

## Capítulo 1

# Controle Proporcional de um Motor de Corrente Contínua

### 1.1 Objetivo

O objetivo desta prática é introduzir ao estudante as noções básicas de realimentação e controle, implementando um sistema de controle proporcional para posicionar o eixo de um motor de corrente contínua. Serão desenvolvidos os conceitos de modelagem da planta e de implementação de um controlador através de uma plataforma de instrumentação virtual baseada em LabVIEW.

### 1.2 Modelo matemático

Um motor elétrico de corrente contínua é composto por uma parte móvel (rotor), definida por um enrolamento de espiras, e uma parte fixa (estator), geradora de campo magnético. O seguinte esquema eletromecânico, Figura 1.1, representa o motor elétrico de corrente contínua:

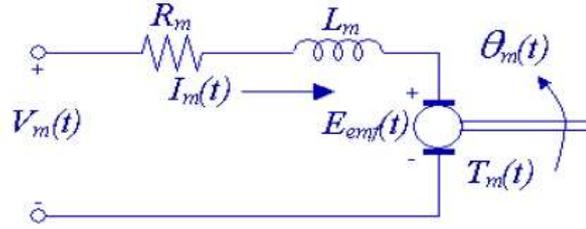


Figura 1.1: Diagrama eletromecânico do motor de corrente contínua.

sendo  $V_m(t)$  a tensão aplicada ao rotor,  $I_m(t)$  a corrente que circula pelo enrolamento,  $R_m$  a resistência do enrolamento ou da armadura,  $L_m$  a indutância característica do rotor,  $E_{emf}$  a força contraeletromotriz induzida no enrolamento pelo campo magnético do estator,  $T_m(t)$  o torque desenvolvido pelo motor e  $\theta_m(t)$  a posição angular do eixo do motor.

Usando a lei de Kirchhoff de tensão, obtém-se a equação abaixo:

$$V_m - R_m I_m - L_m \frac{dI_m}{dt} - E_{emf} = 0. \quad (1.1)$$

Como geralmente  $L_m \ll R_m$ , pode-se desconsiderar a indutância do motor, assim:

$$I_m = \frac{V_m - E_{emf}}{R_m}. \quad (1.2)$$

Sabe-se que a força contraeletromotriz gerada pelo motor é proporcional à velocidade do rotor,  $\omega_m$ , tem-se:

$$I_m = \frac{V_m - K_m \dot{\theta}_m}{R_m} \quad (\dot{\theta}_m = \omega_m), \quad (1.3)$$

sendo  $K_m$  a constante contraeletromotriz.

Do ponto de vista mecânico, aplicando a segunda lei de Newton ao movimento do

rotor do motor:

$$J_m \ddot{\theta}_m = T_m - \frac{T_l}{\eta_g K_g}, \quad (1.4)$$

sendo  $T_l$  o torque na carga,  $K_g$  a relação de engrenagens entre o motor e a carga, e  $\eta_g$  a eficiência da caixa de engrenagens; e ao movimento da carga acoplada ao eixo do motor:

$$J_l \ddot{\theta}_l = T_l - B_{eq} \dot{\theta}_l, \quad (1.5)$$

sendo  $B_{eq}$  o coeficiente viscoso de amortecimento.

Obtém-se a equação dinâmica do movimento dada por:

$$J_l \ddot{\theta}_l = \eta_g K_g T_m - \eta_g K_g J_m \ddot{\theta}_m - B_{eq} \dot{\theta}_l. \quad (1.6)$$

Utilizando as transformações  $\theta_m = K_g \theta_l$  e  $T_m = \eta_m K_t I_m$  (sendo  $\eta_m$  a eficiência do motor e  $K_t$  a constante de torque do motor), a equação (1.6) pode ser reescrita como:

$$J_l \ddot{\theta}_l + \eta_g K_g^2 J_m \ddot{\theta}_l + B_{eq} \dot{\theta}_l = \eta_g \eta_m K_g K_t I_m. \quad (1.7)$$

Finalmente, combinando as equações elétrica, (1.3), e mecânica, (1.7), a função de transferência que estabelece a relação entre a posição angular da carga acoplada ao eixo,  $\theta_l$  e a tensão aplicada ao motor,  $V_m$ , é dada por:

$$\frac{\theta_l(s)}{V_m(s)} = \frac{\eta_g \eta_m K_t K_g}{J_{eq} R_m s^2 + (B_{eq} R_m + \eta_g \eta_m K_m K_t K_g^2) s}, \quad (1.8)$$

sendo  $J_{eq} = J_l + \eta_g J_m K_g^2$ .

### 1.3 Pré-laboratório

Considerando os parâmetros descritos no Apêndice A, a função de transferência do motor CC utilizado nesta prática é dada por: (Obs.: Atenção à transformação de

unidades: radianos  $\rightarrow$  graus.)

$$G(s) = \frac{\theta_l}{V} = \frac{3673.7}{s^2 + 36.4s}. \quad (1.9)$$

O controlador proporcional, Figura 1.2, é implementado no controle do motor pela equação:

$$V_m = K_P(\theta_l^d - \theta_l) \quad (1.10)$$

sendo  $\theta_l^d$  o valor desejado para o ângulo da carga.

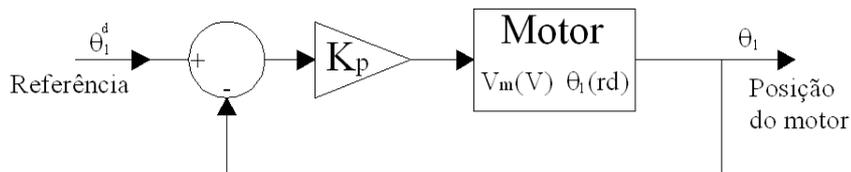


Figura 1.2: Planta do controlador proporcional.

1. Calcule  $K$  para  $t_r < 0.065$  e  $MP < 10\%$ .
2. Calcule o erro de regime para o sistema em malha fechada.

## 1.4 Procedimento de laboratório

### 1.4.1 Ligações e conexões

A primeira tarefa é assegurar que todo o sistema está ligado corretamente. Se você está inseguro com a ligação, chame o professor. Com todos os sinais ligados corretamente você estará pronto para iniciar o laboratório.

## 1.4.2 Implementação

Segue abaixo os passos para a construção do programa de controle proporcional no LabVIEW para o motor de corrente contínua.

1. No diagrama de blocos do LabVIEW, adicione um ciclo **Timed Loop** ( **Programming** >> **Structures** >> **Timed Structures**). Em seguida clique com o botão direito do mouse em **Loop Condition** (canto inferior direito do quadro **Timed Loop**), e clique em **Create** >> **Control**.
2. Para controlar o tempo de amostragem do programa, clique com o botão direito do mouse em **dt** e selecione **Create** >> **Control**. Em seguida, ajuste o valor para *30ms* no Painel Frontal.

### LEITURA DO ENCODER:

3. Na lateral esquerda e exterior ao ciclo **Timed Loop** adicione o bloco **Create Channel** (**MeasurementI/O** >> **NI-DAQmx**). Em seguida, selecione a opção **Count Edges** (**CI Cnt Edges**>>**Counter Input**). No mesmo bloco adicione uma constante na entrada **Count Direction** e outra na entrada **Counter**. Na primeira selecione a opção **Externally Controlled** e na segunda **I/O Dev/ctr0**.
4. Adicione o bloco **Start** (**MeasurementI/O** >> **NI-DAQmx**). Conecte a saída **task out** do bloco **CI Cnt Edges** em **task/channels in** do bloco **Start**, e a saída **error out** do bloco **CI Cnt Edges** à entrada **error in** do bloco **Start**.
5. Agora, dentro do ciclo **Timed Loop**, adicione um bloco **Read** (**MeasurementI/O** >> **NI-DAQmx**). Selecione o modo **Counter U32 1 Samp** (**Analog BBL** >> **Counter** >> **Single Sample** >> **U32**). Em seguida conecte-o ao bloco **Start** inserido na etapa anterior. Esse bloco realiza a leitura dos sinais do encoder.
6. Na lateral direita e externa ao bloco **Timed Loop**, adicione um bloco **Stop** (**Measurement I/O** >> **NI-DAQmx**) e conecte as entradas **task/channels in** e **error in** às saídas **task out** e **error out**, respectivamente, do bloco **Read** adicionado no passo anterior. Em seguida, adicione o bloco **Clear** (**Measurement I/O** >> **NI-DAQmx**) e conecte-o ao bloco **Stop** da mesma forma.

7. Dentro do ciclo **Timed Loop**, uma conversão numérica torna-se necessária para identificar a direção de rotação do encoder, positiva ou negativa. Assim, conecte o bloco **I32 (Programming >> Numeric >> Conversion)** à saída **Data** do bloco **Read**.
8. Na sequência, acrescente um bloco de divisão (**Programming>>Numeric**), conectando a saída do bloco **I32** na entrada do numerador e uma constante com valor **1** na entrada do denominador. Essa transformação aplica a relação da redução mecânica à leitura de posição do eixo caso o encoder esteja posicionado antes do sistema de engrenagens.
9. Adicione agora um bloco de multiplicação (**Programming>>Numeric**), conectando a saída do bloco anterior a uma das entradas e adicione uma constante com valor **360** na outra entrada. Esse bloco matemático transforma a unidade dos dados lidos no encoder para graus ( $^{\circ}$ ).
10. Em seguida, adicione outro bloco de divisão (**Programming>>Numeric**), conectando a saída do bloco matemático anterior à entrada do numerador e uma constante com valor **4096** na entrada do denominador. Esse valor é referente à resolução do encoder utilizado.

#### **SISTEMA DE CONTROLE PROPORCIONAL:**

11. Agora, adicione no **Painel Frontal** um Pointer Slide (**Num Ctrls**) para colocar o valor do ângulo de referência.
12. Para visualizar graficamente a evolução da posição do eixo do motor e a referência desejada, adicione no **Painel Frontal** um **Waveform Chart (Graph Indicators)**. Na janela de **Diagrama de Blocos**, adicione um bloco **Bundle (Programming >> Cluster & Variant)**, conecte a saída do ângulo de referência e a leitura do encoder às entradas **element 0** e **element 1**, respectivamente. Conecte a saída **output cluster** do bloco **Bundle** à entrada do bloco criado pelo **Waveform Chart**.
13. Em seguida, calcule o erro entre o ângulo de referência e o ângulo medido pelo encoder. Adicione um bloco de subtração (**Programming>>Numeric**) para calcular o erro. Conecte na entrada superior (+) do bloco de subtração o valor

---

do ângulo de referência (saída do controlador numérico adicionado no passo 12), e na entrada inferior (-) do bloco de subtração o valor do ângulo obtido a partir do encoder (saída do bloco de divisão adicionado no passo 10).

14. Adicione no **Painel Frontal** um controle numérico (**Num Ctrls**) para colocar o valor do ganho Proporcional  $K_p$ .
15. Multiplique o ganho proporcional  $K_p$  com o valor de erro do ângulo, adicionando um bloco de multiplicação (**Programming**>>**Numeric**). Em uma das entradas do bloco conecte o valor do erro do ângulo (saída do bloco de subtração), e na outra entrada conecte o valor do ganho Proporcional. O valor que sai deste bloco de multiplicação é o valor de tensão que deve ser aplicado ao motor para que seja alcançada a posição desejada.

#### ACIONAMENTO DO MOTOR CC:

16. Para que o motor não receba valores de tensão acima do permitido, adicione o bloco **In Range and Coerce** (**Programming** >> **Comparison**) para que o mesmo opere na faixa de 6 à -6 V. Desse modo, adicione um constante com valor **6** na entrada de **upper limit**, uma constante com valor **-6** na entrada de **lower limit** e na entrada **x** conecte a saída do bloco de multiplicação adicionado no passo anterior.
17. Para controlar a habilitação de envio de tensão para o motor, inicialmente adicione no **Painel Frontal** um **Push Button** (**Buttons**). Agora, na janela de **Diagrama de Blocos**, adicione o bloco **Select** (**Programming** >> **Comparison** >> **Select**). Conecte a saída do bloco **In Range and Coerce**, adicionado no passo 17, na entrada **t**, adicione uma constante com o valor 0 na entrada **f**, e por fim, conecte o **Push Button** na entrada **s**. Assim, se o controlador não estiver habilitado, o sistema enviará tensão nula ao motor.
18. Crie uma variável local do **Push Button** (**Create** >> **Local Variable**) e coloque-a do lado externo da estrutura **Timed Loop**. Adicione à sua entrada uma constante com o valor **F** (**Create** >> **Constant**). Esse procedimento faz com que o controlador esteja sempre desabilitado quando o instrumento virtual é reiniciado.

19. Na lateral esquerda e externa ao bloco **Timed Loop**, adicione outro bloco **Create Channel (MeasurementI/O >> NI-DAQmx)** e selecione o modo **Voltage (AO Voltage>>Analog output>>Voltage)**. Em seguida adicione duas constantes a esse bloco: **minimum value** e **maximun value**, com os valores **-6** e **+6**, que são as limitações da tensão enviada ao motor CC. Na entrada **Physical channels** desse mesmo bloco adicione outra constante, selecionando-a como **I/O Dev1/ao0**.
20. Ainda no exterior do bloco **Timed Loop**, adicione outro bloco **Start (Programming >> MeasurementI/O >> NI-DAQmx)** e conecte-o ao bloco **Voltage** através dos sinais **task out**→**task/channels in** e **error out**→**error in**.
21. Agora, dentro da estrutura **Timed Loop**, adicione um bloco **Write (Measurement I/O >> NI-DAQmx >> Write)**. Selecione o modo **Analog DBL 1Chan 1Samp (Analog >> Single Channel >> Single Sample >> DBL)**. Conecte as entradas **task/channels in** e **error in** às saídas **task out** e **error out**, respectivamente, do bloco **Start**, adicionado no passo anterior. Em seguida conecte a saída do bloco **Select**, adicionado no passo 18, na entrada **data**.
22. Para visualizar graficamente a evolução da tensão no motor, adicione no **Painel Frontal** um **Waveform Chart (Graph Indicators)**. Na janela de **Diagrama de Blocos**, conecte a saída do bloco **Select**.
23. Na lateral direita e externa ao bloco **Timed Loop**, adicione outro bloco **Stop (Measurement I/O >> NI-DAQmx)** e conecte as entradas **task/channels in** e **error in** às saídas **task out** e **error out**, respectivamente, do bloco **Write**, adicionado no passo anterior. Em seguida, adicione outro bloco **Clear (Measurement I/O >> NI-DAQmx)** e conecte-o ao bloco **Stop** da mesma forma.

Seguindo estes passos, o **Painel Frontal** e o **Diagrama de Blocos** deverão ficar similares às Figuras 1.3 e 1.4.

**Dica:** Configure o **Waveform Chart** no **Painel Frontal** para melhor visualização das curvas. Clique com o botão direito no gráfico e selecione para exibir a barra de rolagem no eixo x (**Properties >> Appearance >> [show x scroll bar]**). Em (**Properties >> Scales**), defina o **grid style** para os eixos **x** e **y** selecionando a 3<sup>a</sup>

opção. Retire a seleção no item **autoscale** e defina a escala fixa para o eixo **x** de **0** a **10000** e para o eixo **y** de **-10** a **30**. Selecione o item **scale x ticks**. Novamente clicando com o botão direito no gráfico, em (**Chart History Length**), insira o valor 10000 para determinar o tamanho do buffer de dados do gráfico. Em (**Y scale**), tire a seleção da opção **Loose Fit**.

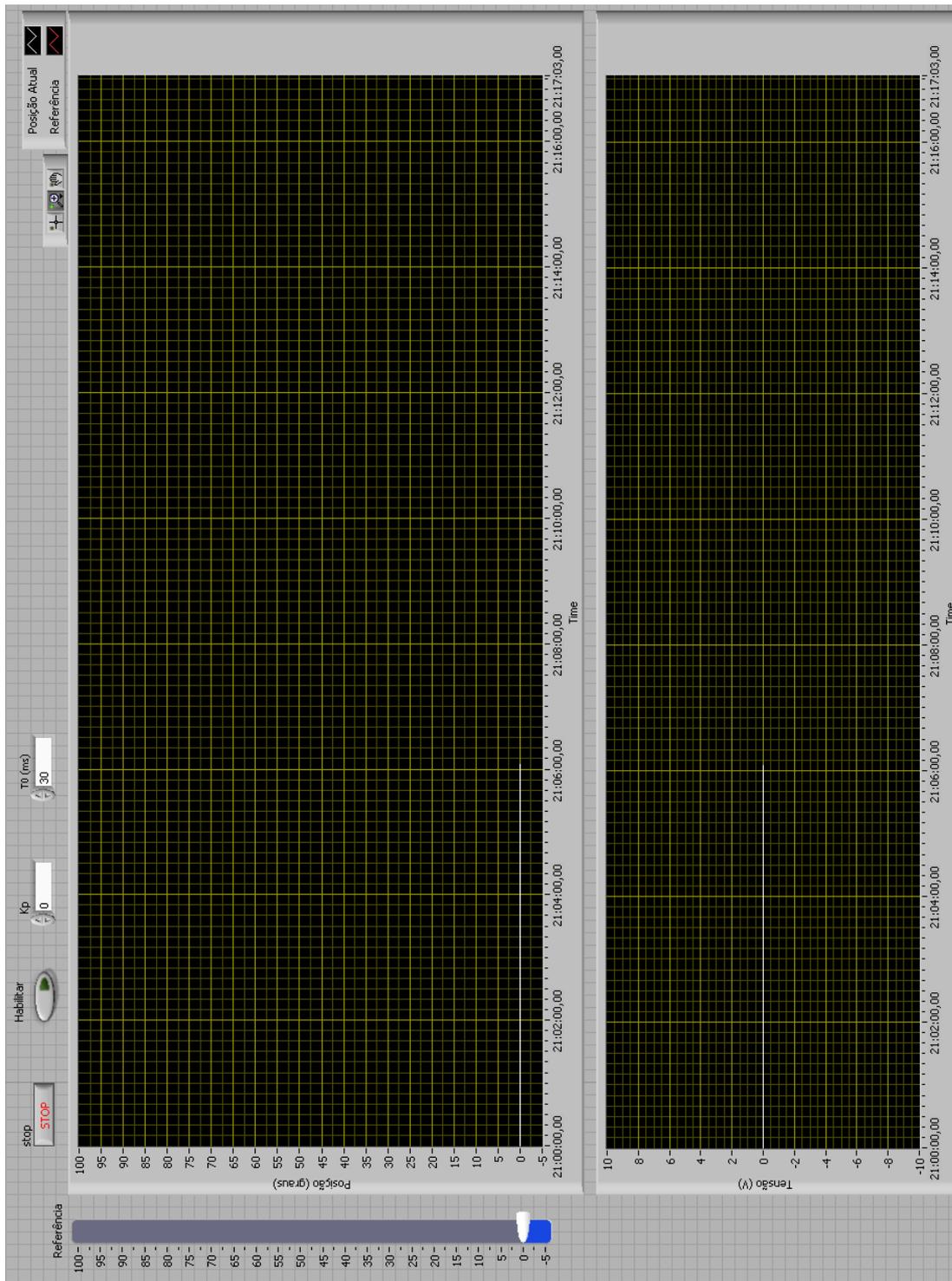


Figura 1.3: Painel Frontal do controlador proporcional.

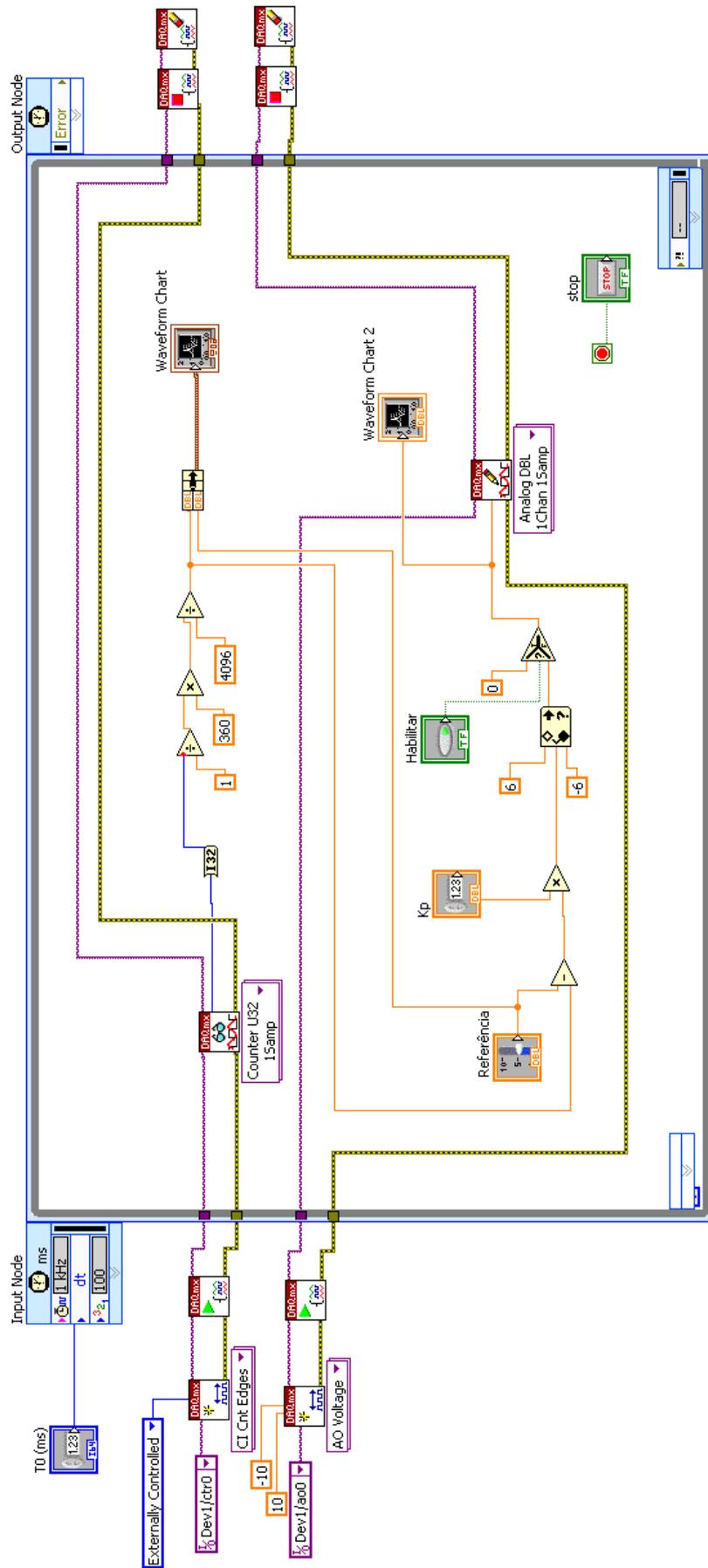


Figura 1.4: Diagrama de Blocos do controlador proporcional.

## 1.5 Apêndice A: Parâmetros do motor (SRV02 - Quanser)

Símbolo	Nome	Valor	Unidades
$K_t$	Constante de Toque do Motor	0.00767	$N.m$
$K_m$	Constante da Força Contra Eletromotriz	0.00767	$V/(rad/s)$
$R_m$	Resistência da Armadura	2.6	$\Omega$
$K_g$	Redução	70	
$B_{eq}$	Coefficiente Viscoso de Amortecimento	$4e^{-3}$	$N.m.s$
$J_m$	Momento de Inércia do Rotor	$3.87e^{-7}$	$kg.m^2$
$J_{eq}$	Momento de Inércia Equivalente da Carga	$2e^{-3}$	$kg.m^2$
$\eta_m$	Eficiência do Motor	0.69	
$\eta_g$	Eficiência da Redução	0.9	

## Prática 01 - Controle Proporcional

Data: \_\_\_\_\_

Integrantes do Grupo:

1: \_\_\_\_\_

2: \_\_\_\_\_

3: \_\_\_\_\_

4: \_\_\_\_\_

5: \_\_\_\_\_

1. Preencha a tabela abaixo

K	MP (%)	$t_r$ (ms)	Erro de regime (°)

2. Analise o efeito do ganho  $K$  no sobre-sinal ( $MP\%$ ) e no tempo de subida ( $t_r$ ).

---

---

---

---

---

---

---

