

DCA 0118 – Processamento Digital de Sinais

Tópico 5.3: Conversão analógico/digital e digital/analógico

Tiago Barros ¹

¹{tbarros@dca.ufrn.br}

Departamento de Engenharia de Computação e Automação (DCA)
Centro de Tecnologia (CT)
Universidade Federal do Rio Grande do Norte (UFRN)

2022.1

Programa

Conteúdo

1 Processamento digital de sinais analógicos:

- 1.1 Pré-filtragem para evitar *aliasing*;
- 1.2 Conversão A/D;
- 1.3 Análise de erros de digitalização;
- 1.4 Conversão D/A;

Bibliografia

Livro texto

Oppenheim, A.V. and Schafer, R.W., 2012. Processamento em tempo discreto de sinais. Tradução Daniel Vieira. 3^a ed.-São Paulo: Pearson Education do Brasil.

- Capítulo 4:
 - Seção 4.8.

Processamento digital de sinais analógicos

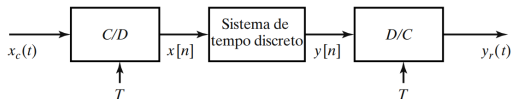
Detalhes matemáticos da relação entre um sinal de banda limitada e suas amostras:

- Amostragem \implies conversor C/D ideal;
- Reconstrução \implies conversor D/C ideal;

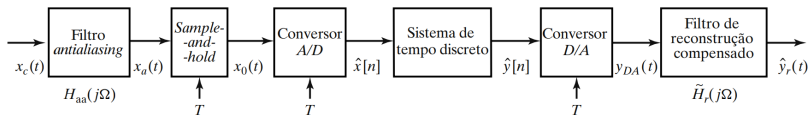
Configuração prática:

- Sinais de TC não são exatamente limitados em banda;
- Filtros ideais não podem ser realizados;
- Conversores C/D e D/C ideais são aproximados por dispositivos:
 - Conversores analógico digital (A/D);
 - Conversores digital analógico (D/A);

Processamento digital de sinais analógicos



(a)

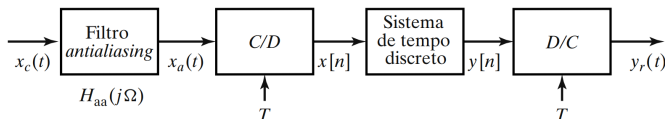


(b)

Pré-filtragem para evitar *aliasing*

Pré-filtragem é útil para:

- Quando entrada não for de banda-limitada;
- Minimizar a frequência de amostragem quando a frequência de Nyquist for muito alta;
- Ruído aditivo de banda-larga preencher faixa de frequência mais alta e acarretar em amostragem com *aliasing* na banda de baixa frequência;



Pré-filtragem para evitar *aliasing*

Resposta em frequência do filtro antialiasing deve ser:

$$H_{aa}(j\Omega) = \begin{cases} 1, & |\Omega| < \Omega_c \leq \pi/T \\ 0, & |\Omega| \geq \Omega_c \end{cases}$$

Sistema global da saída do filtro antialiasing, $x_a(t)$, até saída $y_r(t)$ é LIT (filtro antialiasing força $x_a(t)$ a ser de banda limitada em frequência abaixo de π/T). Resposta em frequência de sistema efetivo é

$$H_{\text{eff}}(j\Omega) = \begin{cases} H(e^{j\Omega T}), & |\Omega| < \Omega_c \\ 0, & |\Omega| \geq \Omega_c \end{cases}$$

Na prática, $H_{aa}(j\Omega)$ não pode ser de banda limitada, mas pode ser pequeno para $|\Omega| > \pi/T$, de forma a minimizar aliasing. Logo

$$H_{\text{eff}}(j\Omega) \approx H_{aa}(j\Omega)H(e^{j\Omega T})$$

Pré-filtragem para evitar *aliasing*

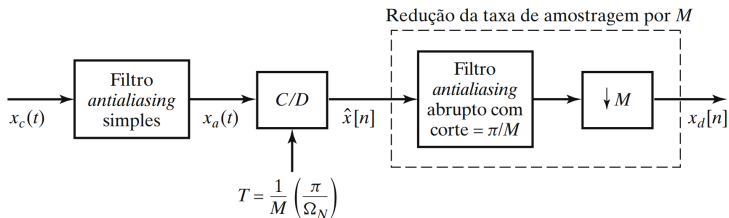
Filtros antialiasing devem ter cortes abruptos, mas apresentam dificuldades neste sentido, pois:

- São construídos a partir de circuitos analógicos (redes ativas, circuitos integrados);
- Custo muito alto;
- Resposta de fase não linear;

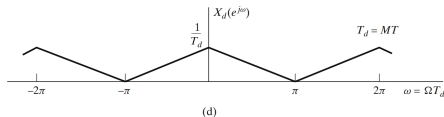
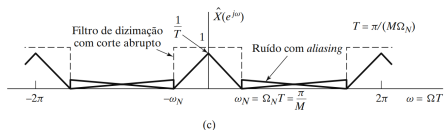
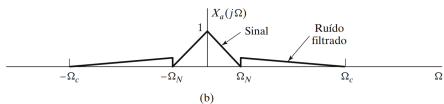
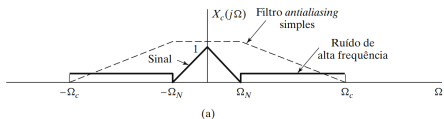
Abordagem alternativa:

- Projetar filtro antialiasing com corte gradual a partir de $M\Omega_N$;
- Realizar conversão C/D com taxa de amostragem maior do que $2\Omega_N$ (por exemplo, $M(2\Omega_N)$);
- Redução de fator de taxa de amostragem por fator M , com filtragem antialiasing com corte abrupto;

Pré-filtragem para evitar *aliasing*



Pré-filtragem para evitar *aliasing*



Conversão A/D

⇒ Conversor C/D:

- Converte sinal de tempo contínuo em sinal de tempo discreto;
- Cada amostra possui precisão infinita;

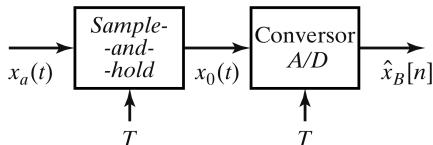
⇒ Sistema real:

- Converte sinal de tempo contínuo (analogico) em sinal digital;
- Cada amostra possui precisão finita (digitalizada);

Conversão A/D

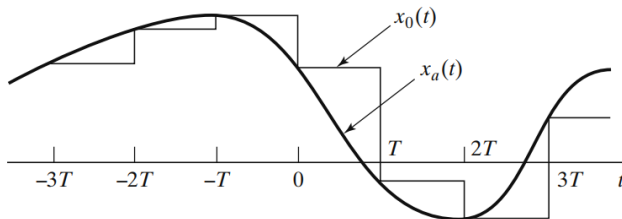
Conversor A/D:

- Dispositivo físico que converte tensão ou corrente em sua entrada em um código binário que representa um valor de amplitude digitalizado mais próximo da amplitude de entrada;
- *Clock* externo: conclui uma conversão A/D a cada T segundos;
- Conversão não é instantânea, necessita de um *sample-and-hold* (amostrador-retentor);



Conversão A/D

Sample-and-hold



Conversão A/D

Sample-and-hold ideal:

$$x_0(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t - nT);$$

onde:

- $x[n] = x_a(nT)$ são as amostras ideais de $x_a(t)$;
- $h_0(t)$ é a resposta ao impulso do sistema de retenção de ordem zero (*zero-order-hold*):

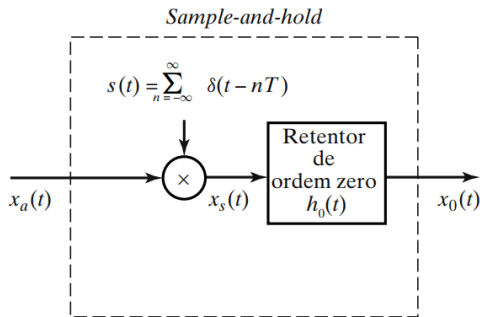
$$h_0(t) = \begin{cases} 1, & 0 < t < T \\ 0, & \text{c.c.} \end{cases}$$

Podemos reescrever como

$$x_0(t) = h_0(t) * \sum_{n=-\infty}^{\infty} x_a(nT)\delta(t - nT);$$

⇒ Sistema *sample-and-hold* ideal é equivalente à modulação pelo trem de impulsos seguida pela filtragem linear com o sistema de retenção de ordem zero.

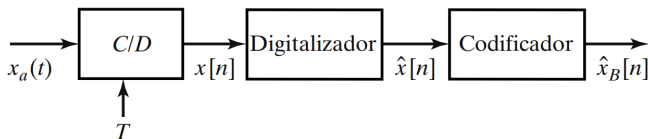
Conversão A/D



Conversão A/D

Representamos por sistema em que:

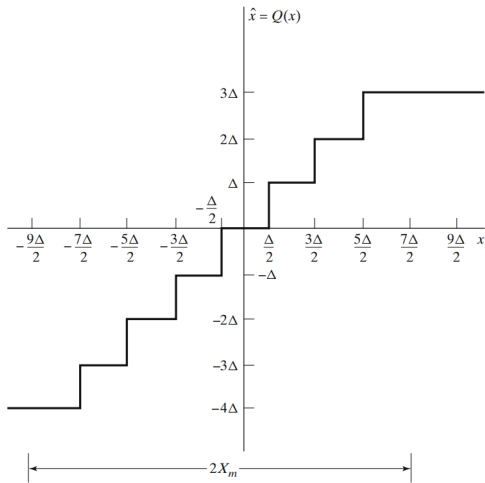
- Conversor C/D ideal \implies representa amostragem realizada pelo *sample-and-hold*;
- Digitalizador + codificador \implies representam operação do conversor A/D;



Digitalizador (ou quantizador):

- Sistema não linear que transforma a amostra de entrada $x[n]$ em conjunto finito de valores $\hat{x}[n]$;
- Amostra digitalizada é $\hat{x}[n] = Q(x[n])$;
- Digitalizador uniforme típico: amostras são arredondadas para nível mais próximo de digitalização;

Conversão A/D



Conversão A/D

Codificação dos níveis de digitalização:

- Figura (anterior) possui 8 níveis de digitalização;
- Representa-se com 3 bits;
- Número de níveis é 2^{B+1} , onde $(B + 1)$ é o número de bits;

⇒ Em um sistema binário com complemento de 2:

- O bit mais significativo (mais à esquerda) é reservado para representar o sinal, os bits restantes representam binários inteiros ou fracionários;
- Assumimos ponto fracionário binário entre dois bits mais significativos;

Conversão A/D

Para $B = 2$:

Símbolo binário	Valor numérico, \hat{x}_B
0 _◇ 11	3/4
0 _◇ 10	1/2
0 _◇ 01	1/4
0 _◇ 00	0
1 _◇ 11	-1/4
1 _◇ 10	-1/2
1 _◇ 01	-3/4
1 _◇ 00	-1

Em geral, para uma fração em complemento de dois com $(B + 1)$ bits na forma

$$a_0 \diamond a_1 a_2 \cdots a_B,$$

seu valor é

$$-a_0 2^0 + a_1 2^{-1} + a_2 2^{-2} + \cdots + a_B 2^{-B}.$$

Conversão A/D

⇒ Símbolo \diamond é ponto binário;

⇒ Parâmetro χ_m é nível de escala completo de conversor A/D;

O comprimento do passo do digitalizador é dado por

$$\Delta = \frac{2\chi_m}{2^{B+1}} = \frac{\chi_m}{2^B}.$$

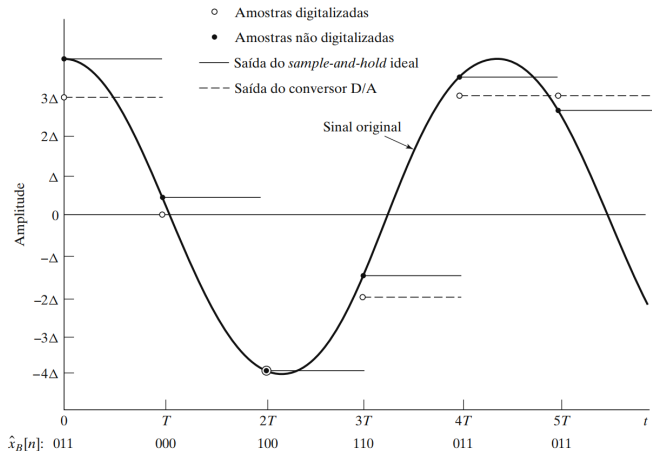
Relação numérica entre as palavras de código e as amostras digitalizadas:

$$\hat{x}[n] = \chi_m \hat{x}_B[n],$$

pois $-1 \leq \hat{x}_B[n] < 1$, para complemento de dois.

⇒ Se sinal de entrada for normalizado no intervalo $[-1, 1[$, $\hat{x}[n] = \hat{x}_B[n]$.

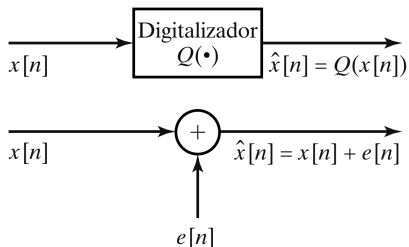
Conversão A/D



Análise de erros de digitalização

Erro (ou ruído) de digitalização:

$$e[n] = \hat{x}[n] - x[n].$$



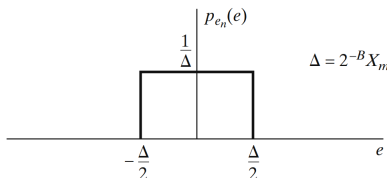
Para digitalizador de $(B+1)$ bits:

- $-\Delta/2 \leq e[n] < \Delta/2$ sempre que $(-\chi_m - \Delta/2) \leq x[n] < (\chi_m - \Delta/2)$;

Análise de erros de digitalização

Hipóteses:

- $e[n]$ é processo aleatório estacionário;
- $e[n]$ e $x[n]$ são descorrelacionados;
- $e[n]$ tem função distribuição de probabilidade uniforme no intervalo $-\Delta/2$ a $\Delta/2$;



Análise de erros de digitalização

Energia do ruído

$$\sigma_e^2 = \int_{-\Delta/2}^{\Delta/2} e^2 p_e(e) de = \frac{\Delta^2}{12}.$$

Para um digitalizador de $(B + 1)$ bits com valor χ_m de fundo de escala:

$$\sigma_e^2 = \frac{1}{12} \left(\frac{2\chi_m}{2^{B+1}} \right)^2.$$

Análise de erros de digitalização

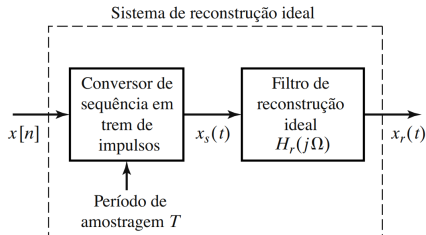
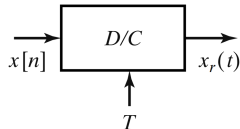
Relação sinal-ruído (SNR) em decibéis (dB)

$$\begin{aligned}\text{SNR} &= 10 \log_{10} \left(\frac{\sigma_x^2}{\sigma_e^2} \right) \\ &= 6,02B + 10,8 - 20 \log_{10} \left(\frac{\chi_m}{\sigma_x^2} \right).\end{aligned}$$

- 1 Para cada bit somado ao comprimento de palavra de amostras digitalizadas, SNR aumenta ≈ 6 dB (número de níveis de digitalização é duplicado);
- 2 Considerando parcela $-20 \log_{10} \left(\frac{\chi_m}{\sigma_x^2} \right)$:
 - Se σ_x for grande, amplitude de pico do sinal de entrada excede fundo de escala do conversor A/D. Equação de erro não é válida e ocorrem distorções severas;
 - Se σ_x for pequeno, parcela aumenta em magnitude e SNR diminui;
 - Controle automático de ganho (AGC);

Conversão D/A

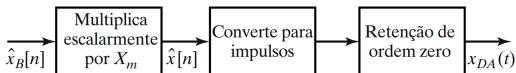
Reconstrução ideal de sinal de banda limitada:



Conversão D/A

Na prática:

- Contraparte fisicamente realizável é conversor digital analógico (conversor D/A) seguido por filtro passa-baixas analógico;
- Toma palavra de código binário $\hat{x}_B[n]$ como entrada e produz saída contínua $x_{DA}(t)$;



$$\begin{aligned}
 x_{DA}(t) &= \sum_{n=-\infty}^{\infty} \chi_m \hat{x}_B[n] h_0(t - nT) \\
 &= \sum_{n=-\infty}^{\infty} \hat{x}[n] h_0(t - nT),
 \end{aligned}$$

onde $h_0(t)$ é a resposta ao impulso do sistema de retenção de ordem zero.

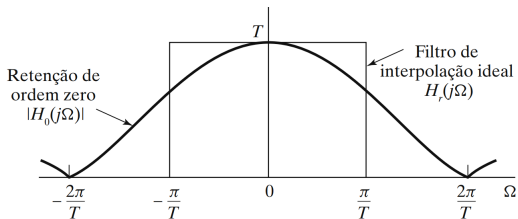
Conversão D/A

Sinal recuperado possui erro devido à digitalização ($\hat{x}[n] = x[n] + e[n]$).

$$\begin{aligned}x_{\text{DA}}(t) &= \sum_{n=-\infty}^{\infty} x[n]h_0(t - nT) + \sum_{n=-\infty}^{\infty} e[n]h_0(t - nT) \\&= x_0(t) + e(t).\end{aligned}$$

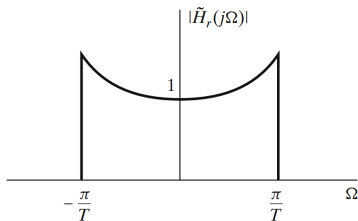
Conversão D/A

Além disso, resposta em frequência do filtro de interpolação (retentor de ordem zero) não é ideal.

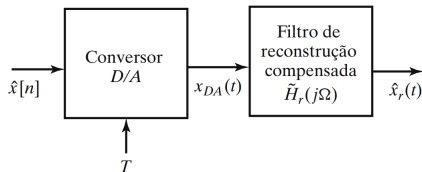


Conversão D/A

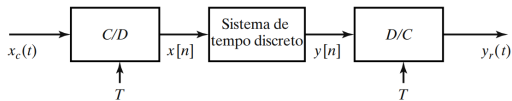
Projeta-se filtro analógico para compensar efeito de $H_0(j\Omega)$:



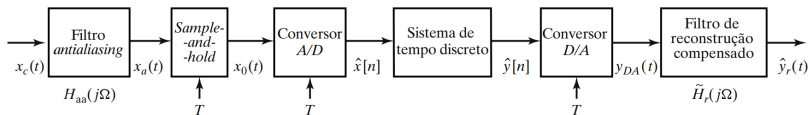
Configuração física do conversor D/A:



Recapitulando



(a)



(b)