MEASA/MEAWA com evento de disparo diferente de 1/ -1/ 2/ -2

Exemplo:

```
N01 MEASA[B]=(1,1,3) B100
```

- MEASA/MEAWA em modo inválido

Exemplo:

```
N01 MEAWA[B]=(4,1) B100
```

- MEASA/MEAWA com evento de disparo programado duas vezes

Exemplo:

```
N01 MEASA[B]=(1,1,-1,2,-1) B100
```

- MEASA/MEAWA faltando eixo geométrico (GEO)

Exemplo:

```
N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100
```

Eixos GEO X/Y/Z

- Medição não consistente com os eixos GEO

Exemplo:

```
N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01  
X50 Y50 Z50 F100
```

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di

5.8 Funções especiais para usuários OEM



Endereços OEM

Os significados dos endereços OEM são definidos pelo usuário OEM.

Suas funções são implementadas através de ciclos compilados. São reservados 5 endereços OEM.

Os identificadores são configuráveis.

Os endereços OEM podem ser programados em quaisquer blocos.



Interpolações OEM

O usuário OEM pode definir duas interpolações adicionais. Suas funções são incorporadas através de ciclos compilados.

Os nomes das funções G (OEMIPO1, OEMIPO2) são definidas pelo usuário OEM.

Endereços OEM (vide acima) podem ser utilizados especificamente para interpolações OEM.



Grupos G 800-819 reservados

Dois grupos com 10 funções G OEM cada são reservados para os usuários OEM.

Isto permite que as funções incorporadas por um usuário OEM possam ser acessadas por aplicações externas.



Funções e subprogramas

Os usuários OEM podem também definir funções de ajuste e subprogramas com transferência de valores.

5.9 Critérios programáveis p/fim de mov. (SW 5.1 em diante)840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di

5.9 Critérios programáveis para fim de movimentação (SW 5.1 em diante)**Programação**

```

FINEA[ <Eixo> ]
COARSEA[ <Eixo> ]
IPOENDA[ <Eixo> ]

```

**Explicação dos comandos**

| | |
|---------|---|
| FINEA | Fim de movimento quando "Parada exata FINE" atingida |
| COARSEA | Fim de movimento quando "Parada exata COARSE" atingida |
| IPOENDA | Fim de movimento quando "Parada do interpolador" atingida |
| Eixo | Nome do eixo no canal (X, Y,) |

**Função**

De forma similar aos critérios para as passagens contínuas (G601, G602 e G603), os critérios de fim de movimento podem ser programados tanto para movimentos simples de um único eixo ou em ações síncronas para eixos de comando/PLC.

Dependendo dos critérios de fim de movimento, os blocos de programa ou blocos de ciclos tecnológicos com movimentação de um eixo serão executados em tempos diferentes.

O mesmo se aplica à comandos de posicionamento através dos Cs 15/16/18 através do PLC.

Variável de sistema \$AA_MOTENDA

Os critérios de fim de movimento podem ser comutados através do uso da variável de sistema

```
$AA_MOTENDA[ <eixo> ].
```

- \$AA_MOTENDA[<eixo>] = 1
- \$AA_MOTENDA[<eixo>] = 2
- \$AA_MOTENDA[<eixo>] = 3

Fim de movimento c/"parada exata FINE"

Fim de movimento c/"parada exata COARSE"

Fim de movimento c/"parada IPO"

5.10 Bloco de parâmetros servo programável (SW 5.1 em diante)

840D
NCU 571840D
NCU 572
NCU 573810D
CCU 2

840Di



Informações adicionais

O último valor programado é mantido após RESET.

Referências: /FB1/V1 avanços



Exemplo de programação

...

```
N110 G01 POS[X]=100 FA[X]=1000 ACC[X]=90 IPOENDA[X]
```

Movimento para a posição X100 com uma velocidade de trajetória de 1000 rpm, um valor de aceleração de 90% e fim de movimento quando atingida a parada do interpolador

...

```
N120 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140 IPOENDA[X]
```

Movimenta para a posição X50 quando a entrada 1 for ativada, com uma velocidade de trajetória de 2000rpm, um valor de aceleração de 140% e o fim do movimento quando atingido o Sto. do interpolador.

...

5.10 Bloco de parâmetros servo programável (SW 5.1 em diante)



Programação

```
SCPARA[<Eixo>]= <vale>
```



Explicação dos comandos

| | |
|--------|---|
| SCPARA | Define o bloco de parâmetros |
| Eixo | Nome do eixo no canal (X, Y, ...) |
| Valor | Bloco de parâmetros desejados (1<= valor <=6) |



Função

Utilizando SCPARA, é possível programar o bloco de parâmetros (composto de Md) no programa de usinagem e em ações síncronas (anteriormente, somente através do PLC).

5.10 Bloco de parâmetros servo programável (SW 5.1 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



810D
CCU 2



840Di

DB3n DBB9 Bit3

Para garantir que não existam cinéfilos entre os comandos do usuário no PLC ou no NC, um bit adicional é definido na interface PLC→NCK: DB3n DBB9 Bit3 "Carga de parâmetro travada através de SCPARA".



Caso o comando SCPARA seja efetuado para um bloco de parâmetros travados, será emitida uma mensagem de erro.

O bloco de parâmetros atual pode ser acessado com o uso das variáveis de sistema

\$AA_SCPAR[<Eixo>].



Informações adicionais

- Até a SW 5.1, o bloco de parâmetros servo podia ser especificado somente através do PLC (DB3n DBB9 Bit0–2). Para G33, G331 e G332, o bloco de parâmetros mais adequado era selecionado pelo controle.
- Caso o **bloco de parâmetros servo** deve ser comutado tanto pelo programa NC, em uma ação síncrona e no PLC, o programa de PLC deve prever estas condições.
- **Referências:** /FB1/V1 Avanços



Exemplo de programação

```

...
N110 SCPARA[X]= 3           O terceiro bloco de parâmetros é transferido para o eixo X
...

```


Frames

| | | |
|-------|---|-------|
| 6.1 | Transformação de coordenadas através de variáveis frame | 6-192 |
| 6.2 | Variáveis frame/carga de valores em variáveis frame | 6-197 |
| 6.3 | Deslocamento grosso/fino (coarse/fine) | 6-204 |
| 6.4 | Deslocamento DRF | 6-205 |
| 6.5 | Deslocamento de origem externo | 6-206 |
| 6.6 | Programando deslocamentos preset, PRESETON | 6-207 |
| 6.7 | Desativando frames | 6-209 |
| 6.8 | Cálculo do frame a partir de três pontos de medição na área, MEAFRAME | 6-210 |
| 6.9 | Frames globais NCU (SW 5 em diante) | 6-213 |
| 6.9.1 | Frames específicos do canal | 6-214 |
| 6.9.2 | Frames ativos no canal | 6-216 |

6.1 Transformação de coordenadas através de variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D

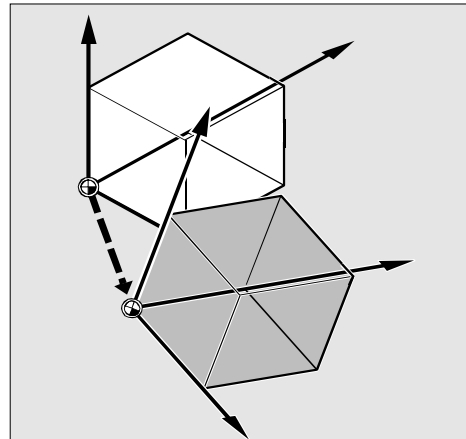


840Di

6.1 Transformação de coordenadas através de variáveis frame

Transformação de coordenadas com variáveis frame

Em adição às opções já descritas no Manual de Programação “Fundamentos”, você pode também criar sistemas de coordenadas com variáveis frame pré definidas



Sistemas de coordenadas

Os seguintes sistemas de coordenadas são definidos:

MCS: Sistema de coordenadas da máquina

BCS Sistema de coordenadas básico

BOS: Sistema de origem básico

SZS: Sistema zero ajustável

WCS: Sistema de coordenadas da peça

O que é uma variável frame pré definida?

Variáveis frame pré definidas são instruções cujo uso e efeito encontram-se definidos na linguagem de programação e podem ser processadas pelo programa NC.

Possíveis variáveis frame:

- Frame base (deslocamento de origem base)
- Frames ajustáveis
- Frames programáveis

6.1 Transformação de coordenadas através de variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Variáveis frame/relação entre frames

A transformação de coordenadas pode ser ativada através da atribuição do valor de um frame a uma variável frame.

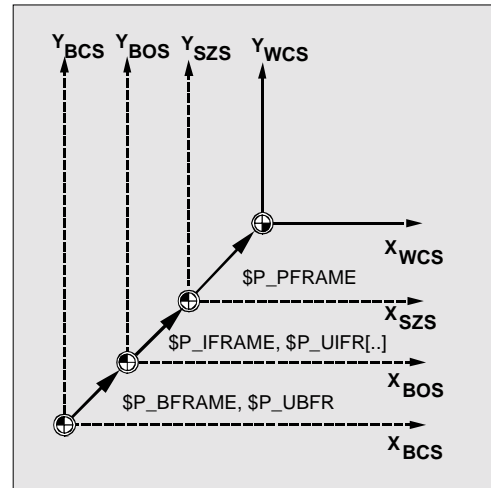
Exemplo: $\$P_PFRAME = CTRANS(X, 10)$

Variável frame:

$\$P_PFRAME$ significa: frame atual programável.

Frame:

$CTrans(X, 10)$ significa: deslocamento de origem programável de X em 10 mm.



Leitura dos valores atuais

Os valores atuais do sistema de coordenadas podem ser lidos através de variáveis pré definidas no programa:

$\$AA_IM[eixo]$ Leitura do valor atual no MCS

$\$AA_IB[eixo]$ Leitura do valor atual no BCS

$\$AA_IBN[eixo]$ Leitura do valor atual no BOS

$\$AA_IEN[eixo]$ Leitura do valor atual no SZS

$\$AA_IW[eixo]$ Leitura do valor atual no WCS

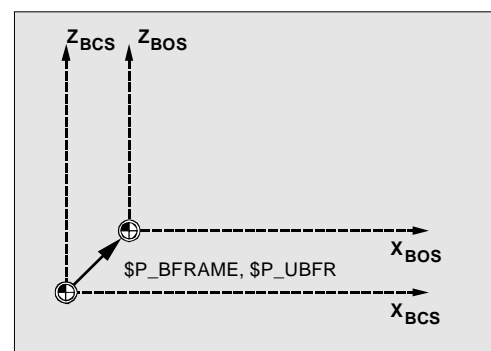
Relação das variáveis frame pré definidas

$\$P_BFRAME$

Variável frame base que estabelece a relação entre o sistema de coordenadas base (BCS) e o sistema de origem básico (BOS).

Para que o frame base descrito através de $\$P_UBFR$ seja imediatamente ativado no programa,

- Você deve programar um G500, G54...G599, ou
- Você deve descrever o $\$P_BFRAME$ com $\$P_UBFR$.



6.1 Transformação de coordenadas através de variáveis frame



840D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

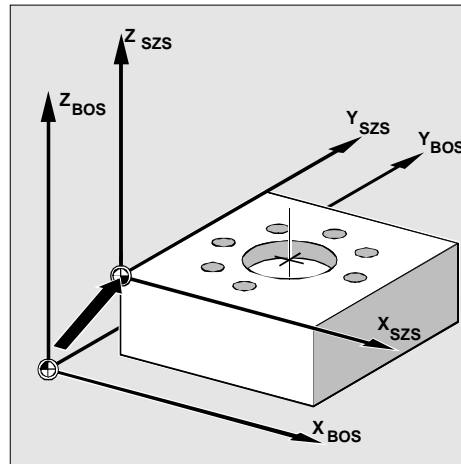
NCU 573

\$P_IFRAME

Variável frame atual e ajustável que estabelece a referência entre o sistema original base (BOS) e sistema de deslocamento de origem ajustável (SZS).

\$P_IFRAME corresponde à \$P_UIFR[\$P_IFRNUM]

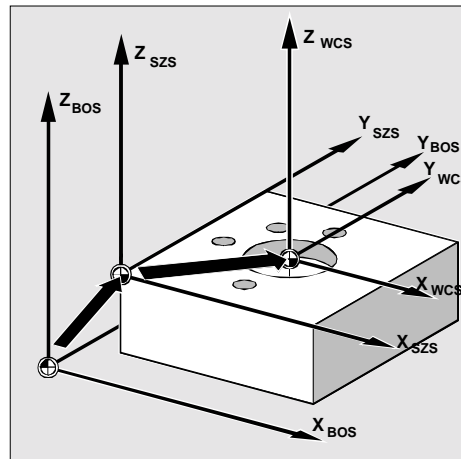
Após a programação de G54, por exemplo, \$P_IFRAME contém a translação, rotação, escala e espelhamento definidos por G54.



\$P_PFRAME

Variável frame atual e ajustável que estabelece a referência entre sistema de deslocamento de origem ajustável (SZS) e o sistema de coordenadas da peça (WCS).

\$P_PFRAME contém o frame resultante da programação de TRANS/ATRANS, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR ou os valores para CTRANS, CROT, CMIRROR, CSCALE do FRAME programável.



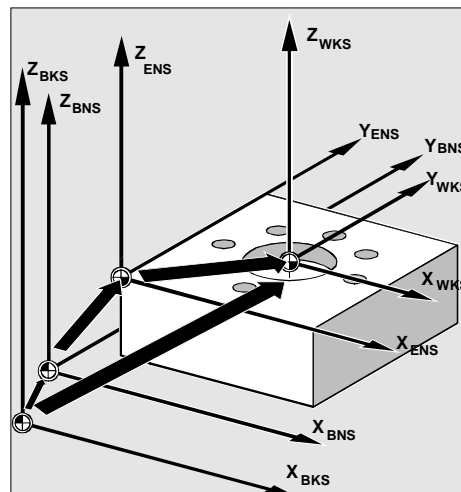
\$P_ACTFRAME

O frame total atual resulta do encadeamento da variável frame base atual \$P_BFRAME, da variável frame ajustável atual \$P_IFRAME e da variável frame atualmente programada \$P_PFRAME.

\$P_ACTFRAME descreve o zero atual válido da peça.

Caso \$P_IFRAME, \$P_BFRAME ou \$P_PFRAME sejam alterados, \$P_ACTFRAME é recalculado.

\$P_ACTFRAME corresponde à
\$P_BFRAME : \$P_IFRAME : \$P_PFRAME



6.1 Transformação de coordenadas através de variáveis frame

840D
NCU 571840D
NCU 572

FM-NC

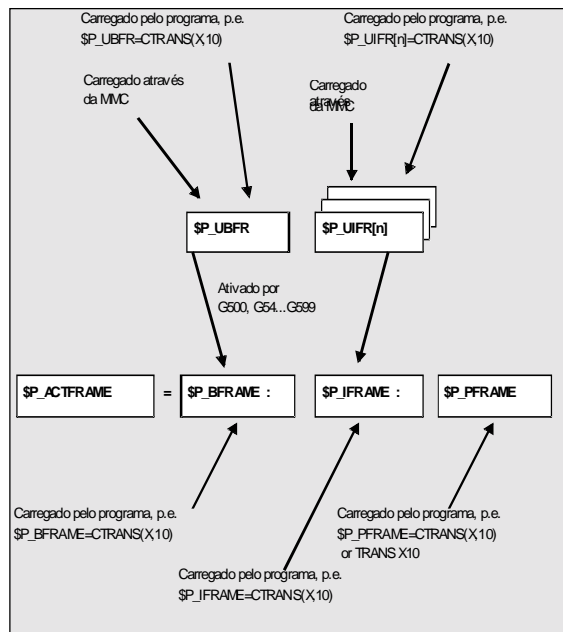


810D



840Di

NCU 573



O frame base e o ajustável serão ativados após Reset caso MD 20110

RESET_MODE_MASK seja programada da seguinte forma

Bit0=1, bit14=1 --> \$P_UBFR (frame base) efetivo

Bit0=1, bit5=1 --> \$P_UIFR [\$P_UIFRNUM] (frame ajustável) efetivo

Frames ajustáveis pré definidos \$P_UBFR

O frame base é programado com \$P_UBFR, mas não ativo simultaneamente no programa. O frame base programado com \$P_UBFR será incluído no cálculo se

- O reset for ativado e os bits 0 e 14 estiverem incluídos em MD RESET_MODE_MASK e
- As instruções G500, G54...G599 forem executadas.

Frames ajustáveis pré definidos \$P_UIFR[n]

A variável frame pré definida \$P_UIFR[n] pode ser usada para ler ou escrever os deslocamentos de origem ajustáveis, de G54 a G599 partindo do programa.

Estas variáveis estão disponíveis em um array de uma dimensão de tipo FRAME, chamado \$P_UIFR[n].

Escrita de valores nos comandos G

Cinco frames ajustáveis estão disponíveis como standard

\$P_UIFR[0]...\$P_UIFR[4] ou 5 comandos G de mesmo significado – G500 e G54 a G57 – nos quais valores podem ser gravados.

6.1 Transformação de coordenadas através de variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

$\$P_IFRAME = \$P_UIFR[0]$ corresponde a G500

$\$P_IFRAME = \$P_UIFR[1]$ corresponde a G54

$\$P_IFRAME = \$P_UIFR[2]$ corresponde a G55

$\$P_IFRAME = \$P_UIFR[3]$ corresponde a G56

$\$P_IFRAME = \$P_UIFR[4]$ corresponde a G57

A quantidade de frames pode ser alterada através de dados de máquina:

$\$P_IFRAME = \$P_UIFR[5]$ corresponde a G505

... ..

$\$P_IFRAME = \$P_UIFR[99]$ corresponde a G599



Isto permite a você gerar até 100 sistemas de coordenadas que podem ser chamados de forma global em programas diferentes, para uso, por exemplo, como ponto zero para várias fixações.



Variáveis frame devem ser programadas em blocos de programa separados.

Exceção: Programação de frames ajustáveis com G54, G55, ...

6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

6.2 Variáveis frame/carga de valores em variáveis frame



Valores podem ser carregados diretamente, vários frames podem ser encadeados ou copiados para outros frames no programa.

Carga direta de valores



Programação

```
$P_PFRAME=CTTRANS (X, valor p/eixo, Y, valor p/eixo, Z, valor p/eixo, ...)
```

```
$P_PFRAME=CROT (X, ângulo, Y, ângulo, Z, ângulo, ...)
```

```
$P_PFRAME=CSCALE (X, escala, Y, escala, Z, escala, ...)
```

```
$P_PFRAME=CMIRROR (X, Y, Z)
```



A programação de \$P_BFRAME é feita de forma análoga a de \$P_PFRAME.



Explicação dos comandos

| | |
|---------|--|
| CTTRANS | Translação de eixo específico |
| CROT | Rotação em torno de eixo específico |
| CSCALE | Alteração de escala em eixo específico |
| CMIRROR | Reverte a direção de eixo específico |



Função

Você pode usar estas funções para carregar os valores em frames/variáveis frame a partir do programa NC.



Seqüência

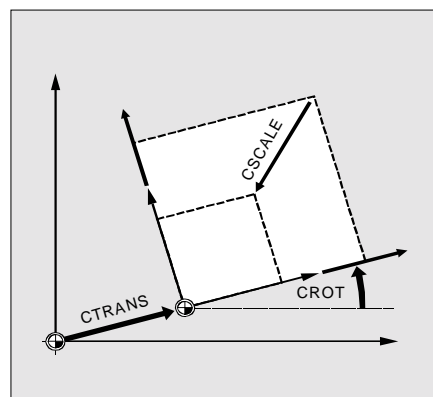
Você pode programar diversas regras aritméticas em seqüência.

Exemplo:

```
$P_PFRAME=CTTRANS (...) : CROT (...) : CSCALE...
```

Favor notar que os comandos devem ser conectados através do operador de encadeamento “dois pontos” - (...):(...).

Isto faz com que os comandos sejam inicialmente conectados e posteriormente executados de forma aditiva na seqüência programada.



6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Informações adicionais

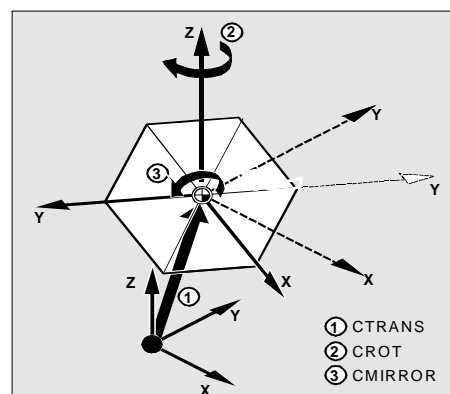
Os valores programados com os comandos acima gravam valores nos frames.

Os valores não serão ativados até atribuídos ao frame através de uma variável ativa \$P_BFRAME ou \$P_PFRAME.



Exemplo de programação

Translação, rotação e espelhamento são ativados através da atribuição de valores ao frame atualmente programado.



```
N10 $P_PFRAME=CTrans(X,10,Y,20,Z,5):CROT(Z,45):CMIRROR(Y)
```


6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Leitura e encadeamento de componentes frame



Programação (exemplos)

```
R10=$P_UIFR[$P_UIFRNUM, X, RT]
```

Carrega o ângulo de rotação RT em torno do eixo X a partir do deslocamento de origem ajustável \$P_UIFRNUM para a variável R10.

```
R12=$P_UIFR[25, Z, TR]
```

Carrega o valor de deslocamento TR em Z a partir dos dados do frame número 25 para a variável R12.

```
R15=$P_PFRAME[Y, TR]
```

Carrega o valor de deslocamento TR em Y do frame programável atual para a variável R15.

```
$P_PFRAME[X, TR]=25
```

Modifica o valor de deslocamento TR em X do frame programável atual. X25 é imediatamente atribuído.



Explicação dos comandos

| | |
|---------------------|--|
| \$P_UIFRNUM | Este comando estabelece de forma automática a referência para o deslocamento de origem ajustável válido. |
| P_UIFR[n, ..., ...] | Especifica o número do frame n para acessar o frame ajustável número n. |
| TR | Especifica o componente a ser lido ou alterado: Translação TR, translação fina FI, rotação RT, alteração de escala SC, espelhamento MI. O eixo correspondente é também especificado (vide exemplos). |
| FI | |
| RT | |
| SC | |
| MI | |

6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Função

Esta característica permite que você acesse dados **individuais** de um frame, p.e. um valor de deslocamento específico ou ângulo de rotação.

Estes valores podem ser alterados ou atribuídos a outra variável.



Seqüência

Chamada do frame

Ao especificar a variável de sistema `$P_UIFRNUM` você pode acessar o deslocamento de origem atual com o uso de `$P_UIFR` ou `G54, G55, ...` (`$P_UIFRNUM` contém o número do frame atual em uso).

Todos os outros frames ajustáveis `$P_UIFR` são chamados através do número apropriado `$P_UIFR[n]`.

Para variáveis frame pré definidas e frames de usuário, especifique o nome, p.e. `$P_IFRAME`.

Chamada dos dados

O nome do eixo e o componente frame do valor que você quer acessar ou alterar é escrito entre colchetes, p.e. `[X, RT]` ou `[Z, MI]`.

6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Conectando componentes frames

Um frame completo pode ser carregado em outro frame.



Programação (exemplos)

```
DEF FRAME SETTING1
SETTING1=CTTRANS(X,10)
$P_PFRAME=SETTING1
```

```
DEF FRAME SETTING4
SETTING4=$P_PFRAME
$P_PFRAME=SETTING4
```

Carrega os valores do SETTING1 ao frame atual programável.

O frame atual programável é gravado temporariamente e pode ser chamado novamente.



Informações adicionais

Faixa de valores para rotação RT

Rotação em torno do primeiro eixo geométrico: -180° to $+180^\circ$

Rotação em torno do segundo eixo geométrico: -89.999° to $+90^\circ$

Rotação em torno do terceiro eixo geométrico: -180° to $+180^\circ$

Encadeamento de frames



Programação (exemplos)

```
$P_IFRAME=$P_UIFR[15]:$P_UIFR[16]
```

```
$P_UIFR[3]=$P_UIFR[4]:$P_UIFR[5]
```

$\$P_UIFR[15]$ contém, por exemplo, dados de deslocamento de origem. Os dados de $\$P_UIFR[16]$, p.e. dados de rotação, são processados em seguida de forma aditiva.

O frame ajustável 3 é gerado através do encadeamento dos frames 4 e 5.

6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Função

O encadeamento de frames é aplicado para descrição de diversas peças, montadas sobre um palete a serem usinadas em um mesmo processo.



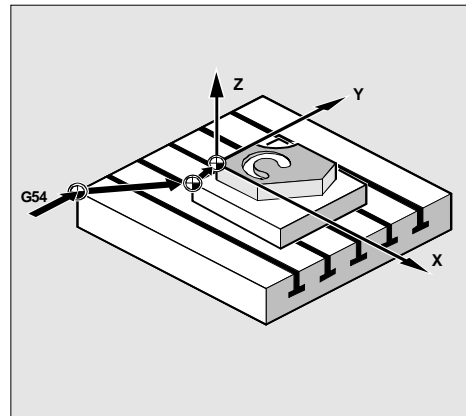
Seqüência

Os frames são encadeados na seqüência programada. Os componentes do frame (translações, rotações, etc.) são executados de forma aditiva.



Os componentes frame podem somente conter valores intermediários para descrição das tarefas nas peças do palete. Estes podem ser encadeados para gerar os vários zeros peça.

Favor constatar que os frames devem ser conectados um a um através do operador “dois pontos” (:).



6.2 Variáveis frame/carga de valores em variáveis frame



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Definição de um novo frame



Programação

```
DEF FRAME PALLET1
```

```
PALLET1=CTrans ( ... ) : CROT ( ... ) ...
```



Função

De forma adicional aos frames ajustáveis pré definidos descritos acima, você tem ainda a opção de criar novos frames.

Isto é realizado através da criação de variáveis tipo FRAME para as quais podem ser atribuídos nomes de sua escolha.



Seqüência

Você pode utilizar as funções CTRANS, CROT, CSCALE e CMIRROR para atribuir valores a seus frames no programa NC.

Você irá encontrar mais informações a este respeito nas páginas anteriores.

6.3 Deslocamento grosso/fino (coarse/fine)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

6.3 Deslocamento grosso/fino (coarse/fine)



Função

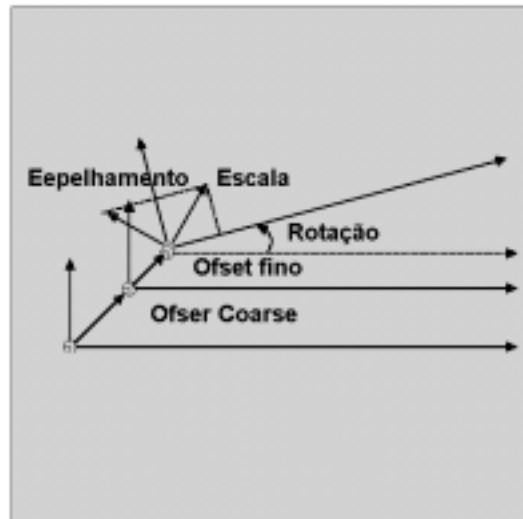
Deslocamento fino

O deslocamento fino dos frames base e de todos os outros frames ajustáveis pode ser programado através do comando `CFINE(X, ..., Y, ...)`.

Deslocamento grosso

O deslocamento grosso é definido através de `CTRANS(...)`.

Os deslocamentos grosso e fino são adicionados ao deslocamento total.



Programação

```
$P_UBFR=CTRANS(x, 10) : CFINE(x, 0.1) : CROT(x, 45) ; encadeamento dos deslocamentos
                                     fino e rotação
$P_UIFR[1]=CFINE(x, 0.5, y, 1.0, z, 0.1) ; o frame total é sobrescrito com
                                     CFINE, incluindo o deslocamento grosso.
```

Acesso às componentes individuais do deslocamento fino é atingido através da especificação da componente FI gravada.



Programação

```
DEF REAL FINEX ; Definição da variável FINEX
FINEX=$P_UIFR[$P_UIFRNUM, x, FI] ; Leitura do deslocamento fino através da
                                     variável FINEX
FINEX=$P_UIFR[3, X, FI] ; Leitura do deslocamento fino do eixo X no terceiro frame através da
                                     variável FINEX
```

O deslocamento fino pode somente ser utilizado caso o MD18600: `MM_FRAME_FINE_TRANS=1`.

Um deslocamento fino carregado pelo operador será ativado somente após o frame correspondente ser ativado, ou seja, a ativação é realizada através de G500, G54...G599. A ativação de um deslocamento fino de um frame permanece ativo enquanto o frame estiver ativo.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



O frame programável não possui deslocamento fino. Caso o frame programável seja carregado em um frame com deslocamento fino, então o deslocamento total será estabelecido através da soma entre os deslocamentos fino e grosso. Durante a leitura do frame programável o deslocamento fino será sempre zero.



Fabricante da máquina

SW 5 em diante

O deslocamento fino pode ser configurado através do MD18600 MM_FRAME_FINE_TRANS com as seguintes variantes:

0: deslocamento fino não pode ser carregado ou programado.

G58 e G59 não são possíveis.

1: O deslocamento fino para frames ajustáveis, base, programáveis, G58 e G59 podem ser carregados/programados

6.4 Deslocamento DRF

Deslocamento com o uso da manivela eletrônica, DRF

Em adição a todas as translações descritas nesta seção, você pode também definir deslocamentos de origem através da manivela eletrônica (deslocamento DRF).

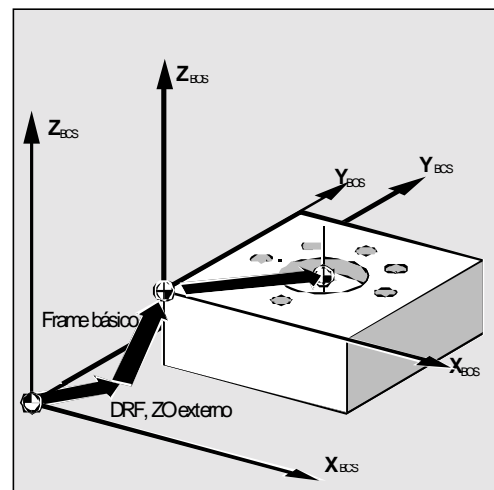
O deslocamento DRF atua no sistema de coordenadas base. Vide diagrama para as relações.

Você vai encontrar maiores informações no manual de operação.



Cancelando o deslocamento DRF, DRFOF

DRFOF cancela/apaga os deslocamentos DRF para todos os eixos do canal. DRFOF deve ser programado em um bloco NC separado.



6.5 Deslocamento de origem externo



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

6.5 Deslocamento de origem externo

Deslocamento de origem externo

Esta é outra forma de se movimentar o ponto zero entre o sistema de coordenadas básico e sistema de coordenadas da peça.

Somente translações lineares podem ser programadas com deslocamentos de origem externos.

Programando valores de deslocamento, \$AA_ETRANS

Os valores de deslocamento são programados através da carga de valores em variáveis de sistema específicas dos eixos.

Carga de valor de deslocamento

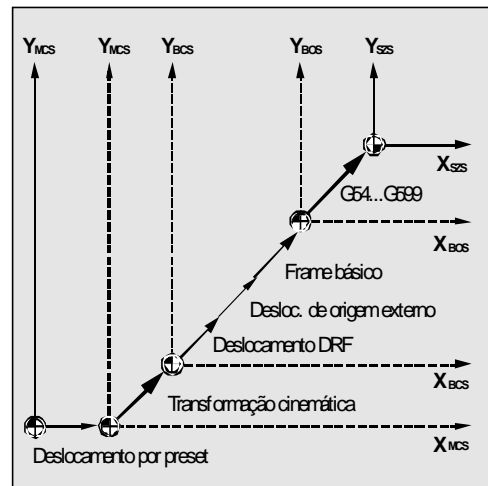
`$AA_ETRANS[eixo]=Ri`

R_i é uma variável aritmética do tipo REAL que contém um novo valor.

O deslocamento externo é geralmente ligado através do PLC e não especificado no programa.



O valor carregado no programa somente será ativado quando o sinal correspondente for habilitado na interface VDI (interface NCU-PLC).



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

6.6 Programando deslocamentos preset, PRESETON



Programação

PRESETON (AXIS , VALUE , ...)



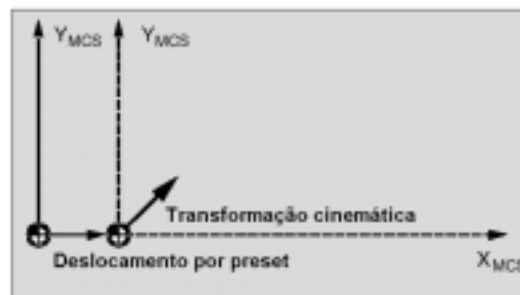
Explicação dos comandos

| | |
|----------|--|
| PRESETON | Carrega valor atual |
| Eixo | Parâmetro eixo de máquina |
| Valor | Novo valor a ser aplicado no eixo específico |



Função

Em aplicações especiais, pode ser necessário carregar um novo valor de posição atual a um eixo ou mais (com eixos parados).



Sequência

Os valores atuais são atribuídos ao sistema de coordenadas da máquina – os valores se referem a eixos de máquina.

Exemplo:

N10 G0 A760

N20 PRESETON(A1, 60)

O eixo A é posicionado em 760. Na posição 760, o eixo de máquina A1 será carregado com um novo valor atual de posição 60.

A partir deste ponto, o posicionamento é realizado de acordo com a nova posição do eixo.



O ponto de referência torna-se inválido após a instrução PRESETON. Esta função deve ser utilizada somente para os eixos que não necessitam de referenciamento.

Caso o sistema original seja reativado, o ponto de referência deve ser novamente ajustado com G74 –

6.5 Deslocamento de origem externo



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC

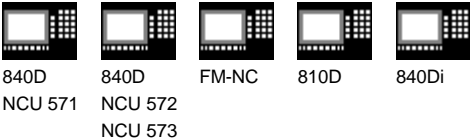


810D



840Di

vide seção 3.1.



6.7 Desativando frames

=?

Explicação dos comandos

| | |
|-------|---|
| DRFOF | Desativa (limpa) os corretores introduzidos via manivela eletrônica (DRF) |
| G53 | Desativação não modal de todos os frames programáveis e ajustáveis |
| G153 | Desativação não modal dos frames: programáveis, base e todos os ajustáveis |
| SUPA | Desativação não modal de todos os frames: programáveis, base, ajustáveis e dos corretores introduzidos via manivela eletrônica (DRF). |

Informações adicionais

Os frames programáveis são cancelados através da atribuição de um “frame zero” (sem especificação de eixos) ao frame programável.

Exemplo:

```

$P_PFRAME=TRANS( )
$P_PFRAME=ROT( )
$P_PFRAME=SCALE( )
$P_PFRAME=MIRROR( )
    
```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

6.8 Cálculo do frame a partir de três pontos de medição na área, MEAFRAME



MEAFRAME é uma expansão da linguagem de programação do 840D utilizada para os ciclos de medição.

Esta função encontra-se disponível a partir da SW4.3.



Função

Quando uma peça é posicionada para usinagem, sua posição ideal geralmente encontra-se deslocada e/ou rotacionada em relação ao sistema de coordenadas cartesiano da máquina.

Para uma usinagem exata, podem ser necessárias medições e mesmo ajustes físicos dispendiosos.

Pode ser necessário ainda, a alteração de coordenadas no programa.

Um frame pode ser determinado a partir de três pontos medidos na área a partir da qual a posição ideal é conhecida. As medições podem ser realizadas por sensores táteis ou ópticos, a partir do toque em furos especiais ou esferas posicionadas de forma precisa na placa de fixação da peça.

A função MEAFRAME calcula o frame a partir de três pontos ideais e três pontos medidos.

Para proporcionar o mapeamento das coordenadas com relação às coordenadas ideais utilizando rotação e translação, o triângulo formado pelos pontos medidos deve ser congruente ao triângulo ideal. Esta relação será gravada sob a forma de um algoritmo de compensação, que minimiza a soma dos quadrados necessários para redefinir a forma do triângulo medido com relação ao triângulo ideal.

Considerando que a distorção efetiva possa ser utilizada para avaliar a qualidade da medição, MEAFRAME devolve uma variável adicional.

6.8 Cálculo do frame a partir de três pontos de medição. MEAFRAME840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Programação**

MEAFRAME (IDEAL_POINT , MEAS_POINT , FIT_QUALITY)

**Explicação dos comandos**

| | |
|-------------|---|
| MEAFRAME | Cálculo do frame de 3 pontos no espaço |
| IDEAL_POINT | Array de 2 dimensões tipo REAL contendo as três coordenadas dos pontos ideais |
| MEAS_POINT | Array de 2 dimensões tipo REAL contendo as três coordenadas dos pontos medidos |
| FIT_QUALITY | <p>Variável tipo REAL que fornece as seguintes informações:</p> <ul style="list-style-type: none"> –1: Os pontos ideais estão localizados próximos à uma linha reta. O frame não pode ser calculado. As variáveis de retorno contém um frame neutro. –2: Os pontos ideais estão localizados aproximadamente em uma linha reta. O frame não pode ser calculado. As variáveis de retorno contém um frame neutro. –4: O cálculo da rotação falhou por qualquer outro motivo <p>Valor positivo:</p> <p>Soma das distorções (distância entre os pontos) necessária para transformar o triângulo medido em um congruente ao ideal.</p> |

**Exemplo de aplicação**

```

; Programa de usinagem 1
;
DEF FRAME CORR_FRAME
;
; Definindo os pontos de medição
DEF REAL IDEAL_POINT[3,3] = SET(10.0,0.0,0.0, 0.0,10.0,0.0, 0.0,0.0,10.0)
DEF REAL MEAS_POINT[3,3] = SET(10.1,0.2,-0.2, -0.2,10.2,0.1, -0.2,0.2, 9.8);      para teste
DEF REAL FIT_QUALITY = 0
;
DEF REAL ROT_FRAME_LIMIT = 5;      permite um máximo de 5 rotações na posição peça
DEF REAL FIT_QUALITY_LIMIT = 3;    permite um máximo de 3 mm de distorção entre os triângulos
                                   ideal e medido

DEF REAL SHOW_MCS_POS1[3]
DEF REAL SHOW_MCS_POS2[3]
DEF REAL SHOW_MCS_POS3[3]
; =====
;
N100 G01 G90 F5000
N110 X0 Y0 Z0
;
N200 CORR_FRAME=MEAFRAME(IDEAL_POINT,MEAS_POINT,FIT_QUALITY)
;

```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

```

N230 IF FIT_QUALITY < 0
SETAL(65000)
GOTO NO_FRAME
ENDIF

;
N240 IF FIT_QUALITY > FIT_QUALITY_LIMIT
SETAL(65010)
GOTO NO_FRAME
ENDIF

;
N250 IF CORR_FRAME[X,RT] > ROT_FRAME_LIMIT;          limita o primeiro ângulo RPY
SETAL(65020)
GOTO NO_FRAME
ENDIF

;
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT;          limita o segundo ângulo RPY
SETAL(65021)
GOTO NO_FRAME
ENDIF

;
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT;          limita o terceiro ângulo RPY
SETAL(65022)
GOTO NO_FRAME
ENDIF

;
N300 $P_IFRAME=CORR_FRAME;          ativa o frame de apalpador através de um frame ajustável
;
; verifica o frame através do posicionamento dos eixos geométricos nos pontos ideais
;
N400 X=IDEAL_POINT[0,0] Y=IDEAL_POINT[0,1] Z=IDEAL_POINT[0,2]
N410 SHOW_MCS_POS1[0]=$AA_IM[X]
N420 SHOW_MCS_POS1[1]=$AA_IM[Y]
N430 SHOW_MCS_POS1[2]=$AA_IM[Z]
;
N500 X=IDEAL_POINT[1,0] Y=IDEAL_POINT[1,1] Z=IDEAL_POINT[1,2]
N510 SHOW_MCS_POS2[0]=$AA_IM[X]
N520 SHOW_MCS_POS2[1]=$AA_IM[Y]
N530 SHOW_MCS_POS2[2]=$AA_IM[Z]
;
N600 X=IDEAL_POINT[2,0] Y=IDEAL_POINT[2,1] Z=IDEAL_POINT[2,2]
N610 SHOW_MCS_POS3[0]=$AA_IM[X]
N620 SHOW_MCS_POS3[1]=$AA_IM[Y]
N630 SHOW_MCS_POS3[2]=$AA_IM[Z]
;
N700 G500;          desativa o frame ajustável, ativa um frame zero (sem valor definido)
;
NO_FRAME:
M0
M30

```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

6.9 Frames globais NCU (SW 5 em diante)



Função

Os frames globais NCU encontram-se disponíveis somente uma vez para todos os canais de cada NCU. Os frames globais podem ser gravados e utilizados em todos os canais. Os frames globais NCU são ativados no respectivo canal.

Deslocamentos, escalas e espelhamentos podem ser aplicados aos eixos do canal e de máquina, através do uso dos frames globais.

Os frames globais não possuem uma relação geométrica entre os eixos. Não é possível, portanto, realizar rotações ou programar identificadores de eixos geométricos.



- Não é possível utilizar os frames globais para rotações. Caso uma rotação seja programada, será rejeitada e o alarme: "18310 canal%1block%2 frame: rotação não permitida" será exibido.
- O encadeamento de frames globais e específicos do canal é possível. O frame resultante contém todos os elementos, incluindo rotações para todos os eixos. Caso um frame com elementos de rotação seja atribuído à um frame global, será rejeitado e o alarme "Frame: rotação não permitida" emitido.

Frames globais NCU: \$P_NCBFR[n]

Você pode configurar até 8 frames globais NCU.



Fabricante da máquina

A quantidade de frames base globais é configurada em dados de máquina. (Vide /FB/ K2, Eixos, Sistemas de coordenadas, Frames)

Os frames base específicos para o canal podem ser ativados simultaneamente.

6.9 Frames globais NCU (SW 5 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Os frames globais podem ser escritos e lidos por todos os canais de uma NCU. Durante a escrita de frames globais, o usuário deve Ter o cuidado de sincronizar os canais, com, por exemplo, marcas wait (WAITMC).

Frames globais NCU ajustáveis: \$P_UIFR[n]

A configuração de todos os frames ajustáveis G500, G54...G599 pode ser feita tanto de forma global para toda a NCU ou apenas para um canal.



Fabricante de máquina

Todos os frames ajustáveis podem ser reconfigurados como frames globais através do MD18601 MM_NUM_GLOBAL_USER_FRAMES. Vide /FB/ K2, Eixos, Sistemas de coordenadas, Frames.

Os identificadores de eixos do canal e da máquina podem ser utilizados para os comandos de programa envolvendo os frames. A programação de identificadores geométricos será rejeitada com a emissão de um alarme.

6.9.1 Frames específicos do canal



Função

A quantidade de frames base pode ser configurada através do MD28081 MM_NUM_BASE_FRAMES. O comando é fornecido inicialmente com um frame por canal. Um máximo de 8 pode ser configurado por canal. Adicionalmente aos 8 base, outros 8 tipo NCU globais podem ser configurados no canal.



Os frames ajustáveis/base podem lidos e escritos através de

- PLC
- Programa de usinagem e
- OPI.

Os frames globais possuem ajuste fino.

A supressão de frames globais pode ser feita, no caso dos frames específicos do canal, através de G53, G153, SUPA e G500.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

\$P_CHBFR[n]

Os frames base podem ser lidos e escritos através da variável de sistema \$P_CHBFR[n]. Durante a escrita de um frame base, o encadeamento de todos os frames base não é ativado; será ativado somente quando as instruções G500, G54..G599 forem executadas. A variável serve principalmente como memória para o processo de escrita do frame base para a MMC e PLC. O conteúdo destas variáveis frame são gravadas durante o back up de dados.

Primeiro frame base no canal

A escrita na variável pré definida \$P_UBFR não ativa o array de frame base com índice 0, pois esta ativação ocorre somente quando as instruções G500, G54..G599 forem executadas. A variável pode também ser lida e escrita no programa.

\$P_UBFR

\$P_UBFR é idêntica à \$P_CHBFR[0].

No estado inicial de fornecimento, existe sempre um frame base de forma a tornar compatível o sistema com as versões anteriores. Caso não exista nenhum frame base específico de canal, será exibido o alarme: "Frame: instrução não permitida".

6.9 Frames globais NCU (SW 5 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

6.9.2 Frames ativos no canal



Função

\$P_NCBFRAME[n]

Frames base globais NCU atuais

Os elementos do array atual do frame base global podem ser lidos e escritos através da variável de sistema \$P_NCBFRAME[n]. O resultado total do frame base é calculado através do processo de escrita no canal.

O frame alterado será ativado somente no canal onde programado. Caso o frame deva ser alterado por outros canais da NCU, ambos [n] e \$P_NCBFRAME[n] devem ser programados. Os outros canais devem manter o frame ativo, com, por exemplo, com G54. Durante a escrita de um frame base, o frame base total é calculado novamente.

\$P_CHBFRAME[n]

Frames base atuais do canal

Os elementos do array frame base atual do canal podem ser lidos e escritos através da variável de sistema \$P_CHBFRAME[n]. O resultado total do frame base é calculado através do processo de escrita no canal. Durante a escrita de um frame base, o frame base total será novamente calculado.

\$P_BFRAME

Primeiro frame base no canal

O frame base atual pode ser escrito e lido no programa de usinagem através da variável de sistema \$P_BFRAME com o índice de array 0 válido no programa de usinagem. O frame base escrito é imediatamente incluído no cálculo.

\$P_BFRAME é idêntico à \$P_CHBFRAME[0]. Por definição, a variável de sistema possui sempre um valor válido. Caso não exista nenhum frame base de canal, será emitido, durante a leitura/escrita o alarme: "Frame: instrução não permitida".

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

\$P_ACTBFRAME

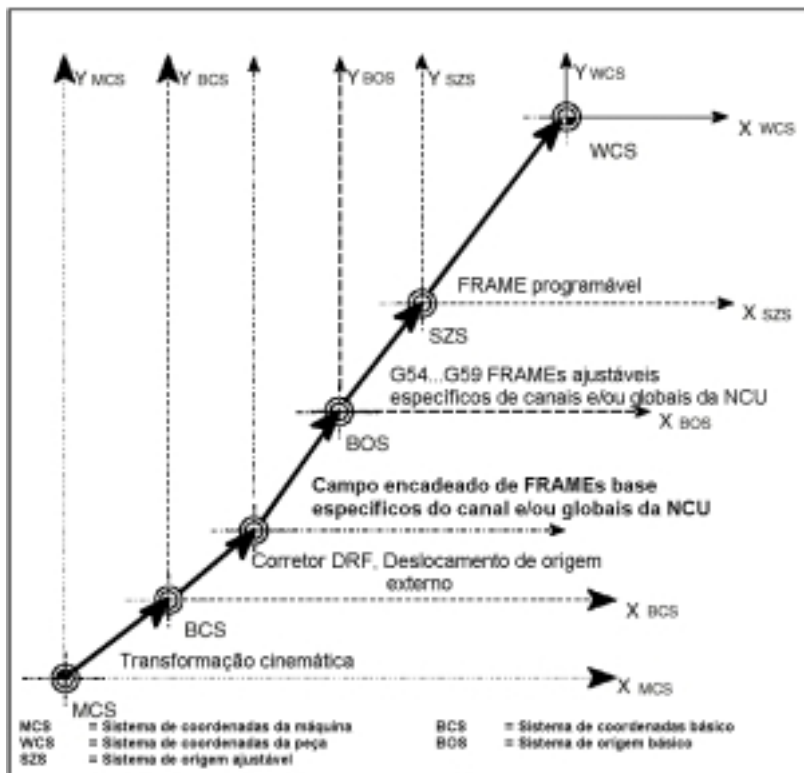
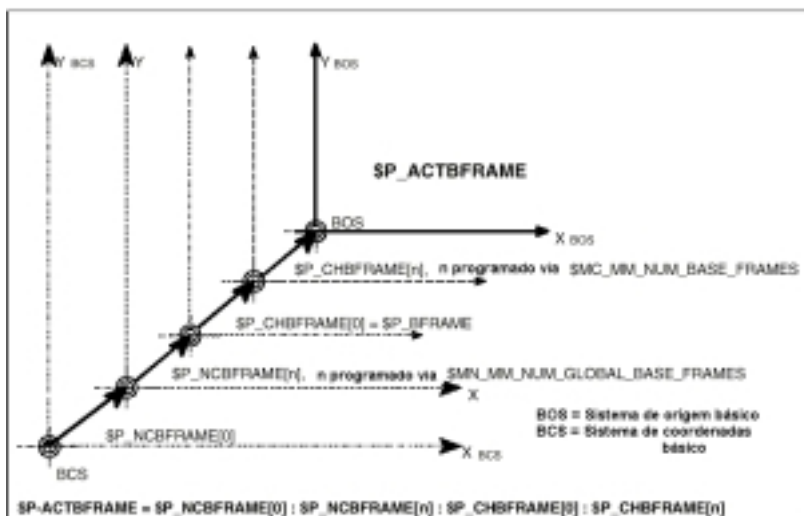
Frame base total

A variável \$P_ACTBFRAME determina o encadeamento total de frames base. A variável pode somente ser lida.

\$P_ACTBFRAME corresponde à

\$P_NCBFRAME[0] : ... : \$P_NCBFRAME[n] :

\$P_CHBFRAME[0] : ... : \$P_CHBFRAME[n].



6.9 Frames globais NCU (SW 5 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Programando o frame base total

O usuário pode selecionar através das variáveis de sistema `$P_CHBFRMASK` e `$P_NCBFRMASK` quais frames base devem ser incluídos no cálculo do frame base "Total". As variáveis podem somente serem programadas no programa de usinagem e lidas através OPI. O valor da variável é interpretado como uma máscara binária específica cujo elemento de array do frame base `$P_ACTBFRAME` deva ser incluída no cálculo.

Você pode especificar através de `$P_CHBFRMASK` quais os frames base globais NCU devem ser incluídos no cálculo.

Através da programação de variáveis o frame base total e o frame total serão recalculados. Após um reset, o valor básico é

`$P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK` e
`$P_NCBFRMASK = $MN_NCBFRAME_RESET_MASK`.

p.e.

`$P_NCBFRMASK = 'H81' ; $P_NCBFRAME[0] : $P_NCBFRAME[7]`
`$P_CHBFRMASK = 'H11' ; $P_CHBFRAME[0] : $P_CHBFRAME[4]`

\$P_IFRAME

Frame ajustável atual

O frame ajustável atual válido no canal pode ser escrito e lido através do programa de usinagem através da variável de sistema `$P_IFRAME`. O frame ajustável escrito será imediatamente incluído no cálculo.

Com os frames NCU globais ajustáveis, o frame alterado será somente ativado no canal no qual o frame tenha sido programado. Caso o frame deva ser alterado para todos os canais em uma NCU, devem ser programados ambos `$P_UIFR[n]`. Os outros canais devem manter o respectivo frame ativo, com, por exemplo, G54.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

\$P_PFRAME

Frame atual programável

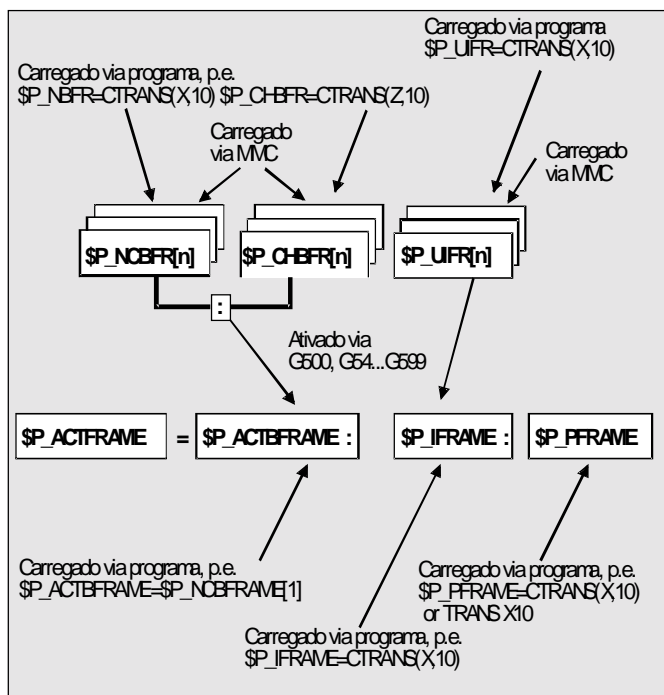
\$P_PFRAME é o frame programável resultante da programação de TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR ou das definições de CTRANS, CROT, CMIRROR, CSCALE ao frame programado.

É a variável frame atual programável que estabelece a referência entre o deslocamento de origem ajustável (SZS) e o sistema de coordenadas da peça (WCS).

\$P_ACTFRAME

O resultado atual do frame total \$P_ACTFRAME resulta do encadeamento de todos os frames base, do frame ajustável atual e do frame programável. O frame atual é sempre atualizado caso um elemento frame seja alterado.

\$P_ACTFRAME corresponde à \$P_ACTBFRAME : \$P_IFRAME : \$P_PFRAME



Transformações

| | | |
|-------|--|-------|
| 7.1 | Transformações em três, quatro e cinco eixos: TRAORI..... | 7-220 |
| 7.1.1 | Programando a orientação da ferramenta | 7-223 |
| 7.1.2 | Eixos de orientação, ORIWCS, ORIMCS | 7-228 |
| 7.1.3 | Posições singulares e como elas são manuseadas | 7-229 |
| 7.1.4 | Eixos de orientação (SW 5.2 em diante) | 7-230 |
| 7.1.5 | Movimento cartesiano PTP (SW 5.2 em diante)..... | 7-233 |
| 7.2 | Fresando em peças torneadas: TRANSMIT | 7-238 |
| 7.3 | Transformação de superfície cilíndrica: TRACYL | 7-241 |
| 7.4 | Eixo inclinado: TRAANG | 7-247 |
| 7.5 | Condições suplementares durante a seleção de uma transformação | 7-251 |
| 7.6 | Desativando transformações: TRAFOOF | 7-253 |
| 7.7 | Encadeamento de transformações | 7-254 |
| 7.8 | Eixos geométricos comutáveis, GEOAX | 7-257 |

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



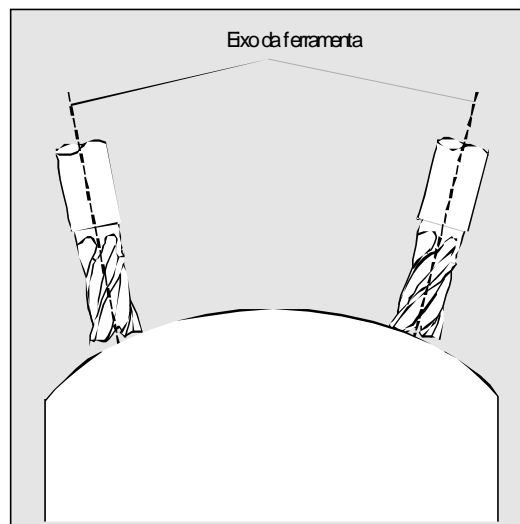
840Di

7.1 Transformações em três, quatro e cinco eixos: TRAORI



Para proporcionar condições otimizadas de usinagem em superfícies curvas, o ângulo de aproximação da ferramenta deve ser variável.

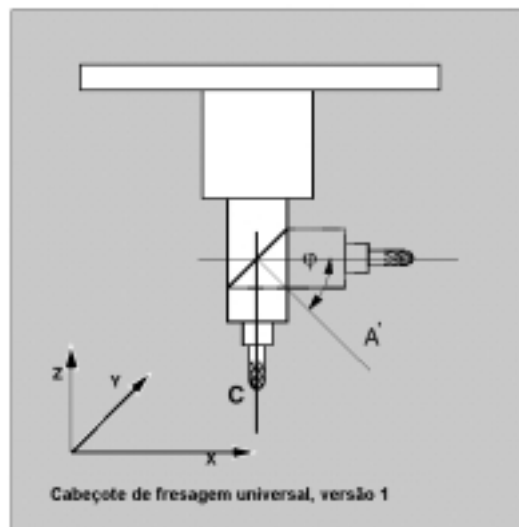
As características da máquinas utilizadas para a obtenção disto são armazenadas nos dados de eixos.



Fresa universal

Aqui os três eixos lineares (X, Y e Z) mais dois eixos de orientação definem o ângulo de ajuste e o ponto de usinagem da ferramenta. Um dos dois eixos de orientação é aplicado como sendo um eixo inclinado – em muitos casos, e no exemplo A – posicionada em um ângulo de 45°.

A seqüência dos eixos e o sentido de orientação da ferramenta são definidos através de dados de máquina em função da cinemática da máquina. No exemplo à direita, os arranjos são ilustrados através da cinemática CA.





840D
NCU 572
NCU 573



840Di

As seguintes relações de interpolação são possíveis:

A' no ângulo φ em relação ao eixo X

B' no ângulo φ em relação ao eixo Y

C' no ângulo φ em relação ao eixo Z

O ângulo φ pode ser configurado na faixa de 0° a $+89^\circ$ através de dados de máquina.

Dependendo da direção selecionada de orientação da ferramenta, o plano de usinagem ativo (G17, G18 e G19) devem ser definidos no programa de NC de forma que o comprimento de compensação da ferramenta atue na direção de orientação da ferramenta.

Transformação com rotação de eixos lineares

Este é um arranjo para situações onde a peça e a ferramenta se movimentam.

A cinemática compreende três eixos lineares (X, Y, Z) e dois rotativos com ângulos retos entre eles. O primeiro eixo rotativo pode ser movimentado, por exemplo, através de um carro movimentado por dois dos eixos lineares, e a ferramenta encontra-se posicionada paralelamente ao terceiro eixo linear.

O segundo eixo rotativo rotaciona a peça.

O terceiro eixo linear (eixo de giro) encontra-se no plano do carro.

A seqüência dos eixos rotativos e a direção de orientação da ferramenta são definidos através de dados de máquina, em função da cinemática da máquina.

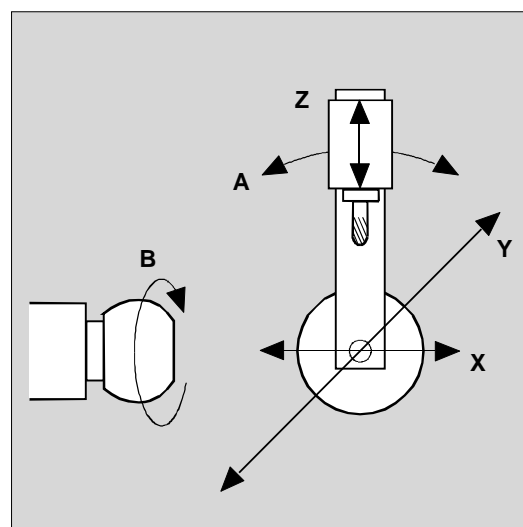
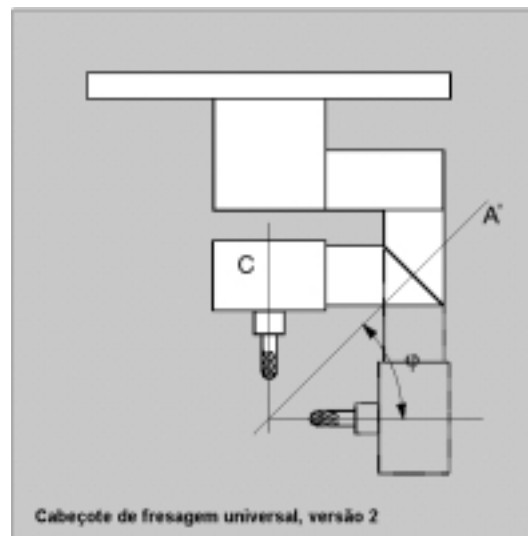
As seguintes relações são possíveis:

Eixos:

Primeiro eixo rotativo

Segundo eixo rotativo

Eixo linear de giro



Seqüência dos eixos

A A B B C C

B C A C A B

Z Y Z X Y X

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di

Transformações de três e quatro eixos



Programação

TRAORI (n)
TRAFOOF



Explicação dos comandos

| | |
|--------------|--|
| TRAORI | Ativa a primeira transformação de orientação selecionada |
| TRAORI (n) | Ativa a transformação de orientação n |
| n | Número da transformação (n=1 ou 2), TRAORI(1) corresponde a TRAORI |
| TRAFOOF | Desativa a transformação |



Informações adicionais

Quando ativada uma transformação, os parâmetros de posicionamento (X, Y, Z) sempre se referem à ponta da ferramenta.

Alterações nas posições dos eixos rotativos participantes da transformação resultam em movimentos de compensação dos outros eixos de máquina, de forma que a posição da ponta da ferramenta permaneça a mesma.



840D
NCU 572
NCU 573



840Di

7.1.1 Programando a orientação da ferramenta



Programas com cinco eixos são geralmente gerados através de sistemas CAD/CAM e não são digitados diretamente no controle. A seguinte descrição é direcionada principalmente para programadores de pós processadores.

Existem três formas para se programar a orientação de uma ferramenta:

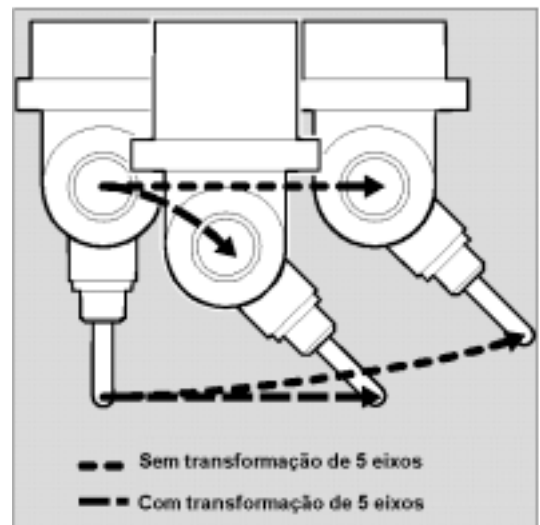
1. Programação do movimento do eixo rotativo. A alteração de orientação sempre é feita no sistema de coordenadas da máquina. Os eixos orientados sofrem movimentos como se fossem eixos síncronos.
2. Programação em ângulos Euler ou RPY com A2, B2, C2 ou programação em vetores de direção com A3, B3, C3. Os pontos dos vetores de direção a partir da ponta da ferramenta em direção ao porta ferramentas.

Programação através do ângulo principal LEAD e do ângulo lateral TILT (fresa de face).

Em todos os casos, a programação orientada é somente permitida caso uma transformação orientada esteja ativa.



Vantagem: Estes programas pode ser transferidos para qualquer tipo de máquina.



7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di



Programação

| | |
|----------------------|---|
| G1 X Y Z A B C | Programação de movimento em eixos rotativos |
| G1 X Y Z A2= B2= C2= | Programação em ângulos Euler |
| G1 X Y Z A3= B3= C3= | Programação em vetores de direção |
| G1 X Y Z A4= B4= C4= | Programação da superfície normal ao vetor no início do bloco |
| G1 X Y Z A5= B5= C5= | Programação da superfície normal ao vetor no fim do bloco |
| LEAD | Ângulo principal para programação da orientação da ferramenta |
| TILT | Ângulo lateral para programação da orientação da ferramenta |



Dados de máquina podem ser utilizados para comutação entre ângulos Euler e RPY.



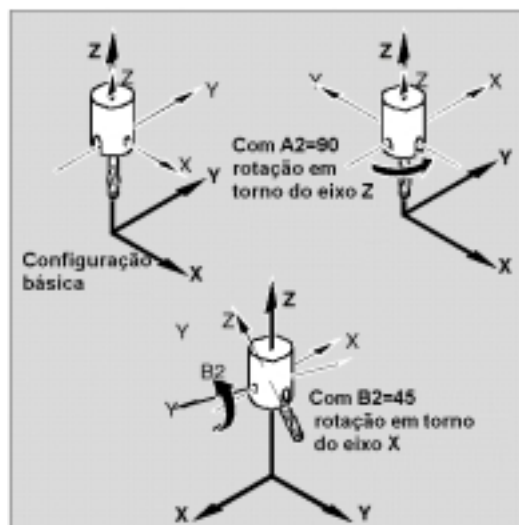
Programação em ângulos Euler

Os valores programados para a orientação com A2, B2, C2 são interpretadas como ângulos Euler (em graus).

O vetor de orientação é produzido pela rotação do na direção Z inicialmente com A2 em torno do eixo Z, e então com B2 em torno do novo eixo X, e finalmente, com C2 em torno do novo eixo Z



Neste caso, o valor de C2 (rotação em torno do novo Z) é irrelevante e não precisa ser programada.





840D
NCU 572
NCU 573



840Di



Programação em ângulos RPY

Os valores programados para orientação com A2, B2, C2 são interpretados como ângulos RPY (em graus).

O vetor de orientação é produzido pela rotação de um vetor na direção Z, inicialmente com C2 em torno de Z, e então com B2 em torno de Y, e finalmente com A2 em torno de X.



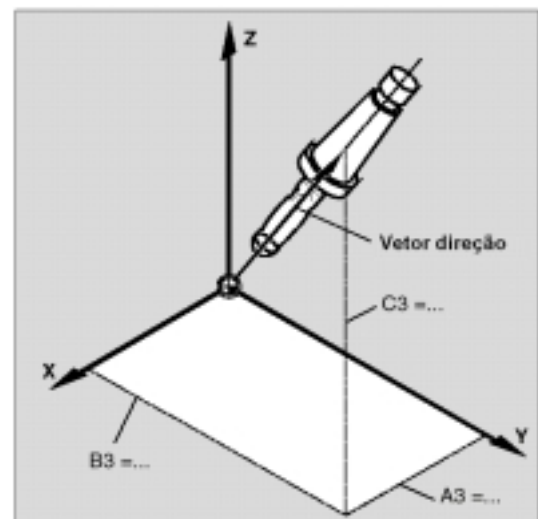
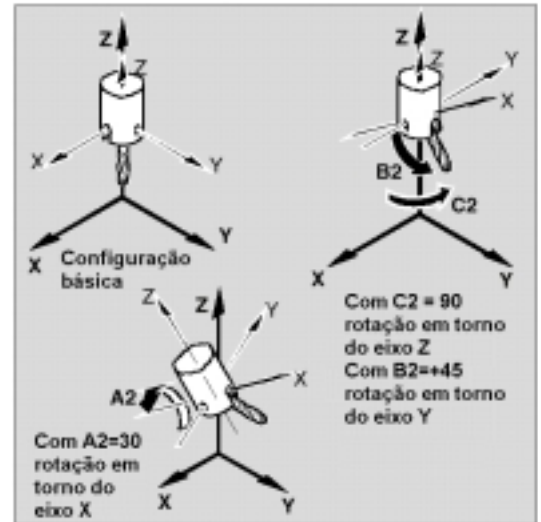
Em contraste com a programação com ângulos Euler, todos os três valores determinam o vetor de orientação.



Programação do vetor de direção

Os componentes do vetor de direção são programados com A3, B3 e C3. O vetor aponta para a direção da fixação da ferramenta e o comprimento do vetor é irrelevante.

Componentes não programadas são interpretadas como zero.





840D
NCU 572
NCU 573



840Di

Fresa de face

O modo fresa de face é utilizado para usinar superfícies com quaisquer graus de curvatura.

Para este tipo de fresagem 3D, você deve definir linha a linha as trajetórias 3D na superfície da peça. A forma e dimensões da ferramenta são levadas em consideração nos cálculos normalmente executados pelo CAM.

Após terem sido calculados, os blocos NC são carregados para o controle através de pós processadores.

Definição de superfícies

A trajetória da curvatura é definida através de superfícies normais aos vetores com as seguintes componentes:

A4, B4, C4 Vetor inicial no começo do bloco

A5, B5, C5 Vetor final no fim do bloco

Caso um bloco contenha somente o vetor inicial, a superfície normal ao vetor permanece constante durante todo o bloco.

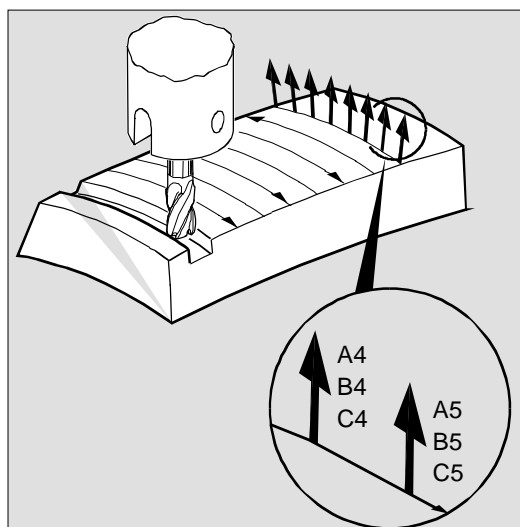
Caso um bloco contenha somente o vetor final, então uma grande interpolação circular será utilizada para fazer a interpolação a partir do valor final do bloco anterior até o ponto final programado.

Caso os vetores inicial e final sejam programados, então uma grande interpolação circular será utilizada para interpolar entre as duas direções, produzindo trajetórias suaves. Isto significa que é possível criar trajetórias contínuas e suaves.

Nas definições básicas, os vetores normais à superfície apontam na direção Z, sem levar em conta os planos ativos de G17 à G19.

O comprimento do vetor não possui significado.

As componentes de vetor não programadas são





840D
NCU 572
NCU 573



840Di

zeradas.

Quando ORIWCS está ativo (vide as próximas páginas), os vetores normais às superfícies se referem ao frame ativo e são rotacionados em conjunto com os frames.

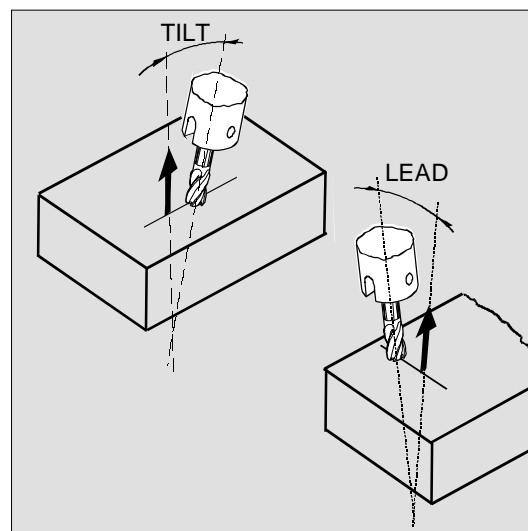
O vetor normal à superfície deve ser perpendicular à tangente da trajetória, e de valor menor a um valor limite carregado em dados de máquina, caso contrário será disparado um alarme.



Programando a orientação da ferramenta: com LEAD e TILT

O resultado da orientação da ferramenta é determinada a partir da:

- tangente da trajetória,
- vetor normal à superfície
- ângulo principal LEAD
- ângulo lateral TILT no fim do bloco



Explicação dos comandos

| | |
|------|--|
| LEAD | Ângulo relativo ao vetor normal à superfície no plano criado a partir da tangente da trajetória e do vetor normal à superfície |
| TILT | Ângulo no plano, perpendicular a tangente da trajetória relativa ao vetor normal à superfície |

Comportamento em cantos internos (com corretores de ferramenta 3D)

Caso o bloco seja reduzido em um canto interno, a orientação da ferramenta resultante é também alcançada no fim do bloco.

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di

7.1.2 Eixos de orientação, ORIWCS, ORIMCS



Programação

N.. ORIMCS

ou

N.. ORIWCS



Explicação dos comandos

| | |
|--------|--|
| ORIMCS | Rotação no sistema de coordenadas da máquina |
| ORIWCS | Rotação no sistema de coordenadas da peça |



Função

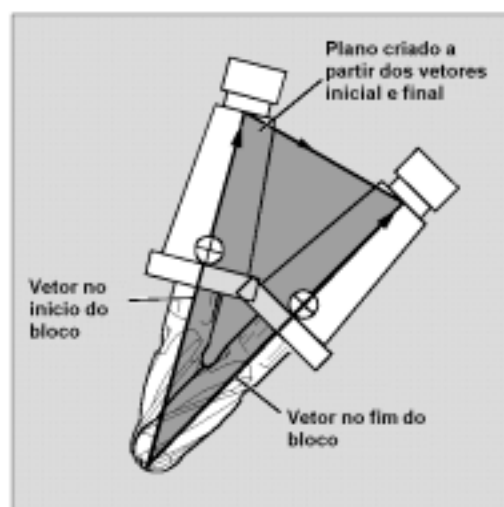
Quando programando a orientação no sistema de coordenadas da peça com ângulos Euler, RPY ou ainda com vetores orientados, o movimento de rotação pode ser definido através de ORIMCS/ORIWCS.



Seqüência

Com ORIMCS, o movimento da ferramenta depende da cinemática da máquina. Com uma alteração na orientação onde a ponta da ferramenta esteja fixada num ponto do espaço, uma interpolação linear é executada entre as posições dos eixos rotativos.

Com ORIWCS, o movimento é realizado independentemente da cinemática da máquina. Com uma alteração na orientação onde a ponta da ferramenta esteja fixada no espaço, a ferramenta se move no plano criado a partir dos vetores iniciais e finais.





840D
NCU 572
NCU 573



840Di



Informações adicionais

ORIWCs corresponde ao estado inicial. Caso isto não esteja claro inicialmente em qual máquina um programa com cinco eixos possa rodar, ORIWCs pode ser selecionado.

Os movimentos que a máquina realmente executa dependem da cinemática da máquina.

Com ORIMCS você pode programar os movimentos atuais da máquina, p.e., de forma a evitar colisões com fixações.

Os tipos de interpolações estão definidos nos dados de máquina \$MC_ORI_IPO_WITH_G_CODE:
ORIMCS/ORIWCs ou ORIMACHAX/ORIVIRTAX (vide seção 7.1.4).

7.1.3 Posições singulares e como elas são manuseadas



Informações adicionais sobre ORIWCs:

Movimentos orientados nos arredores de posições singulares em uma máquina com cinco eixos podem ativar grandes deslocamentos dos eixos da máquina. (por exemplo, para uma rotação onde C seja o eixo rotativo e A seja o eixo a ser rotacionado, todas as posições onde A=0 são singulares).

Para evitar um sobrecarga da máquina, o controle de velocidade reduz a velocidade de trajetória da ferramenta consideravelmente nas vizinhanças das posições singulares.

Com os dados de máquina

\$MC_TRAFO5_NON_POLE_LIMIT

\$MC_TRAFO5_POLE_LIMIT

a transformação pode ser parametrizada de forma que os movimentos orientados nas vizinhanças de um polo passem por este polo de forma a acelerar a usinagem.



Notas para a SW 5.2:

A partir da SW5.2, as posições singulares são tratadas somente em dados de máquina MD \$MC_TRAFO5_POLE_LIMIT
(vide Descrição de Funções, parte 3, seção 2.8.4).

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di

7.1.4 Eixos de orientação (SW 5.2 em diante)



Programação

N.. ORIEULER ou ORIRPY
ou
N.. ORIVIRT1 ou ORIVIRT2
N.. G1 X Y Z A2= B2= C2=



Explicação dos comandos

| | |
|----------------------|---|
| ORIEULER | Programação orientada com o uso de ângulos Euler |
| ORIRPY | Programação orientada com o uso de ângulos RPY |
| ORIVIRT1 | Programação orientada com o uso de ângulos virtuais (Definição 1), de acordo com MD \$MC_ORIAX_TURN_TAB_1 |
| ORIVIRT2 | Programação orientada com o uso de ângulos virtuais (Definição 2), de acordo com MD \$MC_ORIAX_TURN_TAB_2 |
| G1 X Y Z A2= B2= C2= | Programação de ângulo dos eixos virtuais |



Programação

N.. ORIAXES ou ORIVECT
N.. G1 X Y Z A B C



Explicação dos comandos

| | |
|----------------|---|
| ORIAXES | Interpolação linear de eixos orientados |
| ORIVECT | Interpolação com círculo grande |
| ORIMCS | Rotação no sistema de coordenadas da máquina Vide descrição na seção 7.1.2 |
| ORIWCS | Rotação no sistema de coordenadas da peça Vide descrição na seção 7.1.2 |
| G1 X Y Z A B C | Programação da posição da máquina |



Função

A função de eixos orientados descrevem a orientação da ferramenta na área. Isto proporciona um terceiro grau de liberdade que descreve a rotação em torno de si mesma, necessária para transformações em eixos.



840D
NCU 572
NCU 573



840Di



O MD \$MC_ORI_DEF_WITH_G_CODE especifica como os ângulos A2, B2 e C2 devem ser programados:

Definição de acordo com MD \$MC_ORIENTATION_IS_EULER (standard) ou

definição de acordo com G_group 50 (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

MD \$MC_ORI_IPO_WITH_G_CODE define qual o tipo de interpolação encontra-se ativa:
ORIWCS/ORIMCS ou ORIAXES/ORIVECT.

Modo JOG

Neste modo, a interpolação dos ângulos orientados é sempre linear. Durante movimentos contínuos e incrementais através das teclas de direção, somente um eixo pode ser movimentado. Com o uso das manivelas eletrônicas ambos eixos de orientação podem ser movimentados ao mesmo tempo.



Para movimentos com os eixos de orientação, a chave de avanço ou a chave de correção de avanço rápido do canal são ativadas com a correção de avanço rápido.

Uma velocidade separada pode ser especificada através dos seguintes dados de máquina:

```
$MC_JOG_VELO_RAPID_GEO
$MC_JOG_VELO_GEO
$MC_JOG_VELO_RAPID_ORI
$MC_JOG_VELO_ORI
```

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di



Programação do avanço

| | |
|-------|--|
| FORI1 | Avanço para giro do vetor de orientação em um círculo grande |
| FORI2 | Avanço para rotação em torno do giro do vetor de orientação |



Com movimentos orientados o avanço programado corresponde à uma velocidade angular [graus/min].

Efetividade dos avanços nos códigos G:

Ao programar ORIAXES o avanço para um eixo de orientação pode ser limitado através da instrução FL[] (limitação de avanço).

Ao programar ORIVECT o avanço deve ser programado com FORI1 ou FORI2. FORI1 e FORI2 podem ser programados somente uma vez em cada bloco NC. Quando programados desta forma, a trajetória será sempre percorrida da forma mais curta o possível. Com movimentos de rotação e giro em torno de eixos, o menor avanço é sempre o utilizado. Com movimentos orientados o avanço programado corresponde à velocidade angular [graus/min].

Caso eixos geométricos e de orientação estejam se movimentando ao longo de uma mesma trajetória, o movimento será realizado com o menor avanço.



840D
NCU 572
NCU 573



840Di

7.1.5 Movimento cartesiano PTP (SW 5.2 em diante)



Programação

```
N.. TRAORI
N.. STAT=`B10` TU=`B100` PTP
N.. CP
```



Explicação dos comandos

| | |
|-------|---|
| PTP | Ponto a Ponto O movimento é executado como movimento síncrono; a menor velocidade dos eixos que participam do movimento será a velocidade dominante. |
| CP | Trajetoória contínua (movimento de trajetória) O movimento é executado como movimento de trajetória cartesiana. |
| STAT= | Posição de articulações, valor depende da transformação. |
| TU= | Informações TURN Isto habilita a aproximação de ângulos entre -360 e +360 graus. |



Função

Esta função permite que uma posição seja programada num sistema de coordenadas cartesiano, mas entretanto, o movimento da máquina seja realizado nas coordenadas da máquina.

A função pode ser utilizada, por exemplo, durante a alteração da posição de articulação, caso o movimento passe por uma singularidade.



Nota:

A função somente tem significado em conjunto com a transformação ativa. Adicionalmente, o "movimento PTP" é possível somente em conjunto com G0 e G1.



Seqüência

A comutação entre o movimento cartesiano para os eixos de máquina é feita através dos comandos modais PTP e CP. Os comandos são modais. CP corresponde ao estado inicial de fornecimento.

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di

Programação da posição (STAT=)

A posição da máquina não pode ser definida somente através da especificação de coordenadas cartesianas e da orientação da ferramenta. De acordo com a cinemática utilizada, existem até 8 tipos de diferentes articulações. Elas são então transformações. Para que sejam feitas as conversões do sistema cartesiano para a posição em ângulo de eixo, você deve especificar a posição das articulações através do comando STAT. O comando STAT contém um bit para cada posição possível.



Referências:

A seguinte documentação fornece uma descrição detalhada das diferentes formas de transformações: SINUMERIK 840D/FM-NC Descrição de Funções (Parte 3), "Pacote Manuseio de Transformações".

Para uma descrição dos bits de posição necessários para a programação de "STAT", favor consultar: SINUMERIK 840D/FM-NC Descrição de funções (Parte 3), "Transformação de três para cinco eixos".

Programação dos ângulos dos eixos (TU=)

Para um determinado ângulo de aproximação dos eixos $\leq \pm 360$ graus, esta informação deve ser programada com o comando "TU=". Este comando não é modal.

Os eixos se movimentam através do caminho mais curto:

- Caso TU não seja programado com uma posição
- Com eixos que possuem uma faixa de movimento $> \pm 360$ graus



840D
NCU 572
NCU 573



840Di

Exemplo:

A posição de partida exibida no diagrama pode ser atingida tanto na direção positiva quanto negativa. A direção é programada no endereço A1.

A1=225°, TU=bit 0, → direção positiva

A1=-135°, TU=bit 1, → direção negativa



Arredondamento de cantos entre movimentos CO e PTP

Um arredondamento de canto programável entre os blocos é possível com G641.

O tamanho da área de arredondamento é a dimensão da trajetória em mm ou polegadas a partir ou para o qual ocorre a transição é arredondada:

- Com ADISPOS para blocos G0
- Com ADIS para todos os outros comandos de movimento

O cálculo da distância da trajetória corresponde a considerar o endereço F em blocos não G0. O avanço é mantido para os eixos especificados no FGROUP(..).

Cálculo do avanço:

Para blocos CO, os eixos cartesianos do sistema de coordenadas básico são utilizados para o cálculo.

Para blocos PTP, os eixos correspondentes ao sistema de coordenadas da máquina são utilizados para o cálculo.

7.1 Transformações em três, quatro e cinco eixos: TRAORI



840D
NCU 572
NCU 573



840Di



Informações adicionais

Alteração do modo

A função “Movimento cartesiano PTO” tem significado somente para os modos MDA e AUTO. A definição é mantida quando o modo é alterado para JOG.

Os eixos são movimentados no MCS caso o código G PTP esteja programado. Caso o código G programado seja o CP, os eixos movimentam-se no WCS.

Ligação/reset

Após a energização do sistema ou de um reset, serão ativadas as funções de acordo com o dado de máquina \$MC_GCODE_RESET_VALUES[48]. O movimento no modo "CP" é atribuído como default.

Repos

Caso a função “Movimento cartesiano PTP” tenha sido programado durante a interrupção do bloco, o reposicionamento será realizado com PTP.

Movimento sobreposto

Deslocamentos DRF ou deslocamentos de origem externos podem ser utilizados com apenas algumas restrições em conjunto com a movimentação PTP cartesiana. Correções de velocidade não devem existir no BCS durante a comutação de movimentos PTP para CP.



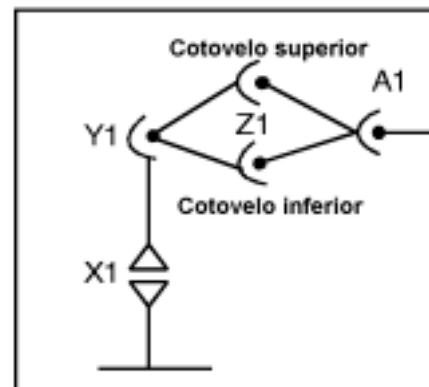
840D
NCU 572
NCU 573



840Di



Exemplo de programação



| | | |
|-----|--|---|
| N10 | G0 X0 Y-30 Z60 A-30 F10000 | Posição inicial → cotovelo para cima |
| N20 | TRAORI(1) | Liga transformação |
| N30 | X1000 Y0 Z400 A0 | |
| N40 | X1000 Z500 A0 STAT='B10' TU='B100' PTP | Reordena sem transformação → cotovelo para baixo |
| N50 | X1200 Z400 CP | Ativa novamente a transformação |
| N60 | X1000 Z500 A20 | |
| N70 | M30 | |

7.3 Transformação de superfície cilíndrica: TRACYL



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

7.2 Fresando em peças torneadas: TRANSMIT



Programação

TRANSMIT OU TRANSMIT (n)

TRAFOOF



Explicação dos comandos

| | |
|--------------|---|
| TRANSMIT | Ativa a primeira função TRANSMIT declarada |
| TRANSMIT (n) | Ativa a n-ésima função transmit declarada, n não deve ser maior que 2 (TRANSMIT (1) corresponde à TRANSMIT) |
| TRAFOOF | Desativa uma transformação ativa |

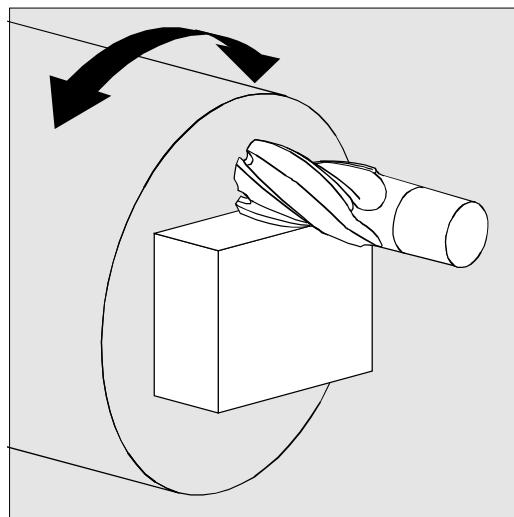


Uma transformação TRANSMIT é desativada também caso uma das outras transformações seja ativada no respectivo canal (p.e. TRACYL, TRAANG, TRAORI).



A função TRANSMIT fornece os seguintes recursos:

- Usinagem da face de peças rotativas na fixação rotativa (furação, contornos).
Um sistema de coordenadas cartesianas pode ser utilizado para a programação desta operação de usinagem.
- O controle transforma os movimentos programados em coordenadas cartesianas para o sistema real de coordenadas da máquina (fornecimento padrão):
 - Eixo rotativo
 - Eixo de avanço perpendicular ao eixo – rotativo .
 - Eixo longitudinal paralelo ao eixo rotativo.
 Os eixos lineares posicionados perpendicularmente um ao outro.
- O deslocamento do centro da ferramenta relativo ao centro do torno é permitido.
- O controle de velocidade fornece limites definidos para os movimentos rotativos.



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Eixo rotativo

O eixo rotativo não pode ser programado pois é relacionado à um eixo geométrico, não podendo portanto ser programado diretamente como eixo de canal.

Polo

Até a SW 3.x

Movimentos através do polo (origem do sistema de coordenadas cartesiano) são inibidos. Caso programado movimentos através dos pólos será emitido um alarme. Com um deslocamento de centro da ferramenta, o movimento para no limite da área que não pode ser atravessada.

SW 4 em diante

Existem duas formas de se atravessar um polo:

1. Atravessar somente com eixos lineares
2. Atravessar o polo com a rotação de um eixo rotativo

As definições são feitas através de MD 24911 e 24951.



Referências

/FB/ M1 Transformações cinemáticas

7.3 Transformação de superfície cilíndrica: TRACYL



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



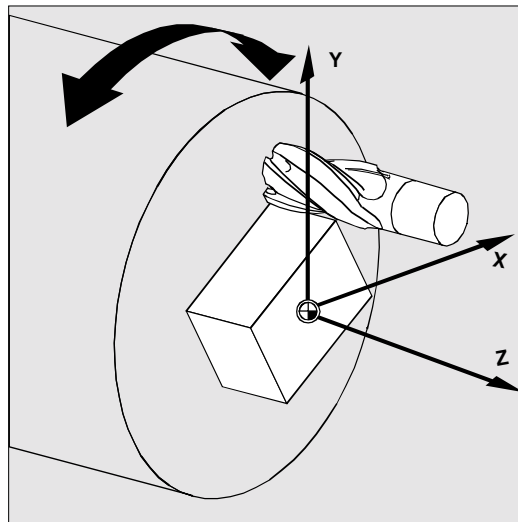
810D



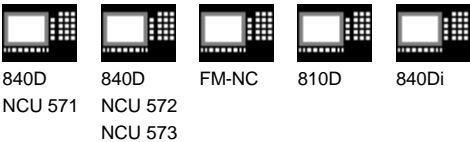
840Di



Exemplo de programação



| | |
|---------------------------------|--|
| N10 T1 D1 G54 G17 G90 F5000 G94 | Seleção da ferramenta |
| N20 G0 X20 Z10 SPOS=45 | Referenciamento |
| N30 TRANSMIT | Ativação da função TRANSMIT |
| N40 ROT RPL=-45 | Ativar frame |
| N50 ATRANS X-2 Y10 | |
| N60 G1 X10 Y-10 G41 OFFN=1 | Desbastar quadrado, tolerância de 1 mm |
| N70 X-10 | |
| N80 Y10 | |
| N90 X10 | |
| N100 Y-10 | |
| N110 G0 Z20 G40 OFFN=0 | Troca de ferramenta |
| N120 T2 D1 X15 Y-15 | |
| N130 Z10 G41 | |
| N140 G1 X10 Y-10 | Acabamento do quadrado |
| N150 X-10 | |
| N160 Y10 | |
| N170 X10 | |
| N180 Y-10 | |
| N190 Z20 G40 | Desligar frame |
| N200 TRANS | |
| N210 TRAFOOF | |
| N220 G0 X20 Z10 SPOS=45 | Referenciar |
| N230 M30 | |



7.3 Transformação de superfície cilíndrica: TRACYL



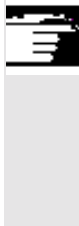
Programação

TRACYL(*d*) ou TRACYL(*d*,*t*)
 TRAFOOF



Explicação dos comandos

| | |
|-------------------------------|---|
| TRACYL(<i>d</i>) | Ativa a primeira função TRACYL |
| TRACYL(<i>d</i> , <i>n</i>) | Ativa a <i>n</i> -ésima função TRACYL declarada, <i>n</i> pode ir até 2. TRACYL(<i>d</i> ,1) é o mesmo que TRACYL(<i>d</i>). |
| <i>d</i> | Valor atual do diâmetro do cilindro a ser usinado |
| TRAFOOF | Desliga a transformação |
| OFFN | Deslocamento de contorno –normal: Distância do lado do canal do contorno de referência programado. |



Uma transformação TRACYL ativa é da mesma forma desativada caso uma outra transformações seja ativada no canal em questão (p.e. TRANSMIT,, TRAANG, TRAORI)

An



Função

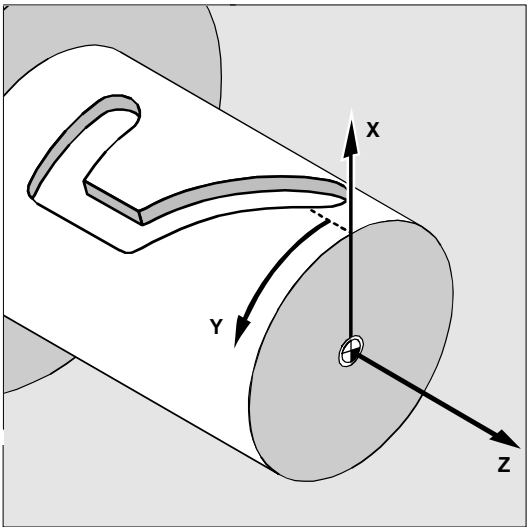
Transformação de superfície cilíndrica TRACYL

A função de transformação de superfície cilíndrica TRACYL pode ser utilizada para:

Usinagem de

- Canais longitudinais em corpos cilíndricos,
- Canais transversais em corpos cilíndrico
- Canais com quaisquer trajetórias em corpos cilíndricos.

A trajetória dos canais é programada com referência à superfície aberta do cilindro.



Sistema de coordenadas da peça

7.3 Transformação de superfície cilíndrica: TRACYL



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Existem dois tipos de transformações de coordenadas de superfícies cilíndrica:

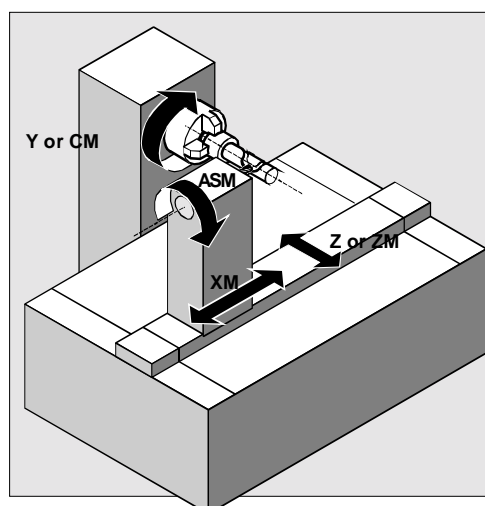
- Sem compensação de lado do canal
- Com compensação de lado do canal

Sem compensação de lado do canal:

O controle transforma os movimentos programados em um sistema de coordenadas cilíndricas em movimentos nos eixos reais da máquina:

- Eixos rotativos
- Eixo de avanço perpendicular ao rotativo
- Eixo longitudinal paralelo ao eixo rotativo

Os eixos lineares são posicionados perpendicularmente um ao outro. O eixo de avanço faz interseção com o eixo rotativo.



Sistema de coordenadas da máquina

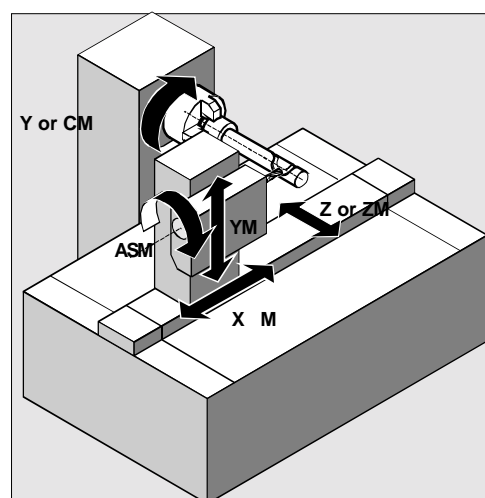
Com compensação de lado do canal:

Mesma cinemática como acima, mais:

- Eixo longitudinal paralelo à direção da circunferência

Os eixos lineares são posicionados perpendicularmente um ao outro.

O controle de velocidade limita os movimentos rotativos.



Sistema de coordenadas da máquina

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D

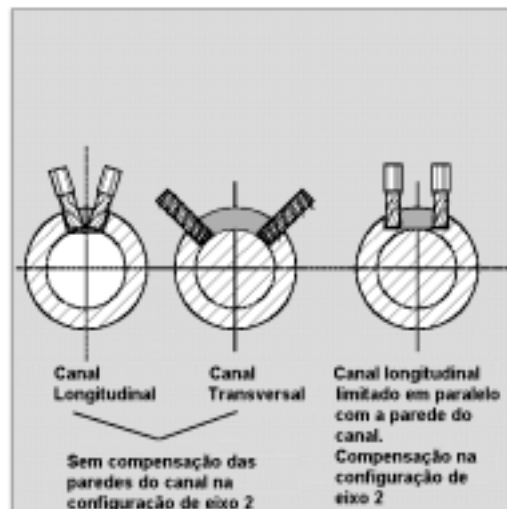


840Di

Vista em corte do canal

Com a configuração de eixos 1, os canais ao longo do eixo rotativo são limitados em paralelo caso a largura do canal não esteja de acordo com o raio da fresa.

Canais paralelos à circunferência (movimento do canal) não são paralelos no início e fim.



Deslocamento de contorno normal OFFN

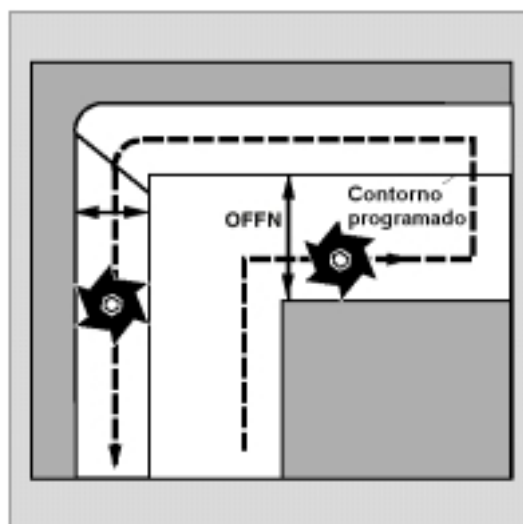
Para a usinagem de canais com TRACYL, o canal:

- A linha de centro é programada na peça
- O programa, a largura do canal através de OFFN.

OFFN somente será ativado quando a compensação de raio da ferramenta for ativada, para proteger o lado canal em ser danificado. Adicionalmente, $OFFN \geq \text{raio da ferramenta}$ será aconselhável para excluir possibilidades de danos ao lado contrário do canal.

Um programa para fresagem de um canal normalmente é composto dos seguintes passos:

1. Seleção da ferramenta
2. Seleção de TRACYL
3. Seleção de um sistema de coordenadas adequado (FRAME)
4. Posicionamento
5. Programação OFFN
6. Seleção TRC
7. Bloco de aproximação (digite TRC e o lado de aproximação do canal)
8. Linha de contorno do centro do canal
9. Desligar TRC
10. Recuo (abandona TRC e recuo da borda do canal)
11. Posiciona
12. TRAFOOF



7.3 Transformação de superfície cilíndrica: TRACYL



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

13. Seleciona o deslocamento de origem original novamente (FRAME)

Casos especiais:

- Seleção TRC:
TRC não é programado de acordo com o TRC, mas sim relativo a linha de centro do canal. G42 é programado de forma que a ferramenta percorra a lateral esquerda do canal (ao invés de G41). Você não precisa proceder desta forma caso a largura do canal esteja especificada com sinal negativo em OFFN.
- OFFN com TRACYL possui um efeito diferente da programação sem TRACYL. Como OFFN é também incluído no cálculo sem TRACYL quando ativo TRC, OFFN pode ser novamente zerado após TRAFOOF.
- É possível alterar OFFN no programa. Isto significa que a linha de centro do canal pode ser movimentada a partir do centro (vide figura).
- Canais de controle:
Com TRACYL o mesmo canal não é gerado para canais de controle (como se produzidos com uma ferramenta de mesmo diâmetro do canal)
Em princípio, não é possível gerar a mesma geometria do outro lado do canal com uma ferramenta cilíndrica menor que outra.
TRACYL minimiza os erros. Para prevenir a ocorrência de erros, o raio da ferramenta deve ser um pouco menor que a metade da largura do canal.

7.3 Transformação de superfície cilíndrica: TRACYL



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Com a curva de transformação de superfície cilíndrica com compensação de lado do canal, o eixo utilizado para compensação deve ser levado à posição 0 ($y=0$), de forma que a usinagem do canal esteja alinhada com a linha de centro do canal programado.

Eixo rotativo

O eixo rotativo não pode ser programado pois encontra-se relacionado a um eixo geométrico, não podendo portanto ser diretamente programado como um eixo de canal.

Eixos disponíveis

Os seguintes eixos não podem ser utilizados como eixos de posicionamento ou correspondentes:

- O eixo geométrico na direção da circunferência da superfície curva do cilindro (eixo Y)
- O eixo linear adicional utilizado na compensação do lado do canal (eixo Z)

7.3 Transformação de superfície cilíndrica: TRACYL



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



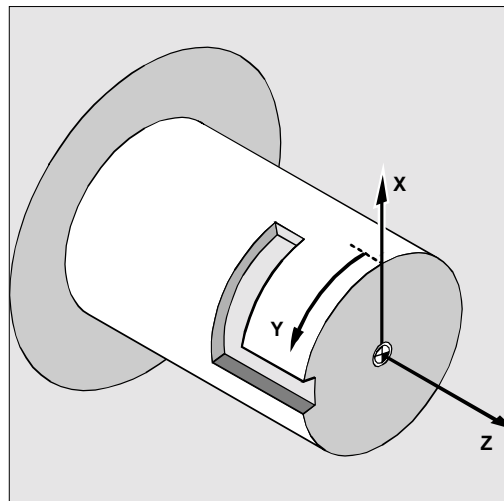
810D



840Di



Exemplo de programação



| | |
|-----------------------------|---|
| N10 T1 D1 G54 G90 F5000 G94 | Seleção da ferramenta, compensação da fixação |
| N20 SPOS=0 | Referenciar |
| N30 G0 X25 Y0 Z105 CC=200 | |
| N40 TRACYL (40) | Comuta para a curva de transformação de superfície cilíndrica |
| N50 G19 | Seleção do plano |

Produzindo um canal em forma de gancho

| | |
|----------------------------|---|
| N60 G1 X20 | Aproxima para a base do canal |
| N70 OFFN=12 | Define a distância de 12 mm relativa à linha de centro do canal |
| N80 G1 Z100 G42 | Aproxima no lado esquerdo do canal |
| N90 G1 Z50 | Seleciona o canal paralelo ao eixo cilíndrico |
| N100 G1 Y10 | Seleção do canal paralelo à circunferência |
| N110 OFFN=4 G42 | Aproxima o lado esquerdo do canal, define distância de 4mm a partir da linha de centro do canal |
| N120 G1 Y70 | Seleção de canal paralelo à circunferência |
| N130 G1 Z100 | Seleção do canal paralelo ao eixo cilíndrico |
| N140 G1 Z105 G40 | Recuo da lateral do canal |
| N150 G1 X25 | Recuo |
| N160 TRAFOOF | |
| N170 G0 X25 Y0 Z105 CC=200 | Referenciar |
| N180 M30 | |



840D
NCU 572
NCU 573



810D



840Di

7.4 Eixo inclinado: TRAANG



Programação

TRAANG (α) OU TRAANG (α , n)
TRAFOOF



Explicação dos comandos

| | |
|-------------------------|---|
| TRAANG (α) | Ativa o primeiro eixo de transformação inclinado declarado |
| TRAANG (α , n) | Ativa o e-nésimo eixo de transformação inclinado declarado. (n) não deve ser superior à 2. TRAANG(α ,1) corresponde à TRAANG(α). |
| α | Ângulo do eixo inclinado |
| TRAFOOF | Desliga a transformação |



Caso α (ângulo) seja omitido ou um valor zero carregado, a transformação é ativada com os parâmetros definidos na seleção anterior. Na primeira seleção, os dados de máquina default são utilizados.

Uma transformação TRAANG ativa será também desativada caso uma das outras transformações (p.e. TRACYL, TRANSMIT, TRAORI) estejam ativadas no mesmo canal.

7.4 Eixo inclinado: TRAANG



840 D
NCU 572
NCU 573



810D



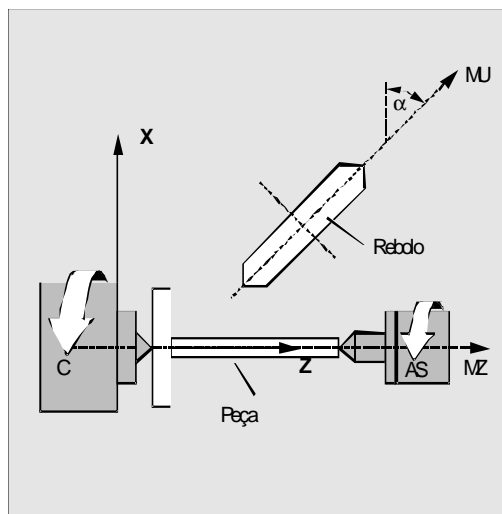
840Di



Função

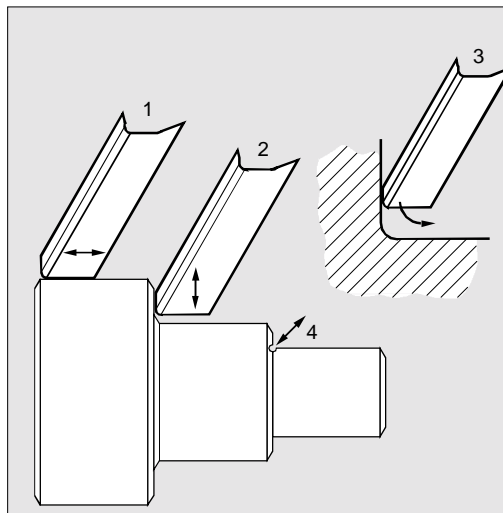
A função eixo inclinado foi definida para uso em retíficas e possui a capacidade de:

- Usinagem com avanço em eixo inclinado
- Um sistema de coordenadas cartesiano pode ser utilizado para programação.
- O sistema transforma os movimentos programados nas coordenadas cartesianas em movimentos nos eixos reais da máquina: avanço no eixo inclinado.



As seguintes operações de usinagem são possíveis:

1. Retífica longitudinal
2. Retífica transversal
3. Retífica de contorno específico
4. Saída inclinada do rebolo





840D
NCU 572
NCU 573



810D



840Di

As seguintes definições devem ser feitas em dados de máquina:

- O ângulo entre o eixo de usinagem e o eixo inclinado.
- A posição do zero ferramenta com referência à origem do sistema de coordenadas declarado na função “Eixo inclinado”.
- Velocidade reserva disponível no eixo paralelo para compensação de movimento.
- Aceleração reserva disponível no eixo paralelo para compensação do movimento.

Configuração de eixo

Para tornar possível a programação em um sistema de coordenadas cartesianas, a relação entre este sistema de coordenadas e os eixos de máquina reais (UM, MC) devem ser declarados no controle:

- Nomes dos eixos geométricos
- Relacionar os eixos geométricos à eixos de canal
 - Caso geral (eixo inclinado não ativo)
 - Eixo inclinado ativo
- Relacionar os eixos de canal à números de eixos de máquina
- Identificação dos fusos
- Definir os nomes dos eixos de máquina

O procedimento é o mesmo utilizado para os eixos normais, com exceção de “eixo inclinado ativo”.

7.4 Eixo inclinado: TRAANG



840 D
NCU 572
NCU 573



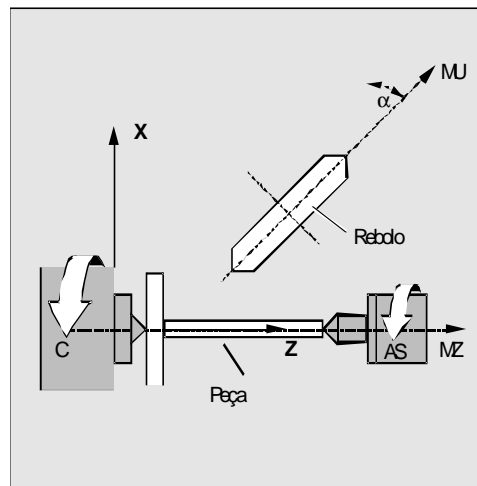
810D



840Di



Exemplo de programação



| | |
|---|--|
| N10 G0 G90 Z0 MU=10 G54 F5000 -> -> G18 G64 T1 D1 | Selecionar a ferramenta, compensar a fixação, selecionar plano |
| N20 TRAANG(45) | Ativar a transformação de eixo inclinado |
| N30 G0 Z10 X5 | Referenciar |
| N40 WAITP(Z) | Habilitar eixos para oscilação |
| N50 OSP[Z]=10 OSP2[Z]=5 OST1[Z]=-2 -> -> OST2[Z]=-2 FA[Z]=5000 | Oscilar de acordo com a dimensão (Vide capítulo 9 para oscilação) |
| N60 OS[Z]=1 | |
| N70 POS[X]=4.5 FA[X]=50 | |
| N80 OS[Z]=0 | |
| N90 WAITP(Z) | Habilita eixos de oscilação como eixos de posicionamento |
| N100 TRAFOOF | Desligar transformação |
| N110 G0 Z10 MU=10 | Recuo |
| N120 M30 | |

-> programado em um bloco

7.5 Condições suplementares na seleção de uma transformação



840D
NCU 572
NCU 573



810D



840Di

7.5 Condições suplementares durante a seleção de uma transformação



A seleção de transformações pode ser realizada através de um programa de usinagem ou em MDA. Favor observar:

- Não serão inseridos blocos intermediários (chanfros/raios).
- Uma seqüência de blocos spline deve Ter sido completada, caso contrário será emitida uma mensagem.
- A correção fina de ferramentas deve ser desativada (FTOCOF), caso contrário será emitida uma mensagem.
- A compensação de raio da ferramenta (G40) deve estar desativada, caso contrário será emitida uma mensagem.
- O controle irá incluir a compensação do comprimento da ferramenta ativada na transformação.
- O frame atual ativo antes da transformação será desativado pelo controle.
- A limitação de área de trabalho ativa será desligada para os eixos envolvidos na transformação (corresponde à WALIMOF).
- A monitoração das zonas de proteção será desativada.
- O modo de passagens contínuas a o posicionamento aproximado são interrompidos.
- O deslocamento DRF dos eixos envolvidos na transformação não deve ser alterado durante a execução do pré-processamento e das subrotinas principais(SW 3 e anteriores).
- Todos os eixos especificados em dados de máquina devem ser do tipo bloco-síncronos.
- Eixos comutados devem ser comutados para seus estados originais, caso contrário será emitida uma mensagem.
- As mensagens são exibidas para os eixos específicos.

Troca de ferramenta

Uma troca de ferramentas pode ser realizada somente caso a compensação de raio da ferramenta esteja desselecionada.

Uma alteração no valor de compensação do comprimento da ferramenta e a seleção/desseleção da correção de raio da ferramenta não devem ser programados no mesmo bloco.

7.5 Condições suplementares na seleção de uma transformação



840 D
NCU 572
NCU 573



810D



840Di

Alteração no frame

Quaisquer instruções que se refiram somente ao sistema de coordenadas básico são permitidas (FRAME, correção do raio da ferramenta). Uma alteração no frame com G91 (dimensão incremental) não será tratada separadamente como quando a transformação estava inativa. O incremento a ser percorrido será calculado no sistema de coordenadas da peça do novo frame – não importando qual p frame ativo no bloco anterior.

Exceções

Eixos envolvidos em transformações não podem ser utilizados

- Como um eixo de preset(alarme)
- Para aproximação em ponto fixo (alarme)
- Para referenciamento (alarme)

7.6 Desativando transformações: TRAFOOF transformação



840D
NCU 572
NCU 573



810D



840Di

7.6 Desativando transformações: TRAFOOF



Programação

TRAFOOF



Explicação dos comandos

TRAFOOF

Desativa todas as transformações/frames ativos



Função

Quando executado o comando TRAFOOF, todas as transformações e frames são desativados.



Frames necessários após esta instrução devem ser reprogramados para que se tornem ativos.

Favor notar:

As mesmas restrições aplicadas à desseleção de uma transformação como para sua ativação (vide a seção “Condições suplementares para desselecionar uma transformação”)

7.7 Encadeamento de transformações



840 D
NCU 572
NCU 573



810D



840Di

7.7 Encadeamento de transformações



SW 5 em diante suporta **duas** transformações, uma após outra, de forma que os elementos de movimentação para os eixos da primeira transformação sejam os dados de entrada da Segunda. Os elementos de movimento da segunda transformação são os executados nos eixos da máquina.

- A partir de SW 5, duas transformações podem ser encadeadas
- A **segunda** transformação deve ser “**eixo inclinado**” (TRAANG).
- A primeira transformação pode ser uma das seguintes:
 - Transformações de orientação (TRAORI), incluindo fresagem universal
 - TRANSMIT
 - TRACYL
 - TRAANG

Aplicações

- Retificação de contornos programados como linha lateral em operação cilíndrica (TRACYL) utilizando um rebolo inclinado.
- Acabamento de contorno não circular que tenha sido gerado através de TRANSMIT com o uso de rebolo inclinado.



Pré requisito para uso do comando de ativação para uma transformação encadeada é que as transformações individuais possam ser encadeadas e estas tenham sido definidas através dos dados de máquina.

As condições suplementares e casos especiais especificados nas descrições de cada transformação serão também aplicadas no encadeamento.



840D
NCU 572
NCU 573



810D



840Di



Informações adicionais

Informações sobre como configurar os dados de máquina para as transformações poderão ser encontrados no manual Descrição de Funções: M1 e F2.



Fabricante da máquina (MH7.1)

Favor ler as especificações do fabricante da máquina relativas a quaisquer transformações pré definidas em dados de máquina.



Transformações e transformações encadeadas são opções. O catálogo fornece as informações sobre disponibilidade das transformações específicas para encadeamento nos diversos controles.

Os seguintes comandos encontram-se disponíveis para as transformações encadeadas:

TRACON para ativar e

TRAFOOF para desativar.

Ativação



Programação

TRACON(trf, par)

Ativa uma transformação encadeada.



Explicação dos parâmetros

trf

Quantidade de transformações encadeadas: 0 ou 1 para a primeira/única transformação encadeada.

Caso nada seja programado neste campo, o efeito será o mesmo que a programação de 0 ou 1, ou seja, somente a primeira/única transformação ativada.

2 para a segunda transformação encadeada. (com valores fora da faixa de 0 a 2, um alarme é emitido).

7.7 Encadeamento de transformações



840 D
NCU 572
NCU 573



810D



840Di

par

Um ou vários parâmetros separados por vírgulas para as transformações na cadeia, os parâmetros não necessários para a carga de, por exemplo, do ângulo do eixo inclinado. Caso os parâmetros não sejam definidos, as definições default ou os valores utilizados da última vez em que a transformação foi utilizada serão carregados. Vírgulas devem ser utilizadas para garantir que os parâmetros programados sejam são utilizados em seqüência correta, caso para os parâmetros anteriores devam ser utilizados valores default ou anteriormente programados. Particularmente, quando especificado ao menos um parâmetro, este deve ser precedido de uma vírgula, mesmo que não necessária para especificar trfg (p.e. TRACON(, 3.7)).



Função

A transformação encadeada encontra-se ativa. Qualquer outra ativação anterior será desativada com TRACON().



Uma ferramenta é sempre definida para uma transformação em uma cadeia. A transformação seguinte irá se comportar como se o comprimento da ferramenta ativa fosse zero. Somente os comprimentos base de uma ferramenta (_BASE_TOOL_) especificados em dados de máquina serão válidos para a **primeira** transformação da cadeia.

Desativação



Programação

TRAFOOF



Função

O comando desativa a transformação (encadeada) anteriormente ativa.



840D
NCU 572
NCU 573



810D



840Di

7.8 Eixos geométricos comutáveis, GEOAX



Programação

GEOAX(n,eixo do canal axis,n,eixo do canal,...)

GEOAX()



Explicação dos parâmetros

| | |
|---|---|
| GEOAX(n,eixo do canal axis,n,eixo do canal,...) | Comuta os eixos geométricos. |
| GEOAX() | Chama a configuração geométrica básica dos eixos |
| n | Quantidade de eixos geométricos (n=1, 2 ou 3) a atribuídos a outro eixo do canal. n=0: Remove o eixo específico do grupo de eixos geométricos sem reposição. |
| Eixo do canal | Nome do eixo geométrico a ser incluído no agrupamento de eixos escravos. |



Função

Com a função “Comutação de eixos geométricos” o grupo de eixos geométricos configurado em dados de máquina pode ser alterado através de instruções no programa de usinagem. Um eixo de canal definido como eixo auxiliar síncrono pode ser comutado com qualquer eixo geométrico.

Exemplo:

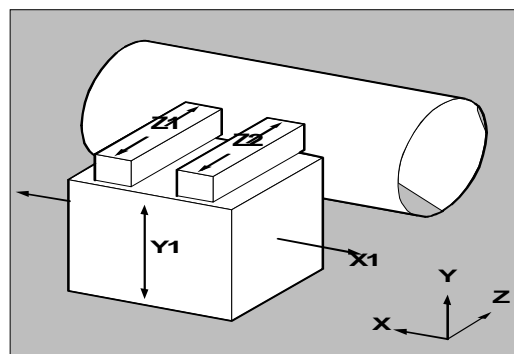
Um carro de ferramenta pode ser movimentado através dos eixos de canal X1, Y1, Z1, Z2. Os eixos Z1 e Z2 devem ser utilizados alternadamente como eixos geométricos no programa. A função GEOAX faz esta comutação no programa.

Após a ativação, a conexão X1, Y1, Z1 encontra-se ativa (definida através de dados de máquina).

N100 GEOAX (3 , Z2)

N110 G1

N120 GEOAX (3 , Z1)



O eixo Z2 opera como eixo Z

O eixo Z1 opera como eixo Z.

7.8 Eixos geométricos comutáveis, GEOAX



840 D
NCU 572
NCU 573



810D



840Di



Seqüência

Número do eixo geométrico

No comando GEOAX(n,eixo do canal...) o número n indica o eixo geométrico ao qual o eixo de canal especificado em seguida deve ser relacionado.

Os números entre 1 e 3 (X, Y e Z) podem ser utilizados para comutar um eixo de canal.

n = 0 remove a atribuição do eixo de canal ao eixo geométrico sem redefinir o eixo geométrico.

Um eixo retirado do grupo de eixos geométricos como resultado da comutação pode ser programado como eixo auxiliar através de seu nome no canal..



Todos os frames, zonas de proteção e limitações de área de trabalho são canceladas quando um eixo geométrico é comutado.

Coordenadas polares:

A troca de eixos geométricos através do comando GEOAX carrega as coordenadas polares modais com o valor 0, de forma análoga à alteração do plano de trabalho (G17–G19).

DRF, NPV:

Quaisquer deslocamentos programados através de nônio (DRF) ou de deslocamentos de origem externos permanecem ativos após a comutação.

Transferindo posições de eixos

Através da alocação de novos números aos eixos de canal definidos é também possível programar a alteração de suas posições dentro do grupo de eixos geométricos.

N... GEOAX (1, XX, 2, YY, 3, ZZ)

N... GEOAX (1, U, 2, V, 3, W)

O eixo de canal XX é o primeiro eixo geométrico, YY o segundo e ZZ o terceiro.

O eixo U será o primeiro, V o segundo e W o terceiro.



840D
NCU 572
NCU 573



810D



840Di



Pré condições e limitações

Desativando a comutação

O comando GEOAX() chama a configuração básica do grupo de eixos geométricos.

A configuração básica será automaticamente ativada após o POWER ON e durante a comutação para o modo de operação referenciamento.



Informações adicionais

Procedimento de comutação e correção de comprimento da

ferramenta Uma compensação de comprimento de ferramenta permanece ativo após a comutação. Isto irá afetar quaisquer novos eixos ou eixos geométricos cuja posição foi comutada. Quando o primeiro comando de movimento for programado para estes eixos, a trajetória resultante, a ser percorrida será portanto a soma do comprimento da ferramenta e a trajetória programada.

7.8 Eixos geométricos comutáveis, GEOAX



840 D
NCU 572
NCU 573



810D



840Di

Os eixos geométricos mantidos em suas posições no grupo mantêm seus estados relacionados à compensação de comprimento da ferramenta.

Configuração de eixos geométricos e alteração de transformação

A configuração de eixos geométricos aplicada a uma transformação ativa (definida em dados de máquina) não pode ser alterada através da função “Comutação de eixos geométricos”.

Caso você necessite alterar uma configuração de eixo geométrico em uma transformação, isto poderá ser feito somente após a programação de outra transformação.

A configuração de eixos geométricos alterada com GEOAX é cancelada ao ser ativada uma transformação.

Caso os valores carregados em dados de máquina para uma transformação e a comutação de eixos geométricos entrem em conflito, os valores carregados para a transformação terão prioridade.

Exemplo:

Uma transformação encontra-se ativa. De acordo com os valores programados em dados de máquina a transformação deve permanecer ativa após um RESET, entretanto, ao mesmo tempo, o RESET restabelece a configuração básica dos eixos geométricos. Neste caso, a configuração de eixos geométricos definida na transformação permanecerá ativa.



840D
NCU 572
NCU 573



810D



840Di



Exemplo de programação

Uma máquina possui seis eixos de canal chamados XX, YY, ZZ, U, V, W.

A configuração básica em dados de máquina é:

Eixo de canal XX = primeiro eixo geométrico (eixo X)

Eixo de canal YY = segundo eixo geométrico (eixo Y)

Eixo de canal ZZ = terceiro eixo geométrico (eixo Z)

| | | |
|------|----------------------------|--|
| N10 | GEOAX () | Configuração básica dos eixos geométricos encontra-se ativa. |
| N20 | G0 X0 Y0 Z0 U0 V0 W0 | Todos eixos para a posição 0 em avanço rápido. |
| N30 | GEOAX (1, U, 2, V, 3, W) | Redefine os eixos U como primeiro (X), V como segundo (Y) e W como terceiro (Z) eixos geométricos. |
| N40 | GEOAX (1, XX, 3, ZZ) | O eixo XX volta a ser o primeiro (X), ZZ a terceiro (Z). O eixo V permanece como segundo eixo geométrico (Y). |
| N50 | G17 G2 X20 I10 F1000 | Círculo completo no plano X, Y. os eixos de canal XX e Y se movimentam. |
| N60 | GEOAX (2, W) | O eixo W torna-se o segundo eixo geométrico (Y). |
| N80 | G17 G2 X20 I10 F1000 | Círculo completo no plano X, Y. os eixos de canal XX e W se movimentam. |
| N90 | GEOAX () | Reset ao estado inicial |
| N100 | GEOAX (1, U, 2, V, 3, W) | O eixo de canal U passa a ser o primeiro eixo geométrico (X), V o segundo (Y) e W o terceiro (Z). |
| N110 | G1 X10 Y10 Z10 XX=25 | Os eixos de canal U, V, W movimentam-se para a posição 10, XX como eixo auxiliar, para a posição 25 |
| N120 | GEOAX (0, V) | V é removido do grupo de eixos geométricos. U e W permanecem como primeiro (X) e terceiro (Z) eixos geométricos. O segundo eixo geométrico (Y) permanece indefinido. |
| N130 | GEOAX (1, U, 2, V, 3, W) | O eixo U permanece como sendo o primeiro eixo geométrico. V torna-se o segundo (Y) e W o terceiro (Z). |
| N140 | GEOAX (3, V) | V torna-se o terceiro eixo geométrico (Z), W é sobrescrito e portanto removido do grupo. O segundo eixo (y) permanece indefinido. |

7.8 Eixos geométricos comutáveis, GEOAX



840 D
NCU 572
NCU 573



810D



840Di

Corretores de ferramenta

| | | |
|-------|---|-------|
| 8.1 | Memória de corretores | 8-264 |
| 8.2 | Comandos para o gerenciamento de ferramentas..... | 8-266 |
| 8.3 | Corretor de ferramenta “online” PUTFTOCF, PUTFTOC, FTOCON, FTOCOF | 8-269 |
| 8.4 | Mantendo a correção de raio em nível constante, CUTCONON (SW 4 em diante) | 8-275 |
| 8.5 | Ativando corretores de ferramenta 3D | 8-278 |
| 8.6 | Orientação da ferramenta..... | 8-286 |
| 8.7 | Utilização livre de números D, número do incerto CE (A partir da SW5)..... | 8-291 |
| 8.7.1 | Checar números D (CHKDNO)..... | 8-292 |
| 8.7.2 | Renomeando números D (GETDNO, SETDNO)..... | 8-293 |
| 8.7.3 | Números T para os números D especificados (GETACTTD)..... | 8-294 |
| 8.7.4 | define números D restantes como inválidos | 8-295 |
| 8.8 | Cinemática do porta ferramenta..... | 8-296 |

8.1 Memória de corretores



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

8.1 Memória de corretores

Estrutura da memória dos corretores

Cada campo de dados pode ser chamado através de um número T e D (com exceção de “Número D Plano (flat)”), e contém outras informações, tal como tipo, além dos dados geométricos da ferramenta.

SW 4 em diante

A estrutura “Número D plano (flat)” é utilizada caso o gerenciamento de ferramentas seja feito externamente à NCK. Neste caso, os corretores são gerados com os blocos de correção associados, mas não estão relacionados à nenhuma ferramenta. Você pode continuar utilizando o número T em seu programa. Entretanto, este T não está associado ao número D programado.

Existem vários campos a serem preenchidos para as variáveis geométricas (p.e., comprimento 1 ou raio). Estes valores são adicionados de forma a produzir um valor (p.e., comprimento total 1, raio total) que serão utilizados em cálculos.

Os campos de correção não utilizados devem ser carregados com zero.



Os valores individuais da memória de correção de p1 à P25 podem ser lidos ou escritos pelo programa através de variáveis de sistema .

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

| Parâmetros da ferramenta | Significado | Comentário |
|-------------------------------------|---|-------------------------------------|
| Número (DP) | | |
| \$TC_DP 1 | Tipo de ferramenta | Vide lista para relação |
| \$TC_DP 2 | Direção da ponta da ferramenta | Somente para ferramentas de tornear |
| Geometria | Comp.comprimento da ferramenta | |
| \$TC_DP 3 | Comprimento 1 | Cálculo depende |
| \$TC_DP 4 | Comprimento 2 | Do tipo e plano |
| \$TC_DP 5 | Comprimento 3 | |
| Geometria | Raio | |
| \$TC_DP 6 | Raio | |
| \$TC_DP 7 | Largura da ranhura b de uma serra, raio de arredondamento para fresas | |
| \$TC_DP 8 | Ressalto k | Somente para serra com ranhuras |
| \$TC_DP 11 | Ângulo para fresas cônicas | |
| Desgaste | Comp.comprimento e raio da ferramenta | |
| \$TC_DP 12 | Comprimento 1 | |
| \$TC_DP 13 | Comprimento 2 | |
| \$TC_DP 14 | Comprimento 3 | |
| \$TC_DP 15 | Raio | |
| \$TC_DP 16 | Largura da ranhura b de uma serra, raio de arredondamento para fresas | |
| \$TC_DP 17 | Ressalto k | Somente para serra com ranhuras |
| \$TC_DP 20 | Ângulo para fresas cônicas | |
| Dimensões básicas/ adaptador | Compensação de comprimento da ferramenta | |
| \$TC_DP 21 | Comprimento 1 | |
| \$TC_DP 22 | Comprimento 2 | |
| \$TC_DP 23 | Comprimento 3 | |
| Tecnologia | | |
| \$TC_DP 24 | Ângulo de recuo | Para ferramentas de tornear |



Informações adicionais

Todos os outros parâmetros são reservados.

Fabricante de máquinas

Dados de inserto do usuário podem ser configurados através de dados de máquina.

8.2 Comandos para o gerenciamento de ferramentas



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

8.2 Comandos para o gerenciamento de ferramentas



Explicação dos comandos

| | |
|-------------------------|---|
| T= "WZ" | Seleção da ferramenta com nome |
| NEWT("WZ" , DUPL0_NO) | Cria nova ferramenta, número duplo opcional |
| DELT("WZ" , DUPL0_NO) | Apaga ferramenta, número duplo opcional |
| GETT("WZ" , DUPL0_NO) | Determina número T |
| SETPIECE(x, y) | Incrementa quantidade de peças |
| GETSEL(x) | Lê o número T pré selecionado (T No.) |
| "WZ" | Nome da ferramenta |
| DUPL0_NO | Quantidade |
| x | Número do fuso, programação opcional |



Caso você esteja utilizando o gerenciamento de ferramentas você pode criar e chamar ferramentas através de um nome, p.e. T="BROCA" ou T="123".



Função NEWT

Com a função NEWT você pode criar uma nova ferramenta no programa NC. A função automaticamente retorna o número T criado, que pode ser utilizado em seguida para endereçar a ferramenta.

Parâmetro de retorno=NEWT("WZ" , DUPL0_NO)

Caso não seja especificado um número duplo, o gerenciamento de ferramentas irá gera-lo automaticamente.

Exemplo:

```
DEF INT DUPL0_NO
DEF INT T_NO
DUPL0_NO = 7
T_NO=NEWT( "DRILL" , DUPL0_NO )
```

Cria a ferramenta "DRILL" com número duplo 7. O número interno da ferramenta criada é gravado em T_NO.

Função DELT

A função DELT pode ser utilizada para apagar uma ferramenta através do nome e duplo, sem se referir ao número T interno.

DELT("WZ" , DUPL0_NO)

8.2 Comandos para o gerenciamento de ferramentas



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Função GETT

A função GETT retorna o número T interno necessário para acessar os dados de uma ferramenta conhecida somente através de seu nome.

Parâmetro de retorno=GETT("WZ", DUPLO_NO)

Caso existam várias ferramentas com o nome especificado, o número da primeira ferramenta disponível será devolvido.

Parâmetro de retorno =-1: O nome da ferramenta ou duplo não pode ser associado a uma ferramenta.

Exemplos:

T="DRILL"

R10=GETT("DRILL", DUPLO_NO)

Retorna o número T de DRILL
com o número duplo =
DUPLO_NO

A ferramenta "DRILL" deve Ter sido anteriormente declarada com NEWT ou \$TC_TP1[].

\$TC_DP1[GETT("DRILL",
DUPLO_NO),1]=100

Escrita em um parâmetro de
ferramenta a partir do nome da
mesma

Função SETPIECE

Esta função é utilizada para atualizar a quantidade de peças nos dados de monitoração. A função decrementa os contadores de todos os incertos que foram utilizados desde a última ativação de SETPIECE para o fuso selecionado.

SETPIECE(x,y)

| | |
|---|---|
| x | Número de peças usinadas |
| y | y número do fuso, 0 indica o fuso principal (default) |

8.2 Comandos para o gerenciamento de ferramentas



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Função GETSELT

Esta função devolve o número da ferramenta pré selecionada a ser carregada no fuso.

Isto permite o acesso a dados da ferramenta antes da programação de M6 e possibilitar a realização de sincronismos antecipados.

Exemplo para troca de ferramentas utilizando o gerenciamento de ferramentas

T1 Pré seleção da ferramenta, ou seja, a ferramenta pode ser trazida do magazine para uma posição onde será posteriormente carregada no fuso.

M6 Carga da ferramenta pré selecionada (dependendo das parametrizações em dados de máquina, esta carga pode ser realizada sem M6)

Exemplo:

| | |
|----------------|--|
| T1 M6 | Carga da ferramenta 1 |
| D1 | Seleção da correção de comprimento da ferramenta |
| G1 X10 ... | Usinagem com T1 |
| T= "DRILL" | Pré seleção de DRILL |
| D2 Y20 ... | Alteração do incerto de T1 |
| X10 ... | Usinagem com T1 |
| M6 | Carga da ferramenta DRILL |
| SETPIECE (4) | Quantidade de peças usinadas |
| D1 G1 X10 ... | Usinagem com DRILL |



Uma lista completa contendo todas as variáveis necessárias ao gerenciamento de ferramentas encontra-se disponível na lista de variáveis de sistema em "Anexo" deste manual.



840D
NCU 572
NCU 573



840Di

8.3 Corretor de ferramenta "online" PUTFTOCF, PUTFTOC, FTOCON, FTOCOF



Programação

```
FCTDEF(Nr_Polinômio, ILimite, Slimite, a0, a1, a2, a3)
PUTFTOCF(Nr_Polinômio, Val_ref, Compr1_2_3, Canal, Fuso)
PUTFTOC(Valor, Compr1_2_3, Canal, Fuso)
FTOCON
```



Explicação dos comandos

| | |
|----------|--|
| PUTFTOCF | Escrita online de todos os corretores continuamente |
| FCTDEF | Define parâmetros para a função PUTFTOCF |
| PUTFTOC | Escrita dos corretores de ferramenta individualmente |
| FTOCON | Ativa os corretores online |
| FTOCOF | Desativa os corretores online |



Explicação dos parâmetros

| | |
|---|---|
| Nr_Polinômio | Valores 1-3: Um máximo de 3 polinômios podem ser programados ao mesmo tempo, com grau até 3 |
| Val_ref | Valor referência a partir dos quais os corretores serão derivados |
| Compr1_2_3 | Parâmetro de desgaste no qual o valor de correção da ferramenta será adicionado |
| Canal | Número do canal no qual o corretor da ferramenta será ativado, especificado somente se o canal for outro se não o atual |
| Fuso | Número do fuso no qual os corretores online atuam: especificação necessária somente para rebolos inativos |
| ILimite | Limite inferior |
| SLimite | Limite superior |
| a ₀ , a ₁ , a ₂ , a ₃ | Coeficientes ou polinômio |
| Valor | Valor adicionado ao parâmetro de desgaste |



840D
NCU 572
NCU 573



840Di

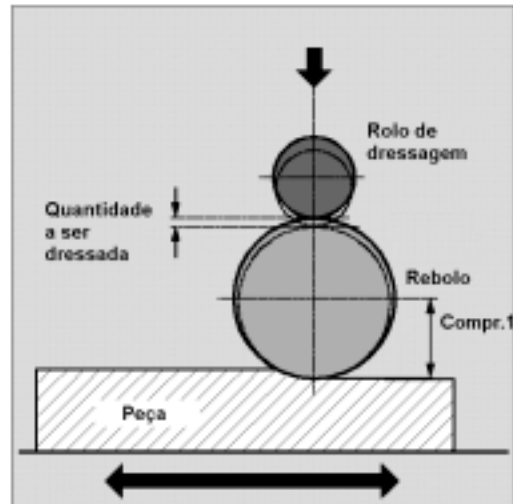


Função

A função aplica imediatamente os corretores de ferramenta resultantes da usinagem através da compensação de ferramenta online (p.e. dressamento: o rebolo é dressado em junto com a usinagem). O comprimento da ferramenta pode ser alterado a partir do canal de usinagem ou de um canal paralelo (canal do dressador).



Somente os corretores online podem ser atribuídos à rebolos.



Informações gerais sobre TO online

Dependendo do tempo de dressagem, as seguintes funções são utilizadas para a escrita de corretores online:

- Escrita contínua, não modal : PUTFTOCF
- Escrita contínua modal ID=1 DO FTOC
(Vide capítulo ações síncronas)
- Escrita discreta: PUTFTOC

No caso de escrita contínua (para cada pulso de interpolação) seguida da instrução de uso, cada alteração é calculada de forma aditiva na memória de desgaste de forma a evitar saltos no valor desejado de movimento.

Em ambos os casos:

O corretor de ferramentas online pode atualizar em cada fuso com os parâmetros de desgaste 1, 2 ou 3.

A referência entre os comprimentos e os eixos geométricos é feita de acordo com o plano selecionado.

A relação entre o fuso e ferramenta é feita de acordo com os dados de ferramenta e com GWPERSON ou TMON pois não se referem ao rebolo ativo (vide manual de programação funções básicas)

8.3 Corretor ferr. "online" PUTFTOCF, PUTFTOC, FTOCON, FTOCOF



840D
NCU 572
NCU 573



840Di

Um corretor é sempre aplicado considerando-se os parâmetros de desgaste do lado atual da ferramenta, ou no lado esquerdo em ferramentas inativas.



Quando os corretores são idênticos para vários lados da ferramenta, os valores podem ser transferidos automaticamente para o segundo lado através de uma relação de encadeamento (Vide descrição no Manual de Operação).



Caso corretores online seja, definidos para um canal de usinagem, você não poderá alterar os valores de desgaste da ferramenta atual neste canal através do programa, ou através de uma intervenção do operador.



O corretor online da ferramenta também é utilizado é também aplicado respectivamente à velocidade periférica constante para retíficas (GWPS) em conjunto com a monitoração da ferramenta (TMON) e retífica centerless (CLGON).



Seqüência

PUTFTOCF = Escrita contínua

O processo de dressagem é realizado em conjunto com a usinagem:

Dressamento por toda a largura de um rebolo com rolo ou diamante de dressagem de um lado para outro do rebolo.

A usinagem e dressagem podem ser realizadas em canais diferentes. Caso não existam canais programados, o corretor terá efeito no canal ativo.

```
PUTFTOCF(Nr_Polinômio, Valor_Ref, Compr1_2_3, Canal, Fuso)
```

O corretor da ferramenta é constantemente modificado no canal de usinagem de acordo com uma função de grau 1, 2 ou 3, que devem Ter sido definidos anteriormente com FCTDEF.



840D
NCU 572
NCU 573



840Di

Conjunto de parâmetros para a função FCTDEF

Os parâmetros são definidos em um bloco separado:

```
FCTDEF(Nr_Polinômio, ILimite, Slimite, a0, a1, a2, a3)
```

O polinômio pode ser de primeiro, segundo ou terceiro grau.

Limites de valores Ilimite= limite inferior, Slimite= limite superior.

Exemplo:

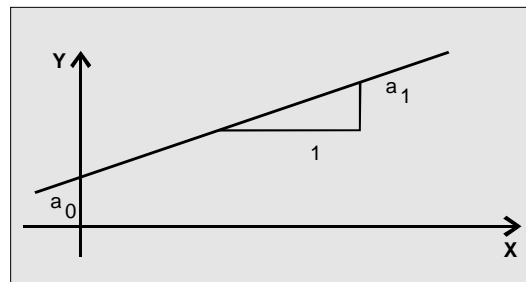
Linha reta ($y = a_0 + a_1x$) com gradiente 1

```
FCTDEF(1, -1000, 1000, -$AA_IW[X], 1)
```

Escrita do corretor online de forma discreta: PUTFTOC

Este comando pode ser utilizado para escrever **uma vez** o valor do corretor. O corretor é ativado imediatamente no canal de destino:

O rebolo é dressado a partir de um canal paralelo, mas não ao mesmo tempo de usinagem.



```
PUTFTOC(Valor, Compr.1_2_3, Canal, fuso)
```

O corretor de ferramenta online para os comprimento especificado 1, 2 **ou** 3 é alterado pelo valor especificado, ou seja o valor é adicionado ao parâmetro de desgaste.

Inclusão do corretor de ferramenta online: FTOCON, FTOCOF

O canal de destino pode somente receber os corretores online enquanto FTOCON estiver ativo.

- FTOCON deve ser escrito no canal em que o corretor deva ser ativado.
Com FTOCOF, o corretor não mais será aplicado, entretanto o valor total escrito com PUTFTOC é corrigido no dado de correção específico do incerto.
- FTOCOF é sempre o valor de reset.
- PUTFTOCF sempre atua no bloco de movimento subsequente.
- O corretor online de ferramenta pode também ser selecionado de forma moda com FTOC.



840D
NCU 572
NCU 573



840Di

Favor consultar a seção "Ações síncronas de movimento" para maiores informações.



Exemplo de programação

Tarefa

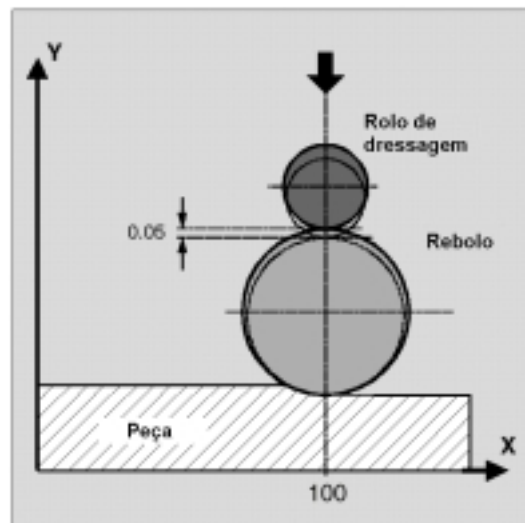
Em uma retífica de superfície com os parâmetros a seguir, o rebolo deve ser dressado em um total de 0.05 após o início do movimento em X100. O valor total de dressagem deve ser ativado com escrita de corretores online contínua.

Y: Eixo de avanço do rebolo

V: Eixo de avanço do rolo de dressagem

Máquina: Canal 1 com os eixos X, Z, Y

Dressador: Canal 2 com eixo V



Programa de usinagem no canal 1:

%_N_MACH_MPF

...

| | |
|-----------------------------|--|
| N110 G1 G18 F10 G90 | Posição básica |
| N120 T1 D1 | Seleção da ferramenta atual |
| N130 S100 M3 X100 | Ligar fuso, movimento para a posição inicial |
| N140 INIT (2, "DRESS", "S") | Seleção do programa de dressagem do canal 2 |
| N150 START (2) | Início do programa de dressagem no canal 2 |
| N160 X200 | Movimento para a posição de destino |
| N170 FTOCON | Ativa corretor online |
| N... G1 X100 | Continua a usinagem |
| N...M30 | |

Programa de dressagem do canal 2:

%_N_DRESS_MPF

...

| | |
|---|--|
| N40 FCTDEF (1, -1000, 1000, -\$AA_IW[V], 1) | Define a função: Linha reta |
| N50 PUTFTOCF (1, \$AA_IW[V], 3, 1) | Escrita contínua do corretor online: Comprimento 3 do rebolo atual é derivado do movimento do eixo V e corrigido no canal 1 |
| N60 V-0.05 G1 F0.01 G91 | Movimento de avanço para dressagem, PUTFTOCF ativo somente neste bloco |

...

N... M30

8.3 Corretor ferr. “online” PUTFTOCF, PUTFTOC, FTOCON, FTOCOF

840D
NCU 572
NCU 573



840Di

Programa de dressagem, modal:

```
%_N_DRESS_MPF
```

```
FCTDEF(1,-1000,1000,-  
$AA_IW[V],1)
```

Define função

```
ID=1 DO FTOC(1,$AA_IW[V],3,1)
```

Seleção do corretor de ferramenta online:
Valor atual do eixo V é o valor de entrada para o polinômio 1, o resultado é adicionado ao comprimento 3 do rebolo ativo no canal 1 como valor de correção.

```
WAITM(1,1,2)
```

Sincronismo com o canal de usinagem

```
G1 V-0.05 F0.01, G91
```

Movimento de avanço para dressagem

```
G1 V-0.05 F0.02
```

```
...
```

```
CANCEL(1)
```

Desseleção do corretor online

```
...
```

8.4 Mantendo a correção de raio em nível constante, CUTCONON



840D
NCU 572
NCU 573



840Di

8.4 Mantendo a correção de raio em nível constante, CUTCONON (SW 4 em diante)



Programação

CUTCONON
CUTCONOF



Explicação

| | |
|----------|--|
| CUTCONON | Ativa a função correção de raio constante |
| CUTCONOF | Desativa a função correção de raio constante (default) |



Função

A função “Correção constante de raio da ferramenta” é utilizada para suprimir a correção de raio da ferramenta para um número definido de blocos enquanto mantida a diferença entre a trajetória programada e atual do ponto central da ferramenta acumulada nos blocos anteriores como um deslocamento.

Isto pode ser prático, por exemplo, caso vários blocos de movimentos sejam necessários nos pontos de reversão durante a fresagem linha a linha, e o contorno gerado pela correção do raio da ferramenta não seja desejado (estratégia de bypass).

Isto pode ser utilizado de acordo com o tipo de correção de raio da ferramenta (fresa de face 2 1/2D, 3D, fresa de circunferência 3D)



Seqüência

A correção de raio da ferramenta encontra-se normalmente ativa antes da supressão de compensação e será reativada após a desativar a supressão novamente.

O ponto de correção na posição final do bloco será atingido através do último bloco antes de CUTCONON.

Todos os blocos a seguir serão executados sem compensação de raio da ferramenta.

8.4 Mantendo a correção de raio em nível constante, CUTCONON



840D
NCU 572
NCU 573



840Di

Entretanto, serão deslocados pelo vetor a partir do ponto final do último bloco de comando até seu ponto de correção.

O tipo de interpolação para estes blocos (linear, circular, polinomial) é arbitrária.

A desativação do bloco de supressão de compensação, ou seja, o bloco contendo CUTCONOF, é geralmente corrigido, sendo iniciado na posição de correção do ponto inicial.

Um bloco linear é inserido entre este ponto e o ponto final do bloco anterior, ou seja, o último movimento programado com CUTCONON ativo.

Blocos circulares nos quais o círculo plano é perpendicular ao plano de compensação (círculos verticais) são tratados como se CUTCONON fosse programado.

Esta supressão implícita da é automaticamente cancelada no primeiro bloco de movimento que não execute um círculo deste tipo e que contenha um movimento no plano de compensação.

Círculos verticais deste tipo podem ocorrer somente em fresagens de circunferências.



Exemplo

| | | |
|------|------------------------------|---|
| N10 | ; | Define ferramenta d1 |
| N20 | \$TC_DP1[1,1]= 110 ; | Tipo |
| N30 | \$TC_DP6[1,1]= 10. ; | Raio |
| N40 | | |
| N50 | X0 Y0 Z0 G1 G17 T1 D1 F10000 | |
| N60 | | |
| N70 | X20 G42 NORM | |
| N80 | X30 | |
| N90 | Y20 | |
| N100 | X10 CUTCONON; | Ativa a supressão de correção |
| N110 | Y30 CONT ; | Inserir o círculo de bypass caso necessário na desativação de supressão de contorno |
| N120 | X-10 CUTCONOF | |
| N130 | Y20 NORM ; | Sem círculo de bypass na desativação de TRC |
| N140 | X0 Y0 G40 | |
| N150 | M30 | |

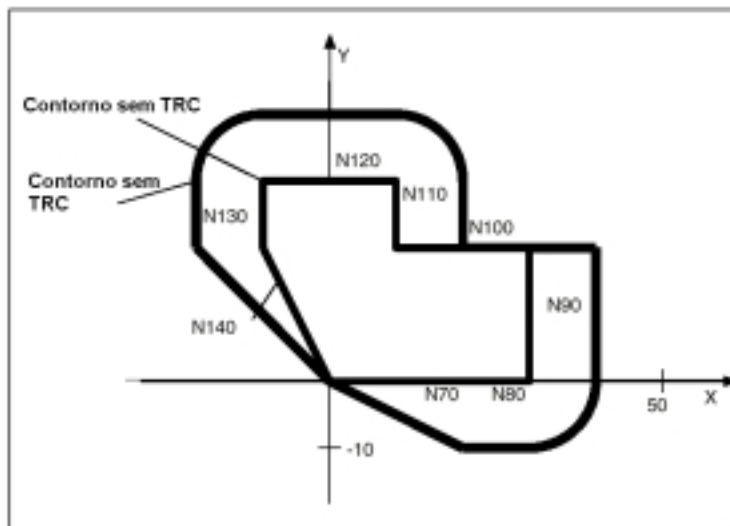
8.4 Mantendo a correção de raio em nível constante, CUTCONON



840D
NCU 572
NCU 573



840Di



Informações adicionais

1. CUTCONON não possui efeito caso a compensação de raio não esteja ativa (G40). Um alarme será emitido. Entretanto, o código G permanece ativo. Isto significa que a correção de raio da ferramenta será ativada em um bloco subsequente com G41 ou G42.
2. É possível alterar o código G no sétimo grupo de código G (correção de raio da ferramenta; G40 / G41 / G42) enquanto CUTCONON permanecer ativo. Uma mudança para G40 será imediatamente ativada. O corretor com o qual o bloco anterior foi executado será aplicado.
3. Caso CUTCONON ou CUTCONOF seja programado em um bloco sem movimento no plano de compensação ativo, a alteração não será efetuada até que atingido um bloco com tal tipo de movimento.

Maiores informações: /FB/, W1 Corretor de ferramenta

8.5 Ativando corretores de ferramenta 3D



840D
NCU 572
NCU 573



840Di

8.5 Ativando corretores de ferramenta 3D



Explicação

| | |
|-----------|--|
| CUT3DC | Ativação de correção de raio 3D para fresa de circunferência |
| CUT3DFS | Corretor de ferramenta 3D para fresa de face com orientação constante. A orientação da ferramenta é determinada por G17-g19 e não é influenciada pelos frames. |
| CUT3DFF | Corretor de ferramenta 3D para fresa de face com orientação constante. A orientação da ferramenta é a direção determinada por g17-G19 e possivelmente rotacionada por um frame |
| CUT3DF | Corretor de ferramenta 3D para fresa de face com alteração de orientação (somente com transformação de cinco eixos) |
| G40 X Y Z | Para desativar: Bloco linear G0/G1 com eixos geométricos |
| ISD=Valor | Profundidade de inserção |



Os comandos são modais e encontram-se no mesmo grupo de CUT2D e CUT2DF.

O comando não é desselecionado até que um próximo movimento no plano atual seja executado. Isto sempre se aplica a G40 e é independente do comando CUT.



Função

Uma alteração na orientação da ferramenta é levada em conta na compensação de raio da ferramenta, 3D para ferramentas cilíndricas.

Os mesmos comandos de programação utilizados na compensação 2D são aplicados à compensação de raio da ferramenta 3D. Com G41/G42, as correções dos lados esquerdo/direito são especificados na direção do movimento. O método de aproximação é sempre NORM.



840D
NCU 572
NCU 573



840Di



Exemplo

| | | | | | | | |
|-----|-----------|-----|-----|----|----|-------|--|
| N10 | A0 | B0 | X0 | Y0 | Z0 | F5000 | |
| N20 | T1 | D1 | | | | | Chamada da ferramenta, chamada dos valores de correção |
| N30 | TRAORI(1) | | | | | | Seleção da transformação |
| N40 | CUT3DC | | | | | | Seleção da compensação de raio 3D da ferramenta |
| N50 | G42 | X10 | Y10 | | | | Seleção da correção de raio da ferramenta |
| N60 | X60 | | | | | | |
| N70 | ... | | | | | | |



Informações adicionais

Blocos intermediários são permitidos com compensação de raio 3D da ferramenta. As regras para correção de raio 2 1/2D são aplicadas.

As correções de raio da ferramenta 3D encontram-se ativas somente quando uma transformação de cinco eixos estiver selecionada.

Um bloco circular é sempre inserido nos cantos externos. G450/G451 não produzem efeito.

O comando DISC é ignorado.

8.5 Ativando corretores de ferramenta 3D



840D
NCU 572
NCU 573



840Di

Diferenças entre as correções de raio de ferramenta 2 1/2D e 3D

Na correção de raio 3D a orientação da ferramenta pode ser alterada.

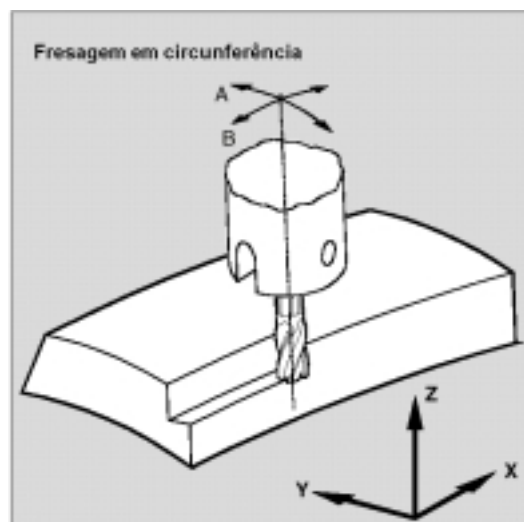
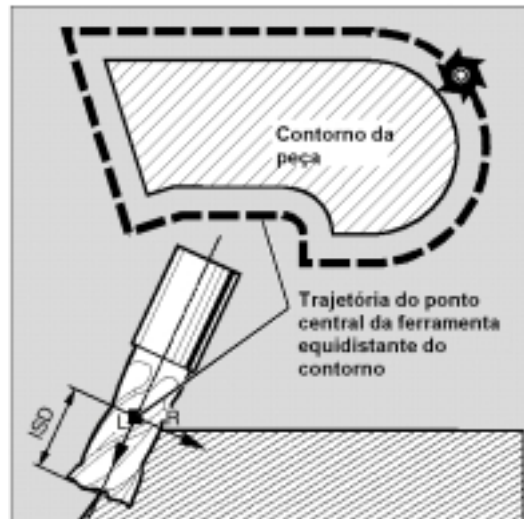
2 1/2D assumem o uso de ferramenta com orientação constante.

A correção de raio 3D é também chamada de correção de raio 5D, pois neste caso existem 5 graus de liberdade de orientação disponíveis para a ferramenta no espaço.

Fresagem circunferencial

O tipo de fresagem aqui utilizada é implementada através da definição de uma trajetória (linha guia) e da orientação correspondente. Neste tipo de usinagem, a forma da ferramenta na trajetória não é relevante. O único fator decisivo é o raio da ferramenta no ponto de inserção.

A função 3D TRC é limitada à ferramentas cilíndricas.





840D
NCU 572
NCU 573



840Di

Fresa de face

Para este tipo de fresagem 3D, você necessita definir linha a linha as trajetória 3D na superfície da peça.

O formato da ferramenta e dimensões são levados em conta nos cálculos, e estes são geralmente executados por um CAM.

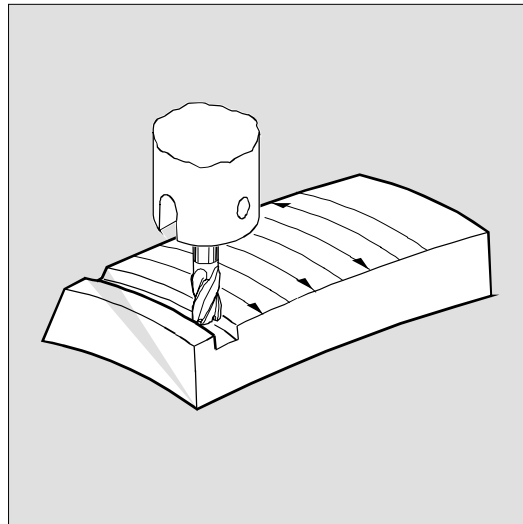
Adicionalmente aos blocos NC, o pós processamento descreve as orientações da ferramenta (enquanto a transformação de 5 eixos estiver ativa) e o código G para os corretores 3D da ferramenta no programa de usinagem.

Esta característica oferece ao operador da máquina a opção de utilizar ferramentas um pouco menores que as utilizadas no cálculo das trajetórias NC.

Exemplo:

Blocos NC foram calculados com uma fresa de 10 mm.

Neste caso, a peça pode ser usinada com uma fresa de diâmetro 9.9 mm, embora isto irá resultar em um contorno diferente de superfície.



8.5 Ativando corretores de ferramenta 3D



840D
NCU 572
NCU 573



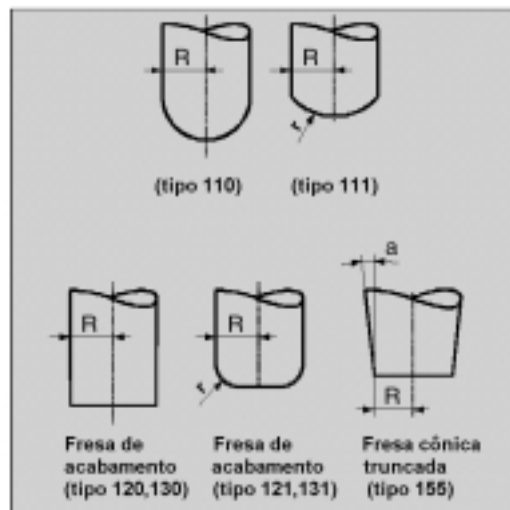
840Di

Formato de fresas, dados de ferramenta

A tabela abaixo fornece uma relação dos formatos de ferramenta que podem ser utilizados em operações de fresagem de faces bem como os valores limite dos dados de ferramenta.

A forma do corpo da ferramenta não é levada em consideração – as ferramentas tipo 120 e 155 são possuem efeitos idênticos.

Caso tipo diferente seja utilizado no programa NC que não esteja listado na tabela, o sistema automaticamente assume o uso de uma ferramenta tipo 110. Um alarme será emitido caso os valores limiares dos dados de ferramenta sejam violados.



Ferramentas tipo fresa

| | Tipo No. | R | r | a |
|--|----------|----|----|----|
| Fresa cilíndrica | 110 | >0 | X | X |
| Fresa de acabamento esférica | 111 | >0 | >R | X |
| Fresa de acabamento, incertos em ângulo | 120, 130 | >0 | X | X |
| Fresa de acabamento, incertos em ângulo com arredondamento de cantos | 121, 131 | >r | >0 | X |
| Fresa cônica truncada | 155 | >0 | X | >0 |

X=não considerado

Compensação de comprimento da ferramenta

A ponta da ferramenta é a referência para a compensação de comprimento (interseção longitudinal eixo/superfície)



840D
NCU 572
NCU 573



840Di

Corretor de ferramenta 3D, troca de ferramenta

Uma nova ferramenta com dimensões alteradas (R, r, a) ou formato diferente pode ser especificada somente através da programação de G41 ou G42 9Transição de G40 para G41 ou G42, reprogramação de G41 ou G42).

Esta regra não se aplica a nenhum outro dado de ferramenta, como por exemplo comprimentos, portanto as ferramentas deste tipo podem ser utilizadas sem a reprogramação de G41 ou G42.

Correção da trajetória

Com respeito à fresa de face, é aconselhável examinar o que acontece quando o ponto de contato “salta” da superfície conforme ilustrado no exemplo à direita, onde uma superfície convexa está sendo usinada com uma ferramenta posicionada na vertical.

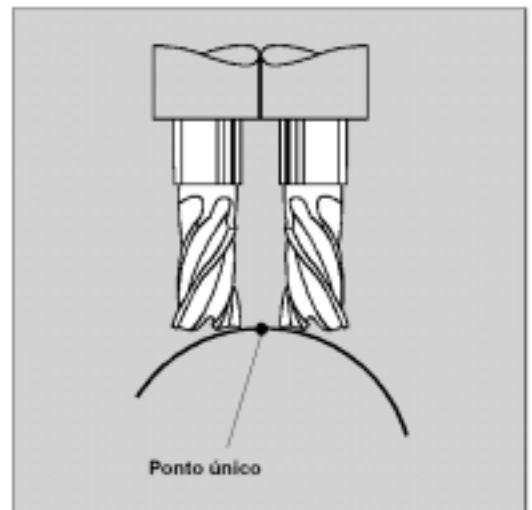
Como regra geral, é aconselhável selecionar um ferramenta com formato e orientação adequados para produzir a superfície desejada.

A aplicação mostrada no exemplo pode entretanto ser considerada como um caso extremo.

Este caso extremo é monitorado pelo controle que detecta alterações abruptas no ponto de usinagem na base dos movimentos de aproximação entre a ferramenta e os vetores normais à superfície. O controle insere blocos lineares nestes pontos, de forma que o movimento possa ser executado.

Estes blocos lineares são calculados com base nas faixas angulares permitidas para o ângulo lateral gravado em dados de máquina.

O sistema emite um alarme caso os valores limites gravados em dados de máquina sejam violados.



8.5 Ativando corretores de ferramenta 3D



840D
NCU 572
NCU 573



840Di

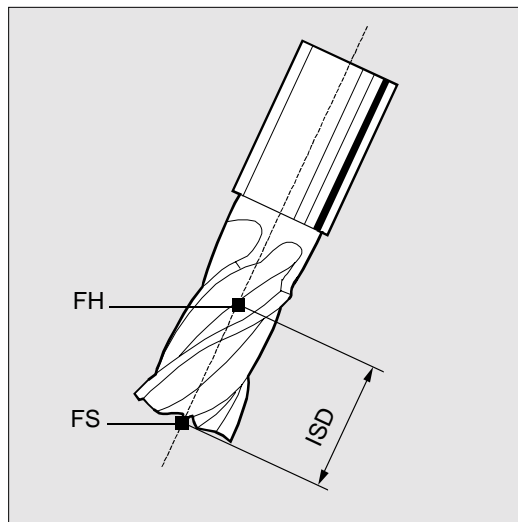
Curvatura de trajetória

A curvatura da trajetória não é monitorada. Em tais casos, é aconselhável utilizar somente ferramentas de tipo que não violem o contorno.

Profundidade de inserção (ISD)

A instrução ISD (profundidade de inserção) é utilizada para programar a profundidade de inserção para operações de fresagem periférica. Isto torna possível a alteração da posição do ponto de usinagem de acordo com a superfície externa da ferramenta.

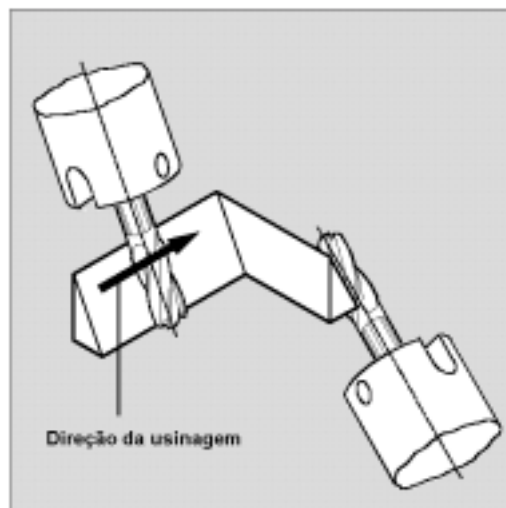
ISD especifica a distância entre o ponto de corte da ferramenta (FS) e o ponto de referência de corte (FH). O ponto FH é produzido através da projeção do ponto de usinagem programado ao longo do eixo da ferramenta. ISD será somente considerado quando a compensação de raio 3d estiver ativa.



Cantos interiores/exteriores

Cantos interiores e exteriores são tratados de forma separada. O termo interior ou exterior depende da orientação da ferramenta.

Quando ocorrem alterações na orientação em um canto, o tipo de canto pode ser alterado durante a usinagem. Caso isto ocorra, a usinagem é interrompida e uma mensagem de erro é gerada.





840D
NCU 572
NCU 573



840Di

Procedimento de interseção para compensação 3D (a partir da SW 5)

Com fresas 3D circunferências, o código G G450/G451 é então avaliado nos cantos externos; isto significa que a interseção dos deslocamentos nas curvas podem ser aproximados. Com a SW4, um círculo era sempre inserido nos cantos externos. O novo funcionamento é particularmente vantajoso para os programas 3D gerados através de CAD, Eles freqüentemente são compostos de pequenos blocos em linha reta (para aproximar uma curva de forma suave), onde as transições são quase tangenciais entre os blocos adjacentes

Até o presente, com a correção de raio da ferramenta na parte exterior do contorno, círculos eram geralmente inseridos para circundar os cantos externos. Estes blocos podem ser bem pequenos com transições quase tangenciais, resultando em quedas não desejadas de velocidade.

Nestes casos, como nas compensações de raio 2 1/2 D, Ambas as curvas envolvidas são alongadas e a interseção de ambas é aproximada.

A interseção é determinada através da extensão dos deslocamentos das curvas em ambos blocos e definindo sua interseção no canto no plano perpendicular à orientação da ferramenta. Caso não exista esta interseção, o canto será processado anteriormente, e com isto, um círculo inserido.



Para maiores informações sobre os procedimentos de interseção, vide /FB/ W5, Compensação de raio de ferramenta 3D

8.6 Orientação da ferramenta



840D
NCU 572
NCU 573

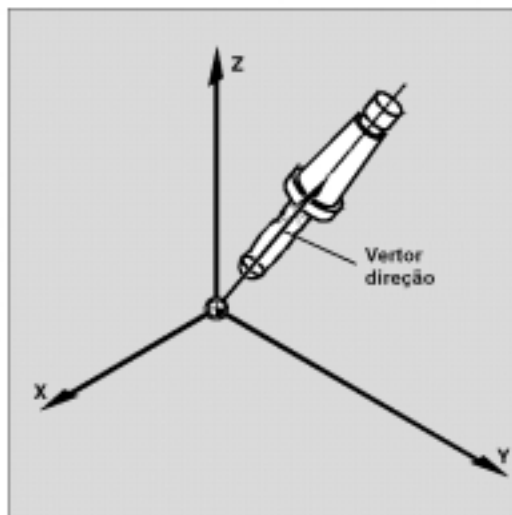


840Di

8.6 Orientação da ferramenta



Orientação da ferramenta é o termo dado para o alinhamento geométrico da ferramenta no espaço. Em uma máquina ferramenta de 5 eixos, a orientação da ferramenta pode ser controlada através de comandos no programa.



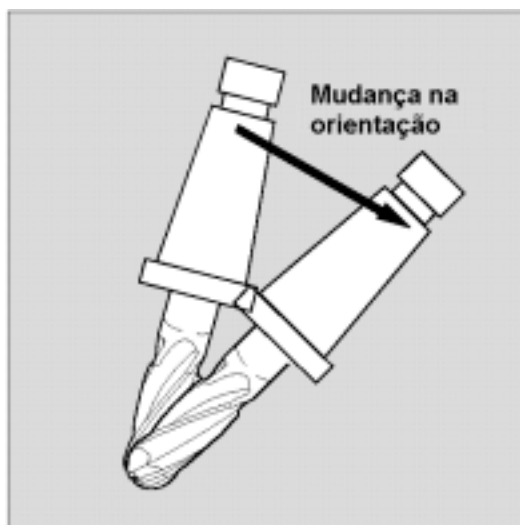
Programando a orientação da ferramenta

Uma alteração na orientação da ferramenta pode ser programada através de:

- Programação direta do eixo rotativo
- Ângulos euler ou RPY
- Vetor de direção
- LEAD/TILT (Fresa de face)

O sistema de coordenadas referência pode ser tanto o sistema de coordenadas da máquina (ORIMCS) como o sistema de coordenadas da peça atual (ORIWCs).

A mudança de orientação pode ser controlada da seguinte forma:



| | |
|------|---|
| ORIC | Movimento de orientação e trajetória em paralelo |
| ORID | Movimento de orientação e trajetória consecutivo |
| OSOF | Sem orientação de alisamento |
| OSC | Orientação constante |
| OSS | Orientação de alisamento no início do bloco |
| OSSE | Orientação de alisamento no fim do bloco |
| ORIS | Velocidade de alteração de orientação em graus por mm, válido para OSS e OSSE |



840D
NCU 572
NCU 573



840Di

Comportamento nos cantos externos

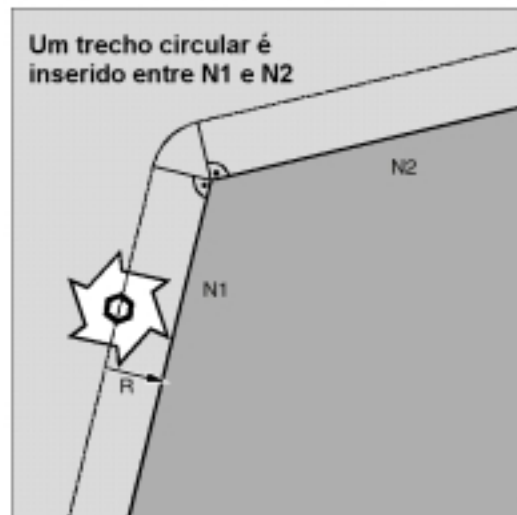
Um bloco circular com o raio de corte é sempre inserido em um canto externo.

Os comandos de programa ORIC e ORID podem ser utilizados para definir quando as alterações de orientação programadas entre os blocos N1 e N2 serão executadas antes do início do círculo inserido ou simultaneamente.

Caso uma alteração na orientação seja necessária nos cantos externos, esta pode ser realizado tanto simultaneamente com a interpolação ou separadamente, junto ao movimento de trajetória.

Com ORID, os blocos inseridos são executados inicialmente sem um movimento de trajetória. A geração do bloco de círculo no canto é imediatamente inserida antes do segundo bloco de movimento.

Caso vários blocos sejam inseridos em um canto externo e ORIC seja selecionado, o movimento circular é dividido entre os blocos inseridos de acordo com os valores da alteração de orientação.



8.6 Orientação da ferramenta



840D
NCU 572
NCU 573

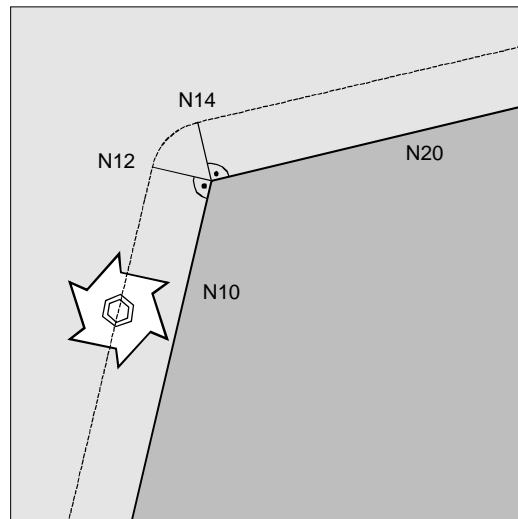


840Di



Exemplo de programação para ORIC

Caso dois ou mais blocos com alteração de orientação (p.e. A2= B2= C2=) sejam programados entre os blocos N10 e N20 enquanto ORIC esteja ativo, o bloco circular inserido é dividido entre estes blocos intermediários de acordo com os valores de alteração dos ângulos.



ORIC

N8 A2=... B2=... C2=...

N10 X... Y... Z...

N12 C2=... B2=...

N14 C2=... B2=...

O bloco circular inserido no canto externo é dividido entre N12 e N14 de acordo com a alteração na orientação. O movimento circular e a alteração de orientação são executados em paralelo.

N20 X =...Y=... Z=... G1 F200



840D
NCU 572
NCU 573

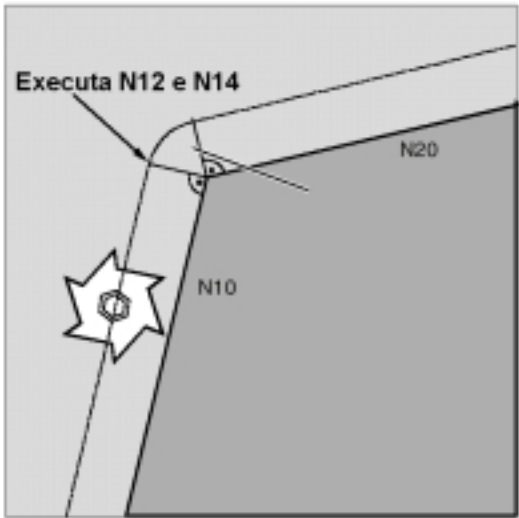


840Di



Exemplo de programação para ORID

Caso ORID esteja ativo, todos os blocos entre os dois blocos serão executados no fim do primeiro bloco. O círculo com orientação constante é executado imediatamente antes do segundo bloco de movimento.



ORID

N8 A2=... B2=... C2=...

N10 X... Y... Z...

N12 A2=... B2=... C2=...

Os blocos N12 e n14 são executados no fim de N10. O bloco de círculo com a orientação atual será comandado em seguida.

N14 M20

Funções auxiliares, etc.

N20 X... Y... Z...



O comando de programa que é ativado no primeiro bloco de movimento de um canto externo determina o tipo de mudança de orientação.

8.6 Orientação da ferramenta



840D
NCU 572
NCU 573



840Di



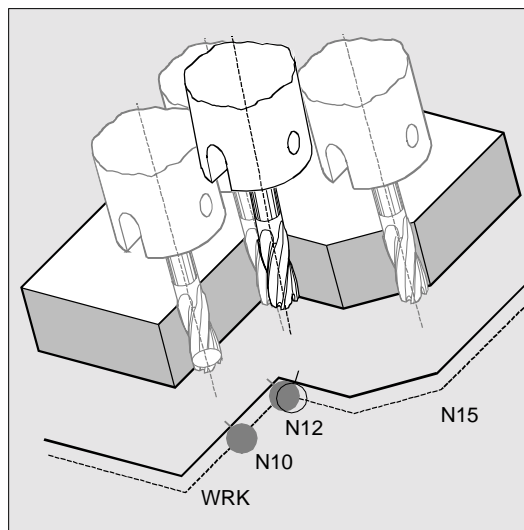
Sem alteração de orientação

Caso a orientação não seja alterada no limiar do bloco, o corte transversal da ferramenta é o círculo que toca ambos os contornos.



Exemplo de programação

Alteração de orientação em um canto interno



ORIC

N10 X ...Y... Z... G1 F500

N12 X ...Y... Z... A2=... B2=..., C2=...

N15 X Y Z A2 B2 C2

8.7 Utilização livre de núm.D, núm.do incerto CE (A partir da SW5)



840D
NCU 572
NCU 573



840Di

8.7 Utilização livre de números D, número do incerto CE (A partir da SW5)

A partir da SW5, você pode usar os números D como números de contornos. Você pode também endereçar o número do incerto através do endereço CE.

Você pode usar o parâmetro de sistema \$TC_DPCE para descrever o número do incerto.

Preset: número do corretor == número do incerto

Referências: FB, W1 (corretor de ferramenta)



Fabricante de máquinas (MH 8.12)

A quantidade máxima de números D (quantidade de incertos) por ferramenta é definido através de dados de máquina. Os seguintes comandos fazem sentido somente quando a quantidade máxima de incertos (MD18105) seja maior que o a quantidade de incertos por ferramenta (MD18106). Favor consultar os dados fornecidos pelo fabricante da máquina.



Informações adicionais

Além do número D relativo, você pode também utilizar os números D “planos”(flat) ou “absolutos” (1-32000), que não possuem relação ao número T (através da função “estrutura de números D plano (flat))

8.7 Utilização livre de núm.D, núm.do incerto CE (A partir da SW5)

840D
NCU 572
NCU 573



840Di

8.7.1 Checar números D (CHKDNO)**Programação**

Estado=CHKDNO (Tno1 , Tno2 , Dno)

**Explicação dos parâmetros**

| | | |
|------------------------|--|--|
| estado | TRUE: | Os números D estão associados somente para as áreas checadas. |
| | FALSE: | Existe conflito de números D ou os parâmetros são inválidos. Tno1, Tno2 e Dno devolvem os parâmetros que causam o conflito. Estes dados podem ser avaliados no programa. |
| CHKDNO (Tno1 , Tno2) | Todos os números D da peça especificada são checados. | |
| CHKDNO (Tno1) | Todos os números D de Tno1 são checados com relação às outras ferramentas | |
| CHKDNO | Todos os números D de todas as ferramentas são checados com relação à todas as outras ferramentas. | |

**Função**

CKKDNO checa quando os números D disponíveis são definidos de forma única.

Os números D de todas as ferramentas definidas em uma unidade TO devem somente estar presentes uma vez. Ferramentas de reposição não são consideradas.

8.7 Utilização livre de núm.D, núm.do incerto CE (A partir da SW5)



840D
NCU 572
NCU 573



840Di

8.7.2 Renomeando números D (GETDNO, SETDNO)



Programação

```
d = GETDNO(t,ce)
```

```
Estado = SETDNO(t,ce,d)
```



Explicação dos parâmetros

| | |
|--------|--|
| D | Número D do incerto da ferramenta |
| T | Número T da ferramenta |
| Ce | Número do incerto (número CE) da ferramenta |
| Estado | Indica quando o comando pode ser excetuado (TRUE ou FALSE) |



Função

GETDNO

Este comando retorna o número D de um incerto em particular (ce) de uma ferramenta com o número t. Caso não exista número D para os parâmetros especificados d é zerado. Caso o número D seja inválido, o retorno será um valor maior que 32000.

SETDNO

Este comando atribui o valor d para o número D para um incerto ce da ferramenta t. O resultado desta instrução é retornado através de estado (TRUE ou FALSE).

Caso não exista o bloco de dados para o parâmetro específico, o valor FALSE será o estado de retorno. Erros de sintaxe produzem um alarme. O número D não pode ser zerado de forma explícita.



Exemplo: (renomeando um número D)

```
$TC_DP2[1,2] = 120
$TC_DP3[1,2] = 5.5
$TC_DPCE[1,2] = 3; número de incerto
CE
...
N10 def int DNoOld, DNoNew = 17
N20 DNoOld = GETDNO(1,3)
N30 SETDNO(1,3,DNoNew)
```


8.7 Utilização livre de núm.D, núm.do incerto CE (A partir da SW5)



840D
NCU 572
NCU 573



840Di

Estas instruções atribuem ao incerto CE=3 um novo valor D 17.

A partir de então, estes dados para o incerto serão endereçados através do número D 17; tanto através dos parâmetros de sistema e na programação dos endereços NC.



Informações adicionais

Você deve definir números D únicos. Dois incertos diferentes de uma ferramenta não podem possuir o mesmo número D.

8.7.3 Números T para os números D especificados (GETACTTD)



Programação

Estado = GETACTTD(Tno, Dno)



Explicação dos parâmetros

| | |
|--------|--|
| Dno | Número D para o qual será procurado o número T. |
| Tno | Número T encontrado |
| Estado | 0: O número T foi encontrado. Tno contém o valor do número T. -1: O número D especificado não possui um número T; Tno=0. -2: O número D não é absoluto. Tno contém o valor da primeira ferramenta encontrada que contém o número D especificado em Dno. -5: Não é possível executar a função por outras razões. |



Função

Para um número D absoluto, GETACTTD determina o número T associado. Não é feita checagem para verificar se foi atribuído mais de uma vez. Caso existam vários números D idênticos em uma unidade TO, o número T da primeira ferramenta encontrada será o retorno da função. Caso números D “planos”(flat) sejam utilizados, o uso deste comando não tem sentido, pois o valor 1 será sempre o valor de retorno (número T na base de dados).

8.7 Utilização livre de núm.D, núm.do incerto CE (A partir da SW5)



840D
NCU 572
NCU 573



840Di

8.7.4 Definir números D restantes como inválidos



Programação

DZERO



Explicação

DZERO

Marca todos os números D de uma unidade TO como inválidos



Função

O comando é utilizado para suporte durante upgrade (troca de versões).
O bloco de corretores marcados desta forma não mais serão checados através da instrução CHKNO.
Para possibilitar novamente o acesso, o número D deve ser novamente definido com SETDNO.

8.8 Cinemática do porta ferramenta



840D
NCU 572
NCU 573

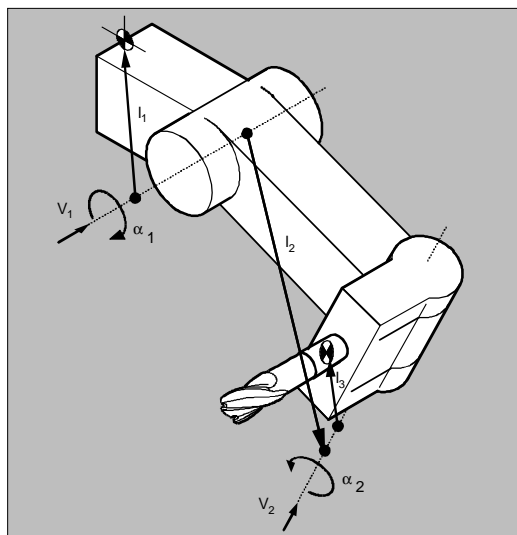


840Di

8.8 Cinemática do porta ferramenta

A cinemática do porta ferramentas com até dois eixos rotativos é programado através de 17 variáveis de sistema, de \$TC_CARR1[m] à \$TC_CARR[17]. Esta descrição consiste de:

- A distância vetorial entre o primeiro eixo rotativo e o ponto de referência l_1 , a distância vetorial entre o primeiro e o segundo eixo rotativo l_2 , a distância vetorial entre o segundo eixo rotativo e o ponto de referência da ferramenta l_3 ;
- Os vetores de referência de ambos eixos rotativos V_1, V_2 ;
- Os ângulos de rotação α_1, α_2 em torno de ambos eixos. Os ângulos de rotação são contados no sentido de visualização dos vetores dos eixos rotativos, positivos, no sentido horário de rotação.



Cinemática resolvida a partir da SW 5.3

Para máquinas com cinemática resolvida (ambas ferramenta e peça podem ser rotacionadas), as variáveis de sistema foram expandidas para incluir os parâmetros \$TC_CARR18[m] à \$TC_CARR23[m], com as funções de especificar a rotação da peça, com as funções de especificar:

- O vetor distância entre o segundo eixo rotativo v_2 e o ponto de referência da mesa giratória da ferramenta l_4 do terceiro eixo rotativo

Os eixos rotativos consistem de:

- Dois identificadores de canal para referenciar os eixos v_1 e v_2 . Estas posições são acessadas quando necessário para determinar a orientação do porta ferramentas orientável.

Os tipos possíveis de cinemática são:

- Cinemática tipo T: Somente a ferramenta pode girar.
- Cinemática tipo P : Somente a peça pode girar.
- Cinemática tipo M: Ferramenta e peça podem girar



840D
NCU 572
NCU 573



840Di



Parâmetros de sistema para porta ferramenta orientável

| | Componentes x | Componentes y | Componentes z |
|---|--|---------------------------------------|---|
| l_1 | \$TC_CARR1[m] | \$TC_CARR2[m] | \$TC_CARR3[m] |
| l_2 | \$TC_CARR4[m] | \$TC_CARR5[m] | \$TC_CARR6[m] |
| v_1 | \$TC_CARR7[m] | \$TC_CARR8[m] | \$TC_CARR9[m] |
| v_2 | \$TC_CARR10[m] | \$TC_CARR11[m] | \$TC_CARR12[m] |
| α_1 α_2 | Ângulo de rotação = \$TC_CARR13[m] ângulo de rotação = \$TC_CARR14[m] | | |
| l_3 | \$TC_CARR15[m] | \$TC_CARR16[m] | \$TC_CARR17[m] |
| Vetor deslocamento l_4 | \$TC_CARR18[m] | \$TC_CARR19[m] | \$TC_CARR20[m] |
| Eixo rotativo v_1 Eixo rotativo v_2 | \$TC_CARR21[m] \$TC_CARR22[m] | | |
| Tipo de cinemática Preset $T \Rightarrow P \Rightarrow M$ | \$TC_CARR23[m] | | |
| | Tipo de cinemática T ou \Rightarrow | Tipo de cinemática P ou \Rightarrow | Tipo de cinemática M |
| | Somente a ferramenta (T) pode ser rotacionada | Somente a Peça pode ser rotacionada | Peça e ferramenta podem ser rotacionadas de forma Miada |



Informações adicionais

O número do respectivo porta ferramentas a ser programado é especificado com "m".

Os pontos inicial/final dos vetores distância podem ser livremente selecionados para os eixos. Os ângulos de rotação em torno dos dois eixos são definidos no estado inicial do porta ferramentas com 0°. Desta forma, a descrição da cinemática para um porta ferramenta pode ser feita de forma não ambígua para qualquer possibilidades.

Caso os dois eixos rotativos sofram interseções, não é necessário especificar a distância entre os dois eixos. Porta ferramentas com somente um ou nenhum eixo rotativo podem ser descritos carregando-se o valor dos vetores direção de um eixo ou ambos com zero. Com um porta ferramentas sem eixo rotativo os vetores distância atuam como corretores de ferramenta adicionais cuja componente não possa ser afetada por alterações no plano de usinagem (G17 à G19).

8.8 Cinemática do porta ferramenta



840D
NCU 572
NCU 573



840Di

Apagando os dados do porta ferramenta

Os dados de todo o porta ferramentas são apagados através do comando $\$TC_CARR1[0] = 0$.

A partir da SW 5.3

O tipo de cinemática $\$TC_CARR23[T] = T$ deve ser pré definida com um dos têm identificadores permitidos, com caracteres maiúsculos ou minúsculos (T,P,M) e não devem ser apagados.

Alterando dados do porta ferramentas

Cada um dos valores descritos podem ser alterados através de comandos no programa principal.

A partir da SW5.3

Através da pré definição do tipo de cinemática, você tem uma das 3 opções:

1. **T**: Somente a ferramenta (**T**) pode gira
2. **P**: Somente a peça (**P**) pode ser rotacionada.
3. **M**: Peça e ferramenta (modo **Mixed**), podem ser rotacionados, dependendo da cinemática.

Qualquer outro caracter, com exceção dos 3 acima, causam um alarme quando você tentar ativar o porta ferramentas orientável.

Leitura dos dados do porta ferramentas

Cada um dos valores descritos podem ser lidos através do programa de usinagem,

Condições suplementares

Um porta ferramentas pode somente orientar a ferramenta em qualquer direção possível no espaço se

- existirem dois eixos rotativos.
- os eixos rotativos estiverem posicionados perpendicularmente um ao outro
- O eixo do comprimento da ferramenta seja perpendicular ao segundo eixo rotativo.

A partir da SW 5.3

O seguinte se aplica também à máquina onde ambos eixos devem girar a mesa, de forma que

- A orientação é perpendicular ao primeiro eixo rotativo.



840D
NCU 572
NCU 573

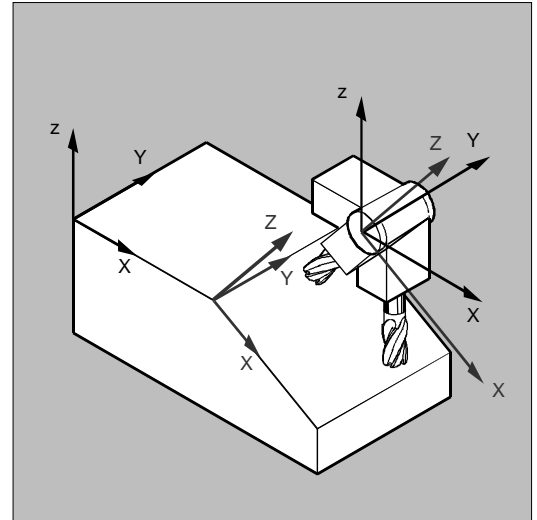


840Di



Exemplo de programação

O porta ferramenta utilizado no exemplo a seguir pode ser totalmente descrito como uma rotação através do eixo Y.



| | |
|--|---|
| N10 \$TC_CARR8[1]=1 | Definição das componentes Y do primeiro eixo rotativo para o porta ferramenta 1 |
| N20 \$TC_DP1[1,1]=120 | Definição de uma fresa de acabamento |
| N30 \$TC_DP3[1,1]=20 | Com comprimento de 20 mm |
| N40 \$TC_DP6[1,1]=5 | E raio 5 mm |
| N50 ROT Y37 | Define frame com 37 graus de rotação em torno do eixo Y |
| N60 X0 Y0 Z0 F10000 | Aproxima a posição inicial |
| N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10 | Liga a compensação de raio, a compensação de comprimento no frame de rotação, seleção do porta ferramenta 1, ferramenta 1 |
| N80 X40 | Executa a usinagem com uma rotação de 37° |
| N90 Y40 | |
| N100 X0 | |
| N110 Y0 | |
| N120 M30 | |



840Di

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Comportamento dos movimentos na trajetória

- 9.1 Controle tangencial TANG, TANGON, TANGOF..... 9-302
- 9.2 Movimento acoplado TRAILON, TRAILOF..... 9-307
- 9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV 9-311
- 9.4 Valor de acoplamento axial principal, LEADON, LEADOF..... 9-320
- 9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO 9-326
- 9.6 Programa c/memória p/pré processamento, STARTFIFO, STOPFIFO, STOPRE .9-331
- 9.7 Reposicionamento no contorno, REPOSA, REPOSL, REPOSQ, REPOSH 9-333

9.1 Controle tangencial TANG, TANGON, TANGOF



840D
NCU 572
NCU 573



810D



840Di

9.1 Controle tangencial TANG, TANGON, TANGOF



Programação

TANG (FAxisF, LAxis1, LAxis2, Coupling, CS)

TANGON (FAxis, Angle)

TANGOF (FAxis)

TLIFT (FAxis)



Explicação dos comandos

| | |
|--------|--|
| TANG | Instrução preparatória para acompanhamento tangencial |
| TANGON | Ativa o controle tangencial, especificando o eixo de acompanhamento e ângulo |
| TANGOF | Desativa o controle tangencial especificando o eixo de acompanhamento |
| TLIFT | Insere bloco intermediário nos cantos do contorno |



Explicação dos parâmetros

| | |
|----------------|---|
| FAxis | Eixo de acompanhamento eixo de acompanhamento tangencial rotativo |
| LAxis1, LAxis2 | Eixo principal: Eixo de trajetória que determina a tangente para o eixo de acompanhamento |
| Coupling | Fator de acoplamento: Relação entre o ângulo de comutação da tangente e do eixo de acompanhamento. Parâmetro opcional: default 1 |
| CSCS | Identificador do sistema de coordenadas "B" = sistema de coordenadas básico; "W" = Sistema de coordenadas da peça Parâmetro opcional; default "B" |
| Ângulo | Ângulo de deslocamento do eixo de acompanhamento |



840D
NCU 572
NCU 573



810D



840Di



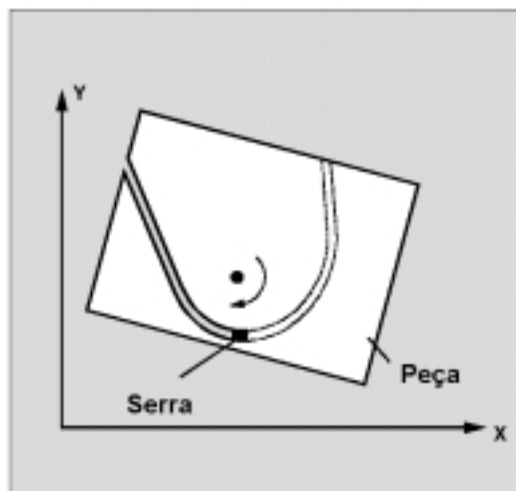
Função

Um eixo rotativo (eixo de acompanhamento, eixo seguidor) segue a trajetória programada para dois eixos principais. O eixo de acompanhamento encontra-se localizado em um deslocamento angular definido em relação à tangente da trajetória.

Aplicações

O controle tangencial pode ser utilizado em aplicações tais como :

- Posicionamento tangencial de uma ferramenta durante puncionamento
- Acompanhamento da orientação da ferramenta em cortes tipo serra
- Posicionamento de uma ferramenta de dressagem em um rebolo (vide diagrama)
- Posicionamento de ferramenta de corte para trabalhos com vidro e papel
- Avanço tangencial para solda a fio com cinco eixos
- ...



Seqüência

Definindo o eixo de acompanhamento e principal

TANG é utilizado para definir os eixos de acompanhamento e principal.

Um fator de acoplamento especifica a relação entre o ângulo de alteração na tangente o eixo de acompanhamento. Este valor é geralmente 1 (default).

O acompanhamento pode ser realizado no sistema de coordenadas básico "B" (default) ou no sistema de coordenadas da peça "W".

Exemplo:

TANG (C, X, Y, 1, "B")

Significado:

O eixo rotativo C segue os eixos geométricos X e Y.

9.1 Controle tangencial TANG, TANGON, TANGOF



840D
NCU 572
NCU 573



810D



840Di

Ativando/desativando o controle tangencial TANGON, TANGOF

O controle tangencial é chamado com TANGON, especificando o eixo de acompanhamento e o ângulo de deslocamento desejado:

TANGON (C , 90)

Significado:

O eixo C é o eixo de acompanhamento. A cada movimento dos eixos de trajetória, o eixo C será rotacionado em uma posição à 90° da tangente da trajetória.

O eixo de acompanhamento é especificado de forma a desativar o controle tangencial:

TANGOF (C)

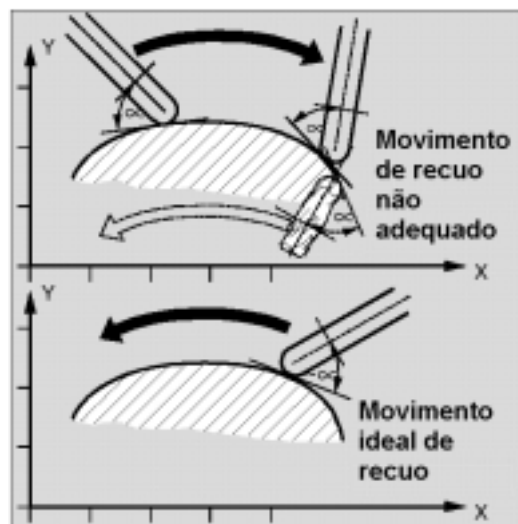
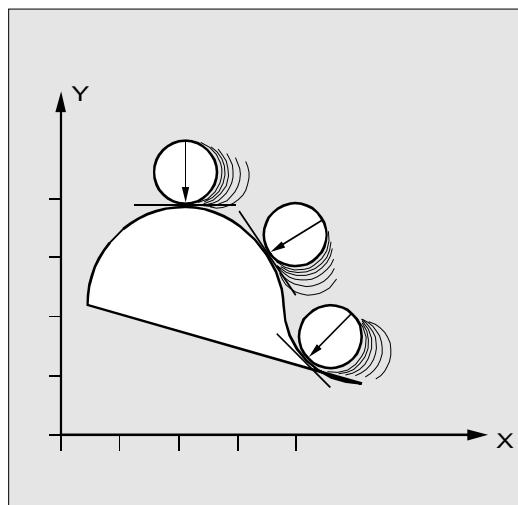
Ângulo limite através da limitação da área de trabalho

Para movimentos de trajetória que oscilam de trás para frente, a tangente salta em 180° no ponto de giro da trajetória, e o eixo de acompanhamento irá obedecer a orientação.

Este comportamento é geralmente inapropriado: o movimento de retorno pode ser executado no mesmo ângulo de deslocamento negativo, como no movimento de aproximação.

Isto pode ser alcançado através da limitação da área de trabalho do eixo de acompanhamento (G25, G26). A área de trabalho deve ser ativada no instante de reversão da trajetória (WALIMON).

Caso o ângulo de deslocamento esteja fora do limite da área de trabalho, será feita uma tentativa para que se retorne à área de trabalho permitida com o ângulo de deslocamento negativo.





840D
NCU 572
NCU 573



810D



840Di

Inserindo blocos intermediários nos cantos do contorno, TLIFT

Em um canto do contorno a tangente é alterada e com isto o valor de posição desejada do eixo de acompanhamento.

O eixo normalmente tenta compensar este degrau de alteração da forma mais rápida possível.

Entretanto, isto causa um desvio entre a posição tangencial desejada em uma certa distância após o canto do contorno. Caso este desvio não seja aceitável por razões tecnológicas, a instrução TLIFT pode ser utilizada para forçar o controle a parar no canto e executar o giro do eixo de acompanhamento para a nova direção da tangente, através de um bloco intermediário automaticamente gerado para este fim. O eixo é rotacionado com a máxima velocidade possível.

A instrução TLIFT(...) deve ser programada imediatamente após a definição do eixo com TANG(...).

Exemplo:

```
TANG ( C , X , Y ... )  
TLIFT ( C )
```

Desativando TLIFT

Para desativar TLIFT, repita a instrução TANG(...) sem TLIFT(...) em sequência.



O limite de alteração angular no qual um bloco é automaticamente inserido é definido através do dado de máquina:

```
$MA_EPS_TLIFT_TANG_STEP.
```

9.1 Controle tangencial TANG, TANGON, TANGOF

840D
NCU 572
NCU 573



810D



840Di

**Informações adicionais****Influência nas transformações**

A posição do eixo de acompanhamento rotativo pode ser um valor de entrada de uma transformação.

Posicionamento explícito do eixo de acompanhamento

Caso um eixo que esteja seguindo seus eixos principais seja posicionado de forma explícita, esta posição será adicionada ao ângulo de deslocamento.

Todas as definições de trajetória são possíveis:
Movimentos de trajetória e posicionamento.

Estado do acoplamento

Você pode consultar o estado do acoplamento através do programa NC utilizando a seguinte variável de sistema :

```
$AA_COUP_ACT[Eixo]
```

- 0 Sem acoplamento ativo
- 1,2,3 Acompanhamento tangencial ativo

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

9.2 Movimento acoplado TRAILON, TRAILOF



Programação

TRAILON(FAxis,LAxis,Coupling)

TRAILOF(FAxis,LAxis,Axis2)



Explicação dos comandos e parâmetros

| | |
|----------|---|
| TRAILON | Ativa e define os eixos acoplados; modal |
| TRAILOF | Desativa o acoplamento dos eixos |
| FAxis | Nome do eixo seguidor |
| LAxis | Nome do eixo principal |
| Coupling | Fator de acoplamento = trajetória do eixo acoplado/Trajetoória do eixo a ser seguido Default = 1 |

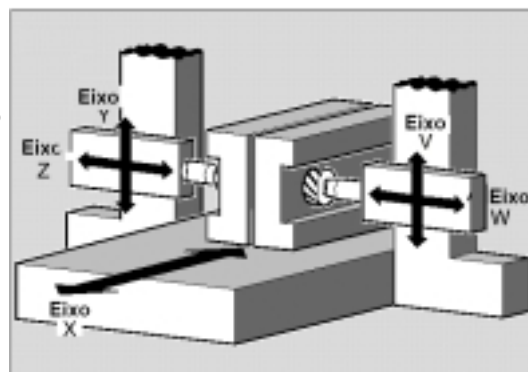


Função

Quando um eixo principal é movimentado, o eixo tracionado (=eixo seguidor) relacionado será movimentado de acordo com o movimento do eixo principal, de acordo com um fator de acoplamento. Juntos, o eixo principal e o seguidor representam os eixos acoplados.

Aplicações

- Movimento de eixo através de eixo simulado. O eixo principal pode ser um eixo simulado e o seguidor um real. O real pode então ser movimentado de acordo com o fator de acoplamento.
- Em máquinas de usinagem de dois lados com 2 pares de eixos combinados:
1º eixo principal Y, eixo seguidor V
2º eixo principal Z, eixo seguidor W



9.2 Movimento acoplado TRAILON, TRAILOF



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Seqüência

Definindo combinações de eixos acoplados, TRAILON

Os eixos acoplados são definidos e ativados simultaneamente através da instrução modal TRAILON.

TRAILON(V, Y)

V = eixo tracionado (seguidor), Y = eixo principal

A quantidade de eixos acoplados que pode ser ativada simultaneamente é restringida somente pela combinação de eixos da máquina.



Os movimentos acoplados sempre são realizados no sistema de coordenadas básico (BCS).

Tipos de eixos acoplados

Um grupo de eixos acoplados pode ser formado por quaisquer combinações entre eixos lineares e rotativos. Um eixo simulado pode ser também definido como eixo principal.

Movimento acoplado de eixos

Até dois eixos principais podem ser definidos simultaneamente a um eixo seguidor. A definição é feita através de diferentes combinações de eixos acoplados.

Um eixo seguidor pode ser programado através de quaisquer comandos de movimento disponíveis (G0, G1, G2, G3, ...etc.). Adicionalmente à trajetórias independentemente definidas, o eixo irá também se movimentar nas distâncias derivadas do eixo principal, obedecendo os fatores de acoplamento.



Um eixo seguidor pode também atuar como eixo principal de outro eixo seguidor. Várias combinações de eixos acoplados podem ser realizadas desta forma.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Fator de acoplamento

O fator de acoplamento especifica a relação desejada entre as trajetórias dos eixos acoplados e dos eixos principais.

$$\text{Fator de acoplamento} = \frac{\text{Traj.eixo seguidor}}{\text{Traj.eixo principal}}$$

Caso o fator de acoplamento não seja especificado no programa, um fator de acoplamento igual a 1 será automaticamente considerado.

O fator é carregado como fração decimal (tipo REAL). Valores negativos causam movimentos opostos entre o eixo principal e o tracionado.

Desativando eixos acoplados

As seguintes instruções desativam o acoplamento com um eixo principal:

TRAILOF(V, Y)

V = eixo tracionado, Y = eixo principal

TRAILOF com dois parâmetros desativa o acoplamento de apenas um eixo principal.

Caso um eixo seguidor esteja relacionado à dois eixos principais, p.e. V=eixo seguidor e X,Y=eixos principais, TRAILOF pode ser chamada com 3 parâmetros:

TRAILOF(V, X, Y)

Informações adicionais

Aceleração e velocidade

Os limites de aceleração e velocidade dos eixos combinados são determinados pelo “eixo mais fraco” do par.

Estado do acoplamento

Você pode consultar a seguinte variável de sistema através do programa NC:

\$AA_COUP_ACT[eixo]

| | |
|---|--------------------------|
| 0 | Sem acoplamentos ativos |
| 8 | Movimento acoplado ativo |

9.2 Movimento acoplado TRAILON, TRAILOF



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D

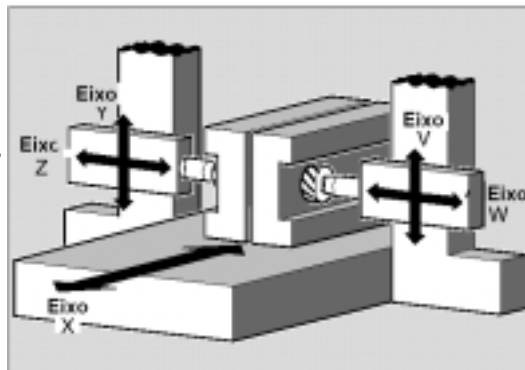


840Di



Exemplo de programação

A peça deve ser usinada em dois lados com a configuração de eixos conforme exibido no diagrama. Para isto, você pode criar 2 combinações de eixos acoplados.



...

N100 TRAILON(V,Y)

Ativa o 1º par combinado

N110 TRAILON(W,Z,-1)

Ativa o 2º par combinado, com o fator de acoplamento negativo, o eixo tracionado se movimenta em direção oposta ao eixo principal.

N120 G0 Z10

Avanço dos eixos Z e W em direções opostas

N130 G0 Y20

Avanço dos eixos Y e V na mesma direção

...

N200 G1 Y22 V25 F200

Sobreposição dependente e movimento independente do eixo tracionado "V"

...

TRAILOF(V,Y)

Desativa o 1º par de eixos acoplados

TRAILOF(W,Z)

Desativa o 2º par de eixos acoplados

9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV



Programação

As tabelas de curvas são definidas no programa de usinagem.

CTABDEF (FAxis , LAxis , n , applim)

Define o início de uma tabela de curvas

CTABEND ()

Define o fim de uma tabela de curvas

CTABDEL (n)

Apaga uma tabela de curvas

R10=CTAB (LW , n , graus , FAxis , Laxis)

Valor de acompanhamento com relação a um valor principal

R10=CTABINV (FW , aproxLW , n , graus , FAxis ,

Valor principal com relação ao de acompanhamento

Laxis)

FAxis=eixo seguidor, LAxis=eixo principal



Para maiores informações sobre os valores principais e seguidores, vide a seção “Valor de acoplamento axial principal” nesta seção.



Explicação

| | |
|---------------|---|
| FAxis | Eixo seguidor: Eixo programado através da tabela de curvas. |
| LAxis | Eixo principal: Eixo a ser programado com o valor principal. |
| n | Número da tabela de curvas |
| applim | Identificador de periodicidade da tabela: 0 Tabela não é periódica 1 Tabela é periódica |
| LW | Valor principal: Valor de posição do eixo principal para o qual o valor a ser seguido deve ser calculado. |
| degrees | Nome do parâmetro para gradiente de parâmetro |
| FW | Valor de acompanhamento: Valor de posição do eixo de seguidor para o qual o valor principal deve ser calculado. |
| aproxLW | Valor de solução aproximada para o valor principal caso não seja possível a determinação deste para um valor de acompanhamento. |
| FAxis , LAxis | Especificação opcional para o eixo seguidor e/ou principal |

9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

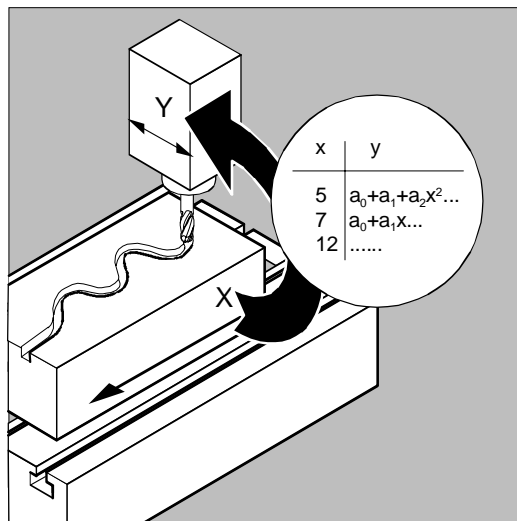


Função

Você pode utilizar a tabela de curvas para programar relações de posição e velocidades entre dois eixos.

Exemplo: troca de placas de cames mecânicos

A tabela de curvas forma a base para o valor axial principal e acoplado através da geração de relações entre os valores principais e de acompanhamento. O controle calcula um polinômio que corresponde à placa de cames a partir das posições relativas dos eixos principais e seguidores.



Informações adicionais

Para a criação de tabelas de curvas é necessário reservar espaço de memória através de dados de máquina.



Definindo uma tabela de curva

CTABDEF, CTABEND

Uma tabela de curvas representa um programa ou parte de um programa que se encontra entre as instruções CTABDEF e CTABEND.

Neste trecho de programa, o eixo seguidor terá suas posições definidas de forma inequívoca com relação ao eixo principal através de comandos de movimento deste. Estas posições serão utilizadas como posições intermediárias para o cálculo de curvas em formato de polinômios de até 3ª ordem.

Os valores iniciais para o começo da faixa de definição da tabela de curva são as primeiras posições de eixos associados especificadas (a primeira instrução com movimentos) dentro da definição da tabela de curvas. O valor final da faixa de definição da tabela curva é determinada de acordo com o último comando de movimento.



840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Informações adicionais

O seguinte não é permitido:

- Parada de pré processamento
- Correção do raio da ferramenta
- Saltos no movimento do eixo principal (p.e., na alteração de transformações)
- Instruções de movimentação apenas dos eixos seguidores
- Reversão dos eixos principais, ou seja, a posição dos eixos seguidores deve sempre ser inequívoca
- As instruções CTABDEF e CTABEND em vários níveis de programa.

Todas as instruções modais feitas dentro de uma definição de tabela de curvas não permanecerão válidas após o fim da definição da tabela. O programa no qual a tabela é definida permanecerá no mesmo estado antes e após a definição da tabela.



Parâmetros R utilizados tem seus valores recuperados.

Exemplo:

```
...
R10=5 R11=20
...
CTABDEF
G1 X=10 Y=20 F1000
R10=R11+5 ;R10=25
X=R10
CTABEND
... ;R10=5
```

Uso repetido de tabelas de curvas

A relação de funções entre os eixos principais e seguidores são calculadas através da tabela de curvas e mantidas sob um número após o encerramento do programa e mesmo o desligamento do comando.

A tabela de curvas gerada pode ser aplicada em quaisquer eixos, independente dos utilizados em sua geração.

9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Comportamento nos limites de uma tabela de curvas

Tabela de curvas **não periódica**

Caso o valor principal esteja fora da faixa de definições, valor seguidor de saída estará em seu limite inferior ou superior.

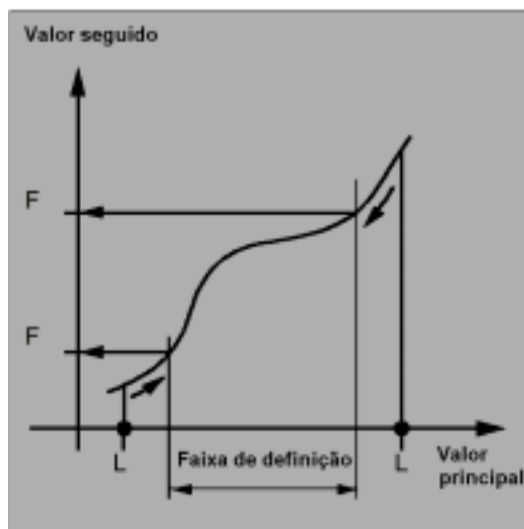
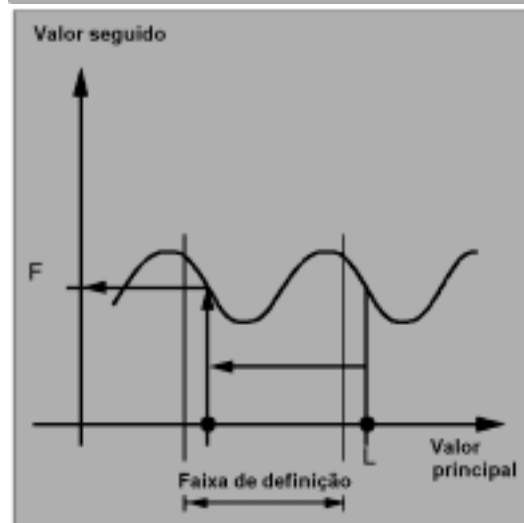


Tabela de curvas **periódica**

Caso o valor principal esteja fora da faixa de definições, este valor principal será considerado como o módulo definição e o valor de saída para o seguidor será o valor correspondente a este valor principal.



840D
NCU 571840D
NCU 572
NCU 573

810D



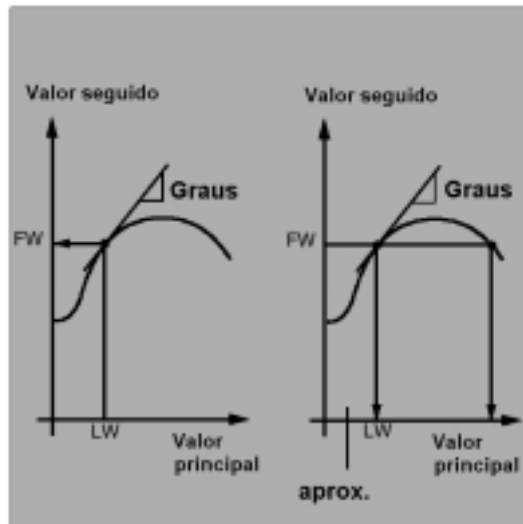
840Di

Leitura da tabela de posições, CTAB, CTABINV

Com CTAB você pode ler os valores de acompanhamento para um determinado valor principal diretamente a partir do programa de usinagem ou a partir de ações síncronas (Capítulo 10).

Com CTABINV, você pode ler o valor principal para um determinado valor de acompanhamento. Esta relação pode nem sempre ser inequívoca. CTABINV necessita entretanto de um valor aproximado (aproxLW) para o valor principal esperado. CTABINV irá retornar o valor mais próximo a este valor aproximado. O valor aproximado pode ser o valor principal do ciclo de interpolação anterior,

Ambas as funções também emitem o gradiente da tabela de funções na posição correta para o parâmetro de gradiente (graus). Desta forma, você pode calcular a velocidade do eixo principal ou seguidor na posição correspondente.



Informações adicionais

Especificações opcionais de eixos principais e seguidores para CTAB/CTABINV são importantes caso os eixos estejam configurados em diferentes unidades de comprimento.



Apagando tabelas de curvas, CTABDEL

Com CTABDEL você pode apagar tabelas de curvas. Tabelas ativas não podem ser apagadas.

Sobrescrevendo tabelas de curvas

Uma tabela de curvas é sobrescrita caso seu número seja utilizado em outra definição de tabela. Tabelas ativas não podem ser sobrescritas.



Informações adicionais

9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

Não são emitidos avisos ou alarmes quando se sobrescreve uma tabela!



Informações adicionais

Através da variável de sistema \$P_CTABDEF é possível consultar, a partir do programa, quando uma tabela de curvas encontra-se ativa.

Uma parte de um programa pode ser utilizada como definição para uma tabela de curvas e, após a retirada das instruções específicas ser utilizado novamente como programa.



Exemplo de programação

Um trecho de programa deve ser utilizado para a definição de uma tabela de curvas. O comando para parada de pré processamento STOPRE pode permanecer, e estará novamente ativo assim que o trecho do programa não mais for utilizado para a definição da tabela, com a remoção das instruções CTABDEF e CTABEND:

```
CTABDEF ( Y , X , 1 , 1 )
...
...
IF NOT ( $P_CTABDEF )
STOPRE
ENDIF
...
...
CTABEND
```



Tabela de curvas e os vários estados de operação

Durante a ativação de uma pesquisa de blocos, cálculos com tabelas de curvas não são possíveis. Caso o bloco de destino esteja dentro de uma área de definição, um alarme será emitido quando atingida CTABEND.

840D
NCU 571840D
NCU 572
NCU 573

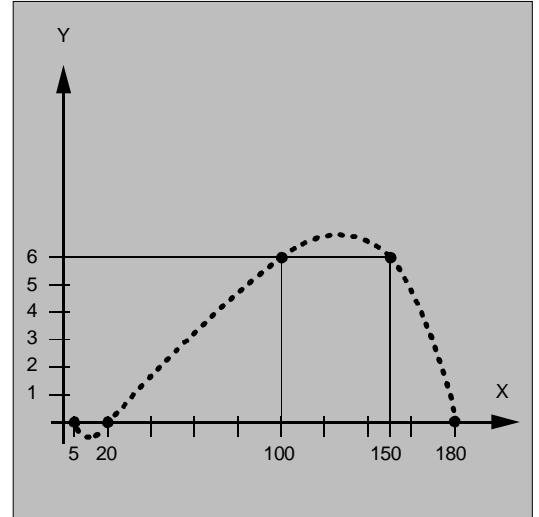
810D



840Di

**Exemplo de programação 1**

Definição de uma tabela de curvas



N100 CTABDEF(Y,X,3,0)

Início da definição de uma tabela não periódica de número 3

N110 X0 Y0

1. A instrução de posicionamento definem os valores iniciais e o primeiro ponto intermediário: valor principal = 5; valor de acompanhamento: 0

N120 X20 Y0

2. ponto intermediário: valor principal: 0...20; valor de acompanhamento: Valor inicial...0

N130 X100 Y6

3. Ponto intermediário: Valor principal: 20...100; Valor de acompanhamento: 0...6

N140 X150 Y6

4. Ponto intermediário: Valor principal: 100...150; valor de acompanhamento: 6...6

N150 X180 Y0

5. Ponto intermediário: Valor principal: 150...180; Valor de acompanhamento: 6...0

N200 CTABEND

Fim da definição;
A tabela de curvas é gerada internamente como polinômio de até 3ª ordem;

O cálculo da curva de definição depende do tipo de interpolação selecionada de forma modal (circular, linear, spline);

O estado do programa de

9.3 Tabelas de curvas, CTABDEF, CTABEND, CTAB, CTABINV840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

usinagem anterior ao início da definição
é restabelecido.

**Exemplo de programação 2**

Definição de uma curva periódica na tabela de
número 2, valor principal na faixa de 0 à 360,
movimento do eixo de acompanhamento entre 0 e
45 e novamente para 0:

N10 DEF REAL DEPPOS;

N20 DEF REAL GRADIENT;

N30 CTABDEF(Y,X,2,1)

Início da definição

N40 G1 X=0 Y=0

N50 POLY

N60 PO[X]=(45.0)

N70 PO[X]=(90.0)

PO[Y]=(45.0,135.0,-90)

N80 PO[X]=(270.0)

N90 PO[X]=(315.0) PO[Y]=(0.0,-
135.0,90)

N100 PO[X]=(360.0)

N110 CTABEND

Fim da definição

Teste da curva através do acoplamento entre Y e
X:

N120 G1 F1000 X0

N130 LEADON(Y,X,2)

N140 X360

N150 X0

N160 LEADOF(Y,X)

Leitura da tabela de função para um valor principal
de 75.0:

N170 DEPPOS=CTAB(75.0,2,GRADIENT)

Posicionamento do eixo principal e do eixo
seguidor:

N180 G0 X75 Y=DEPPOS

Após ativar o acoplamento não é necessária a
sincronização do eixo de acompanhamento:

N190 LEADON(Y,X,2)

N200 G1 X110 F1000

N210 LEADOF(Y,X)

N220 M30



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

9.4 Valor de acoplamento axial principal, LEADON, LEADOF



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

9.4 Valor de acoplamento axial principal, LEADON, LEADOF



Programação

LEADON(FAxis,LAxis,n)

LEADOF(FAxis,LAxis,n)



Explicação

| | |
|--------|---|
| LEADON | Ativa o valor de acoplamento principal |
| LEADOF | Desativa o valor de acoplamento principal |
| FAxis | Eixo de acompanhamento |
| LAxis | Eixo principal |
| n | Número da curva na tabela |



Função

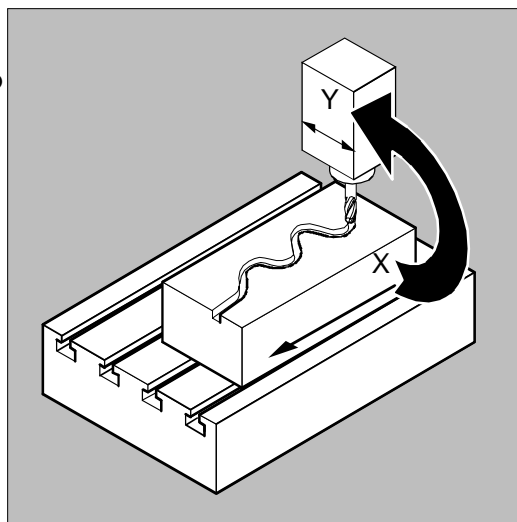
Com valor de acoplamento axial principal, um eixo principal e um seguidor se movimentam em sincronismo. É possível atribuir a posição do eixo seguidor através de uma tabela de curvas ou da resultante de um polinômio em função da posição do eixo principal – Eixo simulado, se necessário.

Eixo principal é o eixo que fornece os valores de entrada para a curva da tabela de curvas.

Eixo seguidor é o eixo que será posicionado de acordo com os cálculos da tabela de curvas.

O acoplamento ao valor principal pode ser ativado ou desativado tanto através do programa de usinagem como através de ações síncronas.

O valor principal de acoplamento sempre se aplica ao sistema de coordenadas da máquina.



Para informações a respeito da geração da tabela de curvas, vide a seção “Tabela de curvas” neste capítulo para informações sobre o valor principal de acoplamento, vide /FB/, M3, Movimento acoplado, movimentação acoplada e valor principal no acoplamento



Seqüência

840D
NCU 571840D
NCU 572
NCU 573

810D

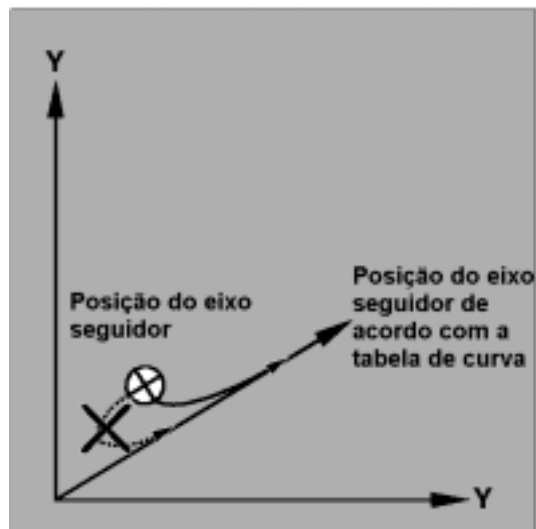


840Di

O valor de acoplamento principal deve necessitar da sincronização entre os eixos principal e seguidor. Este sincronismo pode ser alcançado somente se o eixo seguidor estiver na faixa de tolerância definida pela curva, calculada a partir da tabela de curvas quando o valor principal de acoplamento for ativado.

A faixa de tolerância para a posição do eixo seguidor é definida através do dado de máquina 37200 COUPLE_POS_TOL_COARSE.

Caso o eixo seguidor não esteja ainda na posição correta quando o valor principal de acoplamento for ativado, o sincronismo é automaticamente iniciado, pois a o valor de posição desejada para o eixo seguidor é aproximadamente a sua posição real. Durante o procedimento de sincronismo, o eixo seguidor é movimentado na direção definida pela velocidade programada (calculada a partir do fuso principal e CTAB).



Informações adicionais

Caso a posição calculada para o eixo seguidor o movimento para fora de sua posição atual quando o valor principal de acoplamento for ativado, não será possível realizar o sincronismo.



Valor atual e desejado para o acoplamento

Os seguintes valores podem ser utilizados como valor principal, isto é, os valores de posição calculados a partir do eixo seguidor:

- Valor atual de posição do eixo principal: valor atual de acoplamento
- Valores desejados para o eixo principal: Acoplamento do valor desejado

9.4 Valor de acoplamento axial principal, LEADON, LEADOF

840D
NCU 571840D
NCU 572
NCU 573

810D



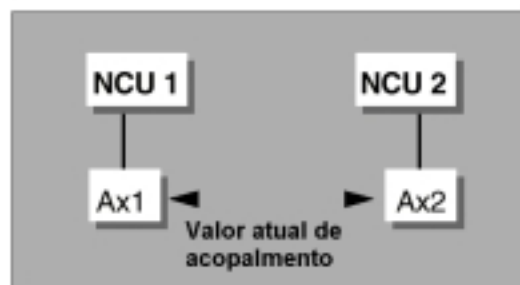
840Di



Informações adicionais

O acoplamento através do valor desejado proporciona um melhor sincronismo entre o eixo principal e seguidor, do que o obtido caso utilizado o valor atual de acoplamento, sendo portanto utilizado por default.

O acoplamento através do valor desejado é somente possível caso os eixos principal e seguidor sejam interpolados pela mesma NCU. Com eixo principal externo, o eixo seguidor somente pode ser acoplado através do valor atual.



Comutação entre os acoplamentos através do valor atual e desejado.

Uma comutação pode ser programada através do dado setting \$SA_LEAD_TYPE

Você deve sempre fazer a comutação entre o valor atual e desejado de acoplamento quando o eixo seguidor parar. Somente será possível re-sincronizar após uma comutação quando o eixo estiver parado.

Exemplo de aplicação

Você não pode ler os valores atuais sem erro durante grandes vibrações da máquina. Caso você utilize o valor principal de acoplamento, pode ser necessária a comutação entre o valor atual de acoplamento para o valor desejado, caso a tarefa seja executada com maiores vibrações.

9.4 Valor de acoplamento axial principal, LEADON, LEADOF

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Simulação do valor principal com acoplamento

através do valor desejado

Através de dados de máquina, você pode desconectar o interpolador do eixo principal do servo acionamento. Desta forma podem ser gerados valores principais sem movimento do eixo principal.

Os valores principais gerados através do acoplamento de valores desejados podem ser lidos a partir das seguintes variáveis de forma que eles possam ser utilizados, por exemplo, em ações síncronas:

- \$AA_LEAD_P
- \$AA_LEAD_V

Valor principal de posição

Valor principal de velocidade



Informações adicionais

Como opção, os valores principais podem ser gerados através de outros métodos auto programados. Os valores principais gerados desta forma são gravados nas variáveis

- \$AA_LEAD_SP
- \$AA_LEAD_SV

Valor principal de posição

Valor principal de velocidade

E podem ser lidos através delas. Antes de utilizar estas variáveis, o setting data \$SA_LEAD_TYPE = 2 deve ser carregado.

Estado do acoplamento

Você pode consultar o estado do acoplamento através do programa

NC utilizando as seguintes variáveis de sistema:

\$AA_COUP_ACT[axis]

- | | |
|----|--------------------------------------|
| 0 | não há acoplamento ativo |
| 16 | valor principal de acoplamento ativo |



Desativando o valor principal de acoplamento LEADOF

Ao desativar o valor principal de acoplamento, o eixo seguidor volta a ser um eixo normal!

Valor axial principal de acoplamento e os diferentes estados de operação

Dependendo das definições feitas através de dados de máquina, o valor principal de acoplamento é desativado com RESET.

9.4 Valor de acoplamento axial principal, LEADON, LEADOF

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di



Exemplo de programação

Em uma presa, um acoplamento mecânico comum entre o eixo principal e o eixo de um sistema de transferência, composto de eixos de transferência e eixos auxiliares devem ser trocados por um sistema de acoplamento eletrônico.

Isto demonstra como um sistema de transferência mecânico pode ser trocado por um acoplamento eletrônico. Os eventos de acoplamento e desacoplamento são implementados como **ações síncronas estáticas**.

A partir do eixo principal LW (prensa), o eixo de transferência e os eixos auxiliares são controlados como eixos seguidores, com relações definidas através de tabelas de curvas.

Eixos seguidores

| | |
|----|--|
| X | Alimentação ou eixo longitudinal |
| YL | Fechamento ou eixo lateral |
| ZL | Eixo da presa |
| U | Cilindro de alimentação, eixo auxiliar |
| V | Eixo guia, auxiliar |
| W | Engraxador, eixo auxiliar |

Gerenciamento do estado

Os eventos de comutação e acoplamento são gerenciados através de variáveis em tempo real:

\$AC_MARKER[i]=n

com:

| | |
|---|-------------------|
| i | número da memória |
| n | Valor do estado |

Ações

As ações a ocorrer incluem, por exemplo, as seguintes ações síncronas:

- Ativa acoplamento, LEADON(eixo seguidor, eixo principal, número da curva na tabela)
- Desativa acoplamento, LEADOF(eixo seguidor, eixo principal)
- Define valor atual, PRESETON(eixo, valor)
- Liga memória, \$AC_MARKER[i]= valor
- Tipo de acoplamento: real/virtual valor principal
- Posiciona eixos, POS[eixo]=valor

Condições

Entradas digitais rápidas, variáveis em tempo real \$AC_MARKER e comparações de posição são conectadas através do operador booleano AND para avaliação das condições.

Nota

No exemplo a seguir, mudanças de linha, distanciamento e caracteres em **negrito** são utilizados somente com o propósito de tornar o programa compreensivo. Para o controle, tudo que o que estiver em seguida de um número de linha constitui uma única linha.

9.4 Valor de acoplamento axial principal, LEADON, LEADOF

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

Comentário

; Define todas as **ações síncronas estáticas**

; **** Reseta memórias

```
N2      $AC_MARKER[0]=0 $AC_MARKER[1]=0
        $AC_MARKER[2]=0 $AC_MARKER[3]=0
        $AC_MARKER[4]=0 $AC_MARKER[5]=0
        $AC_MARKER[6]=0 $AC_MARKER[7]=0
```

; **** E1 0=>1 Liga acopl. da transferência

```
N10     IDS=1      EVERY ($A_IN[1]==1) AND
        ($A_IN[16]==1) AND ($AC_MARKER[0]==0)
DO      LEADON(X,LW,1) LEADON(YL,LW,2)
        LEADON(ZL,LW,3) $AC_MARKER[0]=1
```

; **** E1 0=>1 Liga acopl.cilindro de aliment.

```
N20     IDS=11     EVERY ($A_IN[1]==1) AND
        ($A_IN[5]==0) AND ($AC_MARKER[5]==0)
DO      LEADON(U,LW,4) PRESETON(U,0)
        $AC_MARKER[5]=1
```

; **** E1 0->1 Liga acopl.da cabeça guia

```
N21     IDS=12     EVERY ($A_IN[1]==1) AND
        ($A_IN[5]==0) AND ($AC_MARKER[6]==0)
DO      LEADON(V,LW,4) PRESETON(V,0)
        $AC_MARKER[6]=1
```

; **** E1 0->1 Liga acopl.engraxador

```
N22     IDS=13     EVERY ($A_IN[1]==1) AND
        ($A_IN[5]==0) AND ($AC_MARKER[7]==0)
DO      LEADON(W,LW,4) PRESETON(W,0)
        $AC_MARKER[7]=1
```

; **** E2 0=>1 Desliga acoplamento

```
N30     IDS=3      EVERY ($A_IN[2]==1)
DO      LEADOF(X,LW) LEADOF(YL,LW)
        LEADOF(ZL,LW) LEADOF(U,LW) LEADOF(V,LW)
        LEADOF(W,LW) $AC_MARKER[0]=0 $AC_MARKER[1]=0
        $AC_MARKER[3]=0 $AC_MARKER[4]=0
        $AC_MARKER[5]=0 $AC_MARKER[6]=0
        $AC_MARKER[7]=0
```

....

```
N110    G04 F01
```

```
N120    M30
```


9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO840D
NCU 571840D
NCU 572
NCU 573

840Di

9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO**Programação**

F... FNORM

F... FLIN

F... FCUB

F=FPO (... , ... , ...)

**Explicação**

| | |
|----------|---|
| FNORM | Definição básica. O valor de avanço é especificado em função do bloco de movimento de trajetória e é válido de forma modal |
| FLIN | Perfil de velocidade linear na trajetória: O valor de avanço é atingido de forma linear durante a trajetória, partindo do valor atual do início do bloco, sendo atingido até o fim do bloco e a partir de então válido de forma modal |
| FCUB | Perfil de velocidade cúbica na trajetória: Os valores não modais F programados são conectados através de um spline com referência ao fim do bloco. O spline inicia e é encerrado tangencialmente com os valores especificados de velocidade prévio e com o próximo. Caso não exista o endereço F no bloco, o último valor f programado será utilizado. |
| F=FPO... | Perfil de velocidade polinomial na trajetória: O endereço F define a característica de velocidade através de um polinômio a partir do valor atual até o término do bloco O valor final será válido a partir deste instante como modal. |

**Função**

Para permitir uma definição flexível na característica de avanço, o avanço programado de acordo com a DIN66205 foi ampliado em suas características lineares e cúbicas. As características cúbicas podem ser programadas tanto diretamente ou como splines de interpolação.

Estas características adicionais tornam possível a programação de velocidade contínuas e suaves, dependendo da curvatura da peça a ser usinada.

**Seqüência**

9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO



840D
NCU 571



840D
NCU 572
NCU 573

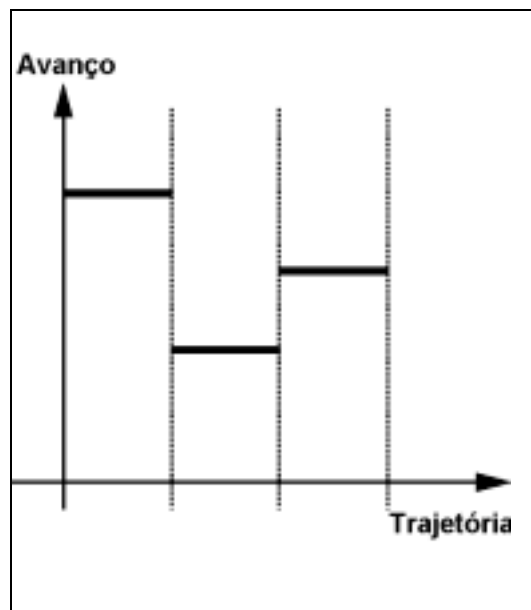


840Di

FNORM

Os endereços de avanço F definem o avanço de trajetória como valores constantes conforme a norma DIN 66025.

Favor consultar o Manual de programação “Fundamentos” para informações mais detalhadas a este respeito.



FLIN

O avanço programado será atingido até o fim do bloco de forma linear, partindo do valor de avanço atual.

Exemplo:

```
N30 F1400 FLIN X50
```



9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO



840D
NCU 571



840D
NCU 572
NCU 573



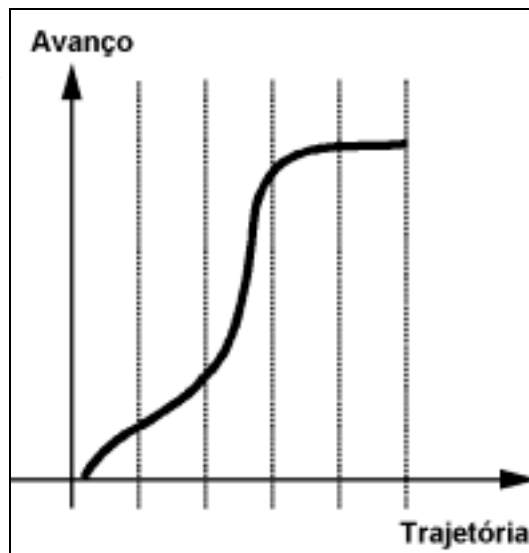
840Di

FCUB

O avanço programado será atingido de acordo com uma curva característica cúbica até o fim do bloco, partindo do valor de avanço atual. O controle utiliza splines para conectar todos os valores programados de forma não modal que possuem FCUB ativo. Os valores de avanço atuam aqui como pontos de interpolação para o cálculo da interpolação spline.

Exemplo:

```
N50 F1400 FCUB X50
N60 F2000 X47
N70 F3800 X52
...
```



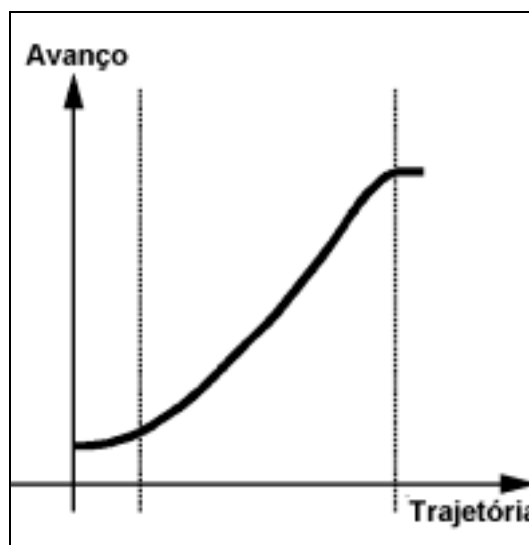
F=FPO(...,...,...)

O avanço é programado sob a forma de um polinômio. Os coeficientes do polinômio são especificados de acordo com o mesmo método utilizado para a interpolação polinomial.

Exemplo:

```
F=FPO(endfeed, quadf, cubf)
```

endfeed, quadf e cubf são variáveis previamente definidas.



| | |
|----------|------------------------------------|
| endfeed: | Avanço no fim do bloco |
| quadf: | Coefficiente quadrado do polinômio |
| cubf: | Coefficiente cúbico do polinômio |

Com FCUB ativo, o spline é conectado tangencialmente à característica definida através de FPO no início e fim do bloco

Condições suplementares

As funções para programação das características de movimento na trajetória são aplicadas sem levar em consideração as características de avanço programadas.

840D
NCU 571840D
NCU 572
NCU 573

840Di

As características de avanço programadas são sempre absolutas, independentes de G90 ou G91.



Informações adicionais

Compressor

Com um compressor ativo COMPON o seguinte se aplica quando vários blocos são conectados para formar um segmento spline:

FNORM:

A palavra F só último bloco no grupo será aplicada ao segmento spline.

FLIN:

A palavra F do último bloco no grupo será aplicada ao segmento spline.

O valor programado F será aplicado até o fim do bloco, sendo atingido de forma linear.

FCUB:

O avanço gerado poderá desviar do programado nos pontos finais em um valor que não deverá exceder os valores definidos nos dados de máquina

\$MC_COMPRESS_VELO_TOL.

F=FPO(...,...,...)

Estes blocos não serão comprimidos.

Otimização de avanço em trechos de trajetória curvos

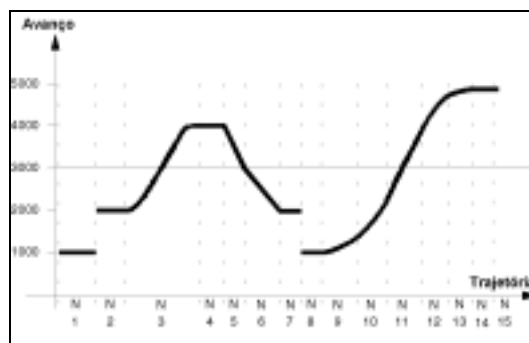
O avanço polinomial F-FPO e o avanço spline FCUB podem sempre serem percorridos com um fator de corte constante CFC, tornando com isto possível se atingir uma característica de velocidade livre de solavancos. Isto habilita a criação de um padrão de aceleração constante para que seja atingido o avanço desejado.

9.5 Característica de avanço, FNORM, FLIN, FCUB, FPO840D
NCU 571840D
NCU 572
NCU 573

840Di

**Exemplo de programação**

Este exemplo mostra a programação e representação gráfica de vários perfis de velocidade.



| | |
|------------------------------|---|
| N1 F1000 FNORM G1 X8 G91 G64 | Perfil de velocidade constante, dimensionamento incremental |
| N2 F2000 X7 | Alteração na velocidade desejada |
| N3 F=FPO(4000, 6000, -4000) | Perfil de avanço via polinômio com avanço de 4000 no fim do bloco |
| N4 X6 | O avanço de 4000 do polinômio é utilizado de forma modal |
| N5 F3000 FLIN X5 | Perfil linear de avanço |
| N6 F2000 X8 | Perfil linear de avanço |
| N7 X5 | Avanço linear aplicado como valor modal |
| N8 F1000 FNORM X5 | Perfil constante de velocidade com alteração abrupta de aceleração |
| N9 F1400 FCUB X8 | Todos os avanços subsequentes, programados de forma não modal são conectados através de splines |
| N10 F2200 X6 | |
| N11 F3900 X7 | |
| N12 F4600 X7 | |
| N13 F4900 X5 | Desativa perfil spline |
| N14 FNORM X5 | |
| N15 X20 | |

9.6 Programa c/memória p/pré proc..STARTFIFO,STOPFIFO, STOPRE

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

9.6 Programa c/memória p/pré processamento, STARTFIFO, STOPFIFO, STOPRE



Explicação dos comandos

| | |
|-----------|---|
| STOPFIFO | Parada de processamento de trecho em alta velocidade, preenche memória de pré processamento, memória de pré processamento até STARTFIFO, detecção de “memória de pré processamento cheia” ou “fim de programa” detectado. |
| STARTFIFO | Inicia o pré processamento em alta velocidade , preenche em paralelo a memória de pré processamento |
| STOPRE | Parada de pré processamento |



Função

Dependendo do nível de expansão, o controle possui uma certa quantidade de memória chamada de memória de pré processamento, na qual os blocos são preparados e gravados antes de sua execução, e então são processados como seqüências de blocos de alta velocidade durante o processo de usinagem.

Estas seqüências permitem que trajetórias curtas sejam percorridas em altas velocidades.

Desde que exista um tempo de controle residual disponível, a memória de pré processamento é sempre preenchida. A função STARTFIFO interrompe o processo de usinagem até que a memória de pré processamento esteja cheia ou até que STOPFIFO ou STOPRE sejam detectados.



Seqüência

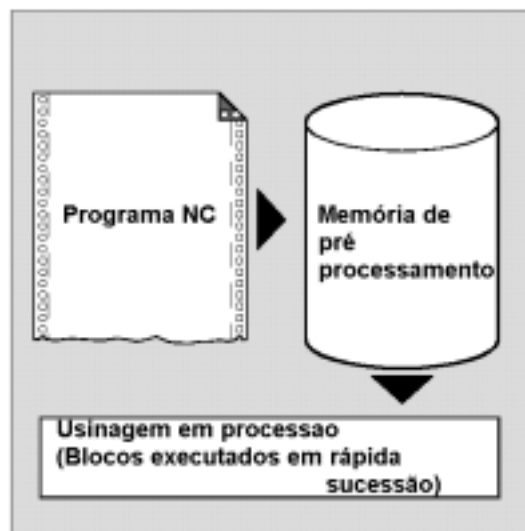
Trecho de processamento marcado

O trecho de processamento em alta velocidade a ser memorizado na memória de pré processamento é marcado em seu início e fim com STARTFIFO e STOPFIFO respectivamente.

Exemplo:

```
N10 STOPFIFO
N20...
N100
N110 STARTFIFO
```

A execução destes blocos não será iniciada até que a memória de pré processamento esteja cheia ou o comando STARTFIFO seja detectado.



9.6 Programa c/memória p/pré proc., STARTFIFO, STOPFIFO, STOPRE840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Restrições

A memória de pré processamento não é preenchida ou o processo de preenchimento for interrompido caso o trecho de processamento contenha comandos que necessitem de operações que não podem ser buferizadas (referenciamento, funções de medição, ...).

Parada de pré processamento

Quando **STOPRE** for programado, o bloco seguinte não será processado até que os blocos previamente preparados tenham sido completamente executados. O bloco anterior sofrerá uma parada conforme a função de parada exata (Como em G9).

Exemplo:

```
N10 ...
N30 MEAW=1 G1 F1000 X100 Y100 Z50
N40 STOPRE
```

O controle inicia uma parada interna de pré processamento durante o acesso a dados de estado da máquina (\$A...).

Exemplo:

```
R10 = $AA_IM[X] ;Leitura do valor atual do eixo X
```

**Nota**

Quando um corretor de ferramenta ou uma interpolação spline estão ativos, você não deve programar a instrução STOPRE, pois esta pode provocar a interrupção de uma seqüência de blocos que deveria ser contínua.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

9.7 Reposicionamento no contorno, REPOSA, REPOS, REPOSQ, REPOSH



Programação

REPOSA RMI DISPR=... OU REPOSA RMB OU REPOSA RME

REPOS, RMI DISPR=... OU REPOS, RMB OU REPOS, RME

REPOSQ RMI DISPR=... DISR=... OU REPOSQ RMB DISR=... OU REPOSQ RME DISR=... OU
REPOSQA DISR=...

REPOSH RMI DISPR=... DISR=... OU REPOSH RMB DISR=... OU REPOSH RME DISR=... OU
REPOSHA DISR=...



Explicação dos comandos

Trajetória de aproximação

| | |
|------------------|--|
| REPOSA | Reposicionamento linear de todos os eixos |
| REPOS, | Reposicionamento linear |
| REPOSQ DISR=... | Reaproximar-se do contorno em um quarto de círculo de raio DISR |
| REPOSQA DISR=... | Reaproximar-se do contorno linearmente com todos os eixos; eixos geométricos em quarto de círculo de raio DISR |
| REPOSH DISR=... | Reaproximar-se do contorno em semicírculo de raio DISR |
| REPOSHA DISR=... | Reaproximar-se do contorno com todos os eixos; eixos geométricos em semicírculo de raio DISR |

Ponto de reposicionamento

| | |
|---------------|---|
| RMI | Reposicionamento no ponto de interrupção |
| RMI DISPR=... | Reposicionamento a uma distância DISPR em frente ao ponto de interrupção |
| RMB | Reposicionamento no ponto inicial de bloco |
| RME DISPR=... | Reposicionamento a uma distância DISPR em frente ao ponto final de bloco |
| A0 B0 C0 | Eixos com os quais será realizada a reaproximação |

9.7 Reposicionamento no contorno, REPOSA, REPOSL, REPOSO, REPOSH840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D

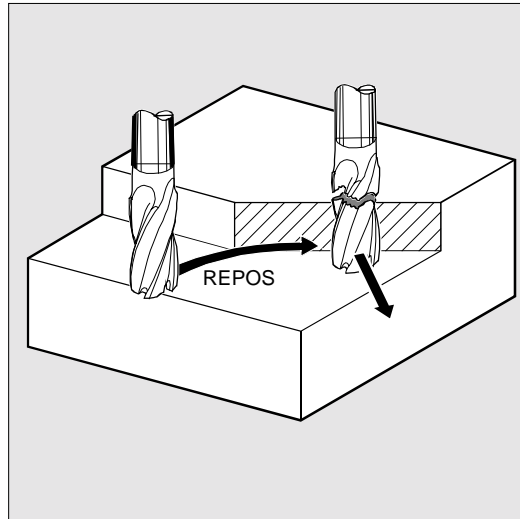


840Di

**Função**

Caso você interrompa a execução de um programa e recue a ferramenta da operação de usinagem por que, por exemplo, a ferramenta quebrou ou você deseja checar alguma medida, você poderá reposicionar em qualquer ponto no contorno sob o controle do programa.

O comando REPOS atua da mesma forma que um salto de retorno de um subprograma (p.e. através de M17). Blocos programados após o comando na rotina de interrupção não serão executados.



Para informações sobre a interrupção na execução de programas, vide também a seção “Rotinas de interrupção” neste manual.

**Sequência****Definição do ponto de reposicionamento**

Com referência ao bloco de programa onde a execução do programa foi interrompida, é possível selecionar um dos três diferentes pontos de reposicionamento:

- RMI, ponto de interrupção
- RMB, início do bloco ou último ponto final
- RME, ponto final do bloco

RMI DISPR=...<F 6 ou RME DISPR=... permite a você selecionar um ponto de reposicionamento o qual se situa antes do ponto de interrupção ou no ponto final do bloco.

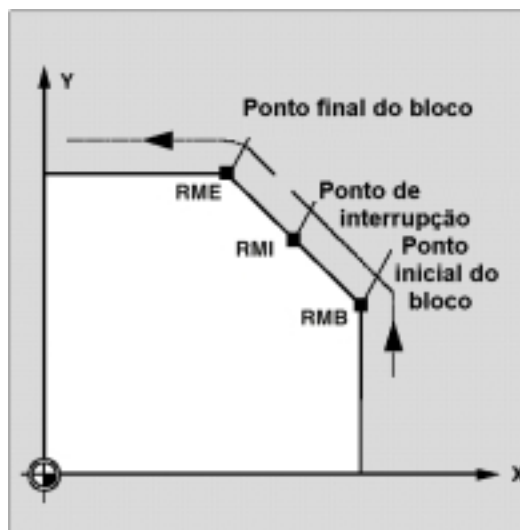
DISPR=... permite a você especificar uma distância em mm/inch entre o ponto de interrupção e o ponto de reposicionamento ou antes do ponto final. Mesmo com valores grandes, este ponto não pode estar localizado antes da posição inicial do bloco.

Caso o comando DISPR=... não tenha sido programado, então DISPR=0 será aplicado no ponto de interrupção (com RMI) ou no ponto final do bloco (com RME).

SW 5.2 em diante:

O sinal antes de DISPR será avaliado.

Em caso de sinal positivo, o comportamento será o anteriormente



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

descrito.

Em caso de sinal negativo, a aproximação será realizada por trás do ponto de interrupção, ou, com RMB, por trás do ponto inicial.

A distância entre o ponto de interrupção e o ponto de aproximação depende do valor de DISPR. Mesmo para valores grandes, o ponto pode, no máximo, atingir o ponto final do bloco.

Exemplo de aplicação:

Um sensor irá reconhecer a aproximação de uma pinça. Uma ASUP é iniciada para bypassar a pinça. Mais tarde, um DISPR negativo será reposicionado em um ponto por trás da pinça e o programa será continuado.

Aproximação com nova ferramenta

O seguinte acontece caso você tenha parado o programa devido à uma quebra de ferramenta: Quando o novo número D for programado, o programa de usinagem irá continuar com um corretor de ferramenta com valores alterados no ponto de reposicionamento.



Quando alterados os valores do corretor de ferramenta, não será possível reaproximar no ponto de interrupção. Nestes casos, o ponto mais próximo ao ponto de interrupção no novo contorno será aproximado (possivelmente alterado por DISPR).

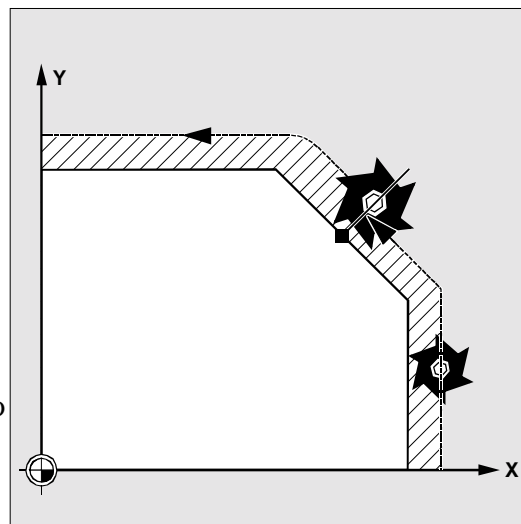
Aproximando-se do contorno

O movimento através do qual a ferramenta é reposicionada no contorno pode ser programado. Programe zero no endereço dos eixos a serem movimentados.

Os comandos REPOSA, REPOSL e REPOSHA automaticamente reposicionam todos os eixos. Nomes individuais de eixos não necessitam ser especificados.

Quando os comandos REPOSL, REPOSQ e REPOSH forem programados, todos os eixos geométricos são posicionados automaticamente, ou seja, eles não precisam ser especificados no comando.

Todos os outros eixos a serem reposicionados devem ser



9.7 Reposicionamento no contorno, REPOSA, REPOSL, REPOSQ, REPOSQA840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

especificados nos comandos.

Aproximação em linha reta, REPOSA, REPOSL

A ferramenta é levada ao ponto de reposicionamento em linha reta.

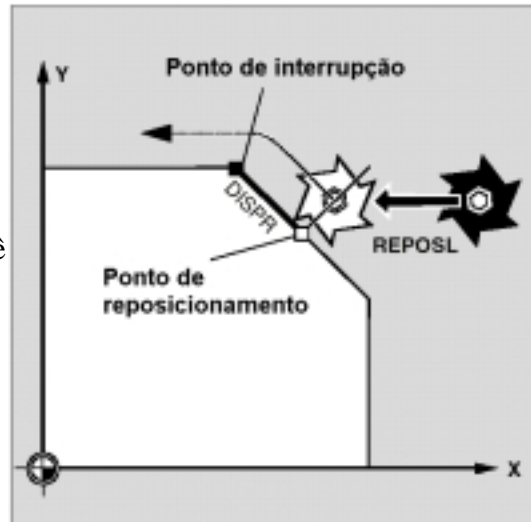
Todos os eixos são automaticamente posicionados através do comando REPOSA. Com REPOSL você pode especificar quais eixos devem ser movimentados.

Exemplo:

```
REPOSL RMI DISPR=6 F400
```

Ou

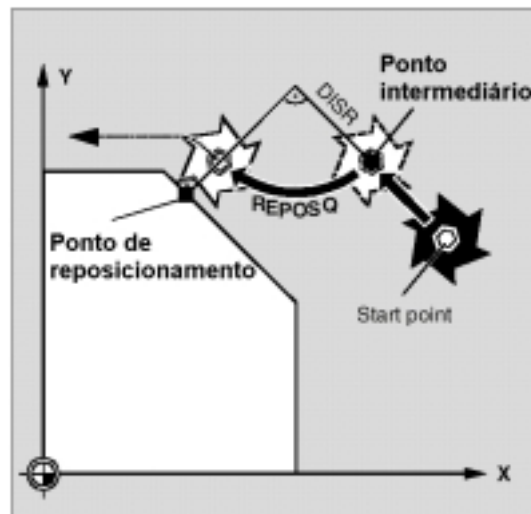
```
REPOSA RMI DISPR=6 F400
```

**Aproximação em ¼ de circunferência, REPOSQ, REPOSQA**

A ferramenta será reposicionada no ponto através de ¼ de circunferência de raio $DISR=...$. O controle calcula automaticamente o ponto intermediário entre o início e os pontos de reposicionamento.

Exemplo:

```
REPOSQ RMI DISR=10 F400
```



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



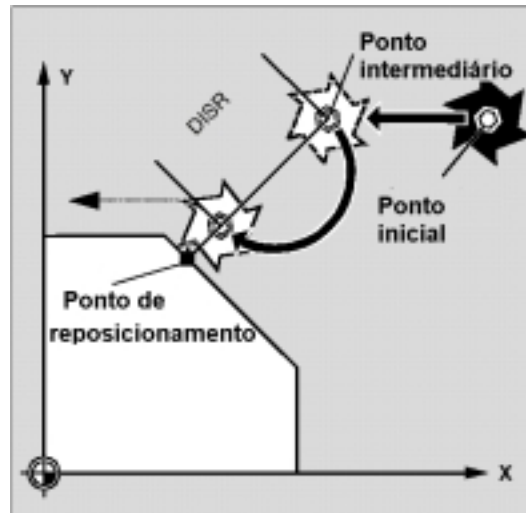
840Di

Aproximação em semicírculo, REPOSH, REPOSHA

A ferramenta se aproxima do ponto de reposicionamento através de um semicírculo de diâmetro $DISR=...$. O controle calcula automaticamente o ponto intermediário entre o ponto inicial e os pontos de reposicionamento.

Exemplo:

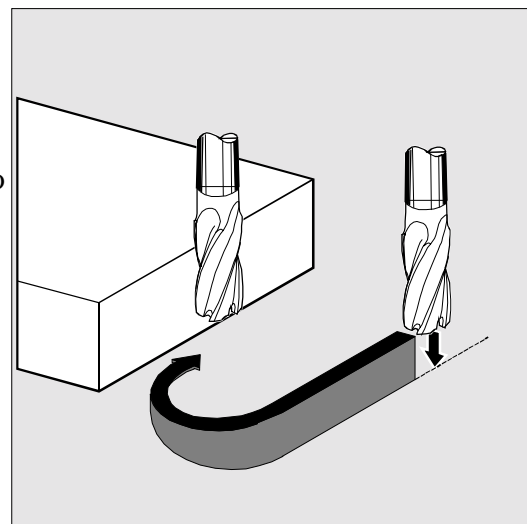
REPOSH RMI DISR=20 F400



O seguinte procedimento é aplicado em movimentos circulares REPOSH e REPOSQ:

O círculo será percorrido nos planos de trabalho específicos, de G17 a G19.

Caso seja especificado um terceiro eixo geométrico (no sentido de avanço) no bloco de aproximação, o ponto de reposicionamento será aproximado em hélice, caso a posição da ferramenta e a posição programada na direção de avanço não estejam coincidindo.



Nos seguintes casos, o controle automaticamente comuta de para aproximação linear REPOSL:

- Você não especificou um valor para DISR.
- Não existe sentido de aproximação definido (programa interrompido em um bloco sem informação de trajetória).
- Com uma direção de aproximação perpendicular ao plano de trabalho atual.

Movimentos com ações síncronas

| | | |
|---------|---|--------|
| 10.1 | Estrutura, informações básicas | 10-339 |
| 10.1.1 | Programação e elementos de comando | 10-341 |
| 10.1.2 | Faixa válida: Número de identificação ID | 10-342 |
| 10.1.3 | Instruções | 10-343 |
| 10.1.4 | Ações | 10-346 |
| 10.1.5 | Relação das ações síncronas | 10-348 |
| 10.2 | Módulos básicos para condições e ações | 10-350 |
| 10.3 | Variáveis em tempo real especiais para ações síncronas | 10-353 |
| 10.3.1 | Memórias/contadores \$AC_MARKER[n] | 10-353 |
| 10.3.2 | Variável tipo temporizador \$AC_TIMER[n], a partir da SW 4 | 10-353 |
| 10.3.3 | Parâmetros em ações síncronas \$AC_PARAM[n] | 10-354 |
| 10.3.4 | Acesso a parâmetros R \$Rxx | 10-355 |
| 10.3.5 | Leitura/escrita em dados de máquina e dados setting , a partir da SW4 | 10-356 |
| 10.3.6 | Variáveis FIFO \$AC_FIFO1[n] ... \$AC_FIFO10[n], SW 4 em diante | 10-357 |
| 10.4 | Comandos nas ações síncronas | 10-359 |
| 10.4.1 | Funções auxiliares | 10-359 |
| 10.4.2 | Bloqueio da execução do programa RDISABLE | 10-360 |
| 10.4.3 | Cancelamento da parada no pré processamento STOPREOF | 10-361 |
| 10.4.4 | Cancelamento da distância a percorrer | 10-362 |
| 10.4.5 | Cancela distância a percorrer c/preparação, DELDTG, DELTG (eixo1,...) | 10-362 |
| 10.4.7 | Definição de polinômio, FCTDEF, sincronizada com o bloco | 10-364 |
| 10.4.8 | Controle de potência laser | 10-366 |
| 10.4.9 | Função avaliação SYNFACT | 10-367 |
| 10.4.10 | Controle adaptável (aditivo) | 10-368 |
| 10.4.11 | Controle adaptável (multiplicativo) | 10-369 |
| 10.4.12 | Controle de tolerância com compensação limitada | 10-370 |
| 10.4.13 | Corretores de ferramenta online FTOC | 10-372 |
| 10.4.14 | Movimentos de posicionamento | 10-374 |
| 10.4.15 | Posicionando eixo POS | 10-376 |
| 10.4.16 | Inicia/para movimento MOV | 10-376 |
| 10.4.17 | Avanço axial: FA | 10-377 |
| 10.4.18 | Limite de software | 10-377 |
| 10.4.19 | Coordenação de eixos | 10-378 |
| 10.4.20 | Carregando o valor atual | 10-379 |
| 10.4.21 | Movimentos com o fuso | 10-380 |
| 10.4.22 | Movimento de eixos acoplados: TRAILON, TRAILOF | 10-381 |
| 10.4.23 | Valor principal de acoplamento LEADON, LEADOF | 10-382 |
| 10.4.24 | Medição | 10-384 |

| | | |
|---------|---|--------|
| 10.4.25 | Marcas de espera (WAIT) ligar/cancelar: SETM, CLEARM..... | 10-384 |
| 10.4.26 | Resposta a erros | 10-385 |
| 10.5 | Ciclos tecnológicos | 10-386 |
| 10.5.1 | Travar, destravar, resetar: LOCK, UNLOCK, RESET | 10-388 |
| 10.6 | Cancelamento de uma ação síncrona: CANCEL | 10-390 |
| 10.7 | Condições adicionais | 10-391 |

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

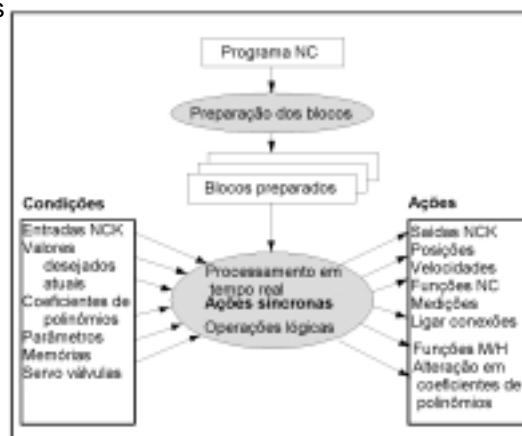
10.1 Estrutura, informações básicas



Função

As ações síncronas permitem que você inicie diferentes ações a partir de um programa e as execute de forma síncrona.

Os pontos de partida destas ações podem ser definidos através de condições e estas serem avaliadas em tempo real (no ciclo de interpolação). Estas ações são portanto a resposta a eventos em tempo real, e sua execução não é limitada ao bloco. Uma ação síncrona também contém informações sobre a efetividade da ação e sobre a frequência com a qual variáveis em tempo real são amostradas e portanto sobre a frequência com a qual as ações são iniciadas. Desta forma, uma ação pode ser disparada somente uma vez, ou de forma cíclica no ciclo de interpolação.



Programação

DO Ação1 Ação2 ...

VOCABULARY_WORD condição DO ação1 ação2 ...

ID=n VOCABULARY_WORD condição DO ação1 ação2 ...

IDS=n VOCABULARY_WORD condição DO ação1 ação2 ...



Explicação

Números de identificação

ID/IDS

ID=n

Ações síncronas **modais** no modo automático, **local para programar**; n = 1... 255

IDS=n

Ações síncronas **modais** em quaisquer modos, **Estáticas**; n = 1... 255

Sem ID/IDS

Ações síncronas **não modais** em modo automático

Instruções

Sem instruções

A execução da ação não está vinculada a quaisquer condições. A ação é executada ciclicamente no ciclo de interpolação.

10.1 Estrutura, informações básicas



840 D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

| | |
|--|--|
| WHEN | A condição é testada até que seja satisfeita uma vez, a ação associada é executada uma vez. |
| WHENEVER | A condição é testada ciclicamente. A ação associada é executada ciclicamente enquanto a condição for satisfeita. |
| FROM | Após a condição Ter sido uma vez satisfeita, a ação é executada ciclicamente enquanto a ação estiver ativa |
| EVERY | A ação é iniciada uma vez quando a condição for satisfeita, e executada novamente quando a condição for alterada do estado falso para o verdadeiro (FALSE para TRUE). A condição é testada ciclicamente. Cada vez que a condição for satisfeita, a ação associada é executada. |
| Condição | Lógica para avaliação de variáveis em tempo real, as condições são checadas a cada ciclo de interpolação. A partir da SW 5 , códigos G podem ser programados em ações síncronas para checagem de condições. |
| DO | Dispara a ação caso a condição seja satisfeita. |
| Ação | Ação iniciada caso a condição seja satisfeita, p.e. escrita de variáveis, ativação de acoplamento de eixos, ligação de saídas NCK, execução de funções M e H, A partir da SW5 , os códigos G podem ser programados em ações síncronas para ações/ciclos tecnológicos. |
| Coordenação de ações síncronas/ciclos tecnológicos | |
| CANCEL[n] | Cancela a ação síncrona |
| LOCK[n] | Inibe o ciclo tecnológico |
| UNLOCK[n] | Habilita o ciclo tecnológico |
| RESET | Reseta o ciclo tecnológico |



Exemplo de programação

```
WHEN $AA_IW[Q1]>5 DO M172 H510 ;caso o valor de posição atual do eixo Q1 ultrapassar a 5mm,
                                as funções auxiliares M172 e H510 são executadas e
                                enviadas à interface do PLC.
```



Caso ocorram variáveis em tempo real em um programa NC (p.e. valores atuais, posição de uma entrada ou saída digital, etc.), o pré processamento será interrompido até que o bloco anterior tenha sido executado e os valores das variáveis em tempo real obtidas. As variáveis em tempo real são amostradas a cada ciclo de interpolação.

Vantagens com as ações síncronas:

O pré processamento não é interrompido

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

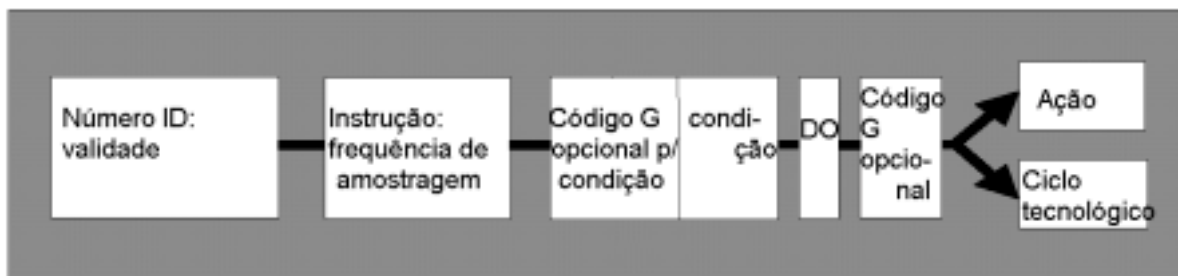
Aplicações possíveis:

- Otimização do tempo de execução de aplicações críticas (p.e. troca de ferramentas)
- Resposta rápida à eventos externos
- Programação de controles AC
- Funções de segurança
-

10.1.1 Programação e elementos de comando**Função**

Uma ação síncrona é programada em um bloco separado e dispara uma função de máquina no próximo bloco executável (p.e. movimentos em G0, G1, G2, G3; bloco com funções auxiliares).

As ações síncronas consistem em até 4 elementos de comando, cada um com uma tarefa distinta:

**Exemplo:**

| | | | | |
|------|----------|--------------|----|--------------|
| ID=1 | WHENEVER | \$A_IN[1]==1 | DO | \$A_OUT[1]=1 |
|------|----------|--------------|----|--------------|

Ação síncrona nr.1: Sempre que A entrada 1 estiver ligada então Ligue a saída 1

10.1 Estrutura, informações básicas



840 D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

NCU 573

10.1.2 Faixa válida: Número de identificação ID



Função

A área de validade de uma ação síncrona é definida pelo seu número de identificação (ID modal):

- **Sem ID modal**

A ação síncrona é ativa somente no modo automático. É válida somente no próximo bloco executável (Bloco com instrução de movimento ou outra ação na máquina), sendo portanto, **não modal**

Exemplo:

```
WHEN $A_IN[3]==TRUE DO $A_OUTA[4]=10
```

```
G1 X20
```

```
;Bloco executável
```

- **ID=n; n=1...255**

A ação síncrona é aplicada de forma modal nos blocos seguintes e é desativada através do comando CANCEL(n), ou pela programação do mesmo ID com uma nova ação síncrona.

As ações síncronas aplicadas no bloco M30 são também ativadas (caso necessário desativar com o comando CANCEL).

As ações síncronas com ID são aplicadas **somente** no **modo automático**.

Exemplo:

```
ID=2 EVERY $A_IN[1]==1 DO POS[X]=0
```

- **IDS=n; n=1...255**

Estas ações síncronas **estáticas** são aplicadas de forma **modal** em **todos os modos de operação**.

Elas podem ser definidas não somente para o ponto inicial de um programa mas também diretamente após a ligação do comando através da partida de um subprograma assíncrono

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

(ASUP) executada pelo PLC. Desta maneira, as ações podem ser executadas independente do modo de operação selecionado no NC.

Exemplo:

```
IDS=1 EVERY $A_IN[1]==1 DO POS[X]=100
```



Aplicação:

- Loops AC em modo JOG
- Operações lógicas para o Safety Integrated
- Funções de monitoração, respostas a estados da máquina em todos os modos

Sequence of execution

As ações síncronas aplicadas de forma modal ou estática são executadas na sequência de seus números ID(S) (dentro do ciclo de interpolação). Ações síncronas não modais (sem número ID) são executadas na sequência programada após a execução das ações modais.

10.1.3 Instruções



Função

As instruções determinam a frequência com que a condição seguinte será amostrada e a ação associada executada.

- Sem instrução:
Caso não seja programada nenhuma instrução, a condição é considerada sempre satisfeita. O comando síncrono é executado ciclicamente.
- WHEN
A condição é amostrada a cada ciclo de interpolação até que seja uma vez satisfeita, quando então a ação será uma vez executada.
- WHENEVER
A condição é amostrada a cada ciclo de interpolação. A

Example:

```
DO $A_OUTA[1]=$AA_IN[X]  
; Output of actual value on  
analog  
output
```

10.1 Estrutura, informações básicas



840 D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

NCU 573

ação é executada a cada ciclo, enquanto a condição for satisfeita.

- **FROM**

A condição é testada a cada ciclo de interpolação até que seja uma vez satisfeita. A partir de então a ação é executada enquanto estiver ativa, mesmo que a condição não mais seja satisfeita.

- **EVERY**

A condição é amostrada a cada ciclo de interpolação. A ação é executada sempre que a condição seja satisfeita.

Controle por borda de pulso:

A ação é iniciada novamente quando a condição for comutada de FALSE para TRUE.

Example:

```
ID=1 EVERY $AA_IM[B]>75 DO
POS[U]=IC(10) FA[U]=900;
```

When the actual value of axis B overshoots the value 75 in machine coordinates, the U axis should move forwards by 10 with an axial feed.

Condição

Define quando uma ação deve ser executada através da comparação de duas variáveis em tempo real, ou uma variável em tempo real mais uma expressão calculada durante o pré processamento.

SW 4 em diante:

Os resultados das comparações podem ser também associados através de operadores booleanos entre parênteses ().

A condição é testada no ciclo de interpolação. Caso satisfeita, a ação associada é executada.

SW 5 em diante:

As condições podem ser especificadas com um código G. Isto significa que é possível definir valores para a avaliação de condições e a ação/ciclo tecnológico independentes do estado do programa atual. É necessário separar as ações síncronas do ambiente de programação, pois as ações síncronas irão executar seus comandos a cada vez que as condições de disparo forem satisfeitas.

Aplicações:

Definição do sistema de medição para avaliação de condições através dos códigos G70, G71, F700 e G710.

A partir da SW5 o uso destes códigos G é permitido.

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

Um código G utilizado para a condição é aplicado tanto na avaliação da condição como na execução da ação, caso não exista outro código G específico para a ação.

Somente um código G do grupo pode ser programado em cada condição.



Exemplo de programação

```
WHENEVER $AA_IM[X] > 10.5*SIN(45) DO ...
```

Comparação com expressão calculada durante o pré processamento

```
WHENEVER $AA_IM[X] > $AA_IM[X1] DO ...
```

Comparação com outra variável em tempo real

```
WHENEVER ($A_IN[1]==1) OR ($A_IN[3]==0) DO
```

Dois operadores lógicos associados

```
...
```



Condições possíveis:

10.1 Estrutura, informações básicas



840 D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

NCU 573

10.1.4 Ações



Função

Em cada ação síncrona você pode programar uma ou mais ações. Todas as ações programadas em um bloco são iniciadas no mesmo ciclo de interpolação.

A partir da **SW5**, as ações podem utilizar códigos G para ações/ciclos tecnológicos. Este código G especifica outro código G a partir do anteriormente ativado como condição para todas as ações no bloco e ciclo tecnológico de necessário. Caso existam ciclos tecnológicos na parte ativa, então após completar o ciclo tecnológico o código G continua a ser aplicado de forma modal nos blocos subsequentes, até a programação de um novo código G.

Somente um código G de cada grupo (G70, G71, G700, G710) pode ser programado.

Ações possíveis

- Carga de variáveis
- Escrita de dados setting
- Carregar parâmetros de controle
- DELDTG: Cancelar distância a percorrer
- RDISABLE: Liga bloqueio de execução de programa
- Programação das funções auxiliares M, S e H
- STOPREOF: Parada de pré processamento
- FTOC: Corretores online
- Definir condições de avaliação (polinômios)
- SYNFACT: Ativa a avaliação de funções: Controle AC
- Comutação entre vários avanços em um bloco de programa, em função de sinais binários e analógicos
- Corretores de avanço
- Comando de eixos de posicionamento e fusos (start/stop/posição)

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

- PRESETON: Define valor atual
- Ativa ou desativa o movimento de eixos acoplados/principal
- Medição
- Ajuste de funções de segurança
- Saída de sinais digitais e analógicos
- ...



Exemplo de programação

Ação síncrona com dois comandos

```
WHEN $AA_IM[Y] >= 35.7 DO M135 $AC_PARAM=50
```

Caso a condição seja satisfeita, M135 será enviado ao PLC e a correção de avanço posta a 50%.



Como ação, você pode também especificar um programa (programa para um eixo, ciclo tecnológico). Isto deve cobrir somente aquelas ações que podem também ser programadas individualmente em ações síncronas. As ações individuais como um programa são executadas de forma sequencial nos ciclos de interpolação.



Nota

As ações podem ser executadas independentemente do modo selecionado.

As seguintes ações são ativadas somente no modo automático quando o programa estiver em andamento:

- STOPREOF,
- DELDTG.

10.1 Estrutura, informações básicas



840 D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

NCU 573

10.1.5 Relação das ações síncronas

Até SW 3.x

- Programação de sequências a nível de usuário (programa de usinagem)
- Resposta a eventos/estados no ciclo de interpolação
- Lógicas em tempo real
- Acesso a I/Os, controle de estados do comando e da máquina
- Programação de sentenças cíclicas executadas no ciclo de interpolação
- Disparo de funções específicas NC (desabilita leitura de programa, Triggering of specific NC functions (read-in disable, movimentos auxiliares, ...))
- Execução de funções tecnológicas em paralelo com os movimentos de trajetória
- Disparo de funções tecnológicas nos limiares do cloco

SW 4 em diante

- Diagnose possível através de ações síncronas
- Expansão das variáveis utilizadas nas ações síncronas
- Condições complexas nas ações síncronas
- Expansão das expressões em ações síncronas: Combinação de variáveis em tempo real com operações aritméticas básicas e funções no ciclo de interpolação, endereçamento indireto de variáveis através de índices que podem ser alterados online Os dados setting podem ser alterados e avaliados online
- Configurações possíveis: Quantidade de ações simultaneamente ativas podem ser definida através dos dados de máquina
- Movimentação de eixos de posicionamento e fusos a partir de ações síncronas (comando de eixos)
- Preset através de ações síncronas



840 D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

CCU2

- Ativação, desativação, parametrização de acoplamento de eixos: valor principal do acoplamento, movimentação dos eixos acoplados
- Ativação/desativação de funções de medição
- Cames software
- Cancelamento do percurso à ser percorrido sem parada de pré processamento
- Programas para um eixo, ciclos tecnológicos
- Ações síncronas ativas em modo jog, além dos limites do programa
- Ações síncronas que podem ser influenciadas através do PLC
- Ações síncrona protegidas
- Expansão dos movimentos controlados/controles independentes

10.2 Módulos básicos para condições e ações



840 D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

NCU 573

10.2 Módulos básicos para condições e ações



Variáveis em tempo real

As variáveis em tempo real são amostradas e escritas dentro do ciclo de interpolação.

As variáveis em tempo real são

- \$A... , variáveis principais de cálculo,
- \$V... , variáveis servo.

Para identificá-las de forma especial, estas variáveis podem ser programadas com **\$\$**:

\$AA_IM[X] é equivalente à \$\$AA_IM[X].

Dados setting e dados de máquina devem ser identificados com \$\$ quando lidos/escritos no ciclo de interpolação.



Uma lista de variáveis encontra-se no Apêndice.



Cálculos em tempo real

Os cálculos em tempo real estão restritos aos dados tipo INT, REAL e BOOL.

Expressões em tempo real são cálculos que podem ser executados dentro do ciclo de interpolação e que podem ser utilizadas nas condições ou ações nos comandos NC ou na escrita de variáveis.

• Comparações

Em condições, variáveis ou expressões parciais com o mesmo tipo de dados podem ser comparados. O resultado é sempre um dado tipo BOOL.

Todos os comparadores usuais podem ser utilizados (==, <>, <, >, <=, >=).

• Operadores booleanos

Variáveis, constantes e comparações podem ser associadas utilizando os operadores booleanos usuais (NOT, AND, OR, XOR)

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

- **Operações binárias**

As operações binárias B_NOT, B_AND, B_OR, B_XOR podem ser utilizadas.

Os operandos são variáveis ou constantes de tipo INTEGER.

- **Operações aritméticas básicas**

As variáveis de tempo real dos tipos INTEGER e REAL podem ser utilizadas nas operações aritméticas básicas, entre elas ou am conjunto com constantes (+, −, *, /, DIV, MOD).

- **Funções matem**

As funções matemáticas não podem ser aplicadas em variáveis de tempo real do tipo REAL (SIN, COS, TAN, ASIN, ACOS, ABS, TRUNC, ROUND, LN, EXP, ATAN2, ATAN, POT, SQRT, CTAB, CTABINV).

Example:

```
DO $AC_PARAM[ 3 ] = COS( $AC_PARAM[ 1 ] )
```

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Notas

A multiplicação e divisão são executadas antes da adição e subtração e o uso de colchetes nas expressões é permitido.

Os operadores DIV e MOD podem ser utilizados para dados de tipo REAL (SW 4 em diante).

Exemplo:

| | |
|--|---|
| DO \$AC_PARAM[3] = \$A_INA[1]-\$AA_IM[Z1] | ;Subtração de duas variáveis em tempo real |
| WHENEVER \$AA_IM[x2] < \$AA_IM[x1]-1.9 DO \$A_OUT[5] = 1 | |
| | ;Subtração de uma constante de uma variável de tempo real |
| DO \$AC_PARAM[3] = \$INA[1]-4*SIN(45.7 \$P_EP[Y])*R4 | |
| | ;Expressão constante, calculada durante o pré processamento |

- **Indexação**

Variáveis em tempo real podem ser indexadas através de variáveis em tempo real.

Notas

Variáveis que não sejam atualizadas em tempo real não devem ser indexadas com variáveis em tempo real.

Exemplo:

| | |
|--|--|
| WHEN...DO \$AC_PARAM[\$AC_MARKER[1]] = 3 | |
| Illegal: | |
| \$AC_PARAM[1] = \$P_EP[\$AC_MARKER] | |

Exemplo de programação

Exemplo de expressão em tempo real

| | |
|---|---|
| ID=1 WHENEVER (\$AA_IM[Y]>30) AND (\$AA_IM[Y]<40) DO \$AA_OVR[S1]=80 | Seleção de uma janela de posição |
| ID=67 DO \$A_OUT[1]=\$A_IN[2] XOR \$AN_MARKER[1] | Avaliação de dois sinais booleanos |
| ID=89 DO \$A_OUT[4]=\$A_IN[1] OR (\$AA_IM[Y]>10) | Resultado de uma comparação em uma saída física |

10.3 Variáveis em tempo real especiais para ações síncronas



840D
NCU 572
NCU 573



810D
CCU2



840Di

10.3 Variáveis em tempo real especiais para ações síncronas

A svariáveis em tempo real listadas abaixo podem ser utilizadas em ações síncronas:

10.3.1 Memórias/contadores \$AC_MARKER[n]



Função

Variáveis de memória podem ser lidas e escritas em ações síncronas.

Memórias/contadores específicos de canal

\$AC_MARKER[n]

Tipo de dado: INTEGER

Uma variável de memória específica de canal existe em outro canal com o mesmo nome.

Example:

```
WHEN ... DO $AC_MARKER[0] = 2
WHEN ... DO $AC_MARKER[0] = 3
WHEN $AC_MARKER == 3 DO $AC_OVR=50
```

10.3.2 Variável tipo temporizador \$AC_TIMER[n], a partir da SW 4



Função

(não disponível para 840D NCU 571, FM-NC)

A variável de sistema \$AC_TIMER[n] permite que as ações sejam iniciadas a partir de tempos de espera.

Tipo de dado: REAL

Unidade: s

n: Número do temporizador

- **Carrega temporizador**

Um temporizador é incrementado através da carga de um valor \$AC_TIMER[n]=valor

10.3 Variáveis em tempo real especiais para ações síncronas



840 D
NCU 572
NCU 573



810 D
CCU2



840Di

n: Número do temporizador
valor: valor inicial (geralmente 0)

- **Parada do temporizador**

O incremento de um temporizador é interrompido através da atribuição de um valor negativo
\$AC_TIMER[n]=−1

- **Leitura do temporizador**

O valor atual do tempo pode ser lido enquanto o temporizador estiver rodando ou parado. Quando o temporizador for parado através da atribuição do valor −1, o último valor de tempo permanece disponível para leitura.

Exemplo:

Saída do valor atual através de uma saída analógica
500ms após a detecção de uma entrada digital

```
WHEN $A_IN[1] == 1 DO $AC_TIMER[1]=0 ; Zera e parte o temporizador
WHEN $AC_TIMER[1]>=0.5 DO $A_OUTA[3]=$AA_IM[X] $AC_TIMER[1]=−1
```

10.3.3 Parâmetros em ações síncronas \$AC_PARAM[n]



Função

Tipo de dado: REAL

n: número do parâmetro 0–n

Os parâmetros \$AC_PARAM[n] para ações síncronas são utilizados para cálculos e como memória temporária nas ações síncronas.

A quantidade destes parâmetros por canal é definida através do dado de máquina MD 28254:
MM_NUM_AC_PARAM.

Os parâmetros encontram-se disponíveis uma vez por canal existente sob o mesmo nome. As memórias \$AC_PARAM encontram-se na região de memória dinâmica.

10.3 Variáveis em tempo real especiais para ações síncronas



840D
NCU 572
NCU 573



810D
CCU2



840Di

10.3.4 Acesso a parâmetros R \$Rxx



Função

Tipo de dado: REAL

Estas variáveis estáticas são utilizadas para cálculos no programa de usinagem, etc. Elas podem ser acessadas no ciclo de interpolação através da associação do caracter \$.

Exemplos:

| | |
|--|---|
| WHEN \$AA_IM[X]>=40.5 DO \$R10=\$AA_MM[Y] | Escrita no parâmetro R10 |
| WHEN \$AA_IM[X]>=6.7 DO \$R[\$AC_MARKER[1]]=30.6 | ; Leitura do parâmetro R cujo endereço está especificado pela memória 1 |



Notas

Aplicação:

O uso de parâmetros R em ações síncronas permite

- Gravar valores a serem utilizados após o fim do programa, reset do NC ou desligamento.
- Visualização do conteúdo na tela de visualização de parâmetros R
- Gravação de valores definidos por ações síncronas

Os parâmetros R podem ser utilizados como parâmetros aritméticos “normais” (Rxx) ou como variáveis em tempo real (\$Rxx).

Caso você queira utilizar um parâmetro R como variável aritmética “normal” novamente após seu uso em uma ação síncrona, certifique-se de programar uma parada de pré processamento explícita através de STOPRE.

Exemplo:

| | |
|---|-----------------------------------|
| WHEN \$AA_IM[X]>=40.5 DO \$R10=\$AA_MM[Y] | R10 utilizado na ação síncrona |
| G01 X500 Y70 F1000 | |
| STOPRE | Parada de pré processamento |
| IF R10>20 | Avaliação do parâmetro aritmético |



840 D
NCU 572
NCU 573



810 D
CCU2



840Di

10.3.5 Leitura/escrita em dados de máquina e dados setting , a partir da SW4



Função

A partir da SW4, é possível a leitura e escrita em dados de máquina e dados setting (MD, SD) através de ações síncronas

- **Leitura fixa MD, SD**

Os dados são acessados pelas ações síncronas da mesma forma que em um programa normal, e devem ser precedidos do caracter \$

Exemplo:

```
ID=2 WHENEVER $AA_IM[z]<$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

;Neste exemplo, assumimos que a posição reversa 2 para oscilação não será alterada./

- **Leitura alterada de em MD, SD**

Os dados a serem acessados através da ação síncrona devem ser precedidos dos caracteres \$\$, sendo com isto amostrados no ciclo de interpolação.

Exemplo:

```
ID=1 WHENEVER $AA_IM[z]<$$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

;Assumimos aqui de a posição reversa pode ser alterada por algum comando durante a usinagem.

- **Escrita em MD, SD**

Pré condição:

O nível de acesso atual deve permitir escrita. A alteração de MD e SD através de ações síncronas somente tem sentido para aqueles dados em que as alterações são ativadas imediatamente. Os estados de ativação são listados para todos os MDs e SDs em :

Referência /LIS/, Listas

Endereçamento:

Os MD e SDs a serem alterados devem Ter seus endereços acrescentados de \$\$.

Exemplo:

```
ID=1 WHEN $AA_IW[X]>10 DO $$SN_SW_CAM_PLUS_POS_TAB_1[0]=20
```

```
$$SN_SW_CAM_MINUS_POS_TAB_1[0]=30
```

;Alteraddo a posição de comutação de cames eletrônicos.

Nota: As posições de comutação devem ser alteradas de dois a três ciclos de

10.3 Variáveis em tempo real especiais para ações síncronas



840D
NCU 572
NCU 573



810D
CCU2



840Di

interpolação

antes do eixo atingir a nova posição

10.3.6 Variáveis FIFO \$AC_FIFO1[n] ... \$AC_FIFO10[n], SW 4 em diante



Função

Tipo de dado REAL

10 variáveis FIFO (memória de armazenamento circular) estão disponíveis para a gravação de sequências de dados.

Aplicação:

- Medições cíclicas
- Execução de passes

Cada elemento pode ser acessado nos modos de leitura e escrita.

A quantidade de variáveis fifo é definida através do dado de máquina MD 28260: NUM_AC_FIFO.

A quantidade de valores a serem escritos em uma variável FIFO é definida pelo dado de máquina

MD 28264: LEN_AC_FIFO. Todas as variáveis FIFO possuem o mesmo comprimento.

Os índices de 0 a 5 possuem significado especial:

n=0: Durante a escrita: Um novo valor armazenado na FIFO
Durante a leitura: O elemento mais antigo será lido e removido da FIFO

n=1: Acesso ao mais antigo elemento gravado

n=2: Acesso ao mais novo elemento gravado

n=3: Soma de todos os elementos da FIFO

n=4: Quantidade de elementos disponíveis na FIFO. A leitura e escrita em cada elemento é possível.

As variáveis da FIFO são resetadas através do zeramento da quantidade de elementos, p.e. para a primeira variável FIFO: \$AC_FIFO1[4]=0

n=5: índice atual de escrita da FIFO, com relação ao endereço inicial

n=6 à 6+n_{max}:

Acesso ao n-ésimo elemento da FIFO

10.3 Variáveis em tempo real especiais para ações síncronas



840 D
NCU 572
NCU 573



810 D
CCU2



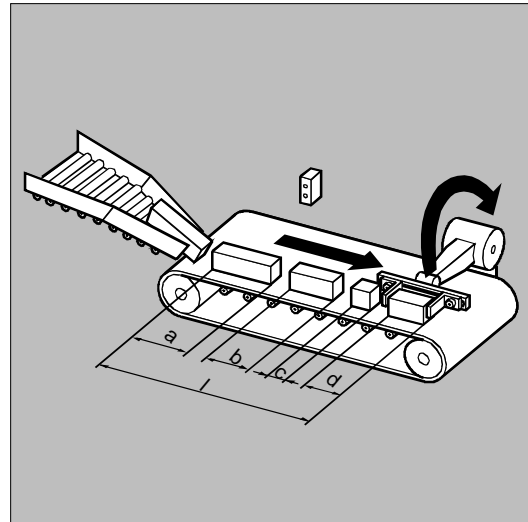
840Di



Exemplo de programação

Memória de circulação

Durante a execução de um programa, uma esteira é utilizada para transportar produtos de diferentes comprimentos (a, b, c, d). A esteira de transporte possui comprimento "l", carregando um número variável de produtos dependendo dos comprimentos individuais envolvidos no processo. Com uma velocidade de transporte constante, a função de remover os produtos da esteira deve ser adaptada ao tempo de chegada variável dos produtos.



| | |
|--|--|
| DEF REAL INTV=2.5 | Distância constante entre os produtos carregados na esteira. |
| DEF REAL TOTAL=270 | Distância entre o comprimento medido e a posição de remição. |
| EVERY \$A_IN[1]==1 DO \$AC_FIFO1[4]=0 | Zeramento da FIFO no início do processo. |
| EVERY \$A_IN[2]==1 DO \$AC_TIMER[0]=0 | Caso um produto interrompa a barreira de luz, inicia uma temporização. |
| EVERY \$A_IN[2]==0 DO \$AC_FIFO1[0]=\$AC_TIMER[0]*\$AA_VACTM[B] | |
| ;caso a barreira de luz esteja livre, calcular e gravar na FIFO o comprimento do produto a partir do tempo medido e da velocidade de transporte. | |
| EVERY \$AC_FIFO1[3]+\$AC_FIFO1[4]*BETW>=TOTAL DO POS[Y]=-30 | |
| \$R1=\$AC_FIFO1[0] | |
| ;Tão logo a soma dos comprimentos de todos os produtos e dos intervalos entre eles for maior ou igual ao comprimento entre as posições de carga e descarga, remover o produto da esteira na posição de descarga, apagar o comprimento do produto da FIFO | |

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

10.4 Comandos nas ações síncronas

10.4.1 Funções auxiliares



Função

Caso as condições sejam satisfeitas, até 10 funções M, H e S podem ser programadas por bloco.

As funções auxiliares são ativadas através da instrução "DO".

As funções auxiliares são emitidas **imediatamente** no ciclo de interpolação. O tempo de emissão definido em dados de máquina para as funções auxiliares não será ativado.

O instante da ativação é determinado quando a condição for satisfeita.

Exemplo:

Liga a refrigeração em uma posição específica do eixo:

```
WHEN $AA_IM[X]>=15 DO M07 G1 X20
F250
```



Sequence

As funções auxiliares devem ser programadas através das instruções WHEN ou EVERY em ações síncronas não modais (sem o ID modal). A ativação ou não da função auxiliar é determinada pelo PLCm p.e. através do NC start.



Notas

Não é possível a partir de uma ação síncrona a programação de:

- M0, M1, M2, M17, M30: parada/fim de programa (M2, M17, M30 possíveis em ciclos tecnológicos)
- M70: Função fuso
- Funções M para troca de ferramentas, M6 ou outra definida através dos dados de máquina.
- M40, M41, M42, M43, M44, M45: troca de marcha



Exemplo de programação

```
WHEN $AA_IW[Q1]>5 DO M172 H510
```

Caso o valor atual do eixo Q1 ultrapasse 5mm as funções auxiliares M172 e H510 serão enviadas ao PLC.

10.4 Comandos nas ações síncronas



840 D
NCU 571



840 D
NCU 572
NCU 573



FM-NC



810 D



840Di

CCU2

10.4.2 Bloqueio da execução do programa RDISABLE



Função

Com RDISABLE a execução do próximo bloco é bloqueada caso a condição seja satisfeita. Um movimento síncrono programado continuará sendo executado, o próximo bloco permanece preparado para execução.

No começo do bloco aco a instrução RDISABLE, o posicionamento exato será sempre ativado, independente da ativação de RDISABLE.



Exemplo de programação

Inicia o programa no ciclo de interpolação em função de entradas externas.

| | |
|--------------------------------------|--|
| ... | |
| WHENEVER \$A_INA[2]<7000 DO RDISABLE | ;Caso a tensão de 7V seja excedida na entrada 2, o programa será parado (1000=1V). |
| N10 G1 X10 | ;Quando a condição for satisfeita, a execução do programa será bloqueada no fim de N10 |
| N20 G1 X10 Y20 | |
| ... | |

840 D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

CCU2

10.4.3 Cancelamento da parada no pré processamento STOPREOF



Função

Caso uma parada explícita de pré processamento STOPRE ou uma parada de pré processamento implícita tenha sido gerada, STOPREOF cancela esta parada tão logo o próximo bloco tenha sido executado e a condição satisfeita.



Notas

STOPREOF deve ser programada com a instrução WHEN e de forma não modal (sem número ID).



Exemplo de programação

Salto rápido para no fim do bloco.

| | |
|--------------------------------|--|
| WHEN \$AC_DTEB<5 DO STOPREOF | ;Cancela a parada de pré processamento quando a distância até o fim do bloco seja menor que 5mm. |
| G01 X100 | ;A parada de pré processamento é cancelada após a execução da interpolação linear. |
| IF \$A_INA[7]>500 GOTOF LABEL1 | ;caso a tensão de 5V seja ultrapassada na entrada 7, saltar para o label1 |
| MARKER1=X100 | |

10.4 Comandos nas ações síncronas



840 D
NCU 571



840 D
NCU 572
NCU 573



FM-NC



810 D



840Di

CCU2

10.4.4 Cancelamento da distância a percorrer

O cancelamento da distância a ser percorrida pode ser disparada para uma trajetória e para um eixo específico dependendo das condições.

As possibilidades são:

- Rápida, cancela a distância a ser percorrida preparada
- Cancela a distância a ser percorrida sem preparação (SW 4.3 em diante)

10.4.5 Cancela distância a percorrer c/preparação, DELDTG, DELTG (eixo1,..)



Função

A distância a percorrer preparada é cancelada com DELTDG permite uma resposta rápida a um evento de disparo e pode ser utilizada em aplicações críticas, como por exemplo, se

- O tempo entre o tempo de cancelamento da distância a percorrer e o próximo bloco deva ser muito pequeno.
- A condição para cancelamento do percurso restante possui grandes chances de ser satisfeita.



Sequence

No final do bloco no qual um comando de cancelamento da distância a percorrer será disparada, uma parada no pré processamento é ativada de forma implícita.

O modo trajetória contínua ou os movimentos de posicionamento dos eixos serão portanto interrompidos ou sofrerão uma parada no final do bloco com o cancelamento da distância a percorrer rápida.

A distância a ser percorrida pode ser recuperada através da variável de sistema \$AC_DELT ou \$AC_DELT[axis].



840 D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

CCU2



Exemplo de programação

Cancelamento rápido da distância a ser percorrida para uma trajetória

```
WHEN $A_IN[1]==1 DO DELDTG
```

```
N100 G01 X100 Y100 F1000 ; Quando a entrada for ativada, o movimento será cancelado
```

```
N110 G01 X...
```

```
IF $AC_DELT>50...
```



Exemplo de programação

Cancelamento rápido da distância a ser percorrida para o eixo

```
POS[X1]=100 G1 Z100 F1000
```

Parando um movimento de posicionamento programado:

```
ID=1 WHEN $A_IN[1]==1 DO MOV[V]=3 FA[V]=700 Parte eixos
```

```
WHEN $A_IN[2]==1 DO DELDTG(V) Cancela a distância a ser percorrida, o eixo será parado  
utilizando-se o comando MOV=0
```

Cancelado a distância a ser percorrida em função de uma
tensão de entrada:

```
WHEN $A_INA[5]>8000 DO DELDTG(X1)
```

```
;Quando a tensão na entrada 5 exceder 8V, cancelar a distância a ser percorrida para o  
eixo X1.
```

```
O movimento de trajetória continua.
```



Restrição

Cancelamento da distância a percorrer preparada

- Não pode ser utilizada com compensação de raio da ferramenta ativo.
- A ação deve somente ser programada de forma não modal (sem número ID).

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

10.4.7 Definição de polinômio, FCTDEF, sincronizada com o bloco



Programação

```
FCTDEF(Polinômio_No., LLIMIT, ULIMIT, a0, a1, a2, a3)
```



Explicação

| | |
|---|--|
| Polinômio_No. | Número do polinômio de grau 3 |
| LLIMIT | Limite inferior para o valor da função |
| ULIMIT | Limite superior para o valor da função |
| a ₀ , a ₁ , a ₂ , a ₃ | Coeficientes do polinômio |



Função

FCTDEF permite a definição de polinômios de grau 3 serem definidos como $y = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3$. Estes polinômios são utilizados pelos corretores de ferramenta online FTOC e para permitir a função SYNFACT calcular valores a partir de variáveis principais (variáveis em tempo real).

Os polinômios são definidos tanto em blocos sincronizados ou através de variáveis de sistema:

| | |
|---------------|---------------------------------|
| \$AC_FCTLL[n] | Valor limite inferior da função |
| \$AC_FCTUL[n] | Valor limite superior da função |
| \$AC_FCT0[n] | a ₀ |
| \$AC_FCT1[n] | a ₁ |
| \$AC_FCT2[n] | a ₂ |
| \$AC_FCT3[n] | a ₃ |
| n | Quantidade de polinômios |

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573



Notas

- As variáveis de sistema podem ser escritas a partir do programa ou de uma ação síncrona. Durante a escrita através do programa, utilize STOPRE para garantir o sincronismo com o bloco de escrita.
- SW 4 em diante:
As variáveis de sistema $\$AC_FCTLL[n]$, $\$AC_FCTUL[n]$, $\$AC_FCT0[n]$ a $\$AC_FCTn[n]$ podem ser alteradas a partir de ações síncronas (não nos SINUMERIK FM-NC, SINUMERIK 840D com NCU 571).

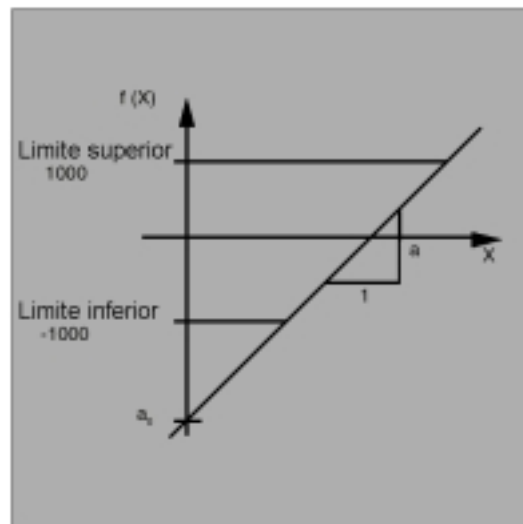
Durante a escrita de ações síncronas os coeficientes do polinômio e os valores limite da função são ativados imediatamente.



Exemplo de programação

Polinômio para um trecho linear:

No limite superior 1000, inferior -1000, a seção ordenada $a_0 = \$AA_IM[X]$ e o gradiente linear 1 do polinômio é:



```
FCTDEF(1, -1000, 1000, $AA_IM[X], 1)
```

10.4 Comandos nas ações síncronas



840D
NCU 572
NCU 573



840Di

10.4.8 Controle de potência laser



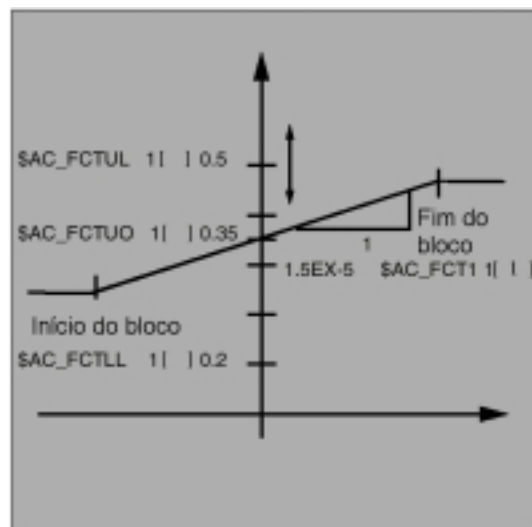
Exemplo de programação

Definição do polinômio utilizando as variáveis

Uma das possíveis aplicações da definição de polinômio é o controle de saída do laser.

O controle de saída laser significa:

Influenciar uma saída analógica em função, por exemplo, da velocidade na trajetória.



```
$AC_FCTLL[1]=0.2
```

Definição dos coeficientes do polinômio

```
$AC_FCTUL[1]=0.5
```

```
$AC_FCT0[1]=0.35
```

```
$AC_FCT1[1]=1.5EX-5
```

```
STOPRE
```

```
ID=1 DO $AC_FCTUL[1]=$A_INA[2]*0.1 +0.35
```

Alteração do limite superior online.

```
ID=2 DO SYNFACT(1,$A_OUTA[1],$AC_VACTW)
```

;em função da velocidade na trajetória (gravada em \$AC_VACTW) o controle de saída laser é feito através da saída analógica 1



Nota

O polinômio definido acima é utilizado com SYNFACT.



840D
NCU 572
NCU 573



840Di

10.4.9 Função avaliação SYNFACT



Programação

SYNFACT(Polinômio_No., Variável de saída em tempo real, Variável de entrada em tempo real)



Explicação

| | |
|-----------------------------------|--|
| Polinômio_No. | Com o polinômio definido com FCTDEF (vide seção "Definição de polinômio"). |
| Variável de saída em tempo real | Variável de escrita em tempo real |
| Variável de entrada em tempo real | Variável de leitura em tempo real |



Função

SYNFACT lê as variáveis em sincronismo com a execução (p.e. entrada analógica, valor atual,...) e as utiliza para calcular os valor da função até o grau 3 (p.e. override, velocidade, posição do eixo, ...) utilizando um polinômio de avaliação (FCTDEF). O resultado é emitido através da variável de saída em tempo real, e sujeitado aos limites inferior e superior com FCTDEF (vide seção 10.4.7).

Como variáveis em tempo real, estas podem ser selecionadas e incluídas diretamente no processo da operação

- Com influência aditiva
- Com influência multiplicativa
- Como deslocamento de posição.



Aplicação

A função de avaliação é utilizada

- Em controle AC (controle adaptável - Adaptive Control)
- Em controle de saída laser
- Com avanço previsto em função da posição

10.4 Comandos nas ações síncronas



840D
NCU 572
NCU 573



840Di

10.4.10 Controle adaptável (aditivo)

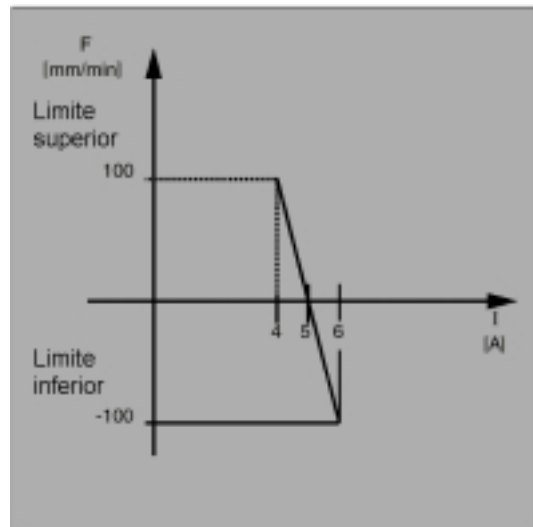


Exemplo de programação

Influência aditiva no avanço programado

Um avanço programado deve ser controlado de forma aditiva em função da corrente do eixo X (eixo de entrada):

A avanço pode variar em ± 100 mm/min e a corrente flutua em ± 1 A em torno do ponto de trabalho de 5A.



1. Definição do polinômio

Determinação dos coeficientes

$$y = f(x) = a_0 + a_1x + a_2x^2 + a_3x^3$$

$$a_1 = -100\text{mm}/1 \text{ min A}$$

$$a_0 = -(-100) \cdot 5 = 500$$

$$a_2 = a_3 = 0 \text{ (sem elementos quadráticos ou cúbicos)}$$

$$\text{Limite superior} = 100$$

$$\text{Limite inferior} = -100$$

Portanto:

```
FCTDEF(1, -100, 100, 500, -100, 0, 0)
```

2. Ativação do controle adaptável (AC)

```
ID=1 DO SYNFACT(1, $AC_VC, $AA_LOAD[x])
```

;Através da leitura da corrente de carga (% da corrente máxima do acionamento) do eixo por intermédio de \$AA_LOAD[x], o avanço da trajetória é calculado de acordo com as definições do polinômio definido acima.



840D
NCU 572
NCU 573



840Di

10.4.11 Controle adaptável (multiplicativo)

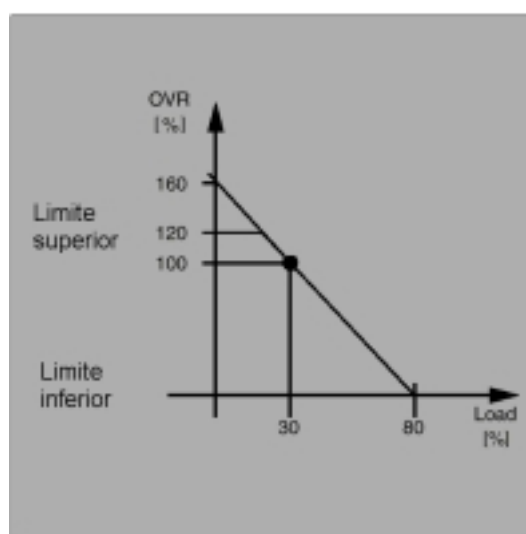


Exemplo de programação

Influencia o avanço programado através de multiplicação

O propósito é influenciar o avanço programado através de uma multiplicação. O avanço não deve exceder certos limites – em função da carga do acionamento:

- O avanço deve ser interrompido caso a carga do acionamento chegue a 80%: Override = 0.
- A uma carga de 30% é possível executar o movimento na velocidade programada: Override = 100%.
- O avanço pode ser excedido em 20%:
Max. override = 120%.



1. Determinação do polinômio

Determinação dos coeficientes

$$y = f(x) = a_0 + a_1x + a_2x^2 + a_3x^3$$

$$a_1 = -100\% / (80 - 30)\% = -2$$

$$a_0 = 100 + (2 \cdot 30) = 160$$

$a_2 = a_3 = 0$ (sem elementos quadráticos ou cúbicos)

Limite superior = 120

Limite inferior = 0

Portanto :

```
FCTDEF ( 2 , 0 , 120 , 160 , -2 , 0 , 0 )
```

2. Ativando o controle adaptável

```
ID=1 DO SYNFACT ( 2 , $AC_OVR , $AA_LOAD[ x ] )
```

;Leitura da corrente de carga do eixo (% da corrente máxima), através de \$AA_LOAD[x],
cálculo do override de avanço conforme o polinômio definido acima.

10.4 Comandos nas ações síncronas



840D
NCU 572
NCU 573



810D



840Di

10.4.12 Controle de tolerância com compensação limitada



Exemplo de programação

Integrando o cálculo das distâncias com controle de limiares

\$AA_OFF_MODE = 1

Importante:

A velocidade de resposta (ganho) do controle externo depende do valor do ciclo de interpolação. Remédio: ler o valor do MD do ciclo de interpolação e levá-lo em conta.

Nota:

Restringir a velocidade do controle externo com MD 32020: JOG_VELO considerando ciclo do interpolação de 12ms:

Velocidade:

$$\frac{0.120\text{mm}}{0.6\text{ms}} / \text{mV} = 0.6 \frac{\text{m}}{\text{min}} / \text{V}$$

Subrotina: Ativa controle de tolerância

%_N_AON_SPF

PROC AON

\$AA_OFF_LIMIT[Z]=1

FCTDEF(1, -10, +10, 0, 0.6, 0.12)

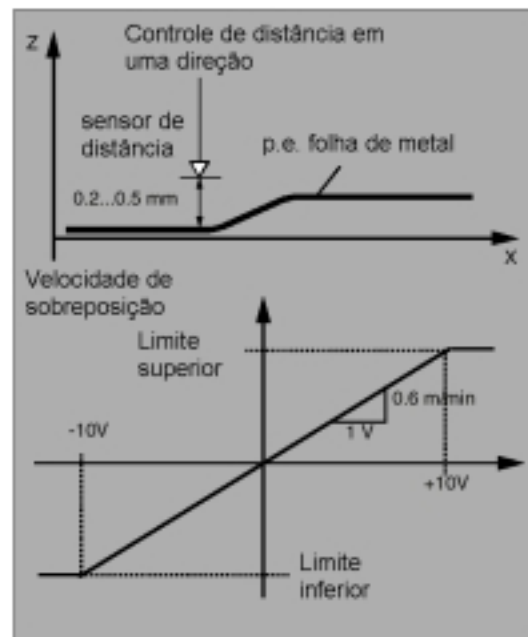
ID=1 DO SYNFACT(1,\$AA_OFF[Z],\$A_INA[3])

ID=2 WHENEVER \$AA_OFF_LIMIT[Z]<>0

DO \$AA_OVR[X] = 0

RET

ENDPROC



Subrotina para ativar o controle de tolerância

Determina o valor limite

Definição do polinômio

Ativa controle de tolerância

Desabilita X caso o valor limite seja ultrapassado

Subrotina: Desativa controle de tolerância

%_N_AOFF_SPF

PROC AOFF

CANCEL(1)

CANCEL(2)

RET

ENDPROC

Subrotina para desligar o controle de tolerância

Cancela a ação síncrona do controle de tolerância

Cancela a checagem de limites



840D
NCU 572
NCU 573



810D



840Di

Programa principal:

```
%_N_MAIN_MPF
```

```
AON
```

Liga controle de tolerância

```
. . .
```

```
G1 X100 F1000
```

```
AOFF
```

Desliga controle de tolerância

```
M30
```



Notas

Deslocamento de posição no sistema de coordenadas básico

Através da variável de sistema \$AA_OFF[eixo] ligada, movimentos externos em quaisquer eixos do canal são possíveis. Irão atuar como deslocamentos de posição no sistema de coordenadas básico.

O deslocamento de posição programado desta forma será sobreposto imediatamente no eixo considerado, tanto estando o eixo o eixo em movimento ou não.

A partir da SW4, é possível limitar o valor absoluto a ser corrigido (valor na variável de saída em tempo real) ao valor definido no dado setting SD 43350: AA_OFF_LIMIT.

A forma de sobrepor a distância definida no dado de máquina MD 36750: AA_OFF_MODE:

0 Avaliação proporcional

1 Avaliação integral

Através da variável de sistema \$AA_OFF_LIMIT[eixo] uma amostragem direcionada para verificar se os valores de deslocamentos se encontram nos limites é possível. Estas variáveis de sistema podem ser amostradas através de ações síncronas e, quando um valor limite for atingido, será possível parar o eixo e disparar um alarme.

0 valor de deslocamento dentro dos limites

1 limite atingido na direção positiva

-1 limite atingido na direção negativa

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

10.4.13 Corretores de ferramenta online FTOC



Programação

FTOC(Polinômio_No., RV, comprimento_2_3 ou raio4, canal, fuso)



Explicação

| | |
|---------------------------|---|
| Polinômio_No. | Para o polinômio definido com FCTDEF, vide a seção “definição de polinômio” neste capítulo. |
| RV | Variável em tempo real para a qual o valor da função para o polinômio especificado deve ser calculado. |
| Comprimento1_2_3 Raio4 | Comprimento da compensação (\$TC_DP1 to 3) ou raio de compensação para o qual o valor calculado da função será adicionado. |
| Canal | Número do canal no qual o corretor será ativado. Nenhuma especificação é feita aqui com relação a deslocamento no canal ativo. FTOCON deve ser ativado no canal de destino. |
| Fuso | Somente especificado caso não o fuso a ser compensado não seja o ativo. |



Função

FTOC permite que eixos geométricos sejam adicionalmente movimentados por intermédio de um polinômio definido com FCTDEF, cujo valor de referência pode ser, por exemplo, a posição atual de outro eixo.

Isto significa que você pode também programar de forma modal, correções de ferramenta online ou controles de tolerâncias por intermédio de ações síncronas.

Aplicação

Usinagem de uma peça e dressamento do rebolo simultâneos, no mesmo canal ou em canais diferentes (canal de usinagem e dressamento).
As condições suplementares e especificações para

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

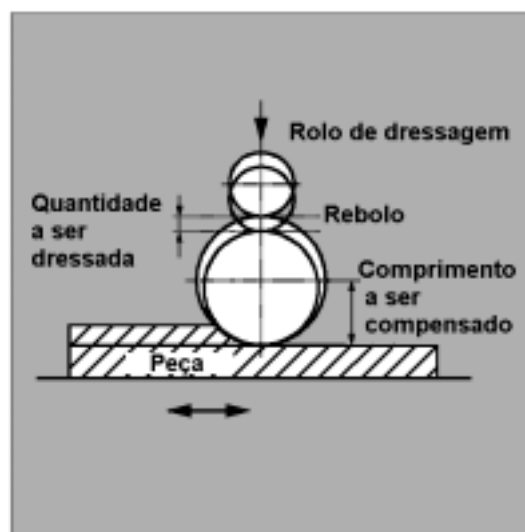
NCU 573

dressamento do rebolo aplicados à FTOC são as mesmas de PUTFTOCF. Para maiores informações, favor consultar o capítulo 5 “Corretores de ferramenta”.



Exemplo de programação

Neste exemplo, devemos fazer a compensação em comprimento do rebolo ativo.



```
%_N_DRESS_MPF
```

```
FCTDEF(1,-1000,1000,-$AA_IW[V],1)
```

Define a função

```
ID=1 DO FTOC(1,$AA_IW[V],3,1)
```

Seleciona a correção de ferramenta online: O valor atual de posição do eixo V é a o valor de entrada para o polinômio 1; o resultado é adicionado ao comprimento 3 do rebolo ativo no canal 1, como sendo o valor deslocamento.

```
WAITM(1,1,2)
```

Sincronismo com o canal de usinagem

```
G1 V-0.05 F0.01 G91
```

Movimento de entrada do dressador

```
G1 V-0.05 F0.02
```

```
...
```

```
CANCEL(1)
```

Desliga o corretor online

```
...
```

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

NCU 573

10.4.14 Movimentos de posicionamento



Função

Eixos podem ser posicionados de forma completamente assíncrona com relação ao programa a partir de ações síncronas. A programação dos eixos de posicionamento a partir de ações síncronas é aconselhável para seqüências cíclicas ou operações que dependam de eventos. Eixos programados a partir de ações síncronas são chamados **eixos de comando**.

A partir da SW5m os códigos G70/G71/G700/G710 podem ser programados em ações síncronas. Eles podem ser utilizados na definição do sistema de medição para tarefas de posicionamento em ações síncronas.

Referências: /PG/ Capítulo 3 "Especificando trajetórias"
/FBSY/ "Introdução aos eixos de comando"



O sistema de medição é definido através de G70/G71/G700/G710.

Ao programar as funções G na ação síncrona, a consideração das dimensões em POLEGADAS/METROS para a ação síncrona pode ser definida independentemente do contexto do programa de usinagem.

Exemplo 1

| | | | |
|------|-----------------------------------|----------|----------|
| N100 | R1=0 | | |
| N110 | G0 X0 Z0 | | |
| N120 | WAITP(X) | | |
| N130 | ID=1 WHENEVER \$R==1 DO POS[X]=10 | | |
| N140 | R1=1 | | |
| N150 | G71 Z10 F10 | Z=10 mm | X=10 mm |
| N160 | G70 Z10 F10 | Z=254 mm | X=254 mm |
| N170 | G71 Z10 F10 | Z=10 mm | X=10 mm |
| N180 | M30 | | |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Exemplo 2

N100 R1=0

N110 G0 X0 Z0

N120 WAITP(X)

N130 ID=1 WHENEVER \$R==1 DO G71 POS[X]=10

N140 R1=1

N150 G71 Z10 F10

Z=10 mm

X=10 mm

N160 G70 Z10 F10

Z=254 mm

X=10 mm (X é sempre
posicionado em 10 mm)
N170 G71 Z10 F10

Z=10 mm

X=10 mm

N180 M30

**Exemplo de programação****Desabilitando um movimento de eixo programado**

Caso você não queira que o movimento do eixo seja realizado no início do bloco, o override para este eixo pode ser mantido em zero até o instante apropriado através de uma ação síncrona.

WHENEVER \$A_IN[1]==0 DO \$AA_OVR[W]=0

G01 X10 Y25 F750 POS[W]=1500 FA=1000

;O eixo de posicionamento permanecerá parado enquanto a entrada digital 1 for =0

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

10.4.15 Posicionando eixo POS



Função

POS[eixo]=valor

De forma diferente da programação através de um programa de usinagem, o movimento do eixo de posicionamento não possui efeito na execução do programa de usinagem.



Explicação

| | |
|--------|--------------------------------|
| Eixo: | Nome do eixo a ser movimentado |
| Valor: | Valor a ser percorrido |



Exemplo de programação

```
ID=1 EVERY $AA_IM[B]>75 DO POS[U]=100
```

O eixo U será movimentado de forma incremental em 100 (mm/inch) ou para a posição 100 (mm/inch) dependendo do modo de movimento especificado

```
ID=1 EVERY $AA_IM[B]>75 DO POS[U]=$AA_MW[V]-$AA_IM[W]+13.5
```

;O eixo U será movimentado através da trajetória calculada a partir das variáveis em tempo real.

10.4.16 Inicia/para movimento MOV



Programação

MOV [Eixo]=valor



Explicação

| | |
|----------------------------|---|
| Eixo: | Nome do eixo a ser movimentado |
| Valor: | Comando para iniciar/parar movimento. O sinal determina o sentido do movimento. O dado é de tipo INTEGER. |
| Valor>0 (normalmente +1): | Direção positiva |
| Valor <0 (normalmente -1): | Direção negativa |
| Valor ==0: | Para o movimento do eixo |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Função

Com MOV[eixo]=valor é possível iniciar o movimento em um eixo sem especificar a posição final. O eixo será movimentado no sentido programado até que outro movimento seja programado ou até a parada do eixo através de um comando de parada.



Exemplo de programação

```
... DO MOV[U]=0
```

O eixo U será parado



Nota

Caso um eixo indexado seja parado com MOV[eixo]=0, o eixo será parado na próxima posição de indexação.

10.4.17 Avanço axial: FA



Exemplo de programação

FA[eixo]=Avanço

```
ID=1 EVERY $AA_IM[B]>75 DO POS[U]=100 FA[U]=990
```

;define valor fixo de avanço

```
ID=1 EVERY $AA_IM[B]>75 DO POS[U]=100 FA[U]=$AA_VACTM[W]+100
```

;calcula o valor de avanço através de variáveis em tempo real

10.4.18 Limite de software



Função

Para os eixos de comando, a limitação da área de trabalho programada com G25/G26 é considerada dependendo do dado setting \$SA_WORKAREA_PLUS_ENABLE. A ativação/desativação da limitação da área de trabalho com as funções G WALIMON/WALIMOF não tem efeito nos eixos de comando.

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

10.4.19 Coordenação de eixos



Função

Tipicamente, um eixo pode ser movimentado tanto através do programa de usinagem em um bloco de movimento ou como eixos de posicionamento através de ações síncronas.

Caso o mesmo eixo deva ser movimentado de forma alternada tanto a partir do programa de usinagem como eixo de trajetória ou como eixo de posicionamento através de ações síncronas, uma transferência de coordenadas é realizada entre ambos movimentos do eixo.

Caso um eixo de comando seja movimentado a partir de um programa de usinagem, o pré processamento necessita ser reorganizado. Isto, em resumo, causa uma interrupção no processamento do programa comparável à uma parada de pré processamento.



Exemplo de programação

O eixo X é movimentado tanto a partir do programa quanto de ações síncronas:

| | |
|---|--|
| N10 G01 X100 Y200 F1000 | Eixo X programado no programa de usinagem |
| ... | |
| N20 ID=1 WHEN \$A_IN[1]==1 DO POS[X]=150 FA[X]=200 | Posição inicial a partir da ação síncrona caso a entrada seja ligada |
| ... | |
| CANCEL(1) | Desligamento da ação síncrona |
| ... | |
| N100 G01 X240 Y200 F1000 | |
| ;X volta a ser eixo de trajetória; antes do movimento ocorrerá um atraso devido à transferência do eixo, caso a entrada digital tenha sido ligada e o eixo tenha se movimentado através da ação síncrona. | |



Exemplo de programação

Alterando o comando de movimento para um mesmo eixo:

| | |
|---|--|
| ID=1 EVERY \$A_IN[1]>=1 DO POS[V]=100 FA[V]=560 | |
| ;Inicia o posicionamento a partir da ação síncrona caso a entrada digital seja >=1 | |
| ID=2 EVERY \$A_IN[2]>=1 DO POS[V]=\$AA_IM[V] FA[V]=790 | |
| O eixo prossegue, a 2ª entrada é ligada, isto é, a posição final e o avanço para o eixo V são continuamente seguidas durante um movimento quando duas ações estão simultaneamente ativas. | |



840D
NCU 572
NCU 573



810D
CCU2



840Di

10.4.20 Carregando o valor atual



Função

Quando PRESETON (eixo, valor) for executado, a posição atual do eixo não será alterada, mas um novo valor será atribuído para esta posição.



Notas

PRESETON pode ser executado a partir de uma ação síncrona nas seguintes situações:

- Eixo rotativo em módulo que foi movimentado a partir do programa de usinagem
- Todos os eixos de comando que foram movimentados a partir de ações síncronas

Restrição:

PRESETON não é possível para eixos que participam de uma transformação.



Exemplo de programação

```
WHEN $AA_IM[a] >= 89.5 DO PRESETON(a4,10.5)
;Carrega o valor 10.5 como posição atual do eixo a4.
```



Restrição

O mesmo eixo pode ser movimentado a partir do programa de usinagem e de ações síncronas somente em diferentes instantes. Por esta razão, atrasos podem ocorrer quando um eixo utilizado no programa de usinagem tenha sido posicionado através de uma ação síncrona anteriormente.

Caso o mesmo eixo seja utilizado alternadamente, a transferência entre os dois movimentos do eixo será coordenada. O programa de usinagem deve ser interrompido para isto.

10.4 Comandos nas ações síncronas



840D
NCU 572
NCU 573



810D
CCU2



840Di

10.4.21 Movimentos com o fuso



Função

Fusos podem ser posicionados de forma completamente assíncrona através de ações síncronas. Este tipo de programação é aconselhável para seqüências cíclicas ou operações que dependam de eventos.



Exemplo de programação

Start/stop/posicionar fusos

| | | | |
|------|--------------------|-------------|----------------------------------|
| ID=1 | EVERY \$A_IN[1]==1 | DO M3 S1000 | Define direção e rotação do fuso |
| ID=2 | EVERY \$A_IN[2]==1 | DO SPOS=270 | Posiciona o fuso |



Seqüência de execução

Caso existam comandos em conflito para um determinado fuso com relação à comandos assíncronos, o fuso obedecerá o mais recente (prioridade para o último comando).



Exemplo de programação

Ativando o sentido de rotação/posição do fuso

| | | | |
|------|---|------------|--|
| ID=1 | EVERY \$A_IN[1]==1 | DO M3 S300 | Define sentido e valor da rotação |
| ID=2 | EVERY \$A_IN[2]==1 | DO M4 S500 | Especifica novo sentido e nova rotação |
| ID=3 | EVERY \$A_IN[3]==1 | DO S1000 | Especifica nova velocidade |
| ID=4 | EVERY (\$A_IN[4]==1) AND (\$A_IN[1]==0) | DO SPOS=0 | Posiciona o fuso |



840D
NCU 572
NCU 573



810D
CCU2



840Di

10.4.22 Movimento de eixos acoplados: TRAILON, TRAILOF



Função

| | |
|---|---|
| DO TRAILON(Eixo seguidor, Eixo principal, Fator de acoplamento) | Ativa o movimento de eixos acoplados |
| DO TRAILOF(Eixo seguidor, Eixo principal, Eixo seguidor 2) | Desativa o movimento de eixos acoplados |

Quando o acoplamento é ativado a partir de uma ação síncrona, o eixo principal pode estar em movimento. Neste caso, o eixo seguidor será acelerado até a velocidade especificada. A posição do eixo principal no instante que as velocidades entrarem em sincronismo será a posição inicial do movimento acoplado. O comportamento do movimento de eixos acoplado é definido na seção “Comportamento dos movimentos na trajetória”.

Ativando de forma assíncrona o movimento acoplado:

... DO **TRAILON**(FA, LA, CF)

Onde: FA: Eixo seguidor
LA: Eixo mestre
CF: Fator de acoplamento

Desativando um movimento de forma assíncrona:

... DO **TRAILOF**(FA, LA, LA2)

Onde: FA: Eixo seguidor
LA: Eixo mestre
LA2: Eixo mestre 2, opcional



Exemplo de programação

| | |
|---------------------------------|--|
| \$A_IN[1]==0 DO TRAILON(Y,V,1) | Ativa o 1º par de eixos combinados através quando o valor da entrada 1 for = 0 |
| \$A_IN[2]==0 DO TRAILON(Z,W,-1) | Ativa o 2º par de eixos combinados |
| G0 Z10 | Avanço dos eixos Z e W em direções opostas |
| G0 Y20 | Avanço dos eixos Y e V na mesma direção |
| ... | |
| G1 Y22 V25 | Sobreposição dependente e movimento independente do eixo acoplado “V” |
| ... | |
| TRAILOF(Y,V) | Desativa o 1º eixo acoplado |
| TRAILOF(Z,W) | Desativa o 1º eixo acoplado |

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

10.4.23 Valor principal de acoplamento LEADON, LEADOF



Função

O valor principal do acoplamento pode ser programado através de ações síncronas sem restrições.

Ativando o valor principal de acoplamento

```
...DO LEADON(FA,LA,NR)
```

Onde: FA: Eixo seguidor
LA: Eixo principal
NR: Número da curva gravada na tabela

Desativando o valor principal de acoplamento

```
...DO LEADOF(FA,LA)
```

Onde: FA: Eixo seguidor
LA: Eixo principal

O eixo a ser acoplado é liberado para acesso das ações síncronas através da função RELEASE para o eixo específico.

Exemplo:

```
RELEASE (XKAN)  
ID=1 every SR1==1 DO LEADON(CACH,XKAN,1)
```



Exemplo de programação

Corte em movimento

Um material contínuo que passa através da área de trabalho de um dispositivo de corte deve ser cortado em peças de comprimentos iguais.

Eixo X: Eixo onde o material corre. WCS

Eixo X1: Eixo de máquina para o material contínuo, MCS

Eixo Y: Eixo no qual o dispositivo de corte se “movimenta” com material contínuo

Assumimos que o posicionamento e controle da ferramenta de corte feito através do PLC. Os sinais da interface PLC podem ser analisados para determinar o grau de sincronismo entre o material contínuo e a ferramenta de corte.

Ações

- Ativa o acoplamento, LEADON
- Desativa o acoplamento, LEADOF
- Define valor atual, PRESETON

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

| | |
|--|--|
| %_N_SHEARS1_MPF | |
| ; \$PATH=/_N_WCS_DIR/_N_DEMOFBE_WPD | |
| N100 R3=1500 | ; Comprimento de uma seção a ser cortada |
| N200 R2=100000 R13=R2/300 | |
| N300 R4=100000 | |
| N400 R6=30 | ; Posição inicial do eixo Y |
| N500 R1=1 | ; Condição inicial para o eixo de transporte |
| N600 LEADOF(Y,X) | ; Cancela qualquer acoplamento existente |
| N700 CTABDEF(Y,X,1,0) | ; Tabela de definições |
| N800 X=30 Y=30 | ; Par de valores |
| N900 X=R13 Y=R13 | |
| N1000 X=2*R13 Y=30 | |
| N1100 CTABEND | ; Fim da definição da tabela |
| N1200 PRESETON(X1,0) | ; PRESET para começar |
| N1300 Y=R6 G0 | ; Posição inicial do eixo Y, eixo é linear |
| N1400 ID=1 WHENEVER \$AA_IW[X]>\$R3 DO PESETON(X1,0) | |
| | ; PRESET após comprimento R3, parte próximo corte |
| N1500 RELEASE(Y) | |
| N1800 ID=6 EVERY \$AA_IM[X]<10 DO LEADON(Y,X,1) | |
| | ; Acopla Y a X através da tabela 1, para X < 10 |
| N1900 ID=10 EVERY \$AA_IM[X]>\$R3-30 DO LEADOF(Y,X) | |
| | ; > 30 antes do corte em movimento, desativa acoplamento |
| N2000 WAITP(X) | |
| N2100 ID=7 WHEN \$R1==1 DO MOV[X]=1 | ; Posiciona o eixo de material em movimento contínuo |
| FA[X]=\$R4 | |
| N2200 M30 | |

10.4 Comandos nas ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

10.4.24 Medição



Comparado com o uso em blocos de movimentação em, programas, as funções de medição podem ser ativadas e desativadas quando necessário.

- Medição axial sem cancelamento da distância a percorrer:

MEAWA[eixo]=(modo, evento de disparo_1, ..._4

- Medição contínua sem cancelamento da distância a percorrer:

MEAC[eixo]=(modo, memória de medição evento de disparo_1, ..._4

Para maiores informações sobre medições: Vide capítulo 5, “Expansão das funções de medição”.

10.4.25 Marcas de espera (WAIT) ligar/cancelar: SETM, CLEARM



Função

SETM (NúmeroDaMarca)

Liga a marca para o canal

CLEARM (NúmeroDaMarca)

Cancela a marca de espera para o canal

Através de ações síncronas, as marcas de espera podem ser ligadas ou desligadas para efeitos de coordenação dos canais, por exemplo.

SETM

O comando SETM pode ser escrito no programa de usinagem e na parte de comando de uma ação síncrona. Ele define o número da marca para o canal onde for executado.

CLEARM

O comando CLEARM pode ser escrito no programa convencional ou na parte de comando de uma ação síncrona. Ele desliga a memória de número da marca para o canal onde for executado.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

10.4.26 Resposta a erros



Função

Respostas a erros podem ser programadas com ações síncronas através da amostragem das variáveis de estado, e com isto disparar as ações apropriadas.

Algumas possibilidades à condições de erros são:

- Parada do eixo: Override=0
- Disparo de alarme: Com SETAL é possível disparar alarmes cíclicos a partir de ações síncronas.
- Ligar saídas
- Todas os comandos possíveis em ações síncronas



Exemplo de programação

```
ID=67 WHENEVER ($AA_IM[X1]-$AA_IM[X2])<4.567 DO $AA_OVR[X2]=0
;Sempre que a distância entre os eixos x1 e X2 for muito pequena, parar o eixo X2
ID=67 WHENEVER ($AA_IM[X1]-$AA_IM[X2])<4.567 DO SETAL(61000)
;caso a distância entre os eixos X1 e X2 for muito pequena, disparar um alarme
```

10.5 Ciclos tecnológicos840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

10.5 Ciclos tecnológicos**Função**

Como uma condição em ações síncronas, você pode chamar programas. Estes devem possuir somente funções permitidas como comandos em ações síncronas. Os programas deste tipo são chamados de ciclos tecnológicos.

Os ciclos tecnológicos são gravados no comando como subrotinas. Do ponto de vista do usuário, devem ser acessadas como subrotinas. Não é possível a transferência de parâmetros.

É possível processar vários ciclos tecnológicos ou ações em paralelo em um canal.

O fim de programa é programado com M02/M17/M30/RET. Pode ser programado no máximo um movimento de eixo por bloco.

**Aplicação**

Os ciclos tecnológicos são programas de eixos: cada ciclo controla apenas um eixo. Desta forma, diferentes movimentos de eixos podem ser iniciados no mesmo ciclo de interpolação através de um evento de controle. O programa de usinagem pode ser utilizado para gerenciar as ações síncronas em casos extremos.

840D
NCU 571840D
NCU 572

FM-NC



810D



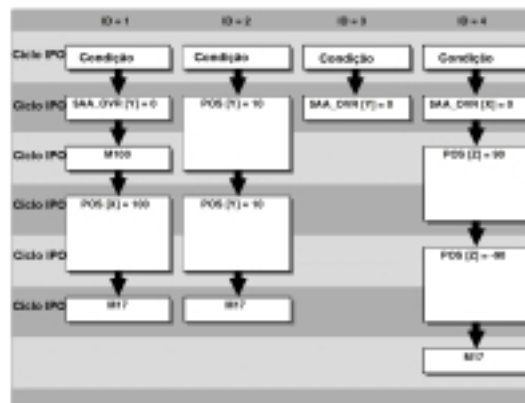
840Di

NCU 573



Exemplo de programação

Os programas de eixos são iniciados através da ativação de entradas digitais.



Programa principal:

| | |
|--|---|
| ID=1 EVERY \$A_IN[1]==1 DO AXIS_X | Caso a entrada 1 esteja em 1, o programa do eixo X inicia |
| ID=2 EVERY \$A_IN[2]==1 DO AXIS_Y | Caso a entrada 2 esteja em 1, prog. eixo Y |
| ID=3 EVERY \$A_IN[3]==1 DO \$AA_OVR[Y]=0 | Caso a entrada 3 esteja em 1, override do eixo Y =0 |
| ID=4 EVERY \$A_IN[4]==1 DO AXIS_Z | Caso a entrada 4 esteja em 1, prog. eixo Z |
| M30 | |

Ciclo tecnológico AXIS_X:

```
$AA_OVR[Y]=0
M100
POS[X]=100 FA[X]=300
M17
```

Ciclo tecnológico AXIS_Y:

```
POS[Y]=10 FA[Y]=200
POS[Y]=-10
M17
```

Ciclo tecnológico AXIS_Z:

```
$AA_OVR[X]=0
POS[Z]=90 FA[Z]=250
POS[Z]=-90
M17
```

10.5 Ciclos tecnológicos



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Os ciclos tecnológicos são iniciados assim que suas condições sejam satisfeitas. Em caso de eixos de posicionamento, vários ciclos de interpolação são necessários para sua execução. Outras funções (OVR) são executadas em um ciclo.

No ciclo tecnológico, os blocos são executados em seqüência.



Notas

Caso várias ações sejam chamados no mesmo ciclo de interpolação e seus comandos entrem em conflito, a ação iniciada será aquela com o mais alto número de ID.

10.5.1 Travar, destravar, resetar: LOCK, UNLOCK, RESET



Programação

| | |
|--------------------|---|
| LOCK (n, n, ...) | Trava o ciclo tecnológico, a ação ativa é interrompida |
| UNLOCK (n, n, ...) | Destrava o ciclo tecnológico |
| RESET (n, n, ...) | Reseta o ciclo tecnológico, a ação ativa é interrompida |
| n | Número de identificação da ação ativa |



Função

A execução de um ciclo tecnológico pode ser travada, destravada ou resetada a partir de uma ação síncrona ou a partir de um ciclo tecnológico.

Travando um ciclo tecnológico, LOCK

Os ciclos tecnológicos podem ser travados através do comando LOCK a partir de uma outra ação síncrona ou a partir de outro ciclo tecnológico.

Exemplo:

```
N100 ID=1 WHENEVER $A_IN[1]==1 DO M130
...
N200 ID=2 WHENEVER $A_IN[2]==1 DO LOCK(1)
```


840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Destravando um ciclo tecnológico, UNLOCK

Um ciclo travado pode ser destravado novamente a partir de outra ação síncrona ou outro ciclo tecnológico através do comando UNLOCK. Com UNLOCK, o ciclo é retomado a partir da posição atual, e isto se aplica também aos processos de posicionamento.

Exemplo:

```
N100 ID=1 WHENEVER $A_IN[1]==1 DO M130
...
N200 ID=2 WHENEVER $A_IN[2]==1 DO LOCK(1)
...
N250 ID=3 WHENEVER $A_IN[3]==1 DO UNLOCK(1)
```

Reset de ciclo tecnológico, RESET

Os ciclos tecnológicos podem ser resetados através do comando RESET, a partir de outra ação síncrona ou outro ciclo tecnológico.

Exemplo:

```
N100 ID=1 WHENEVER $A_IN[1]==1 DO M130
...
N200 ID=2 WHENEVER $A_IN[2]==1 DO RESET(1)
```



Travando a partir do PLC

As ações síncronas modais podem ser intertravadas com o PLC através dos números ID **n=1 ... 64**. As condições associadas não serão mais analisadas e a execução da ação será travada na NCK. Todas as ações podem ser travadas de forma indiscriminada através de um sinal na interface do PLC.



Notas

Uma ação síncrona programada é, por definição sempre ativada e pode ser protegida contra sobrescrita/trava através da configuração de um dado de máquina.

Aplicação:

Pode não ser desejado que os usuários finais alterem as ações síncronas definidas pelo fabricante da máquina.

10.6 Cancelamento de uma ação síncrona: CANCEL840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

10.6 Cancelamento de uma ação síncrona: CANCEL**Programação**`CANCEL (n, n, . . .)`

Cancela uma ação síncrona

N

Número de identificação da ação síncrona

**Explicação**

As ações síncronas modais com identificadores ID(S)=n somente poderão ser canceladas diretamente a partir do programa de usinagem através do comando CANCEL.

Exemplo:`N100 ID=2 WHENEVER $A_IN[1]==1 DO M130`

...

`N200 CANCEL (2)`

Cancela a ação síncrona Nr. 2

**Notas**

Caso a ação seja cancelada em meio a movimentos d eixos, estes movimentos serão completados conforme programados.

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

10.7 Condições adicionais

- **Ligação do comando (Power ON)**

Na ligação do comando (Power ON) nenhuma ação síncrona é ativada.

Entretanto, ações síncronas estáticas podem ser ativadas na ligação do comando através de uma subrotina assíncrona (ASUP), inicializada através do PLC.

- **Troca de modo**

Ações síncronas ativadas através da função IDS permanecem ativas mesmo após a comutação do modo de operação .

Todas as outras ações são desativadas com a alteração do modo de operação (p.e. posicionamento de um eixo) e tornam-se ativas novamente após o reposicionamento e retorno ao modo automático.

- **Reset**

Com o reset do NC, , todas os comandos já disparados pelas ações são parados. As ações síncronas estáticas permanecem ativas. Elas podem iniciar novas ações. O comando de **RESET** pode ser utilizado a partir de uma ação síncrona ou de um ciclo tecnológico para resetar uma ação síncrona ativada de forma modal. Caso uma ação síncrona seja resetada durante um movimento de posicionamento de eixo, o movimento é interrompido.

As ações síncronas do tipo já executadas não serão executadas novamente após o reset.

10.7 Condições adicionais

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

| Resposta após um RESET | | |
|---------------------------------|--|---|
| Ação síncrona/ciclo tecnológico | Modal/não modal | Estáticas (IDS) |
| | As ações ativas são resetadas, as ações sincronizadas são canceladas | A ação ativa é cancelada e o ciclo é resetado |
| Eixo/posicionamento de fuso | O movimento é resetado | O movimento é resetado |
| Fuso com velocidade controlada | \$MA_SPIND_ACTIVE_AFTER_RESET==1: O fuso permanece ativo \$MA_SPIND_ACTIVE_AFTER_RESET==0: O fuso é parado | \$MA_SPIND_ACTIVE_AFTER_RESET==1: O fuso permanece ativo \$MA_SPIND_ACTIVE_AFTER_RESET==0: O fuso é parado |
| Valor principal de acoplamento | \$MC_RESET_MODE_MASK, Bit13 == 1: O valor principal permanece ativo \$MC_RESET_MODE_MASK, Bit13 == 0: O valor principal é cancelado | \$MC_RESET_MODE_MASK, Bit13 == 1: O valor principal permanece ativo \$MC_RESET_MODE_MASK, Bit13 == 0: O valor principal é desconectado |
| Procedimentos de medição | As medições iniciadas a partir de ações síncronas são canceladas | As medições iniciadas a partir de ações síncronas estáticas são canceladas. |

- **NC Stop**

As ações síncronas **estáticas** permanecem ativas após um NC stop. Movimentos iniciados através de ações síncronas estáticas não são canceladas.

As ações síncronas **locais (no programa)** e pertencentes a um bloco já iniciado permanecerão ativas, os movimentos iniciados a partir deste ponto serão interrompidos.

- **Fim de programa**

Fim de programa e as ações síncronas não possuem influência um sobre o outro.

As ações síncronas ativas serão completadas mesmo após o término do programa.

As ações síncronas ativadas no bloco de M30 permanecem ativas. Caso isto não seja desejável, cancele-as com CANCEL antes do término do programa (vide capítulo anterior).

840D
NCU 571840D
NCU 572

FM-NC



810D



840Di

NCU 573

| Resposta ao fim de programa | | |
|---|--|---|
| Ação síncrona/ciclo tecnológico | Modal e não modal são resetadas | Estática (IDS) permanecem ativas |
| Posicionamento de eixos/fusos | M30 é atrasado até que os eixos e fusos estejam parados. | O movimento continua |
| Fusos com controle de velocidade | Fim de programa: \$MA_SPIND_ACTIVE_AFTER_RESET==1: Fuso permanece ativo \$MA_SPIND_ACTIVE_AFTER_RESET==0: Fuso é parado O fuso permanece ativo após uma alteração de modo | O fuso permanece ativo |
| Valor principal de acoplamento | \$MC_RESET_MODE_MASK, Bit13 == 1: O valor principal do acoplamento permanece ativo \$MC_RESET_MODE_MASK, Bit13 == 0: O valor principal é desligado | Um acoplamento iniciado a partir de uma ação síncrona permanece ativo |
| Procedimentos de medição | As medições iniciadas através de ações síncronas são canceladas. | As medições a partir da sincronização estática permanecem ativas |

- **Pesquisa de bloco**

As ações síncronas encontradas durante uma pesquisa de bloco são coletadas e avaliadas no NC start; as ações associadas serão partidas se necessário. As ações síncronas são ativadas durante a pesquisa de bloco

Caso coeficientes de polinômios sejam encontrados durante a pesquisa de bloco, estes serão escritos diretamente em setting data.

- **Interrupção através de subrotina assíncrona**

Partida de ASUP:

Ações modais e de movimentação permanecem ativa e estarão também ativas na subrotina assíncrona.

10.7 Condições adicionais



840D
NCU 571



840D
NCU 572



FM-NC



810D



840Di

• Reposicionamento

Durante o reposicionamento, REPOS, as ações síncronas interrompidas serão reativadas.

As ações síncronas modais alteradas a partir de uma subrotina assíncrona não serão ativadas após REPOS, durante a execução do restante do bloco. Os coeficientes programados com FCTDEF não são afetados por subrotinas assíncronas e por REPOS. Não importam onde tenham sido programados, eles podem ser utilizados em qualquer ponto da subrotina e nos programas principais após a execução de REPOS.

• Desativação com CANCEL

Caso alguma ação síncrona tenha sido cancelada com **CANCEL**, isto não irá alterar a ação ativa. Os movimentos serão encerrados conforme programado.

O comando CANCEL é utilizado para interromper uma ação síncrona modal ou estática ativa.

Caso uma ação síncrona seja cancelada durante o movimento de um eixo de posicionamento disparado pela ação, o movimento deste eixo de posicionamento será interrompido. Caso isto não seja desejado, o movimento do eixo pode ser desacelerado antes do comando CANCEL através de um cancelamento do percurso restante:

Exemplo:

```
ID=17 EVERY $A_IN[3]==1 DO POS[X]=15 FA[X]=1500 ; Posição inicial do movimento
```

...

```
WHEN ... DO DELDTG(X) ; Término do movimento
```

```
CANCEL(1)
```

Oscilações

| | |
|--|--------|
| 11.1 Oscilação assíncrona | 11-396 |
| 11.2 Oscilação controlada através de ações síncronas | 11-403 |

11.1 Oscilação assíncrona



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

11.1 Oscilação assíncrona



Explicação dos comandos

| | |
|---------------|---|
| OSP1[eixo]= | Posição do ponto reverso 1 |
| OSP2[eixo]= | Posição do ponto reverso 2 |
| OST1[eixo]= | Tempo de parada nos pontos reversos em segundos |
| OST2[eixo]= | |
| FA[eixo]= | Avanço para o eixo oscilatório |
| OSCTRL[eixo]= | (Opções para ligar/desligar) |
| OSNSC[eixo]= | Quantidade de paradas para faiscamento |
| OSE[eixo]= | Posição final |
| OS[eixo]= | 1 = ativa a oscilação; 0 = desativa a oscilação |



Função

Um eixo oscilatório se movimenta para frente e para trás entre 2 pontos de reversão 1 e 2, a uma velocidade pré definida, até que o movimento de oscilação seja desativado.

Outros eixos podem ser interpolados como normalmente durante o movimento de oscilação.

Um movimento de trajetória ou eixo de posicionamento pode ser utilizado para executar um movimento de avanço, entretanto, **sem** relação com o movimento oscilatório.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Eixo de oscilação

Para o eixo de oscilação as seguintes regras são aplicadas:

- Qualquer eixo pode ser utilizado como eixo de oscilação.
- Vários eixos de oscilação podem ser ativados simultaneamente (máximo: a quantidade de eixos de posicionamento).
- Para a oscilação sempre é ativada uma interpolação linear G1 – independente do comando G atualmente válido no programa.

O eixo de oscilação pode

- Atuar como parâmetro de entrada (eixo) para uma transformação dinâmica
- Atuar como eixo guia para eixos tipo gantry ou para movimentos de eixos combinados
- Ser movimentado
 - sem limitação de solavancos (BRISK) ou
 - com limitação de solavancos (SOFT) ou
 - com curva de aceleração com joelho (como para os eixos de posicionamento).

Pontos reversos de oscilação

Os deslocamentos atualmente ativos devem ser levados em consideração durante a definição das posições de oscilação:

- Especificação absoluta

$OSP1[Z]=valor$

Posição do ponto reverso = soma dos deslocamentos + valor programado

- Especificação relativa

$OSP1[Z]=IC(valor)$

Posição no ponto de reversão = ponto de reversão 1 + valor programado

Exemplo:

N10 $OSP1[Z]=100$ $OSP2[Z]=110$

.

.

N40 $OSP1[Z]=IC(3)$

11.1 Oscilação assíncrona



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Propriedades da oscilação assíncrona

- A oscilação assíncrona permanece ativa após a execução do bloco, para o eixo específico.
- A ativação do movimento de acordo com o programado no bloco é garantido pelo programa NC.
- Combinação de diversos eixos e sobreposição de trajetórias oscilação não são possíveis.

Dados setting

Os dados setting necessários para a realização de uma oscilação assíncrona podem ser carregados através do programa NC.

Caso os dados setting sejam programados diretamente no programa, as alterações serão efetuadas durante o pré processamento. Utilize STOPRE para sincronizar as alterações e comandos.

Exemplo:

Oscilação com alteração online do ponto reverso

```
$SA_OSCILL_REVERSE_POS1[Z]=-10
```

```
$SA_OSCILL_REVERSE_POS2[Z]=10
```

```
G0 X0 Z0
```

```
WAITP ( Z )
```

```
ID=1 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS1[Z] DO  
$AA_OVR[X]=0
```

```
ID=2 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS2[Z] DO  
$AA_OVR[X]=0
```

```
;Caso o valor atual do eixo de oscilação
```

```
;tenha ultrapassado o ponto reverso,
```

```
;o avanço deve ser interrompido.
```

```
OS[Z]=1 FA[X]=1000 POS[X]=40
```

```
;Ligar a oscilação
```

```
OS[Z]=0
```

```
;Desligar a oscilação
```

```
M30
```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Informações a respeito das funções individuais

Os seguintes endereços permitem que oscilações assíncronas sejam ativadas e controladas a partir do programa NC.

Os valores programados são carregados nos setting data correspondentes durante a execução do programa NC e permanecem ativos até serem novamente programados.

Ativando, desativando a oscilação: OS

OS[eixo] = 1: ativar

OS[eixo] = 0: desativar



WAITP (eixo):

- Caso oscilações devam ser realizadas em um eixo geométrico, este eixo deve ser habilitado para a oscilação com WAITP.
- Quando a oscilação for encerrada, este comando será utilizado para redefinir o eixo de oscilação como um eixo de posicionamento, para uso normal

Tempos de parada nos pontos de reversão: OST1, OST2

| Tempo de permanência | Movimento na parada exata no ponto de reversão |
|----------------------|--|
| -2 | A interpolação continua sem esperar pela parada exata |
| -1 | Espera pela parada exata grossa (coarse) |
| 0 | Espera pela parada exata fina (fine) |
| >0 | Espera pela parada exata fina e após esta pelo tempo de parada |

A unidade de tempo utilizada é a mesma utilizada para G4.



Nota

Oscilação através de ações síncronas e tempos de parada "OST1/OST2"

Após esperar pelo tempo de parada, a alteração interna de blocos ocorre durante a oscilação (visível através da trajetória residual dos eixos).

Após completar uma mudança interna de bloco, a função de desativação é checada. Durante esta checagem, a desativação da função é definida para

11.1 Oscilação assíncrona



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

o controle de acordo com o programado nas seqüências de movimentos “OSCTRL”

Esta temporização é ativada pelo override de avanço.

Sob certas circunstâncias, uma paradas ocorrem durante a oscilação antes das paradas de faiscamento serem iniciadas ou quando a posição final for atingida.

Isto gera a impressão que a resposta foi alterada durante a desativação. Entretanto, este não é o caso.

Carregando o valor de avanço FA

O avanço é o definido para o eixo de posicionamento.

Caso não haja definições, o valor será o definido em dados de máquina.

Definindo a seqüência de movimentos: OSCTRL

Os valores carregados no controle para o movimento são carregados com opções de habilitação e de reset.

Opções de Reset

Estas opções são desativadas (somente caso previamente ativadas como opções).

Opções de ligação

Estas opções são comutáveis. Quando OSE (posição final) for programada, a opção 4 será ativada de forma implícita.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

| Valor da opção | Significado |
|----------------|--|
| 0 | Quando a oscilação for desativada, parar no próximo ponto de reversão (default). Somente possível com o cancelamento dos valores 1 e 2 |
| 1 | Quando a oscilação for desativada, para no ponto reverso 1 |
| 2 | Quando a oscilação for desativada, para no ponto reverso 2 |
| 3 | Quando desativada a oscilação, não atingir o ponto reverso caso não tenham sido programadas paradas para faiscamento. |
| 4 | Ir para a posição final após faiscamento |
| 8 | Caso o movimento de oscilação seja cancelado através do cancelamento do caminho restante: executar as paradas de faiscamento e ir para a posição final caso apropriado |
| 16 | Caso o movimento seja cancelado através do cancelamento da distância a ser percorrida: A posição reversa será atingida como na desativação |
| 32 | O novo avanço será ativado somente após próximo ponto reverso |
| 64 | FA = 0: alteração de trajetória ativa FA = 0: alteração de velocidade ativa |
| 128 | DC para eixo rotativo (caminho mais curto) |
| 256 | 0= A avanço de faiscamento é duplo (default); 1 =somente uma parada de avanço |

Várias opções podem ser ativadas simultaneamente, através do caractere +.

Exemplo:

OSCTRL[Z] = (1+4,16+32+64)

11.1 Oscilação assíncrona



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Exemplo de programação

O eixo oscilatório Z deve oscilar entre 10 e 100. Aproximar do ponto de reversão 1 com parada de avanço fina, e do ponto reverso 2 com parada exata grossa. A usinagem é feita com um avanço de 250 para o eixo de oscilação. Ao término da usinagem, 3 paradas para faiscamento devem ser executadas e o eixo oscilatório deve ser posicionado em 200. A velocidade para o eixo de avanço é 1, o fim do avanço na direção X é 15.

| | |
|-------------------------------|---|
| WAITP(X,Y,Z) | Posição inicial |
| G0 X100 Y100 Z100 | Comuta para operação como eixos de posicionamento |
| N40 WAITP(X,Z) | |
| N50 OSP1[Z]=10 OSP2[Z]=100 -> | Ponto de reversão 1,ponto de reversão 2 |
| -> OSE[Z]=200 -> | Posição final |
| -> OST1[Z]=0 OST2[Z]=-1 -> | Tempo de parada em U1, parada exata fina |
| -> FA[Z]=250 FA[X]=1 -> | Tempo de parada em U2, parada exata grossa |
| -> OSCTRL[Z]=(4,0) -> | Avanço para o eixo de oscilação, eixo da avanço |
| -> OSNSC[Z]=3 -> | Especifica opções |
| N60 OS[Z]=1 | Três paradas para faiscamento |
| | Inicia a oscilação |
| N70 WHEN \$A_IN[3]==TRUE -> | Cancela percurso restante |
| -> DO DELDTG(X) | |
| N80 POS[X]=15 | Posição inicial eixo X |
| N90 POS[X]=50 | |
| N100 OS[Z]=0 | Para a oscilação |
| M30 | |

-> Pode ser programado em um único bloco.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

11.2 Oscilação controlada através de ações síncronas



Programação

1. Define parâmetros para oscilação
2. Define as ações síncronas de movimento
3. Definir eixos, definir avanço

Parâmetros para a oscilação

| | |
|--------------------------|--|
| OSP1[eixo oscilatório]= | Posição do ponto reverso 1 |
| OSP2[eixo oscilatório]= | Posição do ponto reverso 2 |
| OST1[eixo oscilatório]= | Tempo de parada no ponto reverso 1 em segundos |
| OST2[eixo oscilatório]= | Tempo de parada no ponto reverso 2 em segundos |
| FA[EixoOscilatório]= | Avanço para o eixo oscilatório |
| OSCTRL[EixoOscilatório]= | Ativa ou desativa opções |
| OSNSC[EixoOscilatório]= | Quantidade de paradas para faiscamento |
| OSE[EixoOscilatório]= | Posição final |
| WAITP(EixoOscilatório) | Habilita eixo para oscilação |

Definição do eixo, avanço

OSCILL[EixoOscilatório] = (EixoDeAvanço1, EixoDeAvanço2, EixoDeAvanço3)
 POSP[EixoDeAvanço] = (PosFinal, Comprimento Parcial, Modo)

| | |
|---------------------|---|
| OSCILL | Define eixo de entrada para eixo oscilatório |
| POSP | Define avanços parciais (vide capítulo 3) |
| Endpos | Posição final do eixo de avanço e todos os avanços parciais foram percorridos. |
| Comprimento parcial | Comprimento dos avanços parciais em pontos/área de reversão |
| Modo | Divisão do avanço completo em avanços parciais 0 = Dois passos residuais de igual tamanho (default); 1 = Todos os avanços parciais de igual tamanho |

Ações síncronas de movimento

| | |
|-----------------|-----------------------|
| WHEN... .. DO | quando ... , faça |
| WHENEVER ... DO | Sempre que ... , faça |

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Controle de oscilações através de ações síncronas

Com este modo de oscilação, o movimento de entrada pode ser executado somente no ponto de reversão, ou dentro de áreas de reversão definidas.

Dependendo da necessidade, os movimentos de oscilação podem:

- Continuar ou
- Parar até que o movimento de avanço tenha sido completado.



Seqüência

1. Definir os parâmetros de oscilação

Os parâmetros para oscilação devem ser definidos antes do bloco de movimento contendo que contenha a programação dos eixos de avanço, de oscilação e a definição de avanço (vide “Oscilação assíncrona”).

2. Definir as ações síncronas de movimento

As seguintes condições de sincronismo podem ser definidas:

- **Supressão do avanço de entrada** até que o eixo de oscilação esteja em uma área de reversão (ii1, ii2) ou em um ponto de reversão (U1, U2).
- **Parada do movimento de oscilação** durante o avanço no ponto de reversão.
- **Reinicia o movimento de oscilação** após completar o avanço parcial.
- Define o início do **próximo avanço parcial**.

3. Define os eixos de oscilação e de avanço bem como os avanços parciais e total.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Definindo os eixos de oscilação e de avanço:****OSCILL**

```
OSCILL[Eixo de oscilação] = (Eixo de avanço1, Eixo de avanço2, Eixo de avanço3)
```

A definição do eixo e o início do movimento de oscilação são definidos através do comando OSCILL.

Até 3 eixos de avanço podem ser definidos para um eixo de oscilação.



Antes de iniciar a oscilação, as condições de sincronismo devem ser especificadas para definir o comportamento dos eixos.

**Definindo avanços: POSP**

```
POSP[EixoDeAvanço] = (PosiçãoFinal, Parte, Modo)
```

As seguintes declarações são realizadas com o comando POSP:

- Avanço completo (com referência à posição final)
- O comprimento do avanço parcial no ponto ou área de reversão
- A resposta do avanço parcial quando a posição final for atingida (com referência ao modo)

Modo = 0

A distância a ser percorrida até o ponto de destino para os dois últimos avanços parciais é dividida em dois passos iguais (default).

Modo = 1

Todos os avanços parciais possuem mesmo tamanho. São calculados a partir do comprimento total de avanço.

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



As ações síncronas

As ações de movimentos síncronos listadas abaixo são utilizadas para as oscilações em geral.

Serão dados exemplos para algumas aplicações os quais você poderá utilizar como modelo para criar seus movimentos de oscilação específicos.



Em alguns casos, as ações síncronas são programadas de forma diferenciada.



Instruções

| | |
|-----------------|------------------------|
| WHEN ... DO ... | Quando, faça |
| WHENEVER ... DO | Sempre que, faça |



Você poderá criar as seguintes funções com os recursos de programação descritos a seguir:

1. Avanço no ponto de reversão.
2. Avanço na área de reversão.
3. Avanço em ambos os pontos de reversão.
4. Parada do movimento de oscilação no ponto de reversão.
5. Reinício do movimento de oscilação.
6. Não iniciar o avanço de forma muito antecipada.

As seguintes considerações são feitas para todos os exemplos onde encontram-se ações síncronas aqui apresentados:

- Ponto de reversão 1 < ponto de reversão 2
- Z = eixo oscilatório
- X = eixo de avanço



Você irá encontrar maiores informações a respeito dos movimentos com ações síncronas na seção 11.3.

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Avanço na área reversa

O movimento de avanço deve sempre ser iniciado dentro de uma área de reversão antes que o ponto de reversão seja atingido.

Estas ações síncronas inibem o movimento de avanço até que o eixo de oscilação esteja dentro da área de oscilação.



As seguinte instruções são utilizadas de acordo com as considerações acima:



Área reversa 1:

```
WHENEVER $AA_IM[Z] > $SA_OSCILL_REVERSE_POS1[Z] + ii1 DO $AA_OVR[X] = 0
```

Sempre que
Maior que
então

A posição atual do eixo de oscilação no sistema MCS for
O início da área de reversão 1
Atribuir o valor 0% ao override do eixo de avanço.

Área de reversão 2:

```
WHENEVER $AA_IM[Z] < $SA_OSCILL_REVERSE_POS2[Z] + ii2 DO $AA_OVR[X] = 0
```

Sempre que
Menor que
Então

A posição atual do eixo de oscilação no sistema MCS for
Menor que o início da área de reversão 2
Atribuir o valor de 0% ao override do eixo de avanço.

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Avanço no ponto de reversão

Enquanto o eixo de oscilação não atingir o ponto de reversão, nenhum movimento será executado no eixo de avanço.



As seguinte instruções são utilizadas de acordo com as considerações acima:

Ponto de reversão 1:

```
WHENEVER $AA_IM[Z] <> $SA_OSCILL_REVERSE_POS1[Z] DO $AA_OVR[X]=0 ->
-> $AA_OVR[Z]=100
```

Sempre que

Maior ou menor que

Então

E

A posição atual do eixo de oscilação Z no sistema MCS for

A posição no ponto de reversão 1

Atribuir o valor 0% ao override do eixo de avanço X

Atribuir o valor 100% ao override do eixo de avanço Z.

Ponto de reversão 2:

Para o ponto de reversão 2:

```
WHENEVER $AA_IM[Z] <> $SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[X]=0 ->
-> $AA_OVR[Z]=100
```

Sempre que

Maior ou menor que

Então

E

A posição atual do eixo de oscilação Z nos sistema MCS for

A posição do ponto de reversão 2

Atribuir o valor de 0% ao override do eixo de avanço X

Atribuir o valor de 100% ao override do eixo de oscilação Z.

11.2 Oscilação controlada através de ações síncronas

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Parada do movimento de oscilação no ponto de reversão

O eixo de oscilação será parado no ponto de reversão, e o movimento de avanço é iniciado ao mesmo tempo.

O movimento de avanço continuará assim que o de avanço tenha terminado.

Esta ação síncrona pode também ser utilizada para iniciar o movimento de avanço caso este tenha sido inibido por uma ação anterior ainda ativa.



As seguinte instruções são utilizadas de acordo com as considerações acima:

Ponto de reversão 1:

```
WHENEVER $SA_IM[Z] == $SA_OSCILL_REVERSE_POS1[Z] DO $AA_OVR[Z] = 0 ->
-> $AA_OVR[X] = 100
```

| | |
|------------|--|
| Sempre que | A posição atual do eixo de oscilação no sistema de medição MCS for |
| Igual a | Posição do ponto de reversão 1 |
| Então | Atribuir o valor de 0% ao override do eixo de oscilação |
| e | Atribuir o valor de 100% ao override do eixo de avanço |

Ponto de reversão 2:

```
WHENEVER $SA_IM[Z] == $SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[Z] = 0 ->
-> $AA_OVR[X] = 100
```

| | |
|------------|--|
| Sempre que | A posição atual do eixo de oscilação no sistema de medição MCS for |
| a | A posição do ponto de reversão 2 |
| Então | Atribuir o valor de 0% ao override do eixo de oscilação |
| E | Atribuir o valor de 100% ao override do eixo de avanço. |

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di



Avaliação do ponto de reversão online

Caso exista uma variável principal codificada com \$\$ à direita da comparação, então ambas as variáveis serão avaliadas e comparadas continuamente no ciclo de interpolação (IPO).



Favor consultar a seção “Movimentos com ações síncronas” para maiores informações.



Reinício dos movimentos de oscilação

Esta ação síncrona é utilizada para dar continuidade ao movimento de oscilação após completado o movimento de avanço parcial.



As seguinte instruções são utilizadas de acordo com as considerações acima:

```
WHENEVER $AA_DTEPW[X]==0 DO $AA_OVR[Z]= 100
```

| | |
|------------|---|
| Sempre que | A distância a ser percorrida para o avanço parcial do eixo de avanço X no sistema WCS for |
| Igual à | Zero |
| Então | Atribuir o valor de 100% ao override do eixo de oscilação |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Próximo avanço parcial

Após completado o avanço, uma partida antecipada do próximo movimento de avanço deve ser interrompida.

Uma memória específica do canal (\$AC_MARKER[índice]) será utilizada para este propósito. Ela será habilitada após o fim do movimento de avanço (distância a percorrer parcial = 0) e desligada após o eixo deixar a área de reversão. Uma ação síncrona será então utilizada para inibir o próximo movimento de avanço.



Com base nestas considerações, as seguintes instruções serão aplicadas no ponto de reversão 1:

1. Ligar a memória

```
WHENEVER $AA_DTEPW[X] == 0 DO $AC_MARKER[1]=1
```

| | |
|------------|--|
| Sempre que | A distância a percorrer para o avanço parcial do eixo X no WCS for |
| Igual a | Zero |
| Então | Atribuir o valor 1 à memória |

2. Zerando a memória

```
WHENEVER $AA_IM[Z] <> $SA_OSCILL_REVERSE_POS1[Z] DO $AC_MARKER[1]=0
```

| | |
|--------------------|---|
| Sempre que | A posição atual do eixo de oscilação Z no sistema MCS for |
| Maior ou menor que | A posição do ponto de reversão |
| Então | Atribuir o valor 1 à memória |

2. Inibir o avanço

```
WHENEVER $AC_MARKER[1]==1 DO $AA_OVR[X]=0
```

| | |
|------------|---|
| Sempre que | A memória 1 for |
| Igual a | 1, |
| Então | atribuir o valor 0% ao override do eixo de avanço |

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D

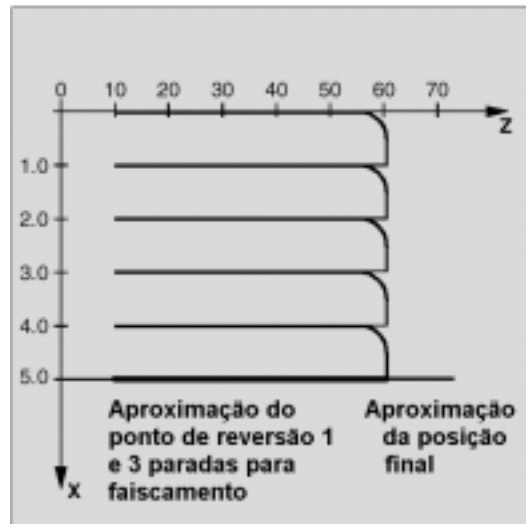


840Di



Exemplo de programação

Nenhum movimento de avanço é realizado no ponto de reversão 1. No ponto de reversão 2, o avanço deveria ter início à distância *ii2* antes do ponto de reversão 2 e o eixo de oscilação não deve naquele ponto o fim do avanço parcial. O eixo Z é o eixo de oscilação e o X o eixo de avanço.



Trecho do programa

1. definir parâmetros para oscilação

| | |
|-----------------------|---|
| DEF INT <i>ii2</i> | Define a variável p/a área de reversão 2 |
| OSP1[Z]=10 OSP2[Z]=60 | Define os pontos de reversão e 2 |
| OST1[Z]=0 OST2[Z]=0 | Ponto de reversão 1: Parada exata fina Ponto de reversão 2: Parada exata fina |
| FA[Z]=150 FA[X]=0.5 | Avanço durante a oscilação do eixo Z, avança o eixo X |
| OSCTRL[Z]=(2+8+16,1) | Desativa os movimentos de oscilação no ponto de reversão 2; após cancelar a distância a percorrer de faiscamento e aproximar da posição final, após apagar a distância a percorrer e aproximar a posição de reversão. |
| OSNC[Z]=3 | 3 paradas para faiscamento |
| OSE[Z]=70 | Posição final = 70 |
| <i>ii2</i> =2 | Valor da área reversa |
| WAITP(Z) | Habilita a oscilação para o eixo Z |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2. Ação síncrona de movimentação

```
WHENEVER $AA_IM[Z]<$SA_OSCILL_REVERSE_POS2[Z]-ii2 DO ->
-> $AA_OVR[X]=0 $AC_MARKER[0]=0
```

| | |
|------------|--|
| Sempre que | A posição do eixo de oscilação Z no sistema de coordenadas MCS |
| Menor que | for |
| Então | O início da área de reversão 2 |
| E | Atribuir o valor de 0% ao eixo de avanço X |
| | Carregar na memória de índice 0 o valor 0. |

```
WHENEVER $AA_IM[Z]>=$SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[Z]=0
```

| | |
|------------------|---|
| Sempre que | A posição atual para o eixo de oscilação Z no sistema MCS for |
| Maior ou igual a | Posição do ponto de reversão 2 |
| Então | Atribuir o valor de 0% ao override do eixo de oscilação Z. |

```
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[0]=1
```

| | |
|------------|--|
| Sempre que | A distância a ser percorrida para o avanço parcial for |
| Igual a | 0, |
| Então | atribuir à memória com índice 0 o valor 1 |

```
WHENEVER $AC_MARKER[0]==1 DO $AA_OVR[X]=0 $AA_OVR[Z]=100
```

| | |
|------------|--|
| Sempre que | A memória com índice 0 for |
| Igual a | 1, |
| Então | atribuir o valor 0% ao override do eixo de avanço X para inibir o avanço antecipado (eixo de oscilação Z ainda não abandonou a área 2, mas o eixo de avanço X encontra-se pronto para um novo movimento), atribuir o valor de 100% ao override do eixo de oscilação Z (isto cancela a 2ª ação síncrona). |

➔ deve ser programado em um bloco separado

3. Partindo a oscilação

```
OSCILL[Z]=(X) POSP[X]=(5,1,1)
```

Parte os eixos
Declara eixo X como eixo de avanço para a oscilação em Z.
O eixo X deve atingir a posição final 5 em passos de 1.

```
M30
```

Fim do programa

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



Notas

[illegible]

11.2 Oscilação controlada através de ações síncronas



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Puncionar e riscar

| | | |
|--------|--|--------|
| 12.1 | Ativação, desativação..... | 12-416 |
| 12.1.1 | Instruções..... | 12-416 |
| 12.1.2 | Uso de comandos M | 12-419 |
| 12.2 | Segmentação automática de trajetória..... | 12-420 |
| 12.2.1 | Segmentação da trajetória para eixos de trajetória | 12-421 |
| 12.2.2 | Segmentação de trajetória para um único eixo..... | 12-422 |
| 12.2.3 | Exemplo de programação | 12-424 |

12.1 Ativação, desativação



840 D
NCU 572
NCU 573



840Di

12.1 Ativação, desativação

12.1.1 Instruções



Programação

```
PDELAYON
PON G... X... Y... Z...
PONS G... X... Y... Z...
PDELAYOF
SON G... X... Y... Z...
SONS G... X... Y... Z...
SPOF
```



Explicação dos parâmetros

| | |
|----------|------------------------------|
| PON | Liga puncionar |
| PONS | Liga puncionar com guia |
| SON | Liga riscar |
| SONS | Liga riscar com guia |
| SPOF | Desliga puncionar e riscar |
| PDELAYON | Liga puncionar com atraso |
| PDELAYOF | Desliga puncionar com atraso |



Função

Puncionar e riscar, ativar/desativar PON/SON

As funções de puncionar e riscar são ativadas com PON e SON, respectivamente. SPOF encerra ambas as operações de puncionar e riscar. Os comandos modais PON e SON são exclusivos mutuamente, ou seja PON desativa SON e vice versa.

Puncionando e riscando com guia, PONS/SONS

Os comandos SONS e PONS também ativam as funções puncionar e riscar. De forma diferente a SON/PON – controle de transições no nível de interpolação – PONS e SONS controlam o início de transições nos acionamentos.

Isto significa que você pode trabalhar com frequências de transição mais altas, e com isto aumentar a capacidade de puncionamento.

Quando os sinais são avaliados em relação ao principal, todas as funções que podem causar alterações nas posições de puncionar e riscar são inibidas.

Exemplo: Modo nônio, alteração em frames através do PLC, funções de medição.

Caso contrário, PONS e SONS trabalham exatamente do mesmo modo que PON e SON.

Puncionando com atraso

PDELAYON causa um atraso na ativação da pancada de puncionamento. O comando é modal e possui uma função preparatória. Esta é geralmente programada antes de PON.

O puncionamento continua normalmente após PDELAYOF.

12.1 Ativação, desativação



840 D
NCU 572
NCU 573



840Di



Iniciando as pancadas

Iniciando a primeira pancada

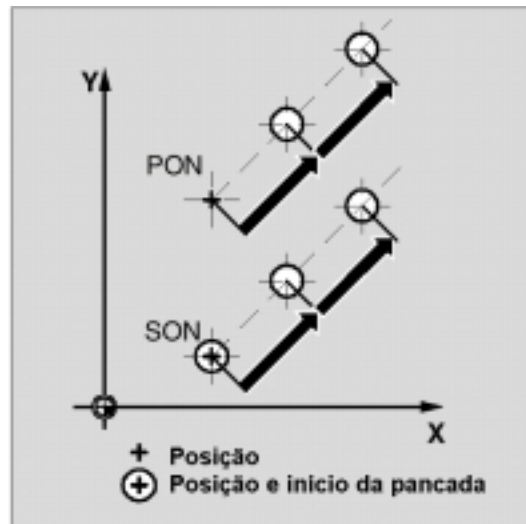
O instante em que a primeira pancada é iniciada após a ativação da função difere dependendo de quando são ativadas as funções de riscar ou puncionar:

PON/PONS:

- Todas as pancadas – Mesmo aquela no primeiro bloco após a ativação – são executadas após o fim do bloco.

SON/SONS:

- A primeira pancada após a ativação das funções de riscar são executadas no início do bloco.
- Cada uma das pancadas seguintes são iniciadas no fim do bloco.



Puncionando e riscando no ponto

Uma pancada é iniciada somente caso o bloco contenha informações para os eixos de punção e riscos (eixos no plano ativo).

Entretanto, caso você deseje iniciar uma pancada na mesma posição, você pode programar um dos eixos de punção/risco com uma trajetória igual à 0.



Informações adicionais

Usinagem com ferramentas rotativas

Utilize a função de controle tangencial caso você deseje posicionar ferramentas rotativas em uma tangente com relação à trajetória programada.

12.1.2 Uso de comandos M



Através do uso da tecnologia das macros você pode também utilizar funções M ao invés das instruções

| | |
|------------------------|----------------------------|
| DEFINE M22 AS SON | Liga riscar |
| DEFINE M122 AS SONS | Liga riscar com guia |
| DEFINE M25 AS PON | Liga puncionar |
| DEFINE M125 AS PONS | Liga puncionar com guia |
| DEFINE M26 AS PDELAYON | Liga puncionar com atraso |
| DEFINE M20 AS SPOF | Desliga puncionar e riscar |
| DEFINE M23 AS SPOF | Desliga puncionar e riscar |

12.2 Segmentação automática de trajetória



840 D
NCU 572
NCU 573



840Di

12.2 Segmentação automática de trajetória



Programação

SPP=

SPN=



Explicação

| | |
|-----|---|
| SPP | Tamanho do trecho da trajetória (Distância máxima entre pancadas); modal |
| SPN | Quantidade de trechos por bloco, não modal |



Função

Segmentação de trajetória

Durante o puncionamento ou execução de riscos, SPP e SPN dividem a distância total programada a ser percorrida pelos eixos de trajetória em trechos desta trajetória de igual comprimento (segmentação constante da trajetória). Cada segmento de trajetória corresponde internamente a um bloco.

Quantidade de pancadas

Enquanto o puncionamento estiver ativo, a primeira pancada é executada no final do primeiro segmento de trajetória.

A quantidade de pancadas para punção/riscos durante a trajetória é obtida através da regra:

Puncionamento:

Quantidade de pancadas = Quantidade de
segmentos de trajetória

Riscos:

Quantidade de riscos = quantidade de segmentos
de trajetória + 1

Funções auxiliares

As funções auxiliares são executadas no primeiro dos blocos gerados.

12.2.1 Segmentação da trajetória para eixos de trajetória



Seqüência

Comprimento da segmentação SPP

Através do comando SPP, você especifica a distância máxima entre as pancadas e desta forma o comprimento máximo dos segmentos de trajetória nos quais a distância a ser percorrida será dividida.

O comando é desativado com SPOF ou SPP=0.

Exemplo:

N10 G1 SON X0 Y0

N20 **SPP=2** X10

Neste exemplo, a distância total percorrida de 10 mm é dividida em 5 segmentos de trajetória com 2 mm cada (SPP=2).



Os segmentos de trajetória afetados por SPP são sempre eqüidistantes, ou seja, todos os segmentos são iguais em comprimento.

Em outras palavras, o comprimento do segmento de trajetória (valor de SPP) é válido somente se o quociente entre a trajetória total e o valor de SPP for um número inteiro

Caso isto não ocorra, o tamanho do segmento de trajetória será reduzido internamente de forma a produzir um resultado inteiro.

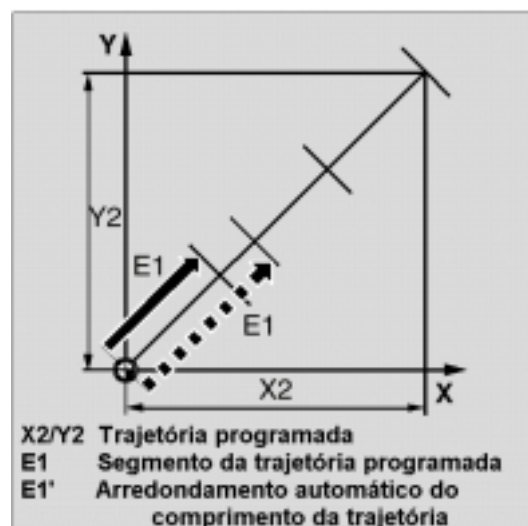
Exemple:

N10 G1 G91 SON X10 Y10

N20 **SPP=3.5** X15 Y15

Quando o comprimento total percorrido for 15mm e o comprimento do segmento de trajetória for 3,5mm, o quociente não será um valor inteiro (4.28).

Neste caso, o valor SPP é reduzido até mais próximo valor que produza um quociente inteiro. O resultado neste exemplo deve ser uma trajetória de comprimento 3 mm.



12.2 Segmentação automática de trajetória



840 D
NCU 572
NCU 573



840Di

Quantidade de segmentos de trajetória SPN

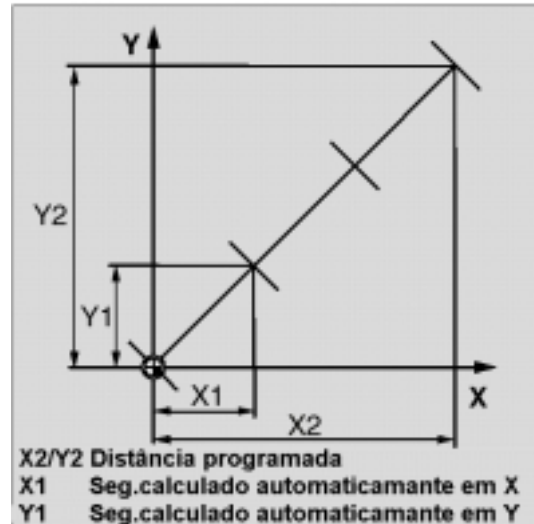
SPN define uma quantidade de segmentos de trajetória a serem gerados a partir de uma distância total a ser percorrida. O comprimento dos segmentos é calculado automaticamente.

Como SPN não é modal, o puncionamento ou riscos devem ser ativados anteriormente com PON e SON respectivamente.

SPP e SPN no mesmo bloco

Caso você programe ambos os comprimento do segmento de trajetória (SPP) e a quantidade de segmentos de trajetória (SPN) no mesmo bloco, então SPN será aplicado a este bloco e SPP aos posteriores.

Caso SPP seja ativado antes de SPN, então terá efeito novamente após o bloco com SPN.



Informações adicionais

Partindo do princípio que as funções para puncionar/riscar estejam disponíveis no controle, então é possível programar uma segmentação automática de blocos através de SPN ou SPP mesmo quando as funções de puncionar ou riscar não estiverem em uso.

12.2.2 Segmentação de trajetória para um único eixo



Caso eixos únicos sejam definidos como eixos de puncionamento/riscar adicionalmente aos eixos de trajetória, então a segmentação automática pode ser ativada para eles.

Resposta de eixo único à SPP

O comprimento do segmento de trajetória programado (SPP) se refere basicamente aos eixos de trajetória.

Por esta razão, o valor de SPP é ignorado em blocos que

tenham movimentos de apenas um eixo e um valor de SPP, mas não um eixo de trajetória programado.

Caso ambos eixos individuais e um eixo de trajetória estiverem programados no bloco, o eixo individual irá responder de acordo com a programação em dados de máquina.

1. Programação default

A trajetória a ser percorrida pelo eixo simples é dividida ao longo dos blocos intermediários gerados através de SPP.

Exemplo:

N10 G1 SON X10 A0

N20 SPP=3 X25 A100

Como um resultado da distância entre pancadas de 3 mm, cinco blocos são gerados para a distância total de 15 mm na direção do eixo X (eixo de trajetória)

O eixo A deve girar em torno de 20° em todos os blocos.

2. Eixos individuais sem segmentação de trajetória.

O eixo individual percorre a distância total no primeiro dos blocos gerados.

3. Com/sem segmentação de trajetória

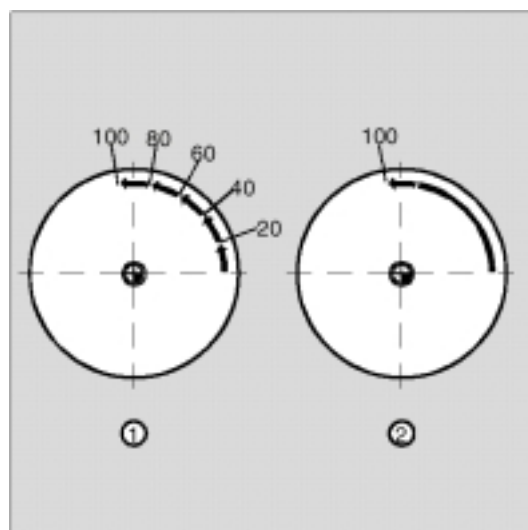
A resposta dos eixos individuais dependem da interpolação dos eixos de trajetória:

- Interpolação circular: Com segmentação de trajetória
- Interpolação linear: Sem segmentação de trajetória

Resposta à SPN

A quantidade programada de segmentos de trajetória é aplicada mesmo quando um eixo de trajetória não esteja programado no mesmo bloco.

Per condição: O eixo deve Ter sido definido como eixo para puncionar/riscar.



12.2 Segmentação automática de trajetória



840 D
NCU 572
NCU 573



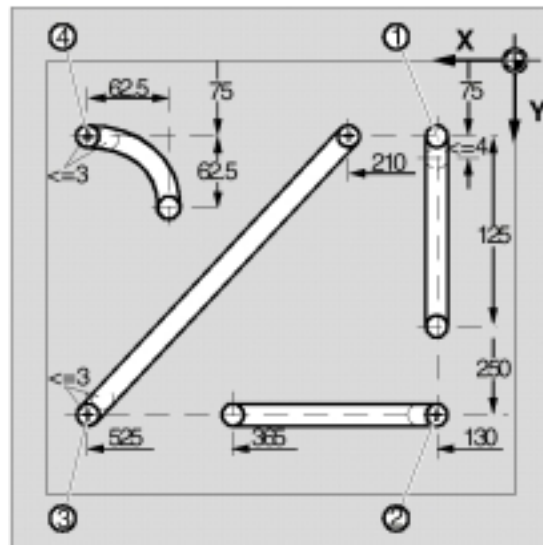
840Di

12.2.3 Exemplo de programação



Exemplo de programação 1

Os trechos programados com riscos devem ser divididos automaticamente com os segmentos de trajetória eqüidistantes



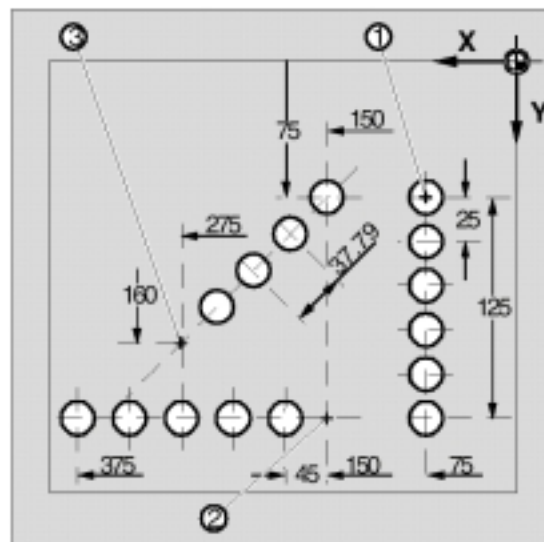
Resumo do programa

| | |
|---|---|
| N100 G90 X130 Y75 F60 SPOF | Posição do ponto inicial 1 |
| N110 G91 Y125 SPP=4 SON | Liga riscar, O comprimento de segmento de trajetória máximo para segmentação: 4mm |
| N120 G90 Y250 SPOF | Desliga riscar, posição inicial do ponto 2 |
| N130 X365 SON | Liga riscar, comprimento de trajetória máximo para trajetórias automáticas: 4 mm |
| N140 X525 SPOF | Desliga riscar, posição inicial no ponto de partida 3 |
| N150 X210 Y75 SPP=3 SON | Liga riscar comprimento máximo da trajetória para segmentação automática: 3mm |
| N140 X525 SPOF | Desliga riscar: posição inicial 4 |
| N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON | Liga riscar, máximo comprimento máximo do segmento: 3 mm |
| N180 G00 G90 Y300 SPOF | Desliga riscar |



Exemplo de programação 2

A segmentação automática deve ser utilizada para criar colunas e buracos. O comprimento máximo de segmento (valor SPP) é especificado em cada caso para efeitos de segmentação.



Programa

| | |
|--------------------------|---|
| N100 G90 X75 Y75 F60 PON | Posição no ponto inicial 1, liga puncionamento, puncionar um furo |
| N110 G91 Y125 SPP=25 | Comprimento máximo de segmentação para segmentação automática: 25 mm |
| N120 G90 X150 SPOF | Desliga puncionar, posicionar no ponto inicial 2 |
| N130 X375 SPP=45 PON | Liga puncionar, máximo comprimento de segmentos para segmentação automática: 45 mm |
| N140 X275 Y160 SPOF | Desliga puncionar, posicionar no ponto de parecida 3 |
| N150 X150 Y75 SPP=40 PON | Liga puncionar, o segmento de trajetória calculada de 37,79 mm é utilizado ao invés de 40mm programado no segmento da trajetória (3mm), |
| N160 G00 Y300 SPOF | Desliga puncionamento |



840 D
NCU 572
NCU 573



840Di

Notas

[illegible]

Funções adicionais

| | | |
|--------|---|--------|
| 13.1 | Funções de eixo AXNAME, SPI, ISAXIS | 13-428 |
| 13.2 | Aprendendo as características de compensação: QECLRNON, QECLRNOF | 13-429 |
| 13.3 | Fuso síncrono | 13-431 |
| 13.4 | EG: Câmbio eletrônico (SW 5 em diante)..... | 13-441 |
| 13.4.1 | Definir o câmbio eletrônico: EGDEF | 13-441 |
| 13.4.2 | Ativando o câmbio eletrônico..... | 13-443 |
| 13.4.3 | Desativando o câmbio eletrônico..... | 13-445 |
| 13.4.4 | Apaga a definição de um câmbio eletrônico..... | 13-446 |
| 13.4.5 | Avanço por rotação (G95)/câmbio eletrônico (SW 5.2)..... | 13-446 |
| 13.4.6 | EG no Power ON, RESET, troca de modo e pesquisa de bloco..... | 13-447 |
| 13.4.7 | Variáveis de sistema para câmbios eletrônicos..... | 13-447 |
| 13.5 | Expansão de parada e recuo (a partir da SW 5)..... | 13-447 |
| 13.5.1 | Reações independentes do acionamento..... | 13-448 |
| 13.5.2 | Fontes possíveis de disparo | 13-449 |
| 13.5.3 | Funções lógicas de disparo: Operação fonte/reação | 13-450 |
| 13.5.4 | Ativação | 13-450 |
| 13.5.5 | Operação como gerador/Backup de tensão DC | 13-451 |
| 13.5.6 | Parada independente do acionamento..... | 13-451 |
| 13.5.7 | Recuo independente do acionamento | 13-452 |
| 13.5.8 | Exemplo: Utilizando uma reação independente do acionamento..... | 13-453 |
| 13.6 | Conexão de comunicação (SW 5.2 em diante)..... | 13-455 |
| 13.7 | Recipiente de eixos (SW 5.2 em diante) | 13-458 |
| 13.8 | Tempo de execução de programa/contagem de peças (a partir da SW 5.2) | 13-461 |
| 13.8.1 | Tempo de execução do programa | 13-461 |
| 13.8.2 | Contagem de peças | 13-462 |

13.1 Funções de eixo AXNAME, SPI, ISAXIS

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

13.1 Funções de eixo AXNAME, SPI, ISAXIS



Programação

```
AXNAME( "Eixo" )
AX[AXNAME( "String" )]
SPI(Numero do fuso(número do eixo
geométrico))
```



Explicação dos comandos

| | |
|--------|---|
| AXNAME | Converte uma string (seqüência de caracteres) de entrada em um identificador de eixo. A string deve conter um nome válido de eixo. |
| SPI | Converte um número de fuso em identificador de eixo. O parâmetro transferido deve conter um número de fuso válido. |
| AX | Variável identificadora do eixo |
| ISAXIS | Verifica a existência do eixo geométrico. |



Função

AXNAME é utilizada, por exemplo, geralmente é utilizada em ciclos onde o nome do eixo não é conhecido (vide também a seção 13.10 “Funções string”).

SPI é utilizada, por exemplo, quando uma função de eixo é utilizada em um fuso, p.e. em um fuso sincronizado.

ISAXIS é utilizado em ciclos universais para garantir a existência do eixo geométrico específico, de forma que qualquer comando \$P_AXNX não seja abortado com uma mensagem de erro.



Exemplo de Programação

Movimentar um determinado eixo como eixo de faceamento.

| | |
|---------------------------------------|-----------------------------------|
| OVRA[AXNAME("Eixo transversal")]=10 | Eixo transversal |
| AX[AXNAME("Eixo transversal")]=50.2 | Posição final do eixo transversal |
| OVRA[SPI(1)]=70 | Override para o fuso 1 |
| IF ISAXIS(1) == FALSE GOTOF CONTINUE | Abscissa existe? |
| AX[\$P_AXN1]=100 | Movimenta abscissa |
| CONTINUE : | |

840D
NCU 571840D
NCU 572
NCU 573

810D



840Di

13.2 Aprendendo as características de compensação: QECLRNON, QECLRNOF



Explicação dos comandos

| | |
|---------------------------|--|
| QECLRNON (axis.1,...4) | Ativa a função “Aprendendo as características de compensação” |
| QECLRNOF | Desativa a função “Aprendendo as características de compensação” |

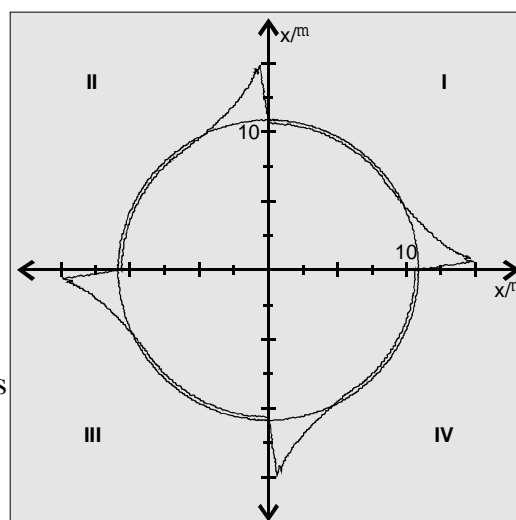


Função

A compensação de erro de quadrante (QEC) reduz os erros de contorno da direção do movimento devido à não linearidades mecânicas (p.e. atrito, folga no fuso de esferas ou torção).

Com base em uma rede neural, os dados otimizados podem ser adaptados pelo controle durante uma fase de avaliação, para que as características de compensação sejam determinadas automaticamente.

A função de avaliação (aprender) pode ser realizada simultaneamente para até quatro eixos.



Seqüência

Os movimentos dos eixos utilizados no processo de “aprender” são gerados com auxílio de um programa de usinagem. Os movimentos de aprendizado são gravados no programa sob a forma de um ciclo de “aprendizado”.

Teach in inicial

Os programas de NC contidos no disco de programas standard do PLC são utilizados nos movimentos de “aprendizado” e para atribuir valores às variáveis de sistema QEC durante a fase de ajuste do controle:

13.2 Aprendendo as carac.compensação: OECLRNON, OECLRNOF



840D
NCU 571



840D
NCU 572
NCU 573



810D



840Di

QECLRN.SPF

Ciclo de “aprendizado”

QECDAT.MPF

Programa NC exemplo para a programação das variáveis de sistema e os parâmetros para o ciclo de aprendizado

QECTEST.MPF

Programa NC exemplo para o ciclo de teste de formato

Aprendizado subsequente

As características de aprendizado podem ser otimizadas através do aprendizado subsequente. Os dados gravados na memória do usuário são utilizados como base para a otimização.

A otimização é realizada através de um programa de exemplo adaptado. Os parâmetros para o ciclo de aprendizado (p.e. QECLRN.SPF) podem também serem alterados para otimização

- Ativa o “Modo aprender” = 1
- Reduz a “Quantidade de passes de aprendizado” caso necessário
- Ativa o “Aprendizado modular” caso necessário e define as áreas limite.

Ativando o processo de aprendizado:

QECLRNON

O processo de aprendizado atual é ativado no programa de NC através do comando QECLRNON e da especificação dos eixos:

QECLRNON (X1, Y1, Z1, Q)

Somente caso este comando seja ativado e os quadrantes alterados.

Desativando o processo de aprendizagem :

QECLRNOF

Quando os movimentos de aprendizagem para os eixos desejados são finalizados, o processo de aprendizagem é desativado simultaneamente para todos os eixos com a instrução QECLRNOF.

840D
NCU 571840D
NCU 572
NCU 573

840Di

13.3 Fuso síncrono



Programação

COUPDEF (FS,LS,SR_{FS},SR_{LS}, comportamento na troca de bloco, acoplamento)
 COUPDEL (FS,LS)
 COUPRES (FS,LS)
 COUPON (FS,LS,PS_{FS})
 COUPOF (FS,LS,POS_{FS},POS_{LS})
 WAITC (FS, comportamento na troca de bloco, FS, comportamento na troca de bloco)



Explicação dos comandos

| | |
|---------|---|
| COUPDEF | Define/altera o acoplamento do usuário |
| COUPON | Ativa o acoplamento |
| COUPOF | Desativa o acoplamento |
| COUPRES | Reseta os parâmetros do acoplamento |
| COUPDEL | Apaga o acoplamento definido pelo usuário |
| WAITC | Espera pela condição de sincronismo |



Explicação dos parâmetros

| | |
|--|---|
| FS, LS | Nome do fuso principal e do fuso seguidor, especificados através do número do fuso: p.e. S2 |
| SR _{FS} , SR _{LS} | Parâmetro com a relação entre a velocidade do fuso principal e do fuso seguidor. Valor default = 1.0; especificação opcional do denominador |
| comportamento na troca de bloco: <ul style="list-style-type: none"> "NOC" "FINE" "COARSE" "IPOSTOP" | Método de troca de bloco; a troca de bloco é implementada através das seguintes formas: Imediata (default) Em resposta à "Sincronização fina (fine)" Em resposta à "Sincronização grossa (coarse)" Em resposta ao IPOSTOP (ou seja, após o sincronismo do valor desejado) |
| Acoplamento <ul style="list-style-type: none"> "DV" "AV" | Tipo de acoplamento: Acoplamento entre FS e LS Acoplamento através do valor desejado (default) Acoplamento através do valor atual |
| PS _{FS} | Ângulo de acoplamento entre o fuso principal e o seguidor |
| POS _{FS} , POS _{LS} | Desativação das posições entre os fusos seguidor e principal |

840D
NCU 571840D
NCU 572
NCU 573

840Di



Função

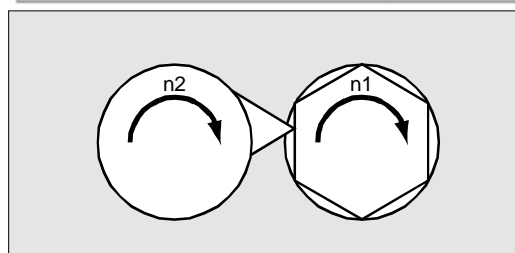
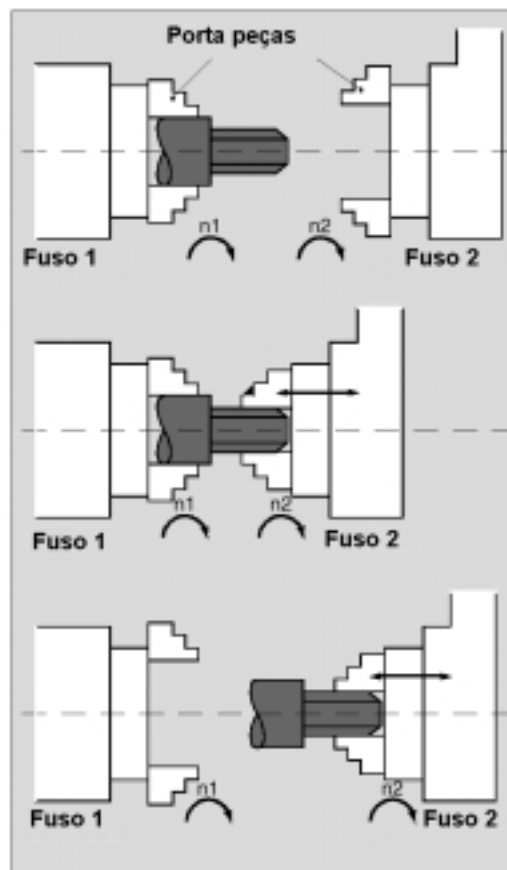
No modo síncrono, existe um fuso principal (LS) e um fuso seguidor (LS). Eles são chamados de **par de fusos síncrono**. O fuso seguidor segue os movimentos do fuso principal enquanto o acoplamento permanecer ativo (modo síncrono) de acordo com a relação especificada nos parâmetros.

Esta função habilita a tornos realizar a transferência de peças entre os fusos 1 e 2 durante o funcionamento, para, por exemplo, usinagem de acabamento. Isto evita perdas de tempo devido às acelerações.

A transferência da peça pode ser realizada através de:

- Sincronismo de velocidade ($n_{FS} = n_{LS}$)
- Sincronismo de posição ($\phi_{FS} = \phi_{LS}$)
- Sincronismo de posição com deslocamento angular
($\phi_{FS} = \phi_{LS} + \Delta\phi$)

Uma relação de velocidade k_U pode também ser especificada entre um fuso principal e um “fuso ferramenta”, para usinagem de cantos múltiplos (torneamento de polígonos).



O par de fusos sincronizados pode ser definido permanentemente para cada máquina em cada através de dados de máquina de canal, ou através do programa de usinagem CNC.

Até dois pares de fuso síncronos podem existir em cada canal de NC.

840D
NCU 571840D
NCU 572
NCU 573

840Di



Sequência

Definir o par de fusos síncronos: Opções

Definição de acoplamento fixo:

Os fusos principal e seguidor são definidos através de dados de máquina.

Através deste acoplamento, os eixos de usinagem definidos para os LS e FS (fusos principal/seguidor) não podem ser alterados através do programa de usinagem NC. O acoplamento, entretanto, pode ser parametrizado através do programa NC através de COUPDEF (desde que a proteção contra escrita não esteja ativada).

Acoplamento definido pelo usuário:

A instrução COUPDEF pode ser utilizada para criar um novo acoplamento e alterar os existentes através de programas NC. Caso uma nova relação de acoplamento deva ser definida, qualquer outra relação definida pelo usuário deve ser apagada com COUPDEL.

Definindo um novo acoplamento COUPDEF

A linha a seguir mostra parâmetros para a subrotina pré definida:

COUPDEF (FS,LS,SR_{FS},SR_{LS}, comportamento na troca de bloco, acoplamento)

Fusos principal e seguidor: LS e FS

Os nomes dos eixos FS e LS são utilizados somente para a identificação do acoplamento. Eles devem ser programados em cada instrução COUP. Os parâmetros adicionais podem ser definidos somente em caso de alterações (modal).

Exemplo:

N... COUPDEF (S2 , S1 , \ddot{U}_{FS} , \ddot{U}_{LS})

Significado:

S2 = Fuso seguidor, S1 = Fuso principal

840D
NCU 571840D
NCU 572
NCU 573

840Di

Posicionando o fuso seguidor: Opções

Quando o acoplamento de fusos síncrono estiver ativo, os fusos seguidores podem também serem posicionados dentro de uma faixa de $\pm 180^\circ$ independente do movimento iniciado pelo fuso principal.

Posicionamento SPOS

O fuso seguidor pode ser interpolado com SPOS=...

Favor consultar o Manual de programação “Fundamentos” para maiores informações a respeito de SPOS.

Exemplo:

```
N30 SPOS[2]=IC(-90)
```

FA, ACC, OVRA :

Velocidade, aceleração

A velocidade e a aceleração para os eixos seguidores podem ser programadas através de FA[SPI(Sn)] ou FA[Sn], ACC[SPI(Sn)] ou ACC[Sn] e OVRA[SPI(n)] ou OVRA[Sn] (Vide o Manual de programação, Fundamentos). "n" indica o número do fuso (1...n).

Comutação programável de bloco

WAITC pode ser utilizada para definir o comportamento da mudança de bloco com várias condições de sincronismo (grossa, fina (coarse, fine), IPOSTOP), p.e. após alterações nos parâmetros de acoplamento ou operações de posicionamento.

WAITC causa um atraso na inserção de novos blocos até que a condição de sincronismo apropriada seja preenchida, com isto permitindo um mais rápido sincronismo.

Caso não existam condições de sincronismo especificadas, então o comportamento de mudança de bloco programado/configurado para o acoplamento relevante será aplicado.

840D
NCU 571840D
NCU 572
NCU 573

840Di

Exemplos:

N200 WAITC

Espera as condições de sincronismo para todos os fusos escravos ativos sem a especificação destas condições.

N300 WAITC(S2, "FINE", S4, "COARSE")

Espera o sincronismo "Coarse" para os fusos escravos S2 e S4.

Relação de velocidade k_U

A relação de velocidade é definida através dos parâmetros para o FS (numerador) e LS (denominador).

Opções:

- Os fusos seguidor e principal giram na mesma velocidade ($n_{FS} = n_{LS}$; SR_T positiva)
- Giram na mesma direção ou em direções opostas (SR_T negativo) entre LS e FS
- Os fusos principais e seguidor giram a diferentes velocidades ($n_{FS} = k_U \cdot n_{LS}$; $k_U \neq 1$)

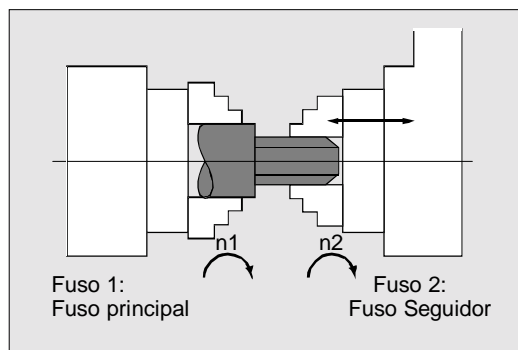
Aplicação: Torneamento de múltiplos lados

Exemplo:

N... COUPDEF(S2, S1, 1.0, 4.0)

Significado:

O fuso seguidor S2e o fuso principal S1 giram a uma relação de velocidade de 0.25 0.25.



- O numerador deve ser programado. Caso não seja, "1" é considerado como default.
- A relação de velocidades pode ser também alterada durante o movimento, enquanto o acoplamento estiver ativo.

840D
NCU 571840D
NCU 572
NCU 573

840Di

Comportamento na troca de bloco

As seguintes opções podem ser selecionadas durante a definição do acoplamento para determinar quando ocorrerá a troca de bloco:

"**NOC**" Imediatamente (default)
 "**FINE**" no "Sincronismo fino"
 "**COARSE**" no "Sincronismo grosso"
 "**IPOSTOP**" no IPOSTOP (ou seja, após sincronismo com o valor desejado)

É suficiente programar apenas os caracteres impressos em negrito para especificar o método de troca de bloco.

A troca de bloco é modal!

Tipo de acoplamento

"**DV**" Valor desejado como acoplamento entre FS e
 LS (default)
 "**AV**" Valor atual como acoplamento entre FS e
 LS

O acoplamento é modal.



Cuidado

O tipo de acoplamento pode ser alterado somente quando o acoplamento estiver desativado!

840D
NCU 571840D
NCU 572
NCU 573

840Di

Ativa modo sincronizado

- A mais rápida forma possível para ativação do acoplamento com qualquer ângulo referência entre LS e FS:

N ... COUPON (S2, S1)

- Ativação com deslocamento angular POS_{FS}
Posicionamento com acoplamento síncrono para peças de mesmo perfil.
 POS_{FS} se refere à posição 0° do fuso principal no sentido de rotação positivo.

Faixa de valor $-POS_{FS}$: $0^\circ \dots 359,999^\circ$:

COUPON (S2, S1, 30)

Você pode utilizar este método para alterar o ângulo de deslocamento mesmo quando o acoplamento já estiver ativo.

Desativando o modo síncrono, COUPOF

Existem 3 possíveis variantes:

- Para a mais rápida forma possível de acoplamento e habilitação imediata da mudança de bloco:

COUPOF (S2, S1)

- Após a que as posições de desativação tenham sido ultrapassadas, a troca de bloco não será habilitada até que a posição de desativação POS_{FS} tenha sido ultrapassada, e, quando apropriado, POS_{LS} tenham sido cruzadas.

Faixa de valores $0^\circ \dots 359,999^\circ$:

COUPOF (S2, S1, 150)

COUPOF (S2, S1, 150, 30)

840D
NCU 571840D
NCU 572
NCU 573

840Di

Apagando acoplamentos, COUPDEL

Um acoplamento de fuso síncrono definido pelo usuário deve ser apagado caso uma nova relação de acoplamento deva ser definida e todos os acoplamentos (1 e 2) já tenham sido definidos.

N ... COUPON (S2,S1)

SPI(2) = Fuso seguidor, SPI(1) = Fuso principal



Um acoplamento somente pode ser apagado caso tenha sido anteriormente desativado (COUPOF).



Um acoplamento permanentemente configurado não pode ser apagado através de COUPDEL.

Reseta parâmetros de acoplamento , COUPRES

A instrução "COUPRES" é utilizada para:

- Ativar os parâmetros gravados em dados de máquina e dados setting (acoplamento permanente) e
- Ativar as pré seleções (definições de acoplamento do usuário)

Os parâmetros programados com COUPDEF (incluindo fatores de transformação) são apagados.

N ... COUPRES (S2,S1)

S2 = Fuso seguidor, S1 = fuso mestre

840D
NCU 571840D
NCU 572
NCU 573

840Di



Variáveis de sistema

Estado atual de acoplamento do fuso seguidor

O estado atual do fuso seguidor pode ser lido através pelo programa NC através da seguinte variável axial de sistema:

`$AA_COUP_ACT[FS]`

FS = nome de eixo do fuso seguidor com o número de fuso, p.e. S2.

O valor lido possui o seguinte significado para um fuso seguidor:

0: Acoplamento não ativo

4: Fuso síncrono com acoplamento ativo

Deslocamento angular atual

O valor desejado do deslocamento de posição atual de FS para LS pode ser lido pelo programa NC através da seguinte variável axial de sistema:

`$AA_COUP_OFFS[S2]`

O valor atual de deslocamento de posição pode ser lido com:

`$VA_COUP_OFFS[S2]`

FS = nome de eixo do fuso seguidor com o número de fuso, p.e. S2.



Quando o controle for desabilitado e subsequentemente re habilitado durante um acoplamento ativo e no modo seguidor, o deslocamento de posição quando o controle for reabilitado será diferente do valor original programado. Neste caso, o novo deslocamento de posição pode ser lido e, se necessário, corrigido através do programa NC.

840D
NCU 571840D
NCU 572
NCU 573

840Di

**Exemplo de programação**

Trabalhando com fusos mestre e escravo.

| | |
|---|--|
| | ;Fuso principal = fuso mestre = fuso 1 |
| | ;Fuso escravo = fuso 2 |
| N05 M3 S3000 M2=4 S2=500 | ;O fuso mestre gira à 3000 rpm, o escravo a 500 rpm |
| N10 COUPDEF (S2, S1, 1, 1, "NOC", "Dv") | ;Definição do acoplamento, pode também ser configurado |
| ... | |
| N70 SPCON | ;Inclui o fuso mestre no controle de posição (acopla valor desejado) |
| N75 SPCON(2) | ;Inclui o fuso escravo no controle de posição |
| N80 COUPON (S2, S1, 45) | ;Acoplamento durante o movimento com um deslocamento de posição de 45° |
| ... | |
| N200 FA [S2] = 100 | ;Velocidade de posicionamento = 100°/min |
| N205 SPOS[2] = IC(-90) | ;Movimento com sobreposição de 90° na direção negativa |
| N210 WAITC(S2, "Fine") | ;Espera por sincronismo "fino" |
| N212 G1 X... Y... F... | ;Usinagem |
| ... | |
| N215 SPOS[2] = IC(180) | ;Movimento com 180° de sobreposição na direção positiva |
| N220 G4 S50 | ;Tempo de permanência = 50 voltas do fuso mestre |
| N225 FA [S2] = 0 | ;Ativa a velocidade configurada (MD) |
| N230 SPOS[2]=IC(-7200) | ;20 rpm. Com a velocidade de projeto na direção negativa |
| ... | |
| N350 COUPOF (S2, S1) | ;Desacopla em movimento, S=S2=3000 |
| N355 SPOSA[2] = 0 | ;Para fuso escravo em zero graus |
| N360 G0 X0 Y0 | |
| N365 WAITS(2) | ;Espera pelo fuso 2 |
| N370 M5 | ;Para o fuso escravo |
| N375 M30 | |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.4 EG: Câmbio eletrônico (SW 5 em diante)



Introdução

A função “Câmbio eletrônico” permite que o movimento de um **eixo seguidor** seja controlado de acordo com o movimento linear programado de até cinco **eixos principais**. As relações entre os eixos principais e o eixo seguidor são definidas por um fator de acoplamento para cada eixo principal.

Os movimentos seguintes do eixo é calculado através da soma dos movimentos individuais dos eixos principais multiplicado pelos seus respectivos fatores de acoplamento.

Durante a ativação de um agrupamento de eixo EG, o eixo seguidor pode ser sincronizado de acordo com uma posição definida.

Um grupo de câmbio pode ser

- Definido,
- Ativado
- Desativado e
- Apagado

A partir de um programa de usinagem.

O movimento do eixo seguidor pode ser derivado opcionalmente de

- Valores desejados dos eixos principais, bem como
- Dos valores atuais dos atuais dos eixos principais

13.4.1 Definir o câmbio eletrônico: EGDEF



Função

Um agrupamento de eixos EG é definido através da especificação do eixo seguidor e um mínimo de um e um máximo de cinco eixos principais e do respectivo tipo de acoplamento:

EGDEF (eixo seguidor, eixo principal 1, tipo de acoplamento 1, eixo principal 2, tipo de acoplamento 2, ...)



Explicação

Eixo seguidor

Eixo influenciado pelo eixo principal

Eixo principal 1, ... eixo principal 5 Eixos que influenciam o eixo seguidor

13.4 EG: Câmbio eletrônico (SW 5 em diante)

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Acoplamento tipo 1, ...
acoplamento tipo 5

Eixo seguidor influenciado por:

0: valor atual

1: valor desejado
do respectivo eixo principal

Programação

EGDEF(C, B, 1, Z, 1, Y, 1)

B, Z, Y influenciam C através do valor
desejado

O tipo de acoplamento não necessita ser idêntico para todos os eixos principais, sendo então especificado individualmente para cada eixo principal.

Os fatores de acoplamento são presetados com 0 na definição do acoplamento EG.

Pré requisito para uma definição de agrupamento EG:

Um eixo a ser acoplado não deve possuir nenhuma outra definição com relação ao eixo seguidor (caso exista, qualquer que seja a definição presente deve ser apagada com EGDEL).

Nota

Um EGDEF dispara uma parada de pré processamento.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.4.2 Ativando o câmbio eletrônico

Existem duas variantes para o comando de ativação:

- **Variante 1:**

O agrupamento de eixos EG é ativado de forma seletiva **sem sincronismo** com:

EGON(FA, "Modo de troca de bloco", LA1, Z1, N1, LA2, Z2, N2, ..LA5, Z5, N5.)



Explicação

| FA | Eixo seguidor |
|------------------------|--|
| Tipo de troca de bloco | Os seguintes modos podem ser utilizados: "NOC" Troca imediata de bloco "FINE" Troca de bloco ocorre na "Sincronização fina" "COARSE" troca de bloco ocorre na "Sincronização grossa" "IPOSTOP" Troca de bloco ocorre na "Sincronização com o valor desejado" |
| LA1, ... LA5 | Eixo principal |
| Z1, ... Z5 | Numerador para o fator de acoplamento i |
| N1, ... N5 | Denominador para o fator de acoplamento i |
| | Fator de acoplamento i = numerador/denominador |

Somente é permitido programar os eixos principais que tenham sido previamente especificados com EGDEF. Ao menos um eixo principal deve ser programado.

As posições dos eixos principais e do eixo seguidor no instante de ativação são gravadas como "posições de sincronismo". As "posições de sincronismo" podem ser lidas através da variável de sistema \$AA_EG_SYN.

- **Variante 2:**

O agrupamento EG é ativado de forma seletiva **com** sincronismo através de:

EGONSYN(FA, "Tipo de troca de bloco", SynPosFA[, LAi, SynPosLAI, Zi, Ni])

13.4 EG: Câmbio eletrônico (SW 5 em diante)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Explicação**

| | |
|-----------------------------|--|
| FA | Eixo seguidor: |
| Tipo de troca de bloco | Os seguintes modos podem ser utilizados: |
| | "NOC" imediato |
| | "FINE" com |
| | "Sincronização fina" |
| | "COARSE" com "Sincronização grossa" |
| | "IPOSTOP" no sincronismo com o valor desejado |
| [, LAi, SynPosLAi, Zi, Ni] | (não escreva os colchetes) |
| LA1, ... LA5 | min. 1, max. 5 seqüências de: |
| SynPosLAi | Eixos principais |
| Z1, ... Z5 | Posição síncrona para o eixo i |
| N1, ... N5 | Numerador para o fator de acoplamento i |
| | Denominador para o fator de acoplamento i |
| | Fator de acoplamento i = Numerador/denominador |

Somente é permitida a programação de eixos principais que tenham sido previamente especificados com EGDEF.

Através das "posições síncronas" programadas para o eixo seguidor (SynPosFA) e para os eixos principais (SynPosLA), posições são definidas nas quais o grupo de acoplamento é validado como síncrono. Caso o câmbio eletrônico não esteja em estado sincronizado quando ativado, o eixo seguidor será movimentado até a posição de sincronismo.

Caso eixos em módulo estejam contidos no grupo de acoplamento, seus valores de posição são reduzidos à módulos. Isto garante que a próxima posição sincronizada seja movimentada (também chamada de *sincronismo relativo*: p.e. para o próximo intervalo de ferramenta).

A posição sincronizada será somente aproximada caso o sinal "Habilitação para override de eixo seguidor" DB(30+nr. Eixo), dbb26 bit 4 esteja ativado para o eixo seguidor. Caso não esteja, o programa sofrerá uma parada no bloco EGONSYN

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

e o aviso 16771 será emitido até que o sinal especificado seja ligado..

13.4.3 Desativando o câmbio eletrônico

Existem três diferentes formas para ativar um eixo agrupamento de eixo EG.

Variante 1:

EGOFS(eixo seguidor)

O câmbio eletrônico será ativado. O eixo seguidor será desacelerado até sua parada.

A chamada causa parada no pré processamento.

Variante 2:

EGOFS(eixo seguidor, eixo principal 1, ... eixo principal 5)

Este valor de parâmetro de comando torna possível remover de forma **seletiva** o controle individual que os eixos principais possuem sobre o eixo seguidor.

Ao menos um eixo principal deve ser especificado.

A influencia do eixo principal sobre o eixo seguidor será desabilitada de forma seletiva. A chamada dispara uma parada no pré processamento.

Caso eixos principais estejam ativos, o eixo seguidor irá continuar a operar sob seu controle.

Caso todos os eixos principais tenham sido desabilitados desta forma, o eixo seguidor será desacelerado até a parada de movimento.

Variante 3:

EGOFC(fuso seguidor)

Um câmbio eletrônico é desativado. O fuso seguidor continua a operar na velocidade atual do instante da desativação.

A chamada dispara uma parada no pré processamento.



Nota

Esta função é válida somente para fusos

13.4 EG: Câmbio eletrônico (SW 5 em diante)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.4.4 Apaga a definição de um câmbio eletrônico

Para apagar uma definição de agrupamento de eixo EG, este deve ser inicialmente desativado conforme descrito na seção anterior.

EGDEL(eixo seguidor)

A definição de acoplamento de agrupamento de eixos é apagada. Agrupamentos adicionais podem ser definidos através de EGDEF até que a quantidade máxima de eixos agrupados. A instrução causa uma parada no pré processamento.

13.4.5 Avanço por rotação (G95)/câmbio eletrônico (SW 5.2)

A partir da SW 5, através do comando FPR(), é também possível definir o eixo seguidor de um câmbio eletrônico como o eixo determinante do avanço por rotação. O seguinte é aplicado neste caso:

- O avanço é dependente da velocidade desejada do eixo seguidor do câmbio eletrônico.
- A velocidade desejada é calculada a partir da velocidade dos fusos principais e dos eixos principais módulo (aqueles que não são eixos de trajetória) e seus fatores de acoplamento definidos.
- As velocidades dos eixos principais não módulo e movimentos de sobreposição do eixo seguidor não são considerados.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.4.6 EG no Power ON, RESET, troca de modo e pesquisa de bloco

Após a ligação do comando (Power ON) **não** há acoplamentos ativos.

Os acoplamentos ativos são retidos após reset ou troca de modo.

Com a função procura de bloco, os comandos de comutação, apagamento e definição de faixa de engrenamento não são executados e nem retidos, em vez disso eles são saltados

13.4.7 Variáveis de sistema para câmbios eletrônicos



Através das variáveis de sistema dos câmbios eletrônicos, o programa pode determinar o estado atual de um acoplamento de eixos EG e reagir conforme a necessidade.



Informações adicionais

As variáveis de sistema para o câmbio eletrônico encontram-se listadas no Anexo. São caracterizadas pelos nomes que começam com os caracteres:

\$AA_EG_ . . .

ou

\$VA_EG_ . . .

13.5 Expansão de parada e recuo (a partir da SW 5)



Função

A função "Expansão de parada e recuo" ESR fornece um meio de executar um retorno de forma flexível de acordo com as causas dos erros, de forma a evitar danos à peça.

A "Expansão de parada e recuo" fornece 3 tipos de reação:

- **"Parada expandida"** (Independente do acionamento, SW 5), é uma parada temporizada.
- **"Recuo"** (Independente do acionamento)
significa "escapar" do plano de usinagem para uma posição segura de recuo. Isto evita riscos de colisão entre a ferramenta e a peça.
- **"Operação de geração"** (independente do acionamento)
Para os casos onde a energia do link DC não seja o suficiente para o recuo, uma operação de geração é possível.
Como uma operação independente do acionamento, ela fornece

13.5 Expansão de parada e recuo (a partir da SW 5)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

uma conexão com a fonte DC do acionamento, que possui a potência necessária para executar uma “parada” e um “recuo” em caso de perda de energia ou causas similares.

Todas as reações podem ser utilizadas independentemente umas das outras.



Para informações adicionais consulte
/FB/ M 3, Acoplamento de eixos e ESR

13.5.1 Reações independentes do acionamento



Função

As reações independentes aos acionamentos são definidas por cada eixo, e caso ativadas, cada processo de parada/recuo é solicitado de forma independente. Não existe acoplamento interpolado em eixos acoplados à trajetória durante a parada/recuo, a referência ao eixos é controlada através de temporizadores.

Durante e após a execução de reações independentes do acionamento, o respectivo acionamento não mais obedece às liberações do NC ou comandos de movimentação. Para retornar às condições normais será necessário desligar e religar o comando (Power OFF/ON). O alarme "26110: Parada/recuo independente do acionamento" sinaliza esta condição.

Operação como gerador

Operação de geração

- Configuração: através de dados de máquina
- Habilitação: Através da variável de sistema \$AA_ESR_ENABLE
- Ativação: Em função da programação em dados de máquina, quando a tensão DC da fonte dos acionamentos estiver abaixo de um determinado valor.

Parada (independente do acionamento)

Uma parada independente do acionamento é

- Configuração: através de dados de máquina, bem como a especificação do atraso;
- Habilitada (\$AA_ESR_ENABLE) e
- Disparada: variável de sistema \$AN_ESR_TRIGGER.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Recuo (independente do acionamento)

Um recuo independente do acionamento é

- Configuração: o tempo de atraso e a velocidade de recuo são configurados através de dados de máquina, vide o exemplo “Utilizando o recuo independente do acionamento” no fim deste capítulo,
- Habilitação: através da variável de sistema \$AA_ESR_ENABLE
- Disparada: através da variável de sistema \$AN_ESR_TRIGGER.

13.5.2 Fontes possíveis de disparo



Função

Existem as seguintes fontes de erro para a “Expansão de parada e recuo”:

- Fontes gerais (globais/externas ou específicas do canal):
 - Entradas digitais (p.e. nos módulos NCU ou blocos de terminação) ou através do mapeamento das saídas digitais internas ao controle (\$A_IN, \$A_OUT)
 - Estado do canal (\$AC_STAT)
 - Sinais VDI (\$A_DBB)
 - Mensagens de grupo a partir de uma quantidade de alarmes (\$AC_ALARM_STAT)
- Fontes axiais:
 - Nível de disparo de recuo de emergência em eixos seguidores (sincronismo do acoplamento eletrônico, \$VA_EG_SYNCDIFF[eixo seguidor])
 - Acionamento: Baixo nível de tensão DC na fonte, \$AA_ESR_STAT[eixo]
 - Acionamento: Gerador de nível mínimo de velocidade (não existe mais energia de regeneração rotativa), \$AA_ESR_STAT[eixo].

13.5 Expansão de parada e recuo (a partir da SW 5)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

13.5.3 Funções lógicas de disparo: Operação fonte/reação



Função

As possibilidades de funções lógicas de disparo flexíveis em ações síncronas estáticas são utilizadas para disparar reações específicas de acordo com as fontes.

A instrução possui várias opções para monitorar todas as fontes relevantes através de ações síncronas estáticas. Os usuários podem avaliar as variáveis de sistema fonte como um todo ou através de mascaramento binário e disparar as reações apropriadas. As ações síncronas estáticas encontram-se ativas em todos os modos de operação.

Para uma descrição mais detalhada da forma de uso das ações síncronas, favor consultar:



Referências: /FBSY/ descrição das ações síncronas

13.5.4 Ativação



Funções de habilitação

\$AA_ESR_ENABLE

A operação de geração, parada e funções de recuo são habilitadas através da programação de sinais de controle associados (\$AA_ESR_ENABLE). Este sinal de controle pode ser alterado através de ações síncronas.

Funções de disparo (disparo geral de todos os eixos habilitados)

\$AN_ESR_TRIGGER

- A operação de geração encontra-se “automaticamente” ativa no acionamento quando um baixo nível de tensão DC da fonte for detectado.
- A parada e/ou recuo independente do acionamento são ativadas quando falhas de comunicação (entre o acionamento e o NC) são detectadas, bem como quando um baixo nível de tensão de alimentação DC for detectado (desde que configurado e habilitado).

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

- Uma parada e/ou recuo independente do acionamento pode também ser disparada através do NC através da programação do sinal de controle "\$AN_ESR_TRIGGER" (comando emitido para todos os acionamentos).
- Recuo controlado pelo NC através de LIFT_FAST

13.5.5 Operação como gerador/Backup de tensão DC



Função

Através da configuração de dados de máquina nos acionamentos e da programação adequada através de ações síncronas (\$AA_ESR_ENABLE), quedas temporárias no nível de tensão DC do barramento de alimentação podem ser compensadas. O tempo que o sistema pode ser alimentado depende de quanta energia gerada é utilizada como back up no barramento DC, bem como da quantidade de energia necessária para manter os eixos em movimento (back up do barramento DC e monitoração para o gerador de velocidade limite). Quando o valor cair abaixo de um limite mínimo no barramento DC, o eixo/fuso relacionado é comutado do modo de controle de posição ou velocidade para a operação como gerador. A desaceleração (valor desejado de velocidade=0) causa a geração de energia no barramento DC.



Para maiores informações vide /FB/ M 3, Movimento acoplado e valor principal de acoplamento

13.5.6 Parada independente do acionamento



Função

Os acionamentos de um grupo previamente acoplado podem ser parados com atraso, mantendo-se uma diferença mínima entre eles, caso o controle não consiga manter esta diferença. A parada independente do acionamento é configurada e habilitada através de dados de máquina (tempo de atraso T1 em MD), e, no programa NC, habilitado através da variável de sistema \$AA_ESR_ENABLE, sendo a parada

13.5 Expansão de parada e recuo (a partir da SW 5)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



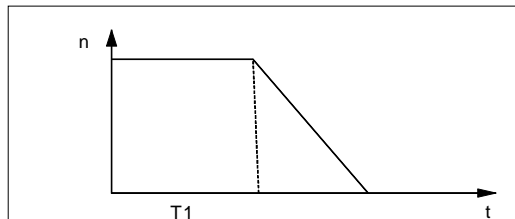
840Di

iniciada através de \$AN_ESR_TRIGGER.

Reações

Durante o intervalo de tempo T1 o valor desejado de velocidade ativo durante a ocorrência do erro permanece ativo. Esta é uma tentativa de se manter o movimento que se encontrava ativo antes da falha até que o contato físico seja anulado ou o movimento de recuo iniciado em outros acionamentos seja completado. Isto pode ser necessário para todos os acionamentos principais/seguidores ou para aqueles acoplados em um agrupamento.

Após o tempo T1, todos os eixos terão seus valores nominais levados à zero e parados com o limite de corrente, e os pulsos cancelados quando a velocidade zero for atingida ou quando o tempo for ultrapassado (dados de máquina de acionamento).



13.5.7 Recuo independente do acionamento



Função

Os eixos com acionamentos digitais 611D podem (caso configurados e habilitados) executar movimentos de recuo independentes

- Durante uma falha no controle (Detecção do sinal de vida)
- Caso a tensão no barramento DC atinja um limite abaixo de um nível de disparo de alarme
- Caso disparado pela variável \$AN_ESR_TRIGGER.

O movimento de recuo é executado de forma independente pelo acionamento 611D.

Uma vez que a fase de recuo seja iniciada, o acionamento mantém de forma independente suas habilitações em valores previamente válidos.

Para maiores informações vide /FB/ M 3, Movimento acoplado e valor principal de acoplamento



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.5.8 Exemplo: Utilizando uma reação independente do acionamento

Exemplo de configuração

- O eixo A deve operar como acionamento gerador,
- O eixo X deve recuar em 10mm com a máxima velocidade na ocorrência de um erro e
- Os eixos Y e Z devem ser parados com um atraso de 100ms, de forma que o eixo de recuo tenha tempo de cancelar o acoplamento mecânico.



Seqüência

1. Ativar as opções “Parada e recuo externos” e “Ações independentes do modo” (inclui “Ações síncronas IDS...”).
2. Definição da função:
`$MA_ESR_REACTION[X]=11,`
`$MA_ESR_REACTION[Y]=12,`
`$MA_ESR_REACTION[Z]=12,`
`$MA_ESR_REACTION[A]=10;`
3. Configuração do acionamento:
`MD1639 RETRACT_SPEED[X] =400000H` direção em posição (Vel.máxima),
`=FFC00000H` em direção negativa,
`D1638 RETRACT_TIME[X] =10ms` (tempo de recuo),
`MD1637 GEN_STOP_DELAY[Y]=100ms,`
`MD1637 GEN_STOP_DELAY[Z]=100ms,`
`MD1635 GEN_AXIS_MIN_SPEED[A] =velocidade mínima do gerador`
(rpm).
4. Habilitação da função (a partir do programa ou ações síncronas):
`$AA_ESR_ENABLE[X]=1, $AA_ESR_ENABLE[Y]=1,`
`$AA_ESR_ENABLE[Z]=1, $AA_ESR_ENABLE[A]=1`
5. Obter a operação de geração na rotação atual (p.e. com fuso em operação M03 S1000)
6. Formular a condição de disparo através de uma ação síncrona, p.e.:
 - Dependendo de intervenção do eixo gerador:
`IDS=01 WHENEVER $AA_ESR_STAT[A]>0 DO`
`$AN_ESR_TRIGGER=1`
 - E/ou em função de alarmes que disparem o modo de cancelamento da supervisão de posição (bit13=2000H):
`IDS=02 WHENEVER ($AC_ALARM_STAT B_AND 'H2000')>0`
`DO $AN_ESR_TRIGGER=1`
 - E dependendo da monitoração de (caso, p.e. o eixo Y seja definido como eixo seguidor EU e a distância máxima de sincronismo para 100 µm):

13.5 Expansão de parada e recuo (a partir da SW 5)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

```
IDS=03 WHENEVER ABS($VA_EG_SYNCDIFF[Y])>0.1
DO $AN_ESR_TRIGGER=1
```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.6 Conexão de comunicação (SW 5.2 em diante)



Função

Com a conexão NCU, que pode conectar várias NCUS de uma instalação, é utilizado em configurações com sistemas distribuídos. Quando são necessários vários eixos e canais, p.e. em máquinas com revólveres, multi fuso, capacidade de cálculo, opções de configuração e áreas de memória podem atingir seus limites caso seja utilizada apenas uma NCU. Várias NCUs podem ser conectadas em rede através de um módulo de conexão entre NCUs, que possibilita uma solução aberta e ampliável, que pode ser adequada às necessidades deste tipos de máquinas ferramenta. O módulo de conexão entre NCUs (hardware) oferece um canal de comunicação em alta velocidade entre NCUs. .



As opções que oferecem estes recursos devem ser pedidas separadamente.



Função

Várias NCUs conectadas através de módulos de conexão podem possuir direitos de acesso para leitura/escrita em uma região de memória global, através das variáveis de sistema descritas a seguir.

- Cada NCU conectada através de um modulo de conexão pode utilizar **variáveis de conexão globais**. Estas variáveis de conexão são endereçadas da mesma forma por todas as NCUs conectadas.
- As variáveis de conexão podem ser programadas como variáveis de sistema.
Como regra geral, o fabricante da máquina define e documenta o significado destas variáveis.
- Algumas aplicações para variáveis de conexão NC :
 - Estado global da máquina
 - Fixação da peça fixada/solta
 - Etc.
- Dados de volume relativo pequeno
- Velocidade de transferência muito alta, entretanto: uso previsto para aplicações criticas em se tratando de tempo.
- Estas variáveis de sistema podem ser acessadas a partir do **programa de usinagem** a através de **ações síncronas**. O tamanho da área de memória para as variáveis de sistema

13.6 Conexão de comunicação (SW 5.2 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

NCU globais é configuráveis.

Quando um valor é escrito em uma variável de sistema global, ele pode ser lido por todas as NCUs conectadas após um ciclo de interpolação.

As variáveis de conexão são **dados de sistema globais** que podem ser endereçados por qualquer NCU conectada como **variáveis de sistema**. O

- **conteúdo** destas variáveis,
 - seu **tipo de dado**,
 - **uso**, e
 - posição (**índice de acesso**) na memória de conexão
- são definidos pelo usuário (neste caso, geralmente, o fabricante da máquina).

As variáveis de conexão são gravadas na memória de conexão. Após a ligação dos equipamentos, esta memória é inicializada com zero.

As seguintes variáveis de conexão podem ser acessadas dentro da memória de conexão:

- INT \$A_DLB[i] ; Byte de dados (8 bits)
- INT \$A_DLW[i] ; Palavra de dados (16 bits)
- INT \$A_DLD[i] ; Palavra de dados dupla (32 bits)
- REAL \$A_DLR[i] ; dado tipo real (64 bits)

De acordo com o tipo em questão, 1, 2, 4 ou 8 bytes são endereçados quando as variáveis de conexão são lidas/escritas. O índice I define o início da respectiva variável em relação ao início da memória de conexão configurada. O índice é contado de 0 para cima.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Faixa de valores

Os diferentes tipos de dados possuem diferentes faixas de valores:

BYTE: 0 to 255

WORD: -32768 to 32767

DWORD: -2147483648 to 2147483647

REAL: -4.19e-308 to 4.19e-307



Os diversos tipos de aplicação da NCU com acesso compartilhado à memória de conexão **ao mesmo tempo** devem utilizar a memória de conexão de um **modo uniforme**. Quando o processo é completamente separado no tempo, a memória de conexão pode ser ocupada de forma diferente.



Atenção

O processo de escrita em uma variável de conexão é somente completado quando a informação escrita estiver disponível também em todas as outras NCUS. Para este processo são necessários aproximadamente dois ciclos de interpolação. A escrita local na memória de conexão é atrasada no mesmo valor de tempo para propósitos de consistência.



Para maiores informações, favor consultar o manual Descrição de funções B3 (SW5).



Exemplo de Programação

`$A_DLB[5]=21`

O 5º byte na memória de conexão compartilhada é carregado com o valor 21.

13.7 Recipiente de eixos (SW 5.2 em diante)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.7 Recipiente de eixos (SW 5.2 em diante)**Função**

Em máquinas tipo revólver ou multi fuso os eixos que seguram a peça se movimentam de uma estação de usinagem para outra.

Como as estações de usinagem são controladas por canais de NCU diferentes, na troca de estação/posição os eixos que seguram a peça devem ser redefinidos a um canal NCU apropriado. O **recipiente de eixos** existe para este propósito.

Somente um eixo/fuso de fixação de peça pode ser ativado em qualquer instante na estação de usinagem local.

O recipiente de eixos junta todas as possíveis conexões com todos os eixos/fusos de fixação, dos quais somente **um é sempre ativado** para a estação de usinagem.

São possíveis as seguintes definições através do recipiente de eixos:

- Eixos locais e/ou
- Eixos conectados (vide manual de Programação básica)

Os eixos disponíveis definidos no recipiente de eixos podem ser alterados através da parâmetros.

A comutação pode ser disparada a partir do **programa NC**.

Os recipientes de eixos com eixos que possuem outros eixos conectados à eles são uma ferramenta válida por todas as NCUs (global) e é coordenada pelo controle.

É também possível possuir recipientes de eixos nos quais somente eixos locais sejam gerenciados.



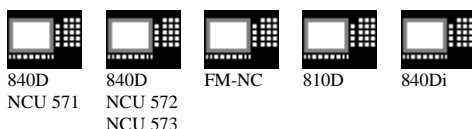
Informações mais detalhadas sobre a configuração de eixos para os recipientes podem ser encontradas em /FB/, B3 (SW 5.2)

Os valores programados no recipiente de eixos podem ser incrementados através do parâmetro n pelo do programa:

**Programação**

AXCTSWE (CT1, CT 2, ...)
AXCTSWED(CT1, CT 2, ...)

Habilita comutar o recipiente de eixos
(AXIS CONTAINER SWITCH ENABLE)
Habilitação direta da comutação do recipiente de eixos
AXIS CONTAINER SWITCH ENABLE DIRECT



Explicação

CT1, CT 2 ... ou

p.e. A_CONT1

Quantidade de recipientes de eixos cujos conteúdos são comutados ou

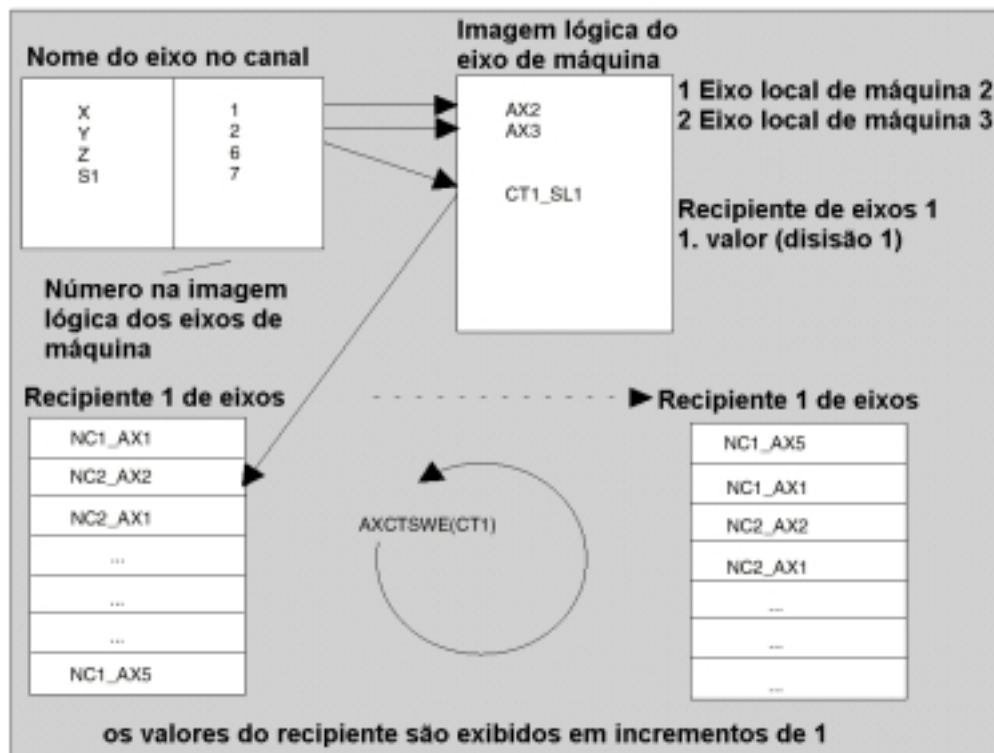
Nomes individuais dos eixos do recipiente de eixos definido através de dados de máquina.



Função

AXCTSWE ()

Cada canal cujos eixos estão contidos no recipiente especificado enviam uma **habilitação para rotação do recipiente**, caso a unidade tenha encerrado a usinagem naquela posição/estação. Uma vez que o controle receba a habilitação de **todos** os canais para os eixos do recipiente, este é rotacionado com o incremento especificado em dados setting (SD).



No exemplo anterior após o recipiente Ter sido rotacionado em 1, o eixo AX5 da NCU1 será associado ao eixo Z daquele canal, ao invés do

13.7 Recipiente de eixos (SW 5.2 em diante)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

eixo AX1 da NCU1.

A variante de comando AXCTSWED(CT1, ...) pode ser utilizada para simplificar a colocação em funcionamento. Sob efeito somente do canal ativo, o recipiente de eixos roda em torno de um incremento gravado em dados setting (SD). Esta chamada pode ser utilizada somente se os outros canais que possuam eixos no recipiente estejam em estado **RESET** (sem programa em andamento).



Após uma rotação do recipiente, **todos** os canais de **NCUs** que possuam referências ao recipiente rotacionado, através da imagem lógica dos eixos, serão afetados pelas novas relações entre os eixos.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

13.8 Tempo de execução de programa/contagem de peças (a partir da SW 5.2)



Função

Informações sobre o tempo de execução de programas e sobre contagem de peças são fornecidas para auxiliar o pessoal que trabalha com a máquina.

Esta informação é especificada nos respectivos dados de máquina e podem ser editados como variáveis de sistema através do programa NC ou PLC. Esta informação encontra-se também disponível para a MMC na interface de operação.

13.8.1 Tempo de execução do programa



Função

Através desta função, temporizadores são fornecidos como variáveis de sistema, que podem ser utilizadas para monitorar os processos tecnológicos.

Estes temporizadores podem ser somente lidos. Eles podem ser acessados a qualquer instante pela MMC no modo leitura.



Explicação

Os seguintes dois temporizadores são definidos como variáveis de sistema específicas NCK e encontram-se sempre ativos.

13.8 Tempo de exec.programa/contagem de peças (a partir da SW 5.2)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

| | |
|-----------------|--|
| \$AN_SETUP_TIME | Tempo em minutos desde a última programação; é resetado com SETUP |
|-----------------|--|

| | |
|-------------------|---|
| \$AN_POWERON_TIME | Tempo em minutos desde a última ligação do comando; é resetado com POWERON (nova ligação do comando) |
|-------------------|---|

Os seguintes temporizadores são definidos como variáveis de sistema específicas para o canal, e podem ser ativadas através de dados de máquina.

| | |
|---------------------|--|
| \$AC_OPERATING_TIME | Tempo total de execução em segundos dos programas NC no modo automático. |
|---------------------|--|

| | |
|-----------------|--|
| \$AC_CYCLE_TIME | Tempo de execução em segundos do programa NC selecionado |
|-----------------|--|

| | |
|-------------------|---|
| \$AC_CUTTING_TIME | Tempo de operação da ferramenta em segundos |
|-------------------|---|

| | |
|--------------------|---|
| \$MC_RUNTIMER_MODE | Tempo de operação da ferramenta em segundos |
|--------------------|---|

Todos os temporizadores são resetados com valores default quando o controle é energizado, e podem ser lidos independente de sua ativação.

Exemplo de Programação

1. Ativação da medição do tempo de execução do programa NC ativo; desativa a medição de tempo caso o modo avanço de teste (dry run) e teste de programas estejam ativos:

| | | |
|------------------------|---|------|
| \$MC_PROCESSTIMER_MODE | = | 'H2' |
|------------------------|---|------|

2. Ativa a medição do tempo de operação da ferramenta; mede inclusive caso o avanço de teste (dry run) e teste de programa estejam ativos:

| | | |
|------------------------|---|-------|
| \$MC_PROCESSTIMER_MODE | = | 'H34' |
|------------------------|---|-------|

3. Ativa a medição do tempo se usinagem total e tempo de operação da ferramenta; medição inclusive durante o teste do programa:

| | | |
|------------------------|---|-------|
| \$MC_PROCESSTIMER_MODE | = | 'H25' |
|------------------------|---|-------|

13.8.2 Contagem de peças**Função**

A função “contador de peças” pode ser utilizada para preparar contadores, p.e. para contagem interna de peças no controle. Estes contadores existem como variáveis de sistema específicas do canal com acesso de leitura/escrita e operam na faixa entre 0 e 999 999 999.

Dados de máquina podem ser utilizados para controlar a ativação do contador, temporizar o reset de contadores e

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

algoritmo de contagem.



Explicação

Os seguintes contadores são fornecidos:

| | |
|---------------------|---|
| \$AC_REQUIRED_PARTS | Quantidade de peças necessárias. Neste contador você pode definir a quantidade de peças com a qual o valor atual do contador \$AC_ACTUAL_PARTS será resetado a zero. Dados de máquina podem ser utilizados para configurar a geração do aviso “Quantidade de peças desejadas atingida” e do sinal de interface VDI “Quantidade de peças atingida”. |
| \$AC_TOTAL_PARTS | Quantidade total de peças produzidas até o momento (valor atual). O contador indica a quantidade total de peças produzidas desde instante inicial. Este contador é automaticamente resetado com valores default somente quando o controle é energizado. |
| \$AC_ACTUAL_PARTS | Quantidade atual de peças. Este contador grava a quantidade total de peças produzidas a partir de um instante inicial. O contador é automaticamente zerado (desde que \$AC_REQUIRED_PARTS não seja zero) quando a quantidade desejada de peças (\$AC_REQUIRED_PARTS) for atingida. |
| \$AC_SPECIAL_PARTS | Quantidade de peças especificadas pelo usuário. Este contador permite ao usuário definir uma contagem de peças específicas. A emissão de um alarme pode ser definida para caso a variável \$AC_REQUIRED_PARTS possua o mesmo valor (peças desejadas). O usuário deve resetar este contador. |



A função “contador de peças” opera independente do gerenciamento de ferramentas.

Todos os contadores podem ser lidos e escritos a partir da MMC.

Todos os contadores são resetados com valores standard quando o NC for energizado, e podem ser lidas/escritas independentes de sua ativação.



Exemplo de Programação

1. Ativação do contador de peças \$AC_REQUIRED_PARTS:

```
$MC_PART_COUNTER= ' H3 '
```

\$AC_REQUIRED_PARTS é ativado, um alarme será emitido quando

```
$AC_REQUIRED_PARTS == $AC_SPECIAL_PARTS
```

13.8 Tempo de exec.programa/contagem de peças (a partir da SW 5.2)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

2. Ativando o contador de peças \$AC_TOTAL_PARTS:

\$MC_PART_COUNTER= 'H10'

\$MC_PART_COUNTER_MCODE[0]=80

\$AC_TOTAL_PARTS é ativado, o contador é incrementado em 1 a cada M02, \$MC_PART_COUNTER_MCODE[0] é irrelevante

3. Ativando o contador de peças \$AC_ACTUAL_PARTS:

\$MC_PART_COUNTER= 'H300'

\$MC_PART_COUNTER_MCODE[1]=17

\$AC_ACTUAL_PARTS é ativado, o contador é incrementado em 1 a cada M17

4. Ativando o contador de peças \$AC_SPECIAL_PARTS:

\$MC_PART_COUNTER= 'H3000'

\$MC_PART_COUNTER_MCODE[2]=77

\$AC_SPECIAL_PARTS é ativado, o contador é incrementado em 1 a cada M77

5. Desativando o contador de peças \$AC_ACTUAL_PARTS:

\$MC_PART_COUNTER= 'H200'

\$MC_PART_COUNTER_MCODE[1]=50

\$AC_ACTUAL_PARTS não é ativado, permanece irrelevante

6. Ativando todos os contadores, exemplos 1–4:\$MC_PART_COUNTER
= 'H3313'

\$MC_PART_COUNTER_MCODE[0]

=80

\$MC_PART_COUNTER_MCODE[1]

=17

\$MC_PART_COUNTER_MCODE[2]

=77

\$AC_REQUIRED_PARTS é ativado
Exibe o alarme \$AC_REQUIRED_PARTS ==
\$AC_SPECIAL_PARTS
\$AC_TOTAL_PARTS é ativado, o contador é incrementado em 1 a cada M02
\$MC_PART_COUNTER_MCODE[0] é irrelevante
\$AC_ACTUAL_PARTS é ativado, o contador é incrementado em 1 a cada M17
\$AC_SPECIAL_PARTS é ativado, o contador é incrementado em 1 a cada M77

Programas de usuário para desbaste

| | | |
|------|---|--------|
| 14.1 | Funções de apoio para desbaste..... | 14-464 |
| 14.2 | Preparação do contorno: CONTPRON | 14-465 |
| 14.3 | Decodificação do contorno: CONTDCON (a partir da SW 5.2)..... | 14-472 |
| 14.4 | Interseção de dois elementos de contorno: INTERSEC | 14-476 |
| 14.5 | Executando movimentos de elementos de contorno de uma tabela: EXECTAB.. | 14-478 |
| 14.6 | Calcular dados circulares: CALCDAT..... | 14-479 |

14.1 Funções de apoio para desbaste



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

14.1 Funções de apoio para desbaste



Programas de usuário para desbaste

Existem programas pré definidos desbaste. Você também utilizar as seguintes funções para desenvolver seus próprios programas.

| | |
|----------|--|
| CONTPRON | Ativa preparação tabular de contorno (11 colunas) |
| CONTDCON | Ativa decodificação tabular de contorno (6 colunas) |
| INTERSEC | Calcula a interseção de dois elementos de contorno (Somente para tabelas criadas com CONTPRON). |
| EXECTAB | Execução bloco a bloco dos elementos de contorno de uma tabela (Somente para tabelas geradas através de CONTPRON). |
| CALCDAT | Calcula os pontos centrais dos raios |



Você pode utilizar estas funções de forma universal, e não somente para desbaste.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

14.2 Preparação do contorno: CONTPRON



Programação

CONTPRON (TABNAME, MACH, NN, MODE)
EXECUTE (ERROR)



Explicação dos parâmetros

| | |
|-------------------------|---|
| CONTPRON | Ativa a preparação do contorno |
| TABNAME | Nome da tabela de contorno |
| MACH | Parâmetros para o tipo de usinagem: "G": Torneamento longitudinal: usinagem interna "L": Torneamento longitudinal: usinagem externa "N": Torneamento facial: usinagem interna "P": Torneamento facial: usinagem externa |
| NN | Quantidade de cortes de alívio, como resultado da variável de tipo INT |
| MODE (SW 4.4 em diante) | Direção da usinagem, tipo INT 0 = Preparação do contorno para frente (default, a partir da SW4.3) 1 = Preparação em ambas as direções |
| EXECUTE | Encerra a preparação do contorno |
| ERROR | Variável de erro para checagem, tipo INT 1 = erro; 0 = sem erro |



Função

Os blocos executados após CONTPRON descrevem o contorno a ser preparado. Os blocos não são processados mas sim preenchidos na tabela de contorno. Cada elemento de contorno corresponde a uma coluna no array de duas dimensões da tabela de contorno. O retorno é a quantidade de cortes de alívio. EXECUTE desativa a preparação de contorno e comuta novamente para o modo normal de execução.

Exemplo:

```
N30 CONTPRON (...)  
N40 G1 X... Z...  
N50...  
N100 EXECUTE (...)
```


14.2 Preparação do contorno: CONTPRON



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Informações adicionais

Pré condições para a chamada

Antes da chamada de CONTPRO

- Os eixos devem ser levados à um ponto inicial que permita uma usinagem livre de colisões,
- A compensação de raio da ferramenta com G40 deve ser desativada.

Comandos de movimento permitidos, sistema de coordenadas

Somente os comandos G de G0 à G3 são permitidos para a programação de contornos em conjunto com os arredondamentos e chanfros.

Da SW4.4 em diante existe o suporte à programação de trajetórias circulares através de CIP e CT.

As funções Spline, Polinômios, e rosqueamento produzem erros.

Não é permitida a alteração do sistema de coordenadas através de frames entre os comandos CONTPRON e EXECUTE. O mesmo se aplica à comutações entre G70 e G71/G700 e G710.

A alteração de eixos geométricos com GEOAX durante a preparação de uma tabela de contorno produz um alarme.

Encerrando a preparação de um contorno

Quando você chamar a subrotina pré definida EXECUTE (variável), a preparação do contorno é encerrada e o sistema é comutado de volta à execução normal, quando o contorno tiver sido descrito. A variável então indica:

1 = erro

0 = sem erro (o contorno não possui erros).

Elementos de corte de alívio

A descrição do contorno para os elementos de alívio individuais pode ser realizada tanto em uma subrotina quando em blocos individuais.

Desbaste independente da direção de contorno programada (SW 4.4 em diante)

A partir da SW4.4, a preparação de contornos foi expandida. Quando chamada a função CONTPRON, a tabela de contornos é tornada disponível independente da direção.

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



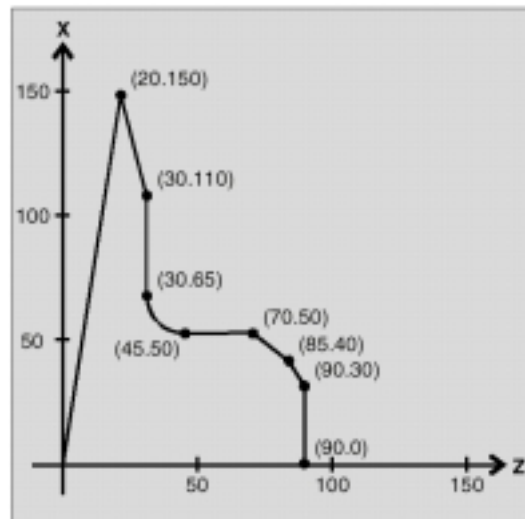
840Di



Exemplo de programação 1

Criando uma tabela de contornos com

- Nome KTAB,
- Até 30 elementos de contorno (círculos, retas),
- Uma quantidade variável de elementos de corte de alívio,
- Uma variável para mensagens de erro



Programa de NC

```
N10 DEF REAL KTAB[30,11]
```

A tabela de contorno de nome KTAB, por exemplo, com um máximo de 30 elementos.

O valor do parâmetro 11 é fixo.

```
N20 DEF INT ANZHINT
```

Variável para a quantidade de cortes de alívio com o nome ANZHINT

```
N30 DEF INT ERROR
```

Variável para checagem
0 = sem erros, 1 = erros

```
N40 G18
```

```
N50 CONTPRON (KTAB, "G", ANZHINT)
```

Chamada da preparação do contorno

```
N60 G1 X150 Z20
```

N60 a N120: descrição do contorno

```
N70 X110 Z30
```

```
N80 X50 RND=15
```

```
N90 Z70
```

```
N100 X40 Z85
```

```
N110 X30 Z90
```

```
N120 X0
```

```
N130 EXECUTE (ERROR)
```

Término do preenchimento da tabela de contorno, comuta para execução normal do programa

```
N140 ...
```

Continua o processamento da tabela

14.2 Preparação do contorno: CONTPRON840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Tabela KTAB**

| (0) | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) | (10) |
|-----|-----|-----|-----|-----|-----|-----|-------|-------------|-----|------|
| 7 | 7 | 11 | 0 | 0 | 20 | 150 | 0 | 82.40535663 | 0 | 0 |
| 0 | 2 | 11 | 20 | 150 | 30 | 110 | -1111 | 104.0362435 | 0 | 0 |
| 1 | 3 | 11 | 30 | 110 | 30 | 65 | 0 | 90 | 0 | 0 |
| 2 | 4 | 13 | 30 | 65 | 45 | 50 | 0 | 180 | 45 | 65 |
| 3 | 5 | 11 | 45 | 50 | 70 | 50 | 0 | 0 | 0 | 0 |
| 4 | 6 | 11 | 70 | 50 | 85 | 40 | 0 | 146.3099325 | 0 | 0 |
| 5 | 7 | 11 | 85 | 40 | 90 | 30 | 0 | 116.5650512 | 0 | 0 |
| 6 | 0 | 11 | 90 | 30 | 90 | 0 | 0 | 90 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Explicação dos conteúdos das colunas

(0) Ponteiro para o próximo elemento de contorno (para o número da linha daquela coluna)

(1) Ponteiro do elemento de contorno anterior

(2) Codificação do modo do contorno para o movimento

Valores possíveis para X = abc

a = 10² G90 = 0 G91 = 1

b = 10¹ G70 = 0 G71 = 1

c = 10⁰ G0 = 0 G1 = 1 G2 = 2 G3 = 3

(3), (4) Ponto inicial dos elementos de contorno

(3) = abscissa, (4) = ordenada no plano atual

(5), (6) Ponto inicial dos elementos de contorno

(5) = abscissa, (6) = ordenada no plano atual

(7) Indicador Max/min: identifica os valores locais máximos e mínimos no contorno

(8) Máximo valor entre o elemento de contorno e a abscissa (para usinagem longitudinal) ou ordenada (para usinagem transversal)

O ângulo depende do tipo de usinagem programada.

(9), (10) Ponto central das coordenadas do elemento de contorno, caso seja um bloco circular.

(9) = abscissa, (10) = ordenada

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



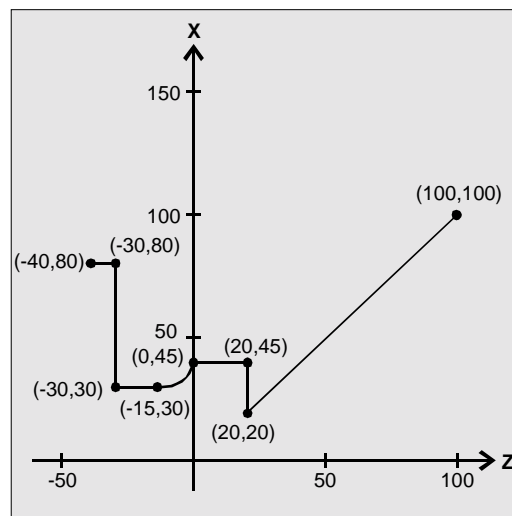
840Di



Exemplo de programação 2

Criar uma tabela de contorno com

- Nome Ktab,
- Até 92 elementos de contorno (círculos, linhas retas),
- Modo: Torneamento longitudinal, usinagem externa
- Preparação adiante e atrás



Programa NC

```
N10 DEF REAL KTAB[92,11]
```

A tabela de contorno chamada KTAB e, por exemplo, um máximo de 92 elemento.

O parâmetro 11 é fixo

```
N20 CHAR BT="L"
```

Modo para CONTPRON: torneamento longitudinal, usinagem externa

```
N30 DEF INT HE=0
```

Quantidade de cortes de alívio=0

```
N40 DEF INT MODE=1
```

Preparação adiante e atrás

```
N50 DEF INT ERR=0
```

Checagem de erros

```
...
```

```
N100 G18 X100 Z100 F1000
```

```
N105 CONTPRON (KTAB, BT, HE, MODE)
```

Prepara a chamada do contorno

```
N110 G1 G90 Z20 X20
```

```
N120 X45
```

```
N130 Z0
```

```
N140 G2 Z-15 X30 K=AC(-15) I=AC(45)
```

```
N150 G1 Z-30
```

```
N160 X80
```

```
N170 Z-40
```

```
N180 EXECUTE (ERR)
```

Encerra o preenchimento da tabela de contorno, comuta para o modo normal de execução de programa.

```
...
```

14.2 Preparação do contorno: CONTPRON840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Tabela KTAB**

Após o término da preparação do contorno, o este é disponibilizado em ambas as direções.

| Linha | Coluna | | | | | | | | | | |
|-------|------------------|-------------------|-----|-----|-----|-----|-----|-------|-----|-----|------|
| | (0) | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) | (10) |
| 0 | 6 ¹⁾ | 7 ²⁾ | 11 | 100 | 100 | 20 | 20 | 0 | 45 | 0 | 0 |
| 1 | 0 ³⁾ | 2 | 11 | 20 | 20 | 20 | 45 | -3 | 90 | 0 | 0 |
| 2 | 1 | 3 | 11 | 20 | 45 | 0 | 45 | 0 | 0 | 0 | 0 |
| 3 | 2 | 4 | 12 | 0 | 45 | -15 | 30 | 5 | 90 | -15 | 45 |
| 4 | 3 | 5 | 11 | -15 | 30 | -30 | 30 | 0 | 0 | 0 | 0 |
| 5 | 4 | 7 | 11 | -30 | 30 | -30 | 45 | -1111 | 90 | 0 | 0 |
| 6 | 7 | 0 ⁴⁾ | 11 | -30 | 80 | -40 | 80 | 0 | 0 | 0 | 0 |
| 7 | 5 | 6 | 11 | -30 | 45 | -30 | 80 | 0 | 90 | 0 | 0 |
| 8 | 1 ⁵⁾ | 2 ⁶⁾ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | ... | | | | | | | | | | |
| 83 | 84 | 0 ⁷⁾ | 11 | 20 | 45 | 20 | 80 | 0 | 90 | 0 | 0 |
| 84 | 90 | 83 | 11 | 20 | 20 | 20 | 45 | -1111 | 90 | 0 | 0 |
| 85 | 0 ⁸⁾ | 86 | 11 | -40 | 80 | -30 | 80 | 0 | 0 | 0 | 0 |
| 86 | 85 | 87 | 11 | -30 | 80 | -30 | 30 | 88 | 90 | 0 | 0 |
| 87 | 86 | 88 | 11 | -30 | 30 | -15 | 30 | 0 | 0 | 0 | 0 |
| 88 | 87 | 89 | 13 | -15 | 30 | 0 | 45 | -90 | 90 | -15 | 45 |
| 89 | 88 | 90 | 11 | 0 | 45 | 20 | 45 | 0 | 0 | 0 | 0 |
| 90 | 89 | 84 | 11 | 20 | 45 | 20 | 20 | 84 | 90 | 0 | 0 |
| 91 | 83 ⁹⁾ | 85 ¹⁰⁾ | 11 | 20 | 20 | 100 | 100 | 0 | 45 | 0 | 0 |

Explicação dos conteúdos das colunas

- (0) Ponteiro para o próximo elemento de contorno (para o número da linha daquela coluna)
- (1) Ponteiro do elemento de contorno anterior
- (2) Codificação do modo do contorno para o movimento
Valores possíveis para X = abc
a = 10² G90 = 0 G91 = 1
b = 10¹ G70 = 0 G71 = 1
c = 10⁰ G0 = 0 G1 = 1 G2 = 2 G3 = 3
- (3), (4) Ponto inicial dos elementos de contorno
(3) = abscissa, (4) = ordenada no plano atual
- (5), (6) Ponto inicial dos elementos de contorno
(5) = abscissa, (6) = ordenada no plano atual
- (7) Indicador Max/min: identifica os valores locais máximos e mínimos no contorno
- (8) Máximo valor entre o elemento de contorno e a abscissa (para usinagem)

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

longitudinal) ou ordenada (para usinagem transversal)

O ângulo depende do tipo de usinagem programada.

(9), (10) Ponto central das coordenadas do elemento de contorno, caso seja um bloco circular.

(9) = abscissa, (10) = ordenada

Explicação do comentário nas colunas

Sempre na linha 0 da tabela: 1) Anterior: A linha n contém o fim do contorno para frente

2) Posterior: A linha n é o fim da tabela para frente

Uma vez cada dentro dos elementos de contorno para frente:

3) Anterior: Início do contorno (para frente)

4) Posterior: Fim do contorno (para frente)

Sempre no fim da linha da tabela de contorno (para frente) +1:

5) Anterior: Quantidade de cortes de alívio para frente

6) Posterior: quantidade de cortes de alívio para trás

Uma vez em cada elemento de contorno para trás:

7) Próximo: Fim do contorno (para trás)

8) Anterior: Início do contorno (para trás)

Sempre na última linha da tabela:

9) Anterior: A linha n é o início da tabela de contorno (para trás)

10) Posterior: A linha n contém o início do contorno (para trás)

14.3 Decodificação do contorno: CONTDCON (a partir da SW 5.2)840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

14.3 Decodificação do contorno: CONTDCON (a partir da SW 5.2)**Programação**

CONTDCON (TABNAME , MODE)
EXECUTE (ERROR)

**Explicação dos parâmetros**

| | |
|----------|--|
| CONTDCON | Ativa a preparação do contorno |
| TABNAME | Nome da tabela de contorno |
| MODE | Direção da usinagem, tipo INT 0 = Preparação do contorno (default) de acordo com a seqüência dos blocos |
| EXECUTE | Encerra a preparação do contorno |
| ERROR | Variável para checagem de erros, tipo INT 1 = erro; 0 = sem erro |

**Função**

Os blocos executados após CONTPRON descrevem o contorno do bloco a ser decodificado.

Os blocos não são processados mas sim armazenados, e a memória é otimizada em uma tabela de contorno de 6 colunas.

Cada elemento de contorno corresponde a uma linha na tabela. Quando você estiver familiarizado com as regras de codificação especificadas abaixo, você poderá combinar as instruções de programação DIN da tabela para produzir aplicações (p.e. ciclos). Os dados para o ponto inicial são gravados. Os dados para o ponto inicial são gravados em uma célula da tabela com número 0. Os códigos G permitidos para uso com CONTDCOM em um trecho de programa podem ser incluídos na tabela e são mais abrangentes que os da função CONTPRON. Adicionalmente, os avanços e os tipos de velocidades são também gravados para cada trecho do contorno.

EXECUTE desativa a preparação do contorno e comuta o programa para o modo normal de execução.

Exemplo:

```
N30 CONTDCON ( ... )
N40 G1 X... Z...
N50...
N100 EXECUTE ( ... )
```

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di



Informações adicionais

Pré condições para a chamada

Antes da chamada de CONTDCON

- A máquina deve ser posicionada em um ponto inicial que permita a usinagem sem colisões,
- A correção do raio da ferramenta com G40 deve ser desativada.

Comandos de movimento permitidos, sistema de coordenadas

Os seguintes grupos G com seus comandos específicos são permitidos para a programação de contornos:

G grupo 1: G0, G1, G2, G3

G grupo 10: G9

G grupo 11: G60, G44, G641, G642

G grupo 13: G70, G71, G700, G710

G grupo 14: G90, G91

G grupo 15: G93, G94, G95, G96

Também cantos e chanfros.

Programação de trajetórias circulares é possível através de CIP e CT. As funções spline, polinômio, e rosqueamento produzem erros.

Não é permitida a mudança de sistema de coordenadas por ativação de frame entre CONDCRON e EXECUTE. O mesmo se aplica para a comutação entre G70 e G71/ G700 e G710. A comutação dos eixos geométricos com GEOAX durante a preparação da tabela de contorno produz um alarme.

14.3 Decodificação do contorno: CONTDCON (a partir da SW 5.2)



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D



840Di

Encerrando a preparação do contorno

Quando você chamar a subrotina pré definida EXECUTE (ERROR), a preparação do contorno é encerrada e o sistema é comutado para o modo normal de execução de blocos como. A variável associada ERRO fornece um valor de retorno:
0 = sem erro (O contorno não produziu erros)
1 = erro

Comandos não permitidos, condições iniciais incorretas, chamada repetida de CONTDCON sem EXECUTE(), poucos blocos de contorno ou definição de tabelas muito pequenas produzem alarmes adicionais.

Desbaste na direção de contorno programada

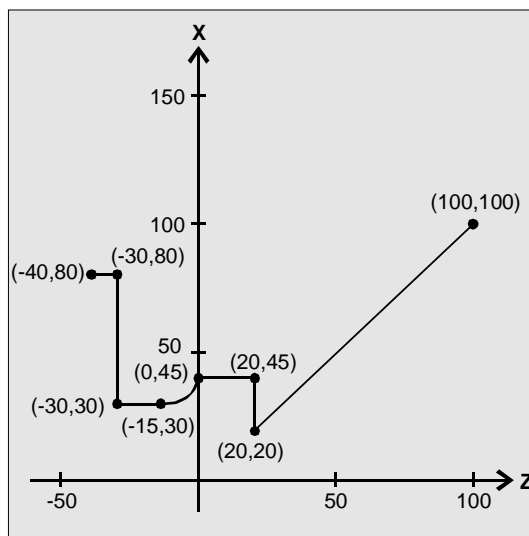
A tabela de contorno produzida com o uso de CONTDCON é utilizada no desbaste na direção do contorno.



Exemplo de programação

Criar uma tabela de contorno com

- Nome KTAB,
- Elementos de contorno(círculos, linhas retas),
- Modo: Torneamento
- Preparação para frente



840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Programa NC

N10 DEF REAL KTAB[9,6]

Tabela de contorno com o nome KTAB e 9 células de tabelamento. Estas permitem 8 conjuntos.

O parâmetro de valor 6 é fixado (quantidade de colunas na tabela).

N20 DEF INT MODE = 0

Valor inicial 0: somente na direção programada. Valor 1 não é permitido.

N30 DEF INT ERROR = 0

Checagem de erros

...

N100 G18 G64 G90 G94 G710

N101 G1 Z100 X100 F1000

N105 CONTDCON (KTAB, MODE)

Chamada da decodificação do contorno
MODE pode ser omitida (veja acima)

N110 G1 Z20 X20 F200

Descrição do contorno

N120 G9 X45 F300

N130 Z0 F400

N140 G2 Z-15 X30 K=AC(-15) I=AC(45) F100

N150 G64 Z-30 F600

N160 X80 F700

N170 Z-40 F800

N180 EXECUTE(ERROR)

Término do preenchimento da tabela,
comuta para o modo normal de execução
do programa

...

| Índice da coluna | 0 | 1 | 2 | 3 | 4 | 5 |
|---------------------|---------------------|-------------------------|-------------------------|------------------------------|------------------------------|------------|
| Índice da linha | Modo do contorno | Ponto final abscissa | Ponto final ordenada | Ponto central abscissa | Ponto central ordenada | Velocidade |
| 0 | 30 | 100 | 100 | 0 | 0 | 7 |
| 1 | 11031 | 20 | 20 | 0 | 0 | 200 |
| 2 | 111031 | 20 | 45 | 0 | 0 | 300 |
| 3 | 11031 | 0 | 45 | 0 | 0 | 400 |
| 4 | 11032 | -15 | 30 | -15 | 45 | 100 |
| 5 | 11031 | -30 | 30 | 0 | 0 | 600 |
| 6 | 11031 | -30 | 80 | 0 | 0 | 700 |
| 7 | 11031 | -40 | 80 | 0 | 0 | 800 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 |

14.4 Interseção de dois elementos de contorno: INTERSEC840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

Explicação e conteúdo das colunasLinha 0 Codificação do **ponto inicial**:

Coluna 0:

 10^0 (unidades): G0 = 0 10^1 (dezenas): G70 = 0, G71 = 1, G700 = 2, G710 = 3

Coluna 1: Ponto inicial da abscissa

Coluna 2: Ponto inicial da ordenada

Colunas 3–4: 0

Coluna 5 Índice da linha da última parte do contorno na tabela

Linhas 1–n: Valores para as **partes do contorno**

Coluna 0:

 10^0 (Unidades): G0 = 0, G1 = 1, G2 = 2, G3 = 3 10^1 (Dezenas): G70 = 0, G71 = 1, G700 = 2, G710 = 3 10^2 (Centenas): G90 = 0, G91 = 1 10^3 (Milhar): G93 = 0, G94 = 1, G95 = 2, G96 = 3 10^4 (Dezenas de milhar): G60 = 0, G44 = 1, G641 = 2, G642 = 3 10^5 (Centenas de milhar): G9 = 1

Coluna 1: Abscissa do ponto final

Coluna 2: Ordenada do ponto final

Coluna 3: Abscissa do ponto central interpolação circular

Coluna 4: Ordenada do ponto central para interpolação circular

Coluna 5: Avanço

14.4 Interseção de dois elementos de contorno: INTERSEC**Programação**

VARIB=INTERSEC (TABNAME1[n1], TABNAME2[n2], TABNAME3)

**Explicação dos parâmetros**

| | | |
|--------------|---|---|
| VARIB | Variável para estado | TRUE: Interseção encontrada FALSE: Interseção não encontrada |
| TABNAME1[n1] | Nome da tabela e 1º elemento de contorno da 1ª tabela | |
| TABNAME2[n2] | Nome da tabela e 2º elemento de contorno da 2ª tabela | |
| TABNAME3 | Nome da tabela para as coordenadas de interseção no plano ativo G17 – G19 | |

**Função**

INTERSEC calcula a interseção de dois contornos normalizados a partir da tabela gerada com CONTPRON. O estado indicado especifica quando existe ou não uma interseção (TRUE = interseção, FALSE = sem interseção).

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

**Informações adicionais**

Favor notar que as variáveis devem ser definidas antes de seu uso.

**Exemplo de programação**

Calcule a interseção do elemento de contorno 3 na tabela KTAB1 e o elemento de contorno 7 na tabela KTAB2. As coordenadas da interseção no plano ativo são gravadas em CUTCUT (1^o elemento = abscissa, 2^o elemento = ordenada).

Caso não existam interseções, o programa salta para NOCUT (interseções não encontradas)

| | |
|--|---|
| DEF REAL KTAB1 [12, 11] | 1 ^a tabela de contornos |
| DEF REAL KTAB2 [10, 11] | 2 ^a tabela de contornos |
| DEF REAL CUT [2] | Tabela de interseção |
| DEF BOOL ISPOINT | Variáveis de estado |
| ... | |
| N10 ISPOINT=INTERSEC (KTAB1[3],KTAB2[7],CUT) | Chamada na interseção dos elementos de contorno |
| N20 IF ISPOINT==FALSE GOTOF NOCUT | Salta para NOCUT |
| ... | |

14.4 Interseção de dois elementos de contorno: INTERSEC840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

14.5 Executando movimentos de elementos de contorno de uma tabela: EXECTAB**Programação**

EXECTAB (TABNAME [n])

**Explicação dos parâmetros**

TABNAME [n]

Nome da tabela com o número n do elemento

**Função**

Você pode usar o comando EXECTAB para executar os movimentos bloco a bloco dos elementos de contorno em uma tabela gerada, por exemplo, com o comando CONTPRON.

**Exemplo de programação**

Os elementos de contorno gravados na tabela KTAB são percorridos de forma não modal através da subrotina EXECTAB. Os elemento de 0 à 2 são executados em chamadas consecutivas.

| | | |
|-----|-------------------|---|
| N10 | EXECTAB (KTAB[0]) | Elemento de movimentação 0 da tabela KTAB |
| N20 | EXECTAB (KTAB[1]) | Elemento de movimentação 1 da tabela KTAB |
| N30 | EXECTAB (KTAB[2]) | Elemento de movimentação 2 da tabela KTAB |

840D
NCU 571840D
NCU 572
NCU 573

FM-NC



810D



840Di

14.6 Calcular dados circulares: CALCDAT



Programação

VARIB = CALCDAT(PT[n, 2], NO, RES)



Explicação dos parâmetros

| | |
|----------|---|
| VARIB | Variável de estado TRUE = círculo, FALSE = sem círculos |
| PT[n, 2] | Pontos de cálculo n = quantidade de pontos (3 ou 4); 2 = coordenadas do ponto |
| NO. | Quantidade de pontos para cálculos: 3 ou 4 |
| RES[3] | Variável para resultados: especificação das coordenadas do ponto central do círculo do raio; 0 = abscissa, 1 = ordenada do ponto central da circunferência; 2 = raio |



Função

Cálculo das coordenadas do raio e do centro do círculo a partir de 3 ou 4 pontos conhecidos do círculo.

14.6 Calcular dados circulares: CALCDAT



840D
NCU 571



840D
NCU 572
NCU 573



FM-NC



810D

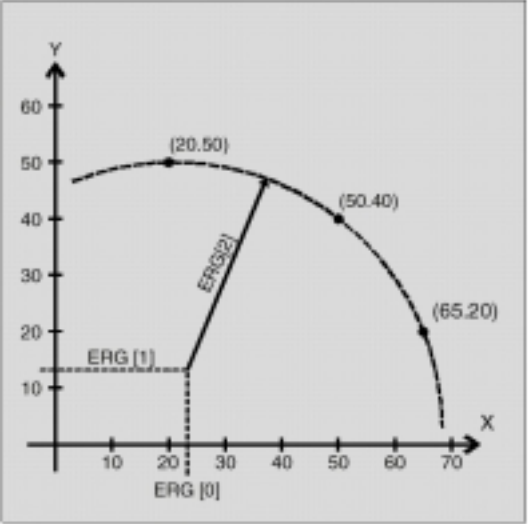


840Di



Exemplo de programação

O programa determina se os três pontos se encontram ao longo de um arco do círculo.



```
N10 DEF REAL  
PT[3,2]=(20,50,50,40,65,20)  
N20 DEF REAL RES[3]  
N30 DEF BOOL STATUS  
N40 STATUS = CALCDAT(PT,3,RES)  
N50 IF STATUS == FALSE GOTOF ERROR
```

Definição do ponto

Resultado

Estado da variável

Cálculo dos dados do círculo

Salta em caso de erro.

Tabelas

| | | |
|---------|--|--------|
| 15.1 | Lista de instruções..... | 15-484 |
| 15.2 | Lista das variáveis de sistema..... | 15-509 |
| 15.2.1 | Parâmetros R | 15-509 |
| 15.2.2 | Frames 1 | 15-509 |
| 15.2.3 | Dados do porta ferramenta | 15-510 |
| 15.2.4 | Zonas de proteção específicas do canal..... | 15-513 |
| 15.2.5 | Parâmetros de ferramenta | 15-514 |
| 15.2.6 | Dados de monitoração para o gerenciamento de ferramentas..... | 15-526 |
| 15.2.7 | Dados de monitoração para usuários OEM | 15-527 |
| 15.2.8 | Dados relacionados à ferramenta | 15-527 |
| 15.2.9 | Dados de ferramenta relativos à retíficas | 15-529 |
| 15.2.10 | Dados de alojamento do magazine..... | 15-530 |
| 15.2.11 | Dados de magazine para usuários OEM | 15-531 |
| 15.2.12 | Dados de descrição do magazine para o gerenciamento de ferramentas | 15-532 |
| 15.2.13 | Descrição do magazine de ferramentas para usuários OEM | 15-533 |
| 15.2.14 | Parâmetro de módulo do magazine | 15-534 |
| 15.2.15 | Valores de compensação do sistema de medição | 15-534 |
| 15.2.16 | Compensação de erro de quadrante | 15-535 |
| 15.2.17 | Compensação de interpolação | 15-536 |
| 15.2.18 | Zonas específicas de proteção NCK..... | 15-537 |
| 15.2.19 | Dados de sistema | 15-538 |
| 15.2.20 | Frames 2 | 15-539 |
| 15.2.21 | Dados de ferramentas..... | 15-539 |
| 15.2.22 | Valores programados | 15-541 |
| 15.2.23 | Grupos G..... | 15-541 |
| 15.2.24 | Estados do canal..... | 15-543 |
| 15.2.25 | Ações síncronas..... | 15-546 |
| 15.2.26 | I/Os..... | 15-548 |
| 15.2.27 | Leitura e escrita de variáveis PLC..... | 15-549 |
| 15.2.28 | Conexão NCU | 15-549 |
| 15.2.29 | I/O PLC direto..... | 15-550 |
| 15.2.30 | Gerenciamento de ferramenta | 15-551 |
| 15.2.31 | Temporizadores | 15-552 |
| 15.2.32 | Movimento da peça | 15-553 |
| 15.2.33 | Velocidades | 15-554 |
| 15.2.34 | Fusos..... | 15-555 |
| 15.2.35 | Valores de polinômios para ações síncronas | 15-557 |
| 15.2.36 | Estado do canal..... | 15-558 |
| 15.2.37 | Posições..... | 15-558 |
| 15.2.38 | Eixos indexados | 15-559 |
| 15.2.39 | Limite de frequência do encoder | 15-559 |
| 15.2.40 | Valores do encoder | 15-560 |
| 15.2.41 | Medição axial | 15-561 |
| 15.2.42 | Deslocamentos | 15-561 |
| 15.2.43 | Distâncias axiais..... | 15-562 |
| 15.2.44 | Oscilação..... | 15-563 |

| | | |
|---------|--|--------|
| 15.2.45 | Velocidades dos eixos | 15-564 |
| 15.2.46 | Dados de acionamento | 15-565 |
| 15.2.47 | Estado do eixo | 15-566 |
| 15.2.48 | Câmbio eletrônico 1 | 15-567 |
| 15.2.49 | Valor principal de acoplamento | 15-568 |
| 15.2.50 | Fuso sincronizado | 15-569 |
| 15.2.51 | Safety Integrated 1 | 15-569 |
| 15.2.52 | Parada prolongada e recuo | 15-570 |
| 15.2.53 | Recipiente de eixos | 15-571 |
| 15.2.54 | Câmbio eletrônico 2 | 15-571 |
| 15.2.55 | Safety Integrated 2 | 15-572 |
| 15.1 | Lista de instruções | 15-484 |
| 15.2 | Lista das variáveis de sistema | 15-509 |
| 15.2.1 | Parâmetros R | 15-509 |
| 15.2.2 | Frames 1 | 15-509 |
| 15.2.3 | Dados do porta ferramenta | 15-510 |
| 15.2.4 | Zonas de proteção específicas do canal | 15-513 |
| 15.2.5 | Parâmetros de ferramenta | 15-514 |
| 15.2.6 | Dados de monitoração para o gerenciamento de ferramentas | 15-526 |
| 15.2.7 | Dados de monitoração para usuários OEM | 15-527 |
| 15.2.8 | Dados relacionados à ferramenta | 15-527 |
| 15.2.9 | Dados de ferramenta relativos à retíficas | 15-529 |
| 15.2.10 | Dados de alojamento do magazine | 15-530 |
| 15.2.11 | Dados de magazine para usuários OEM | 15-531 |
| 15.2.12 | Dados de descrição do magazine para o gerenciamento de ferramentas | 15-532 |
| 15.2.13 | Descrição do magazine de ferramentas para usuários OEM | 15-533 |
| 15.2.14 | Parâmetro de módulo do magazine | 15-534 |
| 15.2.15 | Valores de compensação do sistema de medição | 15-534 |
| 15.2.16 | Compensação de erro de quadrante | 15-535 |
| 15.2.17 | Compensação de interpolação | 15-536 |
| 15.2.18 | Zonas específicas de proteção NCK | 15-537 |
| 15.2.19 | Dados de sistema | 15-538 |
| 15.2.20 | Frames 2 | 15-539 |
| 15.2.21 | Dados de ferramentas | 15-539 |
| 15.2.22 | Valores programados | 15-541 |
| 15.2.23 | Grupos G | 15-541 |
| 15.2.24 | Estados do canal | 15-543 |
| 15.2.25 | Ações síncronas | 15-546 |
| 15.2.26 | I/Os | 15-548 |
| 15.2.27 | Leitura e escrita de variáveis PLC | 15-549 |
| 15.2.28 | Conexão NCU | 15-549 |
| 15.2.29 | I/O PLC direto | 15-550 |
| 15.2.30 | Gerenciamento de ferramenta | 15-551 |
| 15.2.31 | Temporizadores | 15-552 |
| 15.2.32 | Movimento da peça | 15-553 |
| 15.2.33 | Velocidades | 15-554 |

| | | |
|---------|--|--------|
| 15.2.34 | Fusos..... | 15-555 |
| 15.2.35 | Valores de polinômios para ações síncronas | 15-557 |
| 15.2.36 | Estado do canal..... | 15-558 |
| 15.2.37 | Posições | 15-558 |
| 15.2.38 | Eixos indexados | 15-559 |
| 15.2.39 | Limite de frequência do encoder | 15-559 |
| 15.2.40 | Valores do encoder | 15-560 |
| 15.2.41 | Medição axial | 15-561 |
| 15.2.42 | Deslocamentos | 15-561 |
| 15.2.43 | Distâncias axiais..... | 15-562 |
| 15.2.44 | Oscilação..... | 15-563 |
| 15.2.45 | Velocidades dos eixos..... | 15-564 |
| 15.2.46 | Dados de acionamento | 15-565 |
| 15.2.47 | Estado do eixo..... | 15-566 |
| 15.2.48 | Câmbio eletrônico 1 | 15-567 |
| 15.2.49 | Valor principal de acoplamento | 15-568 |
| 15.2.50 | Fuso sincronizado | 15-569 |
| 15.2.51 | Safety Integrated 1 | 15-569 |
| 15.2.52 | Parada prolongada e recuo..... | 15-570 |
| 15.2.53 | Recipiente de eixos | 15-571 |
| 15.2.54 | Câmbio eletrônico 2 | 15-571 |
| 15.2.55 | Safety Integrated 2 | 15-572 |

15.1 Lista de instruções

Legenda:

¹ Definição default no início de programa (na configuração de fábrica do controle, se não nada mais for programado).

² A numeração dos grupos corresponde à tabela "Lista das funções G/funções preparatórias" da seção 12.3

³ Pontos finais absolutos: efeito modal; pontos finais incrementais: não modal; nos outros casos efeito modal/não modal dependendo da sintaxe da função G

⁴ Como centros de círculo, parâmetros de IPO têm efeito incremental. Mediante AC, eles podem ser programados absolutamente. Em outros significados (p.ex. passo de rosca), a modificação de endereço será ignorada.

⁵ Palavra chave não é válida para SINUMERIK FM-NC/810D

⁶ Palavra chave não é válida para SINUMERIK FM-NC/810D/NCU571

⁷ Palavra chave não é válida para SINUMERIK 810D

⁸ O usuário OEM pode incluir dois tipos de interpolação adicionais. Os nomes podem ser alterados pelo usuário OEM.

⁹ Palavra chave só é válida para SINUMERIK FM-NC

¹⁰ Para estas funções, o formato estendido de endereço não é permitido

| Nome | Significado | Valores | Descrição, Comentário | Sintaxe | modal (m)/ por bl. (s) | Grupo ² |
|------------------|--|---|---|-------------------------------------|---------------------------------|--------------------|
| : | Número do bloco - bloco principal (ver N) | 0 ... 9999 9999 só inteiros, sem sinal | indicação especial de blocos - em vez de N... ; este bloco deveria conter todas as instruções para uma fase de trabalho completa subsequente | p.ex. :20 | | |
| A | Eixo | Real | | | m,s ³ | |
| A2 ⁵ | Orientação da ferramenta: ang. euleriano | Real | | | s | |
| A3 ⁵ | Orientação da ferramenta: componente do vector de direção | Real | | | s | |
| A4 ⁵ | Orientação da ferramenta para o início de bloco | Real | | | s | |
| A5 ⁵ | Orientação da ferramenta para o fim de bloco; componente do vetor normal | Real | | | s | |
| ABS | Valor absoluto | Real | | | | |
| AC | Dimensão absoluta | 0, ..., 359.9999° | | X=AC(100) | s | |
| ACC ⁵ | Aceleração axial (acceleration axial) | Real, sem sinal | | | m | |
| ACN | Dimensão absoluta para eixos circulares, ir para a posição em direção negativa | | | A=ACN(...) B=ACN(...) C=ACN(...) | s | |
| ACP | Dimensão absoluta para eixos circulares, ir para a posição em direção positiva | | | A=ACP(...) B=ACP(...) C=ACP(...) | s | |
| ACOS | Arco coseno (função trigonométrica) | Real | | | | |
| ADIS | Distância de alisamento para funções de trajetória G1, G2, G3, ... | Real, sem sinal | | | m | |
| ADISPOS | Distância de alisamento para movimento | Real, sem | | | m | |

| Nome | Significado | Valores | Descrição, Comentário | Sintaxe | modal (m)/ por bl. (s) | Grupo ² |
|----------------------|--|--|-------------------------------------|--|---------------------------------|-----------------------|
| | rápido G0 | sinal | | | | |
| ALF | Ângulo de levantamento rápido (angle tilt fast) | Inteiro, sem sinal | | | m | |
| AMIRROR | Espelhamento programável (additive mirror) | | | AMIRROR X0 Y0 Z0 ; bloco próprio | s | 3 |
| AND | Operação lógica AND | | | | | |
| ANG | Ângulo de contorno | | | | s | |
| AP | Ângulo polar | 0, ..., ± 360° | | | m,s ³ | |
| APR | Proteção de acesso a leitura/exibição (Proteção de acesso à leitura) | Inteiro, sem sinal | | | | |
| APW | Proteção de acesso a escrita (Proteção de acesso à escrita) | Inteiro, sem sinal | | | | |
| AR | Ângulo de abertura (ângulo circular) | 0, ..., 360° | | | m,s ³ | |
| AROT | Rotação programável (rotação aditiva) | Rotação em torno do 1. eixo geo.: -180° .. 180° 2.eixo geo.: -89.999° ... 90° 3.eixo geo.: -180° .. 180° | | AROT X... Y... Z... ; AROT RPL= Programado em bloco separado | s | 3 |
| AS | Definição de macro | String | | | | |
| ASCALE | Alteração de escala programável (Additive SCALE) | | | ASCALE X... Y... Z... ; bloco separado | s | 3 |
| ASIN | Arco seno (função trigonométrica) | Real | | | | |
| ASPLINE ⁷ | Spline Akima | | | | m | 1 |
| ATAN2 | Arco tangente 2 | Real | | | | |
| ATRANS | Deslocamento aditivo programável (Additive TRANSlation) | | | ATRANS X... Y... Z... ; bloco separado | s | 3 |
| AX | Inteiro sem sinal | Real | | | m,s ³ | |
| AXCSWAP | Comuta recipiente de eixo | | | AXCSWAP(CTn,CTn+1,...) | | 25 |
| AXIS | Tipo de dado: nome de eixo | | Nome de arquivo pode ser adicionado | | | |

| | | | | | | |
|----------|---|--------|---|--|------------------|--|
| AXNAME | Converte uma sequência de caracteres de entrada em nome de eixo | String | Será emitido um alarme caso a sequência de caracteres não contenha um nome de eixo válido | | | |
| AXSTRING | Converte o nome do eixo em string | AXIS | Nome de arquivo pode ser adicionado | | | |
| B | Eixo | Real | | | m,s ³ | |

| | | | | | |
|----------------------|---|--|--------------------------------|------------------|----|
| B_AND | AND binário | | | | |
| B_NOT | Inversão de bit | | | | |
| B_OR | OR binário | | | | |
| B_XOR | OR exclusivo binário | | | | |
| B2 ⁵ | Orientação da ferramenta: ângulo euleriano | Real | | s | |
| B3 ⁵ | Orientação da ferramenta: componente do vector de direção | Real | | s | |
| B4 ⁵ | Orientação da ferramenta para o início de bloco | Real | | s | |
| B5 ⁵ | Orientação da ferramenta para o fim de bloco; componente do vetor normal | Real | | s | |
| BAUTO ⁷ | Especificação do primeiro segmento do spline através dos 3 pontos subsequentes (begin not a knot) | | | m | 19 |
| BLSYNC | O processamento da subrotina de interrupção somente no início da próxima comutação de bloco | | | | |
| BNAT ^{1,7} | Transição natural ao primeiro bloco de spline (begin natural) | | | m | 19 |
| BOOL | Tipo de dado: Valor booleano TRUE / FALSE ou 0 / 1 | | | | |
| BRISK ¹ | Aceleração brusca ao longo da trajetória | | | m | 21 |
| BRISKA | Ligar aceleração brusca ao longo da trajetória para os eixos programados | | | | |
| BSPLINE ⁷ | Spline B | | | m | 1 |
| BTAN ⁷ | Transição tangencial ao primeiro bloco de spline (begin tangencial) | | | m | 19 |
| C | Eixo | Real | | m,s ³ | |
| C2 ⁵ | Orientação da ferramenta: ângulo euleriano | Real | | s | |
| C3 ⁵ | Orientação da ferramenta: componente do vector de direção | Real | | s | |
| C4 ⁵ | Orientação da ferramenta para o início de bloco | Real | | s | |
| C5 ⁵ | Orientação da ferramenta para o fim de bloco; componente do vetor normal | Real | | s | |
| CAC | Aproximar posição absoluta (posição codificada: coordenada absoluta) | Valor codificado é índice de tabela; o valor posicionado é o da tabela | | | |
| CACN | Movimento absoluto na direção negativa do valor gravado na tabela. (Posição codificada negativa absoluta) | Permitida para programação de eixos rotativos como eixos de posicionamento | | | |
| CACP | Movimento absoluto na direção positiva do valor gravado na tabela. (Posição codificada positiva absoluta) | | | | |
| CALCDAT | Calcula o raio e o ponto central de 3 ou 4 pontos (calculate circle data) | VAR Real [3] | Os pontos devem ser diferentes | | |
| CALL | Chamada indireta de subrotina | | CALL PROGVAR | | |

| | | | | | | |
|-----------------------------------|--|--------------------|---|--|---|----|
| CANCEL | Cancela ação síncrona modal | INT | Cancela o ID especificado. Sem parâmetro: todas as ações síncronas modais serão canceladas. | | | |
| CASE | Ramificação condicional de programa | | | | | |
| CDC | Posicionamento direto (Posição codificada: coordenada direta) | | Vide CAC | | | |
| CDOF ¹ | Desliga monitoração de colisão (collision detection OFF) | | | | m | 23 |
| CDON | Liga monitoração de colisão (collision detection ON) | | | | m | 23 |
| CFC ¹ | Avanço constante no contorno (constant feed at contour) | | | | m | 16 |
| CFIN | Avanço constante na curvatura interna, aceleração na curvatura externa (constant feed at internal radius) | | | | m | 16 |
| CFTCP | Avanço constante no ponto de referência do gume de ferramenta (trajetória de ponto médio) (constant feed in tool-center-point) | | | | m | 16 |
| CHAN | Faixa específica de validade para dados | | Uma vez por canal | | | |
| CHANDATA | Define um número de canal para acesso aos dados | INT | Somente permitido no bloco de inicialização | | | |
| CHAR | Tipo de dado: carácter ASCII | 0, ..., 255 | | | | |
| CHF SW 3.5 em diante CHR | Chanfro; valor= comprimento do chanfro em direção ao movimento Chanfro; valor= comprimento do chanfro | Real, sem sinal | | | s | |
| CHKDNO | Checar número D | | | | | |
| CIC | Movimentar de forma incremental (Posição codificada: Coordenada incremental) | | Vide CAC | | | |
| CIP | Interpolação circular através de pontos intermediários | | | CIP X... Y... Z... I1=... J1=... K1=... | m | 1 |
| CLAL | Cancela alarme | INT | Parâmetro: número do alarme | | | |
| CLEARM | Reseta um/várias memórias para coordenação do canal | INT, 1 - n | Não influencia a usinagem do próprio canal | | | |
| CLGOF | Velocidade constante da peça para retífica centerless: Desligar (OFF) | | | | | |
| CLGON | Velocidade constante da peça para retífica centerless: Ligar (ON) | | | | | |
| CLRINT | Desliga a interrupção selecionada | INT | Parâmetro: número da interrupção | | | |
| CMIRROR | Espelhamento em um eixo de coordenada | FRAME | | | | |
| COMPOF ^{1,6} | Desliga compressor (OFF) | | | | m | 30 |
| COMPON ⁶ | Liga compressor (ON) | | | | m | 30 |
| COMPCURV | Liga compressor (ON): polinômios de curvas constantes | | | | m | 30 |
| CONTPRON | Ativa a preparação de contornos | | | | m | 49 |

| | | | | | | |
|------------------------|--|-----------------|--|--------------------|---|----|
| COS | Coseno (função trigonométrica) | Real | | | | |
| COUPDEF | Definição de grupo ELG / Grupo de fusos síncronos (definição de acoplamento) | String | Resposta na comutação de bloco (software): NOC: sem controle software, FINE/COARSE: software em "sincronismo grosso/fino", IPOSTOP: software em função de atingida a posição desejada em movimento sobreposto | | | |
| COUPDEL | Apara agrupamento ELG (couple delete) | | | | | |
| COUPOF | Desliga grupo ELG / par de fusos síncronos (couple OFF) | | | | | |
| COUPON | Liga grupo ELG / par de fusos síncronos (couple ON) | | | | | |
| COUPRES | Reseta grupo ELG (acoplamento) | | Valores programados inválidos; dados de máquina válidos | | | |
| CP | continuous path (trajetória contínua); movimento de trajetória | | | | m | 49 |
| CPRECOF ^{1,6} | Desliga precisão de contorno programável (Contour PRECision OFF) | | | | m | 39 |
| CPRECON ⁶ | Liga precisão de contorno programável (Contour PRECision ON) | | | | m | 39 |
| CPROT | Liga/desliga zona de proteção do canal | | | | | |
| CPROTDEF | Define zona de proteção do canal | | | | | |
| CR | Raio de círculo (circle radius) | Real, sem sinal | | | s | |
| CROT | Rotação do sistema de coordenadas | FRAME | Quantidade máxima de parâmetros: 6 | | | |
| CSCALE | Fator de escala para eixos múltiplos | FRAME | Quantidade máxima de parâmetros: 2 * quantidade máxima de eixos | | | |
| CSPLINE ⁷ | Spline cúbico | | | | m | 1 |
| CTAB | Define a posição do eixo seguidor em relação ao eixo principal a partir da tabela de curva | Real | Caso os parâmetros 4/5 não programados: escala standard | | | |
| CTABDEF | Liga a definição de tabela | | | | | |
| CTABDEL | Limpa tabela de curva | | | | | |
| CTABEND | Desliga a definição de tabela | | | | | |
| CTABINV | Define a posição do eixo principal de acordo com a posição do eixo seguidor a partir da tabela | Real | Vide CTAB | | | |
| CT | Círculo com transição tangencial | | | CT X... Y.... Z... | m | 1 |
| CTRANS | Deslocamento de origem para eixos múltiplos | FRAME | Max. de 8 eixos | | | |
| CUT2D ¹ | Correção de ferramenta 2 1/2D (CUTter compensation type 2Dimensional) | | | | m | 22 |

| | | | | | | |
|----------------------|---|--|---|--|---|----|
| CUT2DF | Correção de ferramenta 2 1/2D (CUTter compensation type 2dimensional frame); A correção de ferramenta produz efeito em relação ao frame atual (plano oblíquo) | | | | m | 22 |
| CUT3DC ⁵ | Correção de ferramenta 3D Fresagem circular (CUTter compensation type 3dimensional circumference) | | | | m | 22 |
| CUT3DF ⁵ | Correção de ferramenta 3D Fresagem frontal (CUTter compensation type 3dimensional face) | | | | m | 22 |
| CUT3DFF ⁵ | Correção de ferramenta 3D Fresagem frontal com orientação constante da ferramenta, dependendo do frame ativo (CUTter compens. type 3dimensional face frame) | | | | m | 22 |
| CUT3DFS ⁵ | Correção de ferramenta 3D Fresagem frontal com orientação constante da ferramenta, independentemente do frame ativo (CUTter compens. type 3dimensional face) | | | | m | 22 |
| CUTCONO ¹ | Desliga compensação constante de raio | | | | m | 40 |
| CUTCONON | Liga compensação constante de raio | | | | m | 40 |
| D | N.º da correção de ferramenta | 1, ..., 9 a partir do SW 3.5 1,....32 000 | contém dados de correção para uma certa ferramenta T... ; D0 → valores de correção para uma ferramenta | D... | | |
| DC | Dimensão absoluta para eixos circulares, ir para posição diretamente | | | A=DC(...) B=DC(...) C=DC(...) SPOS=DC(...) | s | |
| DEF | Definição de variável | Inteiro, sem sinal | | | | |
| DEFAULT | Ramificação em CASE | | Salta caso a expressão não seja satisfeita com outros valores especificados quaisquer | | | |
| DEFINE | Define macro | | | | | |
| DELDTG | Cancela caminho restante | | | | | |
| DELT | Apaga a ferramenta | | Número duplo pode ser omitido | | | |
| DIAMOF ¹ | Desliga programação em diâmetro (Diametral programming OFF) | | | | m | 29 |
| DIAMON | Liga programação em diâmetro (Diametral programming ON) | | | | m | 29 |
| DIAM90 | Programação em diâmetro para G90, programação em raio para G91 | | | | m | 29 |
| DILF | Comprimento para levantamento rápido | | | | m | |
| DISABLE | Desliga interrupções | | | | | |
| DISC | Sobressalto do círculo de transição – compensação de raio | 0, ..., 100 | | | m | |
| DISPLOF | Suprime exibição do bloco atual (display OFF) | | | | | |
| DISPR | Distância em percurso para posicionamento | Real, sem sinal | | | s | |

| | | | | | | |
|---------------------|--|-----------------|--|--|---|----|
| DISR | Distância para reposicionamento | Real, sem sinal | | | s | |
| DITE | Trajectoria para saída da rosca | Real | | | m | |
| DITS | Trajectoria para entrar na rosca | Real | | | m | |
| DIV | Divisão inteira | | | | | |
| DL | Soma de compensações da ferramenta | INT | | | m | |
| DRFOF | Desativa deslocamentos através de nônio (DRF) | | | | m | |
| DRIVE ⁹ | Aceleração ao longo da trajetória dependente da velocidade | | | | m | 21 |
| DRIVEA | Comuta a curva característica de aceleração para os eixos programados | | | | | |
| DZERO | Carrega como inválidos o número D para todas as ferramentas do grupo TO do canal | | | | | |
| EAUTO ⁷ | Define o último segmento spline através dos últimos 3 pontos | | | | m | 20 |
| EGDEF | Definição de um câmbio eletrônico (Electronic gear define) | | Para um eixo mestre e até 5 seguidores | | | |
| EGDEL | Apaga a definição de acoplamento para o eixo seguidor (Electronic gear delete) | | Dispara parada de pré processamento | | | |
| EGOFC | Desliga comutação contínua de câmbio eletrônico (Electronic gear OFF continuous) | | | | | |
| EGOFS | Desliga câmbio eletrônico de forma seletiva (Electronic gear OFF selective) | | | | | |
| EGON | Liga câmbio eletrônico (electronic gear ON) | | <u>Sem</u> sincronismo | | | |
| EGONSYN | Liga câmbio eletrônico (electronic gear ON synchronized) | | <u>Com</u> sincronismo | | | |
| ELSE | Ramificação do programa caso a condição IF seja preenchida | | | | | |
| ENABLE | Liga interrupções | | | | | |
| ENAT ^{1,7} | Curva de transição natural para o próximo bloco de movimentos (end natural) | | | | m | 20 |
| ENDFOR | Última linha para um loop FOR | | | | | |
| ENDIF | Última linha para uma ramificação IF | | | | | |
| ENDLOOP | Última linha para um loop sem fim (LOOP) | | | | | |
| ENDPROC | Última linha para um programa iniciado com PROC | | | | | |
| ENDWHILE | Última linha para um loop WHILE | | | | | |
| ETAN ⁷ | Curva de transição tangencial para o próximo bloco de movimentação no início de um spline (fim tangencial) | | | | m | 20 |
| EVERY | Executa a ação síncrona caso a condição seja comutada de FALSE para TRUE | | | | | |
| EXECTAB | Executa um elemento de uma tabela de movimentos (execute table) | | | | | |

| | | | | | | |
|---------|---|------------------------|---|--------------|--|--|
| EXECUTE | Liga a execução do programa | | Comuta para o modo normal de execução de programa a partir de um ponto referência ou após a criação de uma zona de proteção | | | |
| EXP | Função exponencial e^x | Real | | | | |
| EXTERN | Transmissão de uma subrotina com passagem de parâmetros | | | | | |
| F | Valor de avanço (tempo de espera é também programada com o endereço F em conjunto com G4) | 0.001, ..., 99 999.999 | Velocidade ao longo da trajetória Ferramenta/Peça; Unidade de medida em mm/min ou mm/rotações, dependendo de G94 ou G95 | F=100 G1 ... | | |

| | | | | | | |
|----------------------|---|--|--|------------------------------|---|----|
| FA | Avanço axial (Feed Axial) | 0.001, ..., 999999.999 mm/min, graus/min; 0.001, ..., 39999.9999 polegadas/min | | FA[X]=100 | m | |
| FAD | Velocidade de entrada para aproximação e recuos suaves (Feed approach / depart) | Real, sem sinal | | | | |
| FALSE | Constante lógica: False | BOOL | Pode ser substituída pelo valor inteiro 0 | | | |
| FCTDEF | Define uma função polinomial | | Utilizada em SYFCT ou PUTFTOCF. | | | |
| FCUB ⁶ | Avanço alterável após spline cúbico (Feed CUBic) | | | | m | 37 |
| FD | Avanço ao longo da trajetória para superposição por manivela (Feed DRF) | Real, sem sinal | | | s | |
| FDA | Avanço axial para superposição por manivela (Feed DRF Axial) | Real, sem sinal | | | s | |
| FFWOF ¹ | Desliga controle piloto (Feed Forward OFF) | | | | m | 24 |
| FFWON | Liga controle piloto (Feed Forward ON) | | | | m | 24 |
| FGREF | Raio referência | | | | m | |
| FGROUP | Especificação do(s) eixo(s) com avanço ao longo da trajetória | | F é válido para todos os eixos indicados sob FGROUPO | FGROUP (Axis1, [Axis2], ...) | | |
| FIFOLEN | Profundidade de pré processamento programável | | | | | |
| FL | Velocidade limite para eixos síncronos (Feed Limit) | Real, sem sinal | é válida a unidade ajustada com G93, G94, G95 (mov. rápido máx.) | FL [Eixo] =... | m | |
| FLIN ⁶ | Avanço linear alterável (feed linear) | | | | m | 37 |
| FMA | Avanço axial múltiplo | Real, sem sinal | | | m | |
| FNORM ^{1,6} | Avanço normal segundo DIN66025 (feed normal) | | | | m | 37 |
| FOR | Contador de loop com quantidade fixa de passagens | | | | | |
| FORI1 | Avanço para orientação do vetor de rotação em círculos grandes | | | | m | |
| FORI2 | Avanço para orientação sobreposta no vetor de rotação | | | | m | |
| FP | Ponto fixo: número do ponto fixo a aproximar | Inteiro, sem sinal | | G75 FP=1 | s | |
| FPO | Característica de avanço programada através de polinômio (feed polynomial) | Real | Coefficiente polinomial quadrático ou cúbico | | | |
| FPR | Designação de eixo circular | 0.001 ... 999999.999 | | FPR (Eixo rotativo) | | |
| FPRAOF | Desligar avanço por rotação | | | | | |

| | | | | | |
|-----------------------|--|---|-------------------------------------|---|----|
| FPRAON | Ligar avanço por rotação | | | | |
| FRAME | Tipo de dado para definir um sistema de coordenadas | Contém para cada eixo geométrico: deslocamento de origem, rotação, ângulo de corte, escala, espelhamento; Para cada eixo especial: Deslocamento, escala e espelhamento | | | |
| FRC | Avanço para chanfros e raios | | | s | |
| FRCM | Avanço para chanfros e raios modal | | | m | |
| FTOC | Comuta corretor de ferramenta fino | Como função de um polinômio de grau e definido por FCTDEF | | | |
| FTOCOF ^{1,6} | Desliga corretor fino de ferramenta online (fine tool offset OFF) | | | m | 33 |
| FTOCON ⁶ | Liga corretor fino de ferramenta online (fine tool offset ON) | | | m | 33 |
| FXS | Ligar movimento para limitador fixo (FiXed Stop) | Inteiro, sem sinal | 1 = seleção, 0 = desliga seleção | m | |
| FXST | Limite torque para movimento para limitador fixo (FiXed Stop Torque) | % | Programação opcional | m | |
| FXSW | Janela de monitoração para movimento para limitador fixo (FiXed Stop Window) | mm, polegada ou grau | Programação opcional | | |

| Funções G | | | | | | |
|------------------|---|---------------------------------------|----------------------------------|---|---|----|
| G | Função G (função preparatória) As funções G estão subdivididas em grupos G. Em um bloco pode ser escrita só uma função G de um grupo. Uma função G pode ter efeito modal (até nova ordem por uma outra função do mesmo grupo), ou ela produz efeito apenas para o bloco, no qual se encontra (efeito não modal). | Só valores inteiros, preestabelecidos | | G... | | |
| G0 | Interpolação linear com movimento rápido | | Instruções de movimento | G0 X... Z... | m | 1 |
| G1 ¹ | Interpolação linear com avanço | | | G1 X... Z... F... | m | 1 |
| G2 | Interpolação circular no sentido horário | | | G2 X... Z... I... K... F... ; centro e ponto final G2 X... Z... CR=... F...; raio e ponto final G2 AR=... I... K... F...; ângulo circular e centro G2 AR=... X... Z... F...; ângulo circular e p.final | m | 1 |
| G3 | Interpolação circular no sentido anti horário | | | G3 ... ; ou como feito para G2 | m | 1 |
| G4 | Tempo de espera, predeterminado | | movimento especial | G4 F... ; Atraso em s, ou G4 S... ; voltas do fuso Bloco separado | s | 2 |
| G9 | Posicionamento exato- desaceleração | | | | s | 11 |
| G17 ¹ | Seleção do plano de trabalho X/Y | | Direção de alim. Z | | m | 6 |
| G18 | Seleção do plano de trabalho Z/X | | Direção de alim. Y | | m | 6 |
| G19 | Seleção do plano de trabalho Y/Z | | Direção de alim. X | | m | 6 |
| G25 | Limite inferior das rotações do fuso | | Valor definido em eixos do canal | G25 X.. Y.. Z.. ; bloco próprio | s | 3 |
| G26 | Limite superior das rotações do fuso | | | G26 X.. Y.. Z..; bloco próprio | s | 3 |
| G33 | Interpolação de rosca com passo constante | 0.001, ..., 2000.00 mm/rot | Instrução de movimento | G33 Z... K... SF=... rosca cilíndrica G33 X... I... SF=... rosca transversal G33 Z... X... K... SF=... rosca cônica (o percurso no eixo Z é superior ao no eixo X) G33 Z... X... I... SF=... rosca cônica (o percurso no eixo X é superior ao no eixo Z) | m | 1 |
| G34 | Incremento no passo da rosca (alteração progressiva) | | Comando de movimento | G34 Z... K... F _{ZU} =... | m | 1 |
| G35 | Decrementa passo da rosca (alteração progressiva) | | Comando de movimento | G35 Z... K... F _{AB} =... | m | 1 |
| G40 ¹ | DESL. correção do raio de ferramenta | | | | m | 7 |
| G41 | Correção do raio de ferramenta à esquerda do contorno | | | | m | 7 |

| Funções G | | | | | |
|-------------------|--|---------------------------------|---|--------|----|
| G42 | Correção do raio de ferramenta à direita do contorno | | | m | 7 |
| G53 | Supressão do deslocamento programável de ponto zero | incl. deslocamentos programados | | s | 9 |
| G54 | 1.º deslocamento programável de ponto zero | | | m | 8 |
| G55 | 2.º deslocamento programável de ponto zero | | | m | 8 |
| G56 | 3.º deslocamento programável de ponto zero | | | m | 8 |
| G57 | 4.º deslocamento programável de ponto zero | | | m | 8 |
| G58 | Deslocamento do ponto zero axial programável absoluto | | | m | |
| G59 | Deslocamento do ponto zero axial programável aditivo | | | m | |
| G60 ¹ | Parada exata – desaceleração | | | m | 10 |
| G63 | Roscar com macho com mandril compensador | | G63 Z... G1 | s | 2 |
| G64 | Posicionamento exato - controlo contínuo da trajetória | | | m | 10 |
| G70 | Dimensões em Polegadas | | | m | 13 |
| G71 ¹ | Dimensões métricas | | | m | 13 |
| G74 | Referenciar | Eixos de máquina | G74 X... Y... Z...; bloco separado | s | 2 |
| G75 | Ir para ponto fixo | | G75 FP=.. X1=... Z1=...; bloco separado | s | 2 |
| G90 ¹ | Dimensão absoluta | | G90 X... Y... Z...(...) Y=AC(...) ou X=AC Z=AC(...) | M s | 14 |
| G91 | Dimensão incremental | | G91 X... Y... Z... ou X=IC(...) Y=IC(...) Z=IC(...) | m s | 14 |
| G94 ¹ | Avanço linear F em mm/min ou inch/min e °/min | | | m | 15 |
| G95 | Avanço por rotação F em mm/rot ou polegadas/rot | | | m | 15 |
| G96 | LIG. velocidade de corte constante (como G95) | | G96 S... LIMS=... F... | m | 15 |
| G97 | DESL. velocidade de corte constante (como G95) | | | m | 15 |
| G110 | Programação polar em relação à última posição teórica programada | | G110 X.. Y.. Z.. | s | 3 |
| G111 | Programação polar em relação ao ponto zero do sistema de coordenadas de peça atual | | G110 X.. Y.. Z.. | s | 3 |
| G112 | Programação polar em relação ao último pólo válido | | G110 X.. Y.. Z.. | s | 3 |
| G140 ¹ | Sentido de aprox.WAB definido por G41/G42 | | | m | 43 |
| G141 | Sentido de aprox.WAB à esquerda do contorno | | | m | 43 |
| G142 | Sentido de aprox.WAB à direita do contorno | | | m | 43 |
| G143 | Sentido de aprox.WAB em função da tangente | | | m | 43 |
| G147 | Aproximação suave em linha reta | | | s | 2 |
| G148 | Recuo suave em linha reta | | | s | 2 |
| G153 | Supressão do frame atual incluindo o frame base | | | s | 9 |
| G247 | Aproximação suave com quadrante | | | s | 2 |
| G248 | Recuo suave com quadrante | | | s | 2 |
| G331 | Rosqueamento | ±0.001, ..., | Instruções de movimento | m | 1 |
| G332 | Recuo (rosqueamento) | 2000.00 mm/rev | | m | 1 |

| Funções G | | | | | | |
|-------------------------------|--|--|---|-------------------------|---|----|
| G340 ¹ | Bloco de aproximação da região (profundidade e plano (hélice)) | | Funciona junto com aprox/rec.suave | | m | 44 |
| G341 | Profundidade inicial perpendicular ao eixo A, depois aproximar no plano | | Funciona junto com aprox/rec.suave | | m | 44 |
| G347 | Aproximação suave em semi-círculo | | | | s | 2 |
| G348 | Recuo suave em semi-círculo | | | | s | 2 |
| G450 ¹ | Círculo de transição | | Comport. nos cantos | | m | 18 |
| G451 | Ponto de interseção dos eqüidistantes | | na correção do raio de ferramenta | | m | 18 |
| G460 ¹ | Aproximação/recuo de acordo com TRC | | | | m | 48 |
| G461 | Aproximação/recuo de acordo com TRC | | | | m | 48 |
| G462 | Aproximação/recuo de acordo com TRC | | | | m | 48 |
| G500 ¹ | Desativar todos os frames ajustáveis caso G500 não contenha valor específico | | | | m | 8 |
| G505 G599 | 5.º ... 99.º deslocamento d programável e ponto zero | | | | m | 8 |
| G601 ¹ | Mudança de bloco em parada de precisão fina | | efetivo só com G60 ativo | | m | 12 |
| G602 | Mudança de bloco em parada de precisão grossa | | ou | | m | 12 |
| G603 | Mudança de bloco em IPO - fim de bloco | | G9 com alisamento de transição | | m | 12 |
| G641 | Parada de precisão - controle contínuo da trajetória | | Arredondamento da transição | G641 ADIS=... | m | 10 |
| G642 | Arredondamento de cantos com precisão axial. | | | | m | 10 |
| G643 | Arredondamento interno do bloco | | | | m | 10 |
| G700 | Dimensões em polegadas e inch/min | | | | m | 13 |
| G710 ¹ | Dimensões Métricas em mm e mm/min | | | | m | 13 |
| G810 ¹ , ..., G819 | Grupo G reservado para o usuário OEM | | | | | 31 |
| G820 ¹ , ..., G829 | Grupo G reservado para o usuário OEM | | | | | 32 |
| G961 | Liga velocidade de corte constante (como G94) | | | G961 S... LIMS=... F... | m | 15 |
| G971 | Desliga velocidade de corte constante (como G94) | | | | m | 15 |
| GEOAX | Define um novo eixo de canal para os eixos geométricos de 1 – 3 | | Sem definição de parâmetro: Definições dos MDs ativas | | | |
| GET | Define eixo/eixos de usinagem | | O eixo deve ser liberado no outro canal com RELEASE | | | |
| GETD | Relaciona o eixo/eixos de usinagem diretamente | | Vide GET | | | |
| GETACTT | Trás uma ferramenta ativa de um grupo de ferramentas com mesmo nome | | | | | |
| GETSELT | Trás n número da ferramenta selecionada | | | | | |
| GETT | Trás o número T a partir do nome de uma ferramenta | | | | | |

| Funções G | | | | | | |
|-------------------|---|--------------------------------|---|----------------|---|----|
| GOTOF | Instrução de salto para frente (sentido do fim do programa) | | | | | |
| GOTOB | Instrução de salto para trás (sentido do início do programa) | | | | | |
| GWPSOF | Desliga a velocidade periférica constante do rebolo (GWPS) | | | GWPSOF (T No.) | s | |
| GWPSON | Seleciona a velocidade periférica constante do rebolo (GWPS) | | | GWPSON (T No.) | s | |
| H... | Emite a função auxiliar para o PLC | Real/INT | Ajustável através de dados de máquina (fabricante da máquina) | H100 ou H2=100 | | |
| I ⁴ | Parâmetro de interpolação | Real | | | s | |
| I1 | Coordenada do ponto intermediário | Real | | | s | |
| IC | Entrada de dimensão incremental | 0, ..., ±99999.999° | | X=IC(10) | s | |
| IDS | Identificação de ação síncrona estática | | | | | |
| IF | Inicia salto condicional | | Estrutura: IF – ELSE – ENDIF | | | |
| INDEX | Define o índice do carácter em uma sequência de caracteres de entrada | 0, ..., INT | String: parâmetro 1, carácter: parâmetro 2 | | | |
| INIT | Selecionar bloco para a execução em um determinado canal | | | | | |
| INT | Tipo de dado: Inteiro com sinal | $-(2^{31}-1), \dots, 2^{31}-1$ | | | | |
| INTERSEC | Calcula a interseção entre dois elementos de contorno | VAR REAL [2] | Estado de erro BOOL | | | |
| IP | Parâmetro de interpolação variável | Real | | | | |
| ISAXIS | Verifica se os eixos geométricos 1-3 especificados como parâmetros existem | BOOL | | | | |
| ISD | Profundidade de imersão (insertion depth) | Real | | | m | |
| ISNUMBER | Verifica se a string de entrada pode ser convertida em um número | BOOL | | | | |
| J ⁴ | Parâmetro de interpolação | Real | | | s | |
| J1 | Coordenada do ponto intermediário | Real | | | s | |
| JERKA | Ativa a resposta de aceleração definida em dados de máquina para os eixos programados | | | | | |
| K ⁴ | Parâmetro de interpolação | Real | | | s | |
| K1 | Coordenada do ponto intermediário | Real | | | s | |
| KONT | Contornar quina com correção de ferramenta | | | | m | 17 |
| L | Número do subprograma | Inteiro, até 7 posições | Zeros precedentes relevantes! | L10 | s | |
| LEAD ⁵ | Ângulo de avanço | Real | | | m | |
| LFOF ¹ | Desliga interrupção de rosqueamento | | | | m | 41 |

| Funções G | | | | | | |
|--------------------|---|----------------------|----------------------------------|--|---|----|
| LEADOFF | Desliga a trajetória principal de acoplamento (lead off path) | | | | | |
| LEADON | Liga o acoplamento de valores (lead on) | | | | | |
| LEADONP | Liga a trajetória principal de acoplamento | | | | | |
| LEAD ⁵ | Ângulo de avanço | Real | | | m | |
| LFOF ¹ | Desliga interrupção de rosqueamento | | | | m | 41 |
| LFON | Liga interrupção de rosqueamento | | | | m | 41 |
| LFTXT ¹ | Direção tangencial no recuo da ferramenta | | | | m | 46 |
| LFWP | Direção não tangencial no recuo da ferramenta | | | | m | 46 |
| LIFTFAST | Levantamento rápido antes da chamada da rotina de interrupção | | | | | |
| LIMS | Limite de rotações (LIMit Spindle Speed) em G96 | 0.001 ... 99 999.999 | | | m | |
| LN | Logaritmo neperiano | Real | | | | |
| LOCK | Desabilita a ação síncrona com o ID (parada do ciclo tecnológico) | | | | | |
| LOG | Logaritmo (comum) | Real | | | | |
| LOOP | Loop sem fim | | Estrutura: LOOP – ENDLOOP | | | |
| M0 ¹⁰ | Parada programada | | | | | |
| M1 ¹⁰ | Parada opcional | | | | | |
| M2 ¹⁰ | Fim de programa do programa principal com reposição ao início de programa | | | | | |
| M3 | Sentido de rotação de fuso à direita para o fuso mestre | | | | | |
| M4 | Sentido de rotação de fuso à esquerda para o fuso mestre | | | | | |
| M5 | Parada de fuso para o fuso mestre | | | | | |
| M6 | Troca de ferramenta | | | | | |
| M17 ¹⁰ | Fim de subprograma | | | | | |
| M19 | Posiciona o fuso | | | | | |
| M30 ¹⁰ | Fim de programa, como M2 | | | | | |
| M40 | Mudança de velocidade automática | | | | | |
| M41... M45 | Nível de transmissão 1, ..., 5 | | | | | |
| M70 | Mudança para o modo de eixo | | | | | |
| MCALL | Chamada modal de subprogramas | | Sem o nome de subrotina: Desliga | | | |
| MEAC | Medição contínua sem cancelar percurso restante | Inteiro, sem sinal | | | s | |
| MEAFRAME | Cálculo do Frame através de pontos medidos | FRAME | | | | |
| MEAS | Medição disparada por apalpador | Inteiro, sem sinal | | | S | |
| MEASA | Medição com cancelamento do percurso | | | | s | |

| Funções G | | | | | | |
|------------------------|---|---|--|-------------------------------------|---|----|
| | restante | | | | | |
| MEAW | Medição disparada por apalpador sem cancelamento do percurso restante | Inteiro, sem sinal | | | S | |
| MEAWA | Medição sem cancelamento da distância a percorrer | | | | s | |
| MI | Acesso aos dados frame: espelhamento | | | | | |
| MINDEX | Define um índice para um carácter de entrada | 0, ..., INT | String: parâmetro 1, carácter, par.2 | | | |
| MIRROR | Espelhamento programável | | | MIRROR X0 Y0 Z0 ; bloco separado | s | 3 |
| MMC | Comando para MMC | STRING | | | | |
| MOD | Divisão cem módulo | | | | | |
| MOV | Inicia posicionamento de eixo (start moving positioning axis) | Real | | | | |
| MSG | Mensagens programáveis | | | MSG("mensagem") | | |
| N | Número de bloco – bloco secundário | 0, ..., 9999 9999 inteiros somente, sem sinal | Pode ser utilizado para identificar blocos através de números, escritos no início do bloco | e.g. N20 | | |
| NCK | Especifica uma faixa válida para os dados | | Uma por NCK | | | |
| NEWCONF | Aceitar o novo dados de máquina alterados | | | | | |
| NEWT | Criar uma nova ferramenta | | O número duplo pode ser omitido | | | |
| NORM ¹ | Ajuste normal no ponto inicial, ponto final na compensação de ferramenta | | | | m | 17 |
| NOT | NOT binário (negação) | | | | | |
| NPROT | Liga/desliga zona de proteção ON/OFF | | | | | |
| NPROTDEF | Define área de proteção da máquina | | | | | |
| NUMBER | Converte string em número | | Real | | | |
| OEMIP01 ^{6,8} | Interpolação OEM 1 | | | | m | 1 |
| OEMIP02 ^{6,8} | Interpolação OEM 2 | | | | m | 1 |
| OF | Instrução para ramificação CASE | | | | | |
| OFFN | Tolerância para o contorno programado | | | OFFN=5 | | |
| OMA1 ⁶ | OEM-Endereço 1 | Real | | | m | |
| OMA2 ⁶ | OEM-Endereço 2 | Real | | | m | |
| OMA3 ⁶ | OEM-Endereço 3 | Real | | | m | |
| OMA4 ⁶ | OEM-Endereço 4 | Real | | | m | |
| OMA5 ⁶ | OEM-Endereço 5 | Real | | | m | |
| OFFN | Correção Offset - normal | Real | | | m | |
| OR | OR lógico | | | | | |
| ORIC ^{1,6} | Alterações da orientação em cantos externos são superpostas ao bloco de círculo a inserir (orient. change continuously) | | | | m | 27 |
| ORID ⁶ | Alterações da orientação são executadas antes do bloco | | | | m | 27 |

| Funções G | | | | | |
|---------------------|---|------|-----------------------|---|----|
| | de círculo (orientation change discontinuously) | | | | |
| ORIEULER | Ângulo de orientação Euler | | | m | 50 |
| ORIMACHAX | Interpolação linear de eixos de usinagem ou eixos de orientação | | | m | 51 |
| ORIMCS ⁶ | Orientação da ferramenta no sistema de coordenadas da máquina | | | m | 25 |
| ORIRPY | Ângulos de orientação utilizando ângulos RPY | | | m | 50 |
| ORIS ⁵ | Alteração da orientação (orientation smoothing factor) | Real | referida à trajetória | m | |
| ORIVIRT1 | Ângulo de orientação através dos eixos virtuais de orientação (definição 1) | | | m | 50 |
| ORIVIRT2 | Ângulo de orientação através dos eixos virtuais de orientação (definição 1) | | | m | 50 |
| ORIVIRTAX | Interpolação circular grande | | | | |

| Funções G | | | | | | |
|--------------------------|--|--------------------------------|---------------------------------|---------------------------------|---|----|
| ORI WCS ^{1,6} | Orientação da ferramenta no sistema de coordenadas da peça | | | | m | 25 |
| OS | Ligar/desligar oscilação | Inteiro, sem sinal | | | | |
| OSC ⁶ | Alisamento constante da orientação da ferramenta | | | | m | 34 |
| OSCILL | Atribuição de eixos para o movimento oscilante - ligar movimento oscilante | | Eixos de 1 a 3: eixos de avanço | | m | |
| OSCTRL | Opções oscilação | Inteiro, sem sinal | | | M | |
| OSE | Oscilação: ponto final | | | | m | |
| OSNSC | Oscilação: número ciclos de chispar (oscillating: numere spark out cycles) | | | | m | |
| OSOF ^{1,6} | DESLIGAR alisamento da orientação da ferramenta | | | | m | 34 |
| OSP1 | Oscilação: ponto de reversão esquerdo (oscillating: Position 1) | Real | | | m | |
| OSP2 | Oscilação: ponto de reversão direito (oscillating: Position 2) | Real | | | m | |
| OSS ⁶ | Orientação suave da ferramenta no fim do bloco | | | | m | 34 |
| OSSE ⁶ | Orientação suave da ferramenta no fim e começo do bloco | | | | m | 34 |
| OST1 | Oscilação: parar no ponto de reversão esquerdo | Real | | | m | |
| OST2 | Oscilação: parar no ponto de reversão direito | Real | | | m | |
| OVR | Correção de avanço | 1, ..., 200% | | | m | |
| OVRA | Correção de avanço axial | 1, ..., 200% | | | m | |
| P | Quantidade de repetições de um subprograma | 1 ... 9999, inteiros sem sinal | | e.g. L781 P... ; bloco separado | | |
| PDELAY-OF ⁶ | DESL. retardamento no punção (Punch with DELAY OFF) | | | | m | 36 |
| PDELAY-ON ^{1,6} | LIG. retardamento no punção (Punch with DELAY ON) | | | | m | 36 |
| PL | Parâmetro de intervalo de comprimento | Real, sem sinal | | | S | |
| PM | Por minuto | | | Avanço por minuto | | |
| PO | Polinômio | Real, sem sinal | | | s | |
| POLF | Posição LIFTFAST | Real, sem sinal | | POLF[Y]=10 | m | |
| POLY ⁵ | Interpolação polinomial | | | | m | 1 |
| PON ⁶ | LIG. funcionamento (punch ON) | | | | m | 35 |
| PONS ⁶ | LIG. funcionamento (punch ON) no ciclo IPO (lento) | | | | m | 35 |
| POS | Posição do eixo | | | POS[X]=20 | | |
| POSA | Posição do eixo ao longo do bloco | | | POSA[Y]=20 | | |

| Funções G | | | | | | |
|-----------|---|---|--|---|---|----|
| POSP | Posicionamento em seções da peça (oscilação) | Real: Posição final, comprimento da peça; Inteiro: opção | | | | |
| POT | quadrado (função aritmética) | Real | | | | |
| PR | Por volta | | | Avanço por volta | | |
| PRESETON | Carrega valor atual para os eixos programados | | O nome de um eixo é programado com o valor correspondente no próximo parâmetro. Possível até 8 eixos | PRESETON(X,10,Y,4.5) | | |
| PRI0 | Instrução para atribuir prioridade no processo de interrupção | | | | | |
| PROC | Primeira instrução em um programa | | | Número do bloco – PROC – identificador | | |
| PTP | Ponto a ponto (point to point) | | | | m | 49 |
| PUTFTOC | PutFineToolCorrection: Correção da ferramenta fina para dressagem contínua | | | | | |
| PUTFTOCF | PutFineToolCorrectionFunctionDependent: Correção da ferramenta fina para dressagem contínua dependendo da função definida por FCtDEF | | | | | |
| PW | Peso de ponto (point weight) | Real, sem sinal | | | S | |
| QECLRNOF | Compensação de erro de quadrante – Desliga aprender | | | | | |
| QECLRNON | Compensação de erro de quadrante – Liga aprender | | | | | |
| QU | Função rápida adicional (auxiliar) | | | | | |
| R... | Parâmetros de cálculo SW 5 em diante: endereço parametrizável com extensão numérica | ± 0.0000001 , ..., 9999 9999 | Quantidade de parâmetros R definida através de MD | R10=3 ;atribuição de valor ao parâmetro R X=R10 ;valor para o eixo R[R10]=6 ;programação indireta. | | |
| RDISABLE | Bloqueio de execução do programa | | | | | |
| READAL | Alarme de leitura | | Os alarmes são pesquisados de acordo em ordem crescente (numérica) | | | |
| REAL | Tipo de dado: variável representada em ponto flutuante com sinal (números reais) | Corresponde m as formato ponto flutuante de 64 bits | | | | |
| REDEF | Valores para dados de máquinas, cujos grupos de usuários são exibidos | | | | | |

| Funções G | | | | | | |
|------------------|---|--|---|---|---|----|
| RELEASE | Libera eixos de máquina | | Múltiplos eixos podem ser programados | | | |
| REP | Instrução para inicialização de todos os elementos em um array com um mesmo valor | | | | | |
| REPEAT | Repetir um loop de programa | | Até (UNTIL) que a condição seja satisfeita | | | |
| REPEATB | Repetir a linha do programa | | nnn vezes | | | |
| REPOSA | Reposiciona os eixos linearmente | | | | s | 2 |
| REPOSH | Reposiciona em semi-círculo | | | | s | 2 |
| REPOSHA | Reaproximar-se do contorno com todos os eixos; eixos geométricos em semicírculo | | | | s | 2 |
| REPOSL | Reaproximar-se do contorno linearmente | | | | s | 2 |
| REPOSQ | Reaproximar-se do contorno em quarto de círculo | | | | s | 2 |
| REPOSQA | Reaproximar-se do contorno linearmente com todos os eixos; eixos geométricos em quarto de círculo | | | | s | 2 |
| RESET | Reseta o ciclo tecnológico | | Um ou vários lds podem ser programados | | | |
| RET | Fim de subprograma | | Utilização em vez de M2 - para a manter o modo de controle contínuo da trajetória | RET | | |
| RINDEX | Define índice para o carácter na string de entrada | 0, ..., INT | String: Parâmetro 1, carácter:Parâmetro 2 | | | |
| RMB | Reposicionamento no ponto inicial de bloco (Repos mode begin of block) | | | | m | 26 |
| RME | Reposicionamento no ponto final de bloco (mode end of block) | | | | m | 26 |
| RMI ¹ | Reposicionamento no ponto de interrupção (Repos mode interrupt) | | | | m | 26 |
| RND | Arredondar canto de contorno | Real, sem sinal | | RND=... | s | |
| RNDM | Arredondamento modal | Real, sem sinal | | RNDM=... RNDM=0: desliga M. V. | m | |
| ROT | Rotação programável (rotation) | Rotação em redor do 1.º eixo geom.: -180° .. 180° 2.º eixo geom.: -89.999°, ..., 90° 3.º eixo geom.: -180° .. 180° | | ROT X... Y... Z... ROT RPL= ; bloco separado | s | 3 |
| ROUND | Arredondamento de casas decimais | Real | | | | |

| Funções G | | | | | | |
|-----------|--|-----------------------|--|--|------------------|----|
| RP | Raio polar (radius polar) | Real | | | m,s ³ | |
| RPL | Rotação no plano (rotation plane) | Real, sem sinal | | | S | |
| RT | Parâmetro para acesso a dados frame: Rotação | | | | | |
| S | Velocidade de rotação do fuso ou (em G4, G96) outro significado | 0.1 ... 99999999.9 | Velocidade de rotação do fuso em rot/min G4: tempo de demora em rotações do fuso G96: velocidade de corte em m/min | S...: Velocidade de rotação para o fuso mestre S1...: Velocidade de rotação para fuso 1 | m, s | |
| SAVE | Atributo para salvar informações em chamadas de subrotinas | | O seguinte é salvo: Todas as funções G modais e o frame atual | | | |
| SBLOF | Suprime bloco a bloco (OFF) | | Os próximos blocos serão executados como um só em bloco a bloco | | | |
| SBLON | Cancela supressão de bloco a bloco (ON) | | | | | |
| SC | Parâmetro de acesso a dados frame: escala | | | | | |
| SCALE | Alteração de escala programável (scale) | | | SCALE X... Y... Z... ; bloco separado | s | 3 |
| SD | Grau de Spline (spline degree) | Inteiro, sem sinal | | | S | |
| SET | Instrução para carga dos elementos de um array com os valores listados | | | | | |
| SETAL | Liga alarme | | | | | |
| SETDNO | Carrega o número D da ferramenta (T) e seu corretor com um novo valor | | | | | |
| SETINT | Define qual rotina de interrupção será ativada quando a entrada NCK estiver ligada | | Avaliada a transição 0 → 1 | | | |
| SETM | Liga uma ou mais memórias para coordenação do canal | | A usinagem no canal; local não é influenciada. | | | |
| SETMS | Retorno ao fuso mestre especificado no dado de máquina | | | | | |
| SETMS(n) | O fuso n deve valer como fuso mestre | | | | | |
| SETPIECE | Incrementa o contador de peças para todas as ferramentas que foram carregadas no fuso. | | Sem definição de fuso: válido para o fuso mestre | | | |
| SF | Desvio do ponto inicial para abrir roscas (spline offset) | 0.0000, ..., 359.999° | | | m | |
| SIN | Seno (função trigonométrica) | Real | | | | |
| SOFT | Aceleração suave | | | | m | 21 |
| SOFTA | Comuta para aceleração suave do eixo para os eixos programados | | | | | |

| Funções G | | | | | | |
|--------------------------|--|---------------------|---|---|----|--|
| SON ⁶ | Ligar punção (stroke ON) | | | m | 35 | |
| SONS ⁶ | Ligar punção no ciclo de IPO (stroke ON slow) | | | m | 35 | |
| SPATH ¹ | Referência para eixos do FGROUPO em comprimento de arco | | | m | 45 | |
| SPCOF | Comutar fuso mestre ou fuso (n) da regulação da velocidade de rotação à regulação da posição | | SPCON SPCON (n) | | | |
| SPCON | Comutar fuso mestre ou fuso (n) da regulação da posição à regulação da velocidade de rotação | | SPCON SPCON (n) | | | |
| SPIF1 ^{1,6} | Entradas/saídas NCK rápidas para punção Byte 1 (stroke/punch interface 1) | | | m | 38 | |
| SPIF2 ⁶ | Entradas/saídas NCK rápidas para punção Byte 2 (stroke/punch interface 2) | | | m | 38 | |
| SPLINE-PATH ⁷ | Define agrupamento spline | Max. de 8 eixos | | | | |
| SPOF ^{1,6} | Desligar punção (stroke/punch OFF) | | | m | 35 | |
| SPN ⁶ | Número das distâncias parciais por bloco (stroke/punch numere) | Inteiro | | s | | |
| SPP ⁶ | Comprimento duma distância parcial (stroke/punch path) | Inteiro | | m | | |
| SPOS | Posição de fuso | | SPOS=10 ou SPOS[n]=10 | m | | |
| SPOSA | Posição do fuso para além de limites de bloco | | SPOSA=5 ou SPOSA[n]=5 | m | | |
| SQRT | Raiz quadrada, função aritmética | Real | | | | |
| SR | Percurso de retrocesso (sparking out retract path) | Real, sem sinal | | S | | |
| SRA | Percurso de retrocesso axial com entrada externa (sparking out retract) | | SRA[Y]=0.2 | m | | |
| ST | Tempo de chispar (sparking out time) | Real, sem sinal | | S | | |
| STA | Tempo de chispar axial (sparking out time axial) | | | m | | |
| STAT | Posição das juntas | Inteiro | | s | | |
| STARTFIFO ¹ | Execução; paral. a ela enchimento do buffer preliminar | | | m | 4 | |
| STOPFIFO | Parada do processamento; enchimento do buffer preliminar até que STARTFIFO for reconhecido, buffer preliminar cheio ou fim de programa | | | m | 4 | |
| STOPRE | Parada no pré processamento até que todos os blocos pré preparados tenham sido executados. | | | | | |
| STOPREOF | Desliga a parada de pré processamento | | | | | |
| STRING | Tipo de dado: String | Max. 200 caracteres | | | | |
| STRLEN | Define comprimento da string | INT | | | | |
| SUBSTR | Define índice do carácter numa string de entrada | Real | String: Parâmetro 1, carácter:parâmetro 2 | | | |
| SUPA | Supressão do deslocamento programável do ponto zero | incl. deslocamentos | | s | 9 | |

| Funções G | | | | | | |
|-----------|--|--------------|--|--|--|--|
| | | | programados, deslocamentos por manivela (DRF), deslocamento externo do ponto zero e deslocamento PRESET | | | |
| SYNFCT | Avaliação ode um polinômio como função de uma condição em movimentos com ações síncronas | VAR REAL | | | | |
| SYNR | A variável é lida de forma síncrona, ou seja, durante a execução (leitura síncrona) | | | | | |
| SYNRW | A variável é lida e escrita de forma síncrona, ou seja, durante a execução (leitura/escrita síncrona) | | | | | |
| SYNW | A variável é escrita de forma síncrona, ou seja, durante a execução (escrita síncrona) | | | | | |
| T | Chamar a ferramenta (trocar somente quando especificado no dado de máquina; caso contrário é necessário o comando M6) | 1 ... 32 000 | Chamada através do número T ou do designador de ferramenta : | p.ex. T3 ou T=3 p.ex. T="BROCA" | | |

| Funções G | | | | | | |
|---------------------|--|---------|--|--|---|----|
| TAN | Tangente (função trigonométrica) | Real | | | | |
| TANG | Determina a tangente para o modo sem supervisão em ambos os eixos principais especificados | | | | | |
| TANGOF | Desliga modo sem supervisão da malha de posição tangencial | | | | | |
| TANGON | Liga modo sem supervisão da malha de posição tangencial | | | | | |
| TCARR | Solicitar porta-ferramenta (número „m“) | Inteiro | m=0: desselecionar porta-ferramenta ativo | TCARR=1 | | |
| TCOABS ¹ | Determinar componentes de comprimento de ferramenta da orientação de ferramenta atual | | Necessário após reajuste, p.ex. por ajuste manual | | m | 42 |
| TCOFR | Determinar componentes de comprimento de ferramenta da orientação do frame ativo | | | | m | 42 |
| TILT ⁵ | Ângulo lateral | Real | | | m | |
| TMOF | Desselecionar monitoração da ferramenta | | n.º T só necessário se a ferramenta com este número não esteja ativa . | TMOF (T no.) | | |
| TMON | Selecionar monitoração da ferramenta | | T-Nr. = 0: desligar a monitoração para todas as ferramentas | TMON (T no.) | | |
| TO | Define o valor final para um loop FOR | | | | | |
| TOFRAME | Ajustar o frame programável atual para o sistema de coordenadas da ferramenta | | | | s | 3 |
| TOLOWER | Converte as letras de uma string em minúsculas | | | | | |
| TOUPPER | Converte as letras de uma string em maiúsculas | | | | | |
| TR | Parâmetro para acesso de dados frame: Translação | | | | | |
| TRAANG | Transformação de eixo inclinado | | Várias transformações são aceitas por canal | | | |
| TRACEOF | Teste de circularidade: Desliga a transferência de valores | | | | | |
| TRACEON | Teste de circularidade: Liga a transferência de valores | | | | | |
| TRACON | Transformação concatenada | | | | | |
| TRACYL | Cilindro: Transformação de superfície periférica | | vide TRAANG | | | |
| TRAFOOF | Desliga a transformação | | | TRAFOOF() | | |
| TRAILOF | Movimento de eixos acoplados – Desliga | | | | | |
| TRAILON | Movimento de eixos acoplados – Liga | | | | | |
| TRANS | Translação programável (translation) | | | TRANS X... Y... Z... ; bloco separado | s | 3 |
| TRANSMIT | Transformação polar | | vide TRAANG | | | |

| Funções G | | | | | | |
|----------------------|---|-------------|--|---|------------------|----|
| TRAORI | Transformação de 4 ou 5 eixos (transformação orientada) | | vide TRAANG | | | |
| TRUE | Constante lógica TRUE | BOOL | Pode ser substituída por constante inteira 1 | | | |
| TRUNC | Trunca casas decimais | Real | | | | |
| TU | Ângulo do eixo | Inteiro | | TU=2 | s | |
| TURN | Quantidade de voltas da hélice | 0, ..., 999 | | | s | |
| UNLOCK | Habilita a ação síncrona com ID (continua o ciclo tecnológico) | | | | | |
| UNTIL | Condição para o fim de um loop REPEAT | | | | | |
| UPATH | A referência de trajetória para os eixos do FGOUP é um parâmetro em curva | | | | m | 45 |
| VAR | Instrução: Tipo de parâmetro passado | | Com VAR: Chamada pela referência | | | |
| WAITC | Espera até que o critério para mudança de bloco seja preenchido (eixos acoplados, espera pela condição de acoplamento) | | Até 2 eixos/fusos podem ser programados | WAITM(1,1,2) | | |
| WAITM | Espera por marca em um canal específico, fim do bloco anterior com parada exata | | | WAITM(1,1,2) | | |
| WAITMC | Espera por marca em um canal específico, fim do bloco anterior com parada exata somente se os outros canais não atingiram ainda a marca | | | WAITMC(1,1,2) | | |
| WAITP | Espera fim do movimento | | | WAITP(X) ; bloco separado | | |
| WAITS | Esperar pela chegada à posição de fuso | | | WAITS (fuso principal) WAITS (n,n,n) | | |
| WALIMOF | DESL. limitação da área de trabalho (working area limitation OFF) | | | ; bloco separado | m | 28 |
| WALIMON ¹ | LIG. limitação da área de trabalho (working area limitation ON) | | | ; bloco separado | m | 28 |
| WHILE | Inicia um loop de programa WHILE | | Fim: ENDWHILE | | | |
| WRITE | Escreve bloco em arquivo de sistema | | | | | |
| X | Eixo | Real | | | m,s ³ | |
| XOR | OU exclusivo lógico | | | | | |
| Y | Eixo | Real | | | m,s ³ | |
| Z | Eixo | Real | | | m,s ³ | |

Legenda:

- ¹ Definição default no início de programa (na configuração de fábrica do controle, se não nada mais for programado).
- ² A numeração dos grupos corresponde à tabela “Lista das funções G/funções preparatórias” da seção 12.3
- ³ Pontos finais absolutos: efeito modal; pontos finais increm.: não modal; nos outros casos efeito modal/não modal dependendo da sintaxe da função G
- ⁴ Como centros de círculo, parâmetros de IPO têm efeito incrementador. Mediante AC, eles podem ser programados absolutamente. Em outros significados (p.ex. passo de rosca), a modificação de endereço será ignorada.
- ⁵ Palavra chave não é válida para SINUMERIK FM-NC/810D
- ⁶ Palavra chave não é válida para SINUMERIK FM-NC/810D/NCU571
- ⁷ Palavra chave não é válida para SINUMERIK 810D
- ⁸ O usuário OEM pode incluir dois tipos de interpolação adicionais. Os nomes podem ser alterados pelo usuário OEM.
- ⁹ Palavra chave só é válida para SINUMERIK FM-NC
- ¹⁰ Para estas funções, o formato estendido de endereço não é permitido

15.2 Lista das variáveis de sistema

Legenda:

| | |
|----------------|---|
| Prog.NC | Programa NC |
| Sínc | Ação síncrona |
| O | O índice pode ser calculado online em ações síncronas. (+) |
| S | Versão de software |
| R | Acesso de leitura possível |
| W | Acesso de escrita possível |
| RS | Uma parada no pré processamento é realizada implicitamente na leitura |
| WS | Uma parada no pré processamento é realizada implicitamente na escrita |
| + | Na coluna O: O índice pode ser calculado online em ações síncronas. |

15.2.1 Parâmetros R

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| R | REAL | Rn ou R[n] A quantidade de parâmetros R (máxima) é definida em dados de máquina | R | W | | 1 |
| \$R | | | | R | W | |

15.2.2 Frames 1

| | | | | | | |
|------------------|-------|---|---|---|--|---|
| \$P_UIFR | FRAME | \$P_UIFR[n] Frames ajustáveis, podem ser ativados através de G500, G54 .. G599. De 5 a 100 frames ajustáveis podem ser configurados através do MD \$MC_MM_NUM_USER_FRAMES. | R | W | | 2 |
| \$P_CHBFR | FRAME | \$P_CHBFR[n] Frames base do canal, podem ser ativados através de G500, G54 .. G599. De 0 a 8 frames base do canal podem ser configurados através do MD \$MC_MM_NUM_BASE_FRAMES. | R | W | | 5 |
| \$P_NCBFR | FRAME | \$P_NCBFR[n] Frames base NCU, podem ser ativados através de G500, G54 .. G599. De 0 a 8 frames base NCU podem ser configurados através do MD \$MN_MM_NUM_GLOBAL_BASE_FRAMES. | R | W | | 5 |

15.2.3 Dados do porta ferramenta

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$TC_CARR1 | REAL | \$TC_CARR1[n] Componente X do vetor de deslocamento l1 Atenção! Todos os parâmetros de sistema iniciados com o prefixo "\$TC_" estão contidos na área TOA. A característica especial desta área é que vários canais podem ter acesso a ela e seus parâmetros, quando o dado de máquina 28085 = MM_LINK_TOA_UNIT estiver adequadamente configurado. Caso o modo de parametrização seja escolhido para a NCK, você deve tomar cuidados especiais, já que alterações podem influenciar outros canais; isto é, você deve se certificar que as alterações estejam afetando somente o canal local. A quantidade máxima de porta ferramentas pode ser configurada através de dados de máquina. O valor default é = 0; ou seja, a NCI não possui tais dados. | R | W | | 4 |
| \$TC_CARR2 | REAL | \$TC_CARR2[n] Componente y do vetor de deslocamento l1 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | 4 |
| \$TC_CARR3 | REAL | \$TC_CARR3[n] Componente z do vetor de deslocamento l1 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | 4 |
| \$TC_CARR4 | REAL | \$TC_CARR4[n] Componente x do vetor de deslocamento l2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | 4 |
| \$TC_CARR5 | REAL | \$TC_CARR5[n] Componente y do vetor de deslocamento l2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | 4 |
| \$TC_CARR6 | REAL | \$TC_CARR6[n] Componente z do vetor de deslocamento l2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | 4 |
| \$TC_CARR7 | REAL | \$TC_CARR7[n] Componente x do eixo de rotação v1 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | 4 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog. | NC | Sínc | O | S |
|---------------|------|--|-------|----|------|---|---|
| \$TC_CARR8 | REAL | \$TC_CARR8[n] Componente y do eixo de rotação v1 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR9 | REAL | \$TC_CARR9[n] Componente z do eixo de rotação v1 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR10 | REAL | \$TC_CARR10[n] Componente x do eixo de rotação v2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR11 | REAL | \$TC_CARR11[n] Componente y do eixo de rotação v2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR12 | REAL | \$TC_CARR12[n] Componente z do eixo de rotação v2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR13 | REAL | \$TC_CARR13[n] Ângulo de rotação alfa1 (em graus) A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR14 | REAL | \$TC_CARR14[n] Ângulo de rotação alfa2 (em graus) A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 4 |
| \$TC_CARR15 | REAL | \$TC_CARR15[n] Componente x do vetor básico b A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 |
| \$TC_CARR16 | REAL | \$TC_CARR16[n] Componente y do vetor básico b A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 |
| \$TC_CARR17 | REAL | \$TC_CARR17[n] Componente z do vetor básico b A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 |

15.2 Lista das variáveis de sistema

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | | Sínc | O | S |
|---------------|------|--|---------|---|------|---|-------------|
| \$TC_CARR18 | REAL | \$TC_CARR18[n] Componente x do vetor de deslocamento l4 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 . 3 |
| \$TC_CARR19 | REAL | \$TC_CARR19[n] Componente y do vetor de deslocamento l4 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 . 3 |
| \$TC_CARR20 | REAL | \$TC_CARR20[n] Componente z do vetor de deslocamento l4 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 . 3 |
| \$TC_CARR21 | AXIS | \$TC_CARR21[n] identificador do eixo para eixo de rotação v1 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 . 3 |
| \$TC_CARR22 | AXIS | \$TC_CARR22[n] identificador do eixo para eixo de rotação v2 A quantidade máxima de porta ferramentas pode ser configurada através de MDs. Default =0, ou seja, a NCI não possui tais dados | R | W | | | 5 . 3 |
| \$TC_CARR23 | CHAR | \$TC_CARR23[n] Tipo cinemático 0:Alarme (14153) n:Alarme(14153) Opções possíveis: T: somente a ferramenta pode girar (default) P: somente a peça pode girar M: ferramenta e peça podem girar (modo misto) Letras maiúsculas e minúsculas podem ser utilizadas. | R | W | | | 5 . 3 |

15.2.4 Zonas de proteção específicas do canal

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|----------------------------|------|--|---------|------|---|---|
| \$SC_PA_ACTIV_IMMED | BOOL | \$SC_PA_ACTIV_IMMED[n] Zona de proteção ativa imediatamente? TRUE: A zona de proteção foi ativada imediatamente coma a ligação do comando e com o referenciamento dos eixos FALSE: A zona de proteção não encontra-se ativa n: número da zona de proteção 0 – (valor máximo em MD) | R | W | | 2 |
| \$SC_PA_T_W | CHAR | \$SC_PA_T_W[n] Zona de proteção orientada à peça/ferramenta 0: Zona orientada para a peça 3: Zona orientada à ferramenta n: Número da zona de proteção 0 – (Valor máximo ajustado através de dados de máquina) | R | W | | 2 |
| \$SC_PA_ORI | INT | \$SC_PA_ORI[n] Orientação da zona de proteção 0: Polígono no plano dos eixos geométricos 1 e 2 1: Polígono no plano dos eixos geométricos 3 e 1 2: Polígono no plano dos eixos geométricos 2 e 3 n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SC_PA_LIM_3DIM | INT | \$SC_PA_LIM_3DIM[n] Código para restringir a zona de proteção no eixo paralelo à definição do polígono 0: = Sem limite 1: = Limite na direção positiva 2: = Limite na direção negativa 3: = Limite em ambas direções n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SC_PA_PLUS_LIM | REAL | \$SC_PA_PLUS_LIM[n] Limite positivo para a zona de proteção no eixo perpendicular à definição do polígono n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SC_PA_MINUS_LIM | REAL | \$SC_PA_MINUS_LIM[n] Limite positivo para a zona de proteção na direção negativa perpendicular à definição do polígono n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SC_PA_CONT_NUM | INT | \$SC_PA_CONT_NUM[n] Quantidade de elementos de contorno válidos n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|------------------------------|------|---|---------|------|---|---|
| \$SC_PA_CONT_T YP | INT | \$SC_PA_CONT_TYP[n,m] Tipo de elemento de contorno (G1, G2, G3) n: Nr. da zona de proteção 0–(valor máximo pode ser definido via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SC_PA_CONT_ ORD | REAL | \$SC_PA_CONT_ORD[n,m] Ponto final do elemento de contorno (ordenada) n: Nr. da zona de proteção 0–(valor máximo pode ser definido via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SC_PA_CONT_ ABS | REAL | \$SC_PA_CONT_ABS[n,m] Ponto final do elemento de contorno (abscissa) n: Nr. da zona de proteção 0–(valor máximo pode ser definido via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SC_PA_CENT_ ORD | REAL | \$SC_PA_CENT_ORD[n,m] Ponto central do elemento de contorno (ordenada) n: Nr. da zona de proteção 0–(valor máximo pode ser definido via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SC_PA_CENT_A BS | REAL | \$SC_PA_CENT_ABS[n,m] Ponto central do elemento de contorno (abscissa) n: Nr. da zona de proteção 0–(valor máximo pode ser definido via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |

15.2.5 Parâmetros de ferramenta

| | | | | | | |
|-----------------|------|--|---|---|--|---|
| \$TC_DP1 | INT | \$TC_DP1[t,d] Tipo da ferramenta Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP1[d] t: Número T 1–32000 d: Número do incerto/Número D 1–9 | R | W | | 2 |
| \$TC_DP2 | REAL | \$TC_DP2[t,d] Posição do incerto na ferramenta Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP2[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP3 | REAL | \$TC_DP3[t,d] geometria - Comprimento 1 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP3[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$TC_DP4 | REAL | \$TC_DP4[t,d] geometria - Comprimento 2 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP4[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP5 | REAL | \$TC_DP5[t,d] geometria - Comprimento 3 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP5[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP6 | REAL | \$TC_DP6[t,d] Geometria - Raio Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP6[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP7 | REAL | \$TC_DP7[t,d] Serra de encaixe: raio do canto Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP7[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP8 | REAL | \$TC_DP8[t,d] Serra de encaixe: Comprimento Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP8[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP9 | REAL | \$TC_DP9[t,d] Reservado Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP9[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$TC_DP10 | REAL | \$TC_DP10[t,d] Ângulo entre a face da ferramenta e a superfície da saliência Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP10[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP11 | REAL | \$TC_DP11[t,d] Ângulo entre o eixo longitudinal da ferramenta e a superfície final da face da saliência Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP11[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP12 | REAL | \$TC_DP12[t,d] Desgaste - Comprimento 1 – \$TC_DP3 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP12[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP13 | REAL | \$TC_DP13[t,d] Desgaste - Comprimento 2 – \$TC_DP4 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP13[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP14 | REAL | \$TC_DP14[t,d] Desgaste - Comprimento 3 – \$TC_DP5 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP14[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP15 | REAL | \$TC_DP15[t,d] Desgaste - Raio – \$TC_DP6 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP15[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP16 | REAL | \$TC_DP16[t,d] Serra de encaixe: Desgaste - Raio do canto – \$TC_DP7 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP16[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$TC_DP17 | REAL | \$TC_DP17[t,d] Serra de encaixe: Desgaste - Comprimento – \$TC_DP8 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP17[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP18 | REAL | \$TC_DP18[t,d] Desgaste – Reservado – \$TC_DP9 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP18[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP19 | REAL | \$TC_DP19[t,d] Desgaste – Ângulo entre a face da ferramenta e a superfície do ressalto – \$TC_DP10 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP19[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP20 | REAL | \$TC_DP20[t,d] Desgaste – Ângulo entre o eixo longitudinal da ferramenta e a superfície final do ressalto – \$TC_DP11 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP20[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP21 | REAL | \$TC_DP21[t,d] Base - Comprimento 1 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP21[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP22 | REAL | \$TC_DP22[t,d] Base – Comprimento 2 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP22[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |

15.2 Lista das variáveis de sistema

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|-------------|
| \$TC_DP23 | REAL | \$TC_DP23[t,d] Base – Comprimento 3 Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP23[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP24 | REAL | \$TC_DP24[t,d] Ângulo de alívio Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP24[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DP25 | REAL | \$TC_DP25[t,d] Reservado Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DP25[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 4 |
| \$TC_DPCE | INT | \$TC_DPCE[t,d] = 'Nr. do incerto' do bloco de dados de correção t,d Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPCE[d] CE significa <C>utting<E>dge (dado do incerto) t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_DPH | INT | \$TC_DPH[t,d] = 'H número do incerto' do bloco de dados de correção t,d for Fanuc0 M Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPH[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 . 1 |

Dados de inserto do usuário OEM

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-------------------|------|--|---------|------|---|-------------|
| \$TC_DPC1 | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPC1[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPC1[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DPC2 | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPC2[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPC2[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DPCi | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPCi[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPCi[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DPC10 | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPC10[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPC10[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_DPCS1 | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPCS1[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPCS1[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 . 2 |
| \$TC_DPCS2 | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPCS2[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPCS2[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 . 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|------------------------------|------|--|---------|------|---|-------------|
| \$TC_DPCS_i | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPCS _i [t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPCS _i [d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 . 2 |
| \$TC_DPCS10 | REAL | O tipo pode ser definido em dados de máquina. O default é REAL \$TC_DPCS10[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_DPCS10[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 . 2 |
| \$TC_SCP13 | REAL | Correção p/ \$TC_DP3: \$TC_SCP13[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP13[d] t: Número T 1–32000 d: Número do incerto/Número D 1– 32000 | R | W | | 5 |
| \$TC_SCP14 | REAL | Correção p/ \$TC_DP4: \$TC_SCP14[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP14[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | ... | ... | | | | |
| \$TC_SCP21 | REAL | Correção p/ \$TC_DP11: \$TC_SCP21[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP21[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP23 | REAL | Correção p/ \$TC_DP3: \$TC_SCP23[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP23[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP24 | REAL | Correção p/ \$TC_DP4: \$TC_SCP24[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP24[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-------------------|------|--|---------|------|---|---|
| ... | ... | ... | | | | |
| \$TC_SCP31 | REAL | Correção p/ \$TC_DP11: \$TC_SCP31[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP31[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP33 | REAL | Correção p/ \$TC_DP3: \$TC_SCP33[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP33[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP34 | REAL | Correção p/ \$TC_DP4: \$TC_SCP34[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP34[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | ... | ... | | | | |
| \$TC_SCP41 | REAL | Correção p/ \$TC_DP11: \$TC_SCP41[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP41[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP43 | REAL | Correção p/ \$TC_DP3: \$TC_SCP43[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP43[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP44 | REAL | Correção p/ \$TC_DP4: \$TC_SCP44[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP44[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | ... | ... | | | | |
| \$TC_SCP51 | REAL | Correção p/ \$TC_DP11: \$TC_SCP51[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP51[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-------------------|------|--|---------|------|---|---|
| \$TC_SCP53 | REAL | Correção p/ \$TC_DP3: \$TC_SCP53[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP53[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP54 | REAL | Correção p/ \$TC_DP4: \$TC_SCP54[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP54[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | | ... | | | | |
| \$TC_SCP61 | REAL | Correção p/ \$TC_DP11: \$TC_SCP61[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP61[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP63 | REAL | Correção p/ \$TC_DP3: \$TC_SCP63[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP63[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_SCP64 | REAL | Correção p/ \$TC_DP4: \$TC_SCP64[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP64[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | | ... | | | | |
| \$TC_SCP71 | REAL | Correção p/ \$TC_DP11: \$TC_SCP71[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_SCP71[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-------------------|------|--|---------|------|---|---|
| \$TC_ECP13 | REAL | Correção p/ \$TC_DP3: \$TC_ECP13[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP13[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP14 | REAL | Correção p/ \$TC_DP4: \$TC_ECP14[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP14[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | | ... | | | | |
| \$TC_ECP21 | REAL | Correção p/ \$TC_DP11: \$TC_ECP21[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP21[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP23 | REAL | Correção p/ \$TC_DP3: \$TC_ECP23[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP23[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP24 | REAL | Correção p/ \$TC_DP4: \$TC_ECP24[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP24[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | | ... | | | | |
| \$TC_ECP31 | REAL | Correção p/ \$TC_DP11: \$TC_ECP31[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP31[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-------------------|------|--|---------|------|---|---|
| \$TC_ECP33 | REAL | Correção p/ \$TC_DP3: \$TC_ECP33[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP33[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP34 | REAL | Correção p/ \$TC_DP4: \$TC_ECP34[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP34[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | | ... | | | | |
| \$TC_ECP41 | REAL | Correção p/ \$TC_DP11: \$TC_ECP41[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP41[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP43 | REAL | Correção p/ \$TC_DP3: \$TC_ECP43[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP43[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP44 | REAL | Correção p/ \$TC_DP4: \$TC_ECP44[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP44[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| ... | | ... | | | | |
| \$TC_ECP51 | REAL | Correção p/ \$TC_DP11: \$TC_ECP51[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP51[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |
| \$TC_ECP53 | REAL | Correção p/ \$TC_DP3: \$TC_ECP53[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP53[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog. | NC | Sínc | O | S |
|-------------------|------|--|-------|----|------|---|---|
| \$TC_ECP54 | REAL | Correção p/ \$TC_DP4: \$TC_ECP54[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP54[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| ... | | ... | | | | | |
| \$TC_ECP61 | REAL | Correção p/ \$TC_DP11: \$TC_ECP61[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP61[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| \$TC_ECP63 | REAL | Correção p/ \$TC_DP3: \$TC_ECP63[t,d] análogo à \$TC_DP12[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP63[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| \$TC_ECP64 | REAL | Correção p/ \$TC_DP4: \$TC_ECP64[t,d] análogo à \$TC_DP13[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP64[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| ... | | ... | | | | | |
| \$TC_ECP71 | REAL | Correção p/ \$TC_DP11: \$TC_ECP71[t,d] análogo à \$TC_DP20[t,d] Quando ativada a função 'Gerenciamento plano do número D', a sintaxe é a seguinte: \$TC_ECP71[d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |

15.2.6 Dados de monitoração para o gerenciamento de ferramentas

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog. | NC | Sínc | O | S |
|---------------|------|---|-------|----|------|---|---|
| \$TC_MOP1 | REAL | \$TC_MOP1[t,d] Limite de pré alarme para vida útil t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 2 |
| \$TC_MOP2 | REAL | \$TC_MOP2[t,d] Vida útil restante t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 2 |
| \$TC_MOP3 | INT | \$TC_MOP3[t,d] Limite de pré alarme para quantidade de peças t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 2 |
| \$TC_MOP4 | INT | \$TC_MOP4[t,d] Valore restante para quantidade de peças t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 2 |
| \$TC_MOP5 | REAL | \$TC_MOP5[t,d] Limite de pré alarme de desgaste t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| \$TC_MOP6 | REAL | \$TC_MOP6[t,d] Desgaste restante t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| \$TC_MOP11 | REAL | \$TC_MOP11[t,d] Valor inicial de vida útil t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| \$TC_MOP13 | INT | \$TC_MOP13[t,d] Valor inicial de contagem de peças t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |
| \$TC_MOP15 | REAL | \$TC_MOP15[t,d] Valor inicial de desgaste t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | | 5 |

15.2.7 Dados de monitoração para usuários OEM

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|-------------|
| \$TC_MOPC1 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MOPC1[t,d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_MOPC2 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MOPC2[t,d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| ... | | ... | | | | |
| \$TC_MOPC10 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MOPC10[t,d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 2 |
| \$TC_MOPCS1 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MOPCS1[t,d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 · 2 |
| \$TC_MOPCS2 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MOPCS2[t,d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 · 2 |
| ... | | ... | | | | |
| \$TC_MOPCS10 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MOPCS10[t,d] t: Número T 1–32000 d: número do incerto/número D 1–32000 | R | W | | 5 · 2 |

15.2.8 Dados relacionados à ferramenta

| | | | | | | |
|----------|------------|--|---|---|--|---|
| \$TC_TP1 | INT | \$TC_TP1[t] Número duplo t: Número T 1–32000 | R | W | | 2 |
| \$TC_TP2 | STRIN G | \$TC_TP2[t] Nome da ferramenta t: Número T 1–32000 | R | W | | 2 |
| \$TC_TP3 | INT | \$TC_TP3[t] Tamanho à esquerda t: Número T 1–32000 | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | | Sínc | O | S |
|---------------|------|---|---------|---|------|---|-------------|
| \$TC_TP4 | INT | \$TC_TP4[t] Tamanho à direita t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP5 | INT | \$TC_TP5[t] Tamanho acima t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP6 | INT | \$TC_TP6[t] Tamanho abaixo t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP7 | INT | \$TC_TP7[t] Tipo de alojamento do magazine t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP8 | INT | \$TC_TP8[t] Estado t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP9 | INT | \$TC_TP9[t] Tipo de monitoração da ferramenta t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP11 | INT | \$TC_TP11[t] Estratégia de troca t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TP10 | INT | \$TC_TP10[t] Informações da ferramenta t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TPC1 | REAL | O tipo pode ser definido em dados de máquina. O default é INT \$TC_TPC1[t] t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TPC2 | REAL | O tipo pode ser definido em dados de máquina. O default é INT \$TC_TPC2[t] t: Número T 1–32000 | R | W | | | 2 |
| ... | ... | ... | | | | | |
| \$TC_TPC10 | REAL | O tipo pode ser definido em dados de máquina. O default é INT \$TC_TPC10[t] t: Número T 1–32000 | R | W | | | 2 |
| \$TC_TPCS1 | REAL | O tipo pode ser definido em dados de máquina. O default é INT \$TC_TPCS1[t] t: Número T 1–32000 | R | W | | | 5 . 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|-------------|
| \$TC_TPCS2 | REAL | O tipo pode ser definido em dados de máquina. O default é INT \$TC_TPCS2[t] t: Número T 1–32000 | R | W | | 5 . 2 |
| ... | | ... | | | | |
| \$TC_TPCS10 | REAL | O tipo pode ser definido em dados de máquina. O default é INT \$TC_TPCS10[t] t: Número T 1–32000 | R | W | | 5 . 2 |

15.2.9 Dados de ferramenta relativos à retíficas

| | | | | | | | | |
|-----------|------|--|---|---|--|--|--|---|
| \$TC_TPG1 | INT | \$TC_TPG1[t] Número do fuso t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG2 | INT | \$TC_TPG2[t] Regra de encadeamento t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG3 | REAL | \$TC_TPG3[t] Raio mínimo do rebolo t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG4 | REAL | \$TC_TPG4[t] Largura mínima do rebolo t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG5 | REAL | \$TC_TPG5[t] Largura atual do rebolo t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG6 | REAL | \$TC_TPG6[t] Máxima rotação t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG7 | REAL | \$TC_TPG7[t] Máxima velocidade periférica t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG8 | REAL | \$TC_TPG8[t] Ângulo de inclinação para rebolo oblíquo t: Número T 1–32000 | R | W | | | | 2 |
| \$TC_TPG9 | INT | \$TC_TPG9[t] Quantidade de parâmetros para cálculo do raio t: Número T 1–32000 | R | W | | | | 2 |

15.2.10 Dados de alojamento do magazine

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$TC_MPP1 | INT | \$TC_MPP1[n,m] Classe do alojamento n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPP2 | INT | \$TC_MPP2[n,m] Tipo do alojamento n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPP3 | BOOL | \$TC_MPP3[n,m] Ativa/desativa considerar alojamento adjacente n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPP4 | INT | \$TC_MPP4[n,m] Estado do alojamento n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPP5 | INT | \$TC_MPP5[n,m] Magazine buffer: índice da classe do alojamento Magazines reais: Número do grupo de desgaste n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPP6 | INT | \$TC_MPP6[n,m] Número T da ferramenta neste alojamento n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPP7 | INT | \$TC_MPP7[n,m] Número do adaptador da ferramenta neste alojamento n: Número físico do magazine m: Número físico do alojamento | R | W | | 5 |

15.2.11 Dados de magazine para usuários OEM

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|-------------|
| \$TC_MPPC1 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MPPC1[n,m] n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPPC2 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MPPC2[n,m] n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| ... | | ... | | | | |
| \$TC_MPPC10 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MPPC10[n,m] n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MPPCS1 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MPPCS1[n,m] n: Número físico do magazine m: Número físico do alojamento | R | W | | 5 . 2 |
| \$TC_MPPCS2 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MPPCS2[n,m] n: Número físico do magazine m: Número físico do alojamento | R | W | | 5 . 2 |
| ... | | ... | | | | |
| \$TC_MPPCS10 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MPPCS10[n,m] n: Número físico do magazine m: Número físico do alojamento | R | W | | 5 . 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$TC_MDP1 | INT | \$TC_MDP1[n,m] Distância entre a posição de troca do magazine n alojamento m do 1º magazine interno Parâmetro de distância do 1º magazine interno n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MDP2 | INT | \$TC_MDP2[n,m] Distância entre a posição de troca do magazine n alojamento m do 2º magazine interno Parâmetro de distância do 2º magazine interno n: Número físico do magazine m: Número físico do alojamento | R | W | | 2 |
| \$TC_MLSR | INT | \$TC_MLSR[n,m]=0 Relação entre os alojamentos buffer n e m m deve identificar um alojamento de tipo 'fuso' n deve identificar um alojamento de tipo diferente de 'fuso'. Isto permite que a definição de garras relacionadas a fusos. O valor do parâmetro fix = 0. O processo de escrita define uma relação, o processo de leitura checa quando uma relação particular é aplicada. Caso não, um alarme será produzido durante a operação de leitura. Define conexões entre garras,... fusos. n: número do alojamento físico do magazine de classe diferente de SPINDLE m: número do alojamento físico de classe idêntica à SPINDLE | R | W | | 3 |
| \$TC_MPTH | INT | \$TC_MPTH[n,m] Tipo de hierarquia do alojamento no magazine Alojamento do magazine (lugar) parâmetro de tipos de hierarquia n: Hierarquia 0 – 7 m: Tipo de alojamento 0 – 7 | R | W | | 3 |

15.2.12 Dados de descrição do magazine para o gerenciamento de ferramentas

| | | | | | | |
|-----------|------------|---|---|---|--|---|
| \$TC_MAP2 | STRIN G | \$TC_MAP2[n] Identificador do magazine n: Magazine número 1 a ... | R | W | | 2 |
| \$TC_MAP1 | INT | \$TC_MAP1[n] Tipo de magazine n: Magazine número 1 a ... | R | W | | 2 |
| \$TC_MAP3 | INT | \$TC_MAP3[n] Estado do magazine n: Magazine número 1 a ... | R | W | | 2 |
| \$TC_MAP4 | INT | \$TC_MAP4[n] Encadeamento com o próximo magazine n: Magazine número 1 a ... | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog. | NC | Sínc | O | S |
|---------------|------|---|-------|----|------|---|---|
| \$TC_MAP5 | INT | \$TC_MAP5[n] Encadeamento com o magazine anterior n: Magazine número 1 a ... | R | W | | | 2 |
| \$TC_MAP6 | INT | \$TC_MAP6[n] Quantidade de linhas n: Magazine número 1 a ... | R | W | | | 2 |
| \$TC_MAP7 | INT | \$TC_MAP7[n] Quantidade de colunas n: Magazine número 1 a ... | R | W | | | 2 |
| \$TC_MAP8 | INT | \$TC_MAP8[n] Posição atual do magazine com referência à posição de troca n: Magazine número 1 a ... | R | W | | | 2 |
| \$TC_MAP9 | INT | \$TC_MAP9[n] Número do grupo atual de desgaste n: Magazine número 1 a ... | R | W | | | 5 |

15.2.13 Descrição do magazine de ferramentas para usuários OEM

| | | | | | | | |
|-------------|-----|---|---|---|--|--|---|
| \$TC_MAPC1 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MAPC1[n] n: Magazine número 1 a ... | R | W | | | 2 |
| \$TC_MAPC2 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MAPC2[n] n: Magazine número 1 a ... | R | W | | | 2 |
| ... | ... | ... | | | | | |
| \$TC_MAPC10 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MAPC10[n] n: Magazine número 1 a ... | R | W | | | 2 |

| | | | | | | | |
|--------------|-----|--|---|---|--|--|-------------|
| \$TC_MAPCS1 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MAPCS1[n] n: Magazine número 1 a ... | R | W | | | 5 . 2 |
| \$TC_MAPCS2 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MAPCS2[n] n: Magazine número 1 a ... | R | W | | | 5 . 2 |
| ... | ... | ... | | | | | |
| \$TC_MAPCS10 | INT | O tipo pode ser definido em dados de máquina. O default é INT \$TC_MAPCS10[n] n: Magazine número 1 a ... | R | W | | | 5 . 2 |

15.2.14 Parâmetro de módulo do magazine

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------------|--|---------|------|---|---|
| \$TC_MAMP1 | STRIN G | \$TC_MAMP1 Identificador de módulo do magazine Variável escalar | R | W | | 2 |
| \$TC_MAMP2 | INT | \$TC_MAMP2 Tipo de pesquisa de ferramenta Variável escalar | R | W | | 2 |
| \$TC_MAMP3 | INT | \$TC_MAMP3 Manuseio de ferramentas com grupos de desgaste Variável escalar | R | W | | 5 |

Dados do adaptador

| | | | | | | |
|------------|------|--|---|---|--|---|
| \$TC_ADPTT | INT | \$TC_ADPTT[a] Número de transformação do adaptador a: Número do adaptador 1–32000 | R | W | | 5 |
| \$TC_ADPT1 | REAL | \$TC_ADPT1[a] Geometria do adaptador: Comprimento 1 a: Número do adaptador 1–32000 | R | W | | 5 |
| \$TC_ADPT2 | REAL | \$TC_ADPT2[a] Geometria do adaptador: Comprimento 2 a: Número do adaptador 1–32000 | R | W | | 5 |
| \$TC_ADPT3 | REAL | \$TC_ADPT3[a] Geometria do adaptador: Comprimento 3 a: Número do adaptador 1–32000 | R | W | | 5 |

15.2.15 Valores de compensação do sistema de medição

| | | | | | | |
|--------------------|------|--|---|---|--|---|
| \$AA_ENC_COMP | REAL | \$AA_ENC_COMP[n,m,a] Valores de compensação a: Eixo da máquina n: Encoder nr. 0–1 m: Ponto nr. 0 – <Valor MD> Eixos: Eixos de máquina | R | W | | 2 |
| \$AA_ENC_COMP_STEP | REAL | \$AA_ENC_COMP_STEP[n,a] Largura do passo a: Eixo da máquina n: Encoder nr. 0–1 Eixos: Eixos de máquina | R | W | | 2 |
| \$AA_ENC_COMP_MIN | REAL | \$AA_ENC_COMP_MIN[n,a] Ponto de compensação inicial a: Eixo da máquina n: Encoder nr. 0–1 Eixos: Eixos de máquina | R | W | | 2 |
| \$AA_ENC_COMP_MAX | REAL | \$AA_ENC_COMP_MAX[n,a] Ponto de compensação final a: Eixo da máquina n: Encoder nr. 0–1 Eixos: Eixos de máquina | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|--------------------------------|------|--|---------|------|---|---|
| \$AA_ENC_COMP_IS_MODULO | BOOL | \$AA_ENC_COMP_IS_MODULO[n,a] Compensação em módulo a: Eixo da máquina n: Encoder nr. 0–1 Eixos: Eixos de máquina | R | W | | 2 |

15.2.16 Compensação de erro de quadrante

| | | | | | | |
|------------------------------|------|---|---|---|--|---|
| \$AA_QEC | REAL | \$AA_QEC[n,m,a] Resultado do processo 'aprender' a: Eixo da máquina n: 0 m: Nr. do ponto: 0 – \$MN_MM_QEC_MAX_POINTS | R | W | | 2 |
| \$AA_QEC_COARSE_STEPS | INT | \$AA_QEC_COARSE_STEPS[n,a] Valor de compensação: característica de equalização grossa a: Eixo da máquina n: 0 | R | W | | 2 |
| \$AA_QEC_FINE_STEPS | INT | \$AA_QEC_FINE_STEPS[n,a] característica de equalização fina a: Eixo da máquina n: 0 | R | W | | 2 |
| \$AA_QEC_ACCEL_1 | REAL | \$AA_QEC_ACCEL_1[n,a] Aceleração na 1ª curva de acordo com a definição atual [mm/s ² ou inch/s ² ou graus/s ²] a: Eixo da máquina n: 0 | R | W | | 2 |
| \$AA_QEC_ACCEL_2 | REAL | \$AA_QEC_ACCEL_2[n,a] Aceleração na 2ª curva de acordo com a definição atual [mm/s ² ou inch/s ² ou graus/s ²] a: Eixo da máquina n: 0 | R | W | | 2 |
| \$AA_QEC_ACCEL_3 | REAL | \$AA_QEC_ACCEL_3[n,a] Aceleração na 3ª curva de acordo com a definição atual [mm/s ² ou inch/s ² ou graus/s ²] a: Eixo da máquina n: 0 | R | W | | 2 |
| \$AA_QEC_MEAS_TIME_1 | REAL | \$AA_QEC_MEAS_TIME_1[n,a] Tempo de medição para a faixa \$AA_QEC_ACCEL_1 a: Eixo da máquina n: 0 | R | W | | 2 |
| \$AA_QEC_MEAS_TIME_2 | REAL | \$AA_QEC_MEAS_TIME_2[n,a] Tempo de medição para a faixa \$AA_QEC_ACCEL_2 a: Eixo da máquina n: 0 | R | W | | 2 |

15.2 Lista das variáveis de sistema

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog. | NC | Sínc | O | S |
|------------------------|------|--|-------|----|------|---|---|
| \$AA_QEC_MEAS_TIME_3 | REAL | \$AA_QEC_MEAS_TIME_3[n,a] Tempo de medição para a faixa \$AA_QEC_ACCEL_3 a: Eixo da máquina n: 0 | R | W | | | 2 |
| \$AA_QEC_TIME_1 | REAL | \$AA_QEC_TIME_1[n,a] 1º tempo de filtro para elemento de previsão de avanço a: Eixo da máquina n: 0 | R | W | | | 2 |
| \$AA_QEC_TIME_2 | REAL | \$AA_QEC_TIME_2[n,a] 2º tempo de filtro para elemento de previsão de avanço a: Eixo da máquina n: 0 | R | W | | | 2 |
| \$AA_QEC_LEARNING_RATE | REAL | \$AA_QEC_LEARNING_RATE[n,a] Relação de 'aprendizado' para a rede a: Eixo da máquina n: 0 | R | W | | | 2 |
| \$AA_QEC_DIRECTIONAL | BOOL | \$AA_QEC_DIRECTIONAL[n,a] TRUE: Compensação direcional FALSE: Compensação não direcional a: Eixo da máquina n: 0 | R | W | | | 2 |

15.2.17 Compensação de interpolação

| | | | | | | | |
|----------------------|------|---|---|---|--|--|---|
| \$AN_CEC | REAL | \$AN_CEC[n,m] Valor de compensação n: Nr. da tabela de compensação 0 – (valor máximo via MD) m: Nr. do ponto de interpolação, 0 – (valor máximo via MD) | R | W | | | 2 |
| \$AN_CEC_INPUT_AXIS | AXIS | \$AN_CEC_INPUT_AXIS[n]: Nome do eixo cujo valor desejado irá atuar como elemento de entrada da tabela n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | | 2 |
| \$AN_CEC_OUTPUT_AXIS | AXIS | \$AN_CEC_OUTPUT_AXIS[n]: Nome do eixo a ser influenciado pelos valores de saída da tabela n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | | 2 |
| \$AN_CEC_STEP | REAL | \$AN_CEC_STEP[n] Distância entre os valores de compensação n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|------------------------------------|------|---|---------|------|---|---|
| \$AN_CEC_MIN | REAL | AN_CEC_MIN[n] Ponto inicial da tabela de compensação n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | 2 |
| \$AN_CEC_MAX | REAL | AN_CEC_MAX[n] Posição final da tabela de compensação n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | 2 |
| \$AN_CEC_DIREC TION | INT | \$AN_CEC_DIRECTION[n] Ativa a ação direcional da tabela de compensação n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | 2 |
| \$AN_CEC_MULT _BY_TABLE | INT | \$AN_CEC_MULT_BY_TABLE[n] Número da tabela para a qual o valor inicial deve ser multiplicado pelo valor inicial da tabela de compensação 0: Ambas direções de movimento do eixo base 1: Direção positiva de movimento do eixo base -1: Direção negativa de movimento do eixo base n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | 2 |
| \$AN_CEC_IS_MO DULO | BOOL | \$AN_CEC_IS_MODULO[n] TRUE: repetição cíclica da tabela de compensação FALSE: sem repetição cíclica da tabela de compensação n: Nr. da tabela de compensação 0 – (valor máximo via MD) | R | W | | 2 |

15.2.18 Zonas específicas de proteção NCK

| | | | | | | |
|---------------------------------|------|---|---|---|--|---|
| \$SN_PA_ACTIV_I MMED | BOOL | \$SN_PA_ACTIV_IMMED[n] Zona de proteção ativada imediatamente? TRUE: A zona de proteção é ativada imediatamente com a ligação do comando e referenciamento dos eixos FALSE: A zona de proteção não é ativada imediatamente n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SN_PA_T_W | CHAR | \$SN_PA_T_W[n] Orientação da zona de proteção – peça/ferramenta 0: Zona de proteção orientada à peça 3: Zona de proteção orientada à ferramenta n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SN_PA_ORI | INT | \$SN_PA_ORI[n] Orientação da zona de proteção 0: Polígono no plano entre os 1º e 2º eixos geométricos 1: Polígono no plano entre os 3º e 1º eixos geométricos 2: Polígono no plano entre os 2º e 3º eixos geométricos n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|--------------------------|------|---|---------|------|---|---|
| \$SN_PA_LIM_3DIM | INT | \$SN_PA_LIM_3DIM[n] Código para restringir a zona de proteção no sentido do eixo paralelo à direção da definição do polígono 0: = Sem limite 1: = Limite na direção positiva 2: = Limite na direção negativa 3: = Limite em ambas as direções n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SN_PA_PLUS_LIM | REAL | \$SN_PA_PLUS_LIM[n] Limite positivo para a zona de proteção no sentido do eixo perpendicular à definição do polígono n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SN_PA_MINUS_LIM | REAL | \$SN_PA_MINUS_LIM[n] Limite negativo para a zona de proteção no sentido do eixo perpendicular à definição do polígono n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SN_PA_CONT_NUM | INT | \$SN_PA_CONT_NUM[n] Número do elemento de contorno válido n: Número da zona de proteção 0 – (valor máximo via MD) | R | W | | 2 |
| \$SN_PA_CONT_TYP | INT | \$SN_PA_CONT_TYP[n,m] Tipo de elemento de contorno (G1, G2, G3) n: Número da zona de proteção 0 – (valor máximo via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SN_PA_CONT_ORD | REAL | \$SN_PA_CONT_ORD[n,m] Ponto final do elemento de contorno (ordenada) n: Número da zona de proteção 0 – (valor máximo via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SN_PA_CONT_ABS | REAL | \$SN_PA_CONT_ABS[n,m] Ponto final do elemento de contorno (abscissa) n: Número da zona de proteção 0 – (valor máximo via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SN_PA_CENT_ORD | REAL | \$SN_PA_CENT_ORD[n,m] Ponto central do elemento de contorno (ordenada) n: Número da zona de proteção 0 – (valor máximo via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |
| \$SN_PA_CENT_ABS | REAL | \$SN_PA_CENT_ABS[n,m] Ponto central do elemento de contorno (abscissa) n: Número da zona de proteção 0 – (valor máximo via MD) m: Número do elemento de contorno 0 – 10 | R | W | | 2 |

15.2.19 Dados de sistema

| | | | | | | | |
|--------------------------|------|---|----|--|---|--|-------------|
| \$AN_SETUP_TIME | REAL | IF \$AN_SETUP_TIME > 60000 GOTOF MARK01 Tempo desde a última ligação do comando com valores default (em minutos) | RS | | R | | 5 . 2 |
| \$AN_POWERON_TIME | REAL | IF \$AN_POWERON_TIME == 480 GOTOF MARK02 Tempo desde a última ligação standard do comando (em minutos) | RS | | R | | 5 . 2 |

15.2.20 Frames 2

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|-------|--|---------|------|---|---|
| \$P_UBFR | FRAME | \$P_UBFR 1. Frame base no canal ativado após G500, G54..G599. Corresponde à \$P_CHBFR[0]. | R | W | | 4 |
| \$P_CHBFRAME | FRAME | \$P_CHBFRAME[n] Frame base atual no canal, configurável de 0 a 8 via MD \$MC_MM_NUM_BASE_FRAMES. Estas dimensões são checadas no acesso à variável. | R | W | | 5 |
| \$P_NCBFRAME | FRAME | \$P_NCBFRAME[n] Número do frame base NCU. Os frames base de 0 a 8 NCU podem ser configurados através do MD \$MN_MM_NUM_GLOBAL_BASE_FRAMES. As dimensões são checadas no acesso à variável | R | W | | 5 |
| \$P_ACTBFRAME | FRAME | \$P_ACTBFRAME Frame geral atualmente conectado | R | | | 5 |
| \$P_BFRAME | FRAME | \$P_BFRAME 1º frame base atual no canal. Corresponde à \$P_CHBFRAME[0]. | R | W | | 4 |
| \$P_IFRAME | FRAME | \$P_IFRAME Frame ajustável atual | R | W | | 2 |
| \$P_PFRAME | FRAME | \$P_PFRAME Frame programável atual | R | W | | 2 |
| \$P_ACTFRAME | FRAME | \$P_ACTFRAME Frame total atual | R | | | 2 |
| \$P_UIFRNUM | INT | \$P_UIFRNUM Número do \$P_UIFR ativo | R | | | 2 |
| \$P_NCBFRMASK | INT | \$P_NCBFRMASK Uma máscara de dados binários é utilizada para definição dos frames base globais da NCU, que são incluídos no cálculo do frame base total. | R | W | | 5 |
| \$P_CHBFRMASK | INT | \$P_CHBFRMASK Uma máscara de dados binários é utilizada para definição dos frames base dos canais, que são incluídos no cálculo do frame base total. | R | W | | 5 |

15.2.21 Dados de ferramentas

| | | | | | | |
|------------|------|---|---|---|--|---|
| \$P_AD | REAL | \$P_AD[n] Corretores de ferramenta ativos n: Número do parâmetro 1 – 27 | R | W | | 2 |
| \$P_TOOL | INT | \$P_TOOL Incerto da ferramenta ativo D0 – D'max.'; 'max'= valor de \$MN_MM_MAX_CUTTING_EDGE_NO | R | | | 2 |
| \$P_TOOLNO | INT | \$P_TOOLNO Número da ferramenta ativa T0 – T32000; T pode possuir 8 dígitos caso os “corretores planos” estejam ativos | R | | | 2 |

15.2 Lista das variáveis de sistema

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|--------|--------|-------------|
| \$P_TOOLL | REAL | \$P_TOOLL[n] Comprimento total ativo da ferramenta n: Comprimento 1 – 3 | R | | | 2 |
| \$P_TCANG | REAL | \$P_TCANG[n] Ângulo ativo do eixo do porta ferramentas n: Ângulo 1 – 2 | R | | | 5 |
| \$P_TOOLR | REAL | \$P_TOOLR Raio da ferramenta ativo (total) | R | | | 2 |
| \$P_TOOLND | INT | \$P_TOOLND[t] Quantidade de incertos da ferramenta t: Número T 1 – 32000 | R | | | 4 |
| \$P_TOOLEXIST | BOOL | \$P_TOOLEXIST[t] Ferramenta com Nr. T. t existe t: Número T 1 – 32000 | R | | | 4 |
| \$P_D | INT | \$P_D Número do corretor atual da ferramenta no modo de instruções ISO_2 | R | | | 5 · 2 |
| \$P_H | INT | \$P_H Número H atual no modo de instruções ISSO_2 | R | | | 5 · 2 |
| \$A_TOOLMN | INT | \$A_TOOLMN[t] Número do magazine da ferramenta t t: Número T 1 – 32000 | | | R | 4 |
| \$A_TOOLMLN | INT | \$A_TOOLMLN[t] Número do magazine da ferramenta t t: Número T 1 – 32000 | | | R | 4 |
| \$A_MONIFACT | REAL | \$A_MONIFACT Fator para monitoração do comprimento da ferramenta | R | W S | R W | 4 |
| \$AC_MONMIN | REAL | \$AC_MONMIN relação entre o valor atual de monitoração e o valor inicial. Nível para pesquisa de ferramentas "Carregar somente ferramentas com valor atual maior que um determinado nível" | R | W S | R W | 5 · 2 |
| \$P_VDITCP | INT | \$P_VDITCP[n] Parâmetros disponíveis para o gerenciamento de ferramentas na interface VDI n: Índice de 1-3 | R | W | | 2 |
| \$A_DNO | INT | \$A_DNO[i] Lê um número D definido pelo PLC através da interface VDI i: Índice 1 – 9 para posição da tabela de número D | | | R | 4 |
| \$P_ATPG | REAL | \$P_ATPG[n] Dados atuais da ferramenta relacionados à retífica n: Número do parâmetro 1 – 9 | R | W | | 2 |

15.2.22 Valores programados

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$P_AXN1 | AXIS | \$P_AXN1 Endereço atual do eixo geométrico – abcissa | R | | | 3 |
| \$P_AXN2 | AXIS | \$P_AXN2 Endereço atual do eixo geométrico – ordenada | R | | | 3 |
| \$P_AXN3 | AXIS | \$P_AXN3 Endereço atual do eixo geométrico – aplicado | R | | | 3 |
| \$P_ACTGEOAX | AXIS | \$P_ACTGEOAX[1] Eixo geométrico atual, dependendo do plano Retorna o eixo geométrico programado com GEOAX(1,X,2,Y,3,Z) Índice do array 1–3 para os eixos geométricos de 1 à 3 n: número da entrada 1 – ... | R | | | 4 |

15.2.23 Grupos G

| | | | | | | |
|-----------|-----|---|---|---|--|---|
| \$P_GG | INT | \$P_GG[n] Função G atual de um grupo (mesmo índice da interface PLC) n: Número do grupo G | R | | | 2 |
| \$P_EXTGG | INT | \$P_EXTGG[n] Pode ser utilizado somente no modo Siemens: Função G atual de um grupo G com linguagem NC externa (índice como na interface PLC) n: Número do grupo G | R | | | 5 |
| \$A_GG | INT | \$A_GG[n] Lê a função G atual do grupo ativo (índice como na interface PLC) a partir de AS (índice como na interface PLC). n: Número do grupo G | | R | | 5 |

| | | | | | | |
|-------------|------|--|---|--|--|---|
| \$P_SEARCH | BOOL | \$P_SEARCH Pesquisa de blocos ativa TRUE (1) | R | | | 2 |
| \$P_SEARCH1 | BOOL | \$P_SEARCH1 Pesquisa de blocos com cálculos ativa TRUE (1) | R | | | 2 |
| \$P_SEARCH2 | BOOL | \$P_SEARCH2 Pesquisa de blocos sem cálculo ativa TRUE (1) | R | | | 2 |
| \$P_SEARCHL | INT | R1 = \$P_SEARCHL Retorna o último tipo de pesquisa selecionada: (código análogo à Piservice_N_FINDBL) 0 : Sem pesquisa de blocos 1 : Pesquisa de blocos sem cálculo 2 : Pesquisa de blocos com cálculo de contorno 3 : Reservado 4 : pesquisa de bloco com cálculo e posição final de bloco | R | | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------------|--|---------|------|---|-------------|
| \$P_SUBPAR | BOOL | \$P_SUBPAR[n] verifica se o subprograma com transferência de parâmetros para n parâmetros foi programado (TRUE) ou caso o sistema tenha utilizado os valores default (FALSE). n: Número do parâmetro 1 a n de acordo com a definição com a instrução PROC | R | | | 5 |
| \$P_CTABDEF | BOOL | \$P_CTABDEF Definição de tabelas de curva caso ativo TRUE (1) | R | | | 4 |
| \$P_MC | INT | \$P_MC Estado da chamada modal de subprogramas FALSE (0) -> Chamada de subprograma não modal TRUE (1) -> Chamada de subprograma modal | R | | | 2 |
| \$P_REPINF | INT | \$P_REPINF Informação do estado durante o reposicionamento com o comando REPOS (0) -> Reposicionar com REPOS não possível devido à: - Chamada não executada em uma ASUP - Chamada executada em uma ASUP, que foi iniciada em estado de reset - Chamada executada em uma ASUP, que foi iniciada em modo JOG (1)-> Reposicionamento com REPOS possível | R | | | 4 |
| \$P_SIM | BOOL | \$P_SIM Simulação em andamento caso TRUE (1) | R | | | 2 |
| \$P_DRYRUN | BOOL | \$P_DRYRUN Dry run (avanço de teste) ligado caso TRUE, caso contrário FALSE | R | | | 2 |
| \$P_OFFN | REAL | \$P_OFFN Programado deslocamento normal de contorno | R | | | 5 . 1 |
| \$PI | REAL | \$PI Constante PI = 3.1415927 | R | | | 2 |
| \$P_PROGPATH | STRIN G | PCALL (\$P_PROGPATH << _N_MYSUB_SPF) Chamada de subprograma a partir do diretório atual Exemplo: O diretório atual é /_N_WCS_DIR/_N_SHAFT_DIR/. A chamada acima inicia o subprograma /_N_WCS_DIR/_N_SHAFT_DIR/_N_MYSUB_SPF. | R | | | 3 |
| \$P_PROG | STRIN G | mmcNum = 474 NAME = \$P_PROG[0] Retorna o nome do programa no nível de programa 0, ou seja, o nome do programa principal, na string NAME. O valor entre colchetes define o nível do programa a partir do qual o nome do programa será lido. | R | | | 5 . 1 |
| \$P_STACK | INT | \$P_STACK progLevel = \$P_STACK , Retorna o nível do programa atual na variável inteira. | R | | | 5 . 1 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|------------------|--------|---|---------|------|---|--------|
| \$P_PATH | STRING | <p>\$P_PATH[0] Retorna o diretório do programa atual principal, p.e. "_N_WCS_DIR/_N_SHAFT_WPD"</p> <p>A variável é utilizada para gravar uma subrotina gerada com WRITE, para, por exemplo, no mesmo diretório onde o programa chamado se encontre:</p> <pre>DEF INT ERROR WRITE (ERROR, \$P_PATH[\$P_STACK – 1] << _N_LIST_MPF, "X10 Y20")</pre> <p>Caso o programa atual seja chamado a partir do diretório de programas principal, um novo arquivo /_N_MPF_DIR/_N_LIST_MPF será criado.</p> <p>Define o nível de programa a partir do qual o diretório de programas deve ser lido.</p> | R | | | 5 1 |
| \$P_ACTID | BOOL | <p>\$P_ACTID[n]</p> <p>Ação síncrona modal com ID n ativa caso TRUE</p> <p>n: 1–16</p> | R | | | 2 |

15.2.24 Estados do canal

| | | | | | | | |
|----------------------|-----|---|----|--|---|--|--------|
| \$AC_STAT | INT | <p>\$AC_STAT</p> <p>–1: Inválido</p> <p>0: Canal em reset</p> <p>1: Canal interrompido</p> <p>2: Canal ativo</p> | | | R | | 4 |
| \$AC_PROG | INT | <p>\$AC_PROG</p> <p>–1: Inválido</p> <p>0: Programa em modo reset</p> <p>1: Programa parado</p> <p>2: Programa ativo</p> <p>3: Programa esperando</p> <p>4: Programa interrompido</p> | | | R | | 4 |
| \$AC_SYNA_MEM | INT | <p>\$AC_SYNA_MEM</p> <p>Memória livre para ações síncronas de movimentação, indica quantos elementos da memória ocupada por \$MC_MM_NUM_SYNC_ELEMENTS ainda se encontram livres, podendo ser lido tanto de um programa NC quanto de ações síncronas</p> | | | R | | 4 |
| \$AC_IPO_BUF | INT | <p>\$AC_IPO_BUF</p> <p>Nível do buffer de interpolação, pode ser lido a partir do programa NC ou de ações síncronas. A leitura não gera parada no controle de previsão de avanço.</p> | | | R | | 4 |
| \$AC_IW_STAT | INT | <p>\$AC_IW_STAT</p> <p>Informação de posição em juntas articuladas (específico para transformações) para movimentos PTP</p> | RS | | R | | 5 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|--------|---|-------------|
| \$AC_IW_TU | INT | \$AC_IW_TU Informação de posição para os eixos (MCS) para movimentos PTP | RS | | R | 5 · 2 |
| \$A_PROBE | INT | \$A_PROBE[1]: Estado do 1º apalpador \$A_PROBE[2]: Estado do 2º apalpador 0 => não acionado 1 => acionado n: número do apalpador | RS | | R | 4 |
| \$AC_MEA | INT | \$AC_MEA[n] O apalpador foi acionado caso TRUE (1) n: Número do apalpador 1 – MAXNUM_PROBE | | | R | 2 |
| \$AC_TRAFO | INT | \$AC_TRAFO Código para o nível de transformação ativo de acordo com o dado de máquina \$MC_TRAFO_TYPE n | RS | | R | 3 |
| \$AC_LIFTFAST | INT | \$AC_LIFTFAST Informações sobre a execução de LIFTFAST: 0: Estado inicial 1: Em execução A variável é ligada (1) pelo NC no início do processo de levantamento rápido. O programa pode analisar quando a variável é novamente zerada (\$AC_LIFTFAST=0) para continuar suas ações subsequentes. | RS | W S | R | W 4 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$AC_ASUP | INT | <p>\$AC_ASUP</p> <p>Um código mostra a forma de ativação da ASUP:</p> <p>BIT0: Ativada por: Interrupção do usuário "ASUP com Blsinc"</p> <p>Ativação por: Sinal VDI na interface analógica/digital</p> <p>Continua com: Reorg ou Ret selecionada pelo usuário</p> <p>BIT1: Ativada por: Interrupção do usuário "ASUP"</p> <p>Para que seja possível continuar o programa com Repos, a posição após a parada é memorizada.</p> <p>Ativação por: Sinal VDI, interface analógica/digital</p> <p>Continua com: seleção do usuário</p> <p>BIT2: Ativada por: Interrupção de usuário "ASUP a partir do estado canal pronto"</p> <p>Ativação por: Sinal VDI, interface analógica/digital</p> <p>Continua com: seleção do usuário</p> <p>BIT3: Ativada por: Interrupção do usuário "ASUP em modo manual e canal em estado pronto"</p> <p>Ativação por: Sinal VDI, interface analógica/digital</p> <p>Continua com: seleção do usuário</p> <p>BIT4: Ativada por: Interrupção do usuário "ASUP"</p> <p>Para que seja possível continuar o programa com Repos, a posição após a parada é memorizada.</p> <p>Ativação por: Sinal VDI, interface analógica/digital</p> <p>Continua com: seleção do usuário</p> <p>BIT5: Ativada por: Cancelamento de repetição de subprograma</p> <p>Ativação por: Sinal VDI</p> <p>Continua com: uso da ASUP de sistema REPOS</p> <p>BIT6: Ativada por: Ativação da decodificação bloco a bloco</p> <p>Ativação por: Sinal VDI (+OPI)</p> <p>Continua com: Uso da ASUP de sistema REPOS</p> <p>BIT7: Ativada por: Ativação de cancelamento da distância a ser percorrida</p> <p>Ativação por: Sinal VDI</p> <p>Continua com: Uso da ASUP de sistema RET</p> <p>BIT8: Ativada por: Ativação do sincronismo de eixos</p> <p>Ativação por: Sinal VDI</p> <p>Continua com: Uso da ASUP de sistema REPOS</p> <p>BIT9: Ativada por: Troca de modo</p> <p>Ativação por: Sinal VDI</p> <p>Continua com: uso da ASUP de sistema REPOS ou RET (vide MD)</p> <p>BIT10: Ativada por: Continuação do programa com teach in ou após a desativação de teach in</p> <p>Ativação por: Sinal VDI</p> <p>Continua com: uso da ASUP de sistema RET</p> <p>BIT11: Ativada por: Seleção OVERSTORE</p> <p>Ativação por: Seleção PI</p> <p>Continua com: uso da ASUP de sistema REPOS</p> <p>BIT12: Ativada por: Reação a alarme de bloco de compensação com REPOS (COMPBLOCKWITHREORG)</p> <p>Ativação por: Interna</p> <p>Continua com: uso da ASUP de sistema REPOS</p> <p>BIT13: Ativada por: Movimento de recuo com G33 e Stop</p> | RS | R | | 4 |

| | | | | | | | | | |
|--|--|---|--|--|--|--|--|--|--|
| | | Ativação por: Interno Continua com: uso da ASUP de sistema RET BIT14: Ativada por: Ativação de avanço de teste (dry run) Ativação por: VDI Continua com: uso da ASUP de sistema REPOS BIT15: Ativada por: Desativação de avanço de teste (dry run) Ativação por: VDI Continua com: uso da ASUP de sistema REPOS BIT16: Ativada por: Ativação da supressão de bloco Ativação por: VDI Continua com: uso da ASUP de sistema REPOS BIT17: Ativada por: Desativação da supressão de bloco Ativação por: VDI Continua com: uso da ASUP de sistema REPOS BIT18: Ativada por: através da ativação de dados de máquina Ativação por: PI Continua com: uso da ASUP de sistema REPOS | | | | | | | |
|--|--|---|--|--|--|--|--|--|--|

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
|---------------|------|--|---------|------|---|---|

| | | | | | | | | | |
|------------|------------|---|----|--------|---|---|--|--|---|
| \$P_ISTEST | BOOL | \$P_ISTEST Checa o modo teste no programa de usinagem TRUE = Teste de programa ativo FALSE = Teste de programa não ativo | R | | | | | | 4 |
| \$P_MMCA | STRIN G | \$P_MMCA Reconhecimento de tarefa MMC | R | W | | | | | 2 |
| \$A_PROTO | BOOL | \$A_PROTO ativa/desativa função protocolo | RS | W S | R | W | | | 4 |

15.2.25 Ações síncronas

| | | | | | | | | | |
|-------------|------|---|----|--------|---|---|---|--|---|
| \$AC_MARKER | INT | \$AC_MARKER[n] Variável de memória para movimentos com ações síncronas Dimensão definida no MD \$MC_MM_NUM_AC_MARKER. | RS | W S | R | W | + | | 2 |
| \$AC_PARAM | REAL | \$AC_PARAM[n] Variável aritmética para movimentos com ações síncronas. Dimensão definida no MD \$MC_MM_NUM_AC_PARAM. | RS | W S | R | W | + | | 3 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S | | | |
|---------------|------|---|---------|------|---|---|---|---|--|
| \$AC_FIFO1 | REAL | <p>\$AC_FIFO1[n]</p> <p>FIFO para movimentos com ações síncronas e ciclos de medição</p> <p>n: Número do parâmetro 0 – max. elemento da FIFO</p> <p>Significado especial:</p> <p>n=0: Na escrita com índice 0, um novo valor é armazenado na FIFO, Na leitura com índice 0, o último elemento é lido e apagado da FIFO</p> <p>n=1: leitura do elemento mais antigo</p> <p>n=2: leitura do elemento mais novo</p> <p>n=3: Total de elementos da FIFO, caso o bit 0 do MD \$MC_MM_MODE_FIFO esteja ligado</p> <p>n=4: Leitura da quantidade atual de elementos da FIFO</p> <p>n=5–m: Acesso á elementos individuais da FIFO</p> <p>5 é o elemento mais antigo</p> <p>6 é o segundo elemento mais antigo, etc.</p> | RS | W | R | W | + | 4 | |
| \$AC_FIFO2 | REAL | <p>\$AC_FIFO2[n]</p> <p>FIFO para movimentos com ações síncronas e ciclos de medição</p> <p>n: Número do parâmetro 0 – max. elemento da FIFO</p> <p>Significado especial:</p> <p>n=0: Na escrita com índice 0, um novo valor é armazenado na FIFO, Na leitura com índice 0, o último elemento é lido e apagado da FIFO</p> <p>n=1: leitura do elemento mais antigo</p> <p>n=2: leitura do elemento mais novo</p> <p>n=3: Total de elementos da FIFO, caso o bit 0 do MD \$MC_MM_MODE_FIFO esteja ligado</p> <p>n=4: Leitura da quantidade atual de elementos da FIFO</p> <p>n=5–m: Acesso á elementos individuais da FIFO</p> <p>5 é o elemento mais antigo</p> <p>6 é o segundo elemento mais antigo, etc.</p> | RS | W | R | W | + | 4 | |
| ... | ... | ... | | | | | | | |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S | | |
|---------------|------|--|---------|------|---|---|---|---|
| ... | ... | ... | | | | | | |
| \$AC_FIFO10 | REAL | <p>\$AC_FIFO10[n]</p> <p>FIFO para movimentos com ações síncronas e ciclos de medição</p> <p>n: Número do parâmetro 0 – max. elemento da FIFO</p> <p>Significado especial:</p> <p>n=0: Na escrita com índice 0, um novo valor é armazenado na FIFO, Na leitura com índice 0, o último elemento é lido e apagado da FIFO</p> <p>n=1: leitura do elemento mais antigo</p> <p>n=2: leitura do elemento mais novo</p> <p>n=3: Total de elementos da FIFO, caso o bit 0 do MD \$MC_MM_MODE_FIFO esteja ligado</p> <p>n=4: Leitura da quantidade atual de elementos da FIFO</p> <p>n=5–m: Acesso á elementos individuais da FIFO</p> <p>5 é o elemento mais antigo</p> <p>6 é o segundo elemento mais antigo, etc.</p> | RS | W | R | W | + | 4 |

15.2.26

I/Os

| | | | | | | | | |
|-----------------|------|---|----|---|---|---|--|---|
| \$A_IN | BOOL | \$A_IN[n] Entrada NC digital n: Número da entrada 1-... A quantidade máxima é determinada através do MD \$MN_FASTIO_DIG_NUM_INPUTS | RS | | R | | | 2 |
| \$A_OUT | BOOL | \$A_OUT[n] Saída NC digital n: Número da saída 1-... A quantidade máxima é determinada através do MD \$MN_FASTIO_DIG_NUM_OUTPUTS | RS | W | R | W | | 2 |
| \$A_INA | REAL | \$A_INA[n] Entrada analógica NC n: Número da saída 1-... A quantidade máxima é determinada através do MD \$MN_FASTIO_ANA_NUM_INPUTS | RS | | R | | | 2 |
| \$A_OUTA | REAL | \$A_OUTA[n] Saída analógica NC Durante a escrita, o valor não é ativado até que o ciclo IPO seja novamente executado e seu valor confirmado. n: Número da saída 1-... A quantidade máxima é determinada através do MD \$MN_FASTIO_ANA_NUM_OUTPUTS | RS | W | R | W | | 2 |
| \$A_INCO | BOOL | \$A_INCO[n] Entrada comparadora n: Número da saída 1 – ... O número máximo de entradas é configurado através de MD | RS | | R | | | 2 |

15.2.27 Leitura e escrita de variáveis PLC

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | | Sínc | O | S |
|---------------|------|--|---------|---|------|---|---|
| \$A_DBB | INT | \$A_DBB[n] Leitura/escrita de um byte de dados (8bits) do/para o PLC n: posição da área de I/O 0 – ... | RS | W | R | W | 4 |
| \$A_DBW | INT | \$A_DBW[n] Leitura/escrita de uma palavra de dados (16 bits) do/para o PLC n: posição da área de I/O 0 – ... | RS | W | R | W | 4 |
| \$A_DBD | INT | \$A_DBD[n] Leitura/escrita de uma palavra dupla de dados (32 bits) do/para o PLC n: posição da área de I/O 0 – ... | RS | W | R | W | 4 |
| \$A_DBR | REAL | \$A_DBR[n] Leitura/escrita de um dado tipo real (32 bits) do/para o PLC n: posição da área de I/O 0 – ... | RS | W | R | W | 4 |

15.2.28 Conexão NCU

| | | | | | | | |
|----------------------------|------|---|----|---|---|---|---|
| \$A_DLB | INT | \$A_DLB[n] Leitura/escrita de um byte de dados (8bits) de/para a conexão NCU n: Posição do dado na área de memória 0-... sincronizada com o programa | RS | W | R | W | 5 |
| \$A_DLW | INT | \$A_DLW[n] Leitura/escrita de palavra de dados (16 bits) de/para a conexão NCU n: Posição do dado na área de memória 0-... sincronizada com o programa | RS | W | R | W | 5 |
| \$A_DLD | INT | \$A_DLD[n] Leitura/escrita de palavra dupla (32 bits) de/para a conexão NCU n: Posição do dado na área de memória 0-... sincronizada com o programa | RS | W | R | W | 5 |
| \$A_DLR | REAL | \$A_DLR[n] Leitura/escrita de um dado tipo real (32 bits) de/para a conexão NCU n: Posição do dado na área de memória 0-... sincronizada com o programa | RS | W | R | W | 5 |
| \$A_LINK_TRANS_RATE | INT | \$A_LINK_TRANS_RATE Quantidade de bytes que podem ainda ser transferidos através da conexão NCU no ciclo IPO atual. | | | R | | 5 |

15.2.29 I/O PLC direto

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|--------|
| \$A_PBB_IN | INT | \$A_PBB_IN[n] Leitura de um byte de dados (8bits) diretamente do I/O do PLC n: número do byte na área de entradas do PLC 0-... | RS | | R | 5 |
| \$A_PBW_IN | INT | \$A_PBW_IN[n] Leitura de uma palavra de dados (16bits) diretamente do I/O do PLC n: número do byte na área de entradas do PLC 0-... | RS | | R | 5 |
| \$A_PBD_IN | INT | \$A_PBD_IN[n] Leitura de palavra dupla de dados (32bits) diretamente do I/O do PLC n: número do byte na área de entradas do PLC 0-... | RS | | R | 5 |
| \$A_PBR_IN | REAL | \$A_PBR_IN[n] Leitura de um dado real (32bits) diretamente do I/O do PLC n: número do byte na área de entradas do PLC 0-... | RS | | R | 5 2 |
| \$A_PBB_OUT | INT | \$A_PBB_OUT[n] Escrita de um byte de dados (8bits) diretamente no I/O do PLC n: número do byte na área de saídas do PLC 0-... em sincronismo com o programa | RS | W | R | 5 |
| \$A_PBW_OUT | INT | \$A_PBW_OUT[n] Escrita de uma palavra de dados (16bits) diretamente no I/O do PLC n: número do byte na área de saídas do PLC 0-... em sincronismo com o programa | RS | W | R | 5 |
| \$A_PBD_OUT | INT | \$A_PBD_OUT[n] Escrita de palavra dupla de dados (32bits) diretamente no I/O do PLC n: número do byte na área de saídas do PLC 0-... em sincronismo com o programa | RS | W | R | 5 |
| \$A_PBR_OUT | REAL | \$A_PBR_OUT[n] Escrita um valor real (32bits) diretamente no I/O do PLC n: número do byte na área de saídas do PLC 0-... em sincronismo com o programa | RS | W | R | 5 |

15.2.30 Gerenciamento de ferramenta

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|----------------|------|--|---------|------|---|---|
| \$AC_TC_FCT | INT | \$AC_TC_FCT Número de comando. Especifica qual a ação desejada. | RS | | R | 5 |
| \$AC_TC_STATUS | INT | \$AC_TC_STATUS Estado do comando – leitura através de \$AC_TC_FCT. | RS | | R | 5 |
| \$AC_TC_THNO | INT | \$AC_TC_THNO Número do porta ferramentas (específico para o número do fuso) onde uma nova ferramenta deve ser trocada. | RS | | R | 5 |
| \$AC_TC_TNO | INT | \$AC_TC_TNO Número T interno da NCK da nova ferramenta (a ser trocada). 0: não há nova ferramenta. | RS | | R | 5 |
| \$AC_TC_MFN | INT | \$AC_TC_MFN Número do magazine de origem da nova ferramenta. 0: não há nova ferramenta. | RS | | R | 5 |
| \$AC_TC_LFN | INT | \$AC_TC_LFN Número do alojamento fonte da nova ferramenta. 0: não há nova ferramenta. | RS | | R | 5 |
| \$AC_TC_MTN | INT | \$AC_TC_MTN Magazine de destino da nova ferramenta. 0: não há nova ferramenta. | RS | | R | 5 |
| \$AC_TC_LTN | INT | \$AC_TC_LTN Alojamento destino da nova ferramenta. 0: não há ferramenta nova | RS | | R | 5 |
| \$AC_TC_MFO | INT | \$AC_TC_MFO Magazine fonte da ferramenta antiga (a ser trocada). 0: não há ferramenta antiga. | RS | | R | 5 |
| \$AC_TC_LFO | INT | \$AC_TC_LFO Magazine fonte da ferramenta antiga (a ser trocada) 0: Não há ferramenta antiga. | RS | | R | 5 |
| \$AC_TC_MTO | INT | \$AC_TC_MTO Magazine fonte da antiga ferramenta (a ser trocada). 0: Não há ferramenta antiga. | RS | | R | 5 |
| \$AC_TC_LTO | INT | \$AC_TC_LTO Magazine destino da ferramenta antiga (a ser trocada) 0: Não há ferramenta antiga. | RS | | R | 5 |

15.2.31 Temporizadores

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-------------------|------|---|---------|--------|-------------|---|
| \$A_YEAR | INT | \$A_YEAR Relógio do sistema, ano | RS | R | | 3 |
| \$A_MONTH | INT | \$A_MONTH Relógio do sistema, mês | RS | R | | 3 |
| \$A_DAY | INT | \$A_DAY Relógio do sistema, dia | RS | R | | 3 |
| \$A_HOUR | INT | \$A_HOUR Relógio do sistema, hora | RS | R | | 3 |
| \$A_MINUTE | INT | \$A_MINUTE Relógio do sistema, minuto | RS | R | | 3 |
| \$A_SECOND | INT | \$A_SECOND Relógio do sistema, segundo | RS | R | | 3 |
| \$A_MSECOND | INT | \$A_MSECOND Relógio do sistema, milissegundos | RS | R | | 3 |
| \$AC_TIME | REAL | \$AC_TIME Tempo desde o início do bloco em segundos Esta variável pode ser somente acessada a partir de ações síncronas | RS | R | | 2 |
| \$AC_TIMEC | REAL | \$AC_TIMEC Tempo desde o início do bloco em ciclos de IPO Esta variável pode ser somente acessada a partir de ações síncronas | RS | R | | 3 |
| \$AC_TIMER | REAL | \$AC_TIMER[n] Temporizador – unidade = segundos O tempo é contado internamente e multiplicado pelo valor do ciclo de interpolação; Para iniciar a contagem de uma variável, carregar um valor em \$AC_TIMER[n]=<valor inicial> Para interromper a contagem, carregar um valor negativo: \$AC_TIMER[n]=-1 O valor atual pode ser lido quando o temporizador estiver rodando ou quando estiver parado. Quando parado através da carga de um valor -1, o valor mais atualizado de tempo pode ser lido. A dimensão é definida no MD \$MC_MM_NUM_AC_TIMER. | RS | W S | R W + | 4 |
| \$AC_PRTIME_M | REAL | \$AC_PRTIME_M "Tempo principal de execução do programa" Carrega (inicializa) o valor acumulado do tempo de execução do programa (tempo principal). | | W | | 4 |
| \$AC_PRTIME_A | REAL | \$AC_PRTIME_A "Tempo auxiliar de execução do programa" Carrega (inicializa) o valor acumulado do tempo de execução do programa (tempo auxiliar). | | W | | 4 |
| \$AC_PRTIME_M_INC | REAL | \$AC_PRTIME_M_INC "Tempo de execução principal do programa em incrementos" Incremento acumulado do tempo de execução do programa (tempo principal) | | W | | 4 |
| \$AC_PRTIME_A_INC | REAL | \$AC_PRTIME_A_INC "Tempo auxiliar de execução do programa em incrementos" Incremento acumulado do tempo de execução do programa (tempo auxiliar) | | W | | 4 |

15.2.32 Movimento da peça

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$AC_PATHN | REAL | \$AC_PATHN Parâmetro normalizado de trajetória Valor entre 0=início do bloco e 1=fim do bloco Esta variável pode ser somente acessada a partir de ações síncronas | RS | R | | 2 |
| \$AC_DTBW | REAL | \$AC_DTBW Distância geométrica a partir do início do bloco no sistema de coordenadas da peça. A posição programada é decisiva para o cálculo da distância; caso o eixo seja um eixo acoplado, a posição não é considerada. Esta variável pode ser somente acessada a partir de ações síncronas | RS | R | | 2 |
| \$AC_DTBB | REAL | \$AC_DTBB Distância geométrica a partir do início do bloco no sistema de coordenadas básico. A posição programada é decisiva para o cálculo da distância; caso o eixo seja um eixo acoplado, a posição não é considerada. Esta variável pode ser somente acessada a partir de ações síncronas | RS | R | | 2 |
| \$AC_DTEW | REAL | \$AC_DTEW Distância geométrica a partir do início do bloco no sistema de coordenadas da peça. A posição programada é decisiva para o cálculo da distância; caso o eixo seja um eixo acoplado, a posição não é considerada. Pode ser acessada somente a partir de ações síncronas | RS | R | | 2 |
| \$AC_DTEB | REAL | \$AC_DTEB Distância geométrica a partir do fim do bloco no sistema de coordenadas básico. A posição programada é decisiva para o cálculo da distância; caso o eixo seja um eixo acoplado, a posição não é considerada. Pode ser acessada somente a partir de ações síncronas | RS | R | | 2 |
| \$AC_PLTBB | REAL | \$AC_PLTBB Distância da trajetória a partir do início do bloco no sistema de coordenadas básico. Pode ser acessada somente a partir de ações síncronas | RS | R | | 3 |
| \$AC_PLTEB | REAL | \$AC_PLTEB Distância da trajetória a partir do fim do bloco no sistema de coordenadas básico. Pode ser acessada somente a partir de ações síncronas | RS | R | | 3 |
| \$AC_DELT | REAL | \$AC_DELT Distância residual (que não foi percorrida) após o comando de cancelamento de percurso restante, através de ações síncronas, no sistema de coordenadas da peça. | | R | | 3 |
| \$P_APDV | BOOL | \$P_APDV Retorna TRUE caso os valores de posição possam ser lidos com \$P_APR[X] ou \$P_AEP[X] (ponto inicial do contorno ou ponto do contorno para aproximação ou recuo suave) sejam válidos. | R | | | 4 |

15.2.33 Velocidades

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$P_F | REAL | \$P_F Último valor de avanço de trajetória F programado | R | | | 2 |
| \$AC_OVR | REAL | \$AC_OVR: Override de trajetória para ações síncronas A componente de multiplicação override atua adicionalmente ao override de usuário, o programado e o de transformação. Entretanto, o fator total é restrito à 200%. Deve ser re-escrito a cada ciclo de interpolação, caso contrário, será considerado como sendo 100%. \$AA_OVR[S1] altera o override do fuso. O override definido através dos dados de máquina \$MN_OVR_FACTOR_LIMIT_BIN, \$MN_OVR_FACTOR_FEEDRATE[30], \$MN_OVR_FACTOR_AX_SPEED[30], não é excedido. Esta variável pode ser somente acessada a partir de ações síncronas | | R | W | 2 |
| \$AC_VC | REAL | \$AC_VC A valor do avanço aditivo para correção da trajetória para ações síncronas. Não é válida com G0, G33, G331, G332 e G63. Deve ser re-escrita a cada ciclo de interpolação, caso contrário, será considerado o valor 0. Com um override de 0, o valor de compensação não possui efeito, caso contrário o override não teria impacto no valor de compensação. O valor de compensação não pode causar um override total negativo. O valor superior é limitado de forma que as velocidades máximas dos eixos e acelerações não sejam excedidas. O cálculo com componentes de avanço diferentes não é afetada por \$AC_VC. Os valores de override definidos pelos dados de máquina \$MN_OVR_FACTOR_LIMIT_BIN, \$MN_OVR_FACTOR_FEEDRATE[30], \$MN_OVR_FACTOR_AX_SPEED[30] e \$MN_OVR_FACTOR_SPIND_SPEED não podem ser excedidos. O override de avanço aditivo é limitado de forma que o avanço resultante não exceda o valor de override máximo do avanço programado. Pode ser acessada somente a partir de ações síncronas | | R | W | 2 |
| \$AC_VACTB | REAL | \$AC_VACTB Velocidade de trajetória no sistema de coordenadas base. Pode ser acessada somente a partir de ações síncronas | RS | R | | 2 |
| \$AC_VACTW | REAL | \$AC_VACTW Velocidade de trajetória no sistema de coordenadas da peça. Pode ser acessada somente a partir de ações síncronas | RS | R | | 2 |

15.2.34 Fusos

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|------------------|------|---|---------|------|---|--------|
| \$P_GWPS | BOOL | \$P_GWPS[n] Velocidade periférica constante do rebolo caso TRUE n: número do fuso, 0 – número máximo de fusos | R | | | 2 |
| \$P_NUM_SPINDLES | INT | \$P_NUM_SPINDLES[n] Quantidade de fusos no canal | R | | | 5 3 |
| \$P_MSNUM | INT | \$P_MSNUM Valor de retorno: 0: não existem fusos 1..n: número do fuso mestre | R | | | 5 2 |
| \$AC_MSNUM | INT | \$AC_MSNUM Valor de retorno: 0: não existem fusos 1..n: número do fuso mestre | RS | R | | 3 |
| \$P_S | REAL | \$P_S[n] Último valor de rotação programado para o fuso n: número do fuso, 0 – número máximo de fusos | R | | | 2 |
| \$AA_S | REAL | \$AA_S[n] Velocidade atual do fuso. O sinal corresponde ao sentido da rotação. n: número do fuso, 0 – número máximo de fusos | RS | R | | 4 |
| \$P_SDIR | INT | \$P_SDIR[n] Último sentido de rotação programado. 3: Horário, 4: Anti horário, 5: parado n: número do fuso, 0 – número máximo de fusos | R | | | 3 |
| \$AC_SDIR | INT | \$AC_SDIR[n] Sentido de rotação atual 3: Horário, 4: Ante horário, 5: parado n: número do fuso, 0 – número máximo de fusos | RS | R | | 3 |
| \$P_SEARCH_S | REAL | \$P_SEARCH_S[n] Último valor de rotação acumulado p/pesquisa de bloco (SSL) 0: Fuso parado, 0 – último valor de rotação programado para o fuso | R | | | 5 3 |
| \$P_SEARCH_SDIR | INT | \$P_SEARCH_SDIR[n] Última direção de fuso programada para pesquisa de blocos 3: M3 valor de rotação no modo controle de velocidade 4: M4 valor de rotação no modo controle de velocidade 5: M5 valor de rotação no modo controle de velocidade -5: Preset do fuso não programado no instante de start SSL -19: M19 modo de posicionamento 70: M70 modo eixo n: número do fuso, 0 – número máximo de fusos | R | | | 5 3 |
| \$P_SEARCH_SGEAR | INT | \$P_SEARCH_SDIR[n] Último valor de marcha para o fuso programado através de função M para SSL 40: M40 troca automática de marcha 41: M41 marcha pré definida no programa NC 42: M42 marcha pré definida no programa NC 43: M43 marcha pré definida no programa NC 44: M44 marcha pré definida no programa NC 45: M45 marcha pré definida no programa NC n: número do fuso, 0 – número máximo de fusos | R | | | 5 3 |

| | | | | | | | | |
|---------------------------|------|--|----|---|---|--|--|-------------|
| \$P_SEARCH_POS | REAL | \$P_SEARCH_SPOS[n] Último valor de posição do fuso programada ou – trajetória para SSL Faixa de valores: de –100000000 a 100000000. –100000000 a –0,001: Trajetória possível na faixa negativa 100000000 a 0,000: Trajetória possível na faixa positiva Trajetória e definições de posições podem ser positivas ou negativas e são definidas com até 3 casas decimais. n: Dados de posicionamento devem estar contidos dentro da faixa | R | W | | | | 5 . 3 |
| \$P_SEARCH_POSMODE | INT | \$P_SEARCH_SMODE[n] Último valor de posição acumulado programado para SSL 0: DC (default) 1: AC 2: IC 3: DC 4: ACP 5: ACN n: número do fuso, 0 – máx. | R | W | | | | 5 . 3 |
| \$P_SAUTOGEAR | BOOL | \$P_SAUTOGEAR[n] Estágio programado do câmbio 0: Sem troca de marcha automática 1: Troca de marcha automática está ativa | R | | | | | 5 . 3 |
| \$P_SGEAR | INT | \$P_SGEAR[n] Última troca de marcha programada/solicitada GS 1: 1º estágio do câmbio programado/requisitado 2: 2. Estágio solicitado 3: 3. Estágio solicitado 4: 4. Estágio solicitado 5: 5. Estágio solicitado n: Estágio do câmbio, 0 – max. estágios | R | | | | | 5 . 3 |
| \$AC_SGEAR | INT | \$AC_SGEAR[n] Estágio atual ativo 1: 1. Estágio solicitado 2: 2. Estágio solicitado 3: 3. Estágio solicitado 4: 4. Estágio solicitado 5: 5. Estágio solicitado n: Estágio do câmbio, 0 – max. estágios | RS | | R | | | 5 . 3 |
| \$P_SMODE | INT | \$P_SMODE[n] Último modo programado para o fuso: 0: Não há fuso no canal ou fuso através do PLC 1: Modo controle de velocidade 2: Modo controle de posição 3: Modo sincronizado 4: Modo eixo n: número do fuso, 0 – número máximo de fusos | R | | | | | 3 |
| \$AC_SMODE | INT | \$AC_SMODE[n] Modo atual do fuso 0: não existem fusos 1: Modo controle de velocidade 2: Modo controle de posição 3: Modo sincronizado 4: Modo eixo n: número do fuso, 0 – número máximo de fusos | RS | | R | | | 3 |

15.2.35 Valores de polinômios para ações síncronas

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|--------|---|-------|
| \$AC_FCT1LL | REAL | \$AC_FCT1LL Limite inferior para avaliação da função FCTDEF 1 | RS | W S | R | W + 2 |
| \$AC_FCT2LL | REAL | \$AC_FCT2LL Limite inferior para avaliação da função FCTDEF 2 | RS | W S | R | W + 2 |
| \$AC_FCT3LL | REAL | \$AC_FCT3LL Limite inferior para avaliação da função FCTDEF 3 | RS | W S | R | W + 2 |
| \$AC_FCT1UL | REAL | \$AC_FCT1UL Limite superior para avaliação da função FCTDEF 1 | RS | W S | R | W + 2 |
| \$AC_FCT2UL | REAL | \$AC_FCT2UL Limite superior para avaliação da função FCTDEF 2 | RS | W S | R | W + 2 |
| \$AC_FCT3UL | REAL | \$AC_FCT3UL Limite superior para avaliação da função FCTDEF 3 | RS | W S | R | W + 2 |
| \$AC_FCT1C | REAL | \$AC_FCT1C[n] Coeficiente do polinômio a0 – a3 para avaliação da função FCTDEF 1 n: Grau do coeficiente 0 – 3 | RS | W S | R | W + 2 |
| \$AC_FCT2C | REAL | \$AC_FCT2C[n] Coeficiente do polinômio a0 – a3 para avaliação da função FCTDEF 2 n: Grau do coeficiente 0 – 3 | RS | W S | R | W + 2 |
| \$AC_FCT3C | REAL | \$AC_FCT3C[n] Coeficiente do polinômio a0 – a3 para avaliação da função FCTDEF 3 n: Grau do coeficiente 0 – 3 | RS | W S | R | W + 2 |
| \$AC_FCTLL | REAL | \$AC_FCTLL[n] Limite inferior de polinômio para ações síncronas (SYNFCT) n: número do polinômio, limitado em dados de máquina | RS | W S | R | W + 4 |
| \$AC_FCTUL | REAL | \$AC_FCTUL[n] Limite superior de polinômio para ações síncronas (SYNFCT) n: número do polinômio, limitado em dados de máquina | RS | W S | R | W + 4 |
| \$AC_FCT0 | REAL | \$AC_FCT0[n] a0 – Coeficiente do polinômio para as ações síncronas (SYNFCT) n: número do polinômio, limitado em dados de máquina | RS | W S | R | W + 4 |
| \$AC_FCT1 | REAL | \$AC_FCT1[n] a1 – Coeficiente do polinômio para as ações síncronas (SYNFCT) n: número do polinômio, limitado em dados de máquina | RS | W S | R | W + 4 |
| \$AC_FCT2 | REAL | \$AC_FCT2[n] a2 – Coeficiente do polinômio para as ações síncronas (SYNFCT) n: número do polinômio, limitado em dados de máquina | RS | W S | R | W + 4 |
| \$AC_FCT3 | REAL | \$AC_FCT3[n] a3 – Coeficiente do polinômio para as ações síncronas (SYNFCT) n: número do polinômio, limitado em dados de máquina | RS | W S | R | W + 4 |

15.2.36 Estado do canal

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------------|------|---|---------|------|-----|----------|
| \$AC_ALARM_STAT | INT | \$AC_ALARM_STAT Reações dos alarmes (selecionáveis) para ações síncronas (SYNFCT) | RS | | R | 5 |
| \$AN_ESR_TRIGGER | BOOL | \$AN_ESR_TRIGGER = 1 Disparo "Parada prolongada e recuo" | | | R W | 5 |
| \$AC_OPERATING_TIME | REAL | IF \$AC_OPERATING_TIME < 12000 GOTOB STARTMARK Tempo total dos programas NC em modo automático (em segundos) | RS | | R | 5 . 2 |
| \$AC_CYCLE_TIME | REAL | IF \$AC_CYCLE_TIME > 2400 GOTOF ALARM01 Tempo de execução do programa selecionado (em segundos) | RS | | R | 5 . 2 |
| \$AC_CUTTING_TIME | REAL | IF \$AC_CUTTING_TIME > 6000 GOTOF ACT_M06 Tempo de operação da ferramenta (em segundos) | RS | | R | 5 . 2 |
| \$AC_REQUIRED_PARTS | REAL | \$AC_REQUIRED_PARTS = ACTUAL_LOS Definição da quantidade de peças necessárias, p.e. para definição do tamanho de um lote, metas de produção diárias, etc. | RS | W S | R W | 5 . 2 |
| \$AC_TOTAL_PARTS | REAL | IF \$AC_TOTAL_PARTS > SERVICE_COUNT GOTOF MARK_END Quantidade total de peças produzidas | RS | W S | R W | 5 . 2 |
| \$AC_ACTUAL_PARTS | REAL | IF \$AC_ACTUAL_PARTS == 0 GOTOF NEW_RUN Quantidade total de peças produzidas For \$AC_ACTUAL_PARTS == \$AC_REQUIRED_PARTS, \$AC_ACTUAL_PARTS = 0 automaticamente. | RS | W S | R W | 5 . 2 |
| \$AC_SPECIAL_PARTS | REAL | \$AC_SPECIAL_PARTS = R20 Quantidade total de peças com contagem de acordo com a estratégia do usuário. Sem efeito interno. | RS | W S | R W | 5 . 2 |

15.2.37 Posições

| | | | | | | |
|---------|------|--|----|--|---|---|
| \$P_EP | REAL | \$P_EP[X] Ultimo valor programado Eixos: Eixos do canal | R | | | 2 |
| \$P_APR | REAL | \$P_APR[X] Posição do eixo no sistema de coordenadas da peça no início do movimento de aproximação para aproximação suave de contorno. Eixos: Eixos do canal | R | | | 4 |
| \$P_AEP | REAL | \$P_AEP[X] Ponto de aproximação: primeiro ponto do contorno no sistema de coordenadas da peça para aproximação suave do contorno Eixos: Eixos do canal | R | | | 4 |
| \$AA_IW | REAL | \$AA_IW[X] Valor atual no sistema de coordenadas da peça (WCS) Eixos: Eixos do canal | RS | | R | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$AA_IEN | REAL | \$AA_IEN[X] Valor atual no sistema ajustável de origem (SOS). Eixos: Eixos do canal | RS | | R | 5 |
| \$AA_IBN | REAL | \$AA_IBN[X] Valor atual no sistema básico de origem (BOS). Eixos: Eixos do canal | RS | | R | 5 |
| \$AA_IB | REAL | \$AA_IB[X] Valor atual no sistema de coordenadas básico (BCS) Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_IM | REAL | \$AA_IM[X] Valor atual no sistema de coordenadas da máquina (MCS). Eixos: GEOAX, eixos do canal, eixos de máquina | RS | | R | 2 |

15.2.38 Eixos indexados

| | | | | | | |
|---------------------------|-----|--|----|--|---|---|
| \$AA_ACT_INDEX_AX_POS_NO | INT | \$AA_ACT_INDEX_AX_POS_NO[X] 0: não há eixos indexados, portanto não há tabela de indexação disponível. > 0: Número da última posição indexada atingida ou ultrapassada Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |
| \$AA_PROG_INDEX_AX_POS_NO | INT | \$AA_PROG_INDEX_AX_POS_NO[X] 0: Não é eixo indexado, portanto não existe posição de indexação disponível ou o eixo indexado não se encontra atualmente em movimentação indexada > 0: Número da posição indexada programada Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |

15.2.39 Limite de frequência do encoder

| | | | | | | |
|------------------|------|---|----|--|---|---|
| \$AA_ENC_ACTIVE | BOOL | \$AA_ENC_ACTIVE[X] O sistema de medição ativo está operando abaixo da frequência limite do encoder Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 4 |
| \$AA_ENC1_ACTIVE | BOOL | \$AA_ENC1_ACTIVE[X] O encoder 1 está operando abaixo de sua frequência limite Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 4 |
| \$AA_ENC2_ACTIVE | BOOL | \$AA_ENC2_ACTIVE[X] O encoder 2 está operando abaixo de sua frequência limite Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 4 |

15.2.40 Valores do encoder

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|--------|---|---|
| \$VA_IM | REAL | \$VA_IM[X] Valor atual do encoder no sistema de coordenadas da máquina (medido no sistema de medição ativo), os valores atuais de compensação são corrigidos (correção de folga do fuso, compensação de erro de passo, erros de quadrante) Eixos: Eixos de máquina | RS | | R | 4 |
| \$VA_IM1 | REAL | \$VA_IM1[X] Valor atual no sistema de coordenadas da máquina (medido com o encoder 1), compensações corrigidas Eixos: Eixos de máquina | RS | | R | 4 |
| \$VA_IM2 | REAL | \$VA_IM2[X] Valor atual no sistema de coordenadas da máquina (medido com o encoder 2), compensações corrigidas Eixos: Eixos de máquina | RS | | R | 4 |
| \$AA_MW | REAL | \$AA_MW[X] Valor medido no sistema de coordenadas da peça Eixos: Eixos do canal | R | W S | R | 2 |
| \$AA_MM | REAL | \$AA_MM[X] Valor medido no sistema de coordenadas da máquina Eixos: Eixos de máquina | R | W S | R | 2 |
| \$AA_MW1 | REAL | \$AA_MW1[X] Resultado da medição axial Dispara o evento 1 no WCS Eixos: Eixos do canal | R | W S | R | 4 |
| \$AA_MW2 | REAL | \$AA_MW2[X] Resultado da medição axial Dispara o evento 2 no WCS Eixos: Eixos do canal | R | W S | R | 4 |
| \$AA_MW3 | REAL | \$AA_MW3[X] Resultado da medição axial Dispara o evento 2 no WCS Eixos: Eixos do canal | R | W S | R | 4 |
| \$AA_MW4 | REAL | \$AA_MW4[X] Resultado da medição axial Dispara o evento 2 no WCS Eixos: Eixos do canal | R | W S | R | 4 |

15.2.41 Medição axial

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | | Sínc | O | S |
|---------------|------|---|---------|--------|------|---|---|
| \$AA_MM1 | REAL | \$AA_MM1[X] Resultado da medição axial Dispara o evento 1 no MCS Eixos: Eixos do canal | R | W S | R | W | 4 |
| \$AA_MM2 | REAL | \$AA_MM2[X] Resultado da medição axial Dispara o evento 2 no MCS Eixos: Eixos do canal | R | W S | R | W | 4 |
| \$AA_MM3 | REAL | \$AA_MM3[X] Resultado da medição axial Dispara o evento 3 no MCS Eixos: Eixos do canal | R | W S | R | W | 4 |
| \$AA_MM4 | REAL | \$AA_MM4[X] Resultado da medição axial Dispara o evento 4 no MCS Eixos: Eixos do canal | R | W S | R | W | 4 |
| \$AA_MEAACT | BOOL | \$AA_MEAACT[X] Valor TRUE caso a medição axial esteja ativa para X Eixos: eixo geométrico, eixo do canal, eixo de máquina | | | R | | 4 |

15.2.42 Deslocamentos

| | | | | | | | |
|-----------------------|------|---|----|---|---|---|---|
| \$AC_DRF | REAL | \$AC_DRF[X] Deslocamento DRF Eixos: Eixos do canal | RS | | R | | 2 |
| \$AC_PRESET | REAL | \$AC_PRESET[X] Último valor de preset especificado Eixos: Eixos do canal | RS | | R | | 2 |
| \$AA_ETRANS | REAL | \$AA_ETRANS[X] deslocamento de origem externo Eixos: Eixos do canal | R | W | | | 2 |
| \$AA_OFF | REAL | \$AA_OFF[X] Movimento sobreposto para o eixo programado Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | W | R | W | 3 |
| \$AA_OFF_LIMIT | INT | \$AA_OFF_LIMIT[eixo] Valor limite para deslocamento axial \$AA_OFF[eixo] 0: Valor limite não atingido 1: Valor limite atingido na direção positiva do eixo -1: Valor limite atingido na direção negativa do eixo Eixos: Eixos do canal | RS | | R | | 4 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$AC_RETPOINT | REAL | \$AC_RETPOINT[X] Ponto de reset no contorno para reaproximação Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_SOFTENDP | REAL | \$AA_SOFTENDP[X] Limite de posição software na direção positiva Eixos: Eixos de máquina | RS | | R | 2 |
| \$AA_SOFTENDN | REAL | \$AA_SOFTENDN[X] Limite de posição software na direção negativa Eixos: Eixos de máquina | RS | | R | 2 |

15.2.43 Distâncias axiais

| | | | | | | |
|-----------|------|---|----|--|---|---|
| \$AA_DTBW | REAL | \$AA_DTBW[X] Trajetória axial a partir do início do bloco no sistema de coordenadas da peça para eixos de posicionamento e sincronismo em movimentos com ações síncronas. A posição programada é decisiva para o cálculo da trajetória; caso o eixo seja um eixo acoplado, a posição não é considerada. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_DTBB | REAL | \$AA_DTBB[X] Distância axial a partir do início do bloco no sistema de coordenadas básico para eixos de posicionamento e sincronismo em movimentos com ações síncronas A posição programada é decisiva para o cálculo da trajetória; caso o eixo seja um eixo acoplado, a posição não é considerada. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_DTEW | REAL | \$AA_DTEW[X] Distância axial para o fim do bloco no sistema de coordenadas da peça para eixos de posicionamento e sincronismo em movimentos com ações síncronas A posição programada é decisiva para o cálculo da trajetória; caso o eixo seja um eixo acoplado, a posição não é considerada. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_DTEB | REAL | \$AA_DTEB[X] Distância axial para o fim do bloco no sistema de coordenadas básico para eixos de posicionamento e sincronismo em movimentos com ações síncronas A posição programada é decisiva para o cálculo da trajetória; caso o eixo seja um eixo acoplado, a posição não é considerada. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |

15.2.44 Oscilação

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------------------------|------|---|---------|------|---|---|
| \$AA_DTEPW | REAL | \$AA_DTEPW[X] Distância a ser percorrida pelo eixo para avanço da oscilação no sistema de coordenadas da peça. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_DTEPB | REAL | \$AA_DTEPB[X] Distância a ser percorrida pelo eixo para avanço da oscilação no sistema de coordenadas básico. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_OSCILL_REVERSE_POS1 | REAL | \$AA_OSCILL_REVERSE_POS1[X] Posição atual de reversão 1 para oscilação. Em ações síncronas, o valor do dado setting \$SA_OSCILL_REVERSE_POS1 é calculado online Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 3 |
| \$AA_OSCILL_REVERSE_POS2 | REAL | \$AA_OSCILL_REVERSE_POS2[X] Posição atual de reversão 2 para oscilação. Em ações síncronas, o valor do dado setting \$SA_OSCILL_REVERSE_POS2 é calculado online Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 3 |
| \$AA_DELT | REAL | \$AA_DELT[X] Distância axial residual memorizada no sistema de coordenadas da peça após um comando de cancelamento de percurso restante com ações síncronas. Eixos: eixo geométrico, eixo do canal, eixo de máquina | | | R | 2 |
| \$P_FA | REAL | \$P_FA[X] Último valor de avanço programado Eixos: Eixos do canal | R | | | 2 |

15.2.45 Velocidades dos eixos

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S | |
|---------------|------|---|---------|------|---|---|---|
| \$AA_OVR | REAL | <p>\$AA_OVR[X]</p> <p>Axial override for movimentos com ações síncronas</p> <p>A componente de multiplicação override atua adicionalmente ao override de usuário, o programado e o de transformação. Entretanto, o fator total é restrito à 200%.</p> <p>Deve ser re-escrito a cada ciclo de interpolação, caso contrário, será considerado como sendo 100%.</p> <p>\$AA_OVR[S1] altera o override do fuso.</p> <p>O override definido através dos dados de máquina</p> <p>\$MN_OVR_FACTOR_LIMIT_BIN,</p> <p>\$MN_OVR_FACTOR_FEEDRATE[30],</p> <p>\$MN_OVR_FACTOR_AX_SPEED[30] e</p> <p>\$AA_OVR_FACTOR_SPIND_SPEED</p> <p>não é excedido.</p> <p>Esta variável pode ser somente acessada a partir de ações síncronas</p> <p>Eixos: Eixos do canal</p> | | | R | W | 2 |
| \$AA_VC | REAL | <p>\$AA_VC[X]</p> <p>A valor do avanço aditivo para correção da trajetória para ações síncronas.</p> <p>A valor do avanço aditivo para correção da trajetória para ações síncronas.</p> <p>Com um override de 0, o valor de compensação não possui efeito, caso contrário o override não teria impacto no valor de compensação.</p> <p>O valor de compensação não pode causar um override total negativo.</p> <p>O valor superior é limitado de forma que as velocidades máximas dos eixos e acelerações não sejam excedidas.</p> <p>O cálculo com componentes de avanço diferentes não é afetada por \$AA_VC.</p> <p>Os valores de override definidos pelos dados de máquina</p> <p>\$MN_OVR_FACTOR_LIMIT_BIN,</p> <p>\$MN_OVR_FACTOR_FEEDRATE[30],</p> <p>\$MN_OVR_FACTOR_AX_SPEED[30] e</p> <p>\$MN_OVR_FACTOR_SPIND_SPEED</p> <p>não podem ser excedidos. O override de avanço aditivo é limitado de forma que o avanço resultante não exceda o valor de override máximo do avanço programado.</p> <p>Eixos: Eixos do canal</p> | | | R | W | 2 |
| \$AA_VACTB | REAL | <p>\$AA_VACTB[X]</p> <p>Velocidade do eixo no sistema de coordenadas base</p> <p>Esta variável pode ser somente acessada a partir de ações síncronas</p> <p>Eixos: Eixos do canal</p> | RS | | R | | 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|---|
| \$AA_VACTW | REAL | \$AA_VACTW[X] Velocidade do eixo no sistema de coordenadas da peça. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 2 |
| \$AA_VACTM | REAL | \$AA_VACTM[X] Velocidade do eixo, valor desejado relacionado ao sistema de coordenadas da máquina. Pode também ser lido para eixos de PLC. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 4 |
| \$VA_VACTM | REAL | \$VA_VACTM[X] Velocidade do eixo, valor atual com relação ao sistema de coordenadas da máquina. A variável retorna um valor indefinido caso a frequência limite do encoder tenha sido atingida. Esta variável pode ser somente acessada a partir de ações síncronas Eixos: Eixos do canal | RS | | R | 4 |

15.2.46 Dados de acionamento

| | | | | | | |
|----------------|------|---|----|--|---|-------------|
| \$AA_LOAD | REAL | \$AA_LOAD[X] Uso do acionamento em % (somente para o 611D) Eixos: Eixos do canal | RS | | R | 2 |
| \$VA_LOAD | REAL | \$VA_LOAD[X] Uso do acionamento em % (somente para o 611D) Eixos: Eixos do canal | RS | | R | 5 . 1 |
| \$AA_TORQUE | REAL | \$AA_TORQUE[X] Valor desejado de torque do acionamento em Nm (só para o 611D) Eixos: Eixos do canal | RS | | R | 2 |
| \$VA_TORQUE | REAL | \$VA_TORQUE[X] Valor desejado de torque do acionamento em Nm (só para o 611D) Eixos: Eixos do canal | RS | | R | 5 . 1 |
| \$AA_POWER | REAL | \$AA_POWER[X] Potência ativa do acionamento em W (somente para o 611D) Eixos: Eixos do canal | RS | | R | 2 |
| \$VA_POWER | REAL | \$VA_POWER[X] Potência ativa do acionamento em W (somente para o 611D) Eixos: Eixos do canal | RS | | R | 5 . 1 |
| \$AA_CURR | REAL | \$AA_CURR[X] Valor atual da corrente para o eixo ou fuso em A (só para o 611D) Eixos: Eixos do canal | RS | | R | 2 |
| \$VA_CURR | REAL | \$VA_CURR[X] Valor atual da corrente para o eixo ou fuso em A (só para o 611D) Eixos: Eixos do canal | RS | | R | 5 . 1 |
| \$VA_VALVELIFT | REAL | \$VA_VALVELIFT[X] Valor de abertura da válvula em mm (somente para 611D hidráulico) Eixos: Eixos do canal | RS | | R | 5 . 1 |

15.2 Lista das variáveis de sistema

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-----------------|------|--|---------|------|---|-------------|
| \$VA_PRESSURE_A | REAL | \$VA_PRESSURE_A[X] Pressão do lado A do cilindro em bar (somente para 611D hidráulico) Eixos: Eixos do canal | RS | | R | 5 . 1 |
| \$VA_PRESSURE_B | REAL | \$VA_PRESSURE_B[X] Pressão do lado B do cilindro em bar (somente para 611D hidráulico) Eixos: Eixos do canal | RS | | R | 5 . 1 |

15.2.47 Estado do eixo

| | | | | | | |
|-----------|-----|--|----|--|---|---|
| \$AA_STAT | INT | \$AA_STAT[X] Estado do eixo: 0: Não há estado disponível 1: Eixo em movimento 2: Eixo atingiu o IPO final (somente para os eixos do canal) 3: Eixo encontra-se em posição (parada exata grossa) -p/todos eixos 4: Eixo encontra-se em posição (parada exata fina) -p/todos eixos Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 4 |
| \$AA_REF | INT | \$AA_REF[X] Estado do eixo: 0: Eixo não referenciado 1: Eixo referenciado Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |
| \$AA_TYP | INT | \$AA_TYPT[X] Tipo de eixo: 0: Eixo em outro canal 1: Eixo de canal de canal local 2: Eixo neutro 3: Eixo PLC 4: Eixo de oscilação 5: Eixo neutro sendo movimentado em JOG 6: Eixo principal para eixo seguidor 7: Eixo seguidor 8: Eixo de comando 9: Eixo de ciclo compilado Eixos: eixo geométrico, eixo de canal | RS | | R | 4 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|----------------------|------|--|---------|------|---|---|
| \$AA_FXS | INT | \$AA_FXS[X] Estado de "Parada contra encosto fixo" 0: Eixo não se encontra no encosto 1: Eixo parado contra encosto com sucesso 2: Aproximação contra encosto fixo falhou 3: Seleção para encosto fixo ativa 4: Desliga seleção de movimento contra encosto fixo Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 2 |
| \$AA_COUP_ACT | INT | \$AA_COUP_ACT[SPI(2)] Estado atual do acoplamento com eixo/fuso seguidor: 0: Eixo/fuso não acoplado a um eixo/fuso principal 3: Eixo seguidor tangencial 4: Fuso sincronizado para acoplamento 8: Eixo é tracionado 16: Eixo seguidor de um valor mestre acoplado Os valores respectivos se aplicam à um acoplamento. Caso existam vários acoplamentos ativos para um eixo seguidor, estes serão representados pela soma numérica dos valores relevantes. Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 2 |

15.2.48 Câmbio eletrônico 1

| | | | | | | |
|------------------------------|------------|---|----|--|---|--------|
| \$AA_EG_SYNFA | REAL | \$AA_EG_SYNFA[a] a: Eixo seguidor Posição sincronizada do eixo seguidor Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |
| \$P_EG_BC | STRIN G | \$P_EG_BC[a] Condição de troca de bloco para EGONSYN, EGON, WAITC. Eixos: Eixos do canal | R | | | 5 2 |
| \$AA_EG_NUM_L A | INT | \$AA_EG_NUM_LA[a] a: Eixo seguidor Número de eixos principais especificados com EGDEF Eixos: eixo geométrico, eixo de canal | RS | | R | 5 |
| \$VA_EG_SYNCDI FF | REAL | \$VA_EG_SYNCDIFF[a] a: Eixo seguidor Diferença de sincronismo Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|---|---------|------|---|-------------|
| \$AA_EG_AX | AXIS | \$AA_EG_AX[n,a] n: Índice para eixo principal a: Eixo seguidor Identificador para o enésimo eixo principal n: Índice para eixo principal (enésimo eixo principal) Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 . 2 |

15.2.49 Valor principal de acoplamento

| | | | | | | | |
|------------------|------|---|----|--------|---|---|---|
| \$AA_LEAD_SP | REAL | \$AA_LEAD_SP[LW] Valor simulado mestre – Posição | RS | W S | R | W | 4 |
| \$AA_LEAD_SV | REAL | \$AA_LEAD_SV[LW] Valor simulado mestre – Velocidade | RS | W S | R | W | 4 |
| \$AA_LEAD_P_TURN | REAL | \$AA_LEAD_P_TURN[LW] O valor de posição principal atual da peça é perdido com a redução de módulo. O valor atual de posição principal (utilizado para cálculos internos) é \$AA_LEAD_P[LW] + \$AA_LEAD_P_TURN[LW] Caso MV seja um eixo módulo, \$AA_LEAD_P_TURN é um inteiro múltiplo de \$MA_MODULO_RANGE. Caso MV não seja um eixo módulo, \$AA_LEAD_P_TURN é sempre 0. Exemplo 1: \$MA_MODULO_RANGE[LW]=360 \$AA_LEAD_P[LW] =290 \$AA_LEAD_P_TURN[LW] =720 O valor principal de posição atual (usado para cálculos internos) é 1010. Exemplo 2: \$MA_MODULO_RANGE[LW]=360 \$AA_LEAD_P[LW] =290 \$AA_LEAD_P_TURN[LW] =-360 O valor principal de posição atual (usado para cálculos internos) é -70. | RS | | R | | 4 |
| \$AA_LEAD_P | REAL | \$AA_LEAD_P[LW] Valor atual mestre – posição (em módulo) Caso MV seja um eixo módulo, a seguinte regra é sempre válida: 0 <= \$AA_LEAD_P[LW] <= \$MA_MODULO_RANGE[LW] | RS | | R | | 4 |
| \$AA_LEAD_V | REAL | \$AA_LEAD_V[LW] Valor atual mestre – Velocidade | RS | | R | | 4 |
| \$AA_SYNC | INT | \$AA_SYNC [FA] Estado do acoplamento ou eixo seguidor com relação ao valor de acoplamento mestre 0 => Sem sincronismo 1 => Sincronismo grosso (myVdiOut->getSinchCoarse() == TRUE) 2 => Sincronismo fino (myVdiOut->getSinchFine() == TRUE) 3 => Grosso e fino Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | | 4 |

15.2.50 Fuso sincronizado

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-----------------------|------|---|---------|------|---|---|
| \$AA_COUP_OFFS | REAL | \$AA_COUP_OFFS[S2] Deslocamento de posição para fuso sincronizado como valor desejado S2 é fuso seguidor | RS | | R | 2 |
| \$VA_COUP_OFFS | REAL | \$VA_COUP_OFFS[SPI(2)] Deslocamento de posição para fuso sincronizado como valor atual SPI(2) é fuso seguidor | RS | | R | 2 |

15.2.51 Safety Integrated 1

| | | | | | | |
|----------------------|------|---|----|--------|--------|---|
| \$VA_IS | REAL | \$VA_IS[X] Posição atual segura (SISITEC) Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 3 |
| \$AA_SCTTRACE | BOOL | \$AA_SCTTRACE[X] = 1 Escrita: disparo no IPO para servo trace 0: Sem ação !0: Inicia disparo Leitura: Sempre 0, pois o bit de auto disparo é retornado pela interface. 0: Valor atual (sem estado) Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | W S | R W | 4 |
| \$AA_SCTTRACE | BOOL | \$AA_SCTTRACE[X] = 1 Escrita: Partida no IPO para servo trace 0: Sem ação !0: Inicia disparo Leitura: Sempre 0, pois o bit de auto disparo é retornado pela interface. 0: Valor atual (sem estado) Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | W S | R W | 4 |
| \$VA_DPE | BOOL | \$VA_DPE[X1] Estado da habilitação de potência para um eixo de máquina Eixos: Eixos de máquina | RS | | R | 5 |
| \$AA_ACC | REAL | \$AA_ACC Valor de aceleração atual para o eixo. \$AA_ACC = \$MA_MAX_AX_ACCEL * Deslocamento programado de aceleração | RS | | R | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$AA_MOTEND | INT | \$AA_MOTEND Critério de fim de movimento para interpolação em 1 eixo 1 = Fim de movimento na parada exata FINE 2 = Fim de movimento na parada exata COARSE 3 = Fim de movimento com parada exata, parada IPO Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |
| \$AA_SCPAR | INT | \$AA_SCPAR Lê conjunto de parâmetros atual Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |

15.2.52 Parada prolongada e recuo

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-----------------|------|---|---------|--------|--------|---|
| \$AA_ESR_STAT | INT | \$AA_ESR_STAT[X] Estado para "Parada prolongada e recuo", codificada em bits BIT0: Gerador da operação disparado BIT1: Recuo disparado BIT2: Parada externa disparada BIT3: Subtenção DC BIT4: Velocidade mínima do gerador Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 |
| \$AA_ESR_ENABLE | BOOL | \$AA_ESR_ENABLE[X] = 1 Habilita "Parada prolongada e recuo" Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | W S | R W | 5 |

15.2.53 Recipiente de eixos

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------|------|--|---------|------|---|---|
| \$AN_AXCTSWA | BOOL | EVERY \$AN_AXCTSWA[n] == TRUE DO M99 Leitura: TRUE: Uma rotação no recipiente de eixos encontra-se em andamento, no recipiente de eixos de nome n. FALSE: Nenhuma rotação encontra-se ativa | | | R | 5 |
| \$AN_AXCTAS | INT | Leitura: Recipiente de eixos rotação atual: O número do alojamento rotacionado para o recipiente atual será indicado para o recipiente de eixo de nome n. A faixa de valores vai de 0 até o número máximo de recipientes de eixo –1. | | | R | 5 |
| \$AC_AXCTSWA | BOOL | IF \$AC_AXCTSWA[n] == TRUE GOTOB MARK1 Leitura: TRUE: O canal habilitou a rotação do recipiente de nome n e a rotação ainda não foi completada. FALSE: A rotação do recipiente está encerrada | | | R | 5 |

15.2.54 Câmbio eletrônico 2

| | | | | | | |
|--------------------|------|--|----|--|---|-------------|
| \$AA_EG_TYPE | INT | \$AA_EG_TYPE[a,b] a: Eixo seguidor b: Eixo principal Tipo de acoplamento para o eixo principal b 0: Valor atual de acoplamento 1: Valor desejado de acoplamento Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 . 2 |
| \$AA_EG_NUMER A | REAL | \$AA_EG_NUMER[a,b] a: Eixo seguidor b: Eixo principal Numerador do fator de acoplamento para o eixo principal b Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 . 2 |
| \$AA_EG_DENOM | REAL | \$AA_EG_DENOM[a,b] a: Eixo seguidor b: Eixo principal Denominador do fator de acoplamento do eixo principal b Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 . 2 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|----------------|------|--|---------|------|---|-------------|
| \$AA_EG_SYN | REAL | \$AA_EG_SYN[a,b] a: Eixo seguidor b: Eixo principal Posição sincronizada para o eixo principal b Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 . 2 |
| \$AA_EG_ACTIVE | BOOL | \$AA_EG_ACTIVE[a,b] a: Eixo seguidor b: Eixo principal O acoplamento para o eixo principal b encontra-se ativo, ou seja, ligado Eixos: eixo geométrico, eixo do canal, eixo de máquina | RS | | R | 5 . 2 |

15.2.55 Safety Integrated 2

| | | | | | | | | |
|-------------|------|--|----|--------|---|---|--|---|
| \$A_INSE | BOOL | \$A_INSE[n] Imagem de uma entrada safety (interface externa NCI) n: Número da entrada 1-... | RS | | R | | | 4 |
| \$A_INSED | INT | \$A_INSED[n] Imagem de um sinal de entrada safety (interface externa NCI) n: Número da palavra de entrada 1-... | RS | | R | | | 5 |
| \$A_INSEP | BOOL | \$A_INSEP[n] Imagem de um sinal de uma entrada safety (interface externa com PLC) n: Número da entrada 1- ... | RS | | R | | | 4 |
| \$A_INSEPD | INT | \$A_INSEPD[n] Imagem de um sinal de entrada safety (interface externa com PLC) n: Número da palavra de entrada 0 – ... | RS | | R | | | 5 |
| \$A_OUTSE | BOOL | \$A_OUTSE[n] Imagem de um sinal de saída safety (Interface externa NCI) n: Número da saída1 – ... | RS | W S | R | W | | 4 |
| \$A_OUTSED | INT | \$A_OUTSED[n] Imagem de um sinal de saída safety (Interface externa NCI) n: Número da palavra de saída1 – ... | RS | W S | R | W | | 5 |
| \$A_OUTSEP | BOOL | \$A_OUTSEP[n] Imagem de um sinal de saída safety (interface externa com PLC) n: Número da saída1 – ... | RS | | R | | | 4 |
| \$A_OUTSEPD | INT | \$A_OUTSEPD[n] Imagem de um sinal de saída safety (interface externa com PLC) n: Número da palavra de saída0 – ... | RS | | R | | | 5 |
| \$A_INSI | BOOL | \$A_INSI[n] Imagem de um sinal de entrada safety (interface interna NCI) n: Número da entrada 1 – ... | RS | | R | | | 4 |
| \$A_INSID | INT | \$A_INSID[n] Imagem dos sinais de entrada safety (interface interna NCI) n: Número da palavra de entrada 1 – ... | RS | | R | | | 5 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|-----------------------|------|--|---------|--------|-------------|-------------|
| \$A_INSIP | BOOL | \$A_INSIP[n] Imagem de um sinal de entrada safety (interface interna com PLC) n: Número da palavra de entrada 1 – ... | RS | | R | 4 |
| \$A_INSIPD | INT | \$A_INSIPD[n] Imagem de um sinal de entrada safety (interface interna com PLC) n: Número da palavra de entrada 1 – ... | RS | | R | 5 |
| \$A_OUTSI | BOOL | \$A_OUTSI[n] Imagem de sinal de saída safety(interface interna NCI) n: Número da saída1 – ... | RS | W S | R W | 4 |
| \$A_OUTSID | INT | \$A_OUTSID[n] Imagem de sinais de saída safety(interface interna NCI) n: Número da palavra de saída1 – ... | RS | W S | R W | 5 |
| \$A_OUTSIP | BOOL | \$A_OUTSIP[n] Imagem de sinal de saída safety(interface interna com PLC) n: Número da saída1 – ... | RS | | R | 4 |
| \$A_OUTSIPD | INT | \$A_OUTSIPD[n] Imagem de sinais de saída safety(interface interna com PLC) n: Número da palavra de saída1 – ... | RS | | R | 5 |
| \$A_MARKERSI | BOOL | \$A_MARKERSI[n] Memórias para programação safety n: Número da memória1 – ... | RS | W S | R W + | 4 |
| \$A_MARKERSID | INT | \$A_MARKERSID[n] Palavra de memória (32bits) para programação safety n: Número da palavra de memória 1 – ... | RS | W S | R W + | 5 . 1 |
| \$A_MARKERSIP | BOOL | \$A_MARKERSIP[n] Imagem das memórias safety no PLC n: Número da memória1 – ... | RS | | R + | 4 |
| \$A_MARKERSIPD | INT | \$A_MARKERSIPD[n] Imagem das palavras safety n: Número da palavra de memória 1 – ... | RS | | R + | 5 . 1 |
| \$A_TIMERSI | REAL | \$A_TIMERSI[n] Temporizador safety – segundos O tempo é contado internamente em múltiplos de ciclo de interpolação; A contagem para as variáveis de tempo é iniciada quando se atribui um valor para a variável \$A_TIMERSI[n]=<valor inicial>. Para parar a contagem de tempo, atribuir um valor negativo: \$A_TIMERSI[n]=-1. O valor atual de tempo pode ser lido durante a contagem ou quando o contador estiver parado. Quando o temporizador é parado através da atribuição do valor -1, o valor mais atualizado de tempo memorizado pode ser acessado n: Número do temporizador 1 – ... | RS | W S | R W + | 4 |

| Identificador | Tipo | Descrição: Variável de sistema/faixa de valores/índice | Prog.NC | Sínc | O | S |
|---------------------|------|--|---------|--------|-------------|---|
| \$A_STATSID | INT | \$A_STATSID Safety: Estado da checagem cruzada entre a NCK e o PLC Caso o valor seja diferente de zero, existe, erros de cross check | RS | | R | 5 |
| \$A_CMDSI | BOOL | \$A_CMDSI[n] Safety: palavra de controle para checagem cruzada entre o PLC e NCK Índice do array n = 1: Aumenta tempo para monitoração de alteração para 10 s. n: Número do sinal de controle para checagem cruzada NCK-PLC | RS | W S | R W + | 5 |
| \$A_LEVELSID | INT | \$A_LEVELSID Safety: Exibe a alteração de sinal no nível de monitoração. Indica que a quantidade atual de sinais marcada para checagem cruzada de dados. | RS | | R | 5 |

Apêndice

| | | |
|---|---------------------------------|-------|
| A | Índice..... | A-577 |
| B | Comandos, Identificadores | A-591 |

A Index**\$**

\$A_CMDSI, 15-574

\$A_DAY, 15-552

\$A_DBB, 15-549

\$A_DBD, 15-549

\$A_DBR, 15-549

\$A_DBW, 15-549

\$A_DLB, 15-549

\$A_DLD, 15-549

\$A_DLR, 15-549

\$A_DLW, 15-549

\$A_DNO, 15-540

\$A_GG, 15-541

\$A_HOUR, 15-552

\$A_IN, 15-548

\$A_INA, 15-548

\$A_INCO, 15-548

\$A_INSE, 15-572

\$A_INSED, 15-572

\$A_INSEP, 15-572

\$A_INSEPD, 15-572

\$A_INSI, 15-572

\$A_INSID, 15-572

\$A_INSIP, 15-573

\$A_INSIPD, 15-573

\$A_LEVELSID, 15-574

\$A_LINK_TRANS_RATE, 15-549

\$A_MARKERSI, 15-573

\$A_MARKERSID, 15-573

\$A_MARKERSIP, 15-573

\$A_MARKERSIPD, 15-573

\$A_MINUTE, 15-552

\$A_MONIFACT, 15-540

\$A_MONTH, 15-552

\$A_MSECOND, 15-552

\$A_OUT, 15-548

\$A_OUTA, 15-548

\$A_OUTSE, 15-572

\$A_OUTSED, 15-572

\$A_OUTSEP, 15-572

\$A_OUTSEPD, 15-572

\$A_OUTSI, 15-573

\$A_OUTSID, 15-573

\$A_OUTSIP, 15-573

\$A_OUTSIPD, 15-573

\$A_PBB_IN, 15-550

\$A_PBB_OUT, 15-550

\$A_PBD_IN, 15-550

\$A_PBD_OUT, 15-550

\$A_PBR_IN, 15-550

\$A_PBR_OUT, 15-550

\$A_PBW_IN, 15-550

\$A_PBW_OUT, 15-550

\$A_PROBE, 15-544

\$A_PROTO, 15-546

\$A_SECOND, 15-552

\$A_STATSID, 15-574

\$A_TIMERSI, 15-573

\$A_TOOLMLN, 15-540

\$A_TOOLMN, 15-540

\$A_YEAR, 15-552

\$AA_ACC, 15-569

\$AA_ACT_INDEX_AX_POS_NO, 15-559

\$AA_COUP_ACT, 9-309, 9-322, 13-439, 15-567

\$AA_COUP_OFFS, 13-439, 15-569

\$AA_CURR, 15-565

\$AA_DELT, 15-563

\$AA_DTBB, 15-562

\$AA_DTBW, 15-562

\$AA_DTEB, 15-562

\$AA_DTEPB, 15-563

\$AA_DTEPW, 15-563

\$AA_DTEW, 15-562

\$AA_EG_ACTIVE, 15-572

\$AA_EG_AX, 15-568
\$AA_EG_DENOM, 15-571
\$AA_EG_NUM_LA, 15-567
\$AA_EG_NUMERA, 15-571
\$AA_EG_SYN, 15-572
\$AA_EG_SYNFA, 15-567
\$AA_EG_TYPE, 15-571
\$AA_ENC_ACTIVE, 15-559
\$AA_ENC_COMP, 15-534
\$AA_ENC_COMP_IS_MODULO, 15-535
\$AA_ENC_COMP_MAX, 15-534
\$AA_ENC_COMP_MIN, 15-534
\$AA_ENC_COMP_STEP, 15-534
\$AA_ENC1_ACTIVE, 15-559
\$AA_ENC2_ACTIVE, 15-559
\$AA_ESR_ENABLE, 15-570
\$AA_ESR_STAT, 15-570
\$AA_ETRANS, 15-561
\$AA_FXS, 15-567
\$AA_IB, 15-559
\$AA_IBN, 15-559
\$AA_IEN, 15-559
\$AA_IM, 15-559
\$AA_IW, 15-558
\$AA_LEAD_P, 15-568
\$AA_LEAD_P_TURN, 15-568
\$AA_LEAD_SP, 9-322, 15-568
\$AA_LEAD_SV, 9-322, 15-568
\$AA_LEAD_V, 15-568
\$AA_LOAD, 15-565
\$AA_MEAFACT, 15-561
\$AA_MM, 15-560
\$AA_MM1, 15-561
\$AA_MM2, 15-561
\$AA_MM3, 15-561
\$AA_MM4, 15-561
\$AA_MOTEND, 15-570
\$AA_MOTENDA, 5-188
\$AA_MW, 15-560
\$AA_MW1, 15-560
\$AA_MW2, 15-560
\$AA_MW3, 15-560
\$AA_MW4, 15-560
\$AA_OFF, 15-561
\$AA_OFF_LIMIT, 15-561
\$AA_OSCILL_REVERSE_POS1, 15-563
\$AA_OSCILL_REVERSE_POS2, 15-563
\$AA_OVR, 15-564
\$AA_POWER, 15-565
\$AA_PROG_INDEX_AX_POS_NO, 15-559
\$AA_QEC, 15-535
\$AA_QEC_ACCEL_1, 15-535
\$AA_QEC_ACCEL_2, 15-535
\$AA_QEC_ACCEL_3, 15-535
\$AA_QEC_COARSE_STEPS, 15-535
\$AA_QEC_DIRECTIONAL, 15-536
\$AA_QEC_FINE_STEPS, 15-535
\$AA_QEC_LEARNING_RATE, 15-536
\$AA_QEC_MEAS_TIME_1, 15-535
\$AA_QEC_MEAS_TIME_2, 15-535
\$AA_QEC_MEAS_TIME_3, 15-536
\$AA_QEC_TIME_1, 15-536
\$AA_QEC_TIME_2, 15-536
\$AA_REF, 15-566
\$AA_SCPAR, 5-190, 15-570
\$AA_SCTRACE, 15-569
\$AA_SOFTENDN, 15-562
\$AA_SOFTENDP, 15-562
\$AA_STAT, 15-566
\$AA_SYNC, 15-568
\$AA_TORQUE, 15-565
\$AA_TYP, 15-566
\$AA_VACTB, 15-564
\$AA_VACTM, 15-565
\$AA_VACTW, 15-565
\$AA_VC, 15-564
\$AC_ACTUAL_PARTS, 15-558
\$AC_ALARM_STAT, 15-558
\$AC_ASUP, 15-545
\$AC_AXCTSWA, 15-571
\$AC_CUTTING_TIME, 15-558
\$AC_CYCLE_TIME, 15-558
\$AC_DELT, 15-553
\$AC_DRF, 15-561

| | |
|-------------------------------------|--|
| \$AC_DTBB , 15-553 | \$AC_REQUIRED_PARTS , 15-558 |
| \$AC_DTBW , 15-553 | \$AC_RETPOINT , 15-562 |
| \$AC_DTEB , 15-553 | \$AC_SDIR , 15-555 |
| \$AC_DTEW , 15-553 | \$AC_SGEAR , 15-556 |
| \$AC_FCT0 , 15-557 | \$AC_SMODE , 15-556 |
| \$AC_FCT1 , 15-557 | \$AC_SPECIAL_PARTS , 15-558 |
| \$AC_FCT1C , 15-557 | \$AC_STAT , 15-543 |
| \$AC_FCT1LL , 15-557 | \$AC_SYNA_MEM , 15-543 |
| \$AC_FCT1UL , 15-557 | \$AC_TC_FCT , 15-551 |
| \$AC_FCT2 , 15-557 | \$AC_TC_LFN , 15-551 |
| \$AC_FCT2C , 15-557 | \$AC_TC_LFO , 15-551 |
| \$AC_FCT2LL , 15-557 | \$AC_TC_LTN , 15-551 |
| \$AC_FCT2UL , 15-557 | \$AC_TC_LTO , 15-551 |
| \$AC_FCT3 , 15-557 | \$AC_TC_MFN , 15-551 |
| \$AC_FCT3C , 15-557 | \$AC_TC_MFO , 15-551 |
| \$AC_FCT3LL , 15-557 | \$AC_TC_MTN , 15-551 |
| \$AC_FCT3UL , 15-557 | \$AC_TC_MTO , 15-551 |
| \$AC_FCTLL , 15-557 | \$AC_TC_STATUS , 15-551 |
| \$AC_FCTUL , 15-557 | \$AC_TC_THNO , 15-551 |
| \$AC_FIFO1 , 15-547 | \$AC_TC_TNO , 15-551 |
| \$AC_FIFO10 , 15-548 | \$AC_TIME , 15-552 |
| \$AC_FIFO2 , 15-547 | \$AC_TIMEC , 15-552 |
| \$AC_IPO_BUF , 15-543 | \$AC_TIMER , 15-552 |
| \$AC_IW_STAT , 15-543 | \$AC_TOTAL_PARTS , 15-558 |
| \$AC_IW_TU , 15-544 | \$AC_TRAFO , 15-544 |
| \$AC_LIFTFAST , 15-544 | \$AC_VACTB , 15-554 |
| \$AC_MARKER , 15-546 | \$AC_VACTW , 15-554 |
| \$AC_MEA , 15-544 | \$AC_VC , 15-554 |
| \$AC_MONMIN , 15-540 | \$AN_AXCTAS , 15-571 |
| \$AC_MSNUM , 15-555 | \$AN_AXCTSWA , 15-571 |
| \$AC_OPERATING_TIME , 15-558 | \$AN_CEC , 15-536 |
| \$AC_OVR , 15-554 | \$AN_CEC_DIRECTION , 15-537 |
| \$AC_PARAM , 15-546 | \$AN_CEC_INPUT_AXIS , 15-536 |
| \$AC_PATHN , 15-553 | \$AN_CEC_IS_MODULO , 15-537 |
| \$AC_PLTBB , 15-553 | \$AN_CEC_MAX , 15-537 |
| \$AC_PLTEB , 15-553 | \$AN_CEC_MIN , 15-537 |
| \$AC_PRESET , 15-561 | \$AN_CEC_MULT_BY_TABLE , 15-537 |
| \$AC_PROG , 15-543 | \$AN_CEC_OUTPUT_AXIS , 15-536 |
| \$AC_PRTIME_A , 15-552 | \$AN_CEC_STEP , 15-536 |
| \$AC_PRTIME_A_INC , 15-552 | \$AN_ESR_TRIGGER , 15-558 |
| \$AC_PRTIME_M , 15-552 | \$AN_POWERON_TIME , 15-538 |
| \$AC_PRTIME_M_INC , 15-552 | \$AN_SETUP_TIME , 15-538 |

\$MC_COMPRESS_VELO_TOL, 9-328**\$P_ACTBFRAME**, 15-539**\$P_ACTFRAME**, 15-539**\$P_ACTGEOAX**, 15-541**\$P_ACTID**, 15-543**\$P_AD**, 15-539**\$P_AEP**, 15-558**\$P_APDV**, 15-553**\$P_APR**, 15-558**\$P_ATPG**, 15-540**\$P_AXN1**, 15-541**\$P_AXN2**, 15-541**\$P_AXN3**, 15-541**\$P_BFRAME**, 15-539**\$P_CHBFR**, 15-509**\$P_CHBFRAME**, 15-539**\$P_CHBFRMASK**, 15-539**\$P_CTABDEF**, 15-542**\$P_D**, 15-540**\$P_DRYRUN**, 15-542**\$P_EG_BC**, 15-567**\$P_EP**, 15-558**\$P_EXTGG**, 15-541**\$P_F**, 15-554**\$P_FA**, 15-563**\$P_GG**, 15-541**\$P_GWPS**, 15-555**\$P_H**, 15-540**\$P_IFRAME**, 15-539**\$P_ISTEST**, 15-546**\$P_MC**, 15-542**\$P_MMCA**, 15-546**\$P_MSNUM**, 15-555**\$P_NCBFR**, 15-509**\$P_NCBFRAME**, 15-539**\$P_NCBFRMASK**, 15-539**\$P_NUM_SPINDLES**, 15-555**\$P_OFFN**, 15-542**\$P_PATH**, 15-543**\$P_PFRAME**, 15-539**\$P_PROG**, 15-542**\$P_PROGPATH**, 15-542**\$P_REPINF**, 15-542**\$P_S**, 15-555**\$P_SAUTOGEAR**, 15-556**\$P_SDIR**, 15-555**\$P_SEARCH**, 15-541**\$P_SEARCH_POSMODE**, 15-556**\$P_SEARCH_S**, 15-555**\$P_SEARCH_SDIR**, 15-555**\$P_SEARCH_SGEAR**, 15-555**\$P_SEARCH1**, 15-541**\$P_SEARCH2**, 15-541**\$P_SEARCHL**, 15-541**\$P_SGEAR**, 15-556**\$P_SIM**, 15-542**\$P_SMODE**, 15-556**\$P_STACK**, 15-542**\$P_SUBPAR**, 15-542**\$P_TCANG**, 15-540**\$P_TOOL**, 15-539**\$P_TOOLEXIST**, 15-540**\$P_TOOLL**, 15-540**\$P_TOOLND**, 15-540**\$P_TOOLNO**, 15-539**\$P_TOOLR**, 15-540**\$P_UBFR**, 15-539**\$P_UIFR**, 15-509**\$P_UIFRNUM**, 15-539**\$P_VDITCP**, 15-540**\$PI**, 15-542**\$SA_LEAD_TYPE**, 9-321, 9-322**\$SC_PA_ACTIV_IMMED**, 15-513**\$SC_PA_CENT_ABS**, 15-514**\$SC_PA_CENT_ORD**, 15-514**\$SC_PA_CONT_ABS**, 15-514**\$SC_PA_CONT_NUM**, 15-513**\$SC_PA_CONT_ORD**, 15-514**\$SC_PA_CONT_TYP**, 15-514**\$SC_PA_LIM_3DIM**, 15-513**\$SC_PA_MINUS_LIM**, 15-513**\$SC_PA_ORI**, 15-513**\$SC_PA_PLUS_LIM**, 15-513**\$SC_PA_T_W**, 15-513

| | |
|-------------------------------------|---------------------------------------|
| \$SN_PA_ACTIV_IMMED , 15-537 | \$TC_DP15 , 15-516 |
| \$SN_PA_CENT_ABS , 15-538 | \$TC_DP16 , 15-516 |
| \$SN_PA_CENT_ORD , 15-538 | \$TC_DP17 , 15-517 |
| \$SN_PA_CONT_ABS , 15-538 | \$TC_DP18 , 15-517 |
| \$SN_PA_CONT_NUM , 15-538 | \$TC_DP19 , 15-517 |
| \$SN_PA_CONT_ORD , 15-538 | \$TC_DP2 , 15-514 |
| \$SN_PA_CONT_TYP , 15-538 | \$TC_DP20 , 15-517 |
| \$SN_PA_LIM_3DIM , 15-538 | \$TC_DP21 , 15-517 |
| \$SN_PA_MINUS_LIM , 15-538 | \$TC_DP22 , 15-517 |
| \$SN_PA_ORI , 15-537 | \$TC_DP23 , 15-518 |
| \$SN_PA_PLUS_LIM , 15-538 | \$TC_DP24 , 15-518 |
| \$SN_PA_T_W , 15-537 | \$TC_DP25 , 15-518 |
| \$TC_ADPT1 , 15-534 | \$TC_DP3 , 15-514 |
| \$TC_ADPT2 , 15-534 | \$TC_DP4 , 15-515 |
| \$TC_ADPT3 , 15-534 | \$TC_DP5 , 15-515 |
| \$TC_ADPTT , 15-534 | \$TC_DP6 , 15-515 |
| \$TC_CARR1 , 15-510 | \$TC_DP7 , 15-515 |
| \$TC_CARR1...14 , 8-296 | \$TC_DP8 , 15-515 |
| \$TC_CARR10 , 15-511 | \$TC_DP9 , 15-515 |
| \$TC_CARR11 , 15-511 | \$TC_DPC1 , 15-519 |
| \$TC_CARR12 , 15-511 | \$TC_DPC10 , 15-519 |
| \$TC_CARR13 , 15-511 | \$TC_DPC2 , 15-519 |
| \$TC_CARR14 , 15-511 | \$TC_DPCE , 15-518 |
| \$TC_CARR15 , 15-511 | \$TC_DPC_i , 15-519 |
| \$TC_CARR16 , 15-511 | \$TC_DPCS1 , 15-519 |
| \$TC_CARR17 , 15-511 | \$TC_DPCS10 , 15-520 |
| \$TC_CARR18 , 15-512 | \$TC_DPCS2 , 15-519 |
| \$TC_CARR18[m] , 8-296 | \$TC_DPCS_i , 15-520 |
| \$TC_CARR2 , 15-510 | \$TC_DPH , 15-518 |
| \$TC_CARR3 , 15-510 | \$TC_ECP13 , 15-523 |
| \$TC_CARR4 , 15-510 | \$TC_ECP14 , 15-523 |
| \$TC_CARR5 , 15-510 | \$TC_ECP21 , 15-523 |
| \$TC_CARR6 , 15-510 | \$TC_ECP23 , 15-523 |
| \$TC_CARR7 , 15-510 | \$TC_ECP24 , 15-523 |
| \$TC_CARR8 , 15-511 | \$TC_ECP31 , 15-523 |
| \$TC_CARR9 , 15-511 | \$TC_ECP33 , 15-524 |
| \$TC_DP1 , 15-514 | \$TC_ECP34 , 15-524 |
| \$TC_DP10 , 15-516 | \$TC_ECP41 , 15-524 |
| \$TC_DP11 , 15-516 | \$TC_ECP43 , 15-524 |
| \$TC_DP12 , 15-516 | \$TC_ECP44 , 15-524 |
| \$TC_DP13 , 15-516 | \$TC_ECP51 , 15-524 |
| \$TC_DP14 , 15-516 | \$TC_ECP53 , 15-524 |

| | |
|------------------------------|------------------------------|
| \$TC_ECP54 , 15-525 | \$TC_MPP2 , 15-530 |
| \$TC_ECP61 , 15-525 | \$TC_MPP3 , 15-530 |
| \$TC_ECP63 , 15-525 | \$TC_MPP4 , 15-530 |
| \$TC_ECP64 , 15-525 | \$TC_MPP5 , 15-530 |
| \$TC_ECP71 , 15-525 | \$TC_MPP6 , 15-530 |
| \$TC_MAMP1 , 15-534 | \$TC_MPP7 , 15-530 |
| \$TC_MAMP2 , 15-534 | \$TC_MPPC1 , 15-531 |
| \$TC_MAMP3 , 15-534 | \$TC_MPPC10 , 15-531 |
| \$TC_MAP1 , 15-532 | \$TC_MPPC2 , 15-531 |
| \$TC_MAP2 , 15-532 | \$TC_MPPCS1 , 15-531 |
| \$TC_MAP3 , 15-532 | \$TC_MPPCS10 , 15-531 |
| \$TC_MAP4 , 15-532 | \$TC_MPPCS2 , 15-531 |
| \$TC_MAP5 , 15-533 | \$TC_MPTH , 15-532 |
| \$TC_MAP6 , 15-533 | \$TC_SCP13 , 15-520 |
| \$TC_MAP7 , 15-533 | \$TC_SCP14 , 15-520 |
| \$TC_MAP8 , 15-533 | \$TC_SCP21 , 15-520 |
| \$TC_MAP9 , 15-533 | \$TC_SCP23 , 15-520 |
| \$TC_MAPC1 , 15-533 | \$TC_SCP24 , 15-520 |
| \$TC_MAPC10 , 15-533 | \$TC_SCP31 , 15-521 |
| \$TC_MAPC2 , 15-533 | \$TC_SCP33 , 15-521 |
| \$TC_MAPCS1 , 15-533 | \$TC_SCP34 , 15-521 |
| \$TC_MAPCS10 , 15-533 | \$TC_SCP41 , 15-521 |
| \$TC_MAPCS2 , 15-533 | \$TC_SCP43 , 15-521 |
| \$TC_MDP1 , 15-532 | \$TC_SCP44 , 15-521 |
| \$TC_MDP2 , 15-532 | \$TC_SCP51 , 15-521 |
| \$TC_MLSR , 15-532 | \$TC_SCP53 , 15-522 |
| \$TC_MOP1 , 15-526 | \$TC_SCP54 , 15-522 |
| \$TC_MOP11 , 15-526 | \$TC_SCP61 , 15-522 |
| \$TC_MOP13 , 15-526 | \$TC_SCP63 , 15-522 |
| \$TC_MOP15 , 15-526 | \$TC_SCP64 , 15-522 |
| \$TC_MOP2 , 15-526 | \$TC_SCP71 , 15-522 |
| \$TC_MOP3 , 15-526 | \$TC_TP1 , 15-527 |
| \$TC_MOP4 , 15-526 | \$TC_TP10 , 15-528 |
| \$TC_MOP5 , 15-526 | \$TC_TP11 , 15-528 |
| \$TC_MOP6 , 15-526 | \$TC_TP2 , 15-527 |
| \$TC_MOPC1 , 15-527 | \$TC_TP3 , 15-527 |
| \$TC_MOPC10 , 15-527 | \$TC_TP4 , 15-528 |
| \$TC_MOPC2 , 15-527 | \$TC_TP5 , 15-528 |
| \$TC_MOPCS1 , 15-527 | \$TC_TP6 , 15-528 |
| \$TC_MOPCS10 , 15-527 | \$TC_TP7 , 15-528 |
| \$TC_MOPCS2 , 15-527 | \$TC_TP8 , 15-528 |
| \$TC_MPP1 , 15-530 | \$TC_TP9 , 15-528 |

\$TC_TPC1, 15-528
\$TC_TPC10, 15-528
\$TC_TPC2, 15-528
\$TC_TPCS1, 15-528
\$TC_TPCS10, 15-529
\$TC_TPCS2, 15-529
\$TC_TPG1, 15-529
\$TC_TPG2, 15-529
\$TC_TPG3, 15-529
\$TC_TPG4, 15-529
\$TC_TPG5, 15-529
\$TC_TPG6, 15-529
\$TC_TPG7, 15-529
\$TC_TPG8, 15-529
\$TC_TPG9, 15-529
\$VA_COUP_OFFS, 15-569
\$VA_CURR, 15-565
\$VA_DPE, 15-569
\$VA_EG_SYNCDIFF, 15-567
\$VA_IM, 15-560
\$VA_IM1, 15-560
\$VA_IM2, 15-560
\$VA_IS, 15-569
\$VA_LOAD, 15-565
\$VA_POWER, 15-565
\$VA_PRESSURE_A, 15-566
\$VA_PRESSURE_B, 15-566
\$VA_TORQUE, 15-565
\$VA_VACTM, 15-565
\$VA_VALVELIFT, 15-565

A

Ações síncronas, 13-454
 estáticas, 9-323
Acoplamento, 13-431
Acoplamento através do valor atual, 13-431
Acoplamento através do valor desejado, 13-431
Agrupamento spline, 5-157
Alteração de Modo, 10-391
Ângulo euler, 8-287
Ângulo lateral, 7-224
Ângulo principal, 7-224

Ângulo referência, 13-437
Ângulo RPY, 8-287
Ângulos de rotação a1, a2, 8-296
Apagando acoplamentos, 13-438
Apagar caminho restante, 5-180
Aprendendo as características de compensação,
 13-429
Arredondamento, 5-170
ASUP, 10-393
Atrito, 13-429
Atual
 Deslocamento angular, 13-439
 Estado atual de acoplamento do fuso seguidor,
 13-439
Avanço
 Axial, 10-377
 movimento, 11-407
 Movimento, 11-409
Avanço axial, 10-377
Avanço de entrada
 Supressão, 11-404
Avanço de trajetória, 5-169
Avanço do eixo, 11-412
Avanço parcial, 11-404

B

Backup de tensão DC, 13-451
Bloco de parâmetros servo programável, 5-189
Blocos de rosqueamento, 5-171
Bloqueio da execução do programa, 10-360

C

Calcula a interseção de dois elementos de
 contorno, 14-464
Calcular dados circulares, 14-479
Cálculo do frame, 6-209
Câmbio eletrônico, 13-441
CANCEL, 10-394
Cancela percurso restante, 11-402
Cancelamento da distância a percorrer, 10-362
Cancelamento da distância a percorrer com
 preparação, 10-362

- Cancelamento de uma ação síncrona, 10-390
- Capacidade de cálculo, 13-454
- Caracteres maiúsculos/minúsculos, 1-50
- Carga de array, lista de valores, 1-32
- Carga de valores, 1-38
- Carregando o valor atual, 10-379
- Chamada de subprograma
 - Indireta, 1-37
- Chamada de subprogramas
 - Pesquisa de diretório, 3-127
- Chamada do frame, 6-200
- Chamada indireta de subprograma, 1-37
- Checagem de erro, 14-465
- Checagem de erros, 14-472
- CHECKSUM, 1-88
- Cicles
 - Carregando parâmetros em ciclos de usuário, 2-113
- Ciclo de interpolação, 13-454
- Ciclo IPO, 11-410
- Ciclos tecnológicos, 10-386
- Cinemática resolvida, 8-296
- Cinemática tipo M, 8-296
- Cinemática tipo P, 8-296
- Cinemática tipo T, 8-296
- Código G, 5-169
 - Grupo, 5-171
- Coeficiente polinomial, 5-164
- Combinações de eixos acoplados, 9-308
- comparação e operações lógicas
 - Prioridade de operadores, 1-44
- Comparação e operadores lógicos, 1-41
- Compensação de raio da ferramenta, 3D
 - Comportamento nos cantos externos, 8-287
 - orientação da ferramenta, 8-287
 - Programando a orientação da ferramenta, 8-287
- Compressor, 5-160, 5-169
- Comprimento da string, 1-51
- Comprimento parcial, 11-403
- Comunicação NCU-NCU, 13-454
- Condições adicionais, 5-171, 10-391
- Condições suplementares, 1-61
- Condições suplementares com transformações, 7-252
- Conexão de comunicação, 13-454
- Contagem de peças, 13-460
- Controle adaptável, aditivo, 10-368
- Controle adaptável, multiplicativo, 10-369
- Controle de oscilações através de ações síncronas, 11-404
- Controle de potência laser, 10-366
- Controle de tolerância, 10-370
- Controle tangencial
 - Ângulo limite através da limitação da área de trabalho, 9-304
 - Definindo eixos de acompanhamento e principal, 9-303
- Controle tangencial, ativação, TANGON, 9-304
- Controle tangencial, desativação, 9-304
- Conversão de tipo, 1-47
- Coordenação de eixos, 10-378
- Coordenação de programa
 - Exemplo, 1-66
- Coordenação de programas
 - Instruções para coordenação de programas, 1-64
- Coordenação do programa, 1-63
- Correção de raio da ferramenta, 3D, 8-279
 - Cantos interiores/exteriores, 8-285
 - Fresagem circular, 8-282
- Correção de raio da ferramenta, fresagem circular, 3D, 8-281
- Correção do raio da ferramenta, 3D
 - profundidade de inserção (ISD), 8-285
- Corretor de ferramenta
 - fresa de face 3D, 8-282
- Corretores de ferramenta
 - Fresa de face, 8-279
 - Memória de corretores, 8-264
- Corretores de ferramenta online, 10-372
- Corte
 - Ponta (FS), 8-285
 - Ponto de referência (**FH**), 8-285
- Cortes de alívio, 14-465
- Coupling, 9-302, 9-307
- cov.com, ciclos de usuário, 2-114

Criar rotina de interrupção como um subprograma, 1-69
Critérios programáveis para fim de movimentação, 5-188
CS, 9-302
CTAB, 9-314
CTABDEF, 9-312
CTABEND, 9-312
CTABINV, 9-314
Curva de transformação de superfície cilíndrica, 7-246
CUT, 14-477

D

Dados de máquina e dados setting, 10-356
Definição de array, 1-30
Definição de polinômio, 10-364
Definição de variáveis, 1-25
Definindo dados de usuário, 3-131
Definindo e iniciando uma subrotina de interrupção, 1-70
degrees, 9-311
DELETE, 1-83
Denominador polinomial, 5-167
Desativando frames, 6-208
desativando transformações
TRAFOOF, 7-254
Desativando/reactivando rotinas de interrupção, 1-71
Desbaste, 14-464
Deslocamento de contorno normal OFFN, 7-244
deslocamento de origem
Deslocamento utilizando a manivela eletrônica, 6-205
Deslocamento de origem
Desativando transformações, 6-208
Deslocamento de origem externo, 6-206
PRESETON, 6-207
Deslocamento de origem externo, 6-206
Deslocamento DRF, 6-205
Deslocamento fino, 6-204
Deslocamento grosso, 6-204
deslocamento preset, 6-207

Diretório
Especificação absoluta, 1-64
Especificação relativa, 1-64
Diretório de peças, 3-125
Dwell time, 1-67

E

EG
Câmbio eletrônico, 13-441
Eixo
Conectado, 13-457
Local, 13-457
Eixo inclinado, TRAANG, 7-231, 7-249
Eixo principal, 9-319
Eixo seguidor, 9-319
Eixo/fuso de fixação, 13-457
Eixos
Depósito, 13-457
Eixos de comando, 10-374
Eixos de orientação, 7-229, 7-231
Eixos de trajetória, 5-169
Eixos geométricos comutáveis, 7-258
Eixos orientados, 7-223
Eixos rotativos
Vetores de direção V1, V2, 8-296
Vetores de distância I1, I2, 8-296
Elemento de contorno, 14-468, 14-470
Elementos de comando, 10-341
Elementos de contorno, interseção, 14-476
Elementos de corte de alívio, 14-466
Encadeamento de frames, 6-201
Encadeando seqüências de caracteres, 1-49
Endereços OEM, 5-187
Erro de compensação de quadrante
Aprendizado subsequente, 13-430
Ativando o processo de aprendizado, 13-430
Desativando o processo de aprendizagem, 13-430
Estado do acoplamento, 9-322
Estado do apalpador, 5-181
Estruturas de controle, 1-58
EXECTAB, 14-464

Execução de programa com memória de pré processamento, 9-330
Executando os movimentos de um elemento de contorno, 14-478
Executando um subprograma externo, 2-111
EXECUTE, 4-140
Exibição do bloco atual, 2-107
Exibindo blocos, 2-107
Expansão das funções de medição, 7-234
Expansão de parada e recuo, 13-447
EXTCALL, 2-111

F

F polinomial, 5-170
FAXis, 9-302, 9-307, 9-311, 9-319
FGROUP
 Eixos, 5-169
Fim de programa, 1-67, 10-392
Fixação da peça, 13-454
Flag variables, 10-353
Folga no fuso de esferas, 13-429
FOR, 1-59
Fresa de circunferência, 8-279
Fresa de face, 7-226
Função avaliação, 10-367
Funções adicionais de medição, 5-177
Funções aritméticas, 1-39
Funções auxiliares, 12-420
Funções auxiliares, 10-359
Funções de eixo, 13-428
Funções M, 12-419
 Três dígitos, 2-119
Funções M/G com três dígitos, 2-119
Funções OEM, 5-187
Fuso sincronizado
 Variáveis de sistema, 1-23
Fuso síncrono, 13-431
 Apagando acoplamento, 13-438
 Comportamento na troca de bloco, 13-436
Fusos síncronos
 Ativar modo síncrono, 13-437
 Definição do par, 13-433
 Desativando o modo síncrono, 13-437

Par, 13-432
Relação de velocidade, 13-435
Tipo de acoplamento, 13-436

G

gerenciamento de ferramentas, 8-266
Gravando a medição, 5-176
GUD
 Ativação automática, 3-137

I

Indicador Max/min, 14-468, 14-470
Índice do array, 1-31
Inicia/para movimento, 10-376
Iniciando pancadas, 12-418
Instrução CASE, 1-56
Instruções, 10-343
Instruções Jump
 Instrução CASE, 1-56
Interpolação circular, 5-171
Interpolação linear, 5-169, 5-171
Interpolação polinomial, 5-163, 5-169
 denominador polinomial, 5-167
Interpolação spline, 5-169
 Compressor, 5-157
 Spline A, 5-152
 Spline B, 5-153
 Spline C, 5-154
Interpolação Spline, 5-151
Interpolações OEM, 5-187
ISD (Profundidade de inserção), 8-279
ISFILE, 1-87

L

LAXis, 9-302, 9-307, 9-311, 9-319
Liga puncionar, 12-416
Liga Riscar, 12-416
Ligação do comando, 10-391
Limite de software, 10-377

M**MAC**

- Ativação automática, 3-137

MACH, 14-465

Macros, 2-118

Máquina

- Estado global, 13-454

Marcas de espera, 10-384

Marcas WAIT, 10-384

MEAFRAME, 6-209, 6-212

Medição, 10-384

Medições com apalpadores

- Programando blocos de medição, 5-175

- Variáveis de estado, 5-175

Memória

- Estrutura de memória, 3-122

- Memória de programa, 3-122

- Memória do usuário, 3-122

Memória de pré processamento, 9-330

Memória de programa, 3-122

- Generalidades, 3-123

- Pesquisa de diretórios na chamada do subprograma, 3-127

memória de programas

- Diretório de peças, 3-125

Memória de programas

- Criando diretórios de peças, 3-126

- Diretório de peças, 3-125

- Diretórios, 3-124

- Selecionando peças, 3-127

- Tipos de arquivos, 3-124

Memória de usuário, 3-128

- Área de dados, 3-128

- Nomes reservados, 3-131

- Programas de inicialização, 3-128

Mesa giratória de ferramenta I₄, 8-296

Modo, 11-403

Monitoração de ferramentas 0 específica para retíficas, 8-271

Movimento acoplado, 9-307

- Fator de acoplamento, 9-309

- Movimento acoplado de eixos, 9-308

Movimento de eixos acoplados, 10-381

Movimento de um único eixo, 12-422

Movimentos com ações síncronas

- Ações, 10-346

- Programação, 10-339

- Relação, 10-348

Movimentos com o fuso, 10-380

Movimentos de posicionamento, 10-374

N

n, 9-311

NC Stop, 10-392

NCU

- Conexão, 13-454

NCUs em rede, 13-454

NEWCONF, 1-80

Nível de encadeamento, 1-60

Número de identificação, 10-342

Número do incerto, 8-291

Números D

- Checar, 8-292

- determinar o número Tr, 8-294

- Renomear, 8-293

- Uso livre, 8-291

O

Operação como gerador, 13-451

Operações com string, 1-46

Operações/funções aritméticas, 1-39

Operadores lógicos, 1-42

orientação da ferramenta

- Com LEAD e TILT, 7-227

Orientação da ferramenta, 7-223, 8-287

Oscilação

- Ativando, desativando, 11-399

- Definindo a sequência de movimentos, 11-400

- Oscilação assíncrona, 11-398

- Oscilação síncrona, 11-403

Oscilação assíncrona, 11-396

Oscilação síncrona

- definindo eixos de avanço e oscilação, 11-405

Oscilações

- Oscilação assíncrona, 11-396

Oscilações síncronas
 Ação síncrona, 11-406
 avanço na área reversa, 11-407
 Definição do avanço, 11-405
Oscilações sincronizadas
 Parada no ponto de reversão, 11-409
Override, 11-410

P

P_SEARCH_POS, 15-556
Para o eixo de oscilação, 11-397
Parada de pré processamento, 10-361
Parada e recuo
 Expansão, 13-447
Parada independente do acionamento, 13-451
Parada para faiscamento, 11-402
Parâmetro de curva, 5-169
Parâmetros aritméticos, 1-22
Parâmetros em ações síncronas, 10-354
Parâmetros R, 10-355
Parâmetros R (lista), 15-509
Pontos reversos de oscilação, 11-397
Porta ferramenta, 8-297
 Apagar/editar/ler dados, 8-298
 Cinemática, 8-296
Posicionando eixo, 10-376
Posicionando em pontos codificados, 5-150
Posições singulares, 7-230
Power On, 10-391
Preparação de contorno, 14-465
Preparação do contorno, 14-472
 Elementos de corte de alívio, 14-466
Procedimento de interseção para compensação
 3D, 8-286
Procura por caracter, 1-51
Programa de inicialização, 3-128
 Definição dos dados de usuário, 3-131
 Definindo dados de usuário, 3-131
 Gerando um programa de inicialização, 3-129
 Gravando, 3-129
Programa de usinagem, 13-454
Programa NC, 13-457
Programa sem fim, 1-62

Programação indireta, 1-36
Programas de inicialização
 Carregando programas de inicialização, 3-129
Protection levels for user data, 3-135
Puncionamento, 12-420
Puncionar, 12-416

Q

Quantidade de peças, fixada, 1-62

R

R, 15-509
Reações independentes do acionamento, 13-448
READ, 1-84
Recipiente de eixos, 13-457, 13-459
Recuo independente do acionamento, 13-452
Referência de trajetória ajustável, 5-169
Relação de velocidade, 13-435
REPEAT, 1-60
Repetir subprograma, 2-103
Reposicionamento, 10-394
 reposicionamento no contorno
 Aproximação com nova ferramenta, 9-334
 Reposicionamento no contorno, 9-332
 Ponto de reposicionamento, 9-333
 reposicionando no contorno
 Aproximação em linha reta, 9-335
 Reposicionando no contorno
 Aproximação em semicírculo, 9-336
 em ¼ de circunferência, 9-335
Reset, 10-391
Resposta a erros, 10-385
reversão
 Ponto, 11-404
Reversão
 Área, 11-404
 Ponto, 11-404
Riscar, 12-416
Rodando subprograma com especificação de
 parâmetros, 2-106
Rotina de interrupção, 1-68
 Direção programável de movimento, 1-68

- Recuo rápido do contorno, 1-72
- Salvar a posição de interrupção, 1-69
- Rotinas de interrupção
 - Definindo a prioridade, 1-70

S

- SBLON, 2-108
- Seções de trajetória, 12-420
- Segmentação automática de trajetória, 12-420
- Segmentação da trajetória para eixos de trajetória, 12-421
- Segmentação de trajetória para um único eixo, 12-422
- Seleção de uma substring, 1-53
- Selecione um único caractere, 1-54
- Setting data, 11-398
- Simulação do valor principal, 9-321
- Sincronismo de posição, 13-432
- Sincronização
 - Sincronização de valor desejado, 13-431
- Sincronização fina (fine), 13-431
- Sincronização grossa (coarse), 13-431
- Subprograma com especificação de diretório e parâmetros, 2-106
- Subprograma, externo, 2-111
- Subprogramas, 2-92
 - Chamada de subprogramas, 2-99
 - Chamada indireta de subprograma, 2-105
 - Chamada modal de subprograma, 2-104
 - Encadeamento, 2-93
 - Mecanismo SAVE, 2-94
 - Repetição de programa, 2-103
 - Subprograma com transferência de parâmetros, 2-99
- Subprogramas com transferência de parâmetros
 - definição de arrays, 2-98
 - Transferência de parâmetros entre um programa e um subprograma, 2-95
- Suprimir bloco a bloco, 2-108

T

- Tabela de contorno, 14-465, 14-472

- Tabelas de curvas, 9-311
- TANG, 9-303
- Tecnologia das macros, 12-419
- Tempo de espera, 1-67
- Tempo de execução do programa, 13-459
- Tempo de permanência, 11-399
- Tempo de processamento, 1-60
- tipo de cinemática, 8-298
- Tipo de cinemática, 8-298
- Tipo de variável, 1-27
- Tipos de cinemática, 8-296
- Torção, 13-429
- Torneamento facial
 - Usinagem externa, 14-465
 - Usinagem interna, 14-465
- Torneamento longitudinal
 - Usinagem externa, 14-465
 - Usinagem interna, 14-465
- TRAFOOF, 7-254
- Transferência de eixo
 - buscar eixo (GET), 1-77
 - GET, 1-76
 - RELEASE, 1-76
- Transferência de fuso
 - GET, 1-76
 - RELEASE, 1-76
- Transformação com rotação de eixos lineares, 7-221
- Transformação de eixo inclinado, 7-248
- Transformação de superfície cilíndrica, 7-242
- Transformação de superfície cilíndrica
 - deslocamento de contorno normal OFFN, 7-244
- Transformação TRACYL, 7-242
- Transformação TRANSMIT, 7-239
- Transformação TRAORI, 7-222
- Transformação, cinco eixos
 - programação do vetor de direção, 7-225
 - programação em ângulos Euler, 7-224
 - programação em ângulos RPY, 7-224
- Transformação, cinco eixos, fresa de face, 7-226
- Transformação, ferramentas orientadas em cinco eixos, 7-223
- Transformação, programação de cinco eixos através de LEADITILT, 7-223

Transformações, 3/4-eixos, 7-222
TRAORI, 7-220
Trecho de trajetória, 12-420
Troca de eixo
 Libera eixo, 1-77
Troca de estação/posição, 13-457

U

Usuário, ciclos, 2-115

V

Valor atual e desejado para o acoplamento, 9-320
Valor de acoplamento axial principal, 9-319
Valor principal de acoplamento, 10-382
Variáveis, 1-22
 Aritméticas, 1-23
 Carga de valores, 1-38
 De sistema, 1-23
 de usuário, 1-22, 1-25
 Definição de arrays (matrizes), 1-30
 Globais NCK, 1-67
 Programação indireta, 1-36
 Tipos de conversão, 1-45
 Tipos de variáveis, 1-22, 1-23
Variáveis de conexão globais, 13-454
Variáveis de sistema, 1-22
 Globais, 13-454
Variáveis em tempo real, 10-350

variáveis frame
 Definição de novos frames, 6-203
Variáveis frame, 6-192
 Carga de valores, 6-197
 Leitura ou alteração de componentes frame, 6-199
 Variáveis frame pré definidas, 6-193
Variáveis Frame
 Chamada coordenada de transformação, 6-192
Variável FIFO, 10-357
Variável temporizador, 10-353

W

WCS, 3-125
WHEN-DO, 11-406
WHILE, 1-59
WPD, 3-125
WRITE, 1-81

Z

Zonas de proteção, 4-139
Zonas de proteção
 Ativando, desativando, 4-144
 Define a zona de proteção da máquina, 4-140
 Define zona de proteção do canal, 4-140
 Definição de contorno das zonas de proteção, 4-142
 Definindo zonas de proteção, 4-141

B Comandos, Identificadores

-, 1-39

*

*, 1-39

/

/, 1-39

:

:, 1-39

+

+, 1-39

<

<, 1-41

<<, 1-41

<=, 1-41

<>, 1-41

=

==, 1-41

>

>, 1-41

>=, 1-41

A

A, 7-248

A1, A2, 8-296

A2, 7-224

A3, 7-224

A4, 7-224

A5, 7-224

ABS, 1-39

ACC, 13-434

Acoplamento

AV, 13-431

DV, 13-431

ACOS, 1-39

ACTFRAME, 6-194

ALF, 1-68

AND, 1-42

ANG, 15-485

Ângulo RPY, 8-287

ANZHINT, 14-467, 14-469

applim, 9-311

APR, 3-135

aproxLW, 9-311

APW, 3-135

AS, 2-119

ASIN, 1-39

ASPLINE, 5-151

ATAN2, 1-39

AV, 13-436

AX, 13-428

AXCTSWE, 13-457

AXIS, 1-27

AXNAME, 1-48, 13-428

AXSTRING, 1-48

B

B_AND, 1-43

B_NOT, 1-43

B_OR, 1-43

B_XOR, 1-43

B2, 7-224

B3, 7-224

B4, 7-224

B5, 7-224

BAUTO, 5-155

BFRAME, 6-193

BNAT, 5-155

BOOL, 1-27

BRISK, 11-397

BSPLINE, 5-151

BTAN, 5-155

C

C2, 7-224

C3, 7-224

C4, 7-224

C5, 7-224

CAC, 5-150

CACN, 5-150

CACP, 5-150

CALCDAT, 14-464, 14-479

CALL, 2-105

CANCEL, 10-340

Carga de arrays, lista de valores, 1-32

CASE, 1-56

CDC, 5-150

CFINE, 6-204

Chamada de subprograma com diretório, 2-106

CHANDATA, 3-130

CHAR, 1-27

CHKDNO, 8-292

CIC, 5-150

CLEARM, 1-65

CLRINT, 1-68

CMIRROR, 6-197

COARSE, 13-431, 13-435, 13-436

COARSEA, 5-188

COMPLETE, 3-128, 3-129

COMPOF, 5-161, 5-169

COMPON, 5-161, 5-169, 9-328

CONTDCON, 14-472

CONTPRON, 14-464, 14-465, 14-476, 14-478

COS, 1-39

COUPDEF, 13-431, 13-433, 13-435

COUPDEL, 13-431, 13-433, 13-438

COUPOF, 13-431, 13-437, 13-438

COUPON, 13-431, 13-437, 13-438

COUPRES, 13-431, 13-438

CP, 7-234

CPROT, 4-144

CPROTDEF, 4-140, 4-142

CROT, 6-197

CSCALE, 6-197

CSPLINE, 5-151

CTAB, 9-311

CTABDEF, 9-311

CTABDEL, 9-311

CTABEND, 9-311

CTABINV, 9-311

CTRANS, 6-197

CUT3DC, 8-279

CUT3DF, 8-279

CUT3DFF, 8-279

CUT3DFS, 8-279

CUTCONOF, 8-276

CUTCONON, 8-276

D

DEF, 1-27

DEFAULT, 1-56

DEFINE, 2-119

DELDTG, 5-185

DELT, 8-266

DISABLE, 1-68

DISPLOF, 2-107

DISPR, 9-332

DIV, 1-39

DO, 10-340, 11-403

DRFOF, 6-208

DUPLO_NR, 8-266

DV, 13-436

DZERO, 8-295

E

EAUTO, 5-155

ELSE, 1-58

ENABLE, 1-68

ENAT, 5-155

ENDFOR, 1-58

ENDIF, 1-58

ENDLOOP, 1-58

Endpos, 11-403

ENDPROC, 10-370

ENDWHILE, 1-58

ERG, 14-479

ERROR, 14-465, 14-472
ETAN, 5-155
EVERY, 10-340
EXECTAB, 14-478
EXECUTE, 4-140, 4-142, 14-465, 14-472
EXP, 1-39
EXTCALL, 2-111
EXTERN, 2-99

F

FA, 11-400, 13-434
FALSE, 1-23
FCTDEF, 8-269
FCUB, 9-325
FINE, 13-431, 13-436
FINEA, 5-188
FLIN, 9-325
FMA, 15-492
FNORM, 9-325
FOR, 1-58
FPO, 9-325
FRAME, 1-27
FRC, 15-493
FRCM, 15-493
FROM, 10-340
FS, 13-431
FTOC, 8-269
FTOCOF, 8-269
FTOCON, 8-269
FW, 9-311

G

G1, 11-397
G153, 6-208
G25,G26, 9-304
G4, 11-399
G642, 5-171
GEOAX, 7-258
GET, 1-76
GETACTTD, 8-294
GETD, 1-76
GETDNO, 8-293

GETSELT, 8-266
GETT, 8-266
GOTOB, 1-56
GOTO, 1-56
GUD, 3-124, 3-128, 3-133, 3-135

I

I1,I2, 8-296
ID, 10-339
IDS, 10-339
IF, 1-58
IF-ELSE-ENDIF, 1-58
IFRAME, 6-194
I11,I12, 11-404
INDEX, 1-51
INIT, 1-64
INITIAL, 3-129
INT, 1-27
INTERSEC, 14-464, 14-476
IPOENDA, 5-188
IPOSTOP, 13-431, 13-434, 13-436
ISAXIS, 13-428
ISD, 8-279, 8-285
ISNUMBER, 1-48

K

KTAB, 14-469, 14-475, 14-478

L

LEAD, 7-224, 8-287
LEADOF, 9-319
LEADON, 9-319
LIFTFAST, 1-68
LN, 1-39
LOCK, 10-340
LOOP, 1-58
LOOP-ENDLOOP, 1-59
LS, 13-431
LW, 9-311

M

M17, 2-95
MATCH, 1-51
MCALL, 2-104
MEAC, 5-177, 5-185
MEAFRAME, 6-210
MEAS, 5-174
MEASA, 5-177
MEAW, 5-174
MI, 6-199
MIRROR, 6-194
MOD, 1-39
MOV, 10-376
MPF, 3-124
MU, 7-250
MZ, 7-250

N

NEWT, 8-266
NN, 14-465
NO., 14-479
NOC, 13-431, 13-436
NOT, 1-42
NPROT, 4-144
NPROTDEF, 4-140, 4-142
NUMBER, 1-48

O

OEMIPO1/2, 5-187
OF, 1-57
OFFN, 7-241, 7-242
OR, 1-42
ORIC, 8-287
ORID, 8-287
ORIMCS, 7-229, 7-231, 8-287
ORIS, 8-287
ORIWCS, 7-229, 7-231, 8-287
OS, 11-396, 11-399
OSC, 8-287
OSCILL, 11-403, 11-405
OSCTRL, 11-396, 11-400
OSE, 11-396, 11-400

OSNSC, 11-396, 11-403
OSO2, 11-396
OSOF, 8-287
OSP, 11-397
OSP1, 11-396, 11-403
OSP2, 11-403
OSS, 8-287
OSSE, 8-287
OST, 11-399
OST1, 11-396, 11-403
OST2, 11-396, 11-403
OVRA, 13-434

P

PDELAYOF, 12-416
PDELAYON, 12-416
PFRAME, 6-194
PKT, 14-479
PL, 5-153, 5-165
PO, 5-165
POLINOMIO, 14-473
POLY, 5-165
POLYNOMIAL, 14-466
PON, 12-416, 12-422
PONS, 12-416
Porta ferramenta, 8-297
POS, 13-437
POSP, 11-403
POT, 1-39
PRESETON, 6-207, 6-210
PRIO, 1-68
PROC, 2-95
PUTFTOC, 8-269
PUTFTOCF, 8-269
PW, 5-153

Q

QEC, 13-429
QECDAT.MPF, 13-430
QECLR.N.SPF, 13-430
QECLRNOF, 13-429
QECLRNON, 13-429

QECTEST.MPF, 13-430

R

RDISABLE, 10-360

REAL, 1-27

RELEASE, 1-76

REP, 1-34

REPEAT, 1-58

REPOS, 1-68, 1-75

REPOSA, 9-332

REPOSH, 9-332

REPOSHA, 9-332

REPOSL, 1-75, 9-332

REPOSQ, 9-332

REPOSQA, 9-332

RET, 2-95

RINDEX, 1-51

Riscando, 12-420

RMB, 9-332

RME, 9-332

RMI, 9-332

ROSCA, 14-473

ROUND, 1-39

RT, 6-199

S

S1,S2, 13-433, 13-438

SAVE, 1-69, 2-94

SBLON, 2-108

SC, 6-199

SCPARA, 5-189

SD, 5-153

SETDNO, 8-293

SETINT, 1-68

SETM, 1-65

SETPIECE, 8-266

SIN, 1-39

SOFT, 11-397

SON, 12-416, 12-421, 12-422

SONS, 12-416

SPI, 13-428, 13-434

SPLINE, 14-466, 14-473

SPLINEPATH, 5-157

SPN, 12-420

SPOF, 12-416

SPOS, 13-434

SPP, 12-420

SQRT, 1-39

START, 1-64

STARTFIFO, 9-330

STOPFIFO, 9-330

STOPRE, 5-174, 5-183, 9-330, 11-398

STOPREOF, 10-361

STRING, 1-27

STRINGFIELD, 1-46

STRINGVAR, 1-46

STRLEN, 1-51

SUBSTR, 1-53

SUPA, 6-208

Suprimir bloco a bloco, 2-108

SYNFCT, 10-367

SYNR, 3-133

SYNRW, 3-133

T

TABNAME, 14-465, 14-472, 14-476, 14-478

TAN, 1-39

TANG, 9-302

TANGOF, 9-302

TANGON, 9-302

TE, 5-177

THREAD, 14-466

TILT, 7-224, 8-287

TLIFT, 9-302

TOLOWER, 1-50

TOUPPER, 1-50

TR, 6-199

TRAANG, 7-242, 7-248

TRACYL, 7-239, 7-242

TRAFOOF, 7-220, 7-239, 7-242, 7-248, 7-254

TRAILOF, 9-307

TRAILON, 9-307

TRANSMIT, 7-239

TRAORI, 7-222

TRUE, 1-23

TRUNC, 1-39

U

U1,U2, 11-404

UNLOCK, 10-340

UNTIL, 1-58, 1-60

V

V1,V2, 8-296

VAR, 2-97

VARIB, 14-476, 14-479

W

WAIT, 1-65

WAITC, 13-431, 13-434

WAITE, 1-65

WAITM, 1-64

WAITMC, 1-65

WAITP, 11-399

WALIMON, 9-304

WCS, 11-410

WHEN, 10-340

WHEN-DO, 11-403

WHENEVER, 10-340

WHENEVER-DO, 11-403, 11-406

WHILE, 1-58

WZ, 8-266

X

x, 8-266

XOR, 1-42

Notes