

Controlli automatici

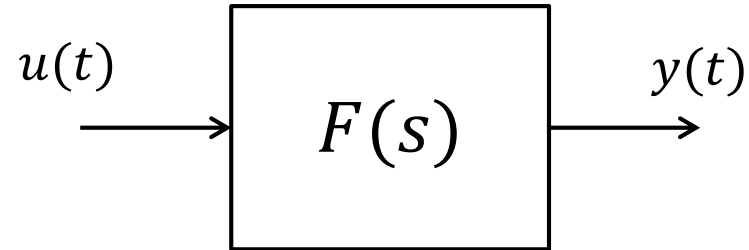
Esercitazione n. 1

Introduzione alla simulazione di sistemi di controllo
in ambiente Matlab-Simulink

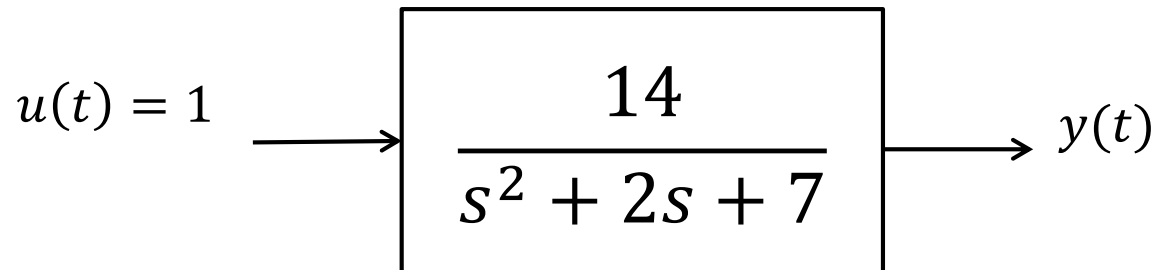
Caso di studio: Simulazione di un cruise control system

Simulazione di un sistema a ciclo aperto

Il primo task che affrontiamo è la realizzazione di un test a **ciclo aperto** in cui un sistema dinamico LTI descritto dalla FdT $F(s)$ viene sottoposto ad un certo segnale di ingresso $u(t)$ e si desidera visualizzarne la risposta $y(t)$



Piu in dettaglio, ci proponiamo di visualizzare la risposta al gradino unitario di un processo del secondo ordine



Realizziamo il modello Simulink

La realizzazione di un modello Simulink per la simulazione dinamica avviene per via grafica, disegnando uno **schema a blocchi** in tutto e per tutto simile a quelli che normalmente tracciamo a lezione per rappresentare i sistemi di controllo.

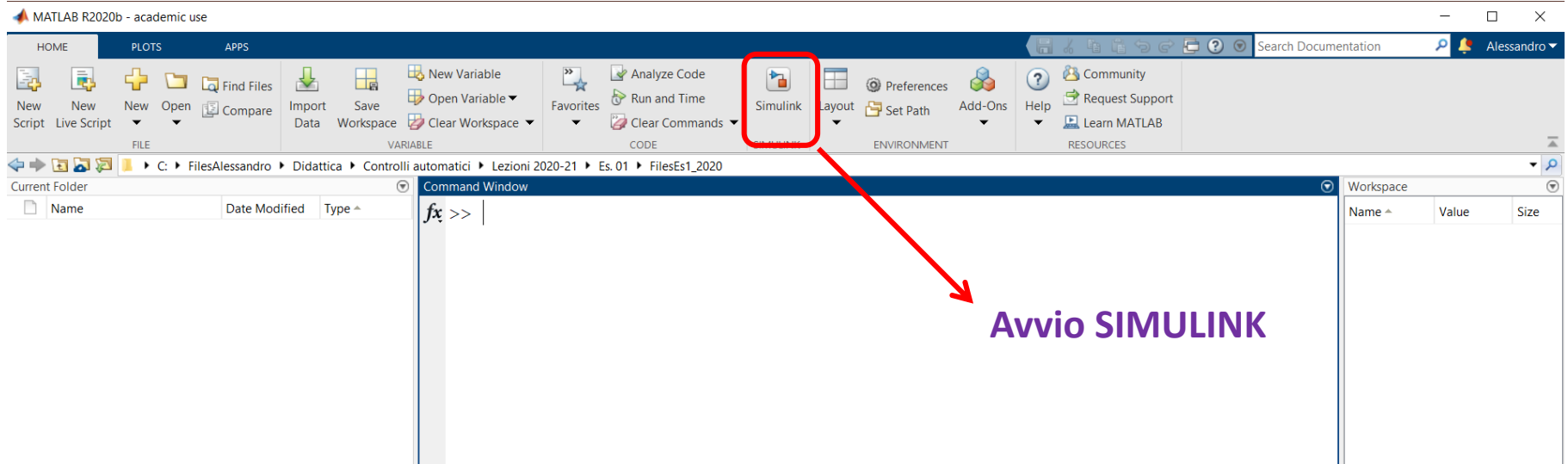
La realizzazione di un modello Simulink avviene attraverso tre fasi:

- si importano in una pagina bianca di lavoro i **blocchi elementari** necessari per la realizzazione dello schema di simulazione
- si **parametrizzano i blocchi** in modo che implementino le funzionalità desiderate
- si **interconnettono tra loro i blocchi** per realizzare lo schema desiderato

Fatto ciò, si può avviare la simulazione e visualizzarne i risultati

Si apra Matlab

Finestra di avvio di Matlab (rel. R2020b)

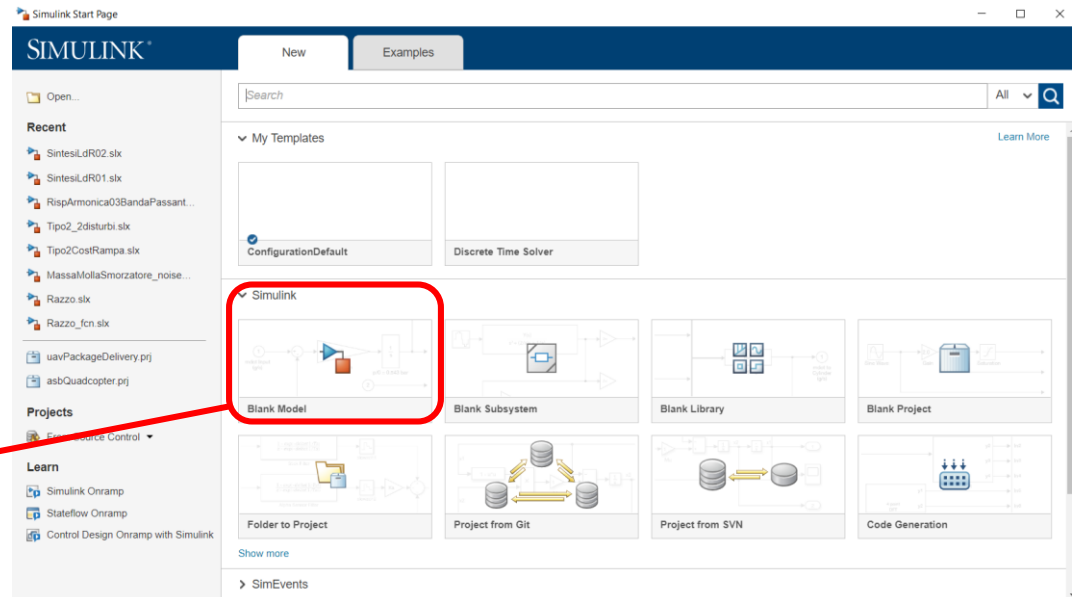


Avvio SIMULINK

Finestra di avvio SIMULINK

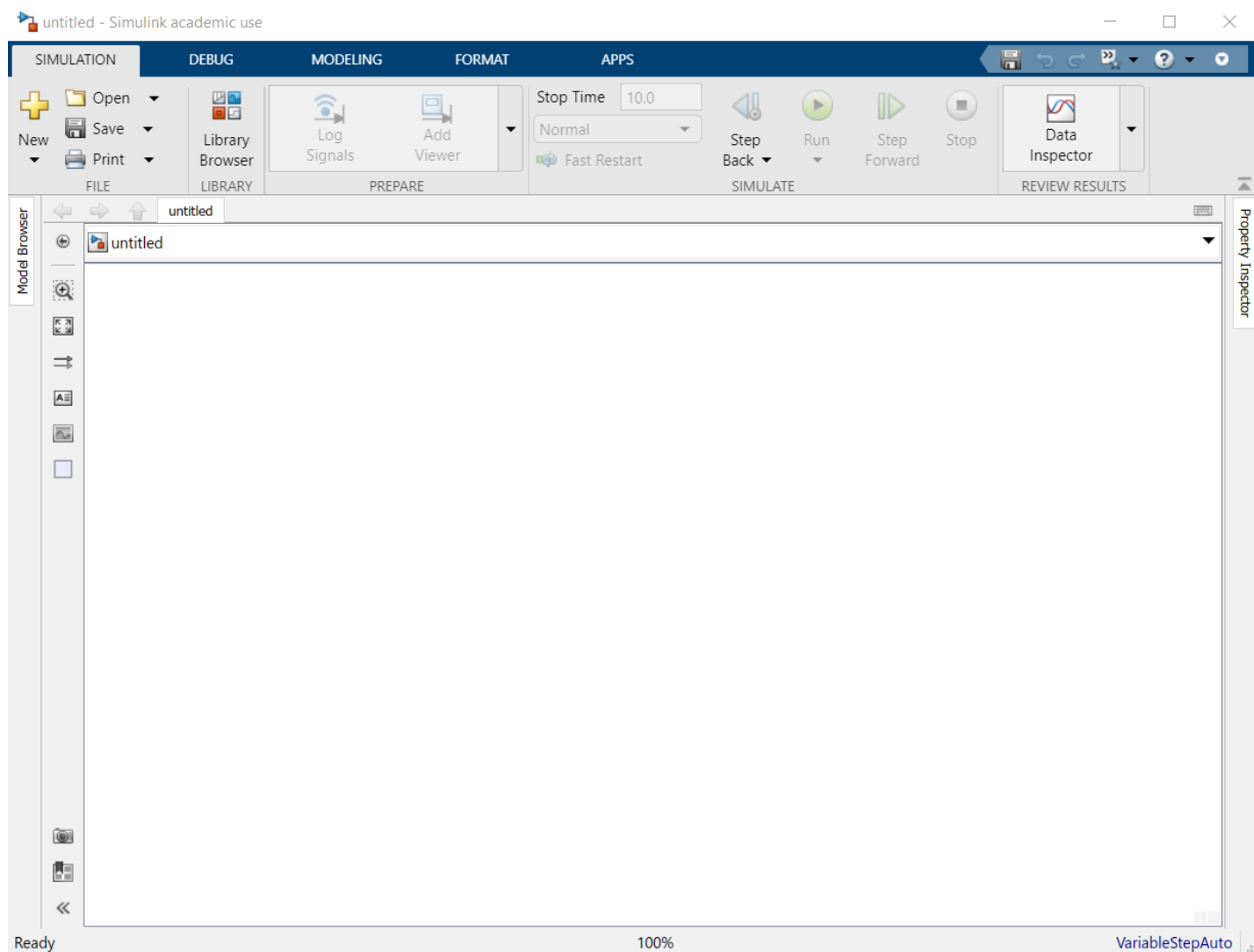
Dalla finestra di avvio di Matlab,
apriamo Simulink.

Si apre la relativa finestra di avvio:



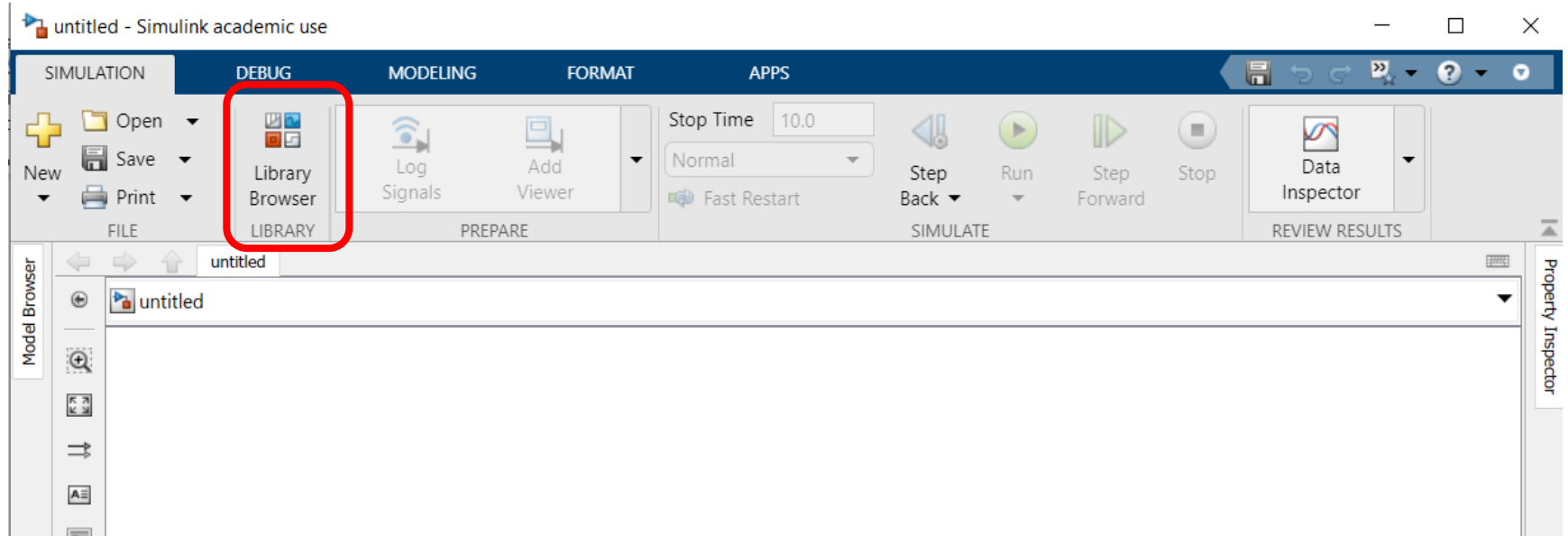
Apertura di un Blank Model

Modello Simulink in bianco

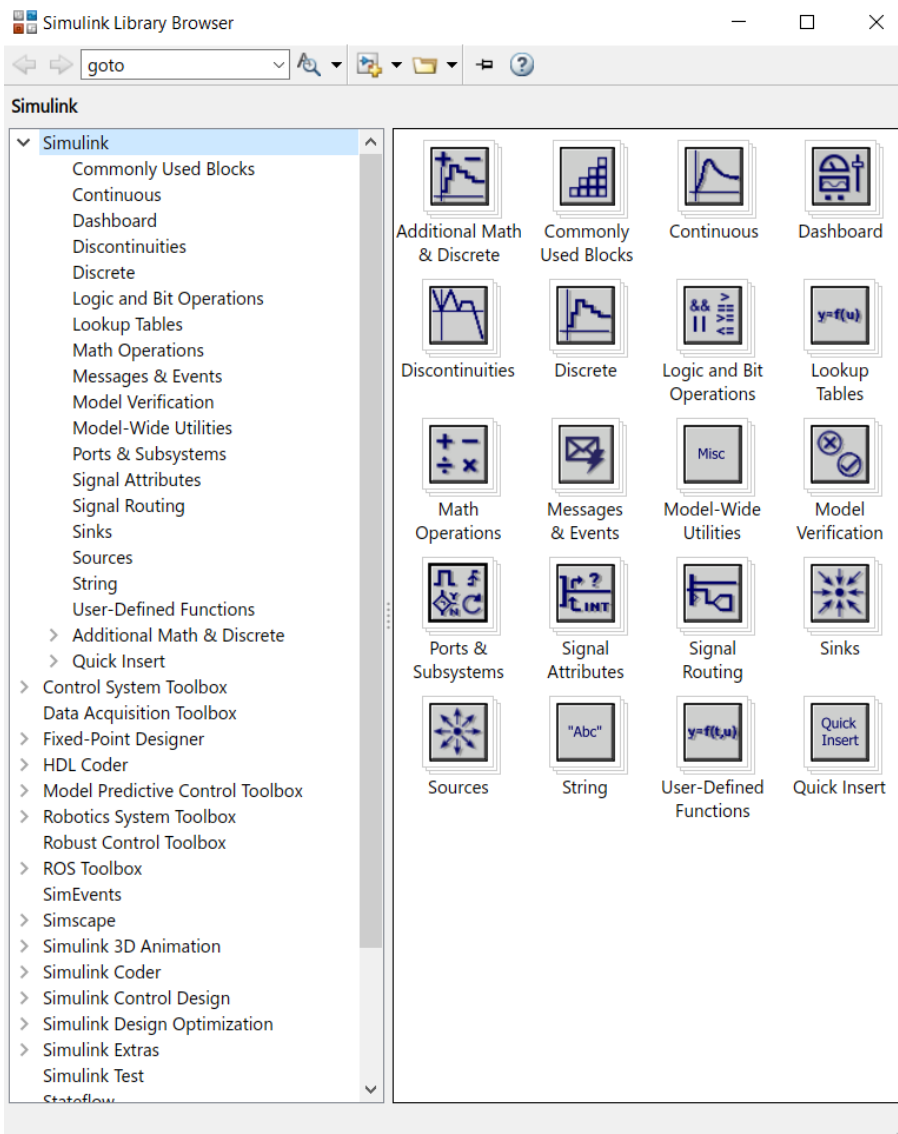


La realizzazione di modelli di simulazione dinamica avviene per via grafica **assemblando fra loro all'interno della pagina di lavoro un certo numero di blocchi Simulink**, in modo da implementare le funzionalità desiderate.

I blocchi Simulink sono allocati all'interno di **librerie**. E' possibile accedere al «Library browser» cliccando il relativo pulsante nella finestra che ospita il modello in bianco (Menu: «SIMULATION»)



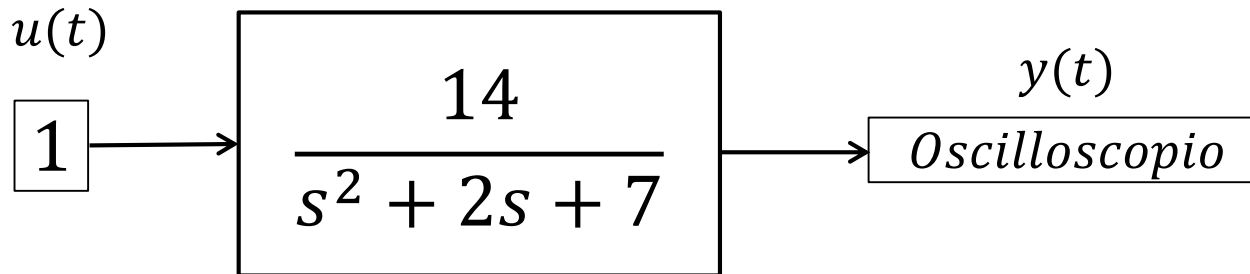
Simulink Library Browser



Contiene le librerie di blocchi elementari SIMULINK, da quelli di base fino a quelli più sofisticati orientati a particolari applicazioni

Per realizzare lo schema a blocchi associato ad una simulazione dinamica a ciclo aperto sono sufficienti **tre blocchi elementari** Simulink:

- 1 un blocco elementare che implementi una FdT
- 2 un blocco elementare che generi il segnale di ingresso
- 3 un blocco elementare «oscilloscopio» che consenta la visualizzazione del segnale di uscita

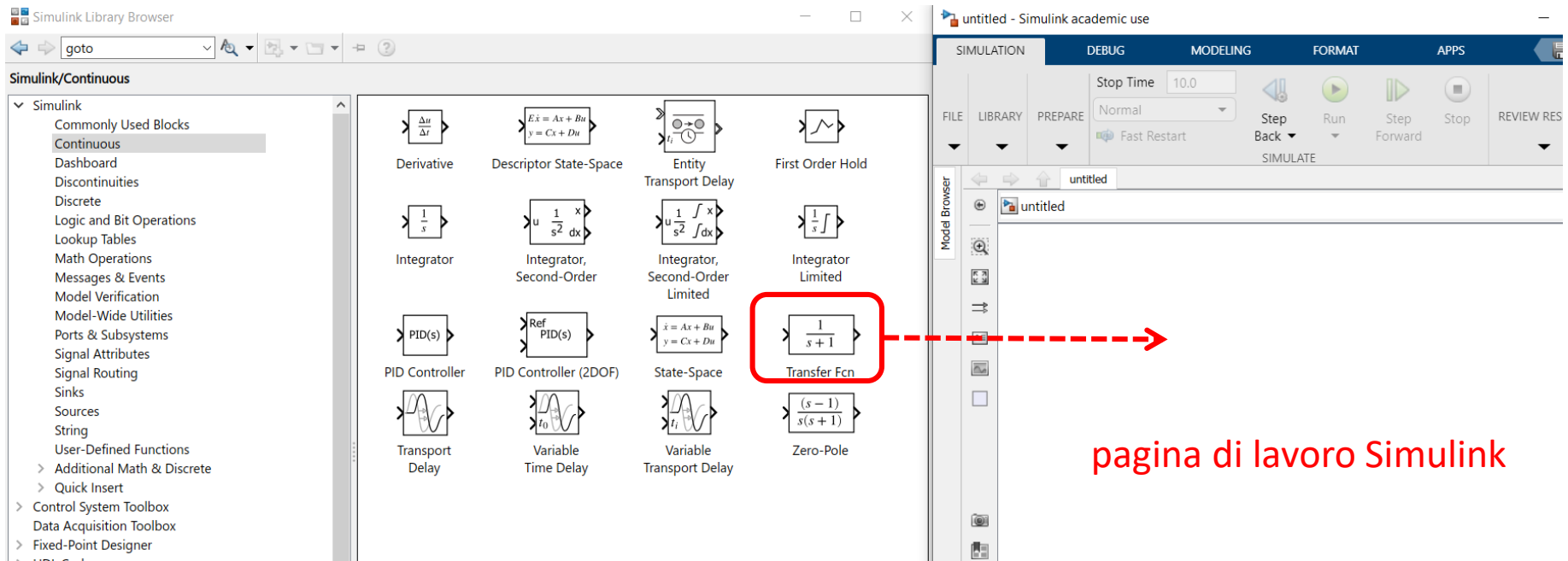


Nella pagina di lavoro Simulink, andremo a “disegnare” una rappresentazione analoga a quella riportata qui sopra

Importiamo nella pagina di lavoro il blocco “Transfer Fcn” (funzione di trasferimento), che si trova nella sotto-libreria “Continuous” della libreria principale “Simulink”

E' uno dei blocchi (non l'unico) per mezzo dei quali si possono realizzare in ambiente Simulink delle funzioni di trasferimento

L'importazione dei blocchi nella pagina di lavoro si effettua con il mouse mediante drag-and-drop



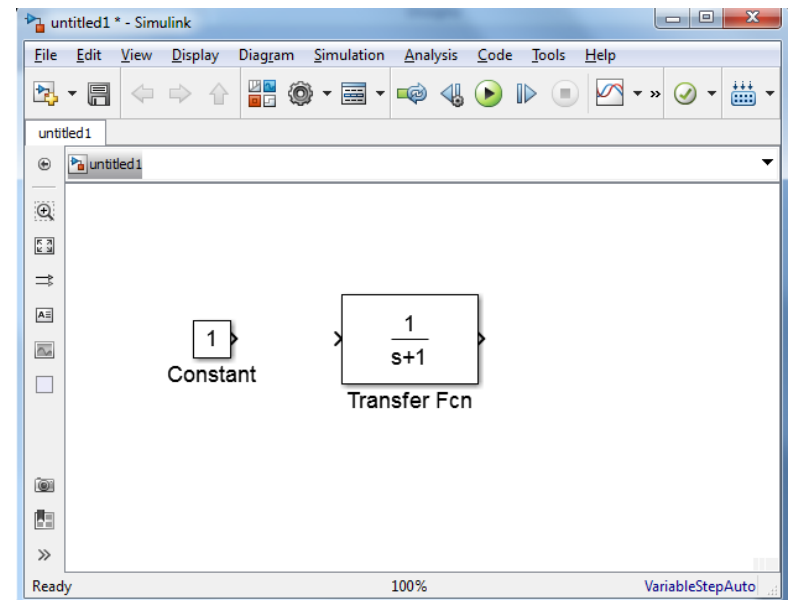
pagina di lavoro Simulink

Ora importiamo il blocco necessario per generare il segnale di ingresso costante.

Tale blocco si chiama `Constant`, e si trova nella sotto-libreria “`Sources`” della libreria principale “`Simulink`”, che contiene una ampia varietà di blocchi elementari preposti alla **generazione di segnali**

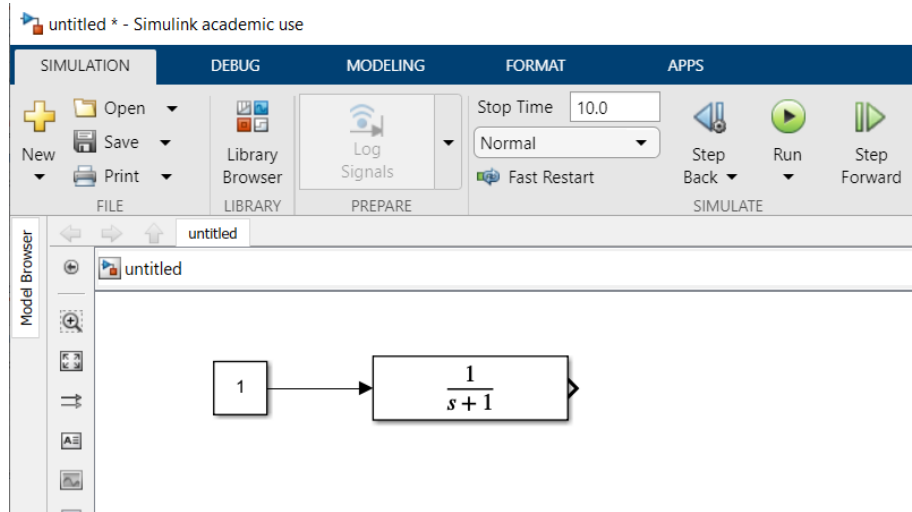
Disponiamo i blocchi come nella figura a lato

Si noti che il blocco `Constant` possiede unicamente un terminale di uscita, mentre invece il blocco `Transfer Fcn` possiede un terminale di ingresso ed un terminale di uscita, come è lecito attendersi.



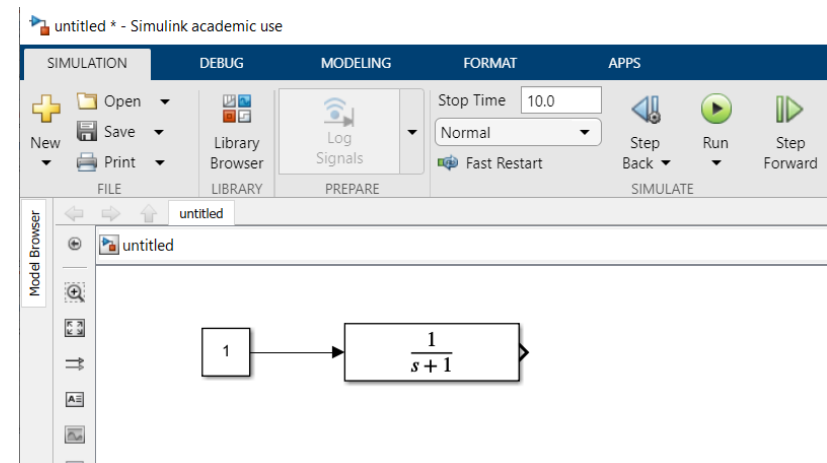
Interconnettiamo i due blocchi.

Il tracciamento di una connessione si effettua portandosi con il mouse nel punto di inizio della linea di connessione (quindi nel terminale di uscita del blocco Constant), premendo il tasto sinistro, e successivamente portandosi con il mouse - mantenendo premuto il tasto - verso il punto di destinazione, in questo caso il terminale di ingresso del blocco Transfer Fcn.



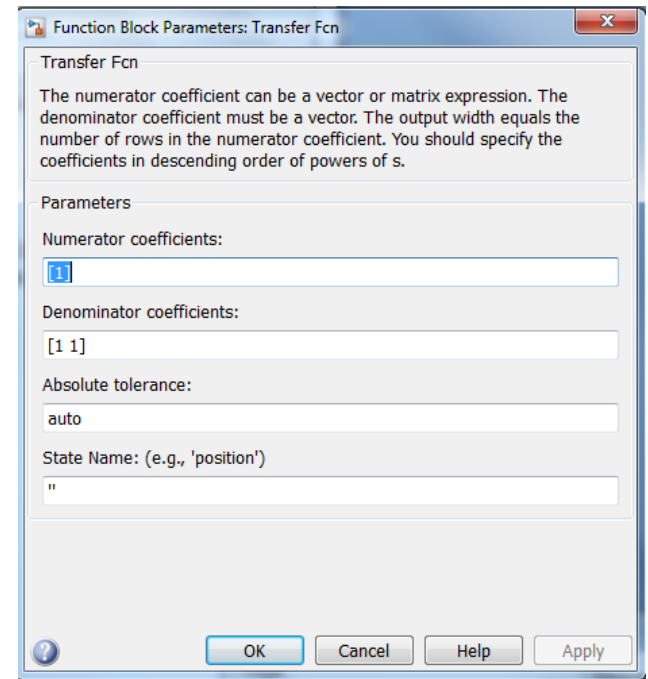
Per effettuare un collegamento tra due blocchi vi è anche una **procedura rapida**. Si deve selezionare il blocco di origine (cliccandovi sopra), e si deve successivamente selezionare il blocco di destinazione con il **tasto ctrl premuto**.

Di default, il blocco Constant è parametrizzato per generare un segnale costante di ampiezza unitaria, mentre il blocco Transfer Fcn è parametrizzato di default con la FdT $\frac{1}{s+1}$



Facendo doppio click su un blocco, si accede alla sua finestra di parametrizzazione.
Il blocco Constant è già configurato correttamente per il presente esempio.

Dobbiamo invece configurare il blocco Transfer Fcn.
Apriamone la finestra di parametrizzazione



Parametrizzazione del blocco Transfer Function

Il blocco deve rappresentare la FdT

$$F(s) = \frac{14}{s^2 + 2s + 7}$$

Si devono specificare i **coefficienti dei polinomi a numeratore e denominatore** della FdT utilizzando la notazione Matlab per la rappresentazione dei polinomi (un vettore che contiene i coefficienti del polinomio **in ordine decrescente** rispetto alle potenze di s)

$$14 \Rightarrow [14]$$

$$s^2 + 2s + 7 \Rightarrow [1 \quad 2 \quad 7]$$

Altri esempi:

$$s \Rightarrow [1 \quad 0]$$

$$s^4 - 2s^3 + 3s + 2 \Rightarrow [1 \quad -2 \quad 0 \quad 3 \quad 2]$$

$$s(s + 1) = s^2 + s \Rightarrow [1 \quad 1 \quad 0]$$

$$s^3 + 4 \Rightarrow [1 \quad 0 \quad 0 \quad 4]$$

Function Block Parameters: Transfer Fcn1

Transfer Fcn

The numerator coefficient can be a vector or matrix expression. The denominator coefficient must be a vector. The output width equals the number of rows in the numerator coefficient. You should specify the coefficients in descending order of powers of s.

Parameters

Numerator coefficients:

[1]

Denominator coefficients:

[1 1]

Absolute tolerance:

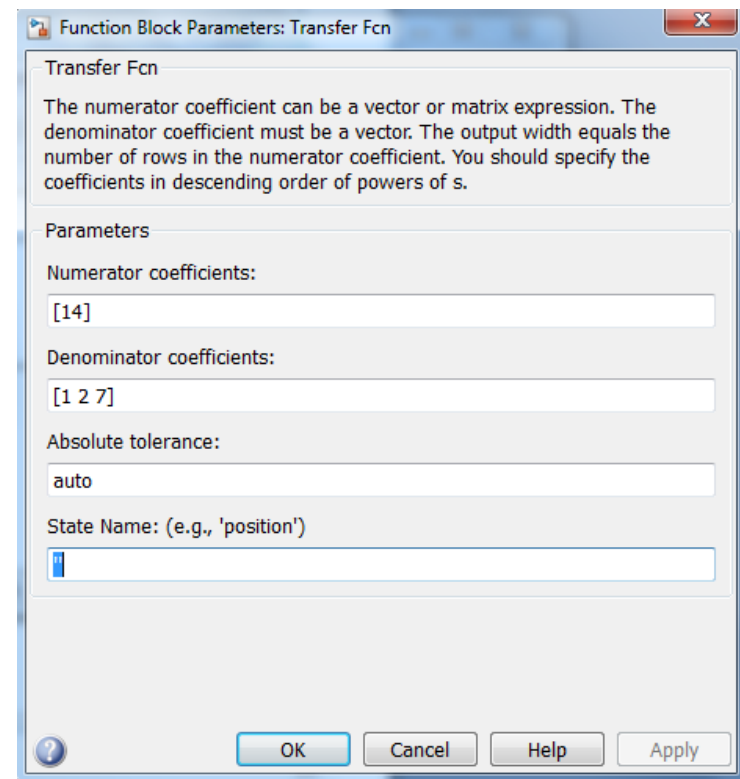
auto

State Name: (e.g., 'position')

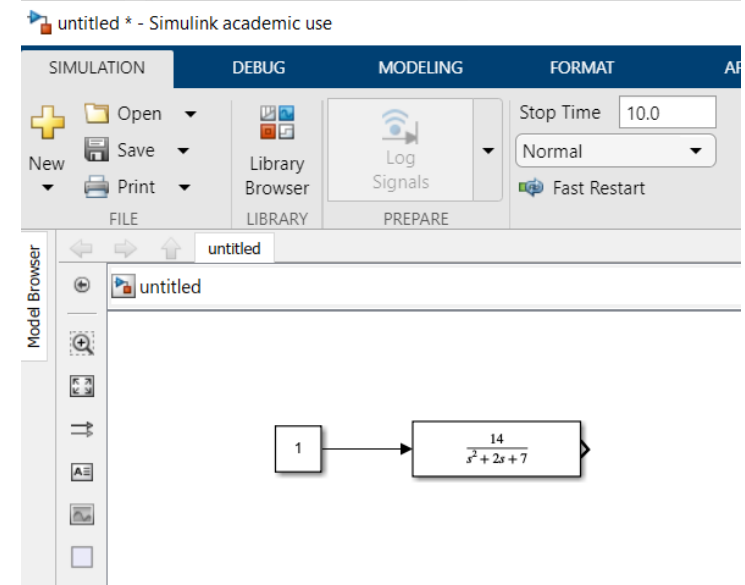
"

OK Cancel Help Apply

Il blocco deve essere pertanto parametrizzato nella seguente maniera.



Dopo aver premuto il tasto OK, l'aspetto del blocco Transfer Fcn nella pagina di lavoro cambia, e visualizza al proprio interno la FdT avente i parametri scelti (eventualmente il blocco deve essere ingrandito affinché la visualizzazione sia corretta)

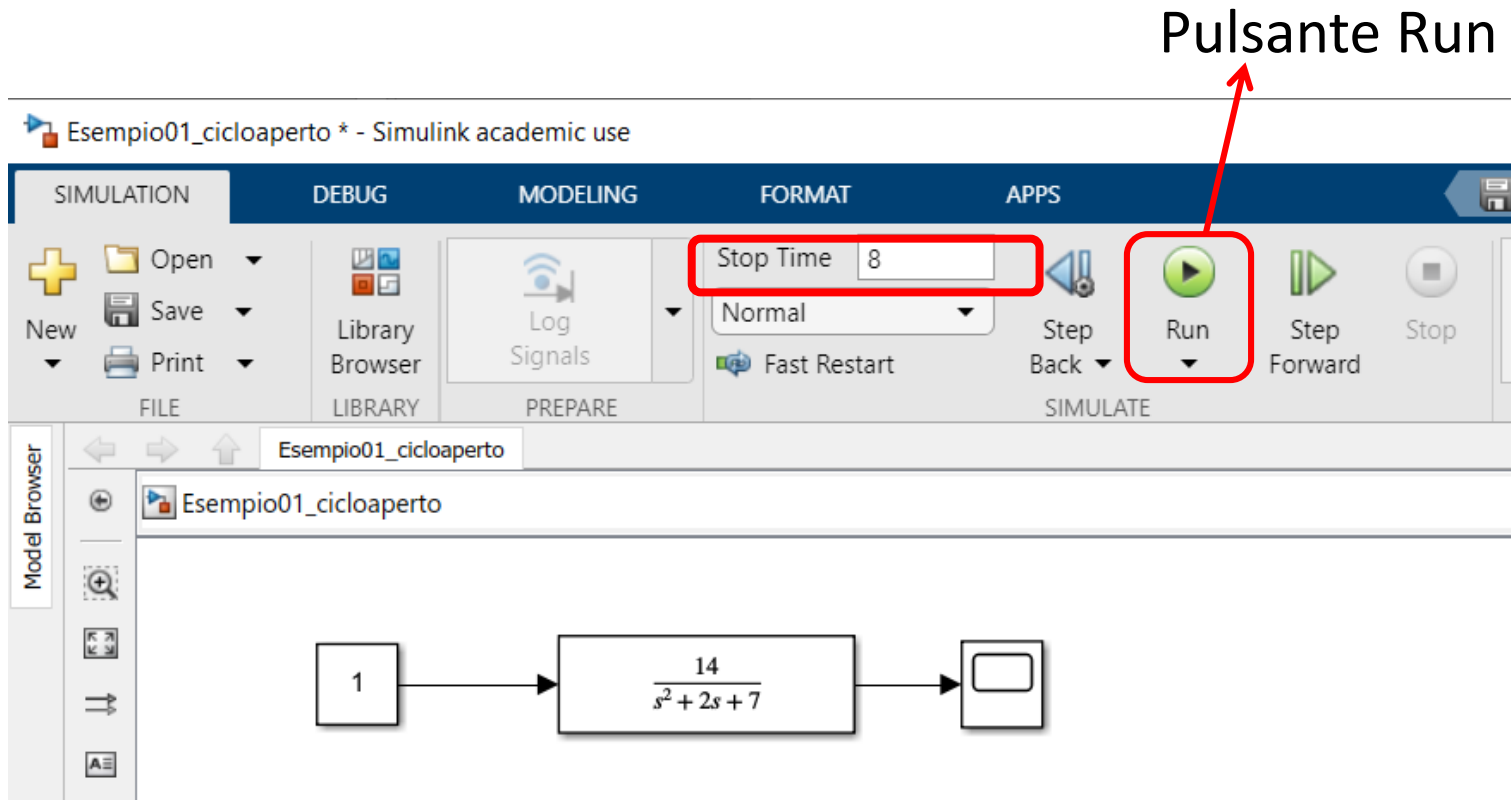


Ora importiamo il blocco “Scope”, che consente la visualizzazione di un segnale.

Il blocco sta nella sotto-libreria “Sinks” della libreria principale “Simulink”.

Collegiamone il terminale di ingresso al terminale di uscita del blocco Transfer Fcn.

Impostare nella casella **Stop Time** la durata della simulazione (il valore di default è 10 secondi), e cliccare sul **pulsante Run** per avviare la simulazione.



Prima di visualizzare la risposta , ne valutiamo alcuni parametri caratteristici mediante le formule viste a lezione

$$F(s) = \frac{14}{s^2 + 2s + 7}$$

Forma generale

$$F(s) = \frac{\mu \omega_n^2}{(s^2 + 2\xi \omega_n s + \omega_n^2)}$$

Determinazione del guadagno statico, della pulsazione naturale e dello smorzamento

$$\omega_n^2 = 7$$

$$\mu \omega_n^2 = 14$$



$$\omega_n = \sqrt{7} \approx 2.645 \text{ rad / sec}$$

$$\mu = 14 / \omega_n^2 = 14 / 7 = 2$$

$$2\xi \omega_n = 2$$



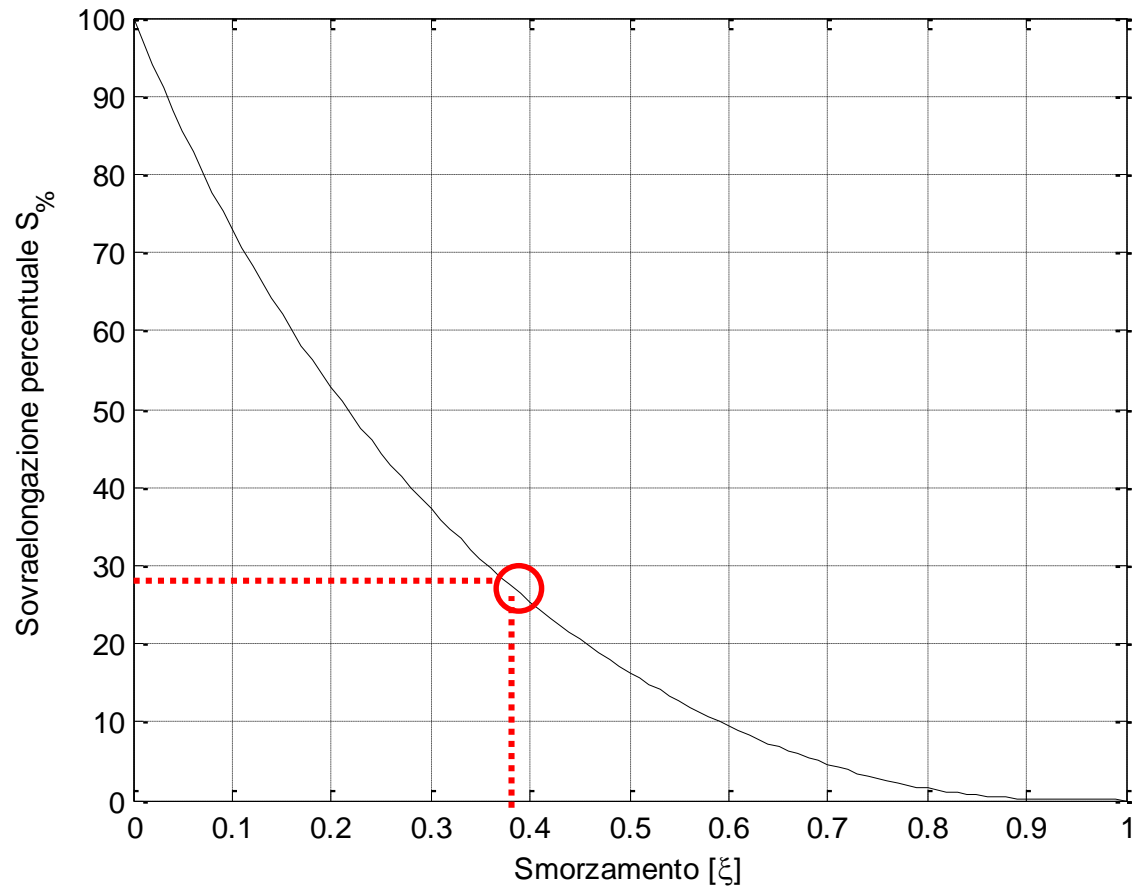
$$\xi = 1 / \omega_n \approx 0.37$$

Sovraelongazione percentuale

$$S_{\%} = 100e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}}$$

$$\xi \approx 0.37$$

$$S_{\%} \approx 28$$



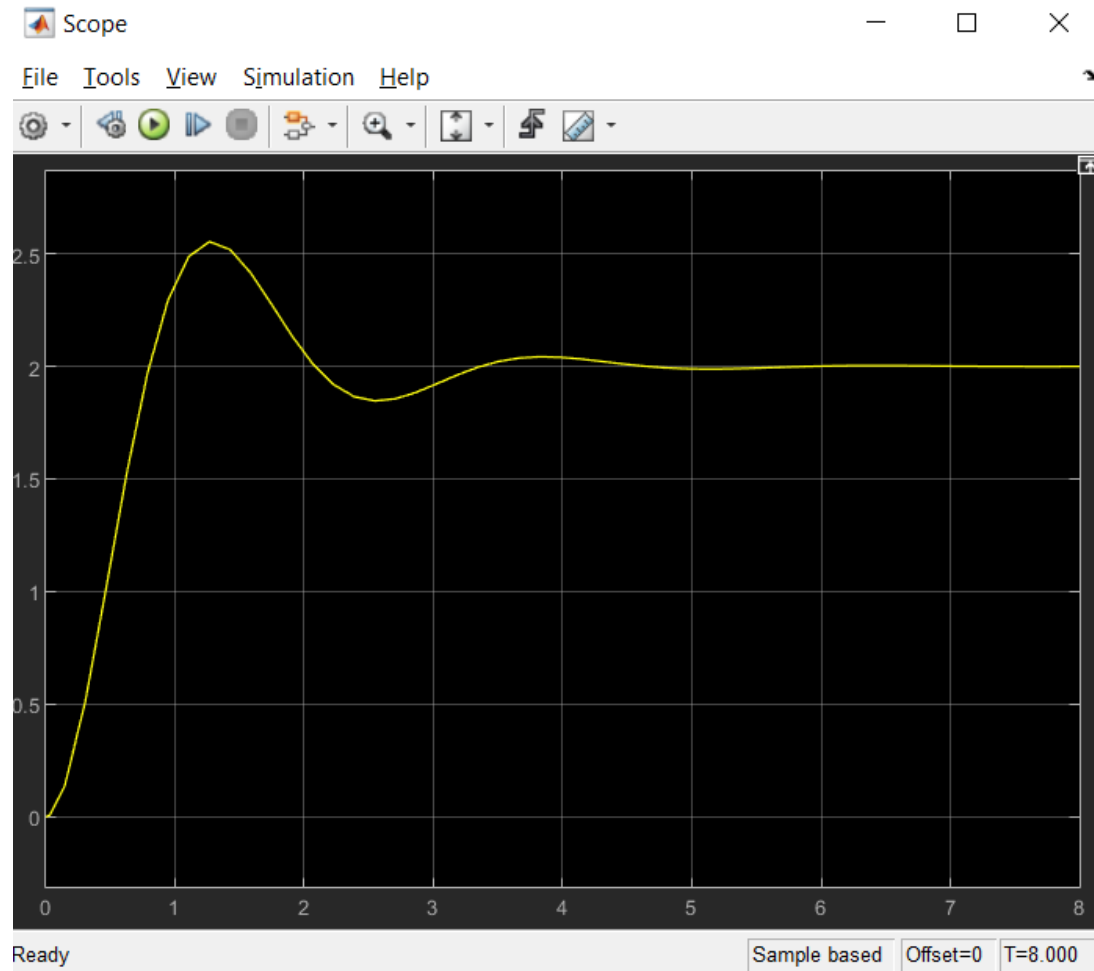
Il valore massimo dell'uscita durante il transitorio sarà pertanto

$$y_{max} = \mu + 0.28 \mu = 1.28\mu = 2.56$$

$$\tau_{eq} = \frac{1}{\xi \omega_n} \approx 1.02 \text{ s} \quad \text{Costante di tempo equivalente e tempi di assestamento}$$

	$T_{a5\%}$	$T_{a2\%}$	$T_{a1\%}$
$F(s) = \frac{\mu \omega_n^2}{(s^2 + 2\xi \omega_n s + \omega_n^2)}$	$3\tau_{eq}$	$4\tau_{eq}$	$5\tau_{eq}$
$F(s) = \frac{14}{s^2 + 2s + 7}$	3.06 s	4.08 s	5.10 s

Ora dopo avere eseguito la simulazione, fare doppio click sul blocco “Scope”. La risposta possiede le caratteristiche attese



FILE: Esemplio01_cicloaperto.slx₁₉

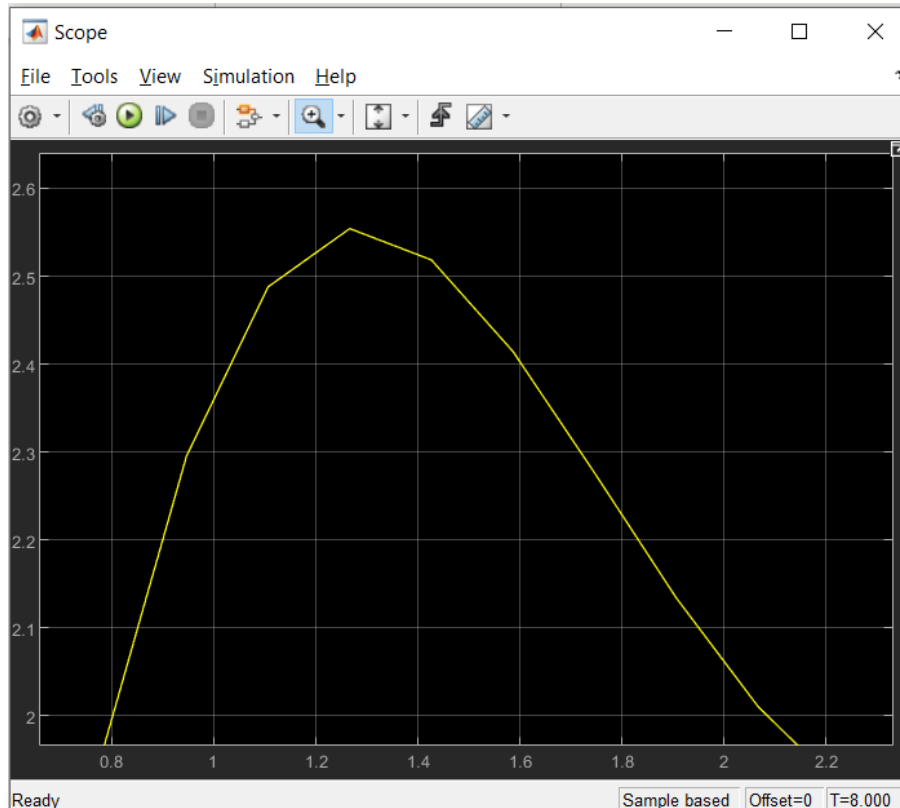
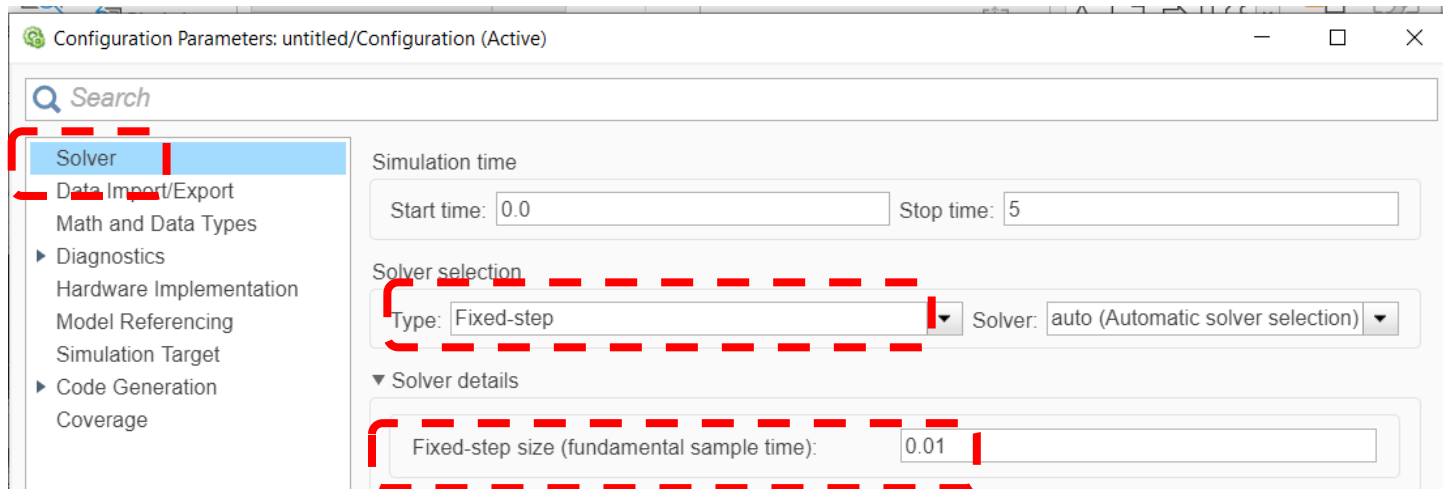


Grafico “spigoloso”

Il grafico è stato realizzato interpolando un numero di punti insufficiente (l’uscita del sistema è stata calcolata in corrispondenza di un numero insufficiente di istanti temporali tali da non consentire la creazione di un grafico sufficientemente regolare)

Dobbiamo modificare i parametri di configurazione del “**solver**”, che definisce (fra le altre cose) il passo di discretizzazione temporale che viene impiegato nella esecuzione del modello (cioè nella **risoluzione numerica delle equazioni differenziali** coinvolte nel modello).

Fare click con il tasto destro in qualunque punto dello schema e scegliere dal menu «Model Configuration Parameters» (Ctrl+E).

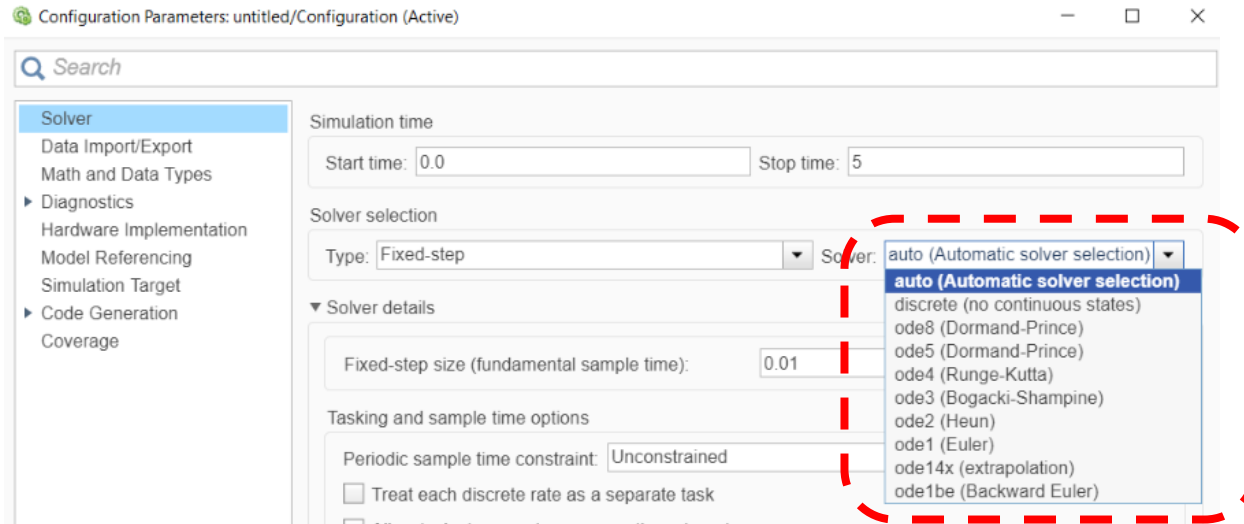


Nel menu Solver impostare il Solver Type ed il Fixed Step Size come in figura. In questo modo **i segnali della simulazione saranno aggiornati (ricalcolati) ogni 0.01 secondi.**

Queste impostazioni vanno eseguite ogni volta che si apre un nuovo modello vuoto (è possibile creare un template di configurazione che consente di creare direttamente un nuovo modello con le impostazioni desiderate)

Naturalmente se si avesse necessita di simulare un sistema dinamico in cui i segnali variano molto rapidamente la scelta di 0.01 secondi per il fixed-step size potrebbe diventare non più appropriata, ed il valore dovrebbe essere ulteriormente ridotto.

E' disponibile una ampia varietà di solutori numerici. Nello screenshot sottostante sono riportate le varie opzioni di scelta per i solutori a passo fisso.



Una **scelta ottimale per il solutore** bilancia, per il problema in esame, la precisione della soluzione e la mole di calcoli richiesta, che influenza il tempo di simulazione.

L'impostazione di scelta automatica del solver, che è la scelta di default, può dar luogo a soluzioni inaccurate in taluni casi «critici» come ad esempio modelli differenziali in cui intervengono segnali con scale di variazione temporale molto diverse fra loro (problemi «stiff») o modelli con elementi discontinui (non-smooth dynamics) e sono necessari solutori specifici.

Per una discussione più approfondita in merito alla scelta del solutore:

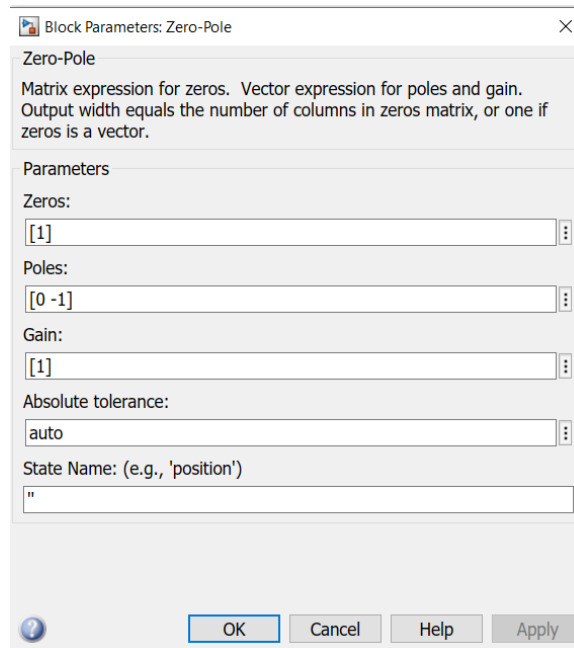
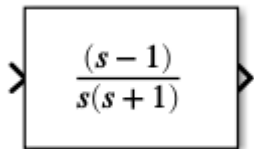
<https://www.mathworks.com/help/simulink/ug/choose-a-solver.html>

Vediamo **un altro blocco elementare per la modellazione di FdT**

Per modellare la seguente FdT il blocco Transfer Fcn risulta poco conveniente

$$F(s) = \frac{2(s + 0.5)}{(s + 1)(s + 2)(s + 5)}$$

Si può utilizzare in alternativa il blocco «Zero-Pole», che si trova sempre nella medesima libreria Continuous dalla quale prelevammo il blocco Transfer Fcn



A screenshot of the 'Block Parameters: Zero-Pole' dialog box. The dialog has a title bar with a close button. The main area contains the following fields:

- Zero-Pole**: Matrix expression for zeros. Vector expression for poles and gain. Output width equals the number of columns in zeros matrix, or one if zeros is a vector.
- Parameters**:
 - Zeros:** [1]
 - Poles:** [0 -1]
 - Gain:** [1]
 - Absolute tolerance:** auto
 - State Name:** (e.g., 'position')

At the bottom, there are buttons for '?', 'OK', 'Cancel', 'Help', and 'Apply'.

Vettore che contiene tutti gli zeri

Vettore che contiene tutti i poli

Guadagno in alta frequenza

$$F(s) = \frac{2(s + 0.5)}{(s + 1)(s + 2)(s + 5)}$$

Block Parameters: Zero-Pole [X]

Zero-Pole
Matrix expression for zeros. Vector expression for poles and gain.
Output width equals the number of columns in zeros matrix, or one if zeros is a vector.

Parameters

Zeros:
[-0.5]

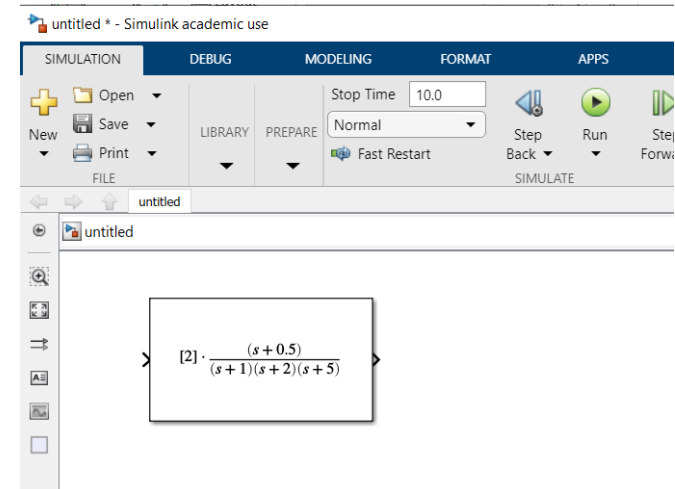
Poles:
[-1 -2 -5]

Gain:
[2]

Absolute tolerance:
auto

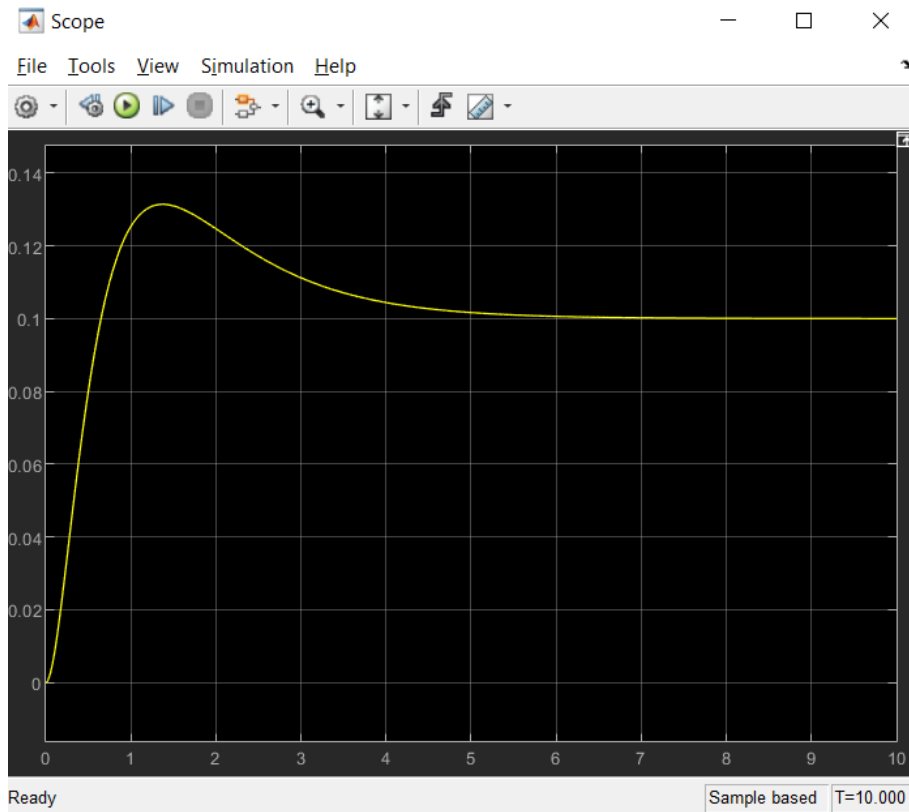
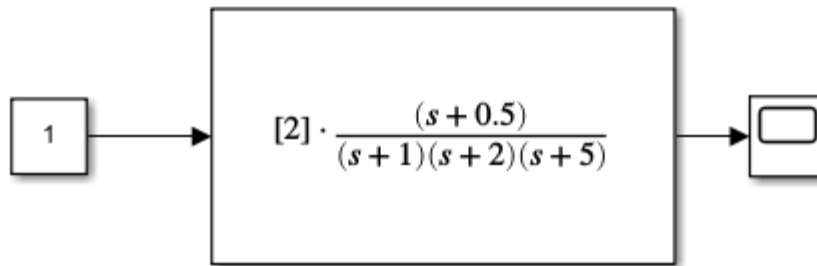
State Name: (e.g., 'position')
"

[?] [OK] [Cancel] [Help] [Apply]



Visualizzare la risposta al gradino.

Che tipo di curva ci aspettiamo ?



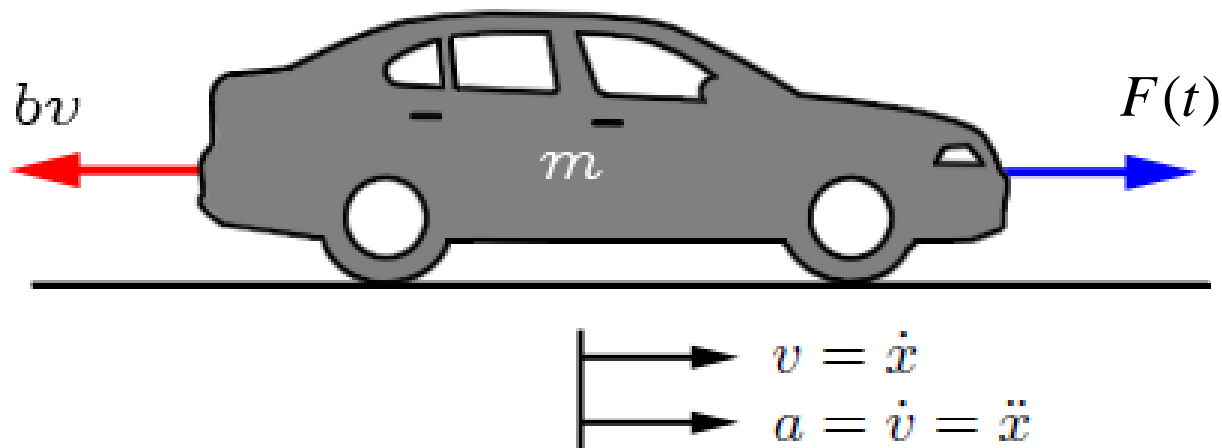
Sovraelongazione indotta dallo zero in bassa frequenza

SIMULAZIONE DEL CRUISE CONTROL

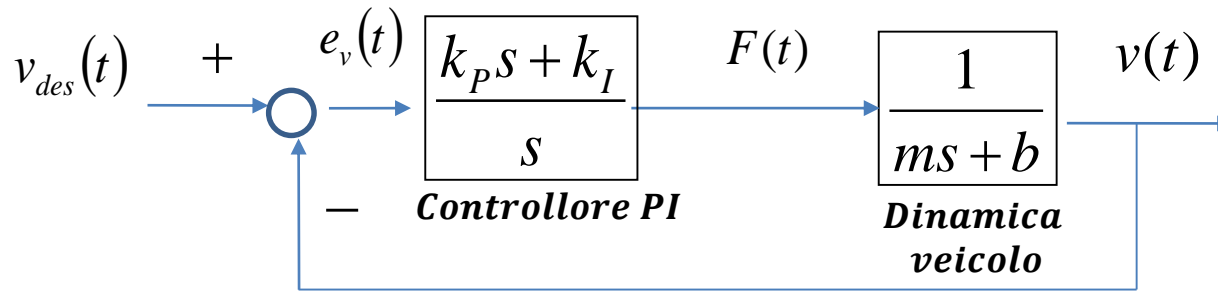
Il cruise control costituisce un semplice ma importante esempio di sistema di controllo di ampia diffusione in molti veicoli moderni.

L'obiettivo di controllo è mantenere una velocità di avanzamento costante a fronte di disturbi esterni quali ad es. il vento e la pendenza della carreggiata.

Ciò viene realizzato misurando la velocità del veicolo, confrontandola con il valore desiderato V_{des} (set-point) e variando automaticamente la spinta $F(t)$ secondo un certo algoritmo di controllo.



Schema a blocchi con controllore PI ricavato a lezione (pendenza della carreggiata nulla)



$$m = 800 \text{ kg}$$

$$b = 40 \text{ N s m}^{-1}$$

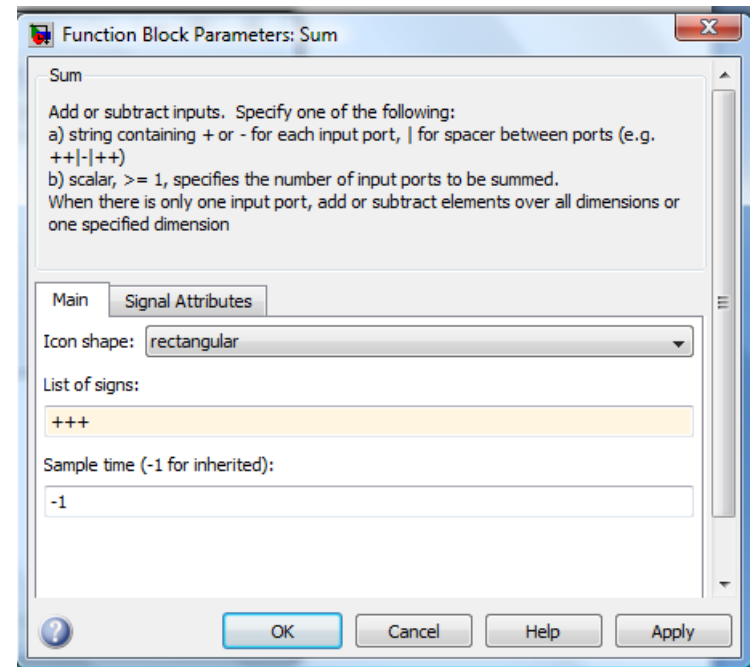
$$v_{des} = 25 \text{ m/s}$$

Dobbiamo disegnare in Simulink lo schema a blocchi mostrato in figura.

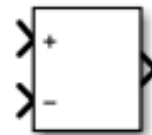
Rispetto ai blocchi elementari impiegati nell'esempio precedente ne serve uno aggiuntivo che implementi il **nodo sommatore** mediante il quale calcolare la differenza fra il set point e la velocità del veicolo.

Il nodo sommatore lo si implementa mediante il blocco **Sum** dalla libreria Math Operations (si trova anche nella libreria Commonly Used Blocks)

Il blocco **Sum** è parametrizzato per mezzo di una stringa (es. **+** **+** **-** **+**) la cui lunghezza corrisponde al numero di segnali in ingresso al blocco mentre il segno **+** o **-** definisce se il corrispondente ingresso sia da sommare agli altri termini o da sottrarre.



Scegliamo **+** **-** e impostiamo la Icon shape in rectangular.
L'aspetto del blocco diventa

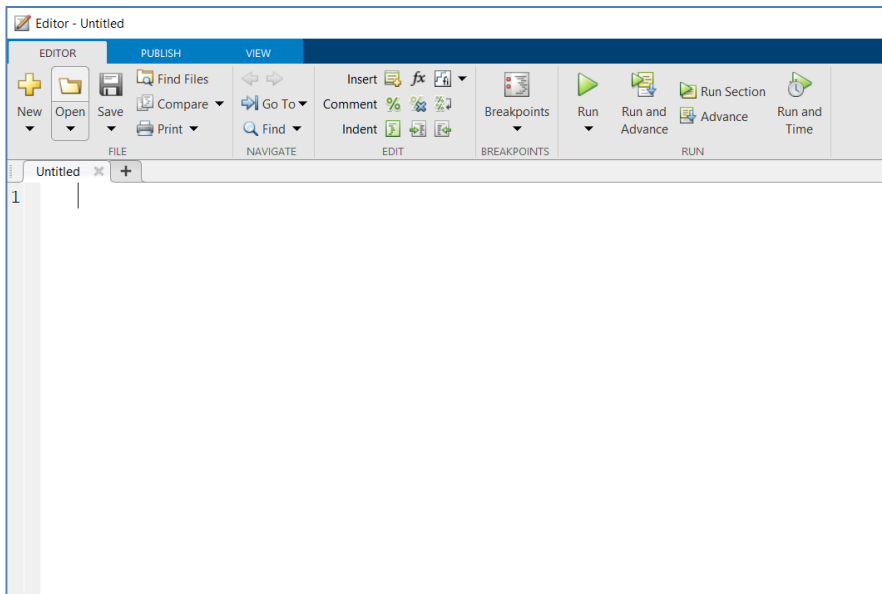


Realizzeremo il modello del Cruise Control in forma **parametrizzata**, utilizzando delle costanti simboliche associate ai parametri del sistema (m e b) ed ai guadagni del controllore (k_p e k_i)

Tali costanti simboliche vengono definite nel **workspace** di Matlab e possono essere impiegate all'interno dei modelli Simulink, che hanno accesso alle variabili presenti nel workspace

Per definire tali costanti simboliche deve essere scritto ed eseguito uno **Script File**.

Nella finestra di avvio di Matlab premere il primo pulsante sulla sinistra: **New Script**



Si apre l'editor di testo

Inserire le seguenti istruzioni e premere il pulsante Run

Editor - Untitled*

EDITOR PUBLISH VIEW

New Open Save Find Files Compare Go To Find Insert Comment Indent Breakpoints Run Run and Advance

FILE NAVIGATE EDIT BREAKPOINTS RUN

Untitled* x +

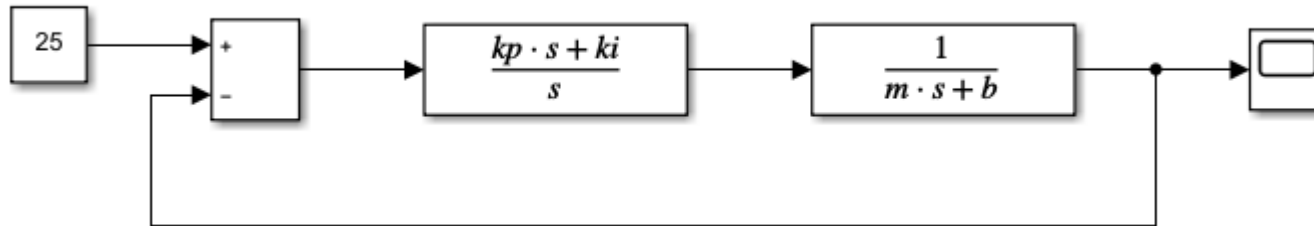
```
1 m=1000;  
2 b=40;  
3 kp=200;  
4 ki=100;  
5 |
```

Workspace

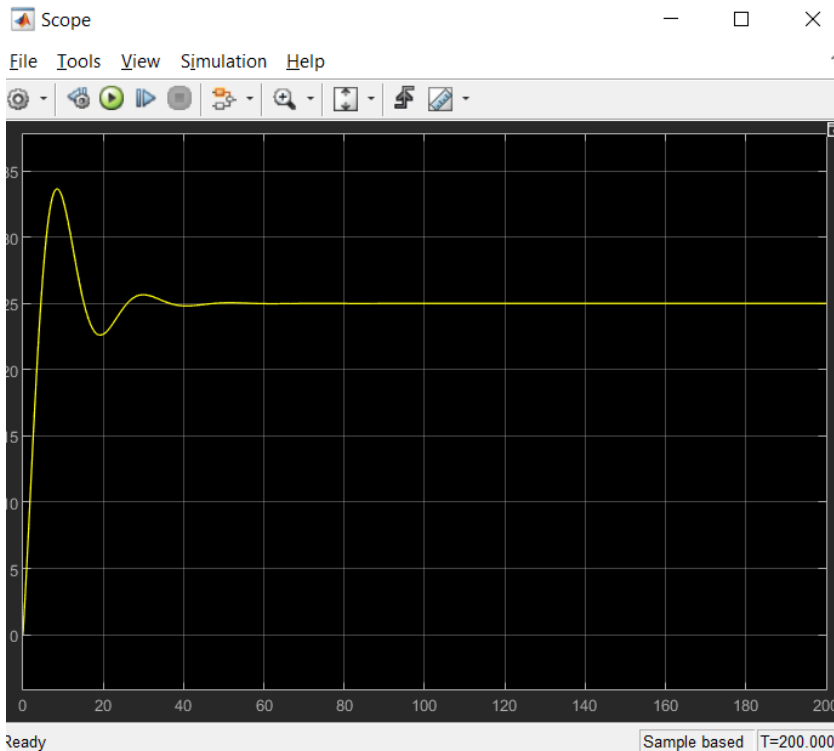
Name ^	Value	Size
b	40	1x1
ki	100	1x1
kp	200	1x1
m	1000	1x1

Nel workspace compaiono le variabili appena definite

Ora realizziamo il modello, che in Simulink assumerà il seguente aspetto:



Impostiamo la durata della simulazione a 200 secondi e lanciamo il test.



Trattandosi di un sistema di controllo di tipo 1, la velocità converge al valore costante del set point

Come mai abbiamo ottenuto questo andamento ?

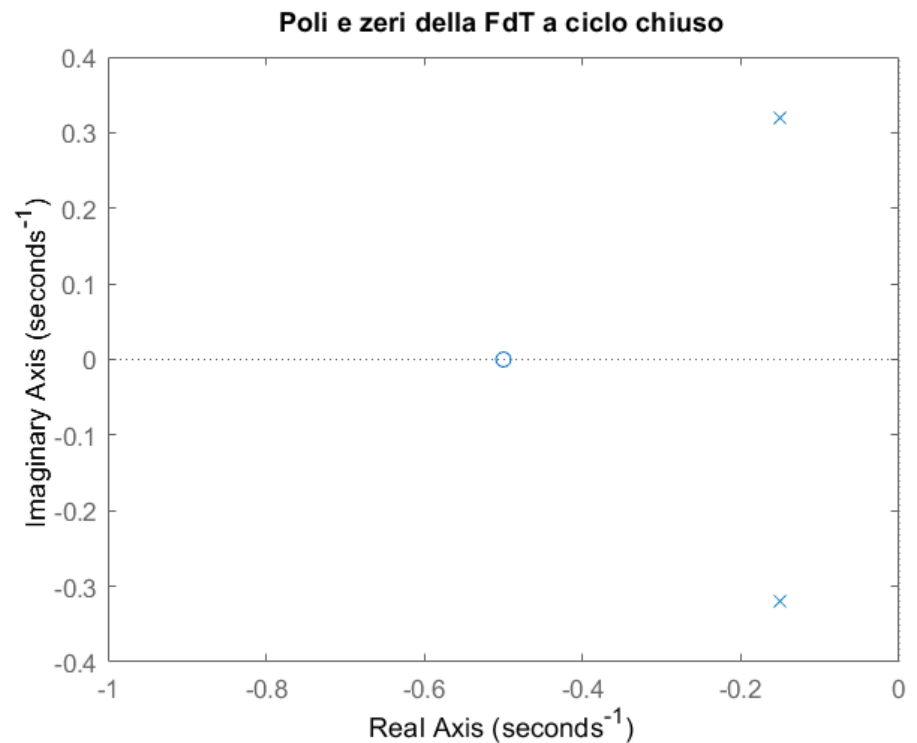
$$W_{vdes}^v(s) = \frac{k_p s + k_i}{s(ms + b) + k_p s + k_i} = \frac{200s + 100}{800s^2 + 240s + 100}$$

```
m=800;
b=40;
kp=200;
ki=100;
```

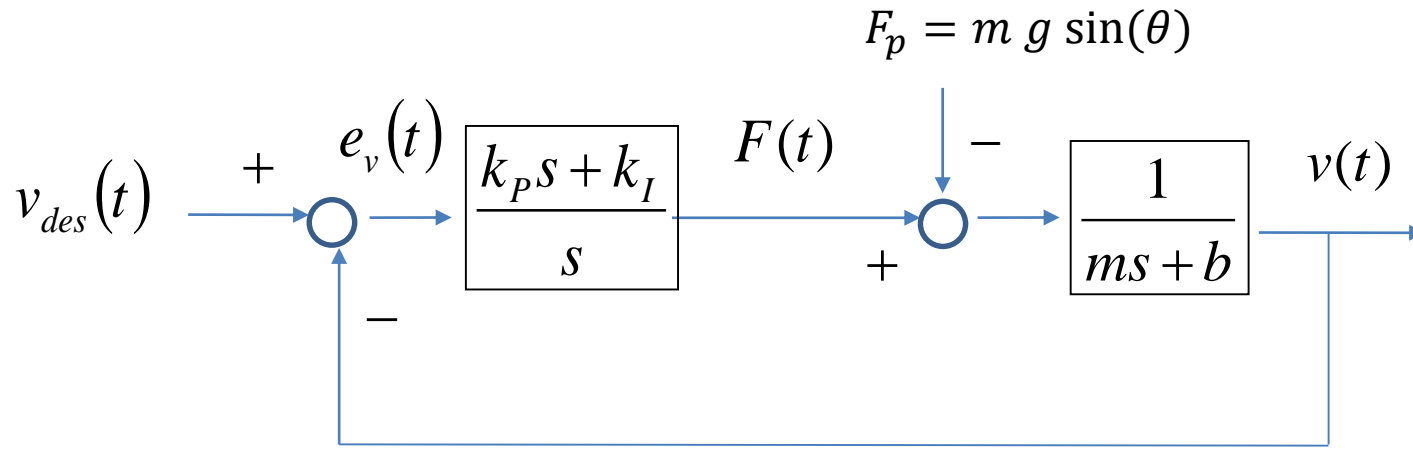
```
W=tf([kp ki],[m b+kp ki])
pzmap(W)
title('Poli e zeri della FdT a ciclo chiuso')
xlim([-1 0])
```

Mappa dei poli e degli zeri della FdT a ciclo chiuso fra il set point e l'uscita.

Si correla la mappa poli zeri alle caratteristiche della risposta al gradino visualizzata nella slide precedente

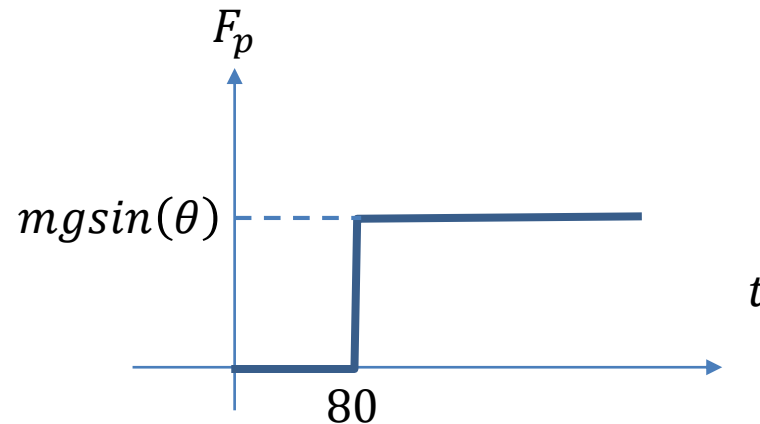


Ora includiamo la **pendenza della carreggiata**

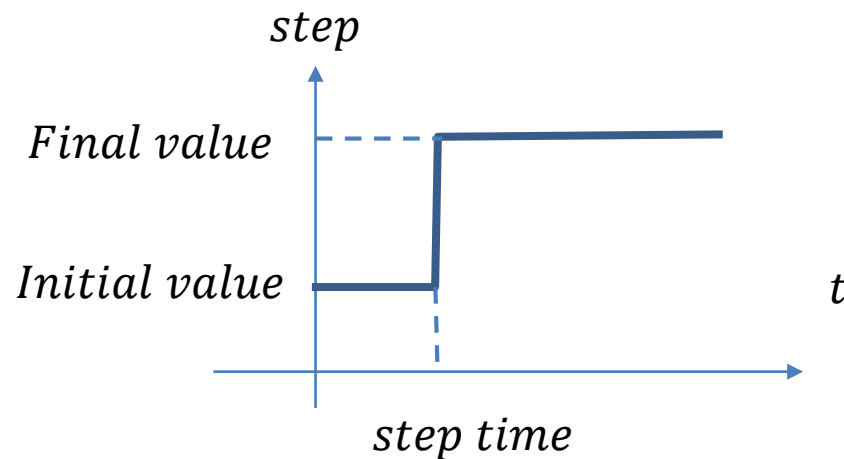


Vogliamo rappresentare la situazione in cui il veicolo viaggia per 80 secondi su una carreggiata a pendenza nulla, e a $t=80$ la pendenza cambia bruscamente da 0 a 10°

Ciò corrisponde ad un segnale F_p (forza peso) con l'andamento seguente



Realizziamo il segnale F_p mediante il blocco Step (Libreria Sources). Il blocco step genera un segnale avente la forma riportata nella figura sottostante. I parametri sono i valori iniziale e finale, e l'istante della transizione (step time).



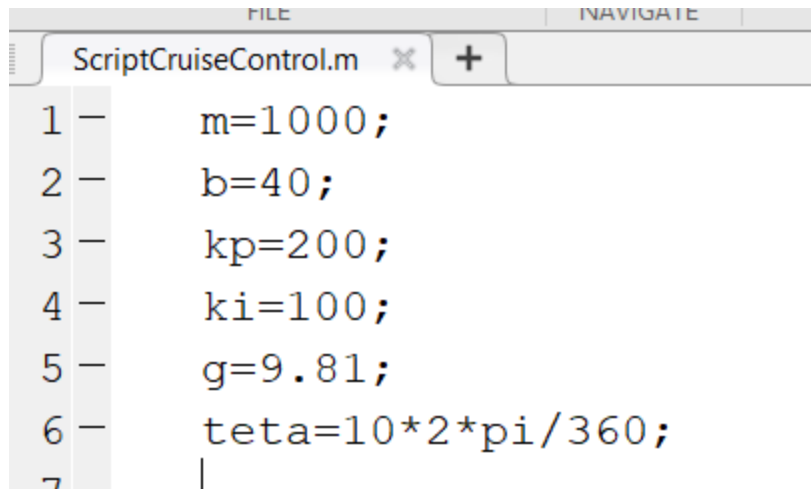
Parametri di configurazione del blocco Step

Initial value = 0

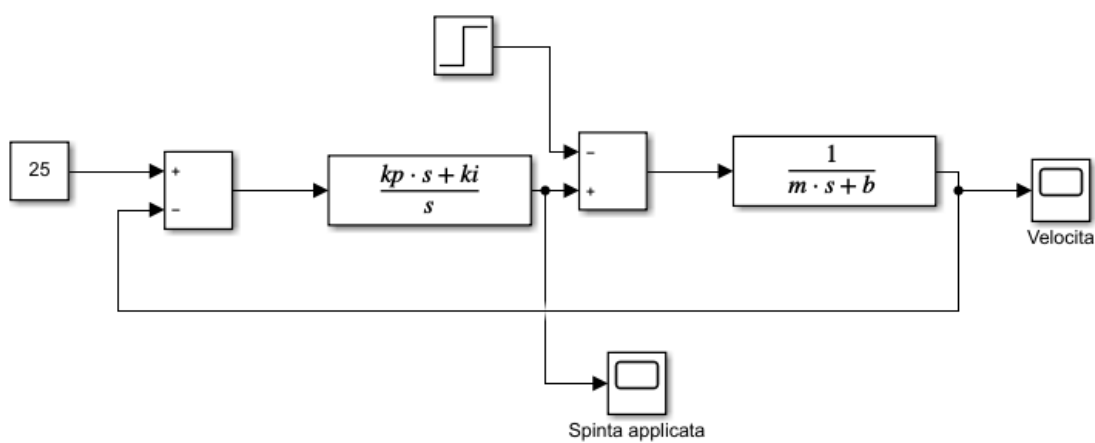
Final value = $m g \sin(\theta)$

Step time = 80

Inseriamo nello script due istruzioni aggiuntive per definire la costante di gravità e l'angolo di inclinazione della carreggiata, pari a 10 gradi



```
FILE NAVIGATE
ScriptCruiseControl.m x +
1 - m=1000;
2 - b=40;
3 - kp=200;
4 - ki=100;
5 - g=9.81;
6 - theta=10*2*pi/360;
7 - |
```



File: Esempio02_cruisecontrol_pendenza.slx

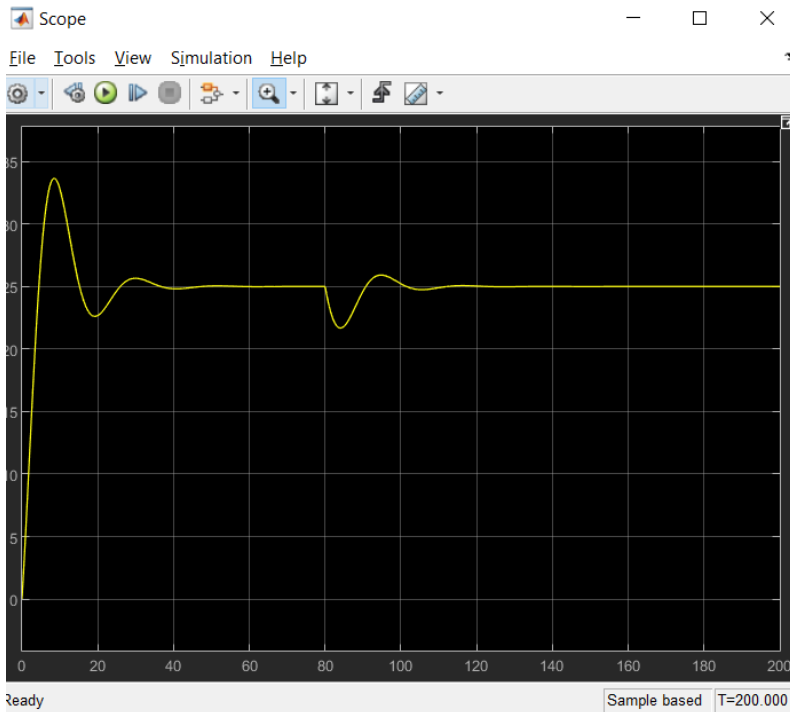
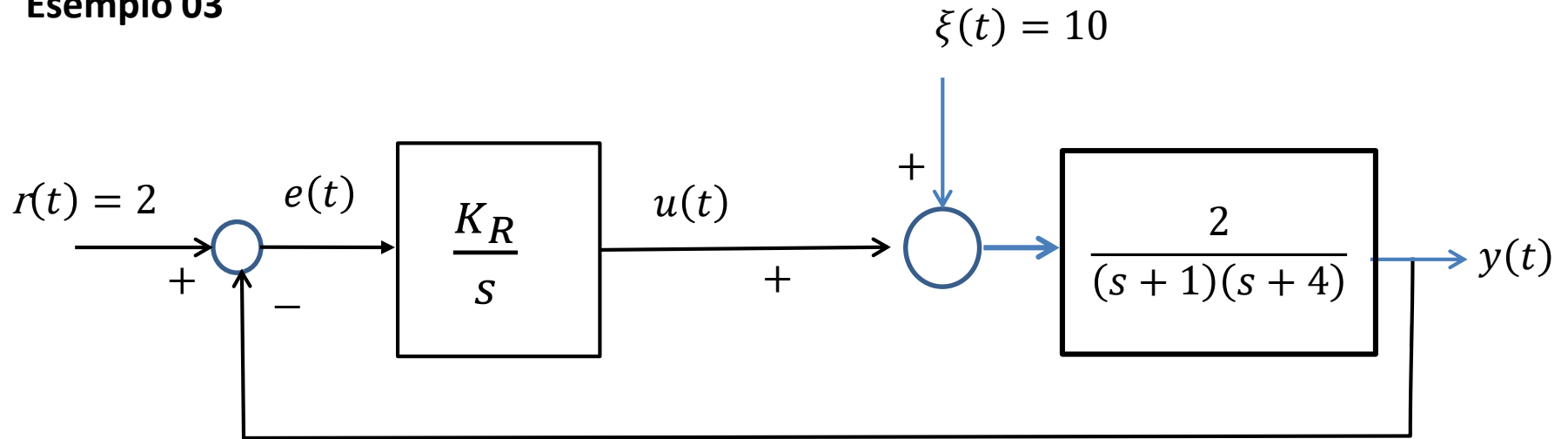


Grafico della velocità

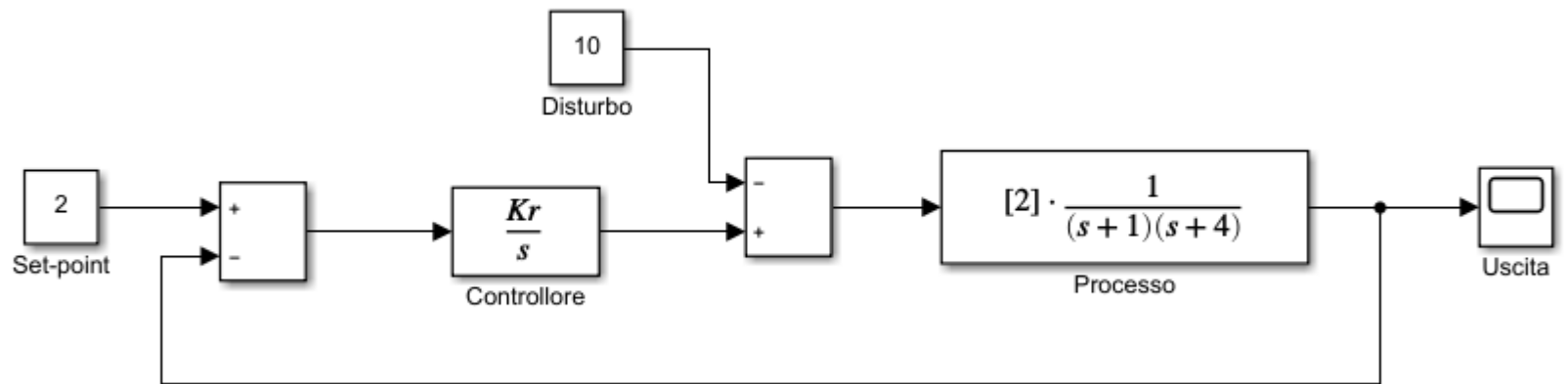
Visualizzare anche il grafico della spinta applicata al veicolo, e interpretarne l'andamento.

Esempio 03



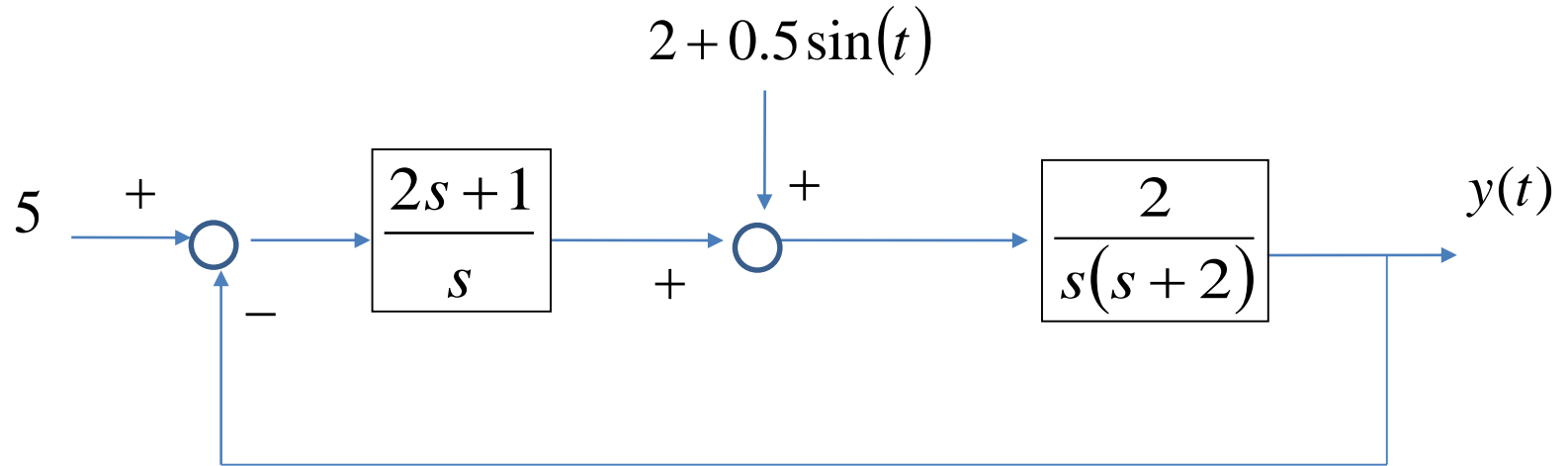
Dato il sistema di controllo in figura, valutare mediante simulazione dinamica il comportamento di regime dell'uscita in corrispondenza dei seguenti valori per il guadagno K_R :

$$K_R = 1 \quad K_R = 5 \quad K_R = 20.$$



File: Esemplio03.slx

Esempio 04



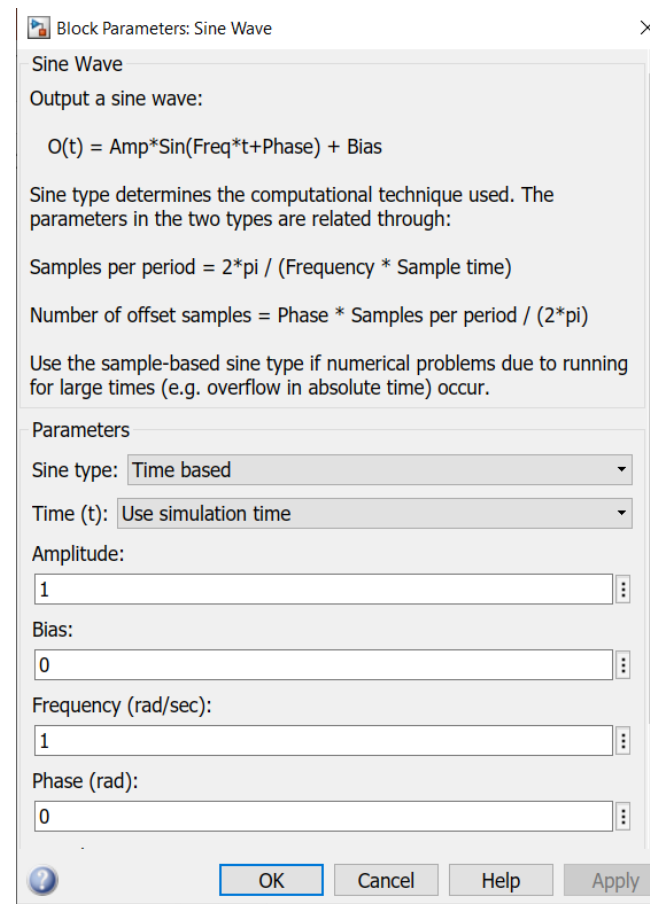
Dato il sistema di controllo in figura, valutare mediante simulazione dinamica il comportamento di regime dell'uscita.

Generazione di un **segnale sinusoidale**.

Importiamo il blocco **Sine Wave** dalla libreria Sources.

Finestra di parametrizzazione con i valori di default, che producono in uscita il segnale $\sin(t)$.

Per generare il segnale $2 + 0.5 \sin(t)$ si parametrizzi il blocco come mostrato nella slide successiva.



Sine Wave

Output a sine wave:

$$O(t) = \text{Amp} * \sin(\text{Freq} * t + \text{Phase}) + \text{Bias}$$

Puo essere impostato anche un valore costante di **Bias** (v. formula in alto). Al termine della configurazione premere il tasto OK.

Ampiezza

Bias

Frequenza

Sfasamento

Tasto OK

Block Parameters: Sine Wave

Sine Wave

Output a sine wave:

$$O(t) = \text{Amp} * \sin(\text{Freq} * t + \text{Phase}) + \text{Bias}$$

Sine type determines the computational technique used. The parameters in the two types are related through:

Samples per period = $2 * \pi / (\text{Frequency} * \text{Sample time})$

Number of offset samples = $\text{Phase} * \text{Samples per period} / (2 * \pi)$

Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.

Parameters

Sine type: Time based

Time (t): Use simulation time

Amplitude: 0.5

Bias: 2

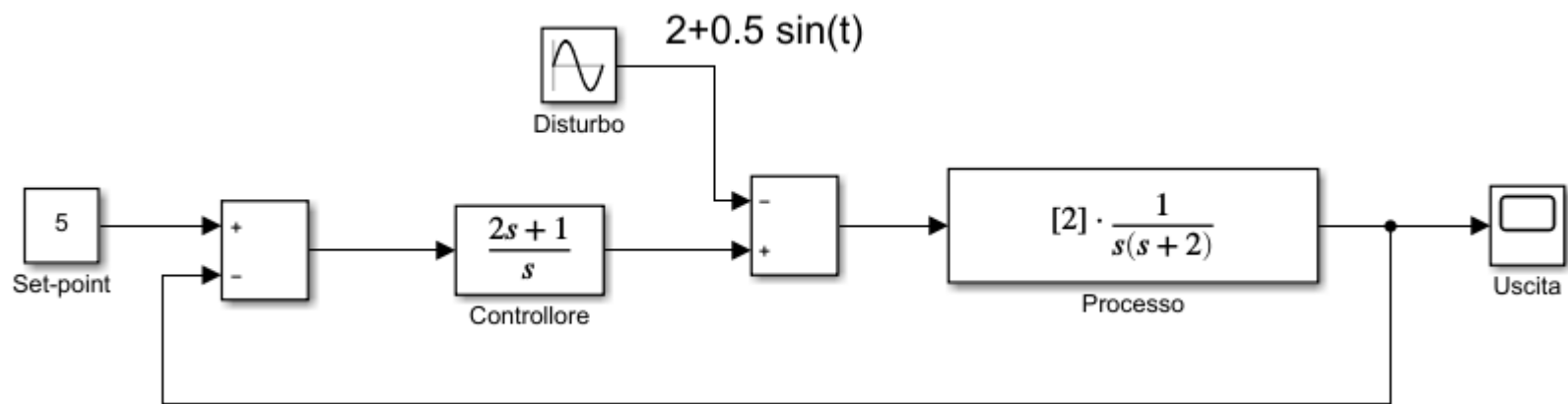
Frequency (rad/sec): 1

Phase (rad): 0

Sample time: 0

☒ Interpret vector parameters as 1-D

OK Cancel Help Apply



File: Esemplio04.slx