

---

# Introduzione al Simulink



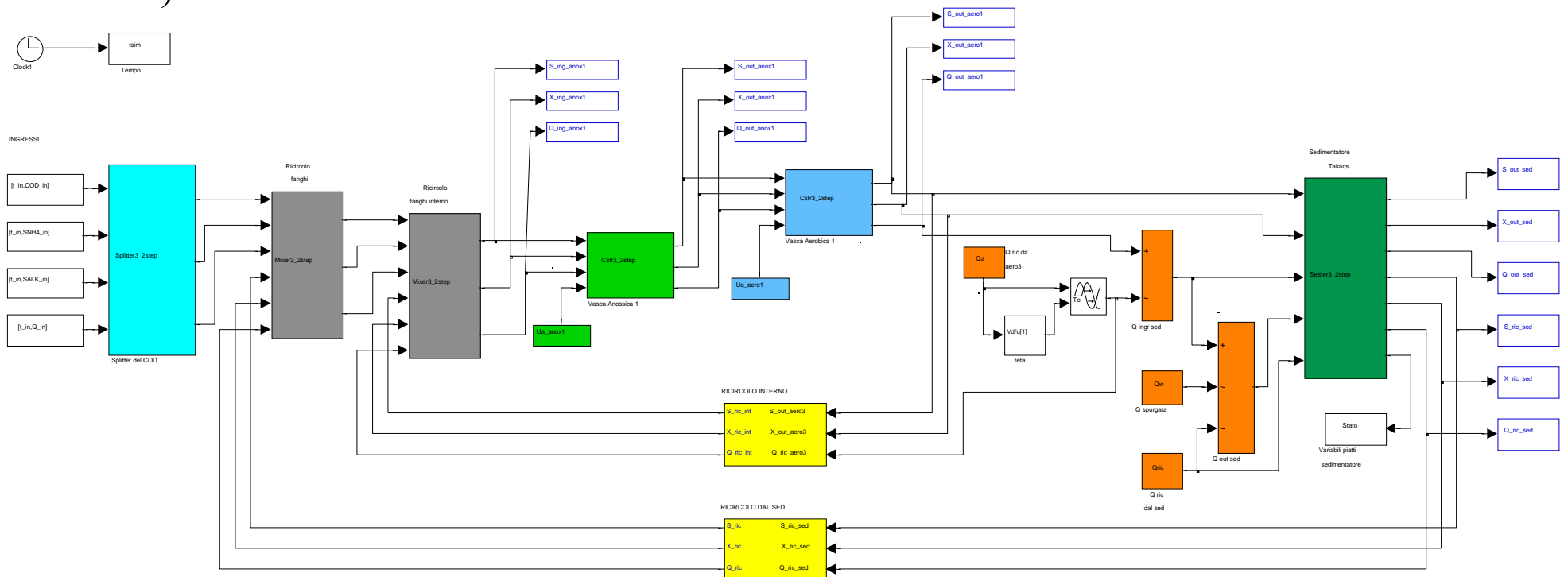
# L'ambiente Simulink

---

- ❑ Simulink è un **ambiente grafico** per la simulazione di sistemi complessi
- ❑ Simulink è composto da una **libreria di blocchi** che descrivono elementi statici e dinamici elementari
- ❑ L'utente compone lo schema a blocchi del sistema da simulare mediante l'interconnessione dei blocchetti elementari
- ❑ Simulink genera automaticamente le equazioni e risolve il problema numerico di simulazione desiderato.
- ❑ I modelli costruiti in Simulink possono essere modelli gerarchici: ogni blocco del sistema può essere a sua volta un sottosistema complesso.
- ❑ Simulink interagisce con Matlab attraverso il Workspace; i modelli Simulink possono contenere variabili del Workspace; allo stesso modo il risultato delle simulazioni può essere esportato nel Workspace e analizzato con Matlab.

# La forza del simulink stà nella modularietà

- Modularità: un sistema complesso può essere rappresentato come interconnessione di svariati sottosistemi. Ciascuno di essi può a sua volta (in maniera ricorsiva) essere composto da sottosistemi di complessità via via inferiore, sino ad arrivare a blocchi che descrivano sottosistemi elementari, cioè descritti da una sola relazione matematica (equazione differenziale oppure alle differenze, un'equazione algebrica ecc.)



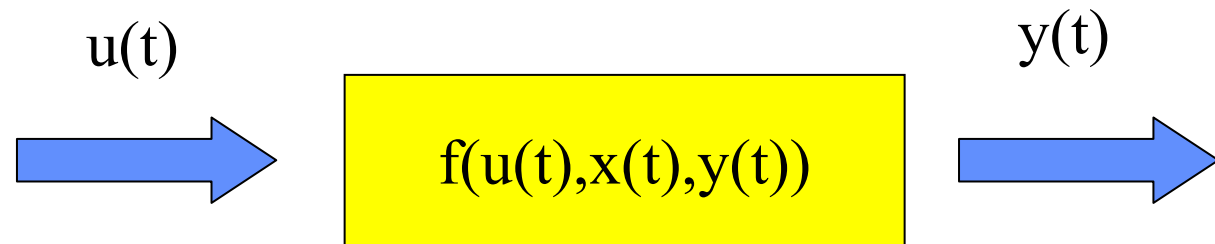
# Blocco elementare

---

Ogni blocco rappresenta un sistema dinamico elementare.

Un blocco comprende le seguenti grandezze (non sempre sono tutte presenti):

- un insieme di ingressi
- un insieme di stati
- un insieme di uscite



Le grandezze di uscita sono una funzione del tempo e delle grandezze di ingresso e degli stati del sistema.

$$y(t) = f(u(t), x(t), t)$$

Si parla di blocco continuo se l'uscita è una funzione continua del tempo. Si parla di blocco discreto se il segnale di uscita è valutato solo in specifici istanti.

# Metodi di integrazione numerica

---

- I metodi numerici per la soluzione di una generica equazione differenziale individuano all'interno dell'intervallo di integrazione un certo numero di “istanti di integrazione”  $T_0=t_1 < t_2 \dots < t_n=T_f$  e calcolano la soluzione  $x(t_i)$  in corrispondenza di questi punti.

$$\dot{x} = f(x, t) \quad t \in [T_0, T_f] \quad x(T_0) = x_0$$

- La soluzione all'istante  $t_{i+1}$  è determinata sulla base delle soluzioni calcolate negli istanti precedenti, con una formula del tipo:

$$x(t_{i+1}) = x(t_i) + h \Delta(t_i, x, h, f)$$

- con  $h_i = t_{i+1} - t_i$  è il passo di integrazione
- $\Delta$  è un'opportuna funzione

# Metodi di integrazione numerica

---

I vari metodi di integrazione si distinguono per una diversa funzione  $\Delta$  e un'opportuna scelta di  $h$ .

- ❑ Metodi a passo fisso:  $h$  è costante
- ❑ Metodi a passo variabile:  $h$  è scelto in modo opportuno ad ogni passo.

Trovata la soluzione  $x_{i+1}$ , viene valutato l'errore nei confronti della soluzione esatta. Se questo supera il limite prefissato si riduce  $h$  e si ricalcola la soluzione. Se l'errore è inferiore al passo successivo  $h$  viene incrementato. In questo modo i tempi di calcolo si riducono.

# Metodi di integrazione numerica

---

Simulink possiede diversi metodi di integrazione, sia a passo fisso che a passo variabile.

Metodi a passo variabile

- ode45(default) non appropriato per sistemi stiff(RungeKutta4,5)
- ode23più veloce, ma meno preciso di ode45, anche per stiff
- ode113 per soluzioni accurate, ma più lento
- ode15s efficiente per sistemi stiff
- ode23s meno preciso, ma più efficiente del precedente
- ode23t per sistemi moderatamente stiff
- ode23tb meno preciso di ode15s
- discrete (default) per sistemi discreti

Metodi a passo fisso

- ode5(default) versione a passo fisso della ode45
- ode4 Metodo di RungeKuttadel 4 ordine
- ode3 versione a passo fisso della ode23
- ode1 metodo di Eulero

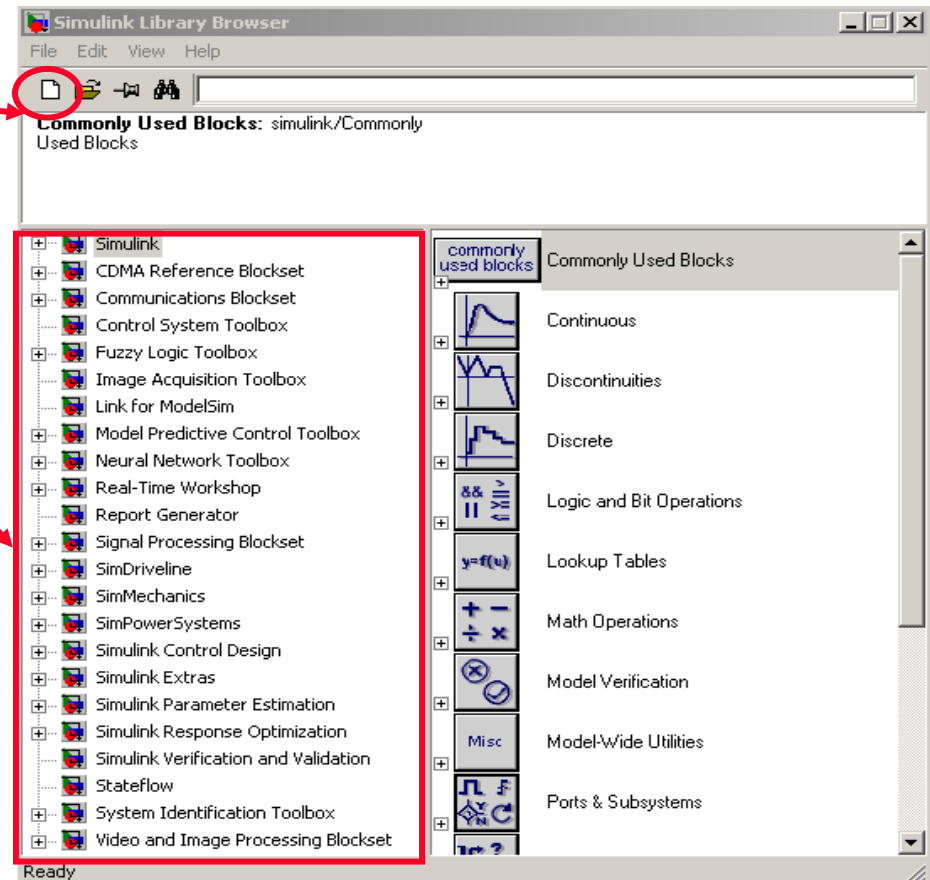
# L'interfaccia grafica

- ❑ Digitando simulink sul prompt di Matlab appare la libreria grafica
- ❑ Da qui è possibile creare un nuovo modello (foglio bianco) e comporre il sistema da simulare mediante i diversi blocchi elementari già disponibili.

Crea nuovo modello

Libreria grafica  
Comprende anche blocchi  
elementari già pronti all'uso

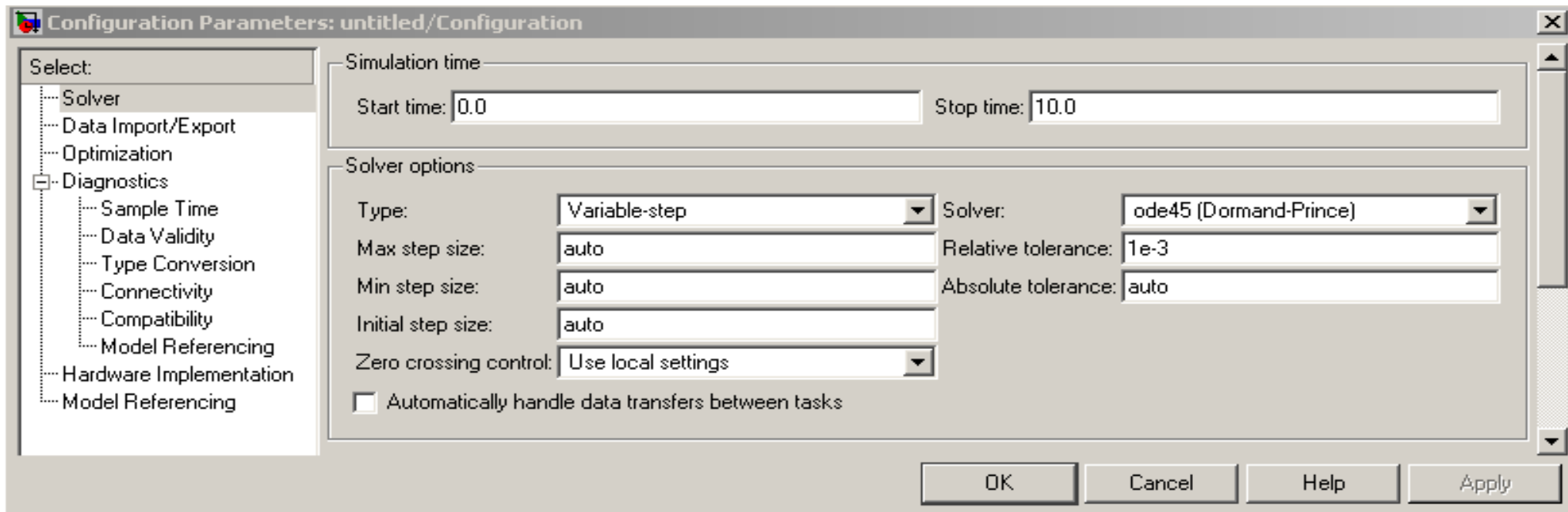
I blocchi-base sono raggruppati in categorie. La maggior parte dei blocchi che useremo è contenuta nella categoria Simulink



# Parametri della simulazione

L'utente deve definire:

- Istanti di inizio e fine della simulazione;
- Tipo di solutore numerico (se il problema richiede metodi particolari);
- Parametri del solutore.



# Parametri della simulazione

---

- ❑ **Start time** istante di inizio
- ❑ **Stop time** istante di fine
- ❑ **Type** metodo a passo fisso o a passo mobile

In presenza di un metodo a passo variabile

- ❑ **Max step size**: è la massima ampiezza del passo di integrazione. Se è auto allora è  $1/50$  dell'intervallo di integrazione. Occorre fissarlo minore di  $0.1 \tau_{\max}$  (costante di tempo più grande).
- ❑ **Min step size**: è la minima ampiezza del passo di integrazione. Occorre fissarlo minore di  $0.01 \tau_{\min}$  (costante di tempo più piccola).
- ❑ **Relative tolerancee Absolute tolerance**: definiscono il massimo errore relativo e assoluto.

In presenza di un metodo a passo fisso

- ❑ **Fixedstep size**: è l'ampiezza del passo di integrazione. Il valore di default auto è pari ad un cinquantesimo dell'intervallo di integrazione.

# Simulink lanciato da Matlab

---

- ❑ Un programma simulink può essere lanciato da uno script o una function (m-file)
- ❑ I vantaggi consistono nel settaggio di tutte le variabili nello script stesso
- ❑ La sintassi è molto semplice: supponiamo di avere un simulink che si chiama “Modello.mdl”
- ❑ Lo script in matlab dovrà contenere il seguente comando:

```
[T,X,Y]=sim('Modello',Timespan,options,UT)
```

Dove

T restituisce il vettore dei tempi dopo la simulazione

X restituisce la matrice degli stati

Y restituisce le uscite

Timespan [Tfinal]

[Tinit Tfinal]

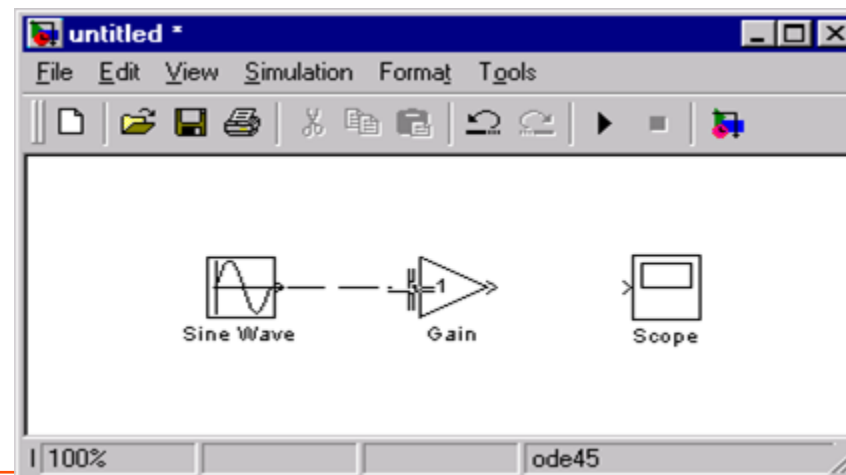
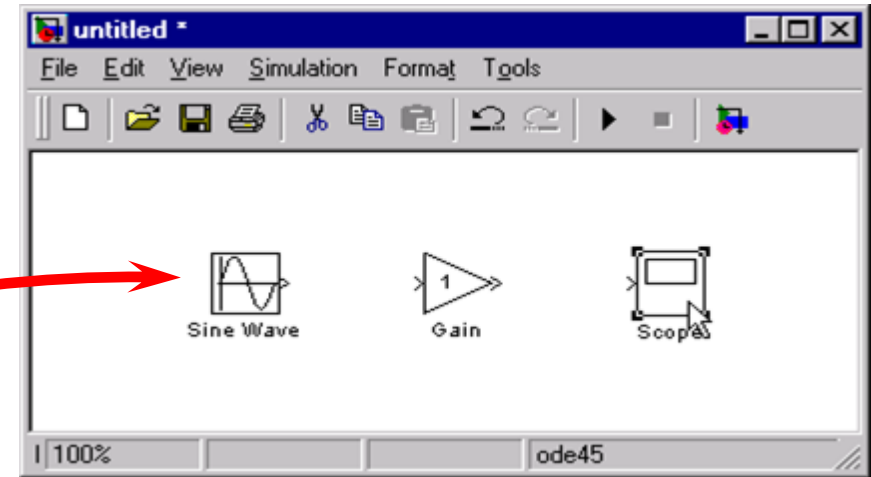
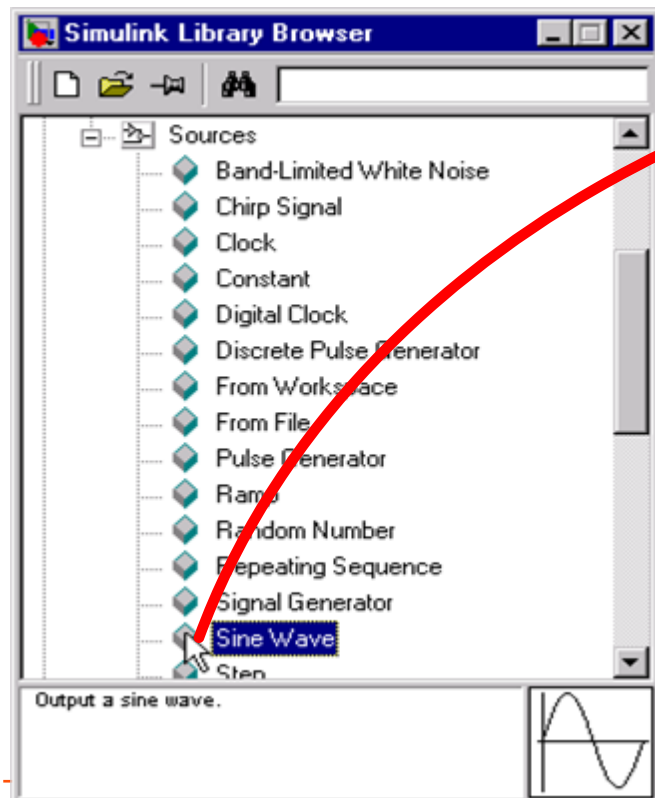
Options da simset

UT (opzionale) input esterni

# Aggiungere blocchi al diagramma e collegarli

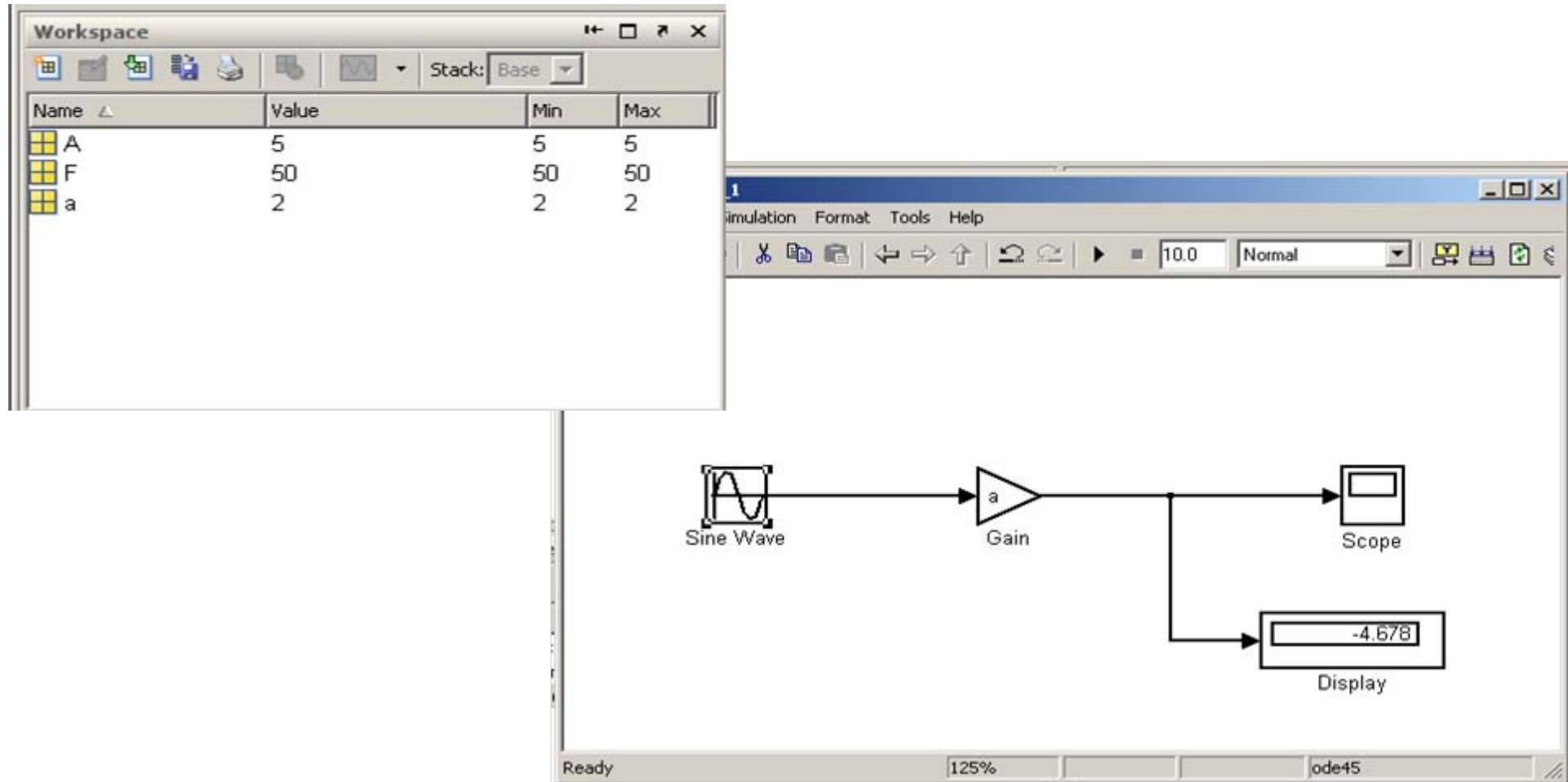
>> Crea\_mod\_ex\_1

Mettere i blocchi necessari nella finestra diagramma dei blocchi di Simulink e connettere le porte.



# Definizione Variabili

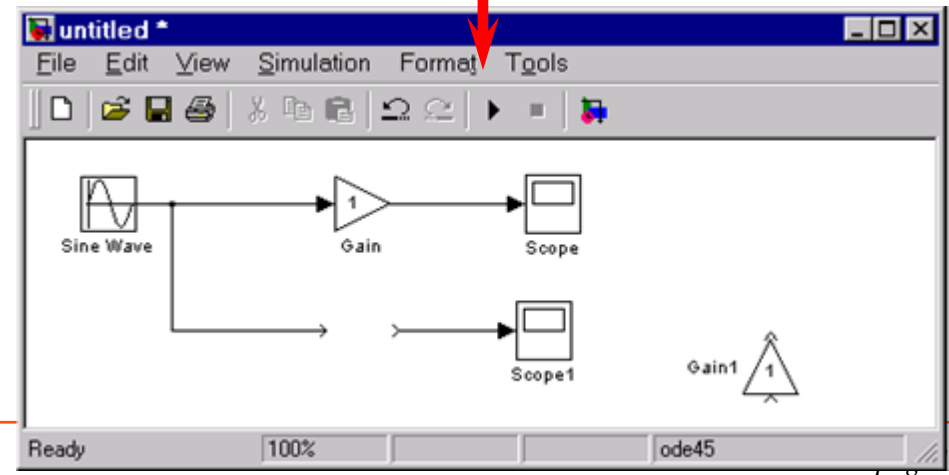
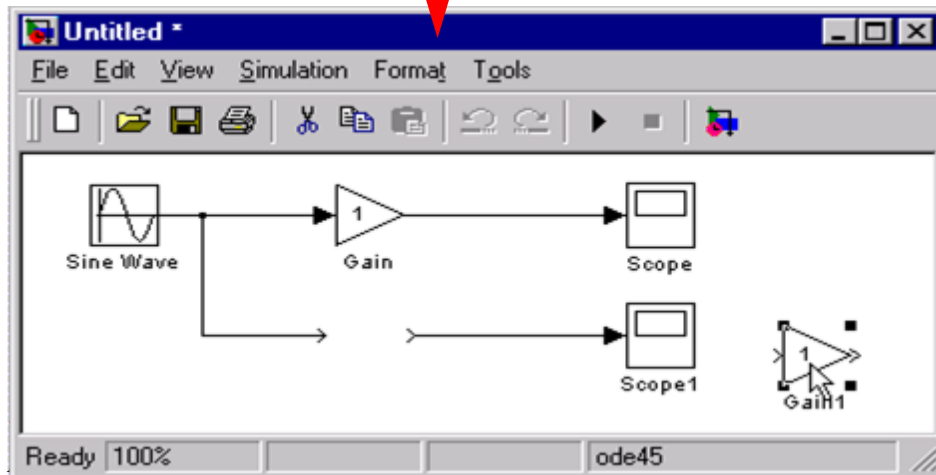
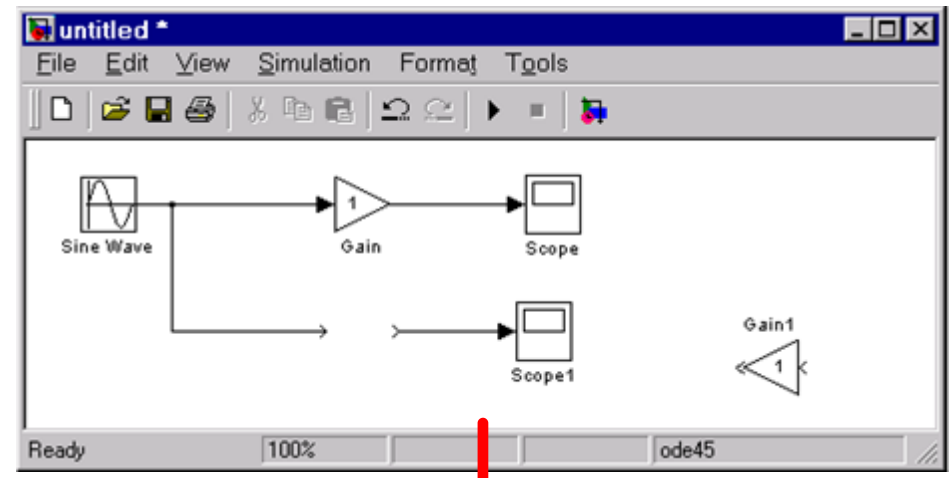
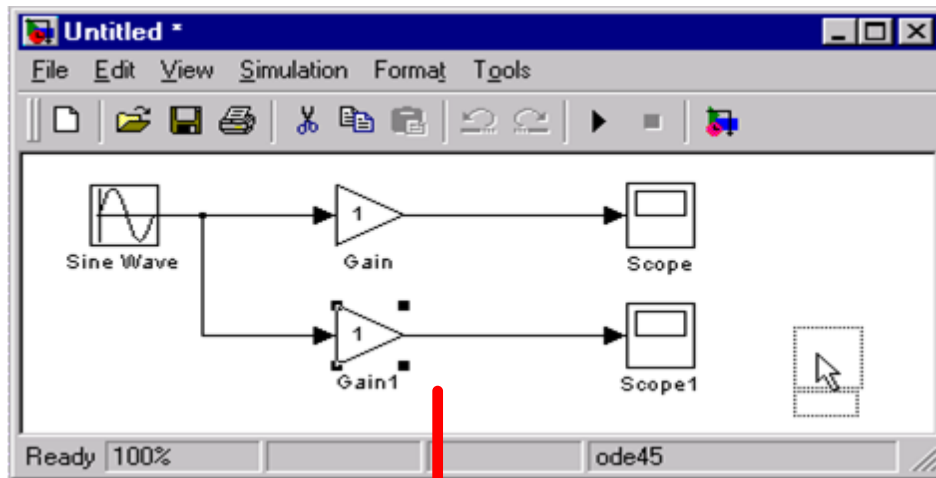
- Sia Matlab che Simulink “vedono” lo stesso Workspace



# Disconnettere, ruotara e invertire i blocchi

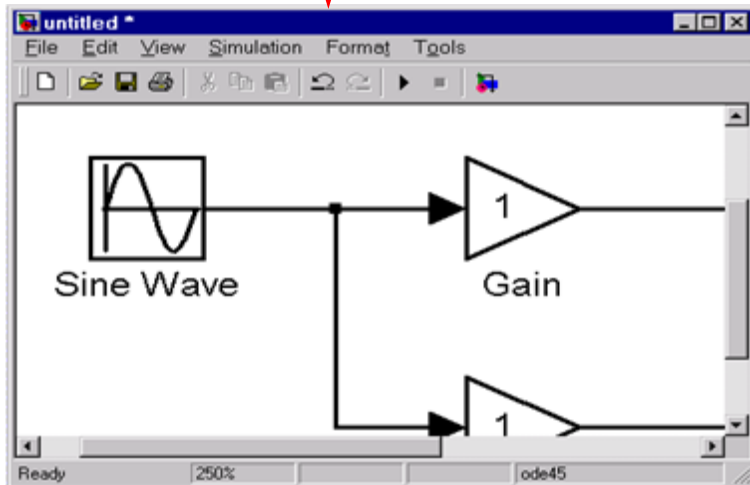
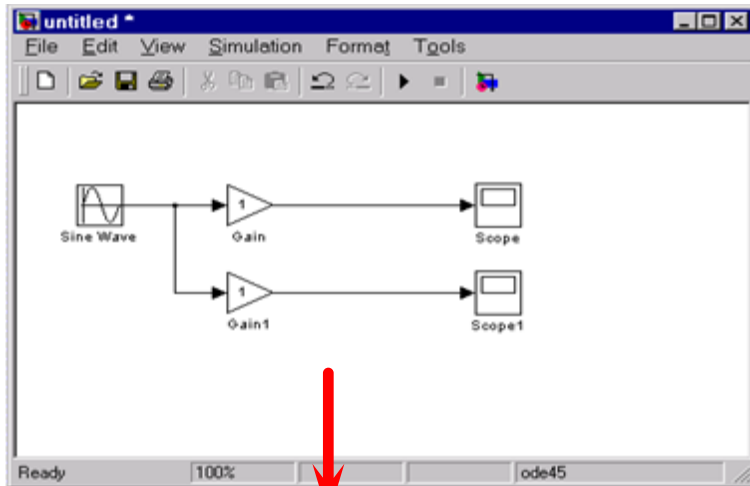
Premere il tasto Shift e spostare il blocco con il mouse

Ctrl-F per invertire, Ctrl-R per ruotare

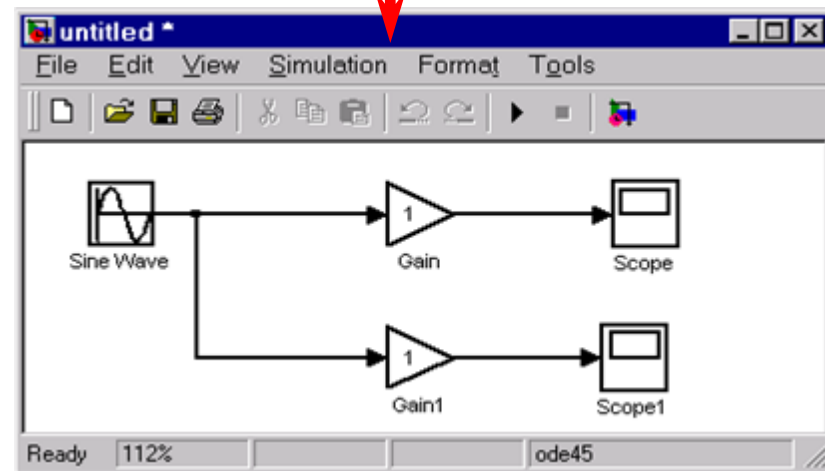
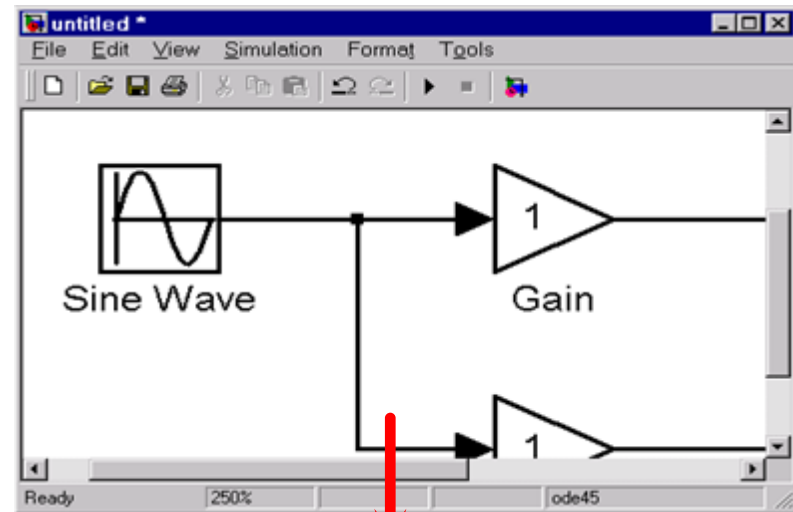


# Zoom del diagramma a blocchi

Selezionare Zoom dal View menu



Premere la spacebar per ottimizzare la vista del diagrammato



# Settare le caratteristiche dei blocchi

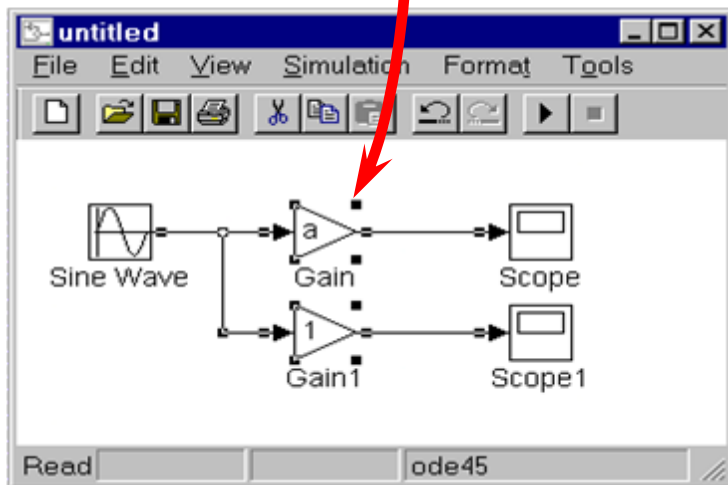
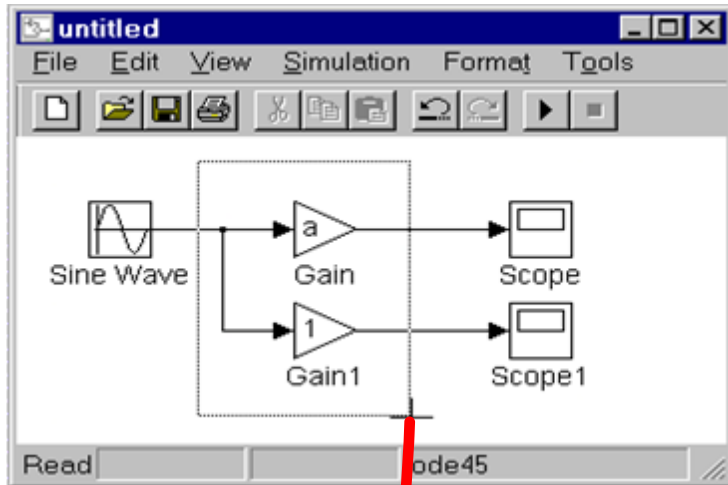
Selezionare i blocchi disegnando una box intorno ad essi

Tasto destro sul blocco per vedere il menu di formattazione

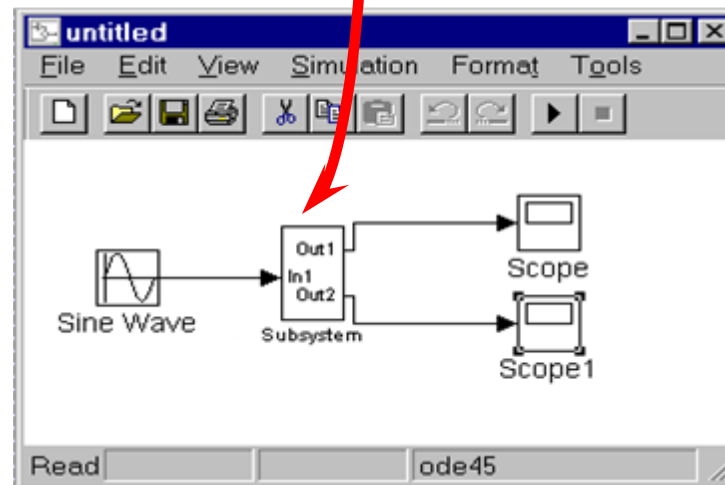
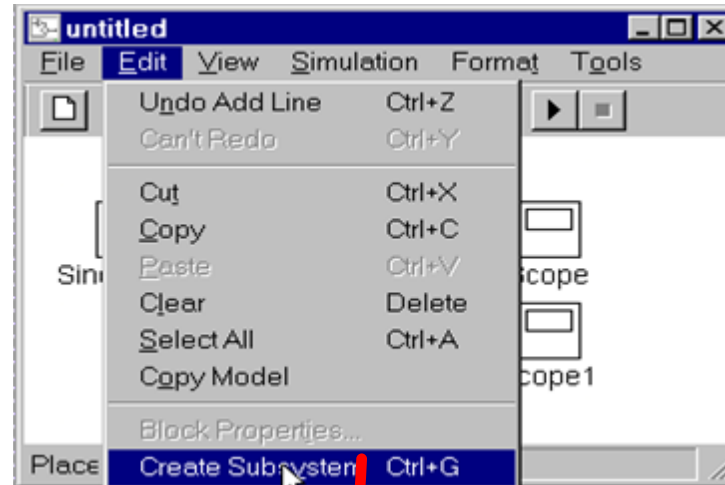
The image consists of three screenshots of a MATLAB/Simulink workspace window titled 'untitled \*'. The top-left screenshot shows a Simulink model with a 'Sine Wave' block, two 'Gain' blocks (labeled 'a' and '1'), and two 'Scope' blocks. A red dashed box is drawn around the 'Sine Wave' block. A red arrow points from this box to the middle screenshot. The middle screenshot shows a context menu opened over the 'Sine Wave' block. The menu items are: Cut, Copy, Clear, Look Under Mask, Go To Library Link, Break Library Link, Format (highlighted), Foreground Color, Background Color (highlighted), Block Parameters, and Block Properties. A sub-menu for 'Background Color' is open, showing options: Black, White, Red, and Green (highlighted). A red arrow points from the 'Background Color' option to the right screenshot. The right screenshot shows the same Simulink model, but the 'Sine Wave' block and both 'Scope' blocks are now colored green. The status bar at the bottom of each window shows 'ode45' and '100%' zoom.

# Creare sottosistemi

Seleziona i componenti da inserire nel subsystem

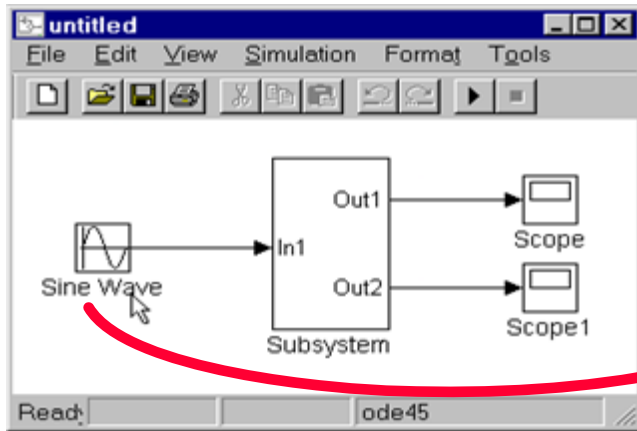


Selezionare 'Create Subsystem' dal menu 'Edit'

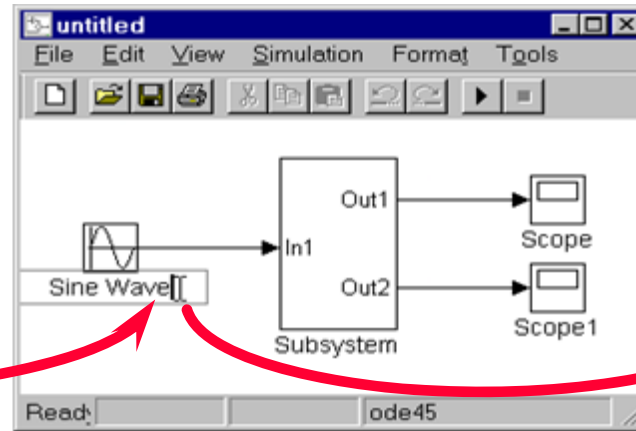


Doppio Clic su Subsystem per la visualizzazione e dei blocchi interni

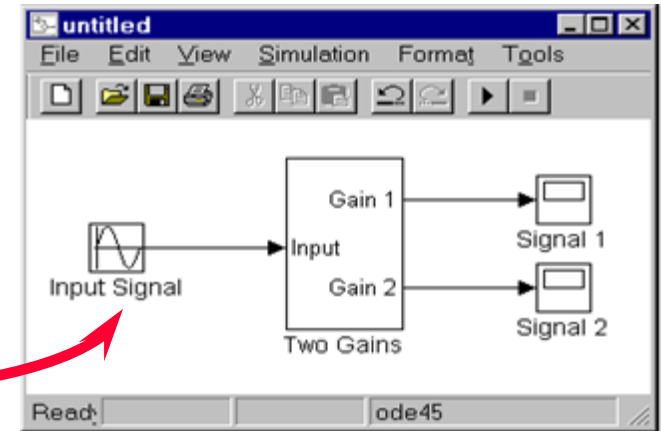
# Cambiare nomi a i blocchi



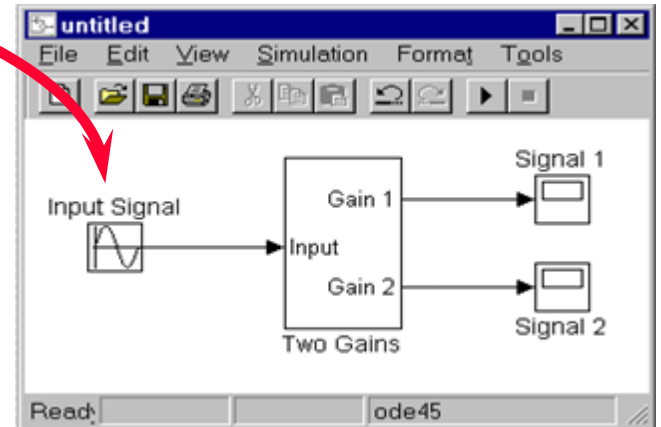
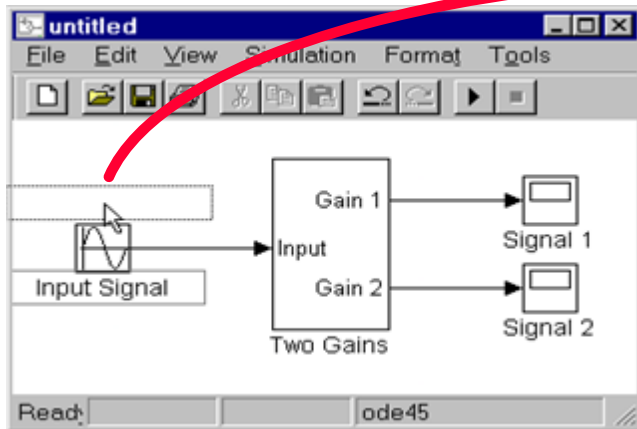
Cliccare sul nome



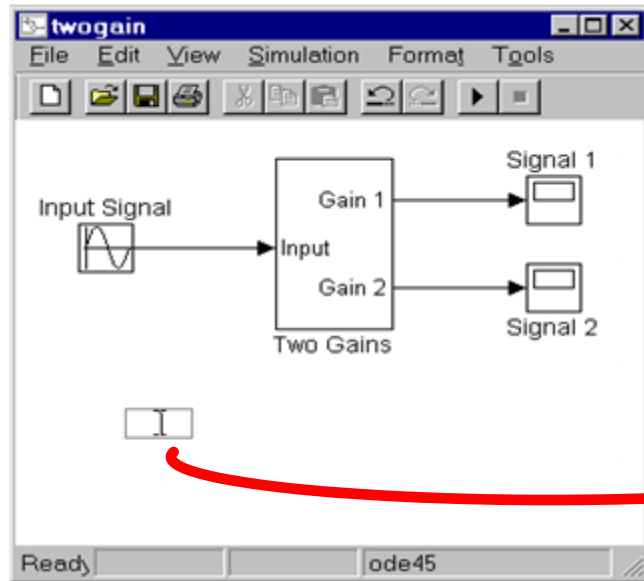
Inserire nuovo nome



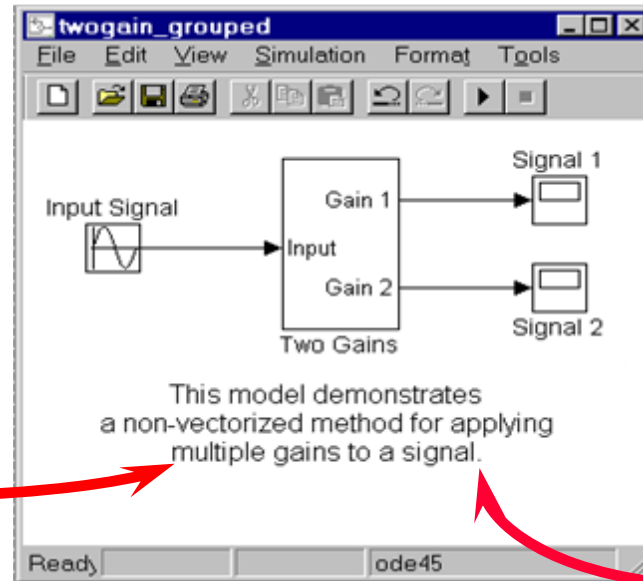
Trascinare il nome  
sul lato opposto  
del blocco



# Annotazioni



Doppi clic sullo sfondo del diagramma



Digitare il testo

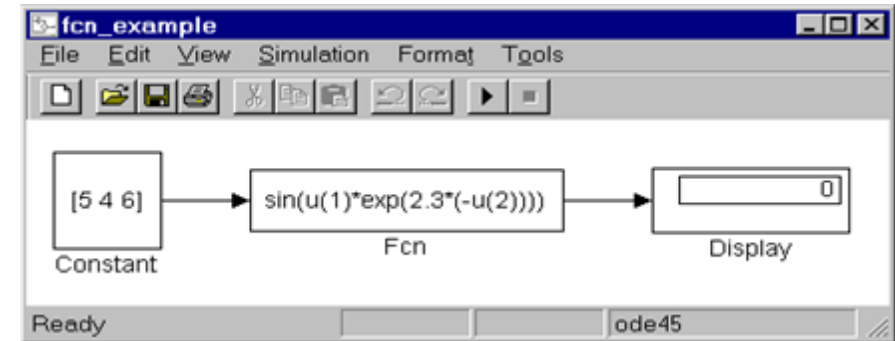
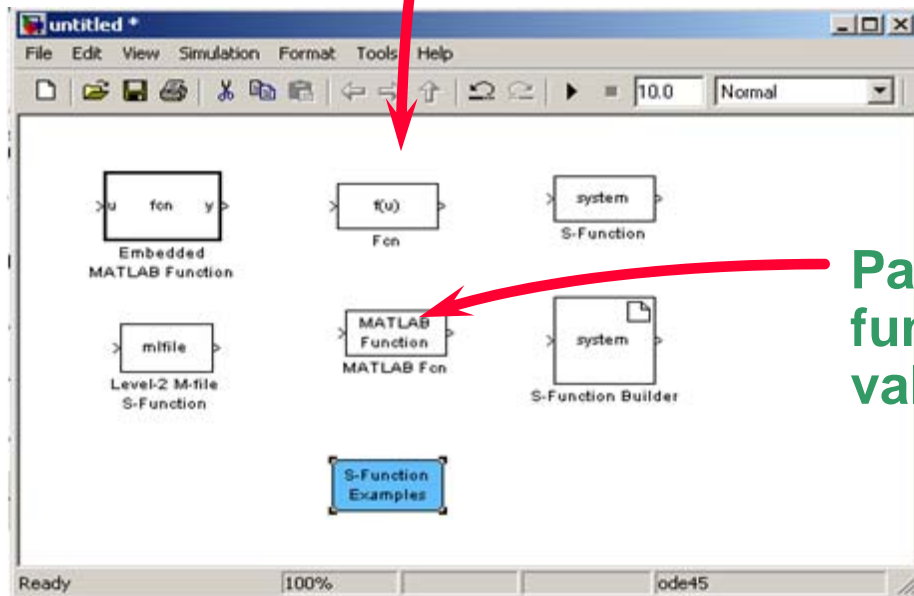
Cliccando con il destro sul testo si può ridimensionare il font

# Blocco Fcn e MATLAB Fcn

Definisce una funzione sulla base del segnale in ingresso u

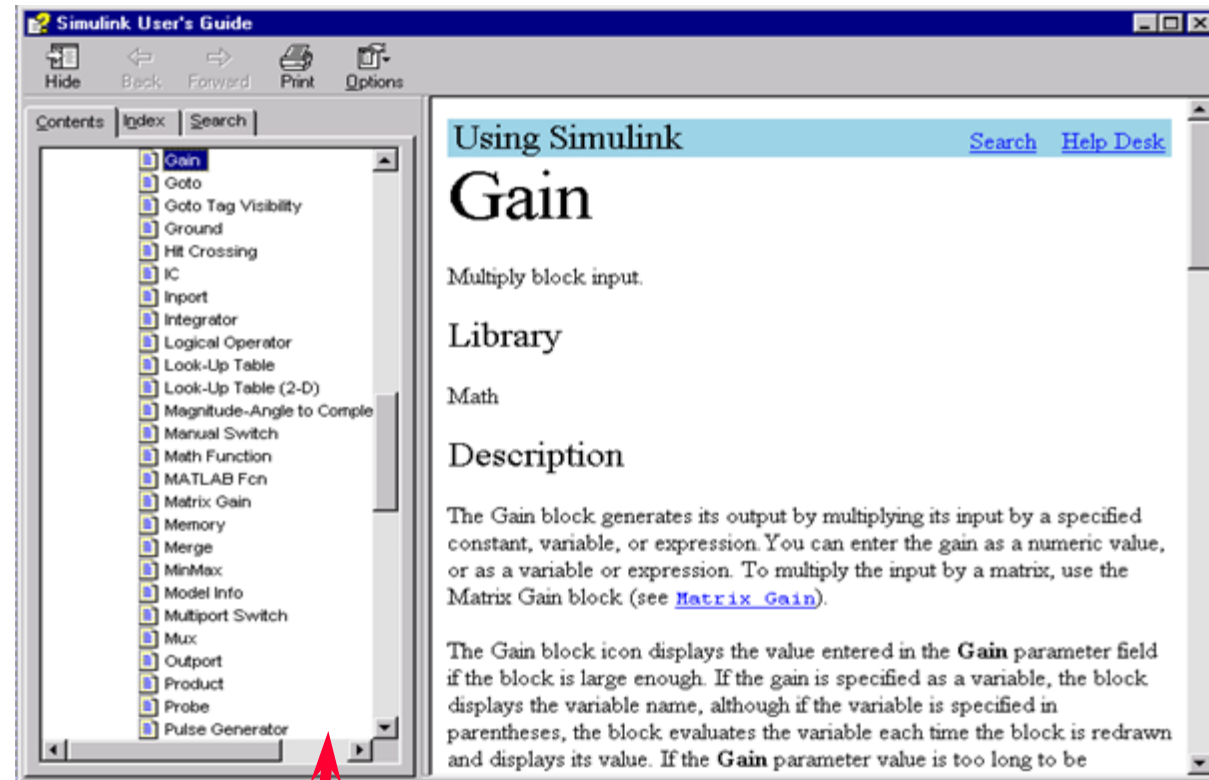
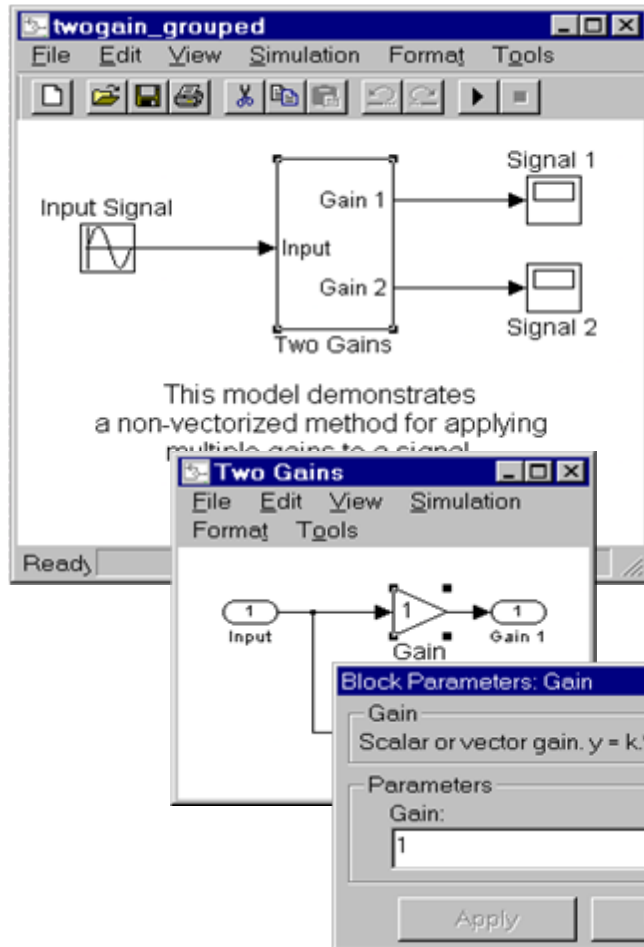
Se il segnale è un vettore si indicizza u : ex : u(1)...u(n)

$\sin(u(1)*\exp(2.3*(-u(2))))$



» [fcn\\_example](#)

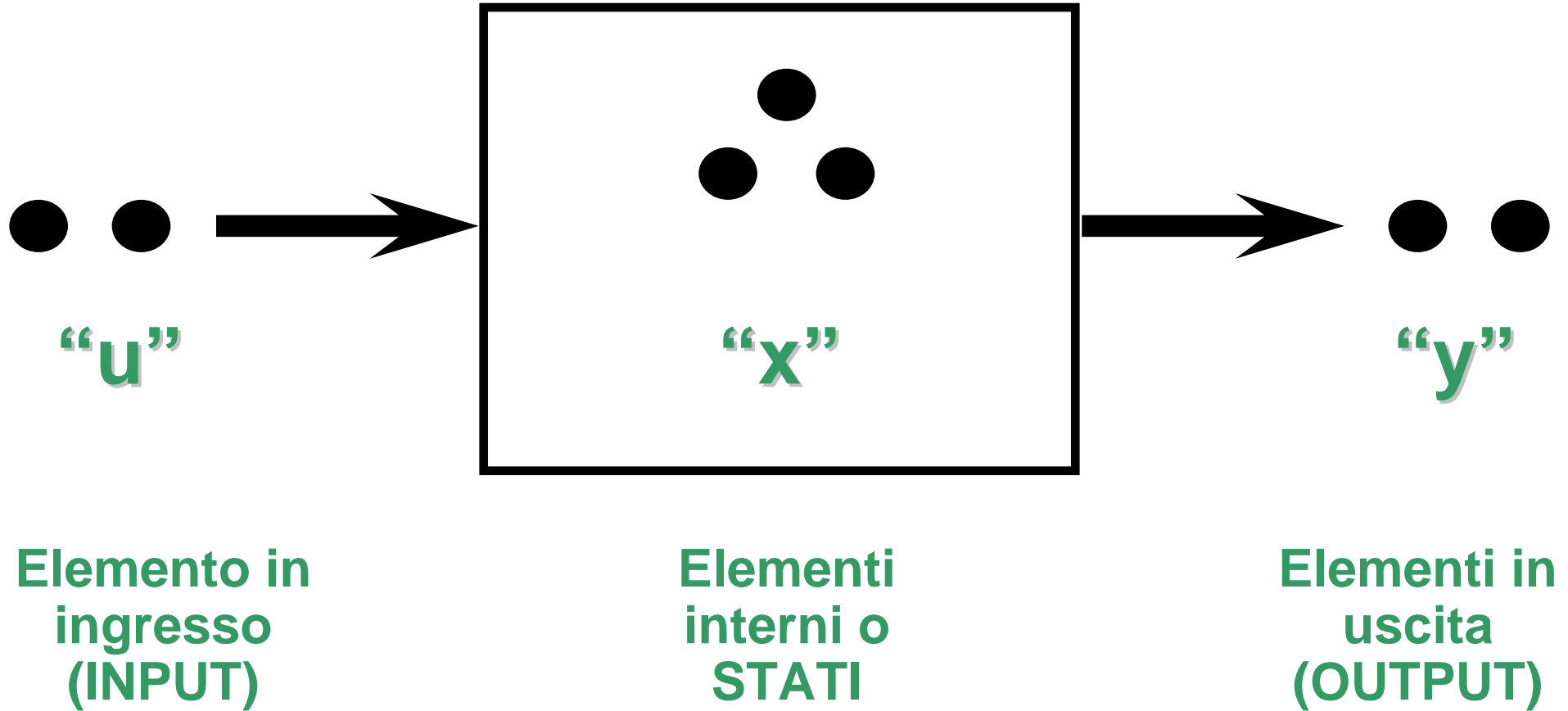
# Chiamare help di simulink



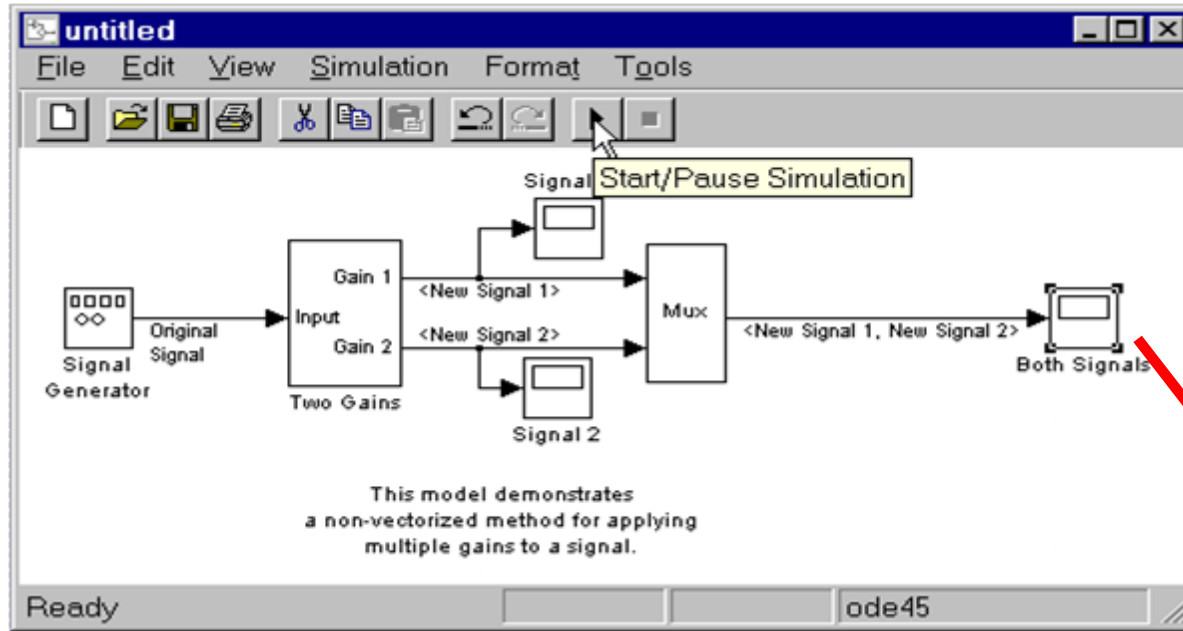
Aprire il blocco e cliccare help in basso a destra

# Che cos è un sistema

---

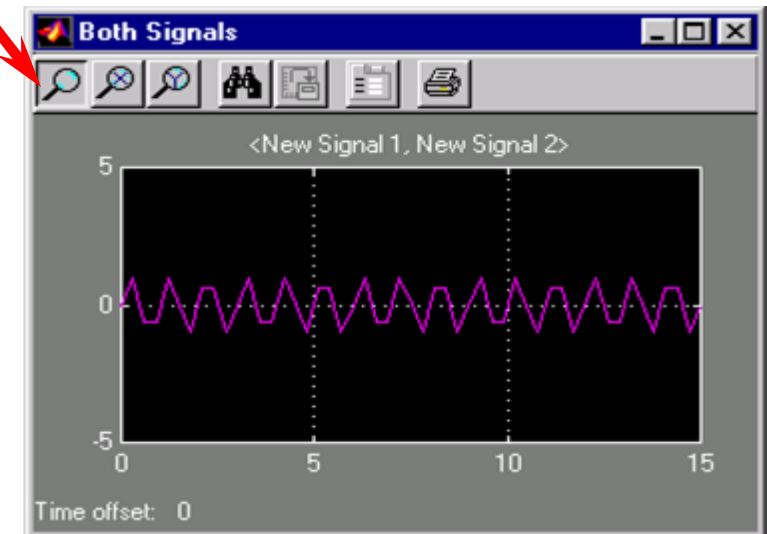


# Simulare



Usare:  
Stop Time: 15 secondi

» twogain\_final



## Settare parametri della simulazioni

“Auto” in max step size in questo caso significa:

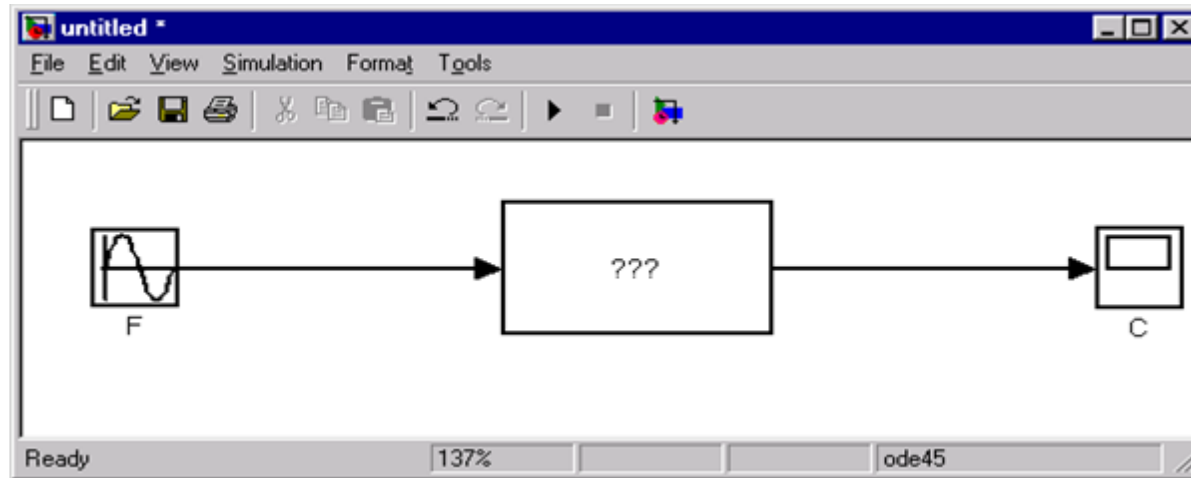
$$h_{\max} = \frac{t_{\text{stop}} - t_{\text{start}}}{50} = \frac{15}{50} = 0.3$$

» [twogain\\_final](#)



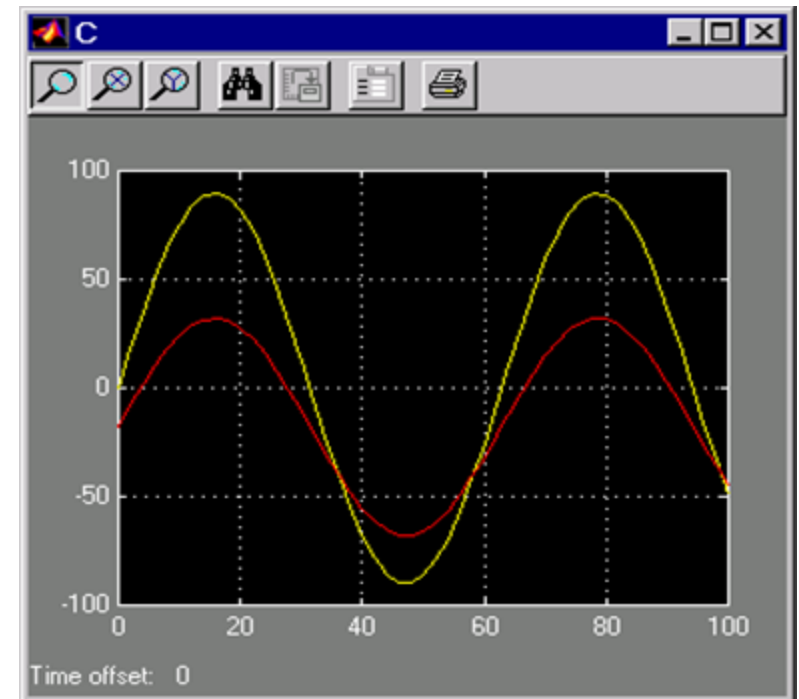
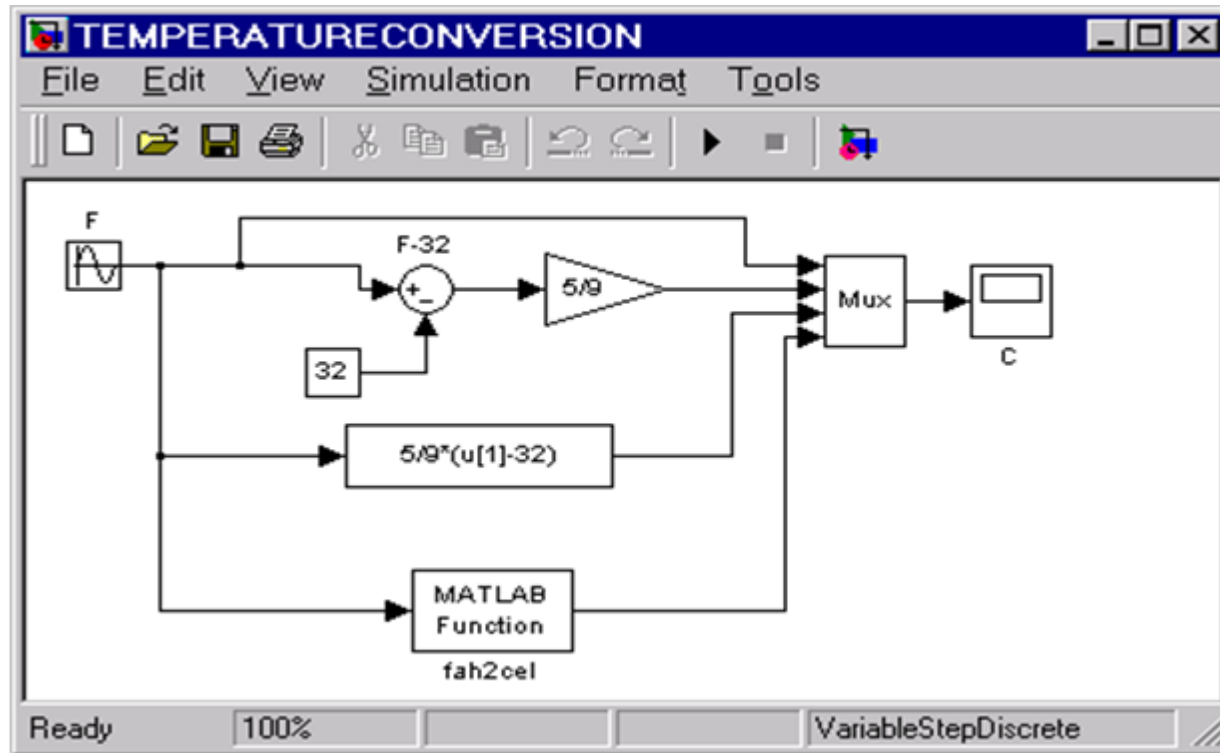
## Esercizio conversione della temperatura

$$C = (5/9) * (F - 32)$$



- 1) Determinare quale blocco è necessario per convertire la temperatura in Simulink?
- 2) Usare una onda sinusoidale come ingresso (settare ampiezza 80 degF), confrontare graficamente F vs. C

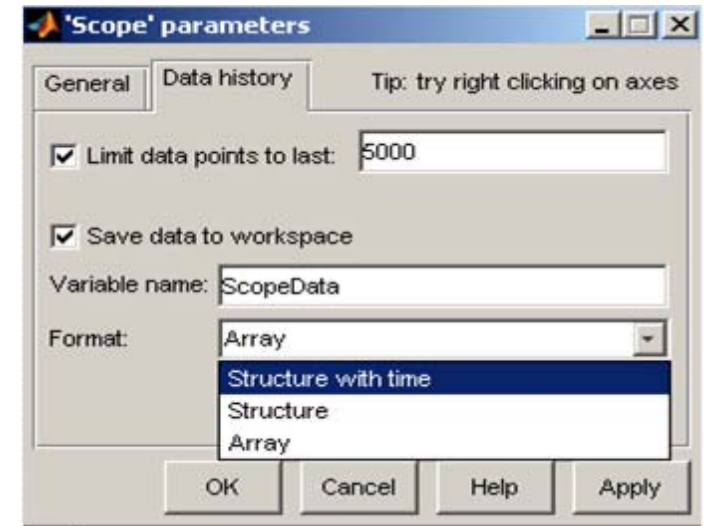
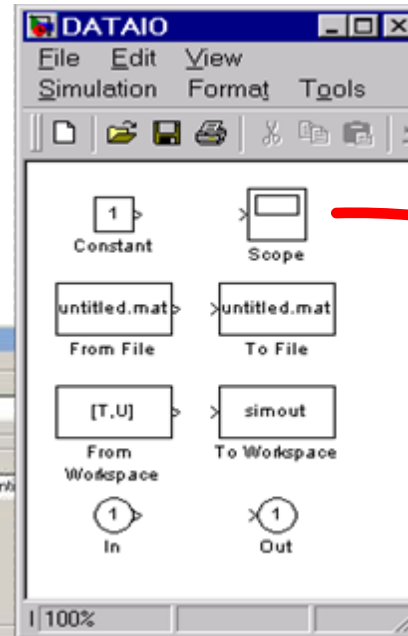
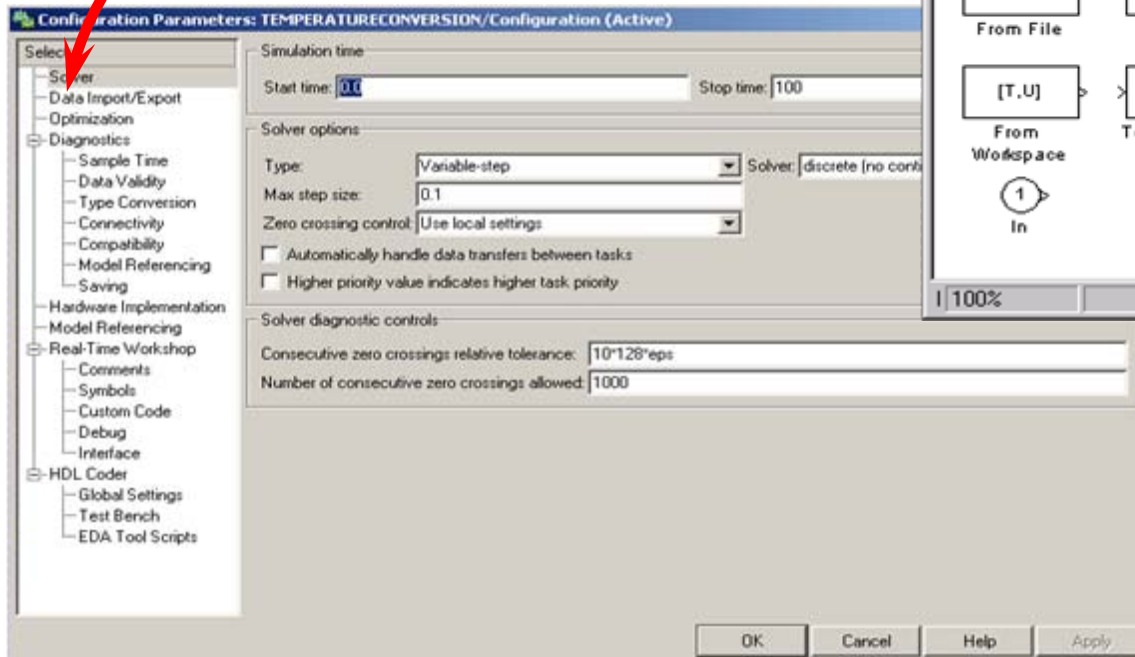
# Soluzione



» **TEMPERATURECONVERSION**

# Esportare dati

Possibilità di settare Inports e Outports (linea diretta con Matlab)

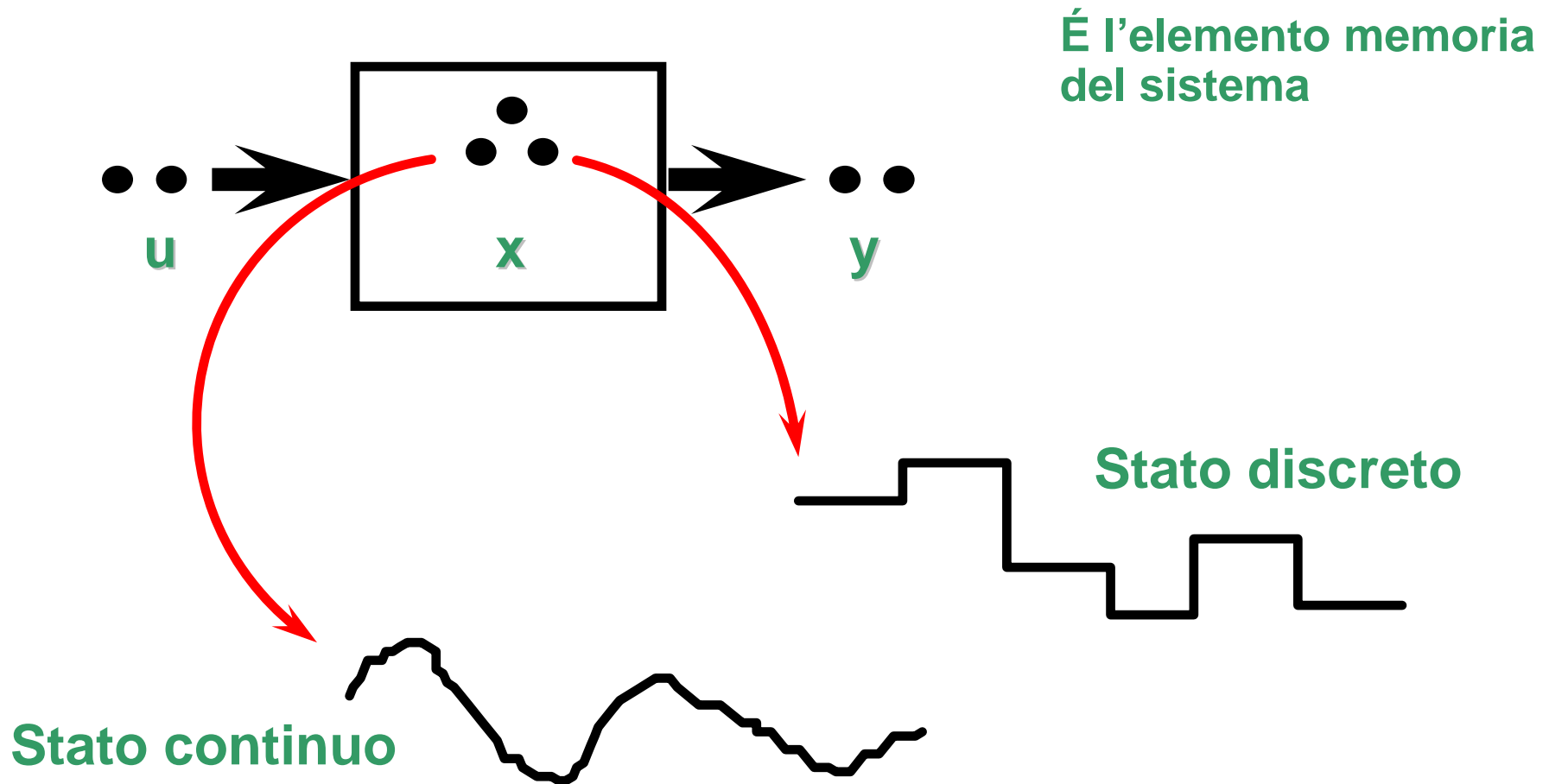


- Un risolutore è un motore che scandisce il sistema in base al tempo.
- Per sistemi senza stati continui il tempo avanza attraverso il calcolo computazionale dei soli blocchi in uscita.
  - Per sistemi con stati continui e sistemi ibridi, si usano routine numeriche come la ode23.

$$\begin{array}{l} \textit{Continuous} \quad : \dot{x} = f_c(x, u) \\ \textit{Discrete} \quad : x_{k+1} = f_d(x_k, u_k) \end{array}$$

Simulink instaura un dialogo tra il sistema ed il risolutore

# Che cosa è lo stato del sistema?

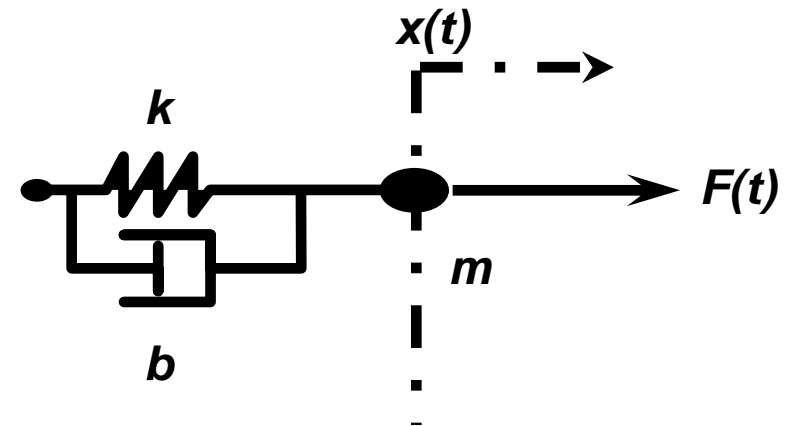


# Modelli continui

Equazioni differenziali ordinarie

$$F = m\ddot{x} + b\dot{x} + kx$$

$$\frac{x}{F} = \frac{y}{u} = \frac{1}{ms^2 + bs + k}$$



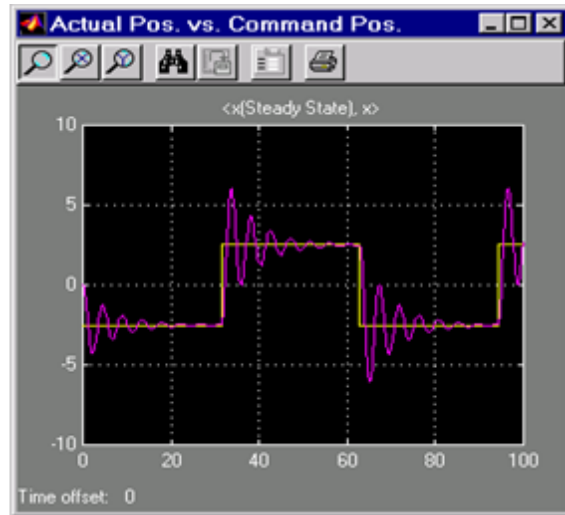
$$[m \ b \ k]$$

# Spring-Mass-Damper

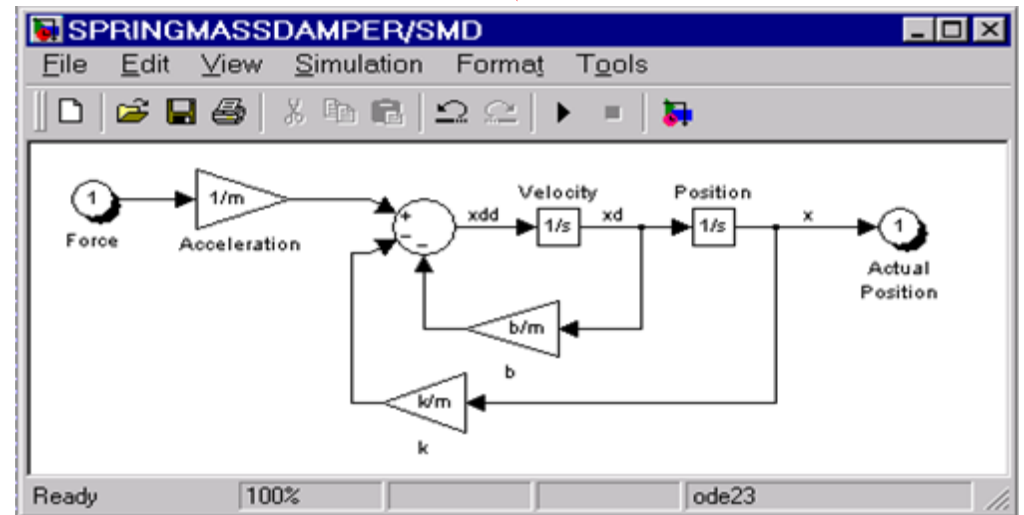
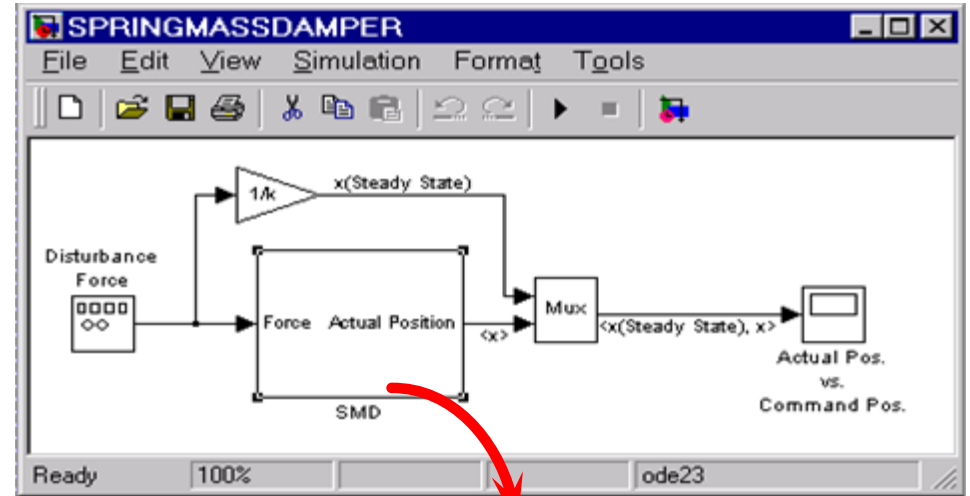
$$\ddot{x} = \frac{F}{m} - \frac{b\dot{x}}{m} - \frac{kx}{m}$$

dati

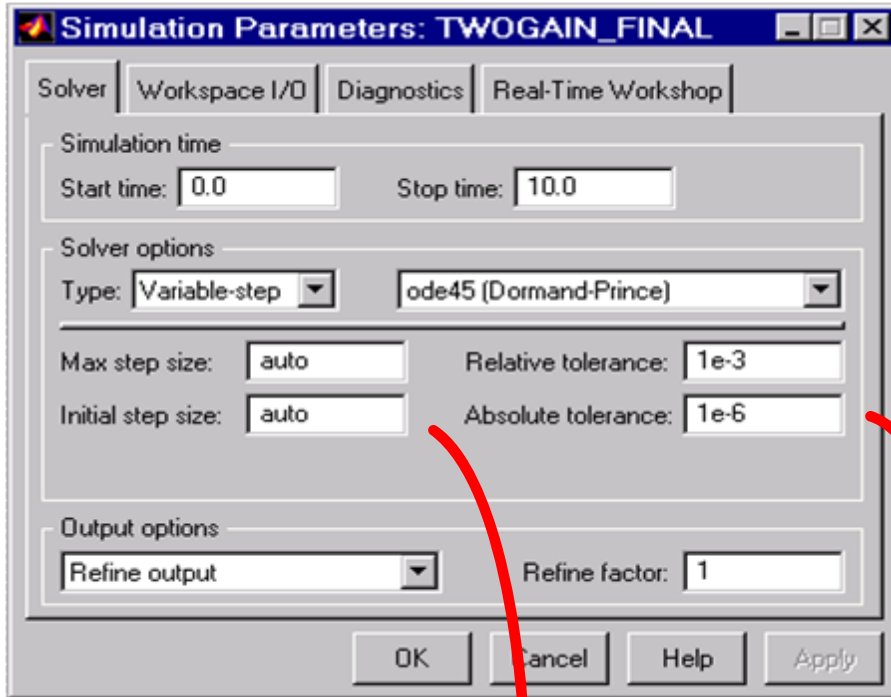
k=100;  
b=20;  
m=10;  
x0=1;  
F=100;



>>SPRINGMASSDAMPER

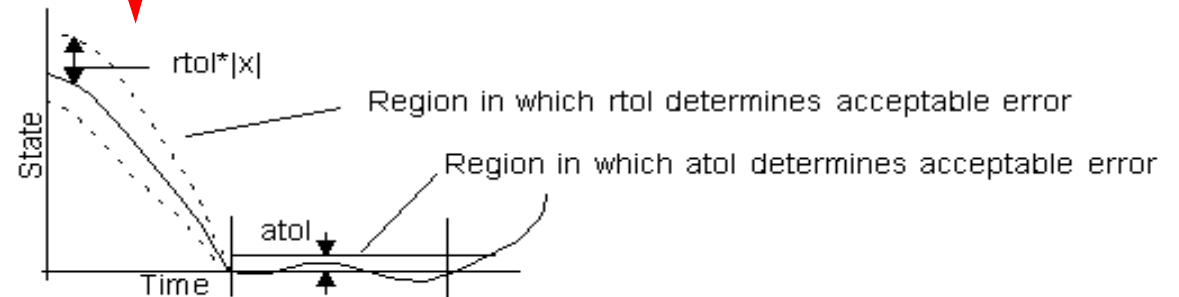


# Setting parameters



**Tolerances control  
accuracy of simulation**

$$h_{\max} = \frac{t_{\text{stop}} - t_{\text{start}}}{50}$$



# Risolutori disponibili

---

## ❑ Risolutori a passo variabili

- Modificano il passo di integrazione durante la simulazione
- Controllo dell'errore e la rivelazione del passaggio dello zero
- ode45, ode23, ode113, ode15s, ode23s, ode23t ode23tb, discrete

## ❑ Risolutore a passo fisso

- Stesso passo di integrazione per tutta la simulazione
- Nessun controllo sull'errore e sul passaggio dello zero
- ode5, ode4, ode3, ode2, ode1, discrete

## ❑ Simulink seleziona un risolutore di base

- ode45 per I modelli a tempo continuo
- Variable-step discrete-time solver per I modelli senza stati continui

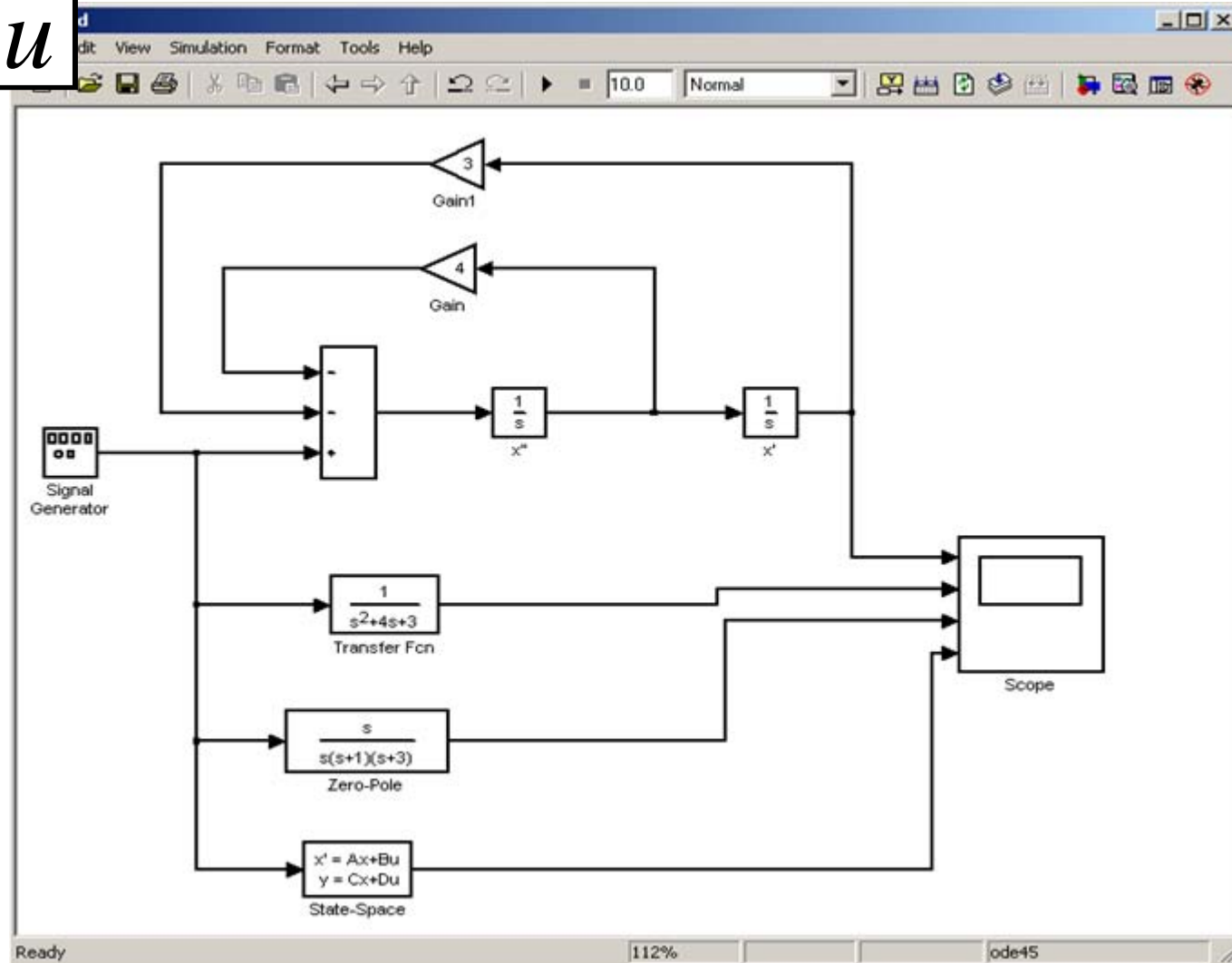
Un sistema stiff è un sistema la quale soluzione può cambiare a secondo della scala dei tempi usata; la soluzione è sensibile all'intervallo di integrazione scelto

# Modellare stati continui

$$\ddot{x} = -4\dot{x} - 3 \cdot x + u$$

$$A = \begin{bmatrix} -4 & -3 \\ 1 & 0 \end{bmatrix}; B = \begin{bmatrix} 1 \\ 0 \end{bmatrix};$$
$$C = \begin{bmatrix} 0 & 1 \end{bmatrix}; D = 0;$$

» **abcd**



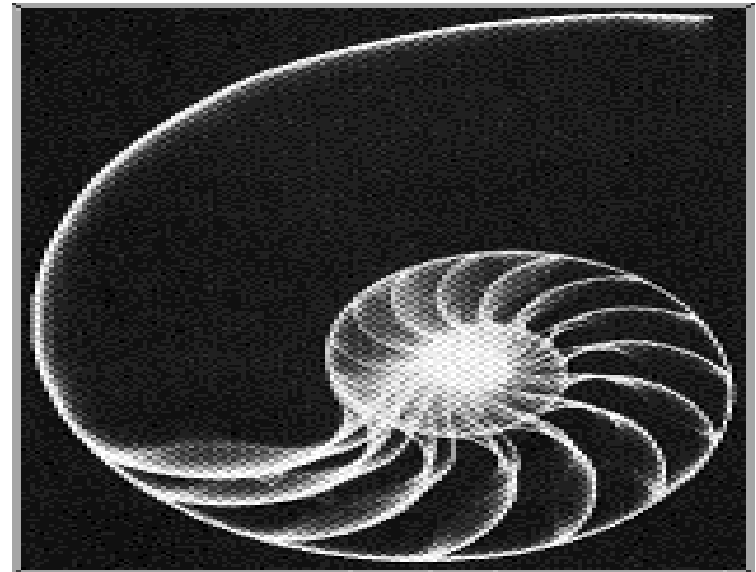
# Esercizio: Fibonacci

---

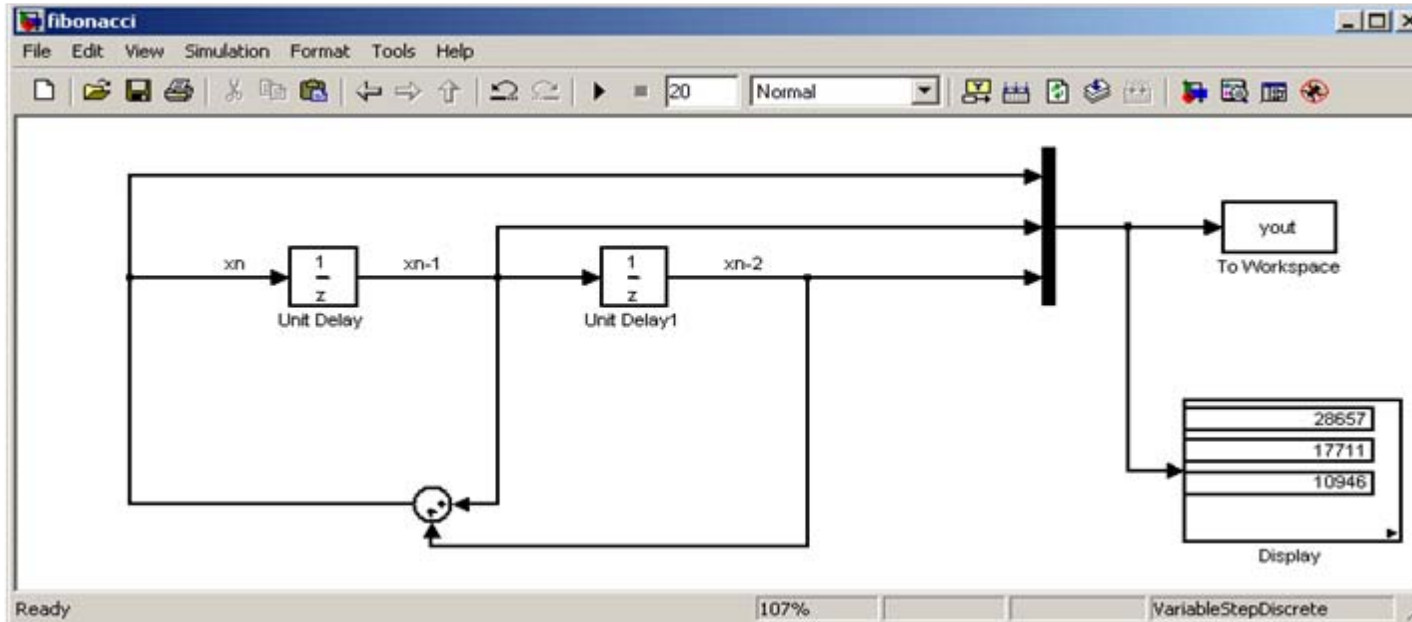
Generare la sequenza di Fibonacci con il simulink usando il ritardo unitario; salvare la serie sul workspace dei primi 20 numeri di tale serie.

$$x_n = x_{n-1} + x_{n-2}$$

$$x_1 = 1, x_2 = 1$$



# Solution: Fibonacci



```
MATLAB Command Window
File Edit Window Help
>> yout
yout =
     2     1     1
     3     2     1
     5     3     2
     8     5     3
    13     8     5
    21    13     8
    34    21    13
    55    34    21
    89    55    34
   144   89    55
   233  144    89
   377  233   144
   610  377   233
   987  610   377
  1597  987   610
  2584 1597   987
  4181 2584  1597
  6765 4181  2584
 10946 6765  4181
 17711 10946 6765
 28657 17711 10946
```

$$x_n = x_{n-1} + x_{n-2}$$
$$x_1 = 1, x_2 = 1$$

» fibonacci

# Esempio Lancio Simulink da Matlab

```
%Lanciare una logistica fatta su simulink da matlab e visualizzare i  
%risultati in un plot
```

```
clc  
clear  
close all  
global r K  
r=1.03;      %rateo di crescita  
K=23;       %capacità portante  
y0=8;       %popolazione iniziale  
tempofin=10;  
% implementazione in Simulink  
[T_sim,X_sim,Y_sim]=sim('logistica_sim',[0 tempofin]);  
hold on % tengo attivo il disegno precedente sulla figura(1)  
plot(T_sim,OUT_logistica,'m')  
legend('logistica calcolata con Simulink','Location','SouthEast')  
grid on
```

