



# **Tecniche di Controllo e Diagnosi**

## **Cenni su Matlab (e toolbox Control Systems)**

**Dott. Ingg. Marcello Bonfè e Silvio Simani**

Dipartimento di Ingegneria - Università di Ferrara

Tel. +39 0532 974839 / 974844

E-mail: [marcello.bonfe@unife.it](mailto:marcello.bonfe@unife.it) / [silvio.simani@unife.it](mailto:silvio.simani@unife.it)

# Matlab: interfaccia principale

The screenshot shows the MATLAB 7.12.0 (R2011a) interface. The main window is divided into several panes:

- Current Folder:** A file browser showing the current directory (C:\Programmi\MATLAB) and its contents.
- Command Window:** The central area for entering and executing MATLAB commands. It shows the prompt `>>`.
- Workspace:** A table listing the variables currently in the workspace. The variables and their values are:

Name	Value
A	[1,2,3;4,5,6;7,8,9]
B	[0;0;1]
C	[1,0,0]
K	<3x1 sym>
ans	<3x3 sym>
k1	<1x1 sym>
k2	<1x1 sym>
k3	<1x1 sym>

- Command History:** A list of the most recently entered commands, including `ctrb(A,B)`, `C=[1 0 0]`, `obsv(A,C)`, `syms k1 k2 k3`, `K=[k1;k2;k3]`, and `A+K*C`.

Red text annotations are overlaid on the image:

- Browser per le cartelle su disco** (Browser for folders on disk) is placed over the Current Folder pane.
- FINESTRA COMANDI (ambiente di lavoro)** (COMMAND WINDOW (work environment)) is placed over the Command Window pane.
- Workspace (variabili)** (Workspace (variables)) is placed over the Workspace pane.
- Ultimi comandi digitati** (Most recently entered commands) is placed over the Command History pane.

# Matlab: definizione di variabili, vettori e matrici

Definire variabile scalare

```
>> x = 3
```

Definire vettore riga (1×3)

```
>> x = [1 2 3]
```

Idem, ma senza echo dell'output

```
>> x = [1 2 3];
```

Definire vettore colonna (3×1)

```
>> x = [1; 2; 3]
```

(oppure >> x = [1 2 3]')

Definire matrice 3×4

```
>> A = [1 2 3 4; 5 6 7 8; 9 10 11 12]
```

Accedere / modificare elemento di riga 2 e colonna 1

```
>> A(2,1) = 0
```

# Matlab: operazioni su matrici

- ➡ Le "solite" operazioni matematiche:  $+$ ,  $-$ ,  $*$ ,  $/$ ,  $^$
- ➡ **Es.** `>> A^3` (potenza di matrice, solo se quadrata!)
- ➡ Precedute dal punto, sono eseguite elemento per elemento anziché in senso matriciale/vettoriale
- ➡ Operazioni specifiche per matrici / vettori:
  - Trasposta: `A'`
  - Determinante: `det(A)`
  - Inversa: `inv(A)`
  - Autovalori: `eig(A)`
  - Rango: `rank(A)`
  - Polinomio caratteristico: `poly(A)`
  - Esponenziale di matrice: `expm(A)`
  - Radici di un polinomio: `roots(x)` (x vettore dei coeff.)

# Matlab: test di Controllabilità / Osservabilità

➡ Grazie al Control Systems Toolbox, il test è eseguibile semplicemente lanciando i comandi:

```
>> P=ctrb(A,B)
```

per il test di controllabilità, e:

```
>> Q=obsv(A,C)
```

per il test di osservabilità

# Matlab: test di Controllabilità / Osservabilità

► In Matlab:

```
>> A=[0 0 0 0 -1; 0 -3 1 0 2; 2 0 1 0 0; -1 0 0 1 1; 0 0 0 0 2]
```

```
>> B=[1; 0; 1; -1; 1]
```

```
>> C=[0 0 -1 0 0]
```

```
>> Q=obsv(A,C)
```

```
Q =
```

```
    0    0   -1    0    0
   -2    0   -1    0    0
   -2    0   -1    0    2
   -2    0   -1    0    6
   -2    0   -1    0   14
```

```
>> Qt=Q'
```

**NOTA:** con quest'ultima operazione, la matrice di osservabilità viene posta in modo da rendere l'analisi omogenea rispetto a quella di controllabilità, in modo cioè che si debba osservare quali COLONNE risultino indipendenti.

# Matlab: test di controllabilità / osservabilità

- ➔ Dall'analisi delle colonne della matrice  $Q_t$ , si può vedere che solo le prime tre risultano indipendenti, pertanto la forma minima del sistema è di ordine 3 ➔  $\text{rank}(Q_t)$

# Matlab: Progetto di retroazione Stato-Ingresso

➡ In Matlab:

```
>> A = [-6 -11 -6; 1 0 0; 0 1 0]
```

```
>> B=[1; 0; 0] % oppure B=[1 0 0]'
```

➡ Il progetto della retroazione uscita-ingresso richiede di determinare gli autovalori della matrice  $A + B^*K$ , con  $K$  in questo caso di dimensione  $1 \times 3$

➡ Con la funzione di Matlab **place** si risolve il problema dell'assegnamento degli autovalori di  $A - B^*K$ :

```
>> K = - place(A,B,[-4 -5 -6])
```

```
>> eig(A + B*K) % Verifica
```

# Matlab: Progetto dell'Osservatore dello Stato

➡ In Matlab:

```
>> A = [-6 -11 -6; 1 0 0; 0 1 0]
```

```
>> B = [1; 0; 0]
```

```
>> C = [0 0 1]
```

➡ Il progetto dell'osservatore dello stato richiede di determinare gli autovalori della matrice  $A + L^*C$ , con  $L$  in questo caso di dimensione  $3 \times 1$

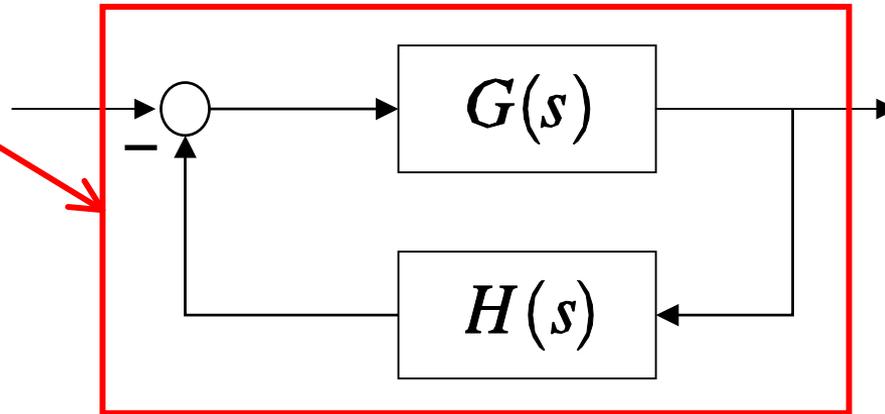
➡ Ancora con la funzione di Matlab **place** si risolve il problema dell'assegnamento degli autovalori di  $A + L^*C = A + ((L^* C)')' = A + (C' * L')' = A' + C' * L'$ , ovvero:

```
>> L = - place(A',C',[-8 -9 -10])'
```

```
>> eig(A + L*C) % Verifica
```

# Matlab: funzioni del Control Systems Toolbox

```
>> Gc1 = feedback(G,H)
```



```
>> Gp = parallel(G1,G2) ← parallelo di FdT
```

```
>> Gs = series(G1,G2) ← serie di FdT
```

```
>> step(G) ← grafica la risposta al gradino
```

```
>> impulse(G) ← grafica la risp. impulsiva
```

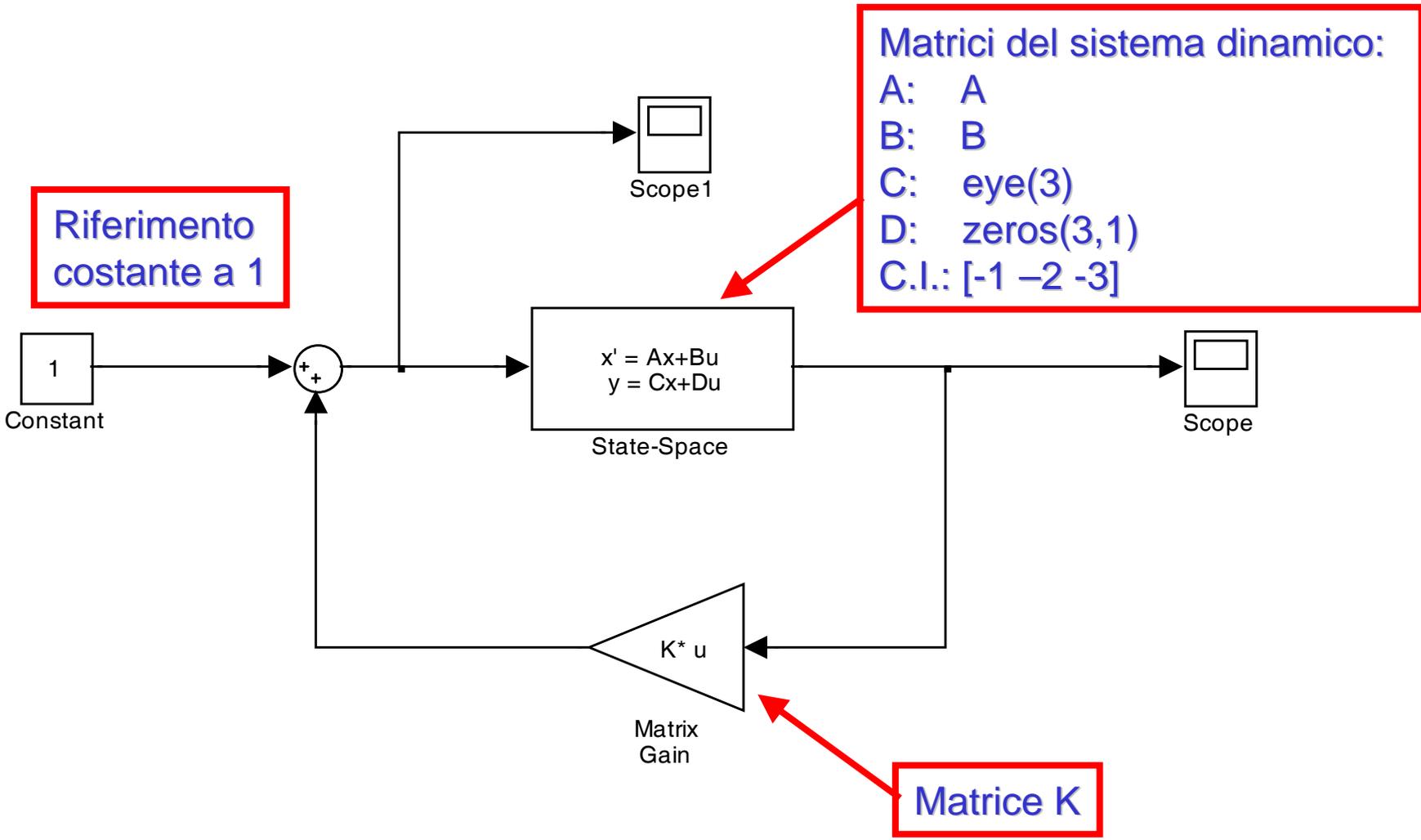


## **CENNI SU MATLAB**

**FINE**

**... Simulink:**

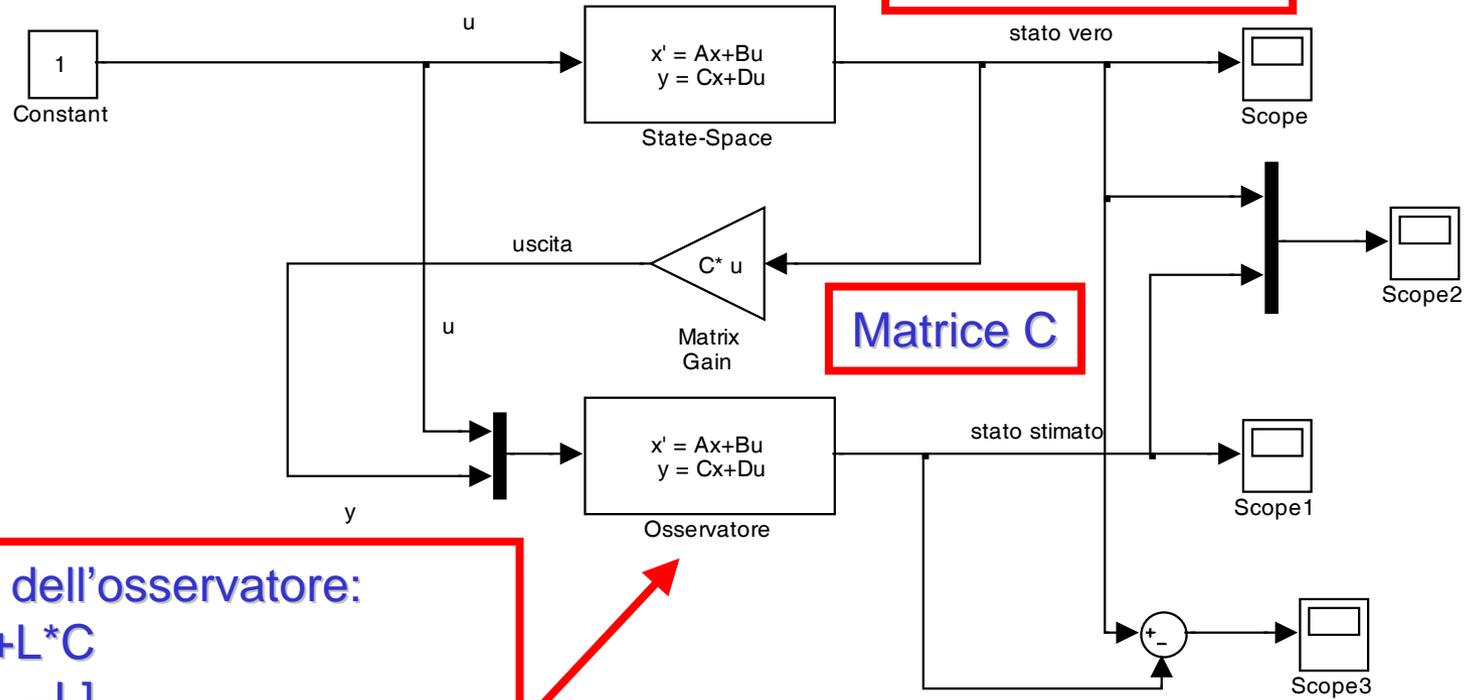
# Retroazione stato-ingresso in Simulink



# Osservatore Dinamico dello Stato in Simulink

Riferimento  
costante a 1

A: A  
B: B  
C: eye(3)  
D: zeros(3,1)  
C.l.: [-1 -2 -3]



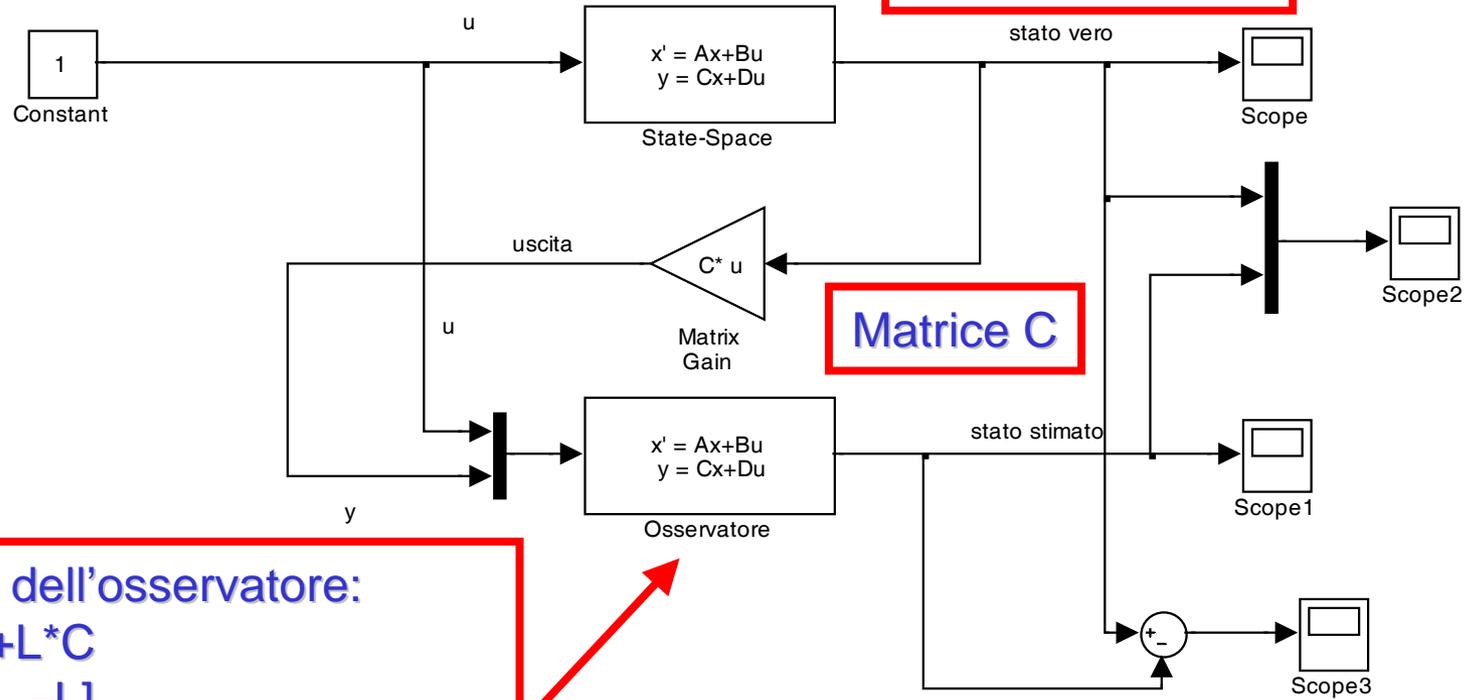
Matrice C

Matrici dell'osservatore:  
A: A+L\*C  
B: [B -L]  
C: eye(3)  
D: zeros(3,2)  
C.l.: 0

# Osservatore Dinamico dello Stato in Simulink

Riferimento  
costante a 1

A: A  
B: B  
C: eye(3)  
D: zeros(3,1)  
C.l.: [-1 -2 -3]



Matrice C

Matrici dell'osservatore:  
A: A+L\*C  
B: [B -L]  
C: eye(3)  
D: zeros(3,2)  
C.l.: 0

# Osservatore Dinamico dello Stato in Simulink



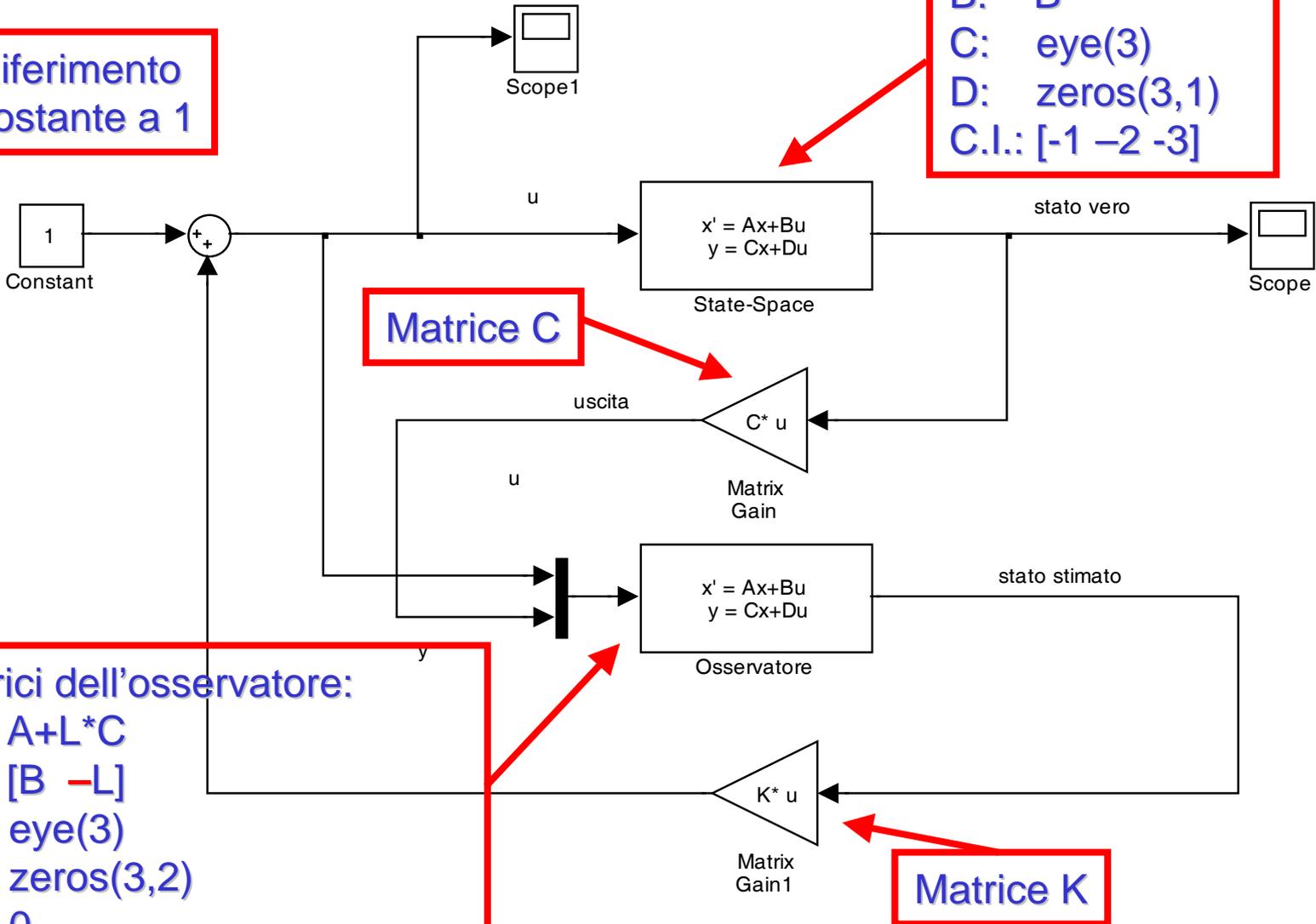
Riferimento  
costante a 1

A: A  
B: B  
C: eye(3)  
D: zeros(3,1)  
C.l.: [-1 -2 -3]

Matrice C

Matrici dell'osservatore:  
A: A+L\*C  
B: [B -L]  
C: eye(3)  
D: zeros(3,2)  
C.l.: 0

Matrice K





## **CENNI SU SIMULINK**

**FINE**

**... Modello NL...**

# Modello Simulink per il Pendolo Inverso (1)



```
M = 1  
m = 0.1  
l = 1  
g = 9.8
```

**Inizializzazione variabili**

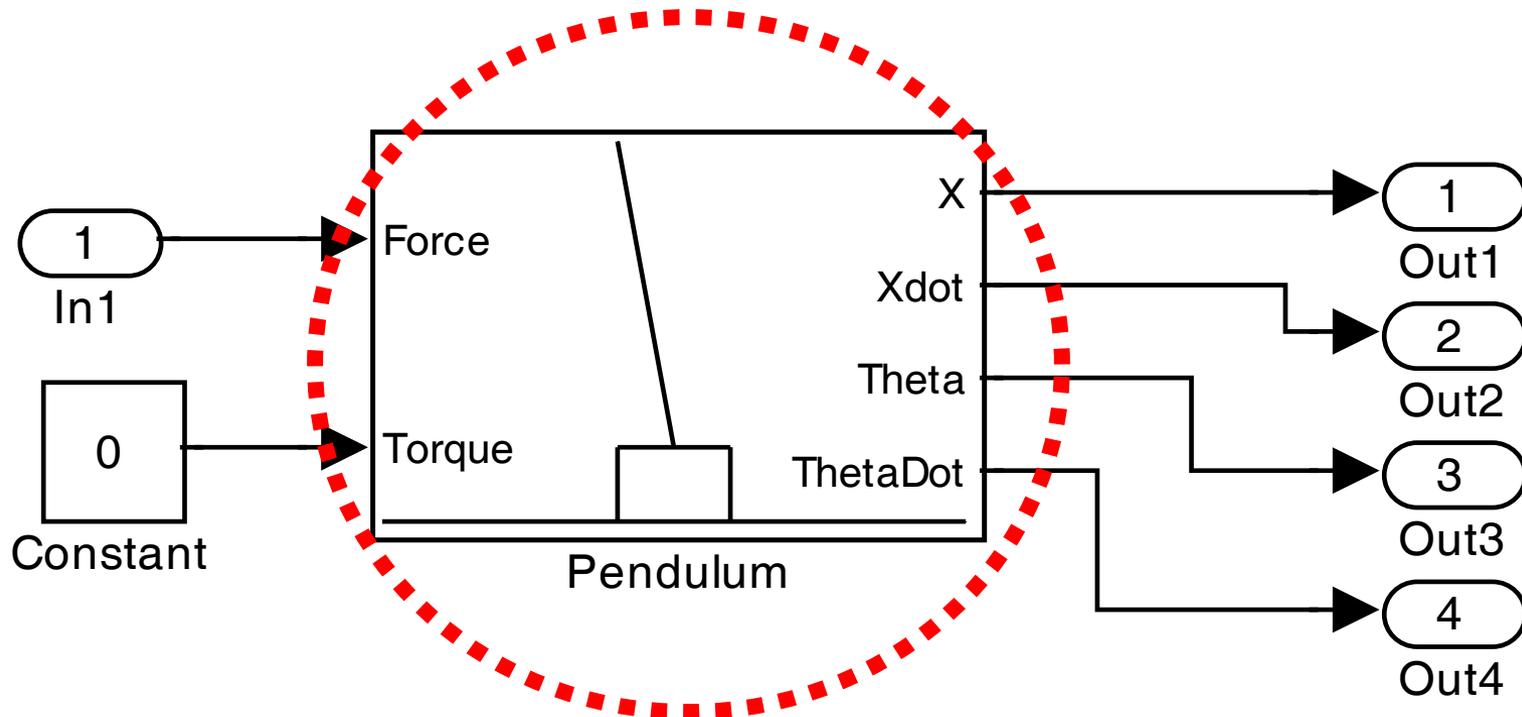
```
Fmax = 20  
Botta = 3  
PeriodoBotta = 16  
PWBotta = 1  
DelayBotta = 6
```

**Parametri ingresso**

```
x0 = 0.2  
theta0 = 0.55  
StepX = 0.8  
TstepX = 15  
Tfin = 30
```

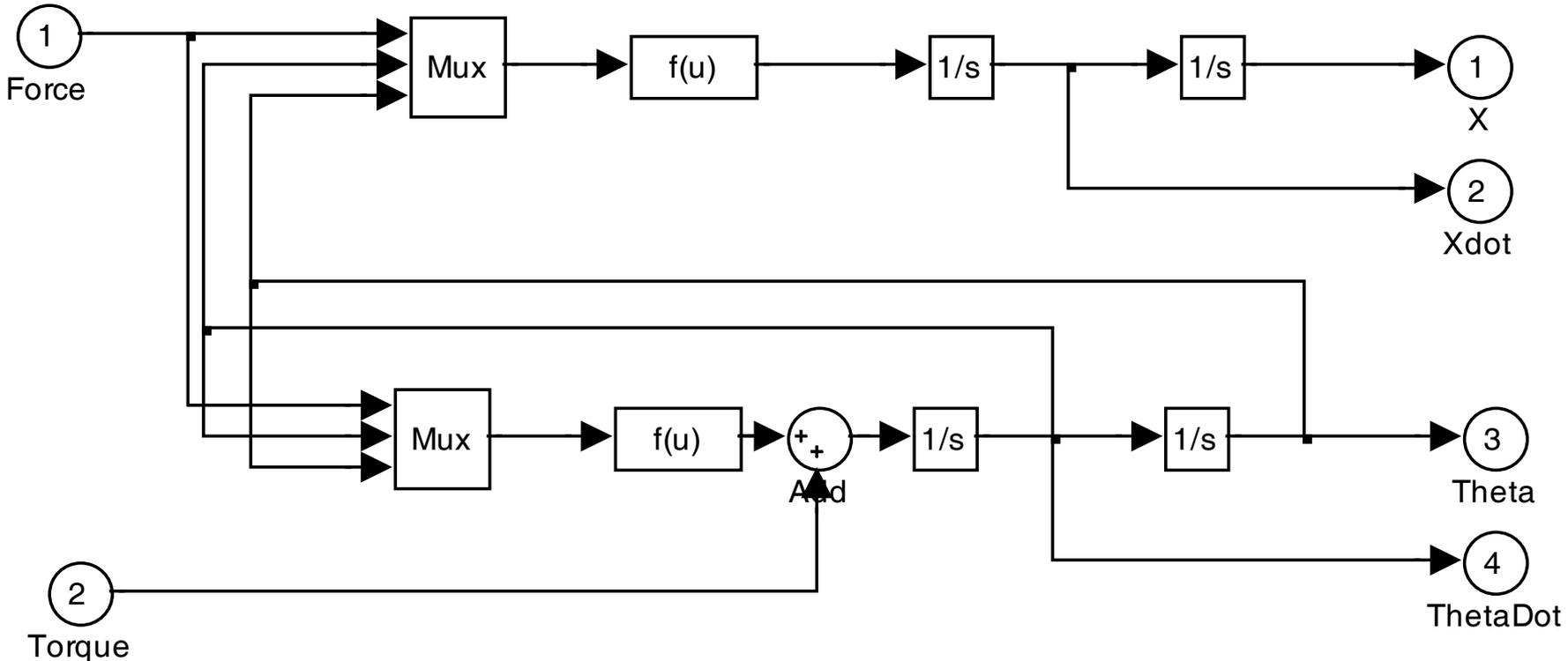
**Condizioni iniziali e  
variabili della simulazione**

# Modello Simulink per il Pendolo Inverso (2)



**Maschera del blocco... Andare dal menù in Edit-> Look under mask...**

# Modello Simulink per il Pendolo Inverso (3)



**Modello non lineare come schema a blocchi di Simulink**

# Modello Matematico del Pendolo Inverso

$$\mathbf{x} = [x_1 \quad x_2 \quad x_3 \quad x_4]^T = [x \quad \dot{x} \quad \vartheta \quad \dot{\vartheta}]^T$$

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{mLx_4^2 \sin x_3 - mg \sin x_3 \cos x_3 + u}{M + m \sin^2 x_3} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{-mLx_4^2 \sin x_3 \cos x_3 + (M + m)g \sin x_3 - u \cos x_3}{(M + m \sin^2 x_3)L} \end{array} \right.$$

**f(u)**

