LAB6 – Alocação de Polos por Realimentação de Estados

PTC 2619/PTC 3418 – Laboratório de Automação
1º semestre de 2017

Bruno Angélico

Laboratório de Automação e Controle
Departamento de Engenharia de Telecomunicações e Controle
Escola Politécnica da Universidade de São Paulo

Objetivo

- Controle Digital por Realimentação de Estados :
  - Projetar controle por alocação de polos via realimentação linear de estados, considerando todos os estados medidos.
  - Projetar observador de estados e efetuar novamente o controle com realimentação de estados estimados.

Introdução

- Considere o sistema LIT:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ y &= \mathbf{C}\mathbf{x} + \mathbf{D}u\end{aligned}$$

os autovalores da matriz A são os polos do sistema.

- Controlabilidade: o sistema será dito controlável no instante t_0 se existir uma entrada capaz de transferir o sistema de $x(t_0)$ para qualquer outro estado, em um intervalo de tempo finito.

Introdução

Para o sistema ser completamente controlável, a matriz

$$\mathcal{C} = \left[\begin{array}{c|c|c|c} \mathbf{B} & \mathbf{AB} & \cdots & \mathbf{A}^{n-1}\mathbf{B} \end{array}\right]$$

deve possuir posto igual a *n*.

- Observabilidade: O sistema é observável em t_0, se com o sistema em $x(t_0)$, for possível determinar esse estado a partir da observação da saída durante um intervalo de tempo finito.

Introdução

Para o sistema ser completamente observável, a matriz

$$\mathcal{O} = \left[\begin{array}{c} \mathbf{C} \\ \hline \mathbf{CA} \\ \hline \vdots \\ \hline \mathbf{CA}^{n-1} \end{array}\right]$$

deve possuir posto igual a *n*.

Introdução

- O equivalente discreto do sistema é dado por:

$$\begin{aligned} \mathbf{x}[k+1] &= \mathbf{\Phi}\mathbf{x}[k] + \mathbf{\Gamma}u[k] \\ y &= \mathbf{C}\mathbf{x}[k] + \mathbf{D}u[k] \end{aligned} \qquad \begin{aligned} \mathbf{\Phi} &= e^{\mathbf{A}T} \\ \mathbf{\Gamma} &= \int_0^{\infty} e^{\mathbf{A}\eta}\, d\eta\, \mathbf{B} \end{aligned}$$

O comando `c2d` do MATLAB faz o serviço:
`[Phi,Gamma] = c2d(A,B,Ts).`

- Os conceitos de controlabilidade, observabilidade para sistemas discretos são idênticos ao caso contínuo.

Realimentação de Estados

- Considere o sistema:

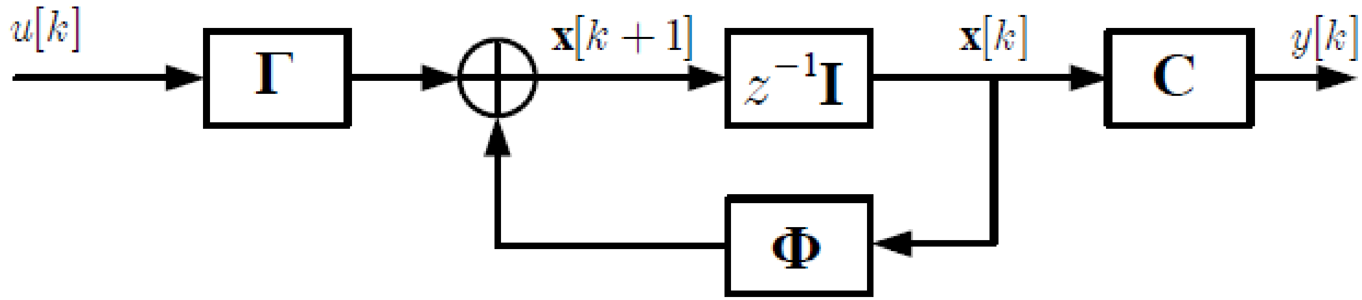
$$
\begin{aligned}
\mathbf{x}[k+1] &= \Phi \mathbf{x}[k] + \Gamma u[k] \\
y &= C\mathbf{x}[k]
\end{aligned}
$$

Ao invés de realimentar a saída *y*, que tal utilizar a retroação das variáveis de estado?

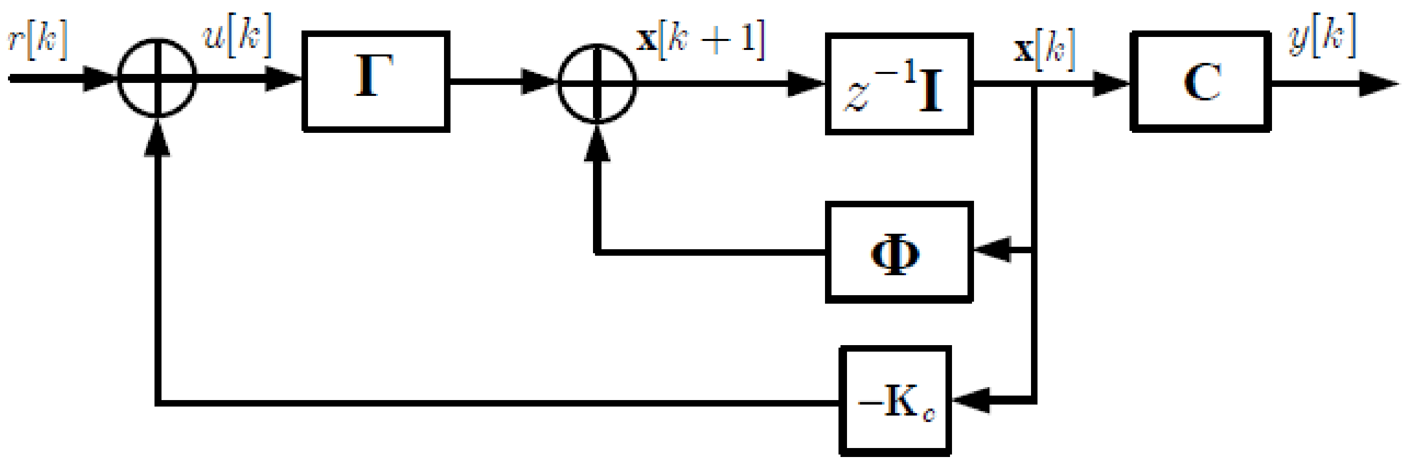
- Realim. linear de estados: $u[k]=-K_c\, x[k]$.
- Em malha fechada:

$$
\mathbf{x}[k+1] = \Phi \mathbf{x}[k] - \Gamma \mathbf{K}_c \mathbf{x}[k] \quad \Longrightarrow \quad \mathbf{x}[k+1] = \;\; \mathbf{\Phi} - \mathbf{\Gamma K}_c \;\; \mathbf{x}[k]
$$

Realimentação de Estados



(a)



(b)

Realimentação de Estados

- Os polos de M.F. são os autovalores de $\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K}_c$
- Para sistemas SISO, a fórmula de Ackermann pode ser utilizada para calcular $\boldsymbol{K}_c$ que aloque os polos de acordo com um vetor $\boldsymbol{p}_c$.
- No MATLAB:

```
K_c = acker(Phi, Gamma,p_c)
```

ou

```
K_c = place(Phi, Gamma,p_c)
```

- OBS: O comando `place` também funciona para sistemas MIMO.

Projeto de Observadores

• Algumas das variáveis de estado podem ser inacessíveis. Assim, as estimativas de estado $\hat{\mathbf{x}}[k]$ são obtidas através de um modelo do sistema, caso o mesmo seja observável.

• A lei de controle fica: $\mathbf{u}[k] = -\mathbf{K}_c \hat{\mathbf{x}}[k]$

• A equação do estimador preditor é dada por:

$$
\begin{aligned}
\hat{\mathrm{x}}[k+1] &= \Phi\hat{\mathrm{x}}[k] + \Gamma u[k] + \mathrm{K}_o(y[k] - \hat{y}[k]) \\
&= \Phi\hat{\mathrm{x}}[k] + \Gamma u[k] + \mathrm{K}_o\mathrm{C}(\mathrm{x}[k] - \hat{\mathrm{x}}[k])
\end{aligned}
$$

Projeto de Observadores

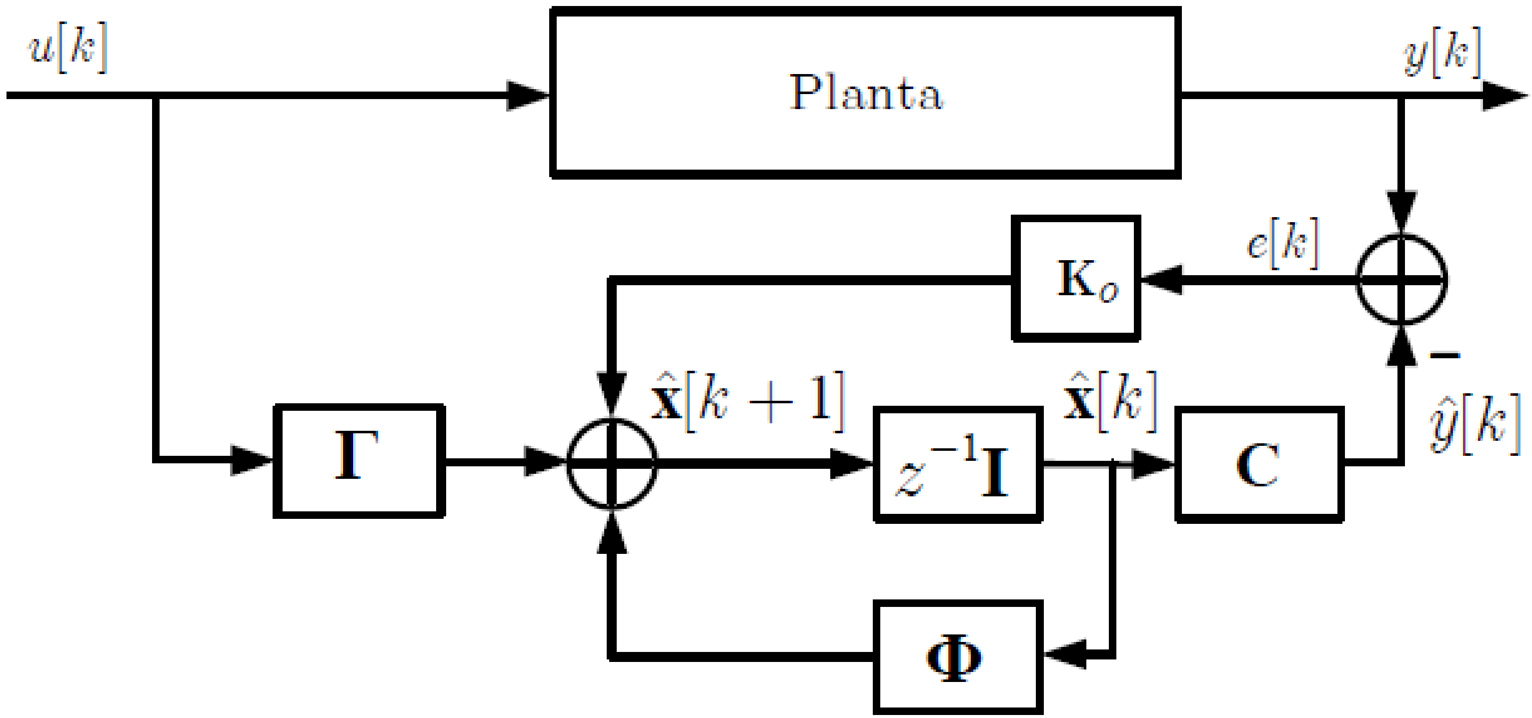


Figura 6.3: Diagrama do estimador em malha fechada.

Projeto de Observadores

- onde $\boldsymbol{K_o}$ é o vetor de os ganhos do estimador.

Ao definir $\tilde{\mathbf{x}}[k] = \mathbf{x}[k] - \hat{\mathbf{x}}[k]$, tem-se:

$$\begin{aligned} \tilde{\mathbf{x}}[k+1] &= \mathbf{x}[k+1] - \hat{\mathbf{x}}[k+1] = \mathbf{\Phi}(\mathbf{x}[k] - \hat{\mathbf{x}}[k]) - \mathbf{K}_o\mathbf{C}\tilde{x}[k] \\ &= (\mathbf{\Phi} - \mathbf{K}_o\mathbf{C})\tilde{x}[k] \end{aligned}$$

- Como os autovalores de $\mathbf{\Phi} - \mathbf{K}_o\mathbf{C}$ são os mesmos de $\mathbf{\Phi}^{\mathrm{T}} - \mathbf{C}^{\mathrm{T}}\mathbf{K}_o^{\mathrm{T}}$, então, para alocar os polos do observador de acordo com o vetor $\boldsymbol{p}_o$, pode-se fazer:

Projeto de Observadores

```
K_o = acker(Phi', Gamma',p_o)'
```

ou

```
K_o = place(Phi', Gamma',p_o)'
```

Inserção de Integradores

- Para o rastreamento da referência ocorrer adequadamente, muitas vezes é necessária a inserção de integradores na malha de controle

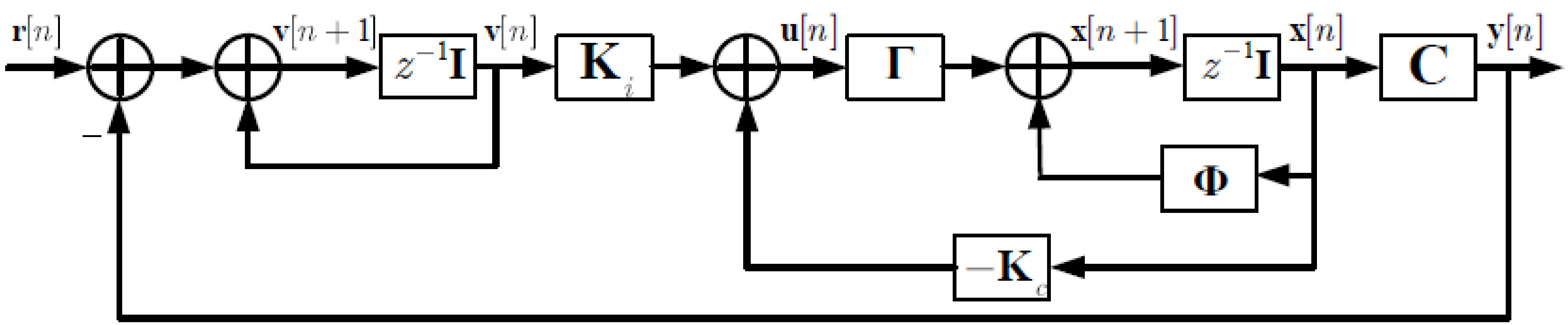


Figura 6.4: Servossistema com realimentação de estados e controle integral.

Inserção de Integradores

A equação de estados do integrador inserido é dada por:

$$\mathbf{v}[n+1] = \mathbf{v}[n] + \mathbf{r}[n] - \mathbf{y}[n] \Rightarrow \mathbf{v}[n+1] = \mathbf{v}[n] + \mathbf{r}[n] - \mathbf{C}\mathbf{x}[n]$$

A equação de estados do sistema em malha fechada é dada por:

$$\mathbf{x}[n+1] = (\mathbf{\Phi} - \mathbf{\Gamma}\mathbf{K})\,\mathbf{x}[n] + \mathbf{\Gamma}\mathbf{K}_i\mathbf{v}[n]$$

Tem-se, portanto, a seguinte equação para o sistema aumentado:

Inserção de Integradores

$$\begin{bmatrix} \mathbf{x}[n+1] \\ \mathbf{v}[n+1] \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi} - \mathbf{\Gamma K} & \mathbf{\Gamma K}_i \\ -\mathbf{C} & \mathbf{I}_{m\times m} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}[n] \\ \mathbf{v}[n] \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{I}_{m\times m} \end{bmatrix} \cdot \mathbf{r}[n],$$

ou seja,

$$\begin{bmatrix} \mathbf{x}[n+1] \\ \mathbf{v}[n+1] \end{bmatrix} = \left(\underbrace{\begin{bmatrix} \mathbf{\Phi} & \mathbf{0}_{k\times m} \\ -\mathbf{C} & \mathbf{I}_{m\times m} \end{bmatrix}}_{\hat{\mathbf{\Phi}}} - \underbrace{\begin{bmatrix} \mathbf{\Gamma} \\ \mathbf{0}_{m\times m} \end{bmatrix}}_{\hat{\mathbf{\Gamma}}} \cdot \underbrace{\begin{bmatrix} \mathbf{K} & -\mathbf{K}_i \end{bmatrix}}_{\hat{\mathbf{K}}} \right) \cdot \begin{bmatrix} \mathbf{x}[n] \\ \mathbf{v}[n] \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{k\times m} \\ \mathbf{I}_{m\times m} \end{bmatrix} \cdot \mathbf{r}[n]$$

Assim, basta determinar o ganho $\hat{\mathbf{K}}$ que aloca os polos para o sistema aumentado.

Atividades Prévias

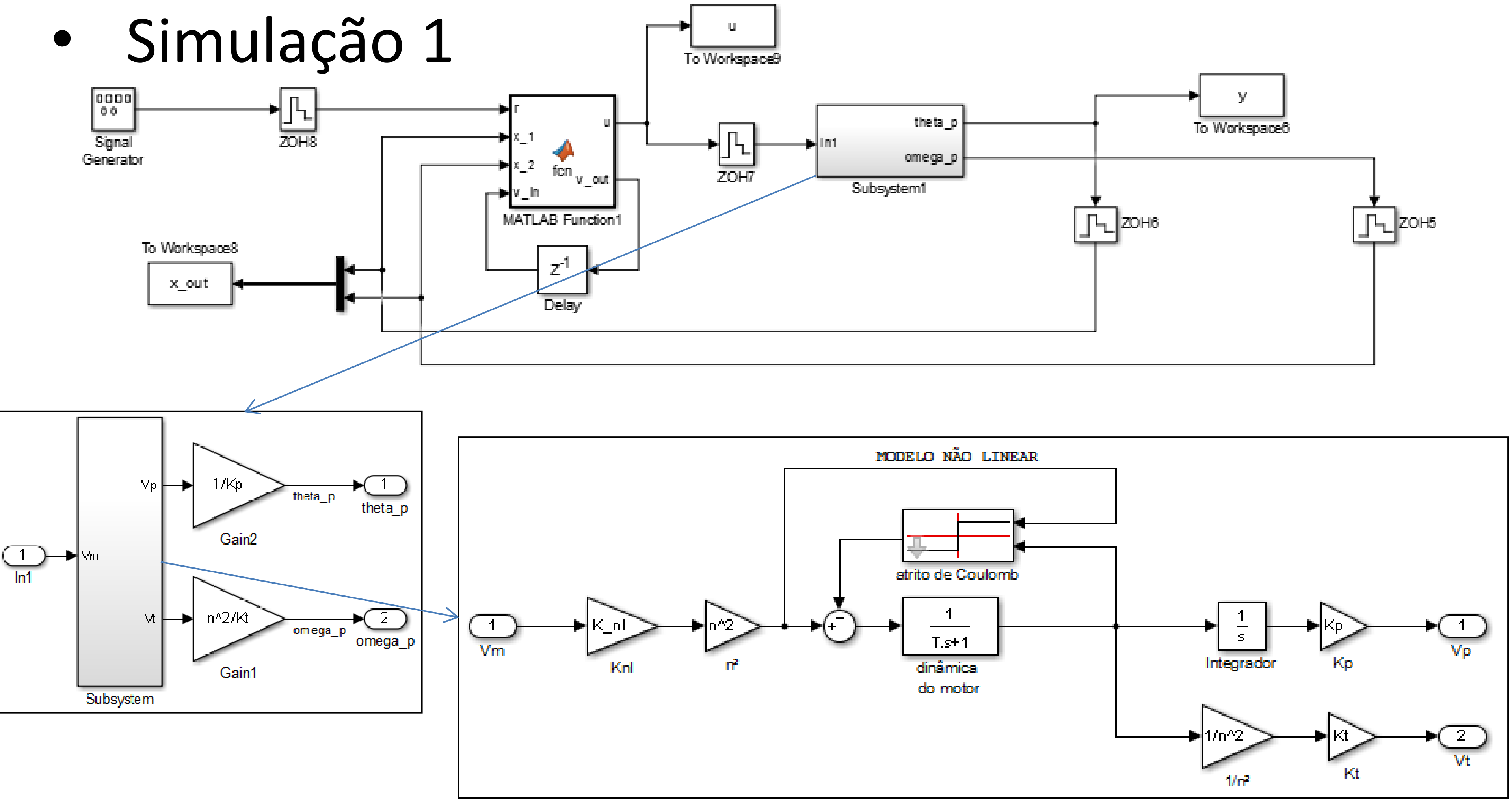
- Encontre uma representação em espaço de estados para o servomecanismo, $x_1 = \theta_p$ e $x_2 = \omega_p$. Verifique se o sistema é controlável e observável.
- Obtenha o equivalente discreto p/ Ts = 1/20 s.
- Projete um controlador tal que os polos de MF satisfaçam : $\xi = 0,707$ e $\omega_n = 10$ rad/s. Escolha o polo do integrador em $s = - 10$.

Atividades Prévias

- Projete um observador de estados para a planta não aumentada , considerando a seguinte especificação para os polos do observador: ξ = 0,707 e ω_n = 30 rad/s.

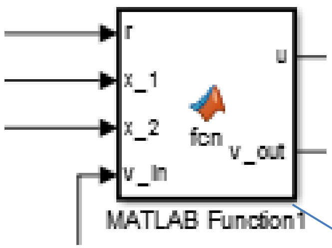
Atividades Práticas

- Simulação 1



Atividades Práticas

- Simulação 1




```
function [u, v_out] = fcn(r, x_1, x_2, v_in, Kc, Ki)
%#eml
y=x_1;
u = -(Kc(1)*x_1 + Kc(2)*x_2)+ Ki*v_in;
v_out = v_in+(r-y);
```

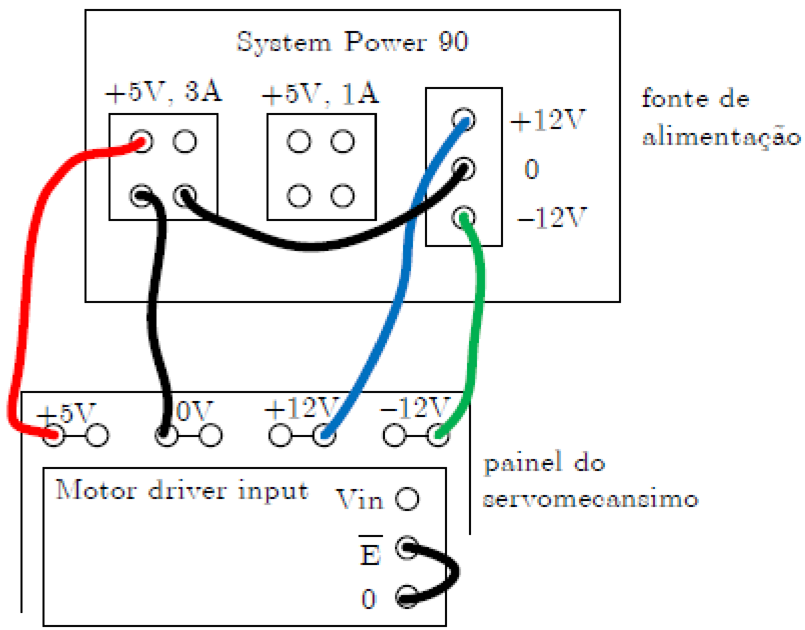
OBS: em versões mais novas do MATLAB, pode-se criar v como variável global e defini-la como *data store memory*, no lugar de criar v_out e v_in.

Atividades Práticas

- Simulação 2: adaptar a simulação 1, sendo que agora a Matlab Function que implementa o controlador deve receber a saída e estimar os estados e realimentar os estados estimados.
- Para o observador de estados, deve-se implementar a primeira linha da Eq. (6.17) da apostila: $\hat{\mathbf{x}}[k+1] \;=\; \Phi\hat{\mathbf{x}}[k] + \Gamma u[k] + \mathbf{K}_o(y[k] - \hat{y}[k])$
- Observe que o estimador é preditor, ou seja, ele estima os estados para o controle da próxima iteração.

Atividades Práticas

- Teste prático:
  - Implemente os controladores diretamente no template `prog4.m`
  - Ligações:



Atividades

- Prática:
  - Ligações: