UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ENGENHARIA ELÉTRICA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

CALLEBE SOARES BARBSOA

RAFAEL DA COSTA BONOTTO

RAPHAEL HENRIQUE SOARES MACHADO

VICTOR EMANUEL SOARES BARBOSA

RELATÓRIOS, EXERCÍCIOS E FUNDAMENTAÇÃO TEÓRICA RELACIONADOS A DISCIPLINA DE CONTROLE DIGITAL

PATO BRANCO

2016

CONTEÚDO

1 INTRODUÇÃO

2 IMPLEMENTAÇÃO DA CONVOLUÇÃO

2.1 OBJETIVOS

O objetivo principal desta prática é a implementação da convolução como ferramenta matemática para obtenção da saída de um sistema dada uma entrada qualquer e a resposta ao impulso.

2.2 FUNDAMENTAÇÃO TEÓRICA

A convolução, pode-se assim dizer, é o equivalente entre sinais da multiplicação. Ela pode ser descrita em tempo contínuo, sendo chamada de integral de convolução e em tempo discreto de soma de convolução.

2.2.1 Integral de convolução

A resposta $y(t)$ a uma entrada $x(t)$ aplicada a um sistema T, sendo este linear e invariante no tempo, pode ser dado pela Equação 2, onde $h(t)$ é a resposta do sistema ao impulso. Esta dedução partiu da propriedade da função impulso dada na Equação 1.

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \tag{1}$$

$$\begin{aligned}
y(t) &= Tx(t) \\
y(t) &= T\int_{-\infty}^{\infty} x(\tau)\delta t - \tau d\tau \\
y(t) &= \int_{-\infty}^{\infty} x(\tau)T\delta t - \tau d\tau \\
y(t) &= \int_{-\infty}^{\infty} x(\tau)ht - \tau d\tau \\
y(t) &= x(t) * h(t) \qquad (2)
\end{aligned}$$

2.2.2 Soma de convolução

A resposta em tempo discreto, $y[n]$, a uma entrada $x[t]$ aplicada a um sistema T, sendo este linear e invariante no tempo, pode ser dado pela Equação 4, onde $h[n]$ é a resposta do sistema ao impulso. Esta dedução partiu da propriedade da função impulso dada na Equação 3.

$$x[n] = \sum_{-\infty}^{\infty} x[k]\delta(n-k) \tag{3}$$

$$\begin{aligned}
y[n] &= Tx[n] \\
y[n] &= T\sum_{-\infty}^{\infty} x[k]\delta[n-] \\
y[n] &= \sum_{-\infty}^{\infty} x[k]T\delta[n-k] \\
y[n] &= \sum_{-\infty}^{\infty} x[k]h[n-k] \\
y[n] &= x[n] * h[n] \qquad (4)
\end{aligned}$$

2.3 PROCEDIMENTOS

Foram resolvidos os exercícios 1 2 da apresentação de slides referente a transformada Z, com código implementado em Matlab.

2.3.1 Exercício 1

Determine a saída do sistema com resposta ao impulso h[n] e para um sinal de entrada x[n], ambos sinais estão mostrados na Figura 1:
A) análise gráfica por impulsos
B) cálculo/tabela de convolução
C) convolução utilizando ferramenta computacional: *script* Matlab

2.3.2 Exercício 2

Considere um sistema que possui resposta ao impulso h[n]=2^{-nT} e o sinal de entrada é uma onda retangular (razão cíclica 40%, D=0,4) com período 10s e amplitude 3,3V.

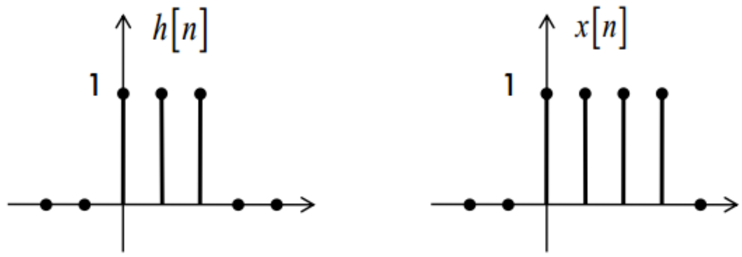

Figura 1: Sinais do Exercício 1, prática 1.

A) Determine a resposta (sinal de saída) do sistema para 3 períodos do sinal de entrada considerando que o periodo de amostragem é T=0.2s.

B) Considere um ruído de 10% no sinal de entrada e repita o item A.

C) Considere um sinal de entrada senoidal com mesmo período, amplitude e ruído dados acima.

2.4 RESULTADOS E DISCUSSÕES

2.4.1 Exercício 1

2.4.1.1 C)

O código implementado em Matlab para a resolução da letra C do exercício 1 está logo abaixo:

```
%% ----------------------------------
%   Universidade Tecnologica Federal do Parana
%   Engenharia Eletrica
%   Controle Digital
%
%   Aluno: Victor Emanuel Soares Barbosa
%
%   Aula 3: Transformada Z
%   Exercicio 1:
%   Determine a saida do sistema com
%   resposta ao impulso h[n] e para um sinal de entrada
%   x[n]:
%   C) convolucao utilizando ferramenta
%   computacional: script Matlab
% ----------------------------------
%% Inicializacao do programa
clc;
clear all;
```

```
close all;

%% Variaveis gerais
numero_pontos = 8; % Numero de pontos simulados
h = [1 1 1 zeros(1,numero_pontos-3)]; % resposta ao impulso
x = [1 1 1 1 zeros(1,numero_pontos-4)]; % sinal de entrada
y = [zeros(1,numero_pontos)]; % resposta do sistema a entrada x
amostras = zeros(1,numero_pontos); % vator de amostras

%% Letra c)

% Execucao
for n=0:(numero_pontos-1)
    for k = 0:3
        if (n-k)>0
            y(n+1) = y(n+1) + x(k+1)*h(n-k);
        end
    end
    amostras(n+1) = n-1;
end

% Graficos
figure
stem(amostras,y)
title('Aula 2 - Exercicio 2 - Letra c');
legend('Sinal y[n]');
ylabel('Amplitude');
xlabel('n');
```

Obtendo como saída o gráfico da Figura 2

Observa-se que os intervalos de simulação para uma convolução teórica deveria ser estabelecido entre $-\infty$ e $+\infty$, com uma valor de passo unitário. Porém computacionalmente esses intervalos são impraticáveis, necessitando de uma adequação nos valores de simulação.

Como os sinais h e x possuem valores diferentes de zero apenas a partir de $n = 0$ até $n = 3$ se faz necessário convolucionar apenas neste intervalo. O numero total de pontos convolucionados é dado pela soma dos valores no intervalo da convolução dos dois sinais, sendo portanto no total 8 pontos.

A figura 2 apresenta o resultado da simulação de forma coerente com a teoria, e portanto comprova a eficacia do algoritmo desenvolvido.

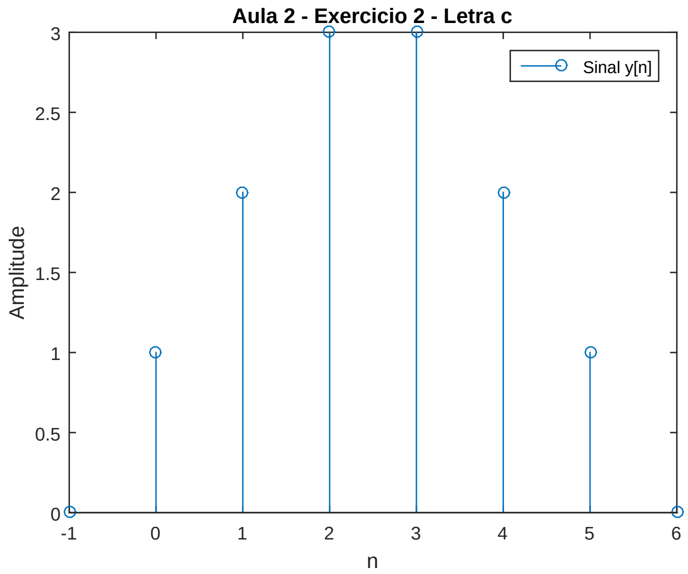

Figura 2: Gráfico de saída do código do Exercício 1, letra C, prática 1.

2.4.2 Exercício 2

O código implementado em Matlab para a resolução do exercício 2 está logo abaixo:

```
%% ————————————————————
%   Universidade Tecnologica Federal do Parana
%   Engenharia Eletrica
%   Controle Digital
%
%   Aluno: Victor Emanuel Soares Barbosa
%
%   Aula 3: Transformada Z
%   Exercicio 2:
%   Considere um sistema que possui resposta
%   ao impulso h[n]=2-nT e o sinal de entrada he uma onda
%   retangular (razao ciclica 40%, D=0,4) com periodo
%   10s e amplitude 3,3V.
%   A) Determine a resposta (sinal de saida) do sistema para
%   3 periodos do sinal de entrada considerando que o
%   periodo de amostragem he T=0.2s.
%   B) Considere um ruido de 10% no sinal de entrada e
%   repita o item A.
%   C) Considere um sinal de entrada senoidal com mesmo
%   periodo, amplitude e ruido dados acima.
% ————————————————————
```

```
%% Inicializacao do programa
clc;
clear all;
close all;

%% Variaveis gerais
periodos = 3; % Quantidade de periodos
R = 0.1; % Nivel de ruido
T = 0.2; % Periodo de amostragem
T_entrada = 10; % Periodo do sinal de entrada
A_entrada = 3.3; % Amplitude do sinal de entrada
D = 0.4; % Razao ciclica do sinal de entrada
total_pontos = periodos*T_entrada/T; % Total de pontos simulados
pontos_periodo = T_entrada/T; % Total de pontos por periodo
amostras = zeros(1,total_pontos); % Vetor de pontos de simulacao
h = zeros(1,total_pontos); % vetor da resposta ao impulso
x = zeros(1,total_pontos); % Vetor do sinal de entrada
y = zeros(1,total_pontos); % Vetor do sinal de saida
cont  = 0;

%% Letra a)

% Execucao
for n = 0:total_pontos-1

    if cont < (pontos_periodo*D)
         x(n+1) = A_entrada;
    else
        x(n+1) = 0;
    end

    if cont == pontos_periodo
        cont = 0;
    end

    cont = cont + 1;

    for k = 0:pontos_periodo*3
        h(n+1) = 2^(-n*T);

        if (n-k)> 0
```

```
                y(n+1) = y(n+1) + x(k+1)*h(n-k)*T;
            end
        end

        amostras(n+1) = (n)*T;
end

% Graficos
figure
stem(amostras,y)
hold
stem(amostras,x)
stem(amostras,h)
title('Aula 2 - Exercicio 2 - Letra a');
legend('Sinal y[n]', 'Sinal x[n]', 'Sinal h[n]');
ylabel('Amplitude');
xlabel('Tempo (s)');

%% Letra b)

% Execucao
for n = 0:total_pontos-1

        if cont < (pontos_periodo*D)
              x(n+1) = A_entrada*(1+0.1*rand);
        else
            x(n+1) = 0.1*rand;
        end

        if cont == pontos_periodo
            cont = 0;
        end

        cont = cont + 1;

        for k = 0:pontos_periodo*3
            h(n+1) = 2^(-n*T);

            if (n-k)> 0
                y(n+1) = y(n+1) + x(k+1)*h(n-k)*T;
            end
```

```
    end

    amostras(n+1) = (n)*T;
end

% Graficos
figure
stem(amostras,y)
hold
stem(amostras,x)
stem(amostras,h)
title('Aula 2 - Exercicio 2 - Letra b');
legend('Sinal y[n]', 'Sinal x[n]', 'Sinal h[n]');
ylabel('Amplitude');
xlabel('Tempo (s)');

%% Letra c)

% Execucao
for n = 0:total_pontos-1

    x(n+1) = A_entrada*sin(2*pi*n/(T_entrada/T))*(1+0.1*rand);

    if cont == pontos_periodo
        cont = 0;
    end

    cont = cont + 1;

    for k = 0:pontos_periodo*3
        h(n+1) = 2^(-n*T);

        if (n-k)> 0
            y(n+1) = y(n+1) + x(k+1)*h(n-k)*T;
        end
    end

    amostras(n+1) = (n)*T;
end

% Graficos
```

```
figure
stem(amostras,y)
hold
stem(amostras,x)
stem(amostras,h)
title('Aula 2 - Exercicio 2 - Letra c');
legend('Sinal y[n]', 'Sinal x[n]', 'Sinal h[n]');
ylabel('Amplitude');
xlabel('Tempo (s)');
```

Obtendo como saída os gráfico das Figuras 3, 4 e 5.

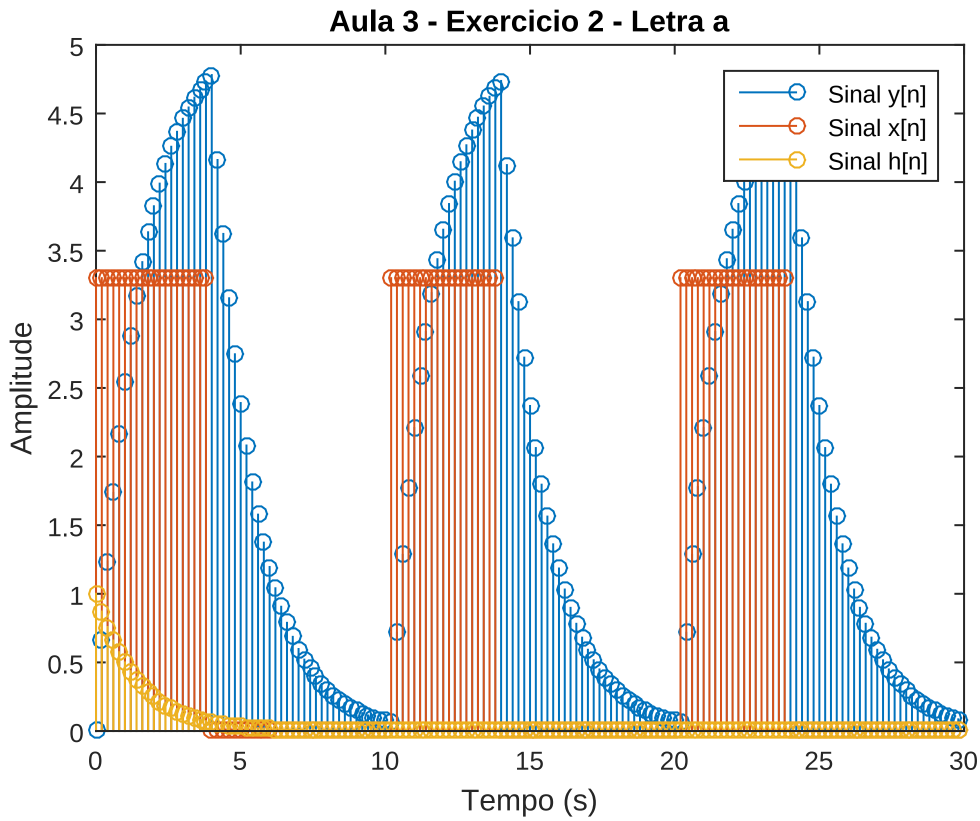

Figura 3: Gráfico de saída do código do Exercício 2, letra A, prática 1.

Neste exercício deve se ter umas atenção especial em relação ao passo da convolução, ou seja a quantidade de pontos simulados entre valores inteiros.

Como temos um período de amostragem de $0,2$ logo temos 5 pontos simulados entre valores inteiros. Assim se fixarmos o numero de períodos e variarmos o número de amostragem variamos também o número de pontos a serem convolucionados. Como a convolução pode ser representada pela soma na forma da equação 4, logo a amplitude de cada ponto n do sinal convolucionado depende do numero de amostras realizado. Para que se possa ter um

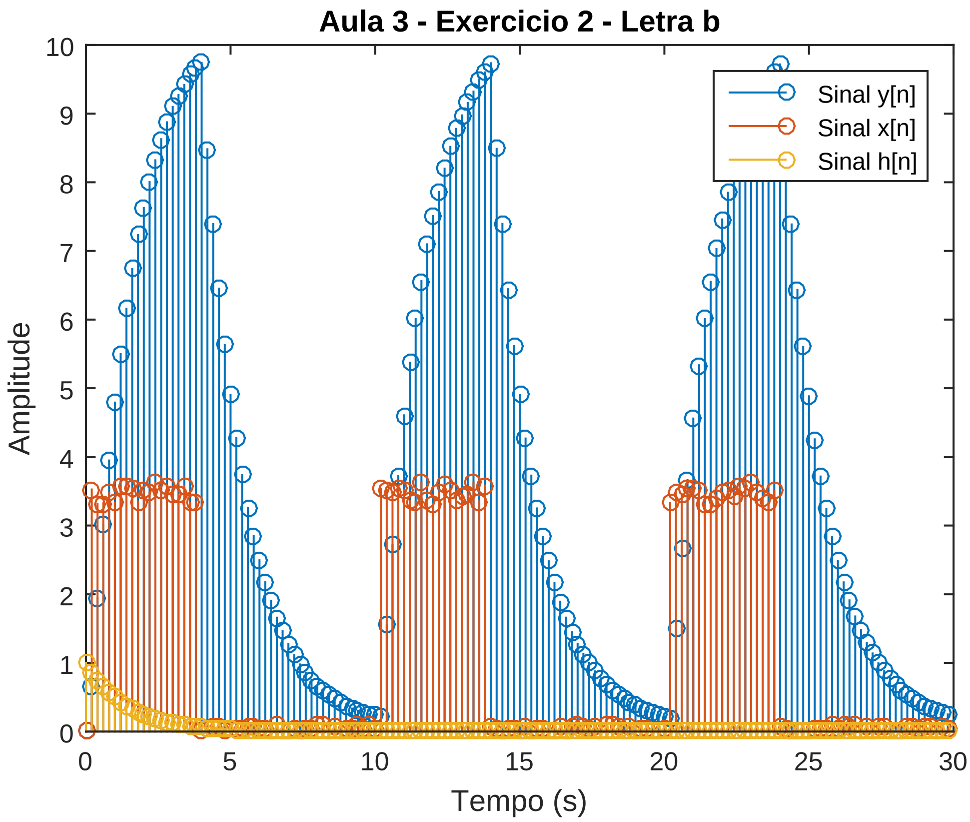

Figura 4: Gráfico de saída do código do Exercício 2, letra B, prática 1.

sinal de amplitude normalizada independente do período de amostragem, o sinal convolucionado é dividido pelo numero de amostras entre intervalo entre dois inteiros, ou numero de amostras por ciclo.

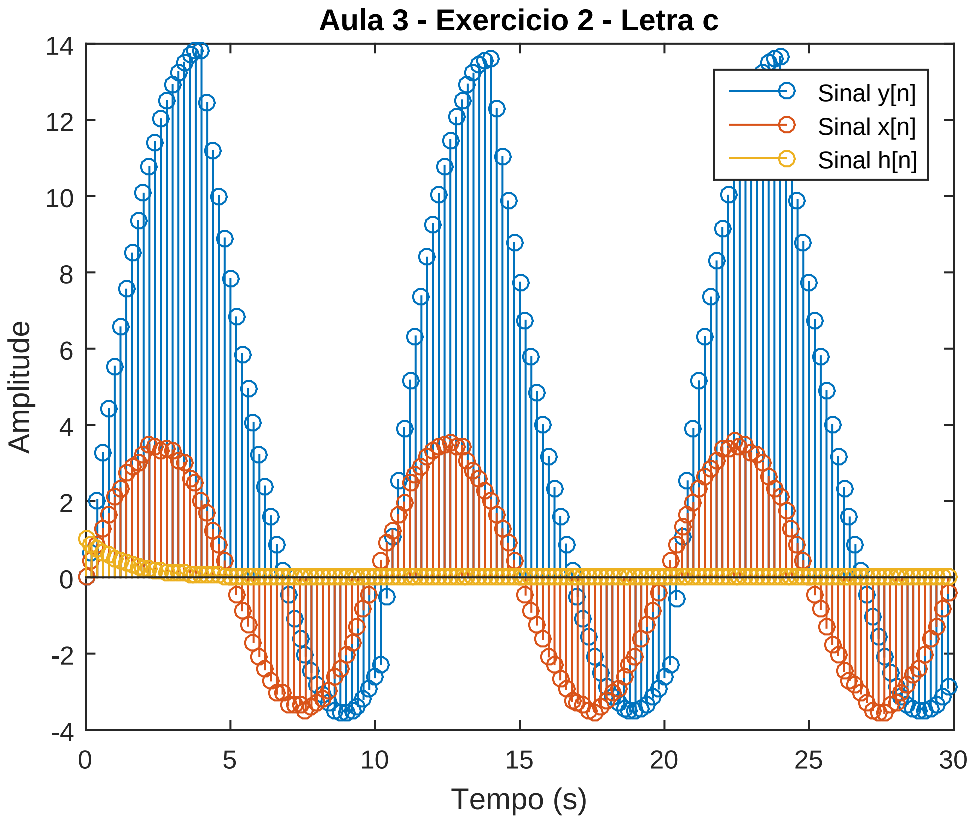

Figura 5: Gráfico de saída do código do Exercício 2, letra C, prática 1.

3 SIMULAÇÃO DE UM SISTEMA DISCRETO COM EQUAÇÕES DIFERENÇAS

3.1 OBJETIVOS

O objetivo principal desta prática é a simulação em Matlab de um sistema de controle digital completo dado as equações em Z que descrevem os blocos constituintes do sistema em malha fechada.

3.2 FUNDAMENTAÇÃO TEÓRICA

Para realização desta prática se utilizou a teoria da transformada Z inversa, em específico o método das equações de diferenças. Este método é facilmente utilizado e computadores digitais por fornecer a equação em tempo discreto da transforma inversa de *z* .

Quando obtemos a transformada inversa de *z*, assumimos que a sequência $x(kT)$ ou $x(k)$ é zero para $k < 0$. Nota-se que em aplicação de engenharia de controle e processamento de sinais, $X(z)$ é frequentemente expressado com a razão polinomial de z^{-1}, como apresentado na equação (1)

$$G(z) = \frac{Y(z)}{X(z)} = \frac{b_0 z^{-(n-m)} + b_1 z^{-(n-m+1)} + ... + b_m z^{-n}}{1 + a_1 z^{-1} + a_2 z^{-2} + ... + a_n z^{-n}} \quad (m \leq n) \tag{1}$$

Pelo método aproximado de equações de diferenças convertemos a equação (1) para a equação (2),

$$(1 + a_1 z^{-1} + a_2 z^{-2} + ... + a_n z^{-n})Y(z) = (b_0 z^{-(n-m)} + b_1 z^{-(n-m+1)} + ... + b_m z^{-n})X(z)$$

$$\begin{aligned} y(k) + a_1 y(k-1) + a_2 y(k-2) + ... + a_n y(k-n) &= b_0 x(k-(n-m)) + ... \\ &...b_1 x(k-(n-m+1)) + ... + b_m x(k-n) \\ y(k) = -a_1 y(k-1) - a_2 y(k-2) + ... + -a_n y(k-n) &+ b_0 x(k-(n-m)) + ... \\ &...b_1 x(k-(n-m+1)) + ... + b_m x(k-n) \end{aligned} \tag{2}$$

Achando a transformada inversa *z* de $Y(z)$, resolve-se a equação de diferença $y(k)$ facilmente por linguagem de programação.

3.3 PROCEDIMENTOS

O exercício consistiu em simular um sistema de tempo discreto para período de amostragem de 0,1 s para sinal de entrada uma onda quadrada de amplitude 0 - 5 V, com período de 10s e 2% de ruído randômico. O sistema está mostrado na Figura 6, bem como as equações dos blocos estão descritas abaixo:

$$C(z) = 0,9 * \frac{z - 0,8}{z - 1} \tag{3}$$

$$G(z) = \frac{0,3z}{(z - 0,5)(z - 0,2)} \tag{4}$$

$$S(z) = \frac{0,2}{z - 0,8} \tag{5}$$

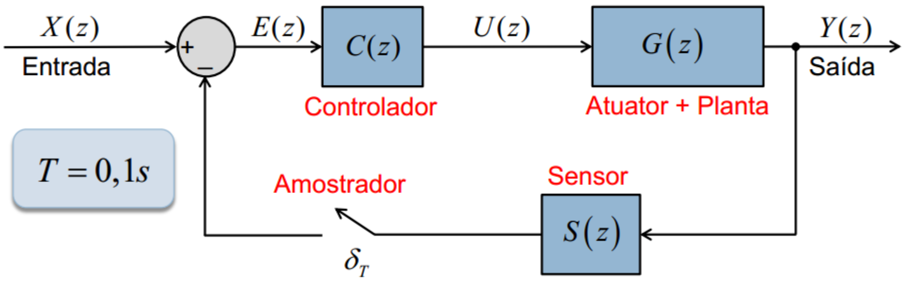

Figura 6: Diagrama de blocos da prática 2.

3.4 RESULTADOS E DISCUSSÕES

Realizando o equacionamento genérico a malha fechada do sistema apresentado na Figura 6 encontra-se a equação (6)

$$\frac{Y(z)}{X(z)} = \frac{C(z)G(z)}{1 + C(z)G(z)S(z)} \tag{6}$$

Substituindo as equações (3), (4) e (5) na equação (6) e expressado com a razão polinomial de z^{-1} temos

$$\frac{Y(z)}{X(z)} = \frac{0,27z^{-1} - 0,216z^{-2}}{1 - 1,7z^{-1} + 0,854z^{-2} - 0,1z^{-3}} \tag{7}$$

Através da equação (7) aplica-se o método da equação de diferença conforme apresentado na equação (2), concebendo a equação (??) que será simulada com o respectivo sinal de entrada.

$$y(k) = 1,7y(k-1) - 0,854y(k-2) + 0,1y(k-3) + 0,27x(k-1) - 0,216x(k-2) \quad (8)$$

O código implementado em Matlab está mostrado abaixo:

```
%% ---------------------
%   Universidade Tecnologica Federal do Parana
%   Engenharia Eletrica
%   Controle Digital
%
%   Aluno: Victor Emanuel Soares Barbosa
%
%   Aula 5: Transformada Z Inversa
%   Exercicio 2:
%   Considere o diagrama de controle em tempo discreto.
%   Determinar o grafico da saida considerando:
% ---------------------
%% Inicializacao do programa
clc;
clear all;
close all;

%% Variaveis gerais
ciclos = 4; % Quantidade de ciclos da onda de entrada
T_entrada = 10; % Periodo da onda de entrada
Ta = 0.1; % Periodo de amostragem
total_pontos = T_entrada/Ta*ciclos; % Total de pontos de simulacao
A_entrada = 5; % Amplitude de entrada
x = zeros(1,total_pontos); % Vetor da entrada
y = zeros(1,total_pontos); % Vetor de saida
tempo = zeros(1,total_pontos); % Vetor de tempo
cont = 0; % contador para auxiliar

for n = 1:total_pontos
    if cont < ((T_entrada/Ta)/2)
        x(n) = A_entrada*(1+0.02*rand);
    else
        x(n) = 0.02*rand*A_entrada;
```

```
    end

    if n > 3
        y(n) = 0.27*x(n-1) - 0.216*x(n-2) + 1.7*y(n-1) - 0.854*y(n-2) ...
            + 0.1*y(n-3);
    end

    tempo(n) = n*Ta;
    cont = cont + 1;

    if cont > T_entrada/Ta
        cont = 0;
    end
end

% Grafico
figure
stem(tempo,y)
hold
stem(tempo,x)
title('Aula 5 - Exercicio 2');
legend('Saida y[n]','Entrada x[n]');
ylabel('Amplitude');
xlabel('Amostras');
```

Obtendo como saída o gráfico da Figura 7.

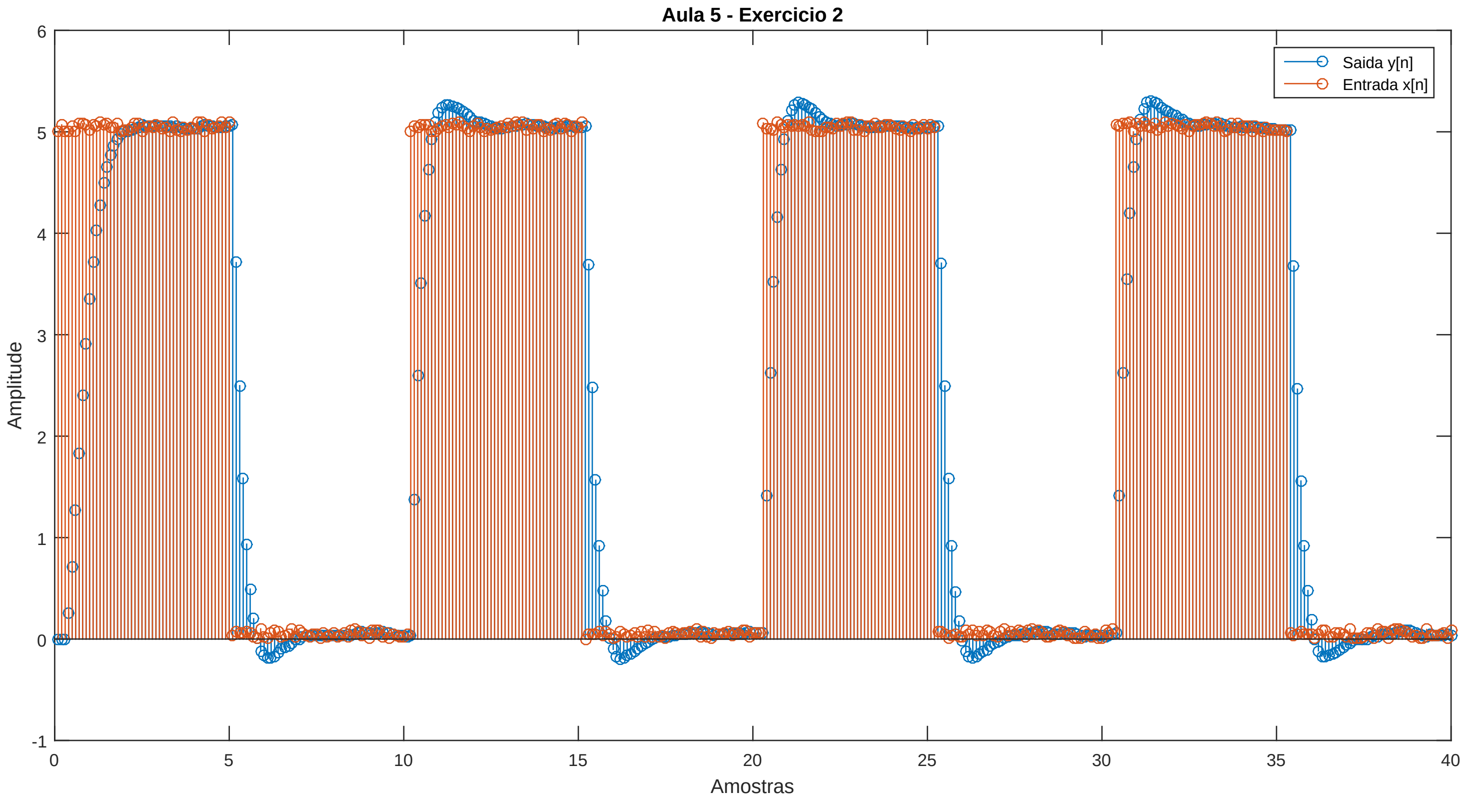

Figura 7: Gráfico de saída do código do exercício da prática 2.

COMENTAR E DISCUTIR RESULTADOS

4 MODULADOR PWM E SIST.COND. SINAIS E ADC

4.1 OBJETIVOS

4.2 FUNDAMENTAÇÃO TEÓRICA

4.3 PROCEDIMENTOS

4.3.1 Exercício 1

Considere um sistema em malha fechada com PID T(z) sendo:

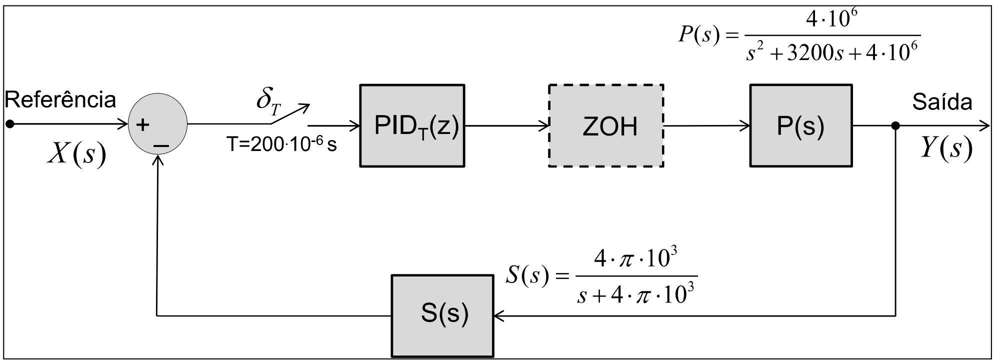

Figura 8

Utilize simulações computacionais (Simulink®) para projetar ganhos para o controlador PID considerando a entrada um degrau unitário.

4.3.2 Exercício 2

Obtenha a função de transferência do PID de tempo discreto utilizando o método de discretização Forward para a parcela integral e considere Kp, Ki, e Kd como ganhos paralelos do controlador PID.

4.3.2.1 Repita o Exercício 1 para esta função de transferência comparando os resultados de simulação de ambos os casos para os mesmos ganhos.

4.3.2.2 Inclua saturação na ação de controle em 150% da referência e analise o comportamento do sistema de controle

4.3.3 Exercício 3

Considerando o sistema descrito no exercício 2, desenvolva um script em Matlab para implementar o PID com os seguintes parâmetros.

- Sinal de referência: Onda quadrada Amplitude $40\ V_{pp}$ Offset $0\ V$ Período $10ms$
- Controlador: PID 'Digital' (equação de diferenças) Saturação do PID (Sat = $0.98\ V_{cc}$)
- Atuador: sinal PWM Resolução 8 bits (2n divisões) $V_{cc} = 40\ V$
- Ruído: Randômico Amplitude 2% da saída
- Conversor A/D: Resolução 10 bits (2n níveis) Vin=0-5V T=200.10-6 s
- Planta $P(s) = \frac{4 \cdot 10^6}{s^2+3200s+4 \cdot 10^6}$
- Sensor $S(s) = \frac{4 \cdot \pi \cdot 10^3}{s+4 \cdot \pi \cdot 10^3}$

4.4 RESULTADOS E DISCURSÕES

5 AMOSTRAGEM DE SINAIS E ANÁLISE EM FREQUENCIA DE SINAIS AMOSTRADOS

5.1 OBJETIVOS

5.2 FUNDAMENTAÇÃO TEÓRICA

5.3 PROCEDIMENTOS

5.4 RESULTADOS E DISCURSÕES

6 CONTROLADOR PID

6.1 OBJETIVOS

6.2 FUNDAMENTAÇÃO TEÓRICA

6.3 PROCEDIMENTOS

6.4 RESULTADOS E DISCURSÕES

7 CONTROLADOR REPETITIVO

7.1 OBJETIVOS

7.2 FUNDAMENTAÇÃO TEÓRICA

7.3 PROCEDIMENTOS

7.4 RESULTADOS E DISCURSÕES

8 TRANSFORMADA Z

8.1 OBJETIVOS

8.2 FUNDAMENTAÇÃO TEÓRICA

8.3 PROCEDIMENTOS

8.4 RESULTADOS E DISCURSÕES

9 TRANSFORMADA Z INVERSA

9.1 OBJETIVOS

9.2 FUNDAMENTAÇÃO TEÓRICA

9.3 PROCEDIMENTOS

9.4 RESULTADOS E DISCURSÕES

10 CONCLUSÃO

REFERÊNCIAS