INTRODUÇÃO AO CONTROLE DIGITAL

Alcindo do Prado Junior

Julho/89

Alcindo do Prado Junior

Professor de Sistemas de Controle
Departamento de Engenharia Elètrica
Faculdade de Engenharia de Joinville

"...pois nele (Jesus Cristo) foram criadas todas as cousas, nos cèus e sobre a Terra, as visìveis e as invisìveis, sejam tronos, sejam soberanias, quer principados, quer potestades. Tudo foi criado por meio dele e para Ele." (Col.1:16)

ÍNDICE

CAPÍTULO 5 - CONTROLADORES DIGITAIS BASEADOS EM CONTROLADORES ANALÓGICOS

CAPÍTULO 6 - PROJETO DE CONTROLADORES DIGITAIS NOL PLANO Z

CAPÌTULO 1 I N T R O D U Ç Ã O

Cresce a cada dia a importância dos dispositivos produzidos pela indústria micro-eletrônica, em particular dos tão comentados microprocessadores. É impossível ignorar a participação crescente da informática em praticamente todos os ramos da atividade humana; o microprocessador, por seu baixo custo, permitui a disseminação intensa das aplicações do computador digital. A área de Controle de Processos, que ja por tradição è das que incorporam com maior rapidez as inovações tecnològicas, teve então possibilidade de aplicar a nível industrial as tècnicas digitais mais modernas.

1.1. BREVE HISTORICO

Atè o fim da dècada dos anos 50 os Controladores de Processo eram construidos exclusivamente por elementos analògicos. Nessa època começaram a surgir os primeiros estudos de viabilidade de aplicação do computador digital na tarefa de Controle, como o realizado pelas companhias Texaco e TWE para uma unidade de polimerização, com o computador valvulado RW-300, que controlava 26 fluxos, 72 temperaturas, 3 pressões e 3 composições. Nessa fase, grandes problemas limitavam as aplicações: custo muito elevado, grande tamanho, grande dissipação de potência, pequena velocidade, pequena capacidade de memòria, e especialmente, pouca confiabilidade.

Com o passar do tempo essa situação foi gradativamente se revertendo. Ja no final dos anos 60 surgiram os chamados mini-computadores, que operavam em 16 bits, de tamanho reduzido e com apreciàvel confiabilidade. Os minicomputadores deram um novo impulso ao desenvolvimento do Controle Digital, embora fosse um sistema relativamente grande e ainda muito caro.

Em meados dos anos 70, com o surgimento dos micro-processadores integrados, o preço dos computadores sofreu uma enorme redução, permitindo então que o Controle Digital fosse uma alternativa viàvel, mesmo para pequenas aplicações: microprocessadores estão substituindo componentes analògicos mesmo para controladores de uma ûnica malha.

1.2. APLICAÇÃO DE COMPUTADORES EM CONTROLE DE PROCESSOS

Computadores Digitais tem sido usados de diversas formas na area de Controle de Processos. Nos primòrdios do Controle Digital, devido à pouquíssima confiabilidade, o computador exercia um papel meramente supervisòrio, ou através de simples monitoração, ou através de Controle de Sinal de Referência (Set-Point Control-SPC).

a) MONITORAÇÃO

Como ilustrado pela figura 1.1. abaixo, neste modo de utilização, o Computador Digital limita-se a fazer aquisição de dados, coletando informações das variàveis do processo, emitindo relatòrios, e podendo indicar a presença de alguma situação anômala (função de alarme).

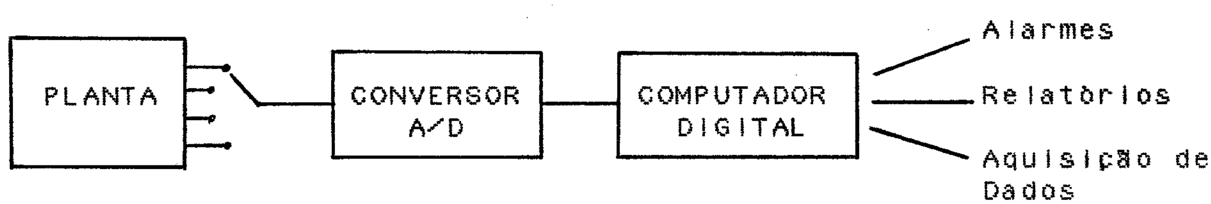


Figura 1.1 - Computador Digital usado em Monitoração

b) CONTROLE DE SINAL DE REFERENCIA (Set-Point Control-SPC)

Neste modo de operação os pontos de referência (set-points) de controladores analògicos convencionais são ajustados pelo computador, como indicado pela Figura 1.2. Note que a tarefa de controle è feita analògicamente, devendo os "set-points" passarem para alguma posição fixa padrão quando ocorrer pane no computador. Observe também a presença dos Conversores Analògico-Digital e Digital-Analògico, que interfaceiam o computador com os processos externos.

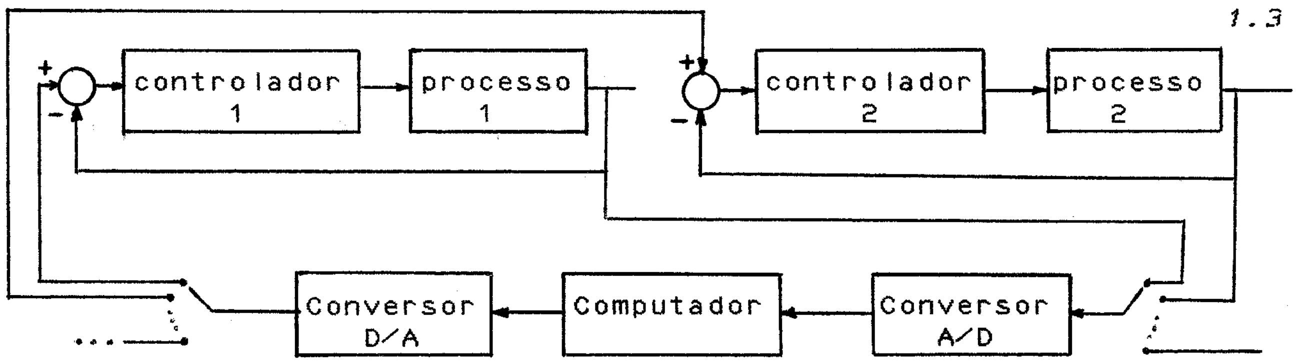


Figura 1.2 - Controle de Sinal de Referência

O ajuste dos pontos de referência è comumente feito por algum algoritmo computacional que visa a otimização de algum funcional relacionado com a qualidade do produto, e/ou com economia de custos.

c) **CONTROLE DIGITAL DIRETO** (Direct Digital Control - DDC)

Esta abordagem de Controle começou em 1962 nas instalações da ICI, uma indústria química inglesa, onde se usou um computador Ferranti Argus para controlar 129 vàlvulas e medir 224 variàveis. O nome Controle Digital Direto significa que o computador controla os processos diretamente, não apenas coordenando as ações de controladores analògicos locais, como no Controle de Sinal de Referência. Como mostrado pela Figura 1.3, ocorre uma sensível redução do número de equipamentos usados.

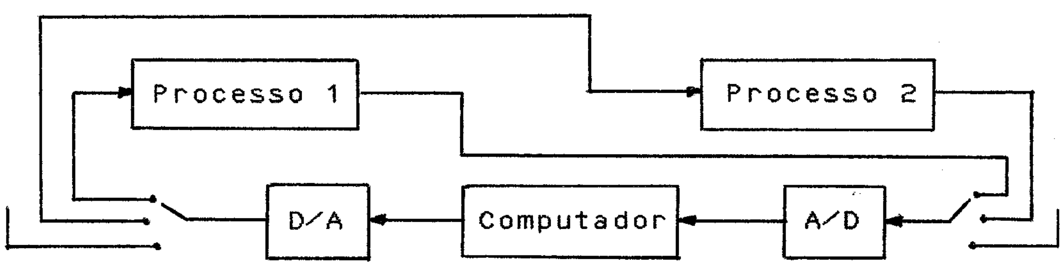


Figura 1.3 - Controle Digital Direto

A grande vantagem deste caso è que os algoritmos de controle podem ser alterados sem necessidade de qualquer mudança a nível de "hardware". Também, um simples monitor de vídeo pode substituir um grande painel de instrumentos de registro e medição. Essas vantagens, mais a grande facilidade de confecção de relatório a respeito do processo, tem tornado o Controle Digital Direto extremamente atrativo. A figura 1.4 esquematiza o Controle Digital Direto de uma única malha, que constitui o caso básico de estudo deste curso, mostrando os sinais de tempo contínuo e de tempo discreto correspondentes.

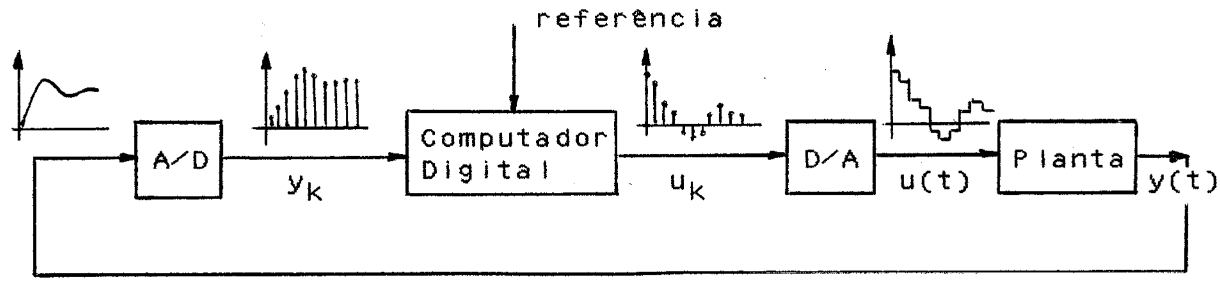


Figura 1.4 - Controle digital direto de uma única malha com os sinais temporais u(t) e y(t) e as sequências numéricas y_k e u_k.

Note que tanto a entrada do conversor analógico-digital como a saída do conversor digital-analógico constituem sinais de tempo contínuo, enquanto que a saída e a entrada correspondentes são sequências numéricas, manipuladas digitalmente pelo computador.

1.3. SISTEMAS DE INFORMAÇÃO E CONTROLE INTEGRADOS

Tais sistemas são redes interconectadas de sistemas de controle e de informação com computadores. A palavra controle è aqui considerada num contexto mais amplo, para representar todos os aspectos de decisão associados a operação de uma indústria, indo do controle de uma máquina, à gestão e gerência da empresa. Os subsistemas que compõe a rede compartilham dados e informação através de vias de comunicação de alta velocidade, executando tarefas de acordo com uma hierarquia funcional de diversos níveis, esquematizada na figura 1.5 abaixo.

As tarefas associadas ao nível mais baixo são definidos por algoritmos dedicados, implementados por programas que possibilitam a operação de, por exemplo, controladores programáveis (CLPs), comandos numéricos computadorizados, controladores digitais universais, etc. As tarefas a nível supervisório são mais complexas, frequentemente necessitando de minicomputadores, ou rede de microcomputadores. As tarefas de nível mais alto podem exigir computadores de maior parte.

NIVEL	CONTROLE	MISSÃO	VARIAVEIS	TEMPOS
Gerência	Controle da Lucratividade	Gerência da Empresa	Economia Mercado	Anos e Meses
Planejamento	Controle da Produtividade	Gerência das Areas	Estoques Pessoal	Meses Semanas
Supervisão	Controle Físico	Monitorização e Otimização dos Processos	Qualidade do Produto Custo/Peça	Dias Horas
Controle Digital Direto		Operação dos Processos	Pressão Temperatura Nível	Min. Seg.

Figura 1.5 - Níveis hierarquicos num Sistema de Informação e Controle Integrado.

Assim idealizado, o Sistema de Informação e Controle Integrado ainda esta longe de ser uma realidade em nosso país, que tem atingido, quando muito, o nível de supervisão. Neste curso nos deteremos no nível mais baixo, o de Controle Digital Direto, o que mostra que ainda restará um longo caminho a ser trilhado se desejarmos efetivamente dominar essa área do conhecimento.

Ligados aos conceitos hieràrquicos acima surgiram na Literatura de Controle Automàtico alguns termos que rapidamente estão se firmando:

CIM - Computer Integrated Manufacturing - Fabricação Integrada por Computador - do qual fazem parte todas as àreas ligadas à produção: projeto, planejamento, fabricaçao, desenvolvimento e controle de qualidade. O CIM divide-se basicamente em duas partes: o CAE e o CAM.

CAE - Computer Aided Engineering - Engenharia Auxiliada por Computador - que tem duas funções bàsicas: desenvolvimento e projeto (CAD) e o planejamento da Produção (CAP).

CAD - Computer Aided Design - Projeto Auxiliado por Computador-mòdulo destinado ao desenvolvimento e projeto de um produto, atravès de desenhos, via monitor ou plotador gràfico. Os dados tècnicos de cada peça são armazenados em um banco de dados, podendo ser utilizados para elaboração de novos projetos.

CAP - Computer Aided Planning - Planejamento Auxiliado por Computador - de onde resulta toda a documentação de dados necessàrios para a produção de um produto, tais como planos de trabalho, montagens, testes, etc. Nessa etapa são levados todos os dados necessàrios para o planejamento do Controle de Qualidade (CAQ).

CAQ - Computer Aided-Quality - Controle de Qualidade Auxiliado por Computador.

CAM - Computer Aided Manufacturing - Fabricação Auxiliada por Computador - Diretamente relacionada com CLPs (Controladores Lògicos Programàveis) e màquinas CNC (Computer Numèrical Control), o CAM executa o planejamento detalhado da produção, como a programação diària e os turnos.

CAPITULO 2 <u>NOÇÕES DE MICROCOMPUTADORES</u>

Embora cada microcomputador tenha suas proprias caracteristicas, certas estruturas são comuns à maioria dos sistemas existentes. Um Computador Digital básico consiste genericamente das seguintes partes, como ilustrado pela figura 2.1 :

- Unidade de Processamento Central (CPU em inglês)
- Memória
- Dispositivos de Entrada e saida (I/O)

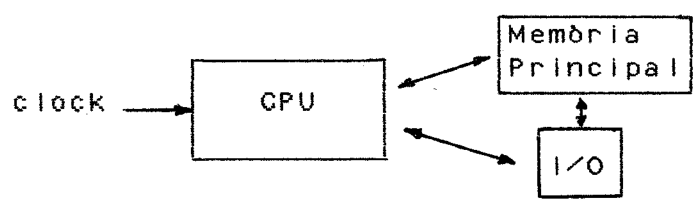


Fig. 2.1. - Partes Básicas de um Computador Digital

2.1. A UNIDADE DE PROCESSAMENTO CENTRAL (CPU)

A CPU é ao mesmo tempo coração e cérebro de um computador digital, contendo a Unidade Aritmética e Lógica (ALU), a Unidade de Controle, sincronizadora e geradora dos sinais necesssários para a execução das instruções , e Registradores de Trabalho.

Um <u>microprocessador</u> é essencialmente a CPU de um microcomputador, composta de um ou mais circuitos integrados LSI.

a) A UNIDADE ARITMETICA E LOGICA (ALU)

A manipulação entre operandos dentro da CPU é feita pela ALU, que executa operações como:

- adição binária
- operações lógicas booleanas
- deslocamento à direita e à esquerda em registradores
- testes de condições
- mudança de posição de dados

A figura 2.2 mostra o esquema de operação da ALU.

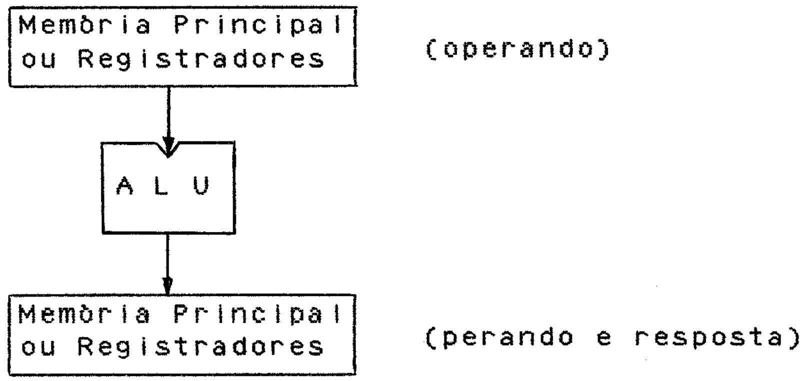


Fig. 2.2. - Operação da ALU

b) UNIDADE DE CONTROLE

A Unidade de Controle é aquela que reconhece o código de operação de cada instrução corrente e gera os sinais de controle necessários à sua execução. Estes são enviados sequencialmente aos registradores e à ALU.

A execução de qualquer instrução é dividida em duas fases: a busca da instrução na memória e sua decodificação (fetch), e a execução propriamente dita, com geração dos sinais de controle, incluindo aqueles necessários para a busca de operandos na memória. A figuira 2.3 ilustra o funcionamento da Unidade de Controle.

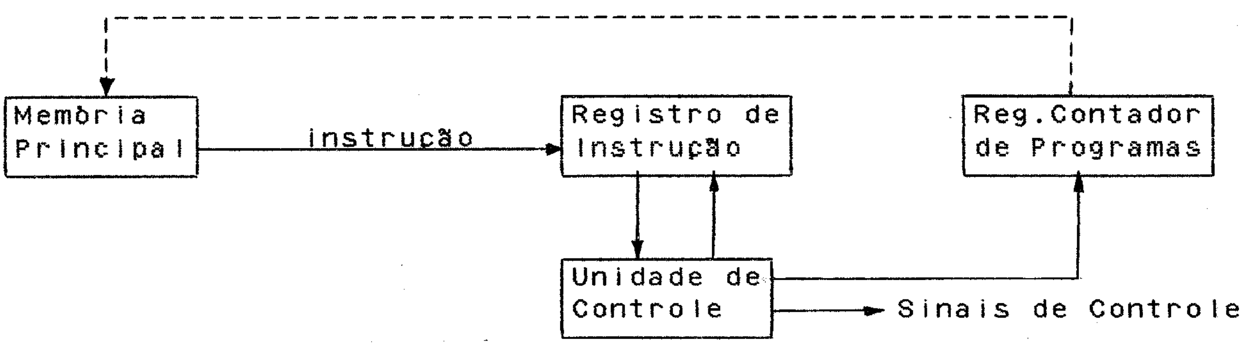


Fig.2.3. - Operação da Unidade de Controle

c) REGISTRADORES DE TRABALHO

Os registradores da CPU são utilizados para armazenamento temporário de dados, códigos de operação e endereços. Mais comumente tem-se os registradores:

- **Acumulador:** registrador onde são armazenados os resultados das operações aritméticas e lógicas efetuados pela ALU. E frequente nos microprocessadores que os dados de I/O passem pelo acumulador.

- **Contador de Programa(PC):** guarda os endereços de memória que contèm a próxima instrução a ser executada. Do número de bits deste registrador depende o número de memórias que podem ser endereçados diretamente. Um programa geralmente è endereçado sequencialmente; o PC è entáo incrementado de forma a percorrer todo o programa. Nas instruções de salto o PC è carregado com o endereço correspondente.

- **Registrador de Endereço da Memória (MAR):** contèm o endereço da memória onde um dado serà gravado, ou de onde um dado serà lido.

- **Registrador de Instruções (IR):** contèm a instrução retirada da memória e que serà decodificada e executada.

- **Registrador de Estado:** cada bit deste registrador serve para indicar ocorrências especiais, como "overflow" no acumulador, sinal do número do acumulador, etc.

- **Registradores Gerais:** para armazenamento temporário de dados e endereços de operandos.

- **Ponteiro da Pilha (Stack Pointer):** indica a posição do topo de um conjunto de memórias chamada de pilha (LIFO - Last In, First Out). A pilha pode ser implementada:

  i) completamenmte na memória principal
  ii) através de um conjunto de registradores especiais na própria CPU
  iii) na CPU com extensão na memória

2.2. BARRAMENTOS

Um importante aspecto na operação de computadores digitais è que a transmissão de informação entre os vários componentes do "hardware" seja rápida e precisa. Isto è conseguido nos microcomputadores através de barramentos. Um barramento (bus) è um conjunto de fios agrupados pela similaridade de suas funções, e que percorrem todos os blocos funcionais do microcomputador. Comumente tem-se:

- **Barramento de Dados:** em que os dados podem fluir em ambas as direções, entre a CPU, a memòria, e os dispositivos de entrada e saìda.

- **Barramento de Endereços:** que conecta a CPU à memòria e aos dispositivos de I/O. Este barramento è usado para identificar uma posição de memòria particular ou um determinado dispositivo de I/O.

- **Barramento de Controle:** liga a CPU, a memòria e os dispositivos de I/O, a fim de indicar o tipo de atividade a ser executada.

A figura 2.4 (compare com a figura 2.1) mostra o diagrama de blocos simplificado de um microcomputador com seus barramentos.

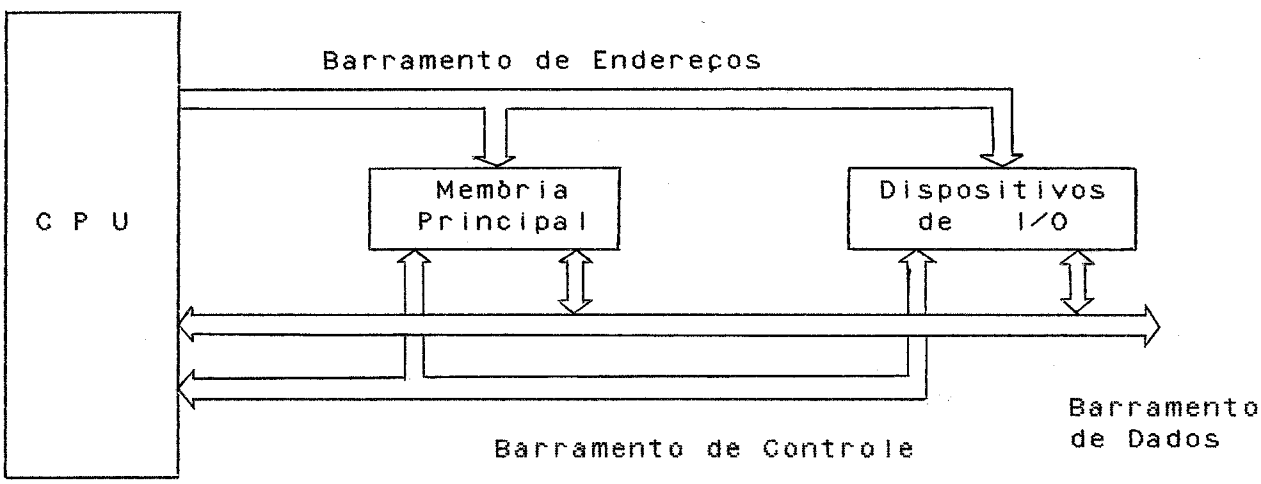


Fig. 2.4. - Barramentos em um Microcompurador

2.3. MEMORIAS EM MICROCOMPUTADORES

As cèlulas bàsicas de memòria, identificàveis por uma posição distinta, são chamadas <u>palavras</u>. O número de bits em uma palavra podem variar de 4 a 16 nos microcomputadores, a 128 nos computadores cientìficos de grande porte.

A memòria de um computador pode ser classificada como Memòria Principal e Memòria de Massa.

A Memòria Principal è aquela que pode ser acessada diretamente pela CPU e pode ser de vàrios tipos:

- RAM (Random Acess Memory) - permitem acesso tanto para leitura como para gravação, mas são voláteis.

- ROM (Read Only Memory) - são memórias não voláteis, mas que só permitem acesso para leitura. Podem ainda ser dos tipos:
  - PROM (Programable ROM): programáveis pelo usuário através de equipamento especial, ficando então fixas,
  - EPROM (Erasable PROM): podem ser programadas sucessivas vezes.

- EEPROM - um meio termo entre as memórias RAM e EPROM, sendo programadas pelo computador, não voláteis, mas ainda lentas na gravação.

As memórias de massa, como discos magnèticos e fitas magnèticas, não são acessadas diretamente pela CPU, e são usadas como memórias de reserva, podendo ser usadas para um grande número de dados. Estas memórias são mais lentas, mas custam muito menos por bit armazenado.

2.4. ENTRADA E SAIDA DE DADOS

a) DISPOSITIVOS DE ENTRADA E SAIDA

Hà muitos tipos diferentes de dispositivos para a comunicação entre o usuàrio e o sistema de computação, tais como impressoras, terminais de vìdeo, plotadores gràficos, discos magnèticos flexìveis, discos magnèticos rìgidos, fitas magnèticas, etc. Cada dispositivo tem sua velocidade caracterìstica, como mostra a tabele 2.1. abaixo:

Perifèrico	velocidade (bytes/seg)
impressoras	100 - 2000
terminais de vìdeo	30 - 960
fita magnètica	10^4 - 10^5
disco flexìvel	10^4 - 10^5
disco rìgido	10^5 - 10^6
ploters	100 - 1000

Tab. 2.1. - Alguns perifèricos de comunicação

A comunicação de dados pode ser efetuada de dois modos: através de transmissão paralela, onde por exemplo 8 bits são transmitidos ao mesmo tempo , ou através de transmissão sèrie, onde esses 8 bits são transmitidos sequencialmente no tempo, através de apenas 1 canal de cominicação, usualmente linhas telefônicas. A comunicação paralela è usada sempre que a comunicação exige grandes velocidades, enquanto que a transmissão sèrie para os casos que podem ser mais lentos, como terminais de vìdeo, e para transmissão de dados à grande distância.

b) CONVERSORES A/D E D/A

Interfaces de I/O que trabalham com sinais analògicos envolvem conversão Analògico/Digital (A/D) e Digital/Analògica (D/A).

Os conversores A/D convertem um sinal de tensão analògico dentro de uma determinada faixa (tais como -10V a +10V ou 0V a 5V, etc.) para um nùmero inteiro na forma binària. O intervalo de variação desse nùmero inteiro, e então a resolução da conversão, dependem do nùmero N de bits do conversor. Para n=8 os nùmeros inteiros poderão ter I= 2^8=256 valores possìveis.

O erro de conversão è dado portanto pela variação de tensão contìnua que não causarà variação do numero digital correspondente, o que equivale a

$$E = \frac{[\text{tensão màxima - tensão mìmina}]_{\text{do conversor}}}{\text{nùmero de intervalos digitais}} \qquad (2.1.)$$

O erro relativo de conversão, para uma tensão v dentro da faixa do conversor, è definido como:

$$ER = E/(v\text{-tensão mìnima}) \qquad (2.2)$$

Pode-se ver então que obtem-se maior precisão para tensões grandes; para tensões pròximas da tensão mìmina os erros relativos podem ser muito altos. Isto nos indica que, para termos boa precisão, è necessàrio condicionar os sinais a serem convertidos para que não tenham excursões apenas em região de baixa precisáo; outro cuidado no condicionamento dos sinais è para não sairem fora da faixa de conversão, o que causarà saturação da saida digital.

A velocidade de conversão A/D està frequentemente compreendida entre 50000 e 100000 conversões por segundo.

O conversor D/A faz a operação inversa do conversor A/D. Um nùmero inteiro variando entre 2^N valores è transformado para uma tensão analògica variando dentro de uma determinada faixa de tensões. Em termos de erros valem as mesmas expressões do conversor A/D.

Para ilustrar o desempenho desses conversores, suponha um sinal de 4,5V entrando num conversor A/D de 8 bits, com faixa de sinais de entrada de −10V a +10V. O número binário resultante será dado por

$$n = \text{INT}\left[\frac{(4,5+10)\times 255}{20}\right] = 184 \qquad (2.3)$$

Apenas para efeito de exercício, considere esse número entrando num conversor D/A. Resultará a tensão

$$v' = \frac{184\times 20}{255} - 10 \implies v' = 4,43V \qquad (2.4)$$

A figura 2.5 abaixo, para fim de visualização dos sinais dos sinais envolvidos, mostra um conversor A/D seguido de um conversor D/A, onde pode-se ver os erros introduzidos pela curva característica não linear em forma de escada.

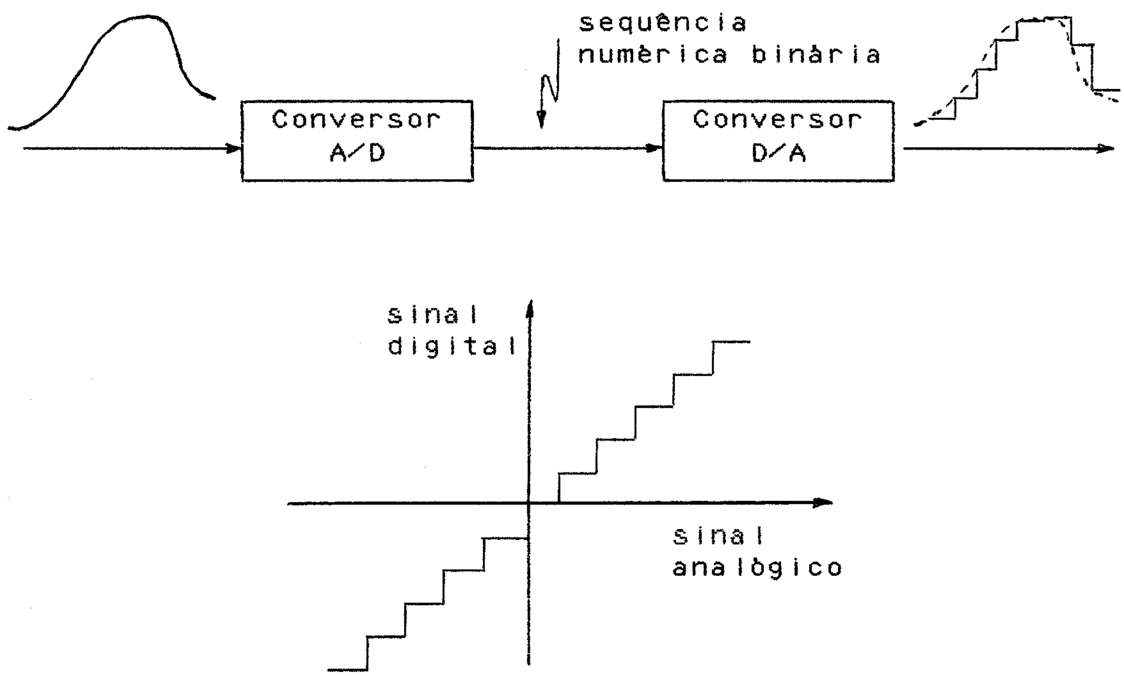


Fig. 2.5 - Conversores A/D e D/A e curva característica de conversão

c) MULTIPLEXAGEM DE DADOS

Uma maneira de usarmos apenas um Conversor A/D para fazermos diversos tipos de medidas é através do uso de <u>mulltiplexadores analógicos</u> (ver figura 2.6). Dessa forma as medidas são transmitidas sequencialmente, segundo uma dada programação.

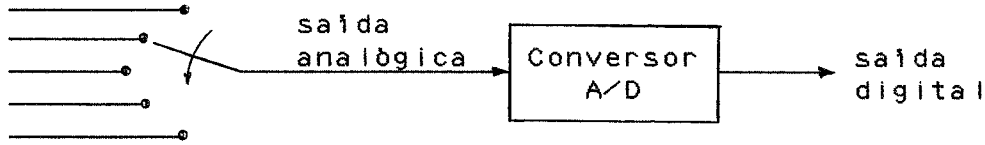


Fig.2.6. - Multiplexador e Conversor D/A

Um canal típico de aquisicão de dados è mostrado esquematicamente pela figura 2.7.

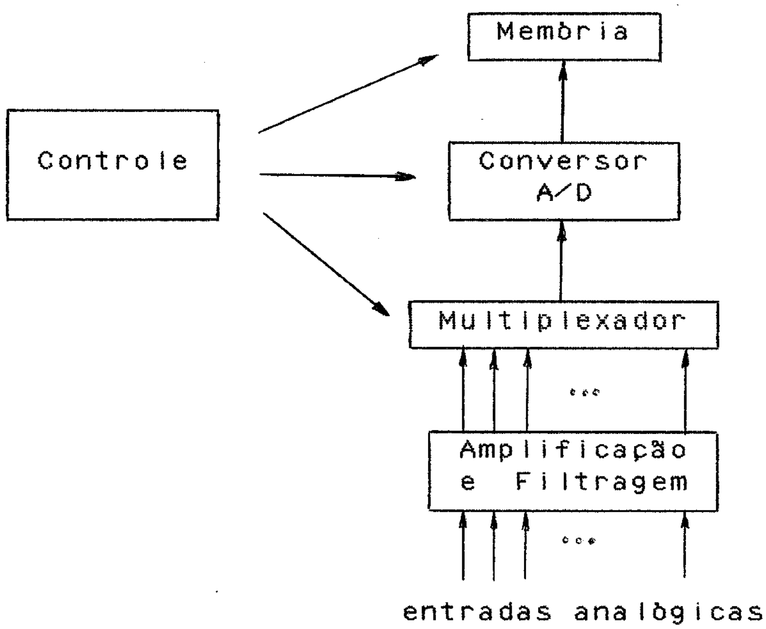


Fig. 2.7. - Sistema de Aquisição de Dados típico

O bloco de amplificação e filtragem è responsável pelo condicionamento dos sinais em termos de amplitudes e de supressão de ruidos, antes de serem processados.

d) ENTRADAS E SAIDAS PROGRAMADAS

A troca de informação entre o computador e os dispositivos externos geralmente è controlada pela CPU, podendo ser classificada de tres formas:

- Transferência Incondicional
- Transferência Condicional
- Transferência por Interrupção

A Transferência Incondiconal pressupõe que o dispositivo de I/O sempre esteja pronto para uso. E determinada exclusivamente pela instrução de I/O. Se o perifèrico não estiver pronto o dado poder se perder.

Na Transferência Condicional a programação è feita em dois passos. No primeiro, através de um "loop" de espera, verifica-se se o disposistivo de I/O està apto para a transferência; no segundo a instrução de I/O executa a transferência. A principal vantagem desse tipo de transferência è a sincronização do computador e os perifèricos, mas às custas de um tempo gasto na sincronização.

A Transferência por Interrupção torna mais eficiente o uso do tempo de computação. A transferência è feita sob controle do computador, não sendo preciso a constante verificação do perifèrico. O computador continua executando o programa de "background" atè que o perifèrico esteja apto para a transferência. Quando isto ocorre, o perifèrico gera um sinal de interrupção, que reconhecido pelo computador, faz com que a execução do programa de "background" seja interrompido, seu conteudo salvo, e sejam executadas as instruções de transferência de dados. Uma vez executada a transferência o computador retorna à execução do programa de "background" a partir do ponto em que foi interrompido.

A figura 2.8 abaixo representa esquematicamente esses tres tipos de transferência de dados.

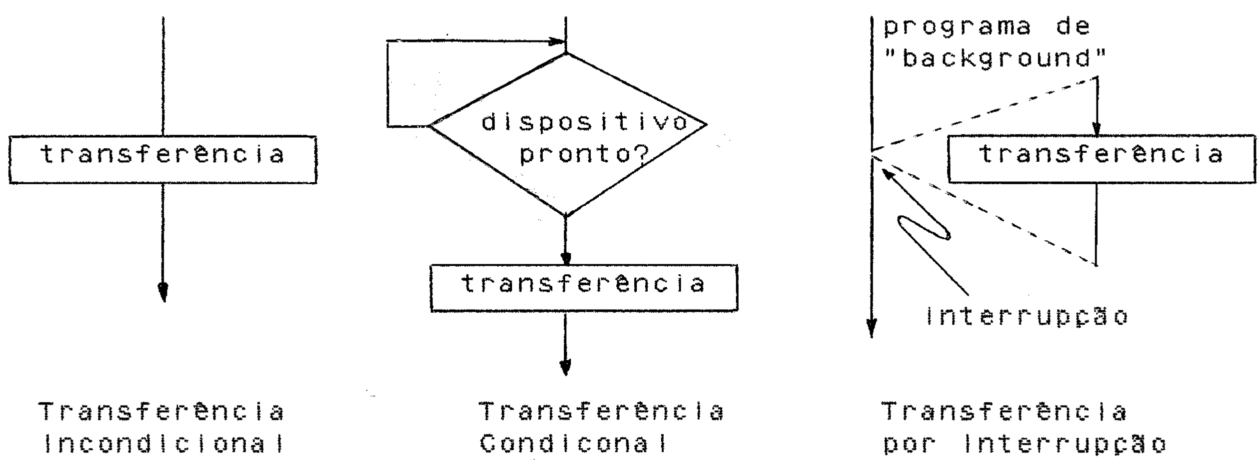


Fig. 2.8. - Modos de Transferência de Dados

Deve-se salientar que numa transferência por interrupção, antes do "salto" de transferência, os conteùdos dos registradores de trabalho devem ser armazenados na meemòria (em geral na pilha), e recarregados apòs a execução da transferência, de modo a não alterar o estado do programa original. Para mais de um perifèrico podem ser estabelecidas prioridades de interrupção, por "software" ou por "hardware".

2.5. SOFTWARE DE MICROCOMPUTADORES

Todas as informações usadas pelo microprocessador são armazenadas na memòria em forma de palavras, quer representem instruções ou dados. Instruções tìpicas tem os seguintes formatos:

uma sò palavra	còdigo de operação		o operando está em algum registrador
duas palavras	còdigo de operação	operando end. reg.	o operando 2 està em outro registrador
tres palavras	còdigo de operação	operando 1 end. mem.	o operando 2 està em algum registrador

De um modo geral essas instruções envolvem as seguintes operações:

- carregar o acumulador com palavra da memòria (Load)
- carregar a memòria com o conteùdo do acumulador (Store)
- soma binària
- subtração binària
- operações lògicas
- deslocamento em registradores à direita e à esquerda
- Saltos condicionais
- Operações de entrada e saida

a) ASSEMBLER

Visto sob o ponto de vista de còdigo binàrio, um programa diz-se estar em Linguagem de Màquina. Para facilitar a programação è interessante o uso de mnemônicos, que deverão gerar uma linguagem de programação, chamada Linguagem Assembly. Por exemplo, LDA (Load Acumulator) corresponde à palavra binària 00111010 no Assembly do Intel 8085. O programa que faz a conversão dos còdigos meneumônicos para os còdigos binàrios chama-se Assembler (Montador).

A figura 2.10 abaixo ilustra a programação em códigos mnemônicos e binários do programa que realiza a operação elementar x = 3+4 , no Intel 8085.

ALGORITMO	ENDEREÇO	MNEMONICO	COD.BINARIO
carregue o acumulador	000	LDA	00111010
com o dado da posição 12	001	12	00001100
	002	0	00000000
acumulador ⟶ reg.E	003	MOV E,A	01011111
carreguer o acumulador	004	LDA	00111010
com o dado da posição 13	005	13	00001101
	006	0	00000000
acumulador+E ⟶ acumulador	007	ADD E	10000011
armazene o conteudo do	008	STA	00110010
acumulador na posição 14	009	14	00001110
	010	0	00000000
pare (fim)	011	HLT	01110110
	012	3	00000011
	013	4	00000100
	014	0	00000000

Fig. 2.10. - Operação, mnemônicos e códigos binários do programa x = 3+4

b) SOFTWARE BASICO

São os programas orientados para tornar eficientes os recursos de computação, compreendendo:

- **Sistemas Operacionais**: administram, e coordenam todas as funções e tarefas executadas pelo computador.

- **Sistemas Utilitários**: Editores de Texto, Ferramentas de Manejo de Arquivos (Tools), Bibliotecas Matemáticas e Gráficas, Programas de Diagnóstico e Manutenção, Sistemas de Base de Dados, etc.

- **Compiladores e Interpretadores**: ASSEMBLER, BASIC, PASCAL, FORTRAN, C, PROLOG, COBOL, ETC.

c) COMPILADORES E INTERPRETADORES

Compiladores e Interpretadores são programas desenvolvidos para fazer a tradução de um programa escrito numa linguagem de alto nível para a linguagem de màquina. Uma instrução em linguagem de alto nível geralmente corresponde a vàrias instruções da linguagem binària.

Os compiladores geram o chamado programa objeto, que constitui a tradução do programa em linguagem de alto nível, também chamado de programa fonte. Uma vez gerado o programa objeto, este poderà ser executado sem necessidade de nova compilação. Compiladores em geral são sistemas sofisticados, que procedem a uma anàlise otimizada do programa objeto, de modo a ocupar o menor espaço de memòria e/ou ser executado em menor tempo.

Contrariamente aos compiladores, os interpretadores não geram um programa objeto, mas interpretam e executam uma instrução fonte de cada vez. Esse fato torna lentos os programas interpretados, e portanto inadequados para rotinas repetitivas que precisem de rapidez, como em Controle de Processos.

Para ilustrar a vantagem de uma linguagem de alto nível em relação à programação, o programa da figura 2.10 em FORTRAN ficaria X = 3 + 4.

d) SOFTWARE DE APLICAÇÃO

Aquele desenvolvido pelo usuàrio, em geral em uma linguagem de alto nível, como PASCAL, FORTAN, BASIC, etc. Em termos de Controle de Processos, è comum também se programar em ASSEMBLER, com o fim de aumentar a compactação e a velocidade das operações de Controle em Tempo Real.

e) LINGUAGENS DE TEMPO REAL

São linguagens voltadas exclusivamente para Controle de Processos, onde o usuàrio simplesmente precisa introduzir dados referentes aos sistemas controlados, como parâmetros da planta, tipos de controladores, parâmetros dos controladores, etc. Como exemplos dessas linguagens podemos citar a ADA (USA) e a PEARL (Alemanha Ocidental).

CAPITULO 3 ANALISE DE SISTEMAS DE TEMPO DISCRETO NO PLANO Z

Neste capitulo estaremos estudando sistemas discretos lineares invariantes no tempo em abordagem que faz uso da Transformada Z. A partir da conceituação dos Sistemas de Tempo Discreto são vistos os conceitos de Soma de Convolução, Resposta ao Impulso, Função de Transferência, Estabilidade, e de Erros de Regime Permanente.

3.1. ALGUMAS DEFINIÇÕES

a) Sistemas de Tempo Contìnuo (ou Sistemas Contìnuos)

São aqueles para os quais todos os sinais neles presentes são sinais de tempo contìnuo, ou seja, sinais definidos em um intervalo contìnuo de tempo, e que portanto podem ser representados por funções de variàvel contìnua. A temperatura $\theta(t)$ de um forno varia sempre de modo contìnuo com o tempo, como ilustra a figura 3.1 abaixo.

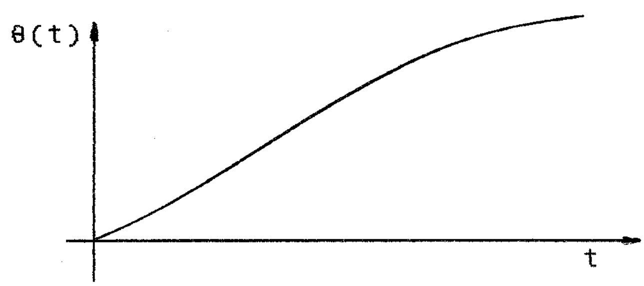


Fig. 3.1. - Sinal de Tempo Contìnuo

b) Sistemas de Tempo Discreto (ou Sistemas Discretos)

São aqueles para os quais as entradas e saìdas são sinais de tempo discreto, ou seja, sinais em que a variàvel independente toma valores discretos, constituindo uma sequência de números reais. Por exemplo, num sistema econômico a taxa de rendimentos do mes k è um sinal de tempo discreto, como ilustra a figura 3.2.

Enquanto a descrição matemàtica de sistemas de tempo contìnuo è geralmente feita por meio de equações diferenciais, em sistemas de tempo discreto è feita por meio de equações a diferenças.

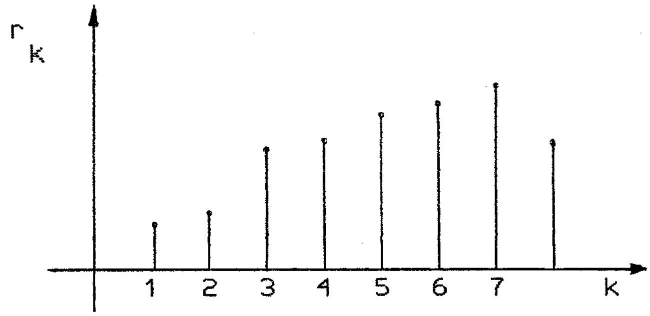


Fig. 3.2. - Sinal de Tempo Discreto

Por exemplo, considere o sistema composto de uma conta bancària em que a verificação do saldo p_k no mes k è feita logo apòs alguma alteração (saque ou depòsito) u_k efetuada nesse mes. Chamando r_k a taxa de rendimentos do mes k, a equação a diferenças que descreverà o comportamento do sistema serà dada por

$$p_k = (1 + r_{k-1})\, p_{k-1} + u_k \qquad (3.1)$$

E interessante ressaltar que o conceito de tempo, como entendemos, è um conceito do mundo contìnuo. Em sistemas discretos o conceito de tempo pode se perder; para tais sistemas o que interessa são os valores das variàveis em cada "passo", muitas vezes não importanto espaçamento de tempo entre eles. Por exemplo, se os rendimentos e alterações sò fossem computados quando os rendimentos atingissem uma taxa fixa r (independente do fato de isso ocorrer em uma semana ou em ano), o saldo do passo k seria agora descrito pela equação

$$p_k = (1 + r)\, p_{k-1} + u_k \qquad (3.2)$$

Temos então um sistema discreto em que o tempo perde significado. De um certo modo, todo sistema discreto deve ser interpretado dessa maneira.

c) <u>Sistemas Digitais</u>

São os sistemas de tempo discreto em que as amplitudes são tambèm discretizadas, podendo assumir portanto apenas determinados nìveis. Todo sistema de controle por computador è desse tipo, dada a natureza digital dos conversores A/D e D/A. O tratamento matemàtico desse tipo de sistema è bastante complicado devido aos erros de quantização (digitalização). Por isso, neste curso, desprezaremos o efeito da digitalização dos sinais, considerando variações contínuas em todas as amplitudes, supondo então um número de bits dos conversores suficientemente grande para não termos resultados distorcidos. Uma análise dos possíveis problemas dessa aproximação será feita em capítulo posterior.

d) Sistemas a Dados Amostrados

Sistemas de tempo discreto podem ter origem mesmo discreta, como os exemplos acima, ou podem se originar de amostragens em sistemas de tempo contínuo (como é o caso do Controle Digital). Neste caso o sistema discreto resultante é denominado Sistema a Dados Amostrados. E desse tipo de sistema que essencialmente trataremos neste curso.

3.2. SINAIS DE TEMPO DISCRETO

Na teoria de sistemas de tempo discreto os sinais serão representados por sequências numéricas. Uma sequência x de números em que o k-ézimo número da sequência é denotado x_k é formalmente escrita como

$$x = \{x_k\} \qquad -\infty < k < \infty \tag{3.3}$$

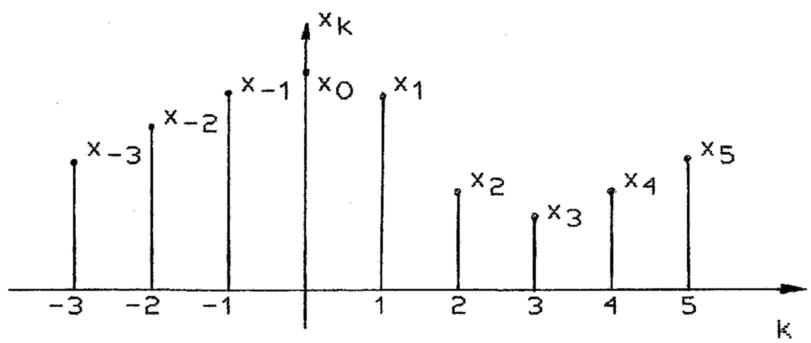


Fig. 3.3. - Gráfico da Sequência $\{x_k\}$

Embora as sequências nem sempre surjam da amostragem de formas de onda contínuas, chamaremos por conveniência x_j como a j-ézima amostra da sequência $\{x_k\}$, ou de modo mais curto, da sequência x_k. Não é correto pensar em x_k como valendo zero para k não inteiro; x_k simplesmente só é definida para k inteiro.

3.4

Exemplos de sequências:

a) Sequência Amostra Unitária

$$\delta_k = \begin{cases} 0, & k \neq 0 \\ 1, & k = 0 \end{cases}$$

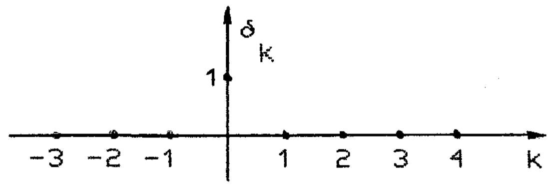


Fig.3.4-Sequência Amostra Unitária

Esta sequência, ilustrada pela figura 3.4, desempenha o mesmo papel em sistemas de tempo discreto que a função impulso unitário em sistemas contínuos; por isso ela è chamada frequentemente de sequência impulso discreto.

b) Sequência Degrau Unitário

$$u_k = \begin{cases} 1, & k \geq 0 \\ 0, & k < 0 \end{cases}$$

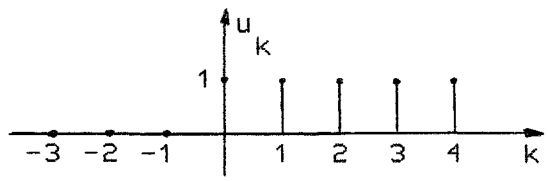


Fig.3.5-Sequência Degrau Unitário

Esta sequência relaciona-se com o impulso discreto atravès das relações

$$u_k = \sum_{i=0}^{\infty} \delta_{k-i} \qquad (3.4)$$

$$\delta_k = u_k - u_{k-1} \qquad (3.5)$$

c) Sequência Senoidal

$$x_k = A \operatorname{sen}(w_0 k + \Phi)$$

Note que :

- $T = 2\pi/w_0$ è inteiro $\longrightarrow$ x_k è periòdica com perìodo T
- T è racional$\neq$inteiro $\longrightarrow$ x_k è periòdica com perìodo >T
- T não è racional $\longrightarrow$ x_k não è periòdica

d) <u>Sequência Qualquer</u>

Para uma sequência x_k qualquer vale a relação

$$x_k = \sum_{i=-\infty}^{\infty} x_i \delta_{k-i} \tag{3.6}$$

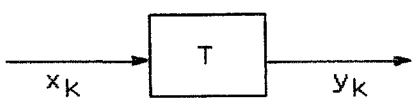
ou seja, a sequência x_k pode ser vista como a soma das sequências δ_{k-i} ponderadas pelos fatores x_i. Note que a expressão (3.4) è um caso particular dessa expressão.

3.3. RESPOSTA AO IMPULSO DE SISTEMAS DISCRETOS

Um sistema pode ser definido matematicamente como uma transformação, ou um perador, que mapeia uma sequência de entrada x_k em uma sequência de saìda y_k. Isto è denotado por

$$y_k = T[x_k] \tag{3.7}$$

ou, em blocos,



A saìda y_k correspondente a uma entrada do tipo impulso discreto è denominada <u>resposta ao impulso</u>, e denotada por h_k, ou seja,

$$h_k = T[\delta_k] \tag{3.8}$$

Um sistema è dito invariante no tempo quando a resposta a um impulso atrasado de n passos corresponde à resposta ao impulso atrasada de n passos, ou seja,

$$y_k = T[\delta_{k-n}] = h_{k-n} \quad , \forall\, n \tag{3.9}$$

3.4. SISTEMAS DISCRETOS LINEARES INVARIANTES NO TEMPO

A classe de sistemas lineares è definida como aquela que obedece o Princìpio da Superposição: se y_{1k} e y_{2k} são as respostas às entradas x_{1k} e x_{2k}, respectivamente, então

$$T[ax_{1k}+bx_{2k}] = aT[x_{1k}] + bT[x_{2k}] = ay_{1k} + by_{2k} \tag{3.10}$$

para constantes arbitràrias a e b.

Portanto, usando-se (3.6) e (3.7) pode-se escrever

$$y_k = T[x_k] = T[\sum_{n=-\infty}^{\infty} x_n \delta_{k-n}] \tag{3.11}$$

Se o sistema è linear, resulta

$$y_k = \sum_{n=-\infty}^{\infty} x_n T[\delta_{k-n}] \tag{3.12}$$

Supondo ainda invariância no tempo, usando (3.9), tem-se a soma de convolução

$$\boxed{y_k = \sum_{n=-\infty}^{\infty} x_n h_{k-n}} \tag{3.13}$$

Atravès da mudança de variàvel k-n=i , (3.13) torna-se

$$\boxed{y_k = \sum_{i=-\infty}^{\infty} x_{k-i} h_i} \tag{3.14}$$

As expressões (3.13) e (3.14) acima mostram um fato muito importante: a resposta de qualquer sistema discreto linear invariante no tempo a uma dada entrada pode ser obtida se tão somente conhecermos a resposta ao impulso desse sistema. Pode-se concluir daì que a resposta ao impulso de tais sistemas corresponde a um modo de representà-los. Como veremos adiante, a resposta ao impulso estarà diretamente ligada à função de transferência do sistema.

Um outro modo importante de representar sistemas discretos è através de equações a diferenças, como vimos no ítem 3.1. Uma sub-classe importante de sistemas lineares invariantes no tempo è aquela em que a saída y_k e a entrada x_k satisfazem a uma equação do tipo

$$\sum_{n=0}^{N} a_n y_{k-n} = \sum_{m=0}^{M} b_m x_{k-m} \qquad (3.15)$$

Os exemplos 3.1 e 3.2 abaixo mostram como esses tipos de representação podem ser intercambiados.

Exemplo 3.1.: Determine a equação a diferenças relativa ao sistema cuja resposta ao impulso è da forma

$$h_n = a^n u_n$$

onde u_n è a sequência degrau unitário.

Solução:

$$y_n = \sum_{i=-\infty}^{\infty} x_{n-i} h_i = \sum_{i=0}^{\infty} x_{n-i} a^i = a \sum_{i=0}^{\infty} x_{n-i} a^{i-1} =$$

$$= a \sum_{i=1}^{\infty} x_{n-i} a^{i-1} + x_n = a \sum_{j=0}^{\infty} x_{(n-1)-j} a^j + x_n$$

o que resulta na equação a diferenças

$$y_n - a\, y_{n-1} = x_n$$

Exemplo 3.2.: Determine a resposta ao impulso do sistema dado pela equação a diferenças acima.

Solução: Para $x_n = \delta_n$ (sequência impulso unitário), e supondo causalidade, ou seja, a resposta ao impulso, h_k, nula para $k<0$, tem-se

$$\begin{aligned}
h_0 &= ah_{-1} + \delta_0 = 1 \\
h_1 &= ah_0 + \delta_1 = a \\
h_2 &= ah_1 + \delta_2 = a^2 \\
&\cdots\cdots\cdots\cdots \\
h_n &= ah_{n-1} + \delta_n = a^n
\end{aligned}$$

ou,

$$h_n = a^n u_n$$

3.5. A TRANSFORMADA Z

A Transformada Z desempenha para os sistemas discretos um papel análogo ao da Transformada de Laplace para os sistemas de tempo contínuo. Equações diferenciais são transformadas em equações algébricas usando-se Transformada de Laplace; equações a diferenças são transformadas em equações algébricas pelo uso da Transformada Z.

Analogamente à Transformada de Laplace, em que se trabalha apenas com sinais definidos para tempos positivos, na Transformada Z trabalharemos apenas com sequências unilaterais à direita, para as quais,

$$x_k = 0 \quad , \quad k<0 \tag{3.16}$$

Define-se Transformada Z de uma sequência unilateral à direita x_k como

$$Z[x_k] = X(z) = \sum_{k=0}^{\infty} x_k z^{-k} \tag{3.17}$$

onde z é uma variável complexa.

A partir dessa definição pode-se observar de imediato a linearidade da Transformada Z, ou seja,

$$Z[ax_k+by_k] = aZ[x_k] + bZ[y_k] \tag{3.18}$$

para quaisquer constantes reais a e b.

Exemplos:

a) Sequência Amostra Unitária

$$Z[\delta_k] = \sum_{k=0}^{\infty} \delta_k z^{-k} = 1 \tag{3.19}$$

b) Sequência Degrau Unitário

$$U(z) = \sum_{k=0}^{\infty} 1.z^{-k} = 1 + z^{-1} + z^{-2} + \ldots$$

$$= \frac{1}{1 - z^{-1}} = \frac{z}{z - 1} \qquad (|z|>1) \tag{3.20}$$

c) <u>Sequência Exponencial</u>

$$x_k = \begin{cases} 0 & , \quad k<0 \\ a^k & , \quad k\geq 0 \end{cases}$$

$$X(z) = \sum_{k=0}^{\infty} a^k z^{-k} = 1 + az^{-1} + a^2 z^{-2} + \ldots$$

$$= \frac{1}{1 - az^{-1}} = \frac{z}{z - a} \qquad (|z|>|a|) \tag{3.21}$$

3.6. FUNÇÃO DE TRANSFERENCIA DISCRETA

Para sistemas discretos lineares invariantes no tempo com entrada x_n unilateral à direita, a expressão (3.14) transforma-se em

$$y_k = \sum_{n=0}^{\infty} x_n h_{k-n} \tag{3.22}$$

A partir daí tem-se

$$Y(z) = \sum_{k=0}^{\infty} y_k z^{-k} = \sum_{k=0}^{\infty} \sum_{n=0}^{\infty} x_n h_{k-n} z^{-k}$$

Para sistemas causais pode-se escrever ainda

$$Y(z) = \sum_{k=n}^{\infty} h_{k-n} z^{-(k-n)} \sum_{n=0}^{\infty} x_n z^{-n} = \sum_{j=0}^{\infty} h_j z^{-j} \sum_{n=0}^{\infty} x_n z^{-n}$$

ou,

$$Y(z) = H(z)X(z) \tag{3.23}$$

A equação (3.23) acima é a versão transformada da equação (3.22) para sistemas causais. H(z), a Transformada Z da Resposta ao Impulso, é chamada de <u>Função de Transferência Discreta</u>. A figura 3.7, através de diagrama de blocos, procura dar uma visão funcional ao sistema e à relação (3.23).

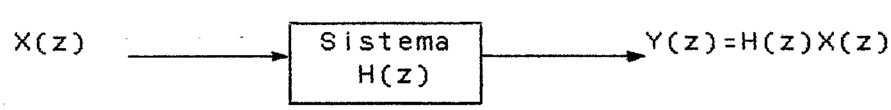


Fig.3.7. - Função de Transferência H(z)

Seja $Y_D(z)$ a resposta do sistema à entrada degrau $X(z)=1/(1-z^{-1})$. H(z) pode ser expressa, como usaremos mais tarde, por

$$H(z) = (1 - z^{-1})\, Y_D(z) \tag{3.24}$$

3.7. OPERAÇÕES DE ATRASO

A partir da definição de Transformada Z tem-se

$$Z[x_k] = \sum_{k=0}^{\infty} x_k z^{-k} = \sum_{k=1}^{\infty} x_k z^{-k} + x_0$$

$$= z^{-1} \sum_{k=0}^{\infty} x_{k+1} z^{-k} + x_0$$

Portanto,

$$Z[x_{k+1}] = zZ[x_k] - zx_0 \tag{3.25}$$

Através de procedimento indutivo, a expressão (3.25) pode ser generalizada para

$$Z[x_{k+n}] = z^n Z[x_k] - \sum_{i=0}^{n-1} x_i z^{n-i} \tag{3.26}$$

Portanto, para condições iniciais nulas pode-se escrever

$$Z[x_k] = z^{-n} Z[x_{k+n}] \tag{3.27}$$

A relação (3.27) acima corresponde a um sistema que atrasa n passos, como ilustra a figura 3.8 abaixo.

$$x_j \longrightarrow \boxed{z^{-n}} \longrightarrow y_j = x_{j-n}$$

Fig. 3.8. - Diagrama de blocos e função de transferência de um atrasador de n passos

3.8. RESOLUÇÃO DE EQUAÇÕES A DIFERENÇAS

Mais facilmente do que em sistemas contínuos com equações diferenciais, equações a diferenças podem ser resolvidas por meio de simples recursões. Por exemplo, considere o sistema discreto descrito por

$$x_{k+2} + 3x_{k+1} + 2x_k = 0 \qquad (3.28)$$

com as condições iniciais $x_0=0$ e $x_1=1$.

A equação (3.28), dada a suposição de causalidade do sistema, pode ser expressa como

$$x_{k+2} = -3x_{k+1} - 2x_k \qquad (3.29)$$

de onde, a partir das condições iniciais resulta, recursivamente,

$$x_2 = -3 \quad ; \quad x_3 = 7 \quad ; \quad x_4 = -15 \quad ; \quad \ldots$$

Com a utilização da Transformada Z pode-se conseguir uma solução não recursiva. Aplicando-se a expressão (3.26) tem-se:

$$\begin{cases} Z[x_{k+1}] = zX(z) - zx_0 \\ Z[x_{k+2}] = z^2X(z) - z^2x_0 - zx_1 \end{cases} \qquad (3.30)$$

Usando esses resultados em (3.28) resulta

$$(z^2 + 3z + 2)\, X(z) = z^2x_0 + zx_1 + 3zx_0 \qquad (3.31)$$

ou

$$X(z) = \frac{z}{z^2 + 3z + 2} = \frac{z}{z+1} - \frac{z}{z+2}$$

Logo,

$$x_k = (-1)^k - (-2)^k \qquad (3.32)$$

Esse exemplo mostra claramente que, por meio da Transformada Z, a equação a diferenças (3.28) foi transformada na equação algébrica em z (3.31). Esse mesmo procedimento pode ser usado na obtenção de funções de transferência discreta.

A equação a diferenças genèrica de um sistema linear, invariante no tempo, causal, da forma

$$y_k + a_1 y_{k-1} + \ldots + a_n y_{k-n} = b_0 u_k + b_1 u_{k-1} + \ldots + b_m u_{k-m} \qquad (3.33)$$

resultarà, para condições iniciais $y_{-1} = y_{-2} = \ldots = y_{-n} = 0$, na função de transferência

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1 z^{-1} + \ldots + b_m z^{-m}}{1 + a_1 z^{-1} + \ldots + a_n z^{-n}} \qquad (3.34)$$

Em termos de polinômios em z, H(z) pode ser escrita como

$$H(z) = \frac{z^{n-m}(b_0 z^m + b_1 z^{m-1} + \ldots + b_m)}{z^n + a_1 z^{n-1} + \ldots + a_{n-1} z + a_n} \quad , \; n \geq m \qquad (3.35)$$

ou

$$H(z) = \frac{b_0 z^m + b_1 z^{m-1} + \ldots + b_m}{z^{m-n}(z^n + a_1 z^{n-1} + \ldots + a_n)} \quad , \; n \leq m \qquad (3.36)$$

Pode-se ver então que a ordem n_0 do sistema, dada pelo grau do denominador de H(z) , vale

$$n_0 = \max[\, n \; , \; m \,] \qquad (3.37)$$

Note-se das expressões (3.35) e (3.36) que o grau do numerador em z è sempre menor ou igual ao grau do denominador. Se isto não ocorrer ,estaremos diante de sistemas não causais, e portanto não realizàveis. Por exemplo:

$$\frac{Y(z)}{U(z)} = \frac{z^2 + 3z + 2}{z + 1} = \frac{z + 3 + 2z^{-1}}{1 + z^{-1}}$$

ou

$$y_k = u_{k+1} + 3u_k + 2u_{k-1} - y_{k-1}$$

Observe que para a determinação de y_k è necessàrio usar o valor da entrada u_{k+1}, um passo adiante, o que mostra a não causalidade do sistema.

3.9. A TRANSFORMADA Z INVERSA

Da mesma forma que a Transformada de Laplace para sistemas contìnuos, a Transformada Z inversa, na Teoria de Controle, não desempenha função de destaque, embora possa ser ùtil na previsão do comportamento temporal de sistemas.

Estudaremos tres mètodos de obtenção da sequência unilateral à direita que deu origem a uma determinada Transformada Z. Em todos os casos estaremos supondo a Transformada Z da forma racional pròpria, ou seja, dada por uma razão de polinômios, em que o grau do numerador è menor ou igual ao grau do denominador (sistema causal).

Considere, por exemplo, o sistema dado pela função de transferência G(z) abaixo, sujeito à entrada x_k degrau unitàrio, ou seja, $X(z)=z/(z-1)$.

$$G(z) = \frac{Y(z)}{X(z)} = \frac{10}{z - 2} \tag{3.38}$$

A Transformada Z da saìda y_k serà então

$$Y(z) = \frac{10\ z}{(z-1)(z-2)} \tag{3.39}$$

a) Mètodo da Recorrência temporal

Este mètodo è muito importante, pois dà origem aos algoritmos que serão usados na implementação de controladores digitais e, na realidade, constitui um mètodo de simulação de sistemas discretos.

A partir da Função de Transferência (3.38) pode-se escrever

$$\frac{Y(z)}{X(z)} = \frac{10\ z^{-1}}{1 - 2z^{-1}} \tag{3.40}$$

ou

$$y_k = 2y_{k-1} + 10\ x_{k-1} \tag{3.41}$$

Se a entrada x_k è uma sequência degrau unitàrio, então

$$x_k = \begin{cases} 0 & , \quad k<0 \\ 1 & , \quad k\geq 0 \end{cases}$$

Por recorrência, o algoritmo (3.41) resultarà então

$$y_0=0 \quad ; \quad y_1=10 \quad ; \quad y_2=30 \quad ; \quad y_3=70 \quad ; \quad y_4=150 \quad ; \quad \ldots$$

Note que a saìda pode ser obtida para qualquer outro tipo de entrada; daì a importância deste mètodo, uma vez que, por meio de um computador digital, a função de transferência G(z) pode ser facilmente implementada através da expressão (3.41).

b) <u>Mètodo da Expansão em Frações Parciais</u>

Este mètodo è paralelo ao mètodo da expansão em frações parciais usado na Transformada de Laplace Inversa. Neste caso, expande-se Y(z)/z em frações parciais e, identificando-se cada fração através de uma tabela de Transformadas Z, obtem-se a sequência y_k.

A partir de (3.39) tem-se

$$\frac{Y(z)}{z} = \frac{-10}{z-1} + \frac{10}{z-2}$$

ou

$$Y(z) = \frac{-10z}{z-1} + \frac{10z}{z-2}$$

Portanto,

$$y_k = 10\,(-1 + 2^k) \tag{3.42}$$

A vantagem deste mètodo è a obtenção de resultados não recursivos, podendo-se então de forma imediata determinar valores das sequências mesmo para valores grandes de k.

c) <u>Mètodo da Expansão em Sèrie de Potências</u>

Este mètodo resulta da pròpria definição da Tranformada Z:

$$Y(z) = \sum_{k=0}^{\infty} y_k z^{-k} = y_0 + y_1 z^{-1} + y_2 z^{-2} + \ldots \tag{3.43}$$

Expandindo $Y(z)$ em série de potências de z^{-1}, os valores de y_k serão dados pelos coeficientes de cada termo.

$$Y(z) = \frac{10z}{(z-1)(z-2)} = \frac{10z^{-1}}{1-3z^{-1}+2z^{-2}} = 10z^{-1} + 30z^{-2} + 70z^{-3} + \ldots \tag{3.44}$$

Comparando (3.43) e (3.44) resulta

$$y_0 = 0 \quad ; \quad y_1 = 10 \quad ; \quad y_2 = 30 \quad ; \quad y_3 = 70 \quad ; \quad \ldots$$

<u>Exercício</u>: Determine a Transformada Z Inversa de

$$H(z) = \frac{a\,z}{(z-a)^2}$$

$$H(z) = \frac{a\,z^{-1}}{1-2az^{-1}+a^2z^{-2}} = az^{-1} + 2a^2z^{-2} + 3a^3z^{-3} + 4a^4z^{-4} + \cdots$$

Logo, por indução, $h_k = k\,a^k$

3.10. ESTABILIDADE DE SISTEMAS DISCRETOS

A resposta de um sistema discreto, em relação aos polos a_i de sua função de transferência e do sinal de entrada, é composta de parcelas da forma

$$k^n a_i{}^k \tag{3.45}$$

onde n está relacionado com a multiplicidade do polo em questão.

É imediato verificar que para $|a_i|>1$ a parcela (3.45) é não limitada. Pelo Critério da Razão de convergência de séries (ver nota de rodapé da próxima página), pode-se mostrar que essa parcela é limitada para

$$|a_i| < 1 \tag{3.46}$$

Portanto pode-se concluir que um sistema discreto linear invariante no tempo é estável se todos os polos de sua função de transferência obedecem a relação (3.46) acima, ou seja, os polos devem estar dentro de um círculo de raio unitário e com centro na origem do plano z (ver figura 3.9). Observe os últimos exemplos estudados.

Se os polos do sistema tem mòdulo unitàrio, a parcela (3.45) sò è limitada (caso de Estabilidade Marginal) se n=0, ou seja, se o polo tiver multiplicidade um.

Para a determinação da estabilidade sem precisar fatorar o polinômio característico (para verificar a localização dos polos), pode-se utilizar o Critèrio de de Routh-Hurwitz aplicado ao polinômio característico modificado pela transformação bilinear

$$z = \frac{1 + w}{1 - w} \quad \text{ou} \quad w = \frac{z - 1}{z + 1} \qquad (3.47)$$

Essa transformação faz o mapeamento do círculo unitàrio do plano z no semi-plano esquerdo do plano w.

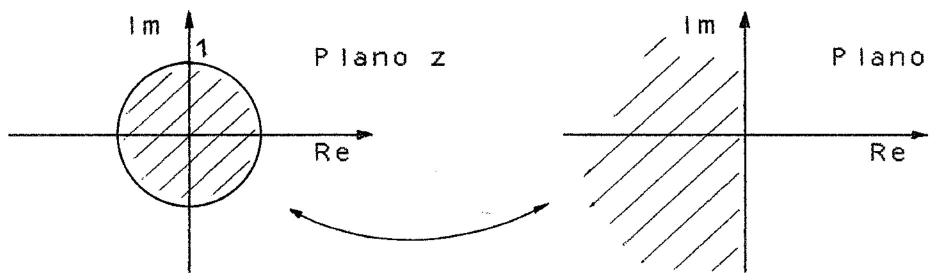


Fig. 3.9 - Mapeamento da Transformação (3.47)

Usando-se portanto a transformação bilinear (3.47), a função de transferência modificada $H_m(w)$ serà estàvel se e somente se os polos em w estiverem localizados no semi-plano esquerdo. Pode-se então aplicar o Critèrio de Estabilidade de Routh Hurwitz a $H_m(w)$.

<u>Critèrio da Razão</u>: Seja a sequência $\{x_k\}$ com $x_k \neq 0$ e

$$\lim_{k \to \infty} | x_{k+1} / x_k | = L$$

Então: - Se $L<1$, $\Sigma |x_k|$ è convergente $\longrightarrow$ $\lim_{k \to \infty} x_k = 0$

$\longrightarrow$ $|x_k|$ è limitado

- Se $L>1$, Σx_k è divergente

(KAPLAN, W., Càlculo Avançado, Ed. Edgar Blucher Ltda, pag.357)

Por exemplo: Determine se o sistema abaixo, dado pela sua função de transferência H(z), è estàvel.

$$H(z) = \frac{z^2 + 13z + 7}{z^3 + .4z^2 - .25z - .1}$$

Aplicando (3.47) ao denominador de H(z) tem-se

$$\left(\frac{1+w}{1-w}\right)^3 + .4\left(\frac{1+w}{1-w}\right)^2 - .25\left(\frac{1+w}{1-w}\right) - .1$$

O denominador de $H_m(w)$ serà então dado pelo polinômio

$$.45w^3 + 2.55w^2 + 3.95w + 1.05$$

Aplicando-se o Critèrio de Routh-Hurwitz, tem-se

w^3	.45	3.95
w^2	2.55	1.05
w	3.764	0
1	1.05	

Como não existe mudança de sinal na primeira coluna, não existem polos no semi-plano w-direito, e portanto o sistema è Estàvel.

Se determinarmos os polos de H(z) obteremos $z_1=.5$, $z_2=-.5$ e $z_3=-.4$, todos estàveis, o que confere com o resultado acima.

3.11. TEOREMAS DE LIMITES TEMPORAIS

a) Teorema do Valor Inicial

"Se a sequência x_k tem Transformada Z X(z) e $\lim_{z\to\infty} X(z)$ existe, então

$$x_0 = \lim_{z\to\infty} X(z) \quad "$$

A prova desse teorema decorre da pròpria definição da Transformada Z.

b) <u>Teorema do Valor Final</u>

"Se a sequência x_k tem Transformada Z $X(z)$ e $X(z)$ não tem polos no exterior do anel unitário $|z|=1$, nem polos múltiplos sobre esse anel ($x(z)$ è estável), então

$$\lim_{k\to\infty} x_k = \lim_{z\to 1} (z-1)X(z) \quad \text{"} \tag{3.49}$$

Prova:

$$(z-1)X(z) = zX(z) - X(z)$$

$$= zx_0 + \sum_{k=0}^{\infty} (x_{k+1} - x_k)z^{-k}$$

Por causa da condição suposta de estabilidade tem-se

$$\lim_{z\to 1} (z-1)X(z) = x_0 + \lim_{k\to\infty} x_k - x_0 = \lim_{k\to\infty} x_k$$

3.12. CONSTANTES DE ERRO ESTACIONARIO

Considere o sistema discreto realimentado, da forma descrita pela figura 3.10.

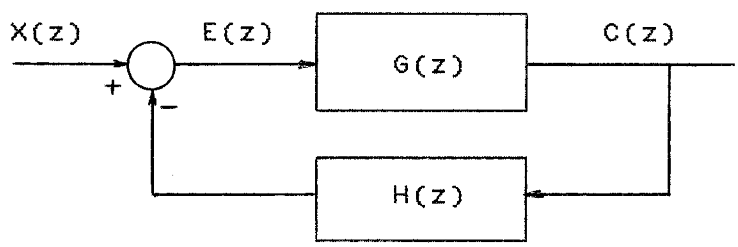


Fig. 3.10. - Sistema Discreto Realimentado

Usando o Teorema do Valor Final, o erro de regime estacionário para a entrada $X(z)$ serà dada por

$$e_{X\infty} = \lim_{z\to 1} (z-1)E(z) = \lim_{z\to 1} (z-1) \frac{X(z)}{1 + G(z)H(z)} \tag{3.50}$$

a) <u>Entrada Degrau</u>

$$X(z) = A \frac{z}{z-1} \tag{3.51}$$

onde A e a amplitude do degrau. A expressão (3.50) torna-se

$$e_{D\infty} = A \frac{1}{1 + G(1)H(1)} \tag{3.52}$$

Definindo-se <u>Constante de Erro de Posição</u>, K_p, como

$$\boxed{K_p = G(1)H(1)} \tag{3.53}$$

tem-se,

$$e_{D\infty} = \frac{A}{1 + K_p} \tag{3.54}$$

b) <u>Entrada Rampa</u>

$$X(z) = \frac{ATz}{(z-1)^2} \tag{3.55}$$

onde A corresponde à inclinação da rampa e T è o intervalo de tempo entre-amostras (*). A expressão (3.50) torna-se

$$e_{R\infty} = \lim_{z \to 1} \frac{A\,T}{(z-1)G(z)H(z)} \tag{3.56}$$

---

* - A presença do termo temporal T indica que estamos supondo que a sequência x_k è formada a partir de amostragem periòdica com período T do sinal rampa $x(t) = At$. Tem-se então $x_k = x(kT) = ATk$, o que leva a (3.55). Para sistemas discretos puros pode-se supor $T = 1$.

Definindo-se a Constante de Erro de Velocidade, K_V, como

$$\boxed{K_V = \lim_{z \to 1} \frac{(z - 1)}{T} G(z)H(z)} \tag{3.57}$$

tem-se, a partir de (3.56),

$$e_{R\infty} = \frac{A}{K_V} \tag{3.58}$$

Note que as definições de k_p e k_V são análogas àquelas de sistemas contínuos; quanto maiores forem estas constantes, menores os erros de regime correspondentes.

Se denominarmos Sistemas Tipo K quando tivermos a função de transferência de malha aberta $G(z)H(z)$ com k polos em $z = 1$, então temos:

sistema tipo zero ----> K_p finito ----> $K_V = 0$
sistema tipo um ----> $K_p = \infty$ ----> k_V finito
sistema tipo dois ----> $k_p = \infty$ ----> $k_V = \infty$

CAPÍTULO 4 ANALISE DE SISTEMAS A DADOS AMOSTRADOS NO PLANO Z

Existem àreas na Teoria de Controle que trabalham com sistemas puramente discretos, como os econômicos; porèm a grande maioria das plantas controladas são de tempo contínuo, como processos de temperatura, pressão, nível, vazão, etc. Muita experiência tem sido acumulada no controle de tais sistemas. A fim de extender a aplicação de computadores digitais e aproveitar esse conhecimento adquirido, è muito importante estudar a discretização de sistemas de tempo contínuo, tendo em vista dois problemas bàsicos: a formulação de modelos discretos de sistemas contínuos, e a transformação de cotroladores analògicos em algoritmos digitais.

Este capítulo tem por objetivo o estudo de sistemas discretos obtidos a partir de amostragens periòdicas de sistemas de tempo contínuo, tendo como ferramental bàsico a Transformada Z, estudada no capítulo anterior. Com a intropção da Trasformada Z Modificada chega-se à modelagem de um sistema de controle computadorizado.

4.1. RELAÇAO ENTRE OS POLOS DE SISTEMAS CONTÍNUOS E DOS SISTEMAS AMOSTRADOS

Considere um sinal de tempo contínuo $x_a(t)$ sendo amostrado periodicamente com período T, como ilustra a figura 4.1, gerando a sequência $\{x_k\}$, da forma

$$x_k = x_a(kT) \tag{4.1}$$

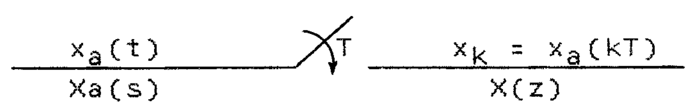


Fig. 4.1. - Amostragem de um sinal contínuo

O sinal $x_a(t)$, em relação à sua Transformada de Laplace $X_a(s)$ pode ser escrito como

$$x_a(t) = \frac{1}{2\pi j} \int_{\sigma-jw}^{\sigma+jw} X_a(s)\ e^{st}\ ds \qquad (4.2)$$

Em relação ao sinal amostrado tem-se

$$X(z) = \sum_{k=0}^{\infty} x_k z^{-k} = \sum_{k=0}^{\infty} z^{-k} \frac{1}{2\pi j} \int_{\sigma-jw}^{\sigma+jw} X_a(s)\ e^{kTs}\ ds$$

$$= \frac{1}{2\pi j} \int_{\sigma-jw}^{\sigma+jw} X_a(s) \sum_{k=0}^{\infty} (e^{Ts} z^{-1})^k\ ds$$

$$= \frac{1}{2\pi j} \int_{\sigma-jw}^{\sigma+jw} \frac{z\ X_a(s)}{z - e^{sT}}\ ds \qquad (4.3)$$

Se $X_a(s)$ è racional pròpria, o Teorema dos Resìduos nos leva a

$$X(z) = \sum_i \text{Res}\ \left[\frac{z}{z - e^{sT}} X_a(s) \right]_{s=si} \qquad (4.4)$$

onde s_i são os polos de $X_a(s)$.

Por exemplo, $X_a(s) = a/(s+a)$,

$$X(z) = \text{Res}\ \left[\frac{z}{z - e^{sT}} \frac{a}{s + a} \right]_{s=-a} = \frac{az}{z - e^{-aT}}$$

Esse resultado jà era esperado, pois, a partir de Xa(s) tem-se $x_a(t) = ae^{-at}$ e $x_k = a(e^{-aT})^k$. Usando (3.21) chega-se imediatamente ao resultado acima.

Considere agora a figura 4.2 abaixo, que mostra um sistema discreto obtido a partir da amostragem de um sistema contínuo.

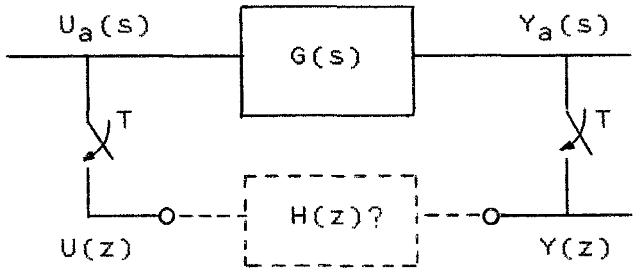


Fig. 4.2. - Discretização de um sistema contínuo

Será possível determinar a função de transferência $H(z) = Y(z)/U(z)$ do sistema discreto cujo comportamento corresponde exatamente ao sistema contínuo amostrado? Vejamos, usando a expressão (4.4) tem-se

$$H(z) = \frac{\sum_i \text{Res} \left[\frac{z}{z-e^{sT}} G(s) U_a(s) \right]_{s=si}}{U(z)} \tag{4.5}$$

Note então que H(z) depende de cada entrada particular $U_a(s)$, o que mostra a impossibilidade de se determinar uma função de transferência discreta única que represente o comportamento do sistema conínuo. Se particularizarmos a entrada, então poderemos calcular H(z) através de (4.5), mas H(z) só terá sentido como função de transferência para esse tipo particular de entrada.

Por exemplo, seja $G(s) = a/(s+a)$. Para entrada degrau unitário tem-se

$$H(z) = \frac{z-1}{z} \left[\frac{z}{z-1} + \frac{-z}{z-e^{-aT}} \right] = \frac{1-e^{-aT}}{z-e^{-aT}} \tag{4.6}$$

Para entrada rampa tem-se

$$H(z) = \frac{(z-1)^2}{T\ z} \left[\frac{-z(z-1-aT)}{a\ (z-1)^2} + \frac{z}{a(z-e^{-aT})} \right]$$

ou $$H(z) = \frac{(aT+e^{-aT}-1)z + 1-(1+aT)e^{-aT}}{aT(z-e^{-aT})} \qquad (4.7)$$

Lembre-se que o residuo de F(s) em um polo s_0 de multiplicidade m e dado por

$$Res[F(s),s_0] = \lim_{s \to s_0} \frac{d^{m-1}}{ds^{m-1}} \left\{ \frac{(s-s_0)^m F(s)}{(m-1)!} \right\}$$

A não existência de uma função de transferência entre U(z) e Y(z) já podia ser esperada, pois diferentes entradas $U_a(s)$ podem ocasionar idênticos sinais amostrados U(z), ocasionando diferentes saidas Y(z). Pode-se então concluir desse fato que o sistema discreto da figura 4.2 não è linear invariante no tempo, dada a inexistência de uma função de transferência que o represente.

A expressão (4.5) mostra ainda um outro fato extremamente importante na digitalização de sistemas contínuos: a cada polo $s_i = \sigma_i + jw_i$ de G(s) corresponderá em H(z) o polo

$$z_i = e^{s_i T} = e^{\sigma_i T} \; (\cos wT + j \; \text{sen} wT) \qquad (4.8)$$

A expressão (4.8), que relaciona os polos do sistema contínuo e do sistema discreto correspondente, è de grande importância na Teoria de Controle Digital, pois mapeia os polos do plano s no plano z. Note, como ilustra a figura 4.3, que a estabilidade è preservada, ou seja, polos no plano s localizados no semi-plano esquerdo resultarão em polos no plano z dentro do círculo unitário.

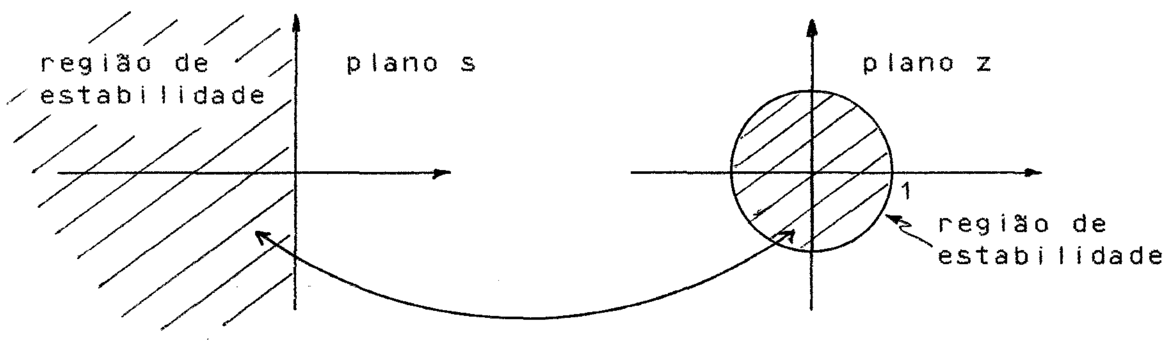


Fig. 4.3. - Preservação da estabilidade da relação (4.8)

A figura 4.4 abaixo ilustra o mapeamento feito pela relação (4.8) para alguns polos

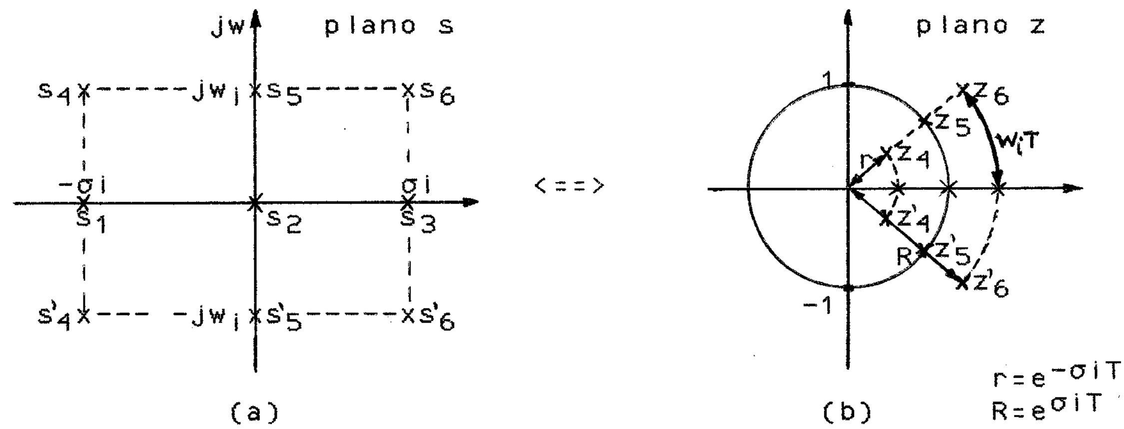


Fig.4.4. - Mapeamento (4.8) para alguns polos

É importante notar que não há uma correspondência um a um no mapeamento (4.8), pois a vários pontos do plano s podem corresponder apenas um ponto no plano z. Para um dado polo $s_i=\sigma_i+jw_i$ isto deverá ocorrer para todos os outros polos da forma

$$s_{iK} = \sigma_i + j(w_i + 2K\pi/T) \qquad (4.9)$$

Isto indica então que o semi-plano s esquerdo pode ser dividido em "tiras", cujo mapeamento pela relação (4.8) preenchem o círculo unitário, como mostra a figura 4.5.

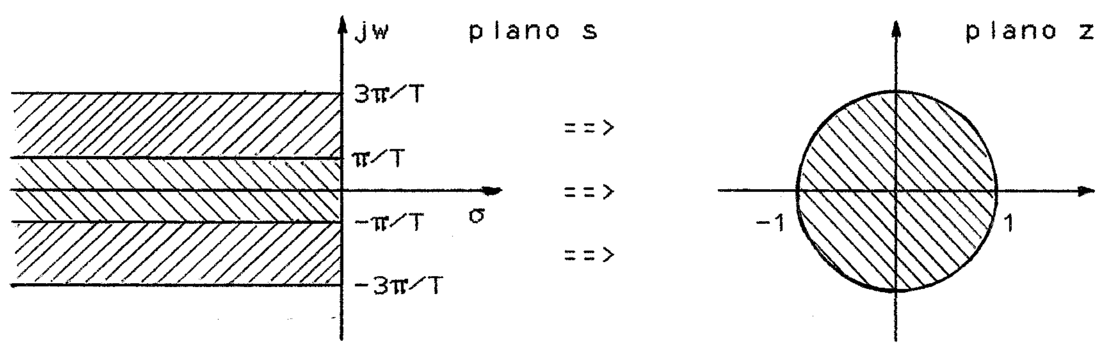


Fig 4.5. - "Tiras" no mapeamento (4.8)

A figura 4.6 abaixo mostra vàrias situações comuns do mapeamento (4.8).

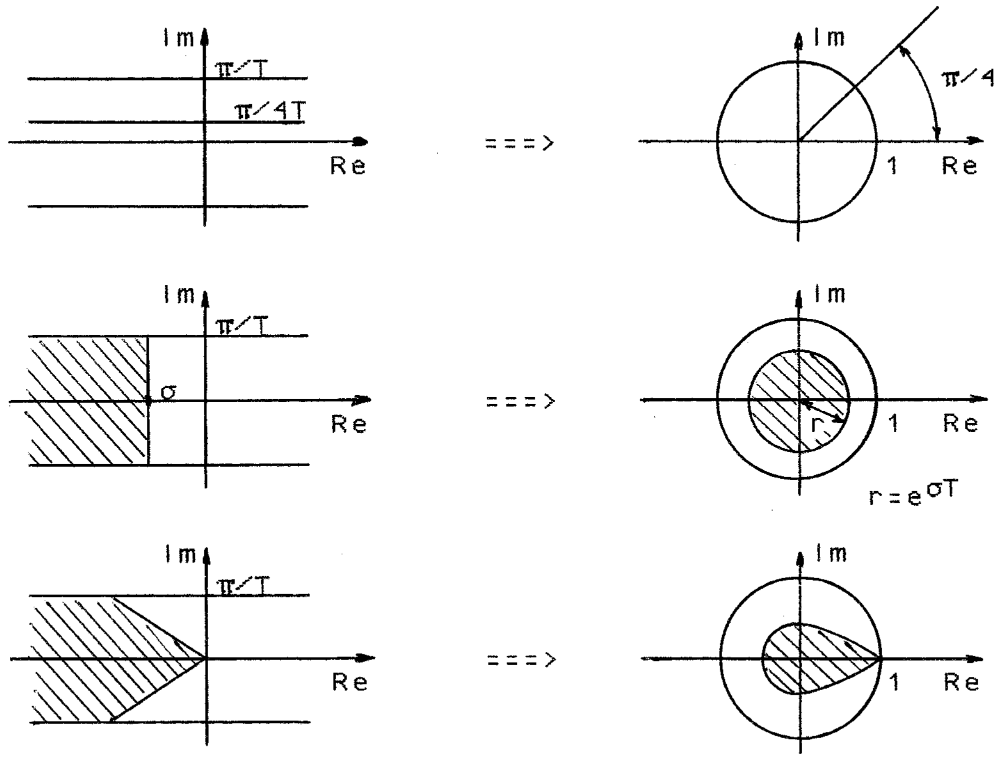


Fig.4.6. - O mapeamento (4.8) em vàrias situações comuns

4.2. SISTEMAS DE SEGUNDA ORDEM

A relação (4.8) è muito importante na determinação de polos de sistemas discretos. Se sabemos o que significa em termos de resposta transitòria uma certa localização de polos no plano s, a relação (4.8) nos mostra onde devemos alocar os polos no plano z que resultem nas mesmas caracterìsticas temporais. Por exemplo, considere o sistema contìnuo de segunda ordem dado por

$$G(s) = \frac{k}{s^2 + 2\zeta w_n s + w_n^2} \qquad (4.10)$$

Se os valores de ζ (constante de amortecimento) e w_n (frequência natural não amortecida) são conhecidos, os polos do sistema discreto correspondente deverão ser dados por

$$z_{1,2} = e^{-\zeta w_n T}[\cos(w_n T\sqrt{1-\zeta^2}) \pm j\,\text{sen}(w_n T\sqrt{1-\zeta^2})] \qquad (4.11)$$

O àbaco da figura 4.7 realiza esse mapeamento em função de ζ e $\theta = w_n T$.

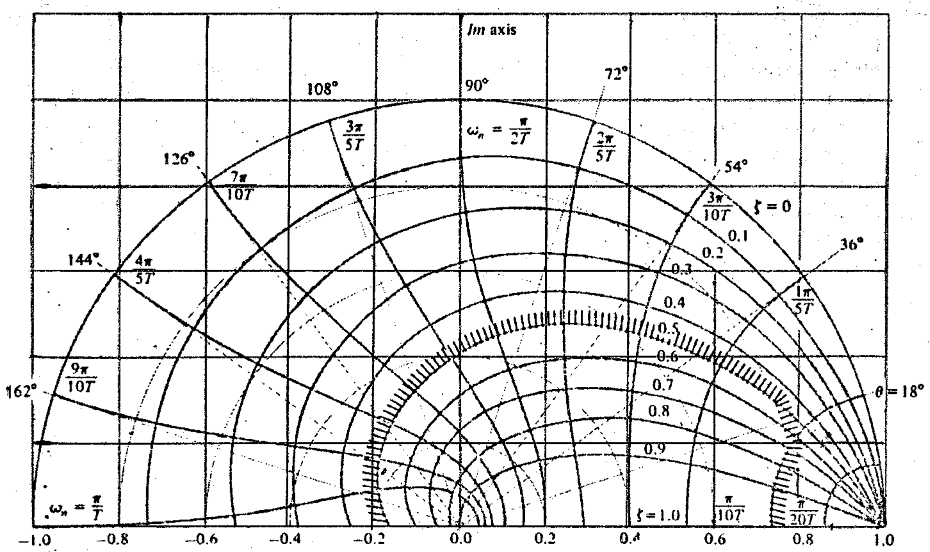


Fig.4.7.- O mapeamento (4.11) de polos de 2ª ordem

Pode-se notar a partir da figura acima que quanto maior o valor de w_n (sistema mais ràpido), os polos discretos tendem mais para a esquerda no interior do cìrculo unitàrio; quanto maior o valor de ζ (sistema mais amortecido), mais os polos discretos se aproximam do eixo real.

4.3. EFEITO DE ZEROS ADICIONAIS SOBRE A RESPOSTA TRANSITORIA

Neste ponto è importante frizar que a relação (4.8) não se aplica aos zeros de sistemas. Não existe uma relação explìcita semelhante para o mapeamento de zeros.

Estudaremos a seguir o feito de um zero adicional em sistemas discretos de 2ª ordem. Considere o sistema discreto dado pela função de transferência

$$G(z) = \frac{K(z-a)}{(z-z_1)(z-z_2)} \qquad (4.12)$$

onde z_1 e z_1 são dados por (4.11) em função de ζ e w_n.

Para $w_nT = 18^o$ e $\zeta = 0.5$, a figura 4.8 apresenta a resposta ao degrau para diversos valores do zero a, ajustando-se em cada caso o valor de k para que a resposta em regime seja a unidade.

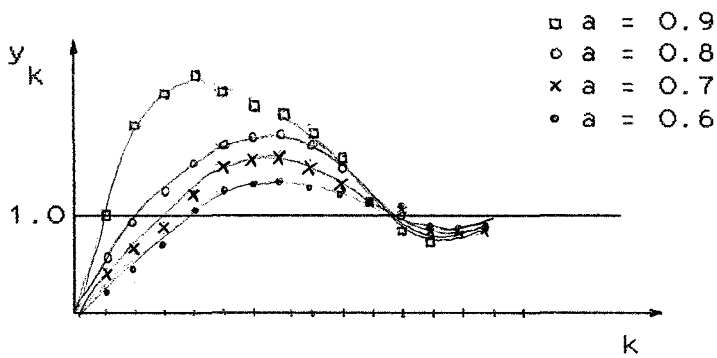


Fig.4.8.- Resposta ao degrau do sistema (4.12) com $\zeta = 0.5$ e $w_nT = 18^o$

O efeito principal do zero $z = a$ na resposta do sistema consiste na variação do "máximo overshoot" percentual. A figura 4.9 mostra essa variação em função da localização de a para $\zeta = 0.5$ e $\zeta = 0.707$, e para alguns valores de w_nT.

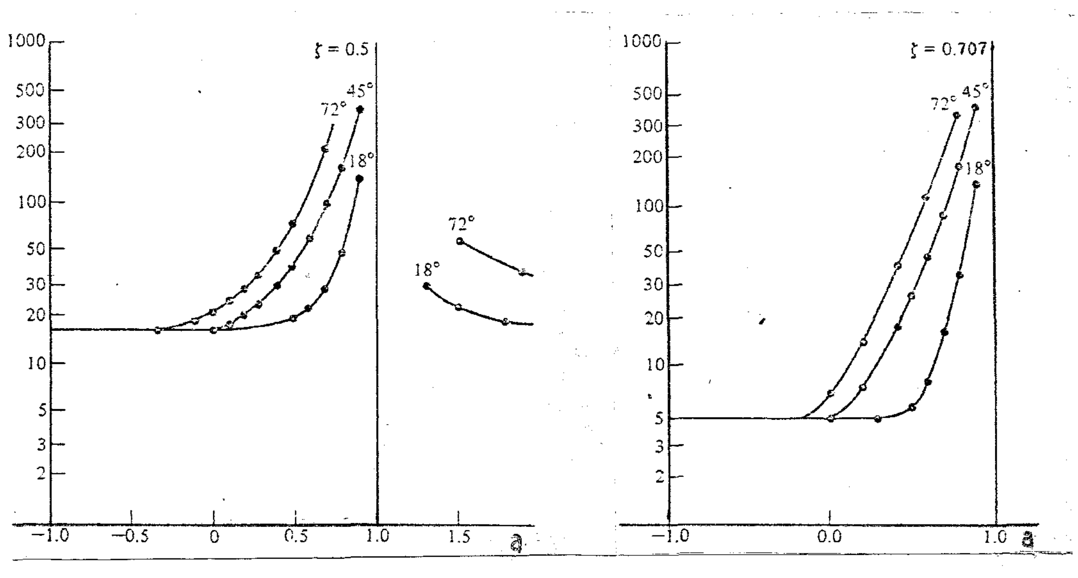


Fig.4.9.-Variação do "máximo overshoot" com a

Note-se que o zero adicional tem muito pouca influência quando está localizado sobre o eixo negativo, mas uma influência dramática quando se aproxima de 1.

Se um sistema apresenta algum zero localizado fora do círculo unitário êle è denominado <u>Sistema de Fase Não Mínima.</u> Tais sistemas, muito comuns quando se considera a digitalização de sistemas contínuos, apresentam certas dificuldades de controle devido a não terem função de transferência inversa estável. Esse fato deverá ser estudado posteriormente.

4.4. MODELAGEM DISCRETA DE SISTEMAS CONTÍNUOS EM CONTROLE DIGITAL

Em sistemas de controle através de computador digital, não è o esquema da figura 4.2 que è encontrado, mas um sistema em que o primeiro chaveador (que pode ser interpretado como um conversor analògico-digital) è substituido por um conversor digital-analògico, resultando o esquema dado pela figura 4.10.

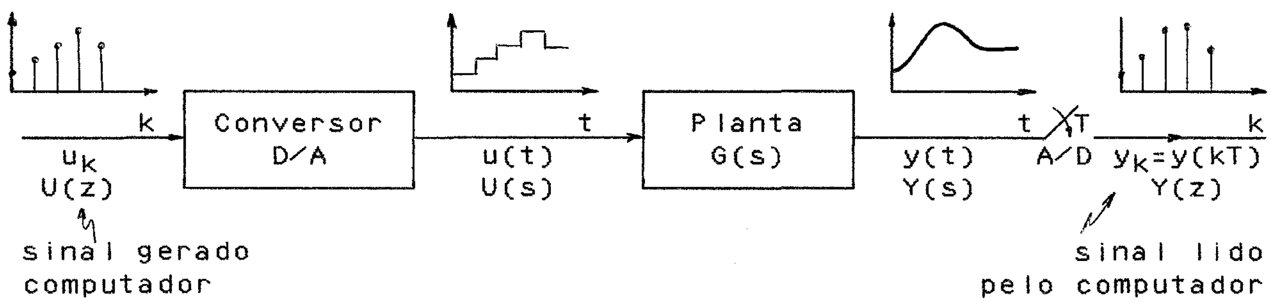


Fig.4.10. - Configuração típica da planta em Controle Digital

Note que agora as entradas e saídas efetivas são os sinais digitais (no caso da figura 4.2 a entrada era contínua). Neste caso sempre teremos uma relação biunívoca entre os sinais u_k e $u(t)$ e pode-se facilmente comprovar que o sistema è linear e invariante no tempo. A função de transferência desse sistema pode então ser determinada aplicando-se a expressão (4.5) com entrada degrau unitário, resultando:

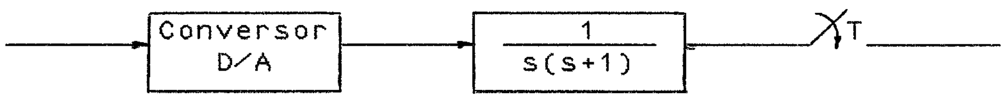
$$H(z) = \frac{Y(z)}{U(z)} = (z-1) \sum_i \operatorname{Res}\left[\frac{G(s)}{s(z-e^{sT})}\right]_{s=s_i} \qquad (4.13)$$

onde s_i são os polos de $G(s)/s$.

A expressão (4.13), que determina a função de transferência da planta discretizada, è pouco pràtica. Uma expressão mais àgil pode ser obtida a partir de (3.24) com entrada degrau unitàrio. Tem-se então

$$\begin{cases} H(z) = (1 - z^{-1})\, Z[y_d(kT)] \\ \\ \text{onde} \quad y_d(t) = L^{-1}[G(s)/s] \end{cases} \qquad (4.14)$$

Exemplo: Determinar a função de transferência discreta do sistema abaixo



$$\frac{G(s)}{s} = \frac{1}{s^2(s + 1)} = \frac{1}{s^2} - \frac{1}{s} + \frac{1}{s + 1}$$

$$y_d(t) = t - 1 + e^{-t} \qquad , \quad t \geq 0$$

$$y_d(kT) = kT - 1 + e^{-Tk} \qquad , \quad k \geq 0$$

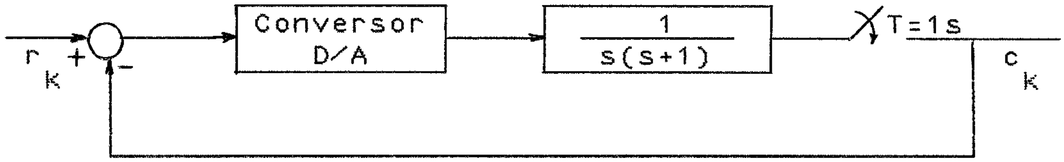
$$Z[y_d(kT)] = \frac{Tz}{(z - 1)^2} - \frac{z}{z - 1} + \frac{z}{z - e^{-T}}$$

$$= \frac{z[(T-1+e^{-T})z + 1 - (T+1)e^{-T}]}{(z - 1)^2(z - e^{-T})}$$

$$H(z) = \frac{(T- 1+ e^{-T})z + 1 - (T+1)e^{-T}}{(z - 1)\,(z - e^{-T})}$$

Note que os polos de G(s) e H(z) estão relacionados da forma (4.8), como esperado, o mesmo não acontecendo com os zeros.

Exercício: Obtenha a resposta ao degrau do sistema dado pela figura abaixo:



Utilizando-se os resultados do exercício anterior tem-se

$$\frac{C(z)}{R(z)} = \frac{\dfrac{e^{-1}z + 1 - 2e^{-1}}{z^2 - (1-e^{-1})z + e^{-1}}}{1 + \dfrac{e^{-1}z + 1 - 2e^{-1}}{z^2 - (1-e^{-1})z + e^{-1}}} = \frac{0.368z + 0.264}{z^2 - z + 0.632}$$

Para entrada degrau, usando-se um dos mètodos do ìtem 3.9, chega-se facilmente ao resultado mostrado pelo gràfico da figura 4.11 abaixo

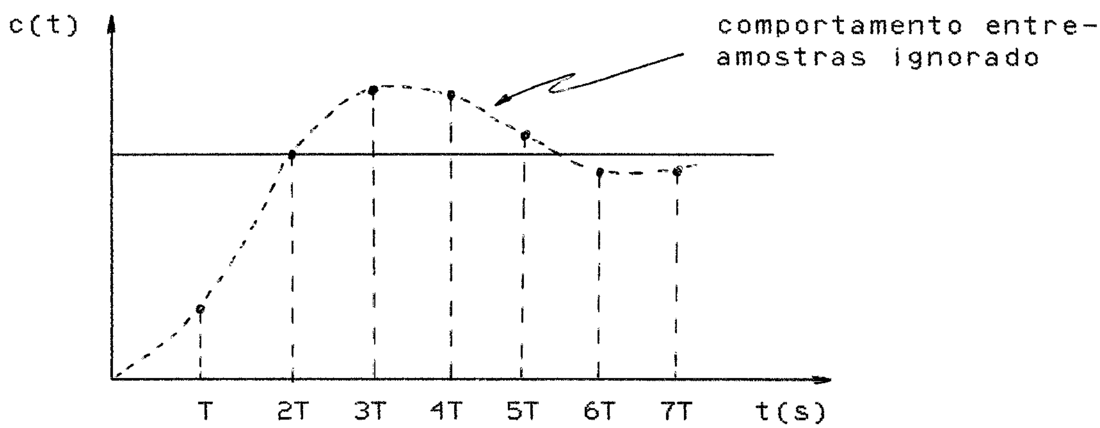


Fig 4.11. - Resposta ao degrau do sistema apresentado

Note que sò podemos obter informaçòes da saìda do sistema nos instantes de amostragem, nada se sabendo do comportamento da resposta entre as amostras.

4.5. COMPORTAMENTO ENTRE-AMOSTRAS - A FUNÇÃO DE TRANSFERENCIA MODIFICADA

O objetivo final do controle por computador è controlar a saída y(t) da planta contínua, como mostrado pela figura 4.10. Contudo, como verificado no exercício anterior, pela utilização da Transformada Z sò è possível determinar o comportamento de y(t) nos instantes discretos dados por t = kT. O que pode ocorrer è que o sistema de controle, realimentado somente a partir das amostras, resulte em um bom comportamento para a sequência y(kT), tal não acontecenco para o sinal contínuo y(t), ou seja, apresenta um mau comportamento entre-amostras. Essa situação è ilustrada pela figura 4.12 abaixo.

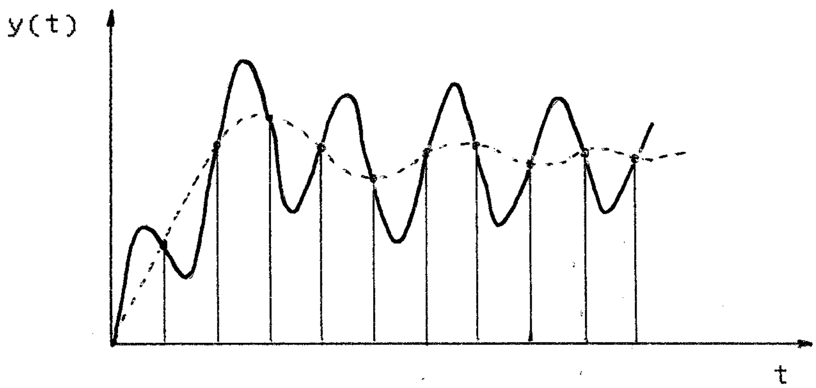


Fig 4.12. - Mau comportamento entre-amostras

A principal razão do mau comportamento entre-amostras està ligada a uma escolha inadequada do período de amostragem, como deveremos estudar em capítulo posterior.

A fim de podermos verificar o comportamento entre-amostras, introduziremos a Função de Transferência Modificada, acrescentando um atraso de tempo λT (0<λ<1) à planta contínua, conforme mostra a figura 4.13.

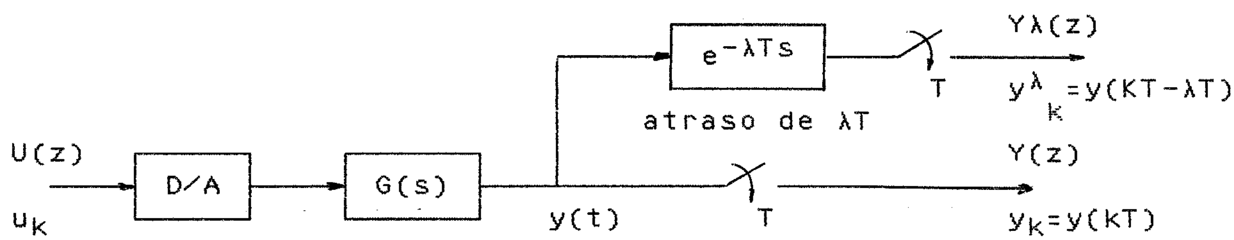


Fig.4.13.- Sistema discretizado com atraso de tempo de λT (0<λ<1)

Define-se a Função de Transferência Modificada , a partir de (4.14), como

$$\begin{cases} H_\lambda(z) = \dfrac{Y_\lambda(z)}{U(z)} = (1 - z^{-1})\ Z[y_d(kT-\lambda T)] \\ \\ \text{onde} \quad y_d(t) = L^{-1}[G(s)/s] \qquad (0<\lambda<1) \end{cases} \tag{4.15}$$

Tendo-se $H_\lambda(z)$ pode-se então determinar a sequência $y^\lambda{}_k = y(kt-\lambda T)$, ou seja, a variável $y(t)$ tomada entre os instantes de amostragem.

Exemplo: Determinar, usando Função de Transferência Modificada, $y(t)$ para $t = kT - 0.5T$, para alguns valores de k, para o sistema abaixo com entrada degrau unitário

$$u_k \longrightarrow \boxed{\text{Conversor D/A}} \longrightarrow \boxed{\dfrac{4}{(s+2)}} \longrightarrow y(t) \qquad T=1s$$

$$\frac{G(s)}{s} = \frac{4}{s(s+2)} = \frac{2}{s} - \frac{2}{s+2}$$

$$y_d(t) = 2 - 2e^{-2t} \qquad , \ t \geq 0$$

$$y_d(t-\lambda T) = \begin{cases} 2 - 2e^{-2(t-\lambda T)} & , \ t \geq \lambda T \\ 0 & , \ 0<t<\lambda T \end{cases}$$

$$Z[y_d(kT-\lambda T)] = \sum_{k=1}^{\infty} 2z^{-k} - \sum_{k=1}^{\infty} 2e^{-2(kT-\lambda T)}\ z^{-k}$$

$$= 2z^{-1} \sum_{m=0}^{\infty} z^{-m} - 2z^{-1}e^{2(\lambda-1)T} \sum_{r=0}^{\infty} (e^{-2T}z^{-1})^r$$

$$= \frac{2z^{-1}}{1-z^{-1}} - \frac{2z^{-1}e^{2(\lambda-1)T}}{1-e^{-2T}z^{-1}} = \frac{2}{z-1} - \frac{2e^{2(\lambda-1)T}}{z-e^{-2T}}$$

Portanto teremos

$$H_\lambda(z) = \frac{2(1 - e^{2(\lambda-1)T})z + 2e^{-2T}(e^{2\lambda T} - 1)}{z(z - e^{-2T})}$$

Para $T = 1s$, $\lambda = 0.5$ e entrada degrau unitàrio tem-se

$$Y_\lambda(z) = \frac{1.264z + 0.465}{(z - 0.135)(z - 1)} = 1.264z^{-1} + 1.9z^{-2} + 1.98z^{-3} + \ldots$$

Logo, $y(-0.5) = 0$, $y(0.5) = 1.264$
$y(1.5) = 1.9$, $y(2.5) = 1.98$ ...

4.6. TABELA DE TRANSFORMADAS Z

A tabela apresentada na pàgina seguinte relaciona $G(s)/s$, $y_d(t)$, $Z[y_d(kT)]$ e $Z[y_d(kT-\lambda T)]$ para vàrios casos, possibilitando então facilmente determinar as funções de transferência $H(z)$ e $H_\lambda(z)$, dadas respectivamente por (4.14) e (4.15).

$Y_d(s) = G(s)/s$	$y_d(t)$	$Z[y_d(kT)]$	$Z[y_d(kT-\lambda T)]$ $m=1-\lambda$
$\frac{1}{s}$	1	$\frac{z}{z-1}$	$\frac{1}{z-1}$
$\frac{1}{s^2}$	t	$\frac{Tz}{(z-1)^2}$	$\frac{mT}{z-1} + \frac{T}{(z-1)^2}$
$\frac{2}{s^3}$	t^2	$\frac{T^2 z(z+1)}{(z-1)^3}$	$T^2 \frac{m^2z^2+(2m-2m^2+1)z+(m-1)^2}{(z-1)^3}$
$\frac{1}{s+a}$	e^{-at}	$\frac{z}{z-e^{-aT}}$	$\frac{e^{-amT}}{z-e^{-aT}}$
$\frac{1}{(s+a)^2}$	te^{-at}	$\frac{Tze^{-aT}}{(z-e^{-aT})^2}$	$\frac{Te^{-amT}[e^{-aT}+m(z-e^{-aT})]}{(z-e^{-aT})^2}$
$\frac{a}{s(s+a)}$	$1-e^{-at}$	$\frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$	$\frac{(1-e^{-amT})z+(e^{-amT}-e^{-aT})}{(z-1)(z-e^{-aT})}$
$\frac{1}{(s+a)(s+b)}$	$\frac{1}{b-a}(e^{-at}-e^{-bt})$	$\frac{1}{b-a}\left[\frac{z}{z-e^{-aT}} - \frac{z}{z-e^{-bT}}\right]$	$\frac{1}{b-a}\left[\frac{e^{-amT}}{z-e^{-aT}} - \frac{e^{-bmT}}{z-e^{-bT}}\right]$
$\frac{a}{s^2(s+a)}$	$t - \frac{1}{a}(1-e^{-at})$	$\frac{Tz}{(z-1)^2} - \frac{(1-e^{-aT})z}{a(z-1)(z-e^{-aT})}$	$\frac{T}{(z-1)^2} + \frac{amT-1}{a(z-1)} + \frac{e^{-amT}}{a(z-e^{-aT})}$
$\frac{a^2}{s(s+a)^2}$	$1-(1+at)e^{-at}$	$\frac{z}{z-1} - \frac{z}{(z-e^{-aT})} - \frac{aTe^{-aT}z}{(z-e^{-aT})^2}$	$\frac{1}{(z-1)} - \left[\frac{1+amT}{(z-e^{-aT})} + \frac{aTe^{-aT}}{(z-e^{-aT})^2}\right]e^{-amT}$
$\frac{w}{(s+a)^2+w^2}$	$e^{-at}\,\mathrm{sen}\,wt$	$\frac{ze^{-aT}\,\mathrm{sen}\,wT}{z^2-2ze^{-aT}\cos wT+e^{-2aT}}$	$\frac{e^{-amT}[z\,\mathrm{sen}\,mwT+e^{-aT}\,\mathrm{sen}(1-m)wT]}{z^2-2ze^{-aT}\cos wT+e^{-2aT}}$
$\frac{s+a}{(s+a)^2+w^2}$	$e^{-at}\cos wt$	$\frac{z^2-ze^{-aT}\cos wT}{z^2-2ze^{-aT}\cos wT+e^{-2aT}}$	$\frac{e^{-amT}[z\cos mwT+e^{-aT}\,\mathrm{sen}(1-m)wT]}{z^2-2ze^{-aT}\cos wT+e^{-2aT}}$
$\frac{w}{s^2-w^2}$	$\sinh wt$	$\frac{z\,\mathrm{senh}\,wT}{z^2-2z\cosh wT+1}$	$\frac{\mathrm{senh}\,mwTz + \mathrm{senh}(1-m)wT}{z^2-2z\cosh wT+1}$
$\frac{s}{s^2-w^2}$	$\cosh wt$	$\frac{z(z-\cosh wT)}{z^2-2z\cosh wT+1}$	$\frac{\cosh mwTz - \cosh(1-m)wT}{z^2-2z\cosh wT+1}$

Tabela 4.1. - Tabela de Transformadas z

4.7. MODELAMENTO DE UM SISTEMA DE CONTROLE COMPUTADORIZADO

Considere o sistema de controle apresentado pela figura 4.14 abaixo, onde uma planta contínua linear invariante no tempo, dada por sua função de transferência G(s) è controlada por meio de um computador digital.

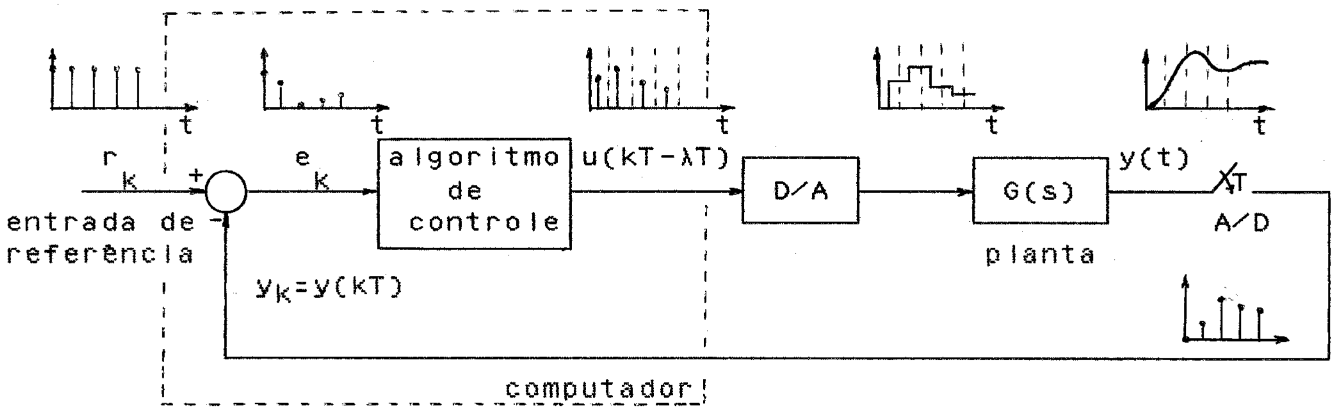


Fig. 4.14. - Diagrama de Blocos de um Sistema de Controle Computadorizado

A sequècia numèrica da saida do controlador sempre apresentarà um atraso de tempo λT (0<λ<1) em relação à sequência de entrada, devido ao tempo de computação do algoritmo de controle. Para controladores lineares, o algoritmo de controle pode então ser modelado como uma função de transferência discreta D(z) seguida de um atrasador de tempo, como ilustra a figura 4.15.

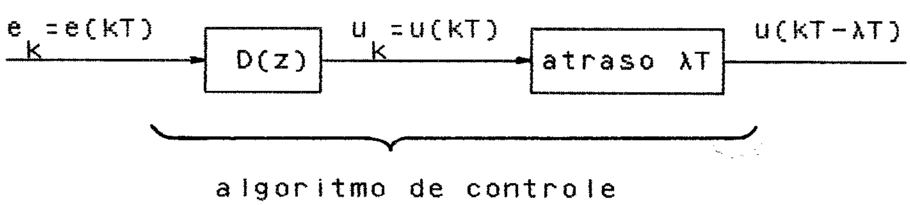


Fig.4.15 - Modelamento do Controlador Digital levando-se em conta o atraso de computação de D(z).

Sob o ponto de vista dos sinais discretos y_K, u_K e e_K, o atraso λT devido à computação do algoritmo de controle pode ser considerado como pertencendo à planta. A figura 4.14 pode então ser redesenhada, resultando os esquemas da figura 4.16.

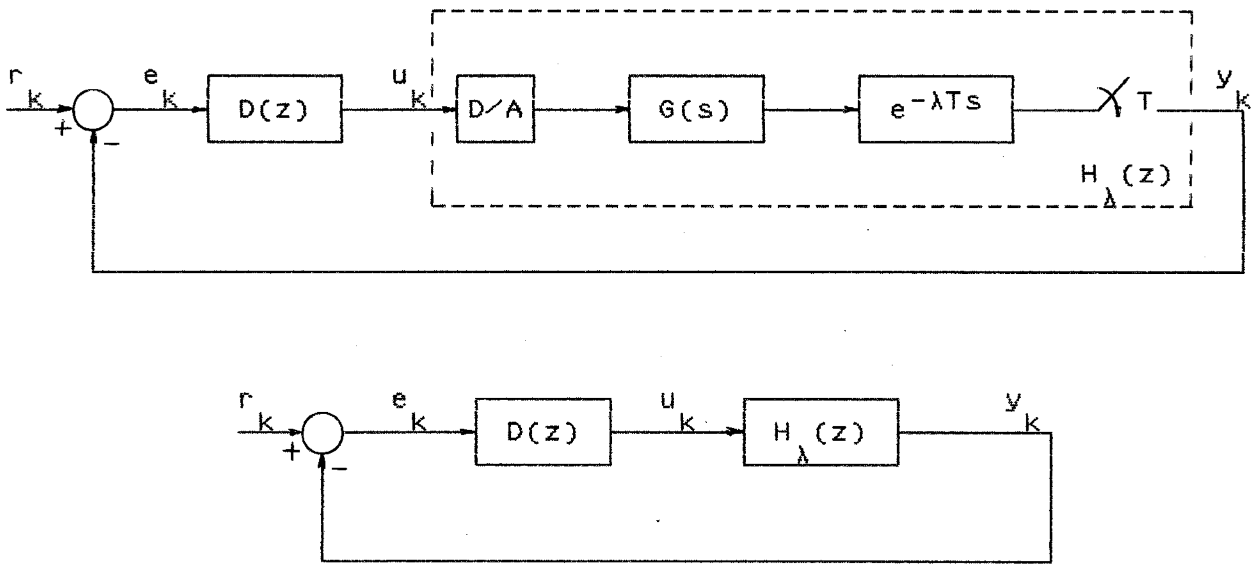


Fig. 4.16. - Modificações do diagrama de blocos da figura 4.15

Em relação à figura 4.16, $H_\lambda(z)$ representa a função de transferência discreta modificada referente à planta G(s), dada por (4.15), ou

$$\begin{cases} H_\lambda(z) = (1 - z^{-1})\ Z[y_d(kT-\lambda T)] \\ y_d(t) = L^{-1}[G(s)/s] \end{cases} \tag{4.16}$$

Para a determinação de $H_\lambda(z)$ é portanto suficiente conhecer G(s), o período de amostragem T, e o atraso do processamento digital, definido por λ. Em grande parte dos casos tem-se $\lambda T << T$, ou seja, o atraso de computação torna-se insignificante em face do período de amostragem. Isso ocorre sempre que o processo a ser controlado é de natureza lenta comparada com a velocidade de cálculo do computador. Nesses casos, e como suporemos deste ponto em diante neste trabalho, $H_\lambda(z)$ pode ser substituida por H(z), como calculada pela expressão 4.14, tomando o sistema controlado a forma da figura 4.17, com

$$\begin{cases} H(z) = (1 - z^{-1})\ Z[y_d(kT)] \\ y_d(t) = L^{-1}[G(s)/s] \end{cases} \tag{4.17}$$

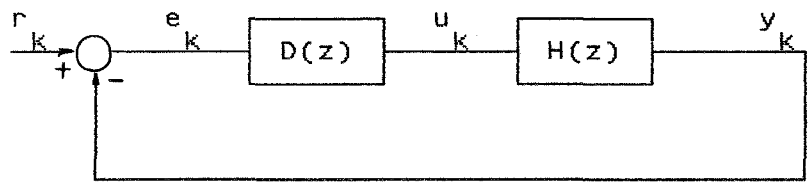


Fig.4.17.- Modelo discretizado do sistema controlado para $\lambda << 1$.

4.8. O CONTROLADOR DIGITAL

A função de transferência D(z) do controlador digital è geralmente racional, da forma

$$D(z) = \frac{U(z)}{E(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}} \qquad (4.18)$$

A realização computacional de D(z) consiste na determinação, em cada passo, de u_k, obtida a partir de (4.18), da forma

$$u_k = -a_1 u_{k-1} - \dots - a_n u_{k-n} + b_0 e_k + b_1 e_{k-1} + \dots + b_m e_{k-m} \qquad (4.19)$$

Para a realização de D(z) è necessàrio então ter armazenados alguns valores passados das sequências u_k e e_k, como mostra a expressão (4.19).

O fluxograma da figura 4.18 mostra como realizar o programa computacional que implementa um controlador digital. Seis partes podem ser distinguidas: de entrada de dados iniciais, de temporização para amostragens, de leitura de dados, de càlculos, de saìda de dados, e de atualização para o pròximo passo.

O tempo de computação do algoritmo digital não restringe-se apenas ao càlculo da expressão 4.19, mas tambèm devem ser levados em conta os tempos de entrada e saìda de dados e de atualização dos resultados. Os diagramas modificados da figura 4.16 nos permitiram raciocinar de modo a considerar nulo o tempo de realização computacional do algoritmo digital, ao torná-lo como que integrante da planta. Em muitos casos esse tempo não pode ser desprezado em face do perìodo de amostragem, e deve-se realmente usar a função de transferência modificada pelo atraso. E evidente que o perìodo de amostragem não pode ser menor que o tempo de computação do algoritmo, pois este não seria realizàvel.

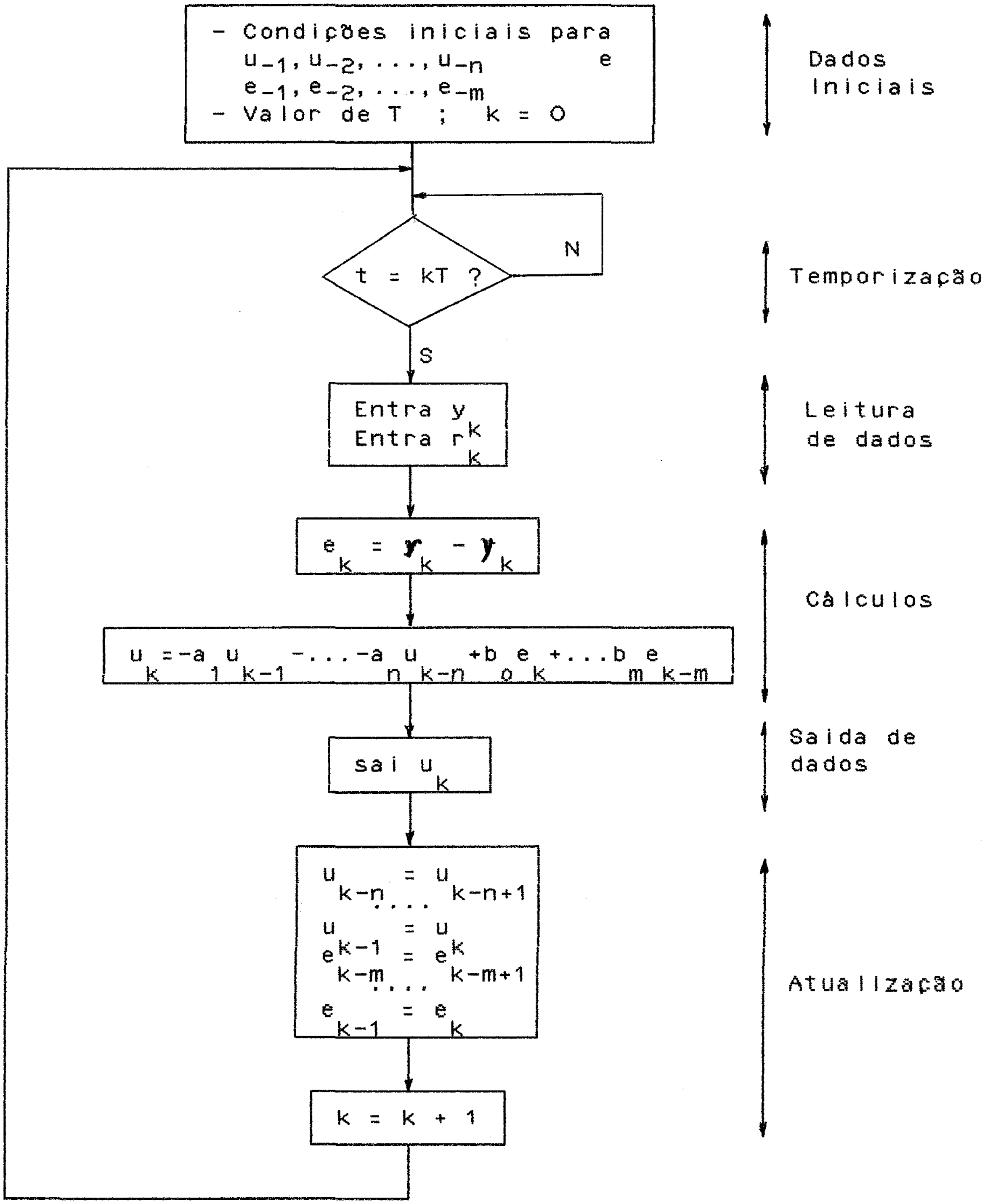


Fig.4.18.- Fluxograma de um programa de Controle Digital

Nos próximos capítulos estaremos estudando métodos de projeto do controlador digital D(z), a fim de que o sistema controlado apresente desempenho tal que satisfaça determinadas especificações, tanto temporais, como de resposta em frequência.

CAPITULO 5 CONTROLADORES DIGITAIS BASEADOS EM CONTROLADORES ANALOGICOS

Durante as últimas dècadas muita experiência tem sido acumulada em relação ao projeto de controladores analògicos, usando tècnicas bem conhecidas, como lugar das raízes e curvas de Bode. Este fato, juntamente com o recente grande crescimento da utilização do computador digital em controle de processos, torna natural o empenho no sentido de se aproveitar a experiência adquirida anteriormente, no projeto de Controladores Digitais. Este capítulo procura estudar vàrias tècnicas relacionadas a esse objetivo, bem como mostrar os cuidados e as limitações em cada mètodo.

A figura 5.1 abaixo ilustra bem o assunto que estaremos tratando: a determinação do Controlador Digital D(z) que, juntamente com os conversores digital/analògico e analògico/digital, possa melhor reproduzir o controlador contínuo $G_c(s)$, suposto conhecido, projetado por alguma tècnica tradicional da Teoria de Controle.

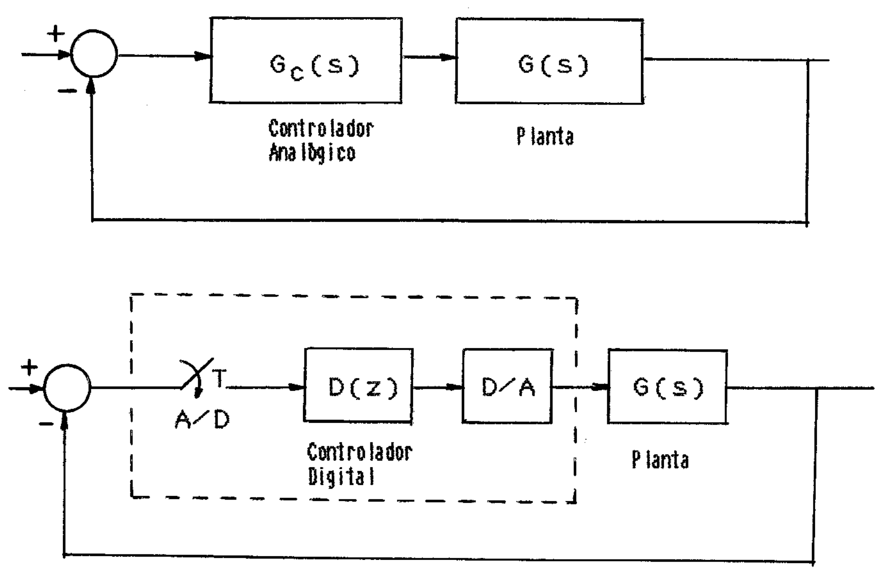


Fig.5.1.- Digitalização de Controladores Contínuos

5.1. SISTEMAS A DADOS AMOSTRADOS NO PLANO s

Considere-se a estrutura do controlador a dados amostrados da figura 5.2 abaixo.

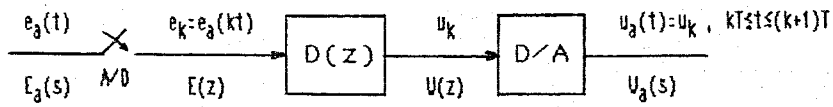


Fig.5.2.- O controlador digital com os conversores A/D e D/A

A Transformada de Laplace da saída $u_a(t)$ è dada por

$$U_a(s) = \int_0^{\infty} u_a(t)e^{-st}dt$$

$$= \sum_{k=0}^{\infty} u_k \int_{kT}^{(k+1)T} e^{-st}dt$$

$$= \frac{1 - e^{-sT}}{s} \sum_{k=0}^{\infty} u_k e^{-Tsk} \tag{5.1}$$

Usando a definiçao de Transformada z tem-se

$$U_a(s) = \frac{1 - e^{-Ts}}{s} U(z) \Big|_{z=e^{Ts}} \tag{5.2}$$

A relação entre as Transformadas de Laplace de $u_a(t)$ e $e_a(t)$ è dada então por

$$G(s) = \frac{U_a(s)}{E_a(s)} = \frac{1-e^{-Ts}}{sE_a(s)} D(z).E(z)\Big|_{z=e^{sT}} \tag{5.3}$$

Note que G(s) não pode ser caracterizada propriamente como uma Função de Transferência, uma vez que depende da entrada $E_a(s)$. Pode-se concluir que o sistema com entrada $e_a(t)$ e saída $u_a(t)$, como mostrado pela figura 2, não é linear invariante no tempo, dada a inexistência de uma Função de Transferência que o represente. Esse fato também poderia ser comprovado de outra forma: suponhamos diferentes sinais de entrada E_{a1} e E_{a2} que gerem o mesmo sinal amostrado e_K; para esses dois sinais teremos portanto a mesma saida $U_a(s)$, o que acarretaria duas funções diferentes para o mesmo sistema.

Se particularizarmos a entrada $e_a(t)$, então G(s) poderá ser determinada, mas só terá sentido como função de transferência para esse tipo particular de entrada.

5.2. PROJETO DE D(z) PARA MESMA RESPOSTA AO DEGRAU

Neste caso a equação (5.3) torna-se

$$G(s) = \frac{1 - e^{-Ts}}{s/s} \; D_1(z) \; \frac{1}{1 - z^{-1}}\bigg|_{z=e^{sT}}$$

$$= D_1(z)\big|_{z=e^{sT}} \qquad (5.4)$$

Se queremos que $G(s) = G_c(s)$, então o controlador digital será dado por

$$D_1(z) = G_c(\frac{1}{T} \ln z) \qquad (5.5)$$

A expressão (5.5) significa que o Controlador Digital $D_1(z)$ é obtido a partir do Controlador Contínuo $G_c(s)$, fazendo $s=(1/T)\ln z$. Como $\ln z$ é uma função transcendental, é comum se usar a aproximação em Série de Taylor

$$\ln x = 2\{ \frac{x - 1}{x + 1} + \frac{1}{3} (\frac{x - 1}{x + 1})^3 + \ldots\} \qquad (5.6)$$

o que resulta, tomando-se apenas o primeiro termo,

$$s \cong \frac{2}{T}\;\frac{z-1}{z+1} \tag{5.7}$$

A transformação (5.7) è comumente chamada de Transformação Bilinear, ou Transformação de Tustin; utilizando-se dessa transformação, o controlador digital (5.5) serà dado por

$$D_1(z) = G_c\left(\frac{2}{T}\;\frac{z-1}{z+1}\right) \tag{5.8}$$

O Controlador $D_1(z)$ è bastante conhecido na literatura de Controle, onde comumente se enfatiza a aproximação feita pela Transformação Bilinear (5.7).

Na anàlise que fizemos evidenciamos um outro tipo de problema: os controladores analògico e digital sò terão o mesmo comportamento se forem sujeitos a erro do tipo degrau, o que, evidentemente não deverà ser o caso na operação em malha fechada. Para outros sinais de erro deve-se então esperar um comportamento diferente para os sistemas controlados por $G_c(s)$ e $D_1(z)$.

5.3. PROJETO DE D(Z) COM ENTRADA SENOIDAL - IMPOSIÇÃO DE RESPOSTA EM FREQUENCIA

Neste caso devemos fazer $G(jw) = G_c(jw)$ para qualquer w, ou seja, impor uma dada resposta em frequência ao Controlador Digital. Sem duvida, essa filosofia è mais apropriada para o objetivo pretendido, do projeto de D(z), do que a do caso anterior, onde se supunha entrada constante, ou seja, apenas de frequência zero.

A partir de (5.3) com entrada senoidal tem-se

$$G(s) = \frac{1-e^{-Ts}}{s}\;D_2(e^{Ts})\;\frac{s^2+w^2}{w}\;\frac{e^{Ts}\ \text{sen}\ wT}{e^{2sT}-(2\cos wT)e^{Ts}+1}$$

$$G(jw) = \frac{1-e^{-jwT}}{jw}\;D_2(e^{jwT})\;\frac{e^{jwT}.\text{sen}wT.2jw}{w.j2Te^{jwT}\text{sen}(wt)}$$

$$G(jw) = \frac{1 - e^{-jwT}}{jwT} D_2(e^{jwt}) \qquad (5.9)$$

Portanto, fazendo $G(jw) = G_c(jw)$ tem-se o Controlador Digital

$$D_2(z) = \frac{\ln z}{1 - z^{-1}} G_c(\frac{1}{T} \ln z) \qquad (5.10)$$

Usando a Transformação Bilinear (5.7) resulta

$$D_2(z) = \frac{2z}{z + 1} G_c(\frac{2}{T} \frac{z - 1}{z + 1}) \qquad (5.11)$$

Comparando-se as expressões (5.8) e (5.11) pode-se notar a existência de um termo adicional em $D_2(z)$, que denominaremos termo compensatòrio $P(z)$,

$$P(z) = \frac{2z}{z + 1} \qquad (5.12)$$

Esse termo ocasionarà uma melhora no desem penho do controlador digital, como mostraremos pelos resultados de simulação do item 5.7; em contrapartida, estaremos aumentando a ordem do controlador digital, o que acarretarà em um aumento do esforço computacional exigido pelo controlador.

Muitas vezes o acrèscimo do polo $z=-1$ em (5.11) poderà acarretar um comportamento instàvel ao sistema realimentado, fato este que pode ser facilmente visualizado pelo mètodo do lugar das raízes. Nesse caso, pode-se tentar resolver o problema, alterando-se o termo compensatòrio $P(z)$, tornando-o da forma

$$P'(z) = \frac{(2-\delta)\, z}{z+1-\delta} \qquad (5.13)$$

com $0<\delta<0.5$. Esse fato serà tambèm ilustrado atravès de exemplos simulados.

5.4. DISTORÇÕES DA TRANSFORMAÇÃO BILINEAR

A resposta em frequência do sistema dado pela figura 5.2, seguindo o raciocínio do ìtem anterior, è dado por

$$G(jw) = \frac{1 - e^{-jwT}}{jwT} D(e^{jwT}) \qquad (5.14)$$

Aplicando os controladores (5.8) e (5.11), que fazem uso da transformação bilinear (5.7), tem-se respectivamente

$$G_1(jw) = \frac{1 - e^{-jwT}}{jwT} G_c\left(\frac{2}{T} \frac{e^{jwT}-1}{e^{jwT}+1}\right) \qquad (5.15)$$

$$G_2(jw) = \frac{1-e^{-jwT}}{jwT} \frac{2e^{jwT}}{1+e^{jwt}} G_c\left(\frac{2}{T} \frac{e^{jwT}-1}{e^{jwT}+1}\right) \qquad (5.16)$$

Estas expressões, com algum trabalho algèbrico, resultam em

$$G_1(jw) = \frac{\mathrm{sen}(wT/2)}{wT/2} e^{-jwT/2} G_c(j\Omega) \qquad (5.17)$$

$$G_2(jw) = \frac{\mathrm{tg}(wT/2)}{wT/2} G_c(j\Omega) \qquad (5.18)$$

onde

$$\Omega = \frac{2}{T} \mathrm{tg}(wT/2) \qquad (5.19)$$

Em relação ao valor desejado, $G_c(jw)$, as expressões (5.17) e (5.18) mostram a existência de dois tipos de distorção:

a) Em vez da frequência w, Gc toma valores em Ω, dada pela expressão (5.19). Note então que para valores de Ω variando de zero a infinito, a transformação bilinear comprime a variação de w de zero a $w_n = \pi/T$ (frequência de Nyquist), conforme ilustrado na figura

5.3. Esse fato è largamente comentado na literatura de Controle. Note que para $w << w_n$ tem-se $\Omega = w$, ocorrendo portanto pouca distorção desse tipo.

O fato de w estar limitada a π/T pode ser explicado facilmente: se $w>\pi/T$, pode-se fazer $w = 2k\pi/T + w'$ com $|w'|<\pi/T$ para algun valor inteiro de k. Então

$$e^{jwT} = e^{j(w'+2k\pi/T)T} = e^{jw'T} \quad ,$$

e portanto G_c com w e com w' terà o mesmo valor.

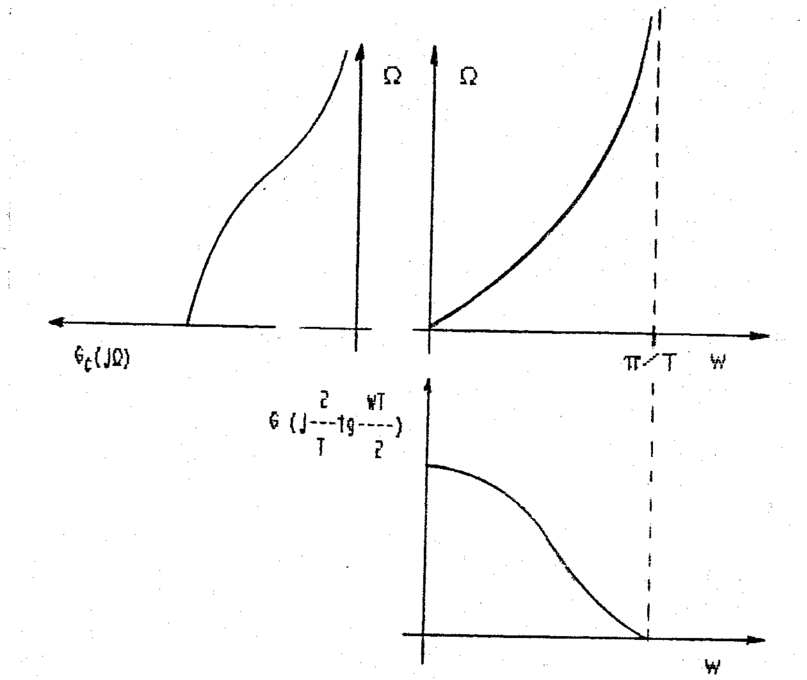


Fig.5.3.- Distorção de Frequência da relação (5.19)

b) Um outro tipo de distorção que agora se evidencia è devido aos termos multiplicativos a G_c. Comparando-se (5.17) e (5.18) pode-se verificar que em G_1, alèm de alteração no mòdulo, o termo multiplicativo tambèm acrescenta um atraso de fase ao sistema, o que não ocorre em G_2. Esse atraso de fase, em sistemas realimentados, contribui fortemente para deteriorar a estabilidade do sistema controlado. Por exemplo, para $wT/2 = 0,52$ (30^o), tem-se

$$\frac{\text{sen } wT/2}{wT/2} = 0.95 \quad \text{e} \quad \frac{\text{tg } wT/2}{wT/2} = 1.10 \quad ,$$

relativamente pròximos da unidade; porèm, um acrèscimo de fase de 30^o è inaceitavel em muitos casos, devido à diminuição da margem de fase do sitema.

5.5 . COMPENSAÇÃO DE FREQUENCIA ("prewarping")

Como vimos no ítem anterior, a transformação bilinear provoca uma distorção de frequência, comumente denominada de "frequency warping". É fàcil introduzir uma modificação que elimina essa distorção em uma frequência específica w_i. Se em vez de (5.7) usarmos a transformação alternativa dada por

$$s = \frac{w_1}{tg(w_1T/2)} \frac{z-1}{z+1} \tag{5.20}$$

As expressões correspondentes a (5.8) e (5.11) serão agora dadas por

$$D_1'(z) = Gc(\frac{w_1}{tg(w_1T/2)} \frac{z-1}{z+1}) \tag{5.21}$$

$$D_2'(z) = \frac{w_1T}{tg(w_1T/2)} \frac{z}{z+1} Gc(\frac{w_1}{tg(w_1T/2)} \frac{z-1}{z+1}) \tag{5.22}$$

Usando-se esses controladores em (5.14) resulta

$$G_1'(jw) = \frac{sen(wT/2)}{wT/2} e^{-jwT/2} G_c(j\frac{w_1}{tg(w_1T/2)} tg(wT/2))$$

$$G_2'(jw) = \frac{w_1}{w} \frac{tg(wT/2)}{tg(w_1T/2)} G_c(j\frac{w_1}{tg(w_1T/2)} tg(wT/2))$$

Para a frequência w_1 tem-se então

$$G_1'(w_1) = \frac{sen(w_1T/2)}{w_1T/2} e^{-jw_1T/2} \quad G_c(w_1) \tag{5.23}$$

$$G_2'(w_1) = G_c(jw_1) \tag{5.24}$$

Observe que a transformação (5.20) corrige todo tipo de distorção na frequência w_1 para G'_2, restando ainda em $G'_1(w_1)$ distorções devido ao termo multiplicativo, tanto em módulo, como principalmente em fase. Dessa forma fica evidenciada a superioridade dos controladores D_2 e D'_2, que apresentam o termo compensatório P(z) (ou P (z)), sobre os controladores D_1 e D'_1, largamente citados na literatura de Controle Digital.

No controle realimentado a frequência w_1 deve ser escolhida de modo a assegurar o mesmo tipo de comportamento transitório ao sistema controlado. Pode-se escolher aquela em que o módulo da função de transferênciade malha aberta vale a unidade (0 dB), onde se mede a margem de fase, e que está diretamente ligada à estabilidade do sistema.

5.6. ESCOLHA DO PERÍODO DE AMOSTRAGEM

As expressões (5.17) e (5.18) indicam ainda como escolher períodos de amostragem adequados para os respectivos controladores digitais. Isso será feito de modo que as distorções dos termos multiplicativos sejam suficientemente pequenas para as frequências determinantes do comportamento do sistema. Pode-se usar, por exemplo, como critério de projeto do período de amostragem, uma distorção máxima de módulo de 10% e uma distorção máxima de fase de 6^{o} para a frequência w_1 descrita no ítem 6. Nesse caso, para o controlador D_1 tem-se

$$\frac{\text{sen}(w_1T/2)}{w_1T/2} > 0.9 \longrightarrow T<1.4/w_1$$

$$\Longrightarrow \quad T<0.2/w_1$$

$$w_1T/2 < 0.1047 \longrightarrow T<0.2/w_1$$

Para o controlador D_2 tem-se

$$\frac{\text{tg}(\ w_1T/2\ )}{w_1T/2} < 1.10 \qquad \Longrightarrow \qquad T<1/w_1$$

Note então que o controlador D_2, por não apresentar distorção de fase, permite, neste caso, o uso de período de amostragem 5 vezes maior que o controlador D_1. Esse resultado mostra que o aumento do esforço computacional ocasionado pelo termo compensatório P(z), e que exigiria um maior período de amostragem para o controlador D_2, não traz prejuízo real para o desempenho deste controlador.

5.7. RESULTADOS DE SIMULAÇÃO

Apresentaremos a seguir dois exemplos citados na literatura de Controle Digital para o controlador D_1, e os aplicaremos para os controladores D'_1, D_2 e D'_2.

O primeiro exemplo, mostrado pela figura 5.4, è apresentado em P. Katz, "Digital Control of Dynamic Systems", 1980, pp. 53-127, onde o controlador è projetado de forma a se ter para o sistema controlado:

a) Màximo "Overshoot" = 42%

b) Margem de ganho = 25 dB

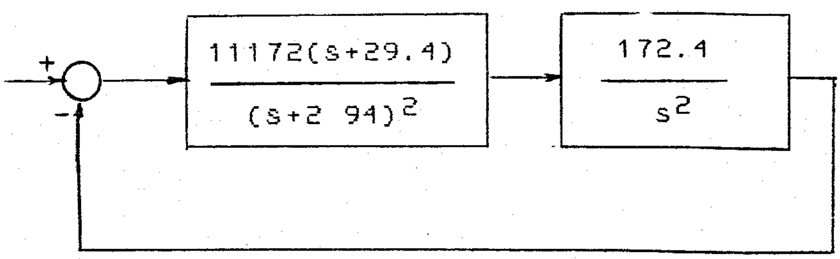


Fig.5.4 - Sistema Servoposicionador

Para $T = 0,01s$ e $w_1 = 30$ rd/s obteve-se:

a) Com $D_1(z)$ e $D'_1(z)$ ---> M.O. = 59% e M.G. = 14,5 dB

b) Com $D_2(z)$ e $D'_2(z)$ ---> M.O. = 45% e M.G. = 18.0 dB

O segundo exemplo, mostrado pela figura 5.5, encontra-se em G.F. Franklin; J.D.Powell, "Digital Control of Dinamical Systems", 1980, pp.38-68, onde o controlador è projetado de forma a se ter para o sistema controlado

a) Màximo "Overshoot"= 16.3% (ξ = 0.5)

b) Tempo de Estabilização (2%) = T_e = 8s

($T_e = 4/\xi w_n$)

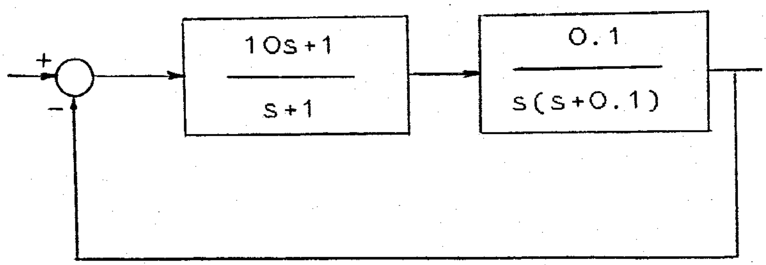


Fig.5.5 - Sistema de Controle Contìnuo

A utilização dos controladores (5.11) e (5.22) para essa planta resulta em sistemas instàveis, como mostra o lugar das raìzes da figura 5.6.

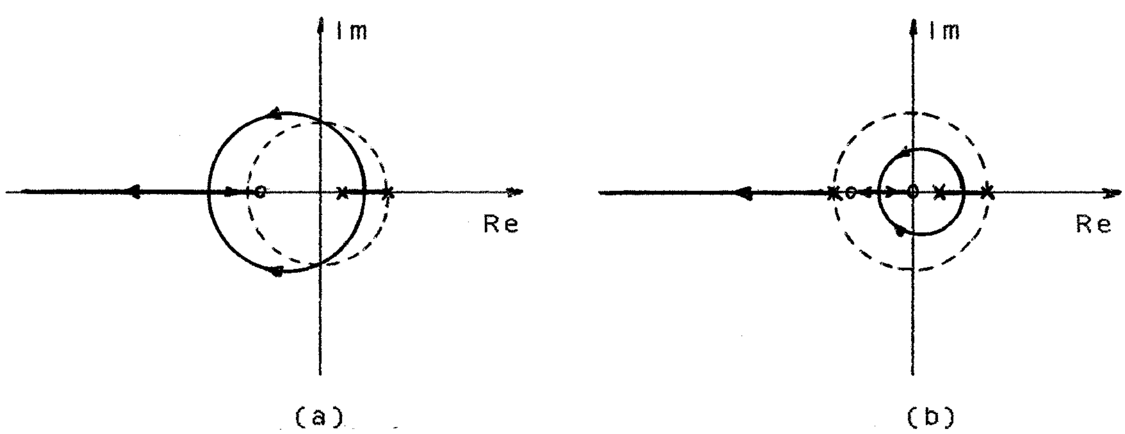


Fig.5.6 - Lugar das raìzes do segundo exemplo
a) Com o controlador (5.8)
b) Com o controlador (5.11)

Note na figura 5.6.b que hà um ramo sempre instàvel no lugar das raîzes.

Utilizando-se o controlador $D_2(z)$, mas com o termo compensatòrio (5.13), resulta o novo lugar das raîzes mostrado pela figura 5.7.

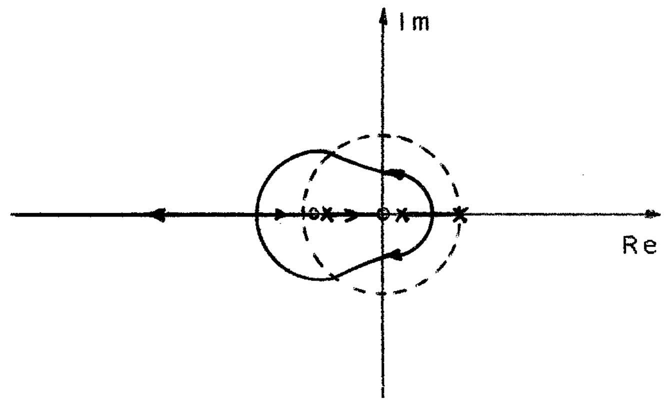


Fig. 5.7 - Lugar das raîzes do segundo exemplo com Termo Compensatòrio (5.13).

Note a estabilização devida à troca de posições de polos e zeros e à eliminação do polo em -1.

Com $T = 1s$, $w_1 = 0,786$ rd/s e $\delta = 0,3$, obteve-se os resultados de simulação:

$D_1(z)$ ----> M.O. = 44.0%, T_e = 16s

$D_2(z)$ ----> M.O. = 21.0%, T_e = 8s

$D'_1(z)$ ----> M.O. = 41.8%, T_e = 16s

$D'_2(z)$ ----> M.O. = 18.2%, T_e = 8s

Estes dois exemplos realmente mostram um melhor desempenho dos controladores D_2 e D'_2 em relação aos precedentes D_1 e D'_1.

5.8. O CONVERSOR D/A FICTÍCIO DE ORDEM UM

Até agora, o conversor digital analógico usado tem sido o tradicional "segurador de ordem zero", para o qual a relação entrada-saída é da forma

$$y_a(t) = y_a(KT) = y_K \,, \quad KT \leq t < (K+1)T \qquad (5.25)$$

Neste item estudaremos um conversor D/A de ordem um, fictício, não realizável por ser não causal, para o qual tem-se

$$y_{a1}(t) = \frac{y_{K+1}-y_K}{T}\, t + y_K - K(y_{K+1}-y_K) \quad , \; KT \leq t < (k+1)T \qquad (5.26)$$

Observe que $y_{a1}(t)$ está entre y_K e y_{K+1} para $KT \leq t < (K+1)T$ e que $y_{a1}(KT) = y_K$ (vide figura 5.8).

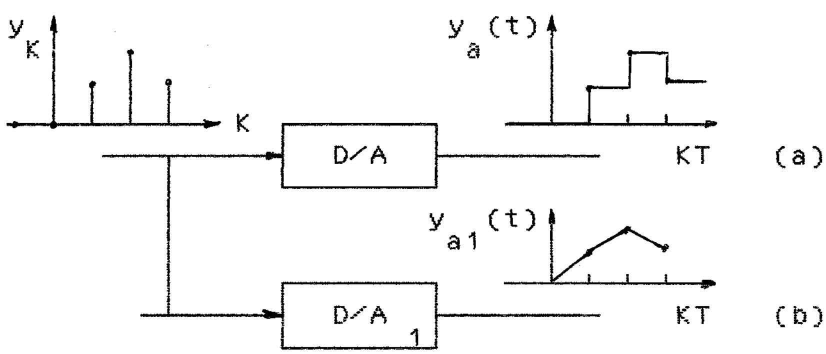


Fig.5.8. Conversores D/A de ordem zero (a) e de ordem um (b)

Note que a expressão (5.26) não é realizável, pois para se determinar $y_{a1}(t)$, é necessário conhecer y_{K+1}, ocorrido no instante $(K+1)T$, à frente de t.

Considere o esquema da figura 5.9, à semelhança daquele da figura 5.2, mas com conversor D/A de ordem um

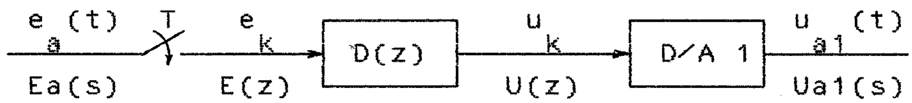


Fig.5.9 - Controlador digital com conversor D/A de ordem um.

Em relação à Ua1(s) tem-se:

$$Ua1(s) = \int_0^{\infty} u_{a1}(t)e^{-st}dt$$

$$= \sum_{K=0}^{\infty} \left[\frac{u_{K+1}-u_K}{T} \int_{KT}^{(K+1)T} te^{-st}dt + [u_K - K(u_{K+1}-u_K)] \int_{KT}^{(K+1)T} e^{-st}dt \right]$$

$$= \frac{1}{Ts^2} \left[(1-e^{-Ts}) \sum_{K=0}^{\infty} (u_{K+1}-u_K)e^{-KTs} - Ts \sum_{K=0}^{\infty} (u_{K+1}e^{-(K+1)Ts} - u_K e^{-KTs}) \right]$$

Note que a última somatória reduz-se a u_0, que deverá ser sempre nulo no algoritmo do controlador D(z). Dessa forma tem-se

$$Ua1(s) = \frac{1-e^{-Ts}}{Ts^2} \; Z[u_{K+1} - u_k] \Big|_{z=e^{Ts}} \qquad (5.27)$$

$$= \frac{1-e^{-Ts}}{Ts^2} \; (z-1)U(z) \Big|_{z=e^{Ts}} \qquad (5.28)$$

ou

$$Ua1(s) = \frac{e^{Ts}+e^{-Ts}-2}{Ts^2} \; D(z)E(z) \Big|_{z=e^{Ts}} \qquad (5.29)$$

A relação entre Ua1(s) e Ea(s) deverá ser dada então por (compare com a expressão (5.3) do conversor de ordem zero):

$$G_1(s) = \frac{U_{a1}(s)}{E_a(s)} = \frac{e^{Ts}+e^{-Ts}-2}{Ts^2} \frac{1}{E_a(s)} D(z)E(z)\Big|_{z=e^{Ts}} \qquad (5.30)$$

Para entrada senoidal, seguindo paralelamente ao item 5.3, tem-se a resposta em frequência do esquema proposto,

$$G_1(jw) = \frac{e^{jwT}+e^{-jwT}-2}{(jwT)^2} D(z)\Big|_{z=e^{jwT}} \qquad (5.31)$$

Expandindo as exponenciais em série de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \ldots + \frac{x^n}{n!} + \ldots$$

tem-se:

$$G_1(jw) = [1-\frac{(wT)^2}{12}+\frac{(wT)^4}{360}+\ldots+\frac{2(-1)^m(wT)^{2m}}{(2m+2)!}+\ldots]D(e^{jwT}) \qquad (5.32)$$

Comparando-se essa expressão com a (5.14), equivalente para o conversor D/A de ordem zero, pode-se perceber que neste caso o termo multiplicativo a $D(e^{jwT})$ tem menos influência do que naquele, uma vez que apresenta fase nula, e que o módulo se aproxima muito mais rapidamente da unidade com a diminuição de T. Pode-se , portanto, com mais tranquilidade, projetar o controlador D(z) a partir da resposta em frequência desejada Gc(jw), da forma

$$D(z) = Gc(\frac{2}{T}\frac{z-1}{z+1}) \quad , \qquad (5.33)$$

ou seja, desprezar em (5.32) o termo multiplicativo, e usar a Transformação Bilinear (5.7). É claro que só obteremos bons resultados se T for projetado de tal modo que para as frequências de trabalho se tenha

$$wT \leq 1 \quad , \qquad (5.34)$$

caso em que tanto a aproximação do termo multiplicativo de (5.32) pela unidade, como a Transformação Bilinear, apresentam boa precisão.

5.9. PROJETO VIA RESPOSTA EM FREQUENCIA DO SISTEMA REDESENHADO

Os resultados acima permitem utilizar o conversor D/A de ordem um, não realizável, de modo fictício, para projetar controladores digitais

A figura 5.7 ilustra os passos de projeto, que apresentamos a seguir:

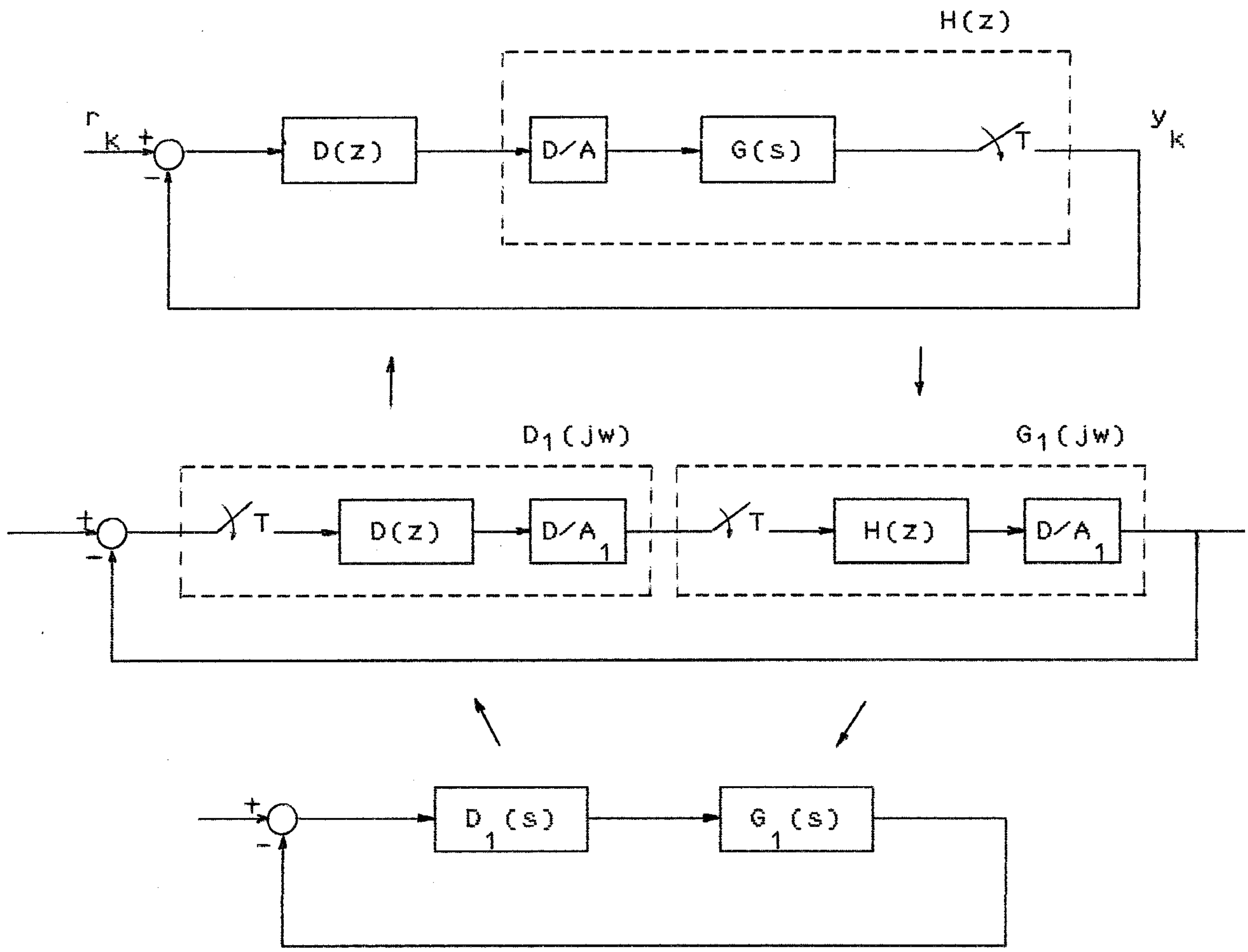


Fig. 5.10 - Ilustração dos passos de projeto com redesenho do sistema controlado

1. Obtenha a função de transferência discreta H(z) da planta mais os conversores A/D e D/A (de ordem zero),

$$H(z) = (1-z^{-1})\ Z[G(s)/s)]$$

2. Introduzindo, ficticiamente, conversores A/D e D/A de ordem 1, que não afetam o desempenho do sistema discreto, determine a função de transferência $G_1(jw)$ equivalente à planta e aos conversores, utilizando a Transformação Bilinear, ou seja,

$$G_1(jw) = H(z)\Big|_{z=\frac{1+jwt/2}{1-jwT/2}} \qquad (5.35)$$

3. Projete o controlador contínuo $D_1(jw)$ por meio de tècnicas tradicionais de controle contínuo, como resposta em frequência ou lugar das raízes.

4. Obtenha o controlador digital $D(z)$ aplicando a transformação bilinear a $D_1(jw)$,

$$D(z) = D_1(jw)\Big|_{jw=\frac{2}{T}\frac{z-1}{z+1}} \qquad (3.36)$$

<u>EXEMPLO</u>: Projetar um controlador digital para a planta dada por

$$G(s) = \frac{0{,}10}{s(s+0.1)}$$

de modo que se tenha: a) $K_v = 1{,}0$
b) margem de fase $\cong 50^0$ $(\xi \cong 0{,}5)$

Resolução: A partir de G(s), para T=1,0 s , tem-se

$$H(z) = 0{,}048\ \frac{(z+0{,}967)}{(z-1)(z-0{,}905)}$$

e

$$G_1(s) = -\frac{(s/120 + 1)(s/2 - 1)}{s(s/0.0999 + 1)}$$

Projetando-se um compensador a fim de satisfazer as especificações desejadas, resulta

$$D_1(s) = \frac{s/0.0999 + 1}{s/6 + 1}$$

A figura 5.11 abaixo mostra os gráficos de Bode da função de transferência de malha aberta $D_1G_1(jw)$, para os casos sem compensação ($D_1=1$), e com o compensador D_1 acima.

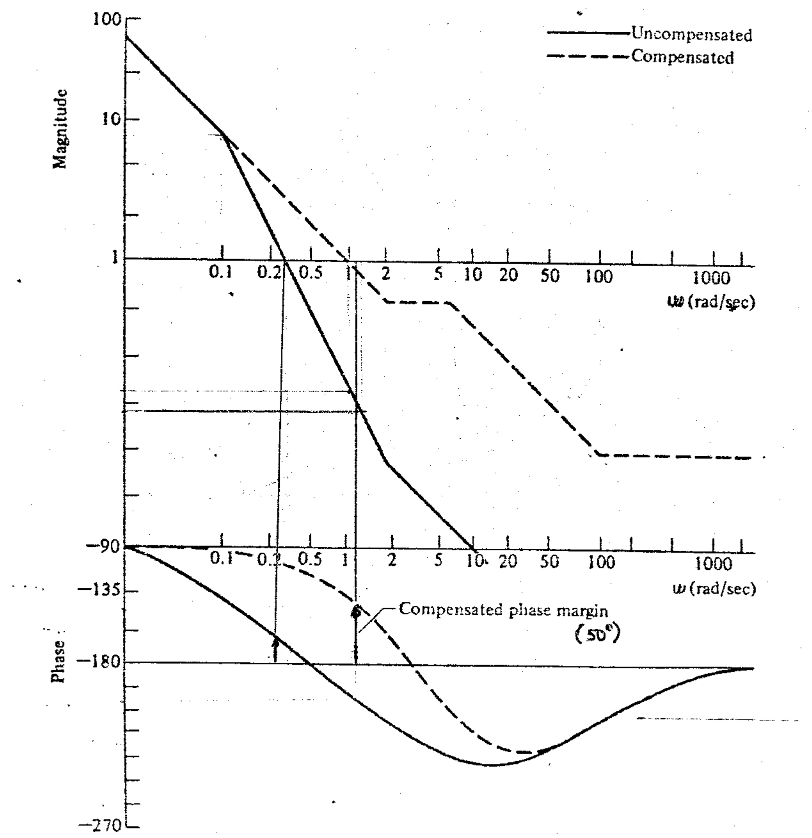


Fig.5.11 - Gráficos de módulo e fase de $D_1G_1(jw)$

A partir de $D_1(z)$ resulta

$$D(z) = \frac{15.8(z - 0.905)}{z + 0,5}$$

Note que a frequência em que o módulo vale zero dB vale $w_c = 1$. Portanto $w_cT=1$, obedecendo-se a relação 5.34, podendo-se então esperar um bom desempenho do sistema controlado.

5.10. O CONTROLADOR PID DIGITAL

Um dos controladores contínuos mais amplamente usados é o controlador proporcional-intregral-derivativo (PID), cuja lei de controle u(t) é dada em função do sinal erro e(t) por

$$u(t) = K.e(t) + K_i . \int_{-\infty}^{t} e(t)dt + K_d . \frac{de}{dt}$$

ou (5.37)

$$U(s) = (k + \frac{K_i}{s} + K_d s)\ E(s)$$

A função do controle integrativo é a de reduzir erros de regime estacionário (se o erro de regime tender a permanecer constante, o sinal de controle aumentará a fim de diminuí-lo). A função do controle derivativo é a de melhorar o transitório (erros mesmo pequenos, mas de crescimento rápido, causam uma ação de controle que tende a corrigí-los de modo antecipado, antes que se tornem grandes, causando consequentemente uma ação estabilizante.

A parte integrativa do controlador pode ser obtida através de aproximação de soma de trapézios, da forma:

$$u_i(KT) = K_i \int_{-\infty}^{KT} e(\tau)d\tau \cong K_i [\int_{-\infty}^{(K-1)T} e(\tau)d\tau + T\frac{e(kT)+e((K-1)T)}{2}] \quad (5.38)$$

ou $$U_i(z) \cong z^{-1}U_i(z) + K_i \frac{T}{2}(1 + z^{-1})E(z) \quad (5.39)$$

o que resulta

$$U_i(z) = K_i \frac{T}{2} \frac{1 + z^{-1}}{1 - z^{-1}} E(z) \quad (5.40)$$

Comparando (5.40) com a parte integrativa de (5.37), pode-se ver que o procedimento adotado equivale à aplicação da Transformação Bilinear (5.7).

A parte derivativa pode ser facilmente aproximada por

$$u_d(KT) \cong K_d \frac{\Delta e}{\Delta t} = K_d \frac{e(KT)-e((K-1)T)}{T} \quad (5.41)$$

ou
$$U_d(z) = \frac{K}{T}\,\frac{z-1}{z}\,E(z) \tag{5.42}$$

A partir de (5.34), (5.40) e 5.42) pode-se descrever o controlador PID digital atravès da função de transferência

$$D(z) = \frac{U(z)}{E(z)} = K + \frac{K_iT}{2}\,\frac{z+1}{z-1} + \frac{K_d}{T}\,\frac{z-1}{z} \tag{5.43}$$

ou

$$D(z) = \frac{[k_iT/2 + K_d/T + K]z^2 + [k_iT/2 - k - 2K_d/T]z + k_d/T}{z\,(z-1)} \tag{5.44}$$

5.11. ALGORITMO PID DE VELOCIDADE

Muitas vezes a lei de controle acima è realizada de outra forma: em vez de se obter $u(t)$ em cada passo, gera-se a diferença $\Delta u(KT)=u(kT)-u((K-1)T)$. Implementado dessa forma, o algoritmo PID è chamado Algoritmo de Velocidade. E fàcil verificar que

$$\frac{\Delta U(z)}{E(z)} = \left(\frac{K_iT}{2} + \frac{K_d}{T} + K\right) + \left(\frac{K_iT}{2} - K - 2\frac{K_d}{T}\right)z^{-1} + \frac{K_d}{T}\,z^{-2} \tag{5.45}$$

Por exemplo, suponha um motor de passo conectado a uma vàlvula; para posicionar o motor basta que se dê o nùmero de passos em relação à posição anterior.

5.12. PROTEÇÃO CONTRA MUDANÇAS RAPIDAS NA REFERENCIA

O erro $e(t)$ è dado por $e(t)=r(t)-y(t)$, onde $r(t)$ è o sinal de referência ("set-point") e $y(t)$ è a saida da planta. Se $r(t)$ è variado bruscamente, a derivada de/dt terà um valor instantâneo muito alto, acarretando uma elevação sùbita no valor do sinal de controle $u(t)$, o que pode ser indesejàvel, podendo causar por exemplo efeitos do tipo "solavanco". Supondo que em operações normais o sinal de referência $r(t)$ varie lentamente, pode-se aproximar a derivada por

$$\frac{de}{dt} \cong - \frac{y(KT)-y((k-1)T)}{T} \tag{5.46}$$

Logo, o algoritmo de controle digital serà dado por

$$U(z) = \frac{[k_i T/2 + K]z + [k_i T/2 - k]}{(z - 1)} E(z) - \frac{k_d}{T} \frac{z-1}{z} Y(z) \tag{5.47}$$

Desse modo, se $r(t)$ for variado bruscamente, a pròpria dinâmica da planta impedirà que de/dt atinja valores elevados, acarretando uma lei de controle mais suave. Essa "suavização" tambèm pode ser efetuada no Algoritmo PID de Velocidade.

CAPÌTULO 6 - PROJETO DE CONTROLADORES DIGITAIS NO PLANO Z

Este capìtulo trata do projeto de controladores digitais para sistemas lineares monovariàveis invariantes no tempo, diretamente no plano z. Neste caso não se pressupõe a exìstencia de controladores projetados no no plano s, e portanto não se pretende fazer nenhuma aproximação de controladores contìnuos, como no capìtulo anterior. Este fato permitirà o uso de perìodos de amostragem maiores, uma vez que o comportamento do sistema serà determinado diretamente no plano z. Serão desenvolvidos tambèm algoritmos de controle que não tem similar no mundo contìnuo, como os controladores "dead-beat", abrindo portanto um novo panorama para a àrea de controle de processos.

6.1. PROJETOS QUE USAM LUGAR DAS RAÌZES NO PLANO Z

Considere o sistema discreto apresentado pela figura 6.1 abaixo:

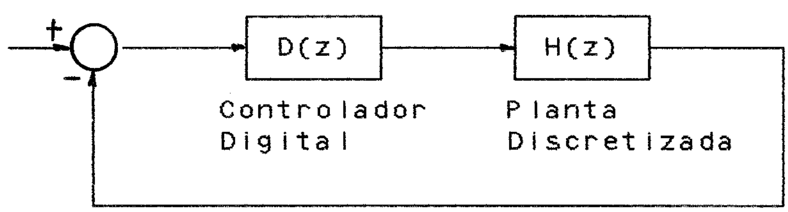


Fig. 6.1 - Sistema de Controle Digital

O lugar das raìzes desse sistema è obtido a partir da equação caracterìstica

$$1 + D(z)H(z) = 0 \tag{6.1}$$

Como essa equação è idêntica à equação caracterìstica de sistemas contìnuos (no plano s), resulta o fato que a construção do lugar das raìzes no plano z, è idêntica à do plano s . O que muda è a interpretação dos gràficos obtidos, onde agora o cìrculo unitàrio è a região de estabilidade (ver figuras 4.7 e 4.9).

Como em sistemas contìnuos, o projeto de controladores usando lugar das raìzes visarà bàsicamente relocação de polos, onde os polos desejados para o sistema controlado são obtidos a partir das especificações de desempenho pretendidas. A seguir, vàrios exemplos ilustrarão esta metodologia: as estruturas de controladores PID, de Avanço de Fase e de Atraso de Fase, serão usadas, mas com o projeto feito diretamente no plano z.

1º Exemplo: CONTROLADORES PI E PID

Considere um sistema contìnuo descrito pela função de transferência

$$G(s) = \frac{10}{(s+1)(s+2)} \tag{6.2}$$

Usando (4.14) com T=0.1s (5 vezes menor que a menor constante de tempo do sistema) tem-se

$$H(z) = \frac{0.0453(z+0.904)}{(z-0.905)(z-0.819)} \tag{6.3}$$

Para $D(z)=K$, o lugar das raìzes do sistema controlado è mostrado pela figura 6.2. Para $k=1$, obtem-se os polos

$$0,84 \pm j0,28 \quad ,$$

que correspondem a um coeficiente de amortecimento $\xi = 0.35$ (conseguido atravès do àbaco da figura 4.7).

A figura 6.4 apresenta a resposta do sistema controlado com $D(z)=K=1$ para entrada degrau unitàrio. Note a presença do erro de regime, dado por

$$e_\infty = \frac{1}{1 + k_p} = \frac{1}{1 + D(1)H(1)} = 0.163 \tag{6.4}$$

Supondo que não queremos modificar substancialmente a resposta transitòria, mas apenas anular o erro de regime, usaremos um controlador PI, da forma (a partir de (5.44))

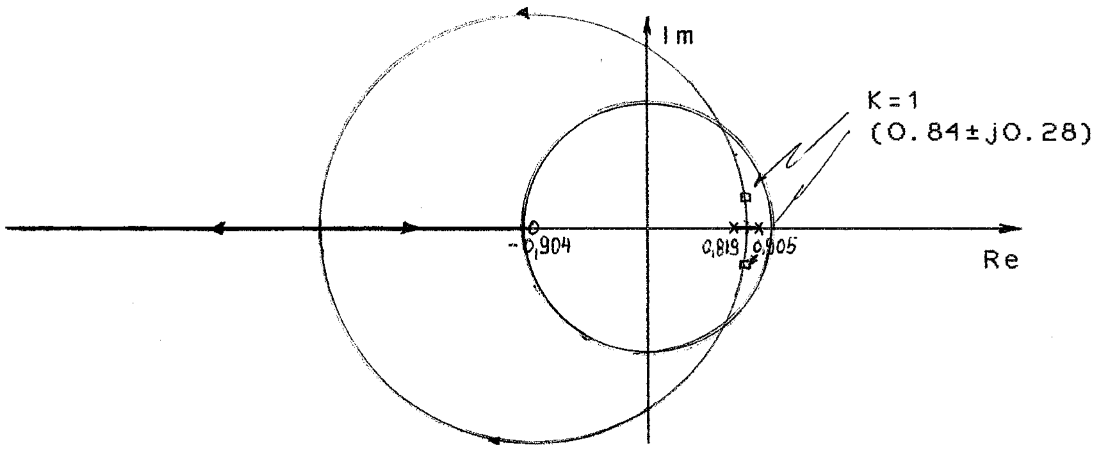


Fig. 6.2 - Lugar das Raízes com Controle Proporcional

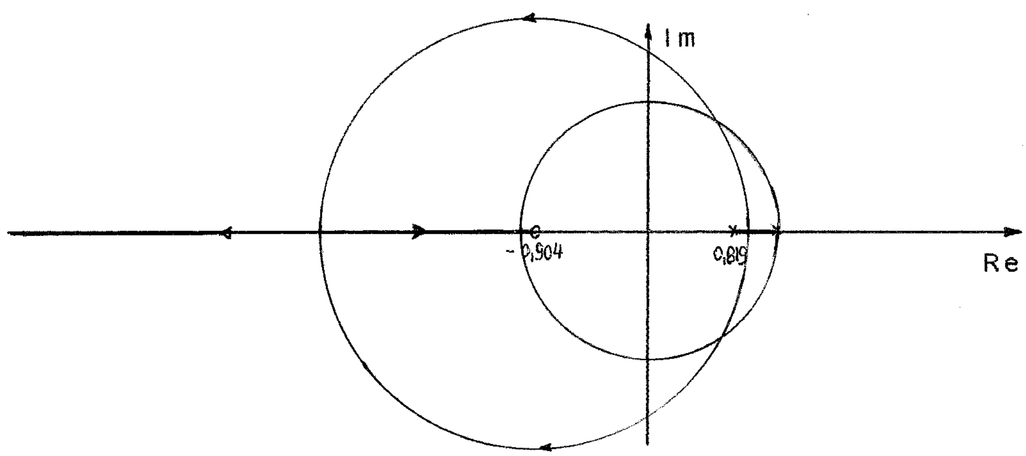


Fig. 6.3. - Lugar das Raízes com Controle PI

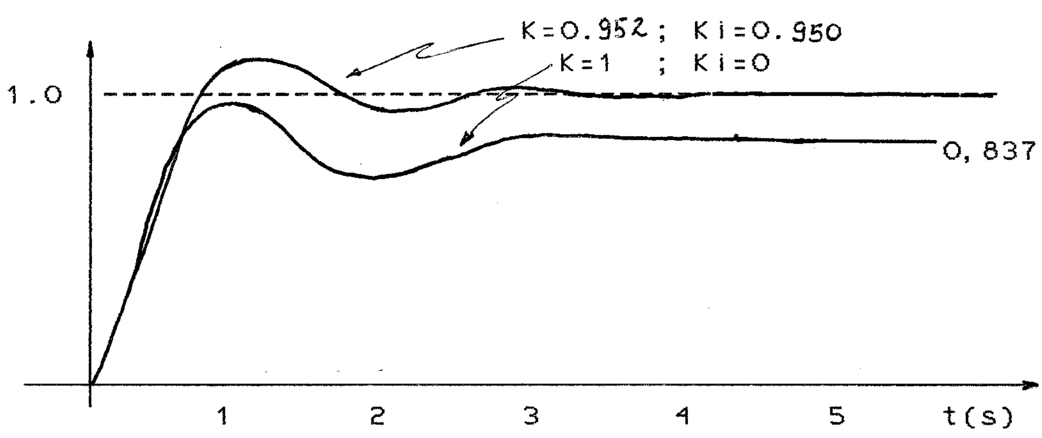


Fig. 6.4 - Resposta ao degrau com controle P e PI

$$D(z) = \frac{(k+k_iT/2)(z - \frac{K-K_iT/2}{K+K_iT/2})}{(z - 1)} \qquad (6.5)$$

Visando substituir o polo z=0,905 planta pelo polo z=1 (lei integrativa), para anular o erro de regime, o zero do controlador deverá cancelar o polo z=0,905, isto é,

$$\frac{k - k_iT/2}{K + K_iT/2} = 0.905 \quad ===> \quad \frac{K}{K_i} = 1.0026 \qquad (6.6)$$

Mantendo essa relação fixa, a figura 6.3 apresenta o lugar das raízes do sistema controlado em função do ganho $k' = K+k_iT/2$ (ver fig.6.5).

Note que o lugar geomètrico pouco se alterou. Para que se obtenha a mesma resposta transitória, deve-se fazer o ganho k' unitário, ou seja,

$$K + K_iT/2 = 1 \qquad (6.7)$$

A partir de (6.6) e (6.7) obtem-se k=0.952 e K_i=0.950. A figura 6.4 apresenta a resposta ao degrau do sistema controlado com esses valores de K e K_i.

A fim de também melhorar o comportamento transitório pode-se usar a lei de controle PID (5.44),

$$D(z) = \frac{(\frac{K_iT}{2}+\frac{K_d}{T}+K)(z^2+\frac{K_iT^2-2KT+4K_d}{2KT+K_iT^2+2K_d}\,z+\frac{2K_d}{2KT+K_iT^2+ 2K_d})}{z(z - 1)} \qquad (6.8)$$

Note que agora o compensador introduz um polo na origem, outro polo e z=1 e dois zeros. Para melhorar o transitório temos que "puxar" os polos de malha fechada mais para a esquerda (aumentar a velocidade) e mais para baixo (aumentar o amortecimento). Para isso, cancelaremos os polos da planta com os zeros do compensador, o que resultará no lugar das raízes da figura 6.5. Faz-se então

$$z^2+\frac{K_iT^2-2KT+4K_d}{2KT+K_iT^2+2K_d}.z+\frac{2K_d}{2KT+K_iT^2+ 2K_d} = (z-0,905)(z-0,819) \qquad (6.9)$$

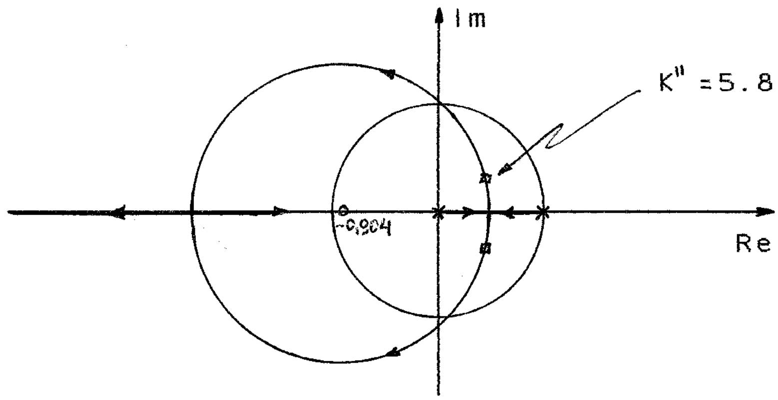


Fig 6.5 - Lugar das raízes com ação PID

Note em (6.8) que o lugar das raízes agora está em função de K dado por

$$K'' = \frac{K_i T}{2} + \frac{K_d}{T} + K \qquad (6.10)$$

Para $k'' = 5.8$ tem-se os polos de malha fechada $0.369 \pm j0.319$, correspondentes a $\xi=0.707$ (menor erro quadrático médio em relação à entrada degrau - máximo overshoot = 4%).

As equações (6.9) e (6.10), com $k''=5.8$, constituem um sistema de equações algébricas de tres equações e tres incógnitas, que resolvido resulta

$$\begin{aligned} K &= 1.45 \\ K_i &= 1.0 \\ K_d &= 0.43 \end{aligned} \qquad (6.11)$$

A figura 6.6 abaixo apresenta a resposta ao degrau do sistema com o controlador PID que utiliza esses valores.

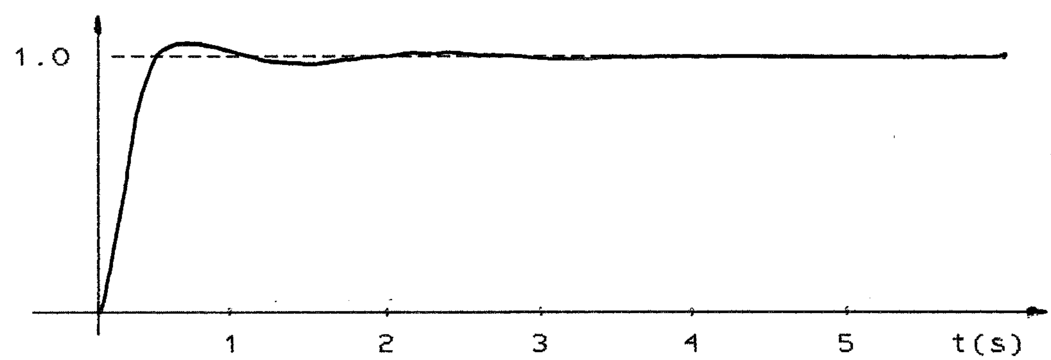


Fig 6.6 Resposta ao degrau com controlador PID

2º Exemplo: CONTROLADOR DE ATRASO DE FASE

Projetar um controlador de atraso de fase digital para a planta (com T=0.5s) dada por

$$H(z) = \frac{0.1K(z+1.31)(z+0.054)}{z(z-1)(z-0.368)} \qquad (6.12)$$

de modo que se tenha a) $K_V = 3.0$
b) polos dominantes com ξ=0.707

Solução: sem compensação, apenas pelo ajuste de K, portanto com D(z) = 1, o sistema tem o lugar das raízes mostrado pela figura 6.7.

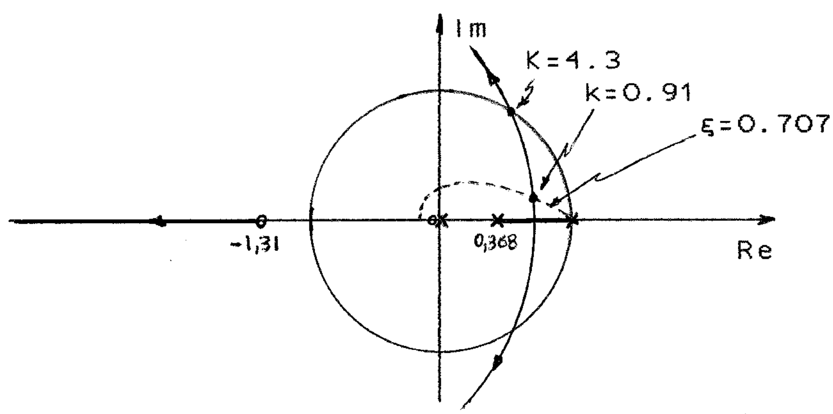


Fig 6.7 - Lugar das raízes do sistema (6.12) sem compensação

Para os polos dominantes com ξ=0,707 tem-se k=0.91. Porèm, com esse valor de K resulta

$$K_V = \lim_{z\to 1} \frac{z-1}{T} \frac{0.1\times 0.91(z+1.31)(z+0.054)}{z(z-1)(z-0.368)} = 0.7 < 3.0 \qquad (6.13)$$

Como ambas as especificações não podem ser satisfeitas apenas com o ajuste de k, algum tipo de compensação deve ser usada. Tentaremos um controlador de atraso de fase, do tipo

$$D(z) = K_c \frac{z - z_1}{z - p_1} \quad , \quad \text{com} \quad K_c = \frac{1-p_1}{1-z_1} \qquad (6.14)$$

onde $z_1 < p_1$ e muito pròximos de $z=1$. Essa atitude implica em que $D(1)=1$ (não altera K_v) e em que o lugar das raìzes quase não è modificado, como mostra a figura 6.8.

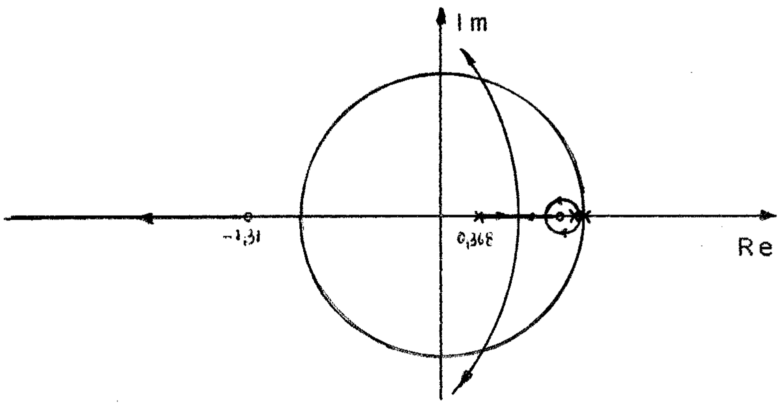


Fig 6.8 - Lugar das raìzes do sistema (6.12)
com o compensador (6.14)

O valor de K deve ser escolhido de modo a estabelecer a especificação de regime, $k_v=3.0$, isto è

$$K_v = \overbrace{\lim_{z-1} D(z)H(z)}^{\frac{z-1}{T}} = 3.0 \quad ====> \quad K=3.9 \qquad (6.15)$$

Como o novo lugar das raìzes està agora em função de $k.k_c$, para se ter os polos dominantes pròximos de $\xi=0.707$ deve-se ter

$$K.K_c = 0.91$$

ou,

$$K_c = 0.233 \qquad (6.16)$$

Arbitrando-se $p_1=0.99$ (pròximo de 1), a partir de (6.14) tem-se $z_1=0.957$.

Portanto o controlador $D(z)$ serà dado por

$$D(z) = 0.233 \frac{z - 0.957}{z - 0.99} \qquad (6.17)$$

6.2. PROJETO DE CONTROLADORES DE IMPOSIÇÃO DE POLOS E ZEROS EM ABORDAGEM ALGEBRICA

No ìtem anterior utilizou-se uma metodologia tradicional, usando o conceito de Lugar das Raìzes para impor polos ao sistema controlado. Essa tècnica consiste basicamente em aplicar mètodos de tentativa e erro, não tendo o projetista uma idèia clara de suas possibilidades. Neste ìtem, utilizando uma abortdagem inteiramente algèbrica, faremos imposição de polos e zeros de forma mais sistemàtica, e com uma visão mais fechada das possibilidades.

Considere a planta discretizada dada por

$$y_K = \frac{z^{-d}B(z^{-1})}{A(z^{-1})} u_K \tag{6.18}$$

onde y_K = saida

u_K = entrada

d = atraso puro

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \ldots + a_{na} z^{-na} \\ B(z^{-1}) &= b_o + b_1 z^{-1} + \ldots + b_{nb} z^{-nb} \quad , \; b_o \neq 0 \end{aligned} \tag{6.19}$$

A figura 6.9 abaixo mostra o esquema de controle que deverà ser usado, onde pode-se ver um bloco $H(z^{-1})$ de prè-alimentação ("feed-forward") e um bloco $G(z^{-1})$ de realimentação ("feed-back"),

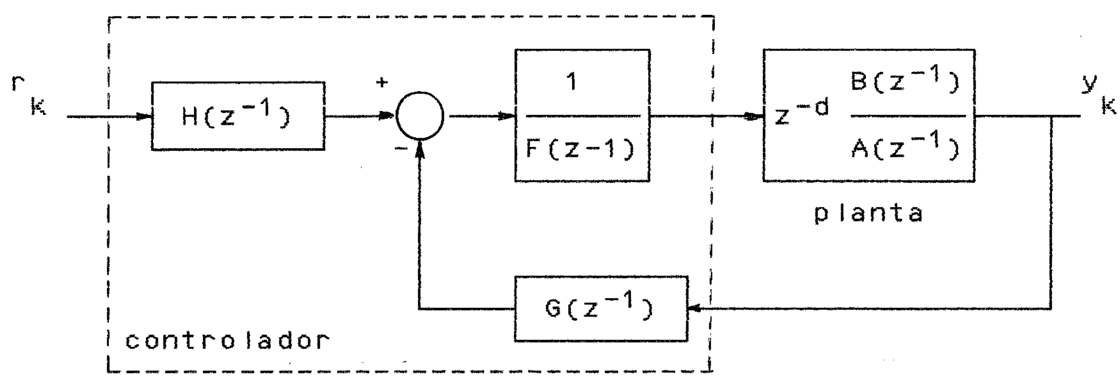


Fig. 6.9 - Controle com realimentação e prè-alimentação

onde: r_K = entrada de referência

$$\begin{aligned} H(z^{-1}) &= h_o + h_1 z^{-1} + \ldots + h_{nh} z^{-nh} \\ G(z^{-1}) &= g_o + g_1 z^{-1} + \ldots + g_{ng} z^{-ng} \\ F(z^{-1}) &= f_o + f_1 z^{-1} + \ldots + f_{nf} z^{-nf} \quad , \ f_o \neq 0 \end{aligned} \tag{6.20}$$

Tem-se então em malha fechada

$$y_K = \frac{z^{-d}\ B.H}{AF + z^{-d}BG}\, r_K \tag{6.21}$$

O objetivo deste projeto é impor ao sistema controlado, dado por (6.21), novos polos e novos zeros. Para isso, suponha o polinômio $T(z^{-1})$, estável (todas as raízes localizadas no interior do círculo unitário), da forma

$$T(z^{-1}) = t_o + t_1 z^{-1} + \ldots + t_{nt} z^{-nt} \quad , \ t_o \neq 0 \tag{6.22}$$

Projetando-se os polinômios F e G tais que

$$AF + z^{-d}BG = T \quad , \tag{6.23}$$

tem-se para o sistema em malha fechada,

$$y_K = \frac{z^{-d}\ B.H}{T}\, r_K \tag{6.24}$$

Podemos ver de imediato , a partir de (6.24), que o atraso puro d permanece no sistema controlado, caso contrário a lei de controle seria não causal, e portanto irrealizável.

É importante dizer que a equação (6.23) tem sempre solução de grau mínimo para

$$\begin{aligned} n_f &= n_b + d - 1 \\ \text{e} \qquad n_g &= \max[n_a - 1 \ , \ n_t - n_b - d] \end{aligned} \tag{6.30}$$

a) Plantas de fase mínima

Plantas de fase mínima são aquelas cujos zeros estão dentro do círculo unitário. Neste caso o polinômio B é estável e pode-se fazer

$$T = B.\bar{A} \qquad (6.25)$$

onde $\bar{A}(z^{-1}) = 1 + \bar{a}_1 z^{-1} + \dots + \bar{a}_{n\bar{a}} z^{-n\bar{a}}$

é o polinômio cujas raízes são os novos polos desejados para o sistema controlado, decorrentes das especificações de desempenho pretendidas, tais como característicasde resposta transitória, margens de estabilidade, erros de regime, etc.

Se fizermos

$$H = \bar{B} \qquad (6.26)$$

onde $\bar{B}(z^{-1}) = \bar{b}_0 + \bar{b}_1 z^{-1} + \dots + \bar{b}_{n\bar{b}} z^{-n\bar{b}}$

é o polinômio cujas raízes são os novos zeros pretendidos para o sistema controlado (decorrentes também das especificações de projeto), a partir de (6.24), (6.25) e (6.26) tem-se

$$y_K = \frac{z^{-d}\,\bar{B}}{\bar{A}}\, r_K \qquad (6.27)$$

O sistema então apresenta os polos e zeros relocados para as posições desejadas.

Resumindo: o projeto dos polinômios F e G é feito a partir da resolução da equação polinomial

e

$$\begin{cases} AF + z^{-d}BG = B\bar{A} & (6.28) \\ H = \bar{B} & (6.29) \end{cases}$$

Exemplo: Controladores "Dead-Beat" são controladores para os quais o sistema controlado se comporta como um atraso puro. Tais controladores não tem similar no contexto de tempo contínuo.

A partir de (6.27) pode-se ver que deve-se ter

$$\bar{A} = \bar{B} = 1 \qquad (6.31)$$

Seja a planta dada por

$$G(z) = \frac{z^{-1}(0.46 + 0.33z^{-1})}{1 - 1.37z^{-1} + 0.37z^{-2}}$$

Neste caso tem-se $d=1$, $n_a=2$, $n_b=1$, $n_{\bar{a}}=0$, $n_{\bar{b}}=0$, $n_f=1$ e $n_g=1$. Como o sistema è de fase mìnima, a equação (6.28) ficarà

$$(1-1.37z^{-1}+0.37z^{-2})(f_0+f_1z^{-1}) + z^{-1}(0.46+0.33z^{-1})(g_0+g_1z^{-1}) =$$
$$= 0.46 + 0.33z^{-1}$$

Efetuando-se a identidade polinomial, resultarà

$$f_0 = 0.46 \quad ; \quad f_1 = 0.33$$
$$g_0 = 1.37 \quad ; \quad g_1 = -0.37$$

Portanto, a lei de controle "dead-beat" è dada pela figura 6.9 com

$$F(z^{-1}) = 0.46 + 0.33z^{-1}$$
$$G(z^{-1}) = 1.37 - 0.37z^{-1}$$
$$H(z^{-1}) = 1$$

b) <u>Plantas de Fase Não-Mìnima</u>

Neste caso o polinômio B è instàvel e T não pode ser construido da forma (6.25). Convèm dizer neste ponto que sistemas de fase não mìnima são bastante frequentes quando discretizamos plantas contìnuas. Neste caso faz-se

$$T = \bar{A}$$
e $$\qquad (6.32)$$
$$H = B \quad ,$$

resultando para o siatema controlado

$$y_K = \frac{z^{-d}\, B.\bar{B}}{\bar{A}}\, r_K \qquad (6.33)$$

Note então que alèm do atraso puro d, o sistema controlado deve admitir os zeros da planta, e continua de fase não-mínima.

Para $\bar{B} = 1$ e $\bar{A} = B(1)$ tem-se o chamado Controle "dead-beat" para plantas de fase não mínima, resultando

$$y_K = \frac{z^{-d}\, B(z^{-1})}{B(1)}\, r_K \tag{6.34}$$

O termo B(1) è incluido para que o valor de regime de y_K seja igual ao de r_K. Neste caso o sistema controlado não corresponde mais a um atraso puro, mas tem um tempo de estabilização finito, de n_{b+d} passos, para entrada do tipo degrau. Por exemplo, a figura 6.10 mostra a resposta ao degrau para d=1 e $B(z^{-1}) = -2 + 4\, z^{-1}$. Note que em n_d+d=2 passos a resposta atinge o valor de regime.

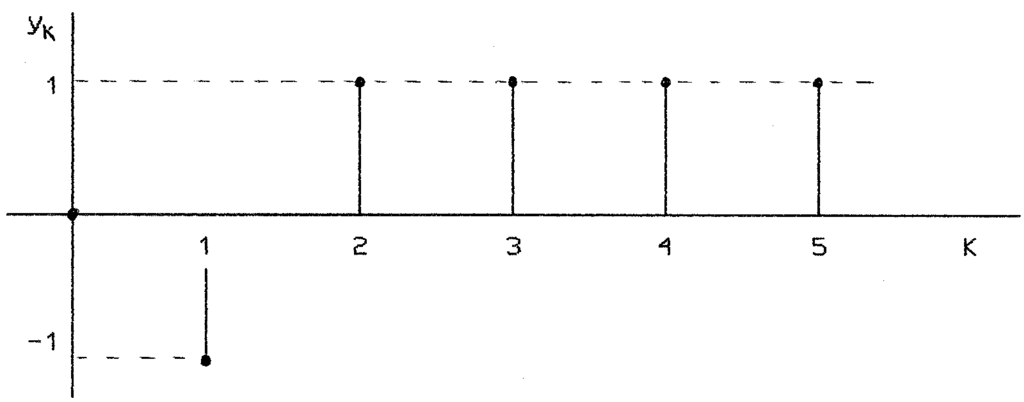


Fig 6.6 Resposta ao degrau unitàrio para o sistema dado por $y_K = z^{-1}(-1+2z^{-1})r_K$