

# PROGETTO DI UN CONTROLLO AUTOMATICO PER UN BOOST CONVERTER



## Controlli Automatici

Corso di Laurea Magistrale in Ingegneria Elettronica LM-29

*Anno Accademico 2016/2017*

**Elaborato a cura di**

*Antonella Pulitanò*

*Antonino Chilà*

*Domenico Caccamo*

# STRUTTURA TESINA

**1. DIMENSIONAMENTO E SCELTA DEI COMPONENTI**

**2. MODELLIZZAZIONE MATEMATICA**

**3. SINTESI E VERIFICHE DEL CONTROLLO**

3.1 Loop shaping

3.2 Verifiche del controllo tempo continuo

3.3 Alternativa: Osservatore asintotico

**4. DISCRETIZZAZIONE DEL CONTROLLO**

4.1 Trasformazione di Tustin

4.2 Alternativa: metodo con ZOH

4.3 Verifiche del controllo tempo discreto

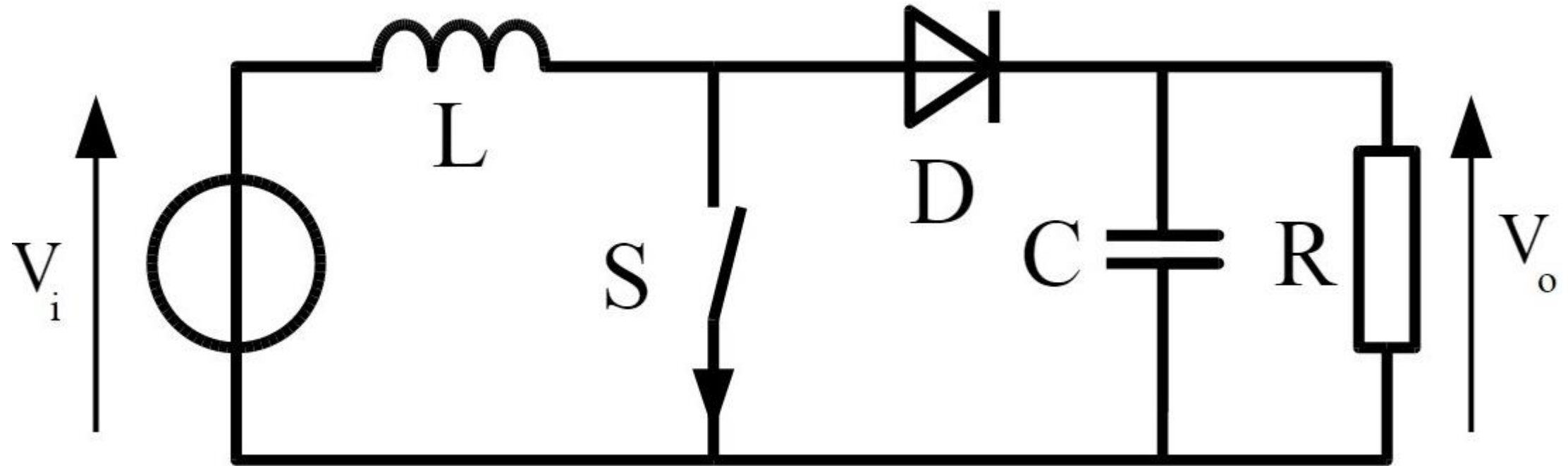
**5. VERIFICHE AL VARIARE DEL CARICO (SIMULINK)**

**6. IMPLEMENTAZIONE E VERIFICHE SPERIMENTALI**

# **1. DIMENSIONAMENTO E SCELTA DEI COMPONENTI**

# INTRODUZIONE

Il Boost Converter è un convertitore DC/DC ai cui morsetti in ingresso è applicata una  $V_i$  continua: in uscita, fornisce una tensione (sempre continua) pari a  $V_o = V_o(V_i, D)$  con  $D$  duty-cycle (parametro di controllo del circuito). Esso è un step-up con una tensione d'uscita maggiore di quella d'ingresso, che viene usato in varie applicazioni come nei sistemi di illuminazione.

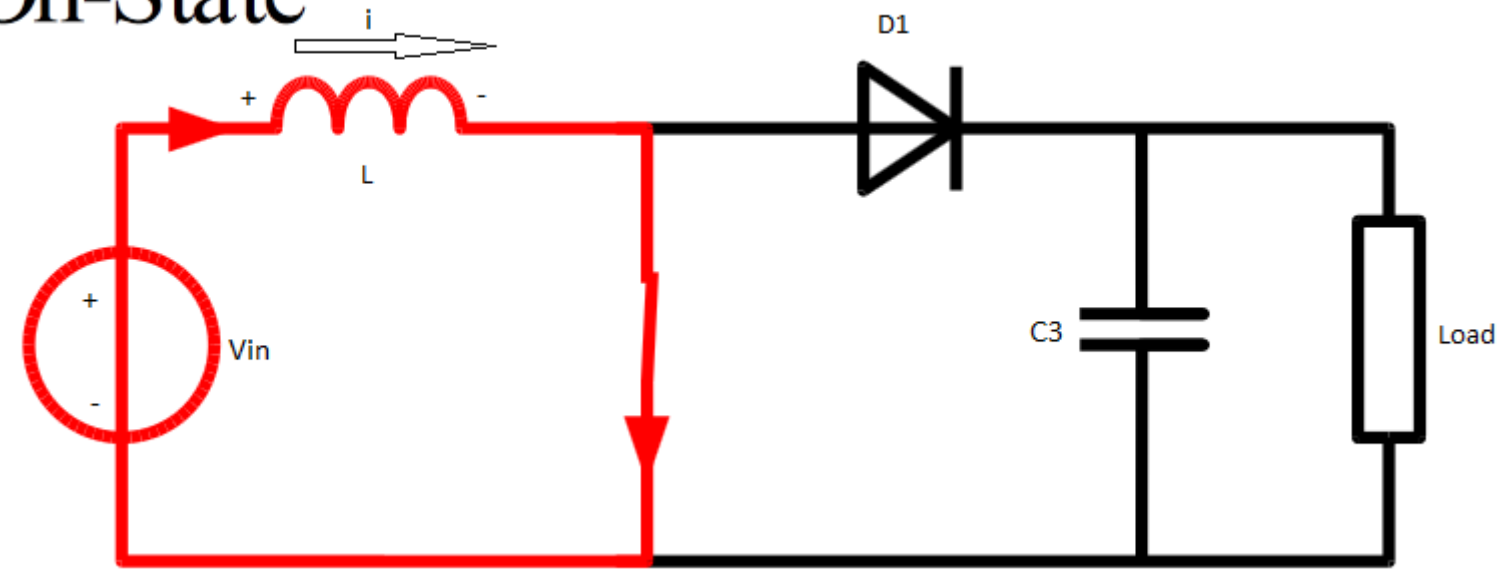


$L$  è inserita in serie all'ingresso e ha lo scopo di minimizzare la  $\Delta I$ ;  $C$  serve per minimizzare la  $\Delta V$  sul carico  $R$ . L'interruttore è posto verticalmente tra induttore e diodo, dividendo in due parti il circuito. Si definisce il periodo di chopping come la somma del tempo in cui  $T$  è aperto e il tempo in cui  $T$  è chiuso:  $T_C = t_{ON} + t_{OFF}$ .

➤  $t_{ON}$  (interruttore chiuso)

$$\begin{cases} \dot{I}_L = \frac{V_{in}}{L} \\ \dot{V}_C = -\frac{V_C}{R \cdot C} \end{cases}$$

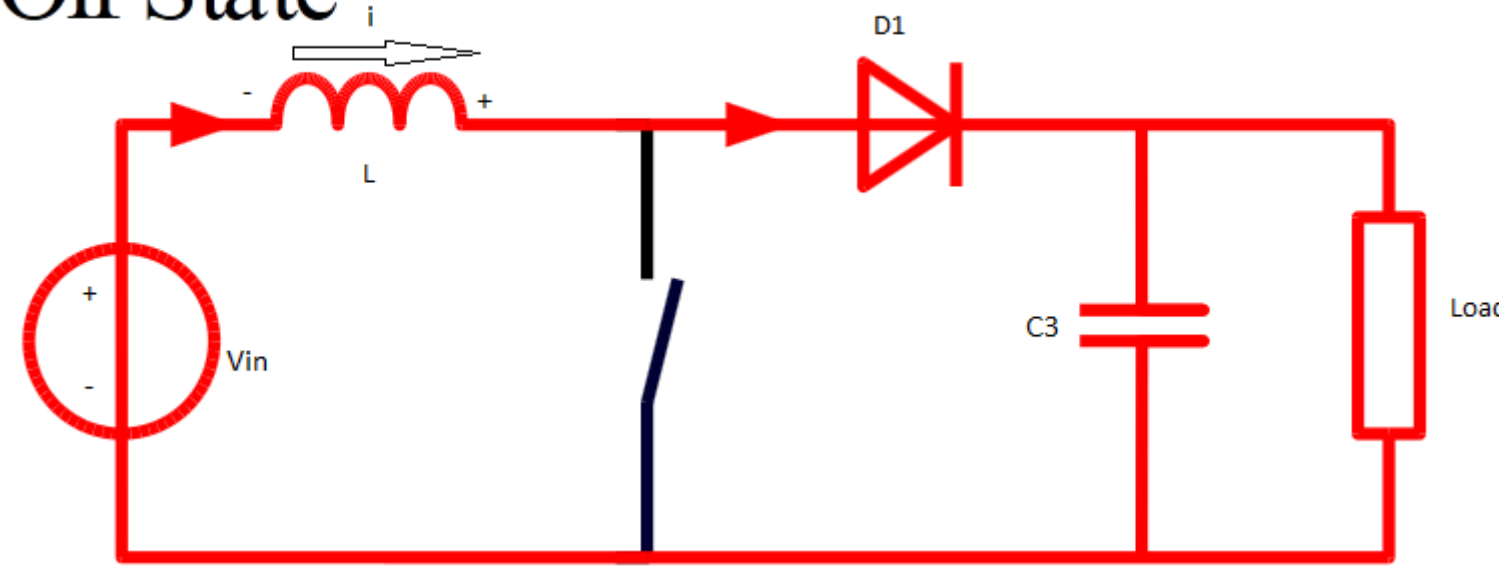
### On-State



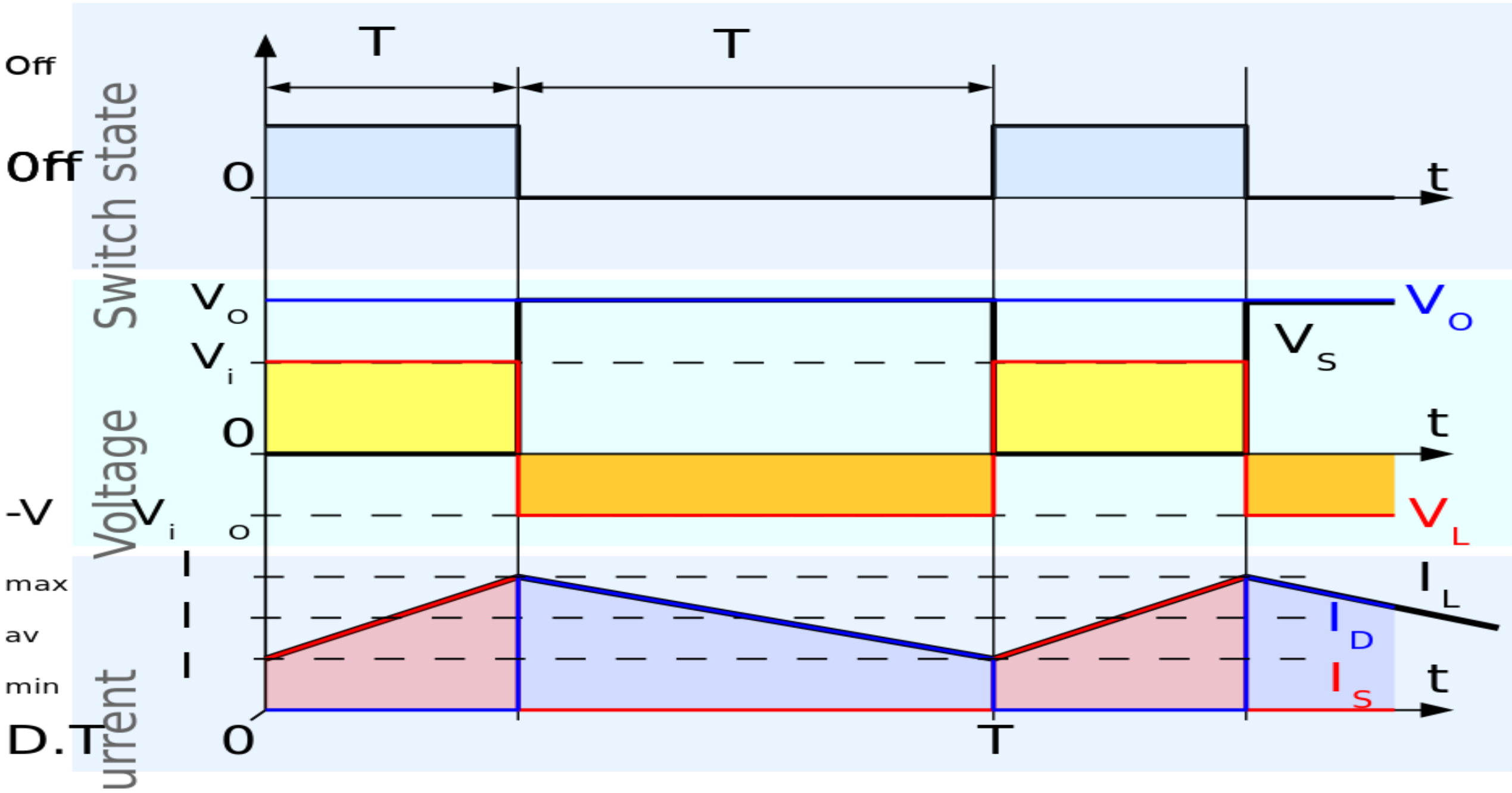
➤  $t_{OFF}$  (interruttore aperto)

$$\begin{cases} \dot{I}_L = \frac{V_{in}}{L} - \frac{V_C}{L} \\ \dot{V}_C = \frac{I_L}{C} - \frac{V_C}{R \cdot C} \end{cases}$$

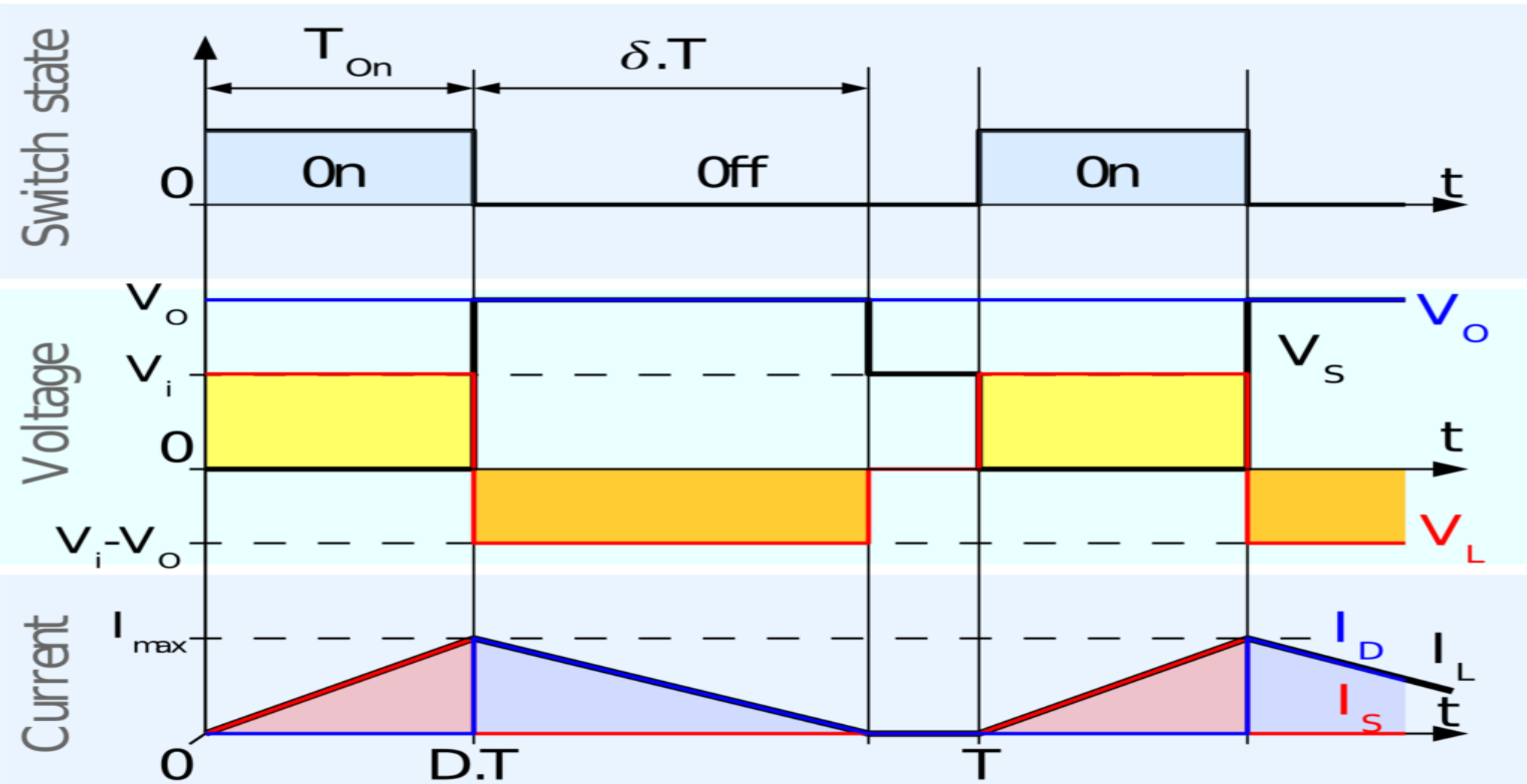
### Off-State



# ANDAMENTO C.C.M.



# ANDAMENTO D.C.M.



# LIMITE TRA C.C.M. E D.C.M.

- Al limite della conduzione continua, la corrente dell'induttanza va a zero alla fine del periodo di OFF.

- Il valore medio della corrente nell'induttore a questo limite è:

$$I_{Lb} = \frac{1}{2} I_{L,peak} = \frac{V_o \cdot T \cdot D \cdot (1-D)}{2 \cdot L} .$$

- La corrente d'uscita media al limite è

$$I_{ob} = \frac{V_o \cdot T \cdot D \cdot (1-D)^2}{2 \cdot L} .$$

- Conoscendo la  $I_{ob}$ , e poiché la  $V_o$  è costante, possiamo trovare il valore massimo del carico entro il quale mantenerci per evitare che la  $I_o$  vada al di sotto della  $I_{ob}$ :

$$R_{load,max} = \frac{V_o}{I_{ob}} .$$



# SPECIFICHE DI PROGETTO

➤  $V_i = 5V, V_o = 15V$

➤  $I_{o,max} = 100\text{ mA}$

➤  $f_{PWM} = 153.850\text{ Hz}$  (più è alta, più sono basse C e L necessarie)

➤  $\Delta I_L = 5\% \cdot I_L = 15\text{ mA}$

➤  $\Delta V_C = 2\% \cdot V_o = 0,3\text{ V}$

# PARAMETRI CALCOLATI PER GARANTIRE LA C.C.M.

$$\blacktriangleright L \geq \frac{V_i \cdot D}{\Delta I_L \cdot f_{PWM}} = 1,45 \text{ mH} \rightarrow L = 1,8 \text{ mH}$$

$$\blacktriangleright C \geq \frac{I_o \cdot D}{\Delta V_C \cdot f_{PWM}} = 1,45 \text{ }\mu\text{F} \rightarrow C = 22 \text{ }\mu\text{F}$$

$$\blacktriangleright D = 1 - \frac{V_i}{V_o} = 0.66666667$$

$$\blacktriangleright I_o \leq 0,1 \text{ A (da specifica)} \rightarrow R_{load,min} = 150 \text{ }\Omega$$

$$\blacktriangleright I_o \geq I_{ob} = 2 \text{ mA (per garantire la C.C.M)} \rightarrow R_{load,max} = 7477,11 \text{ }\Omega$$

# Altri componenti usati per il boost

➤ DIODO SCHOTTKY 1N5819

➤ MOSFET IRF510

## **2. MODELLIZZAZIONE MATEMATICA**

# MODELLI NON LINEARE E MEDIATO DEL SISTEMA

Associamo: all'interruttore chiuso  $A = 1$ ; all'interruttore aperto  $A = 0$  (cioè  $\bar{A} = 1 - A$ ). Sommando i due sistemi, si ottiene il **sistema non lineare**:

$$\begin{cases} \dot{I}_L = \frac{V_i}{L} A + \left(\frac{V_i - V_C}{L}\right) \cdot (1 - A) \\ \dot{V}_C = -\frac{V_C}{R \cdot C} A + \left(\frac{I_L}{C} - \frac{V_C}{R \cdot C}\right) \cdot (1 - A) \cdot \\ y = V_C \end{cases}$$

Si definisce successivamente un periodo temporale per avere un modello mediato:

$$A = \frac{D}{T}, \quad 1 - A = \frac{T - D}{T};$$

Il duty-cycle è definito così:  $D = \frac{t_{ON}}{T} = \frac{t_{ON}}{t_{ON} + t_{OFF}}$ .

L'analisi del modello medio (average model analysis) è un metodo per calcolare la media nel tempo delle forme d'onda in un circuito a commutazione. Esso consiste

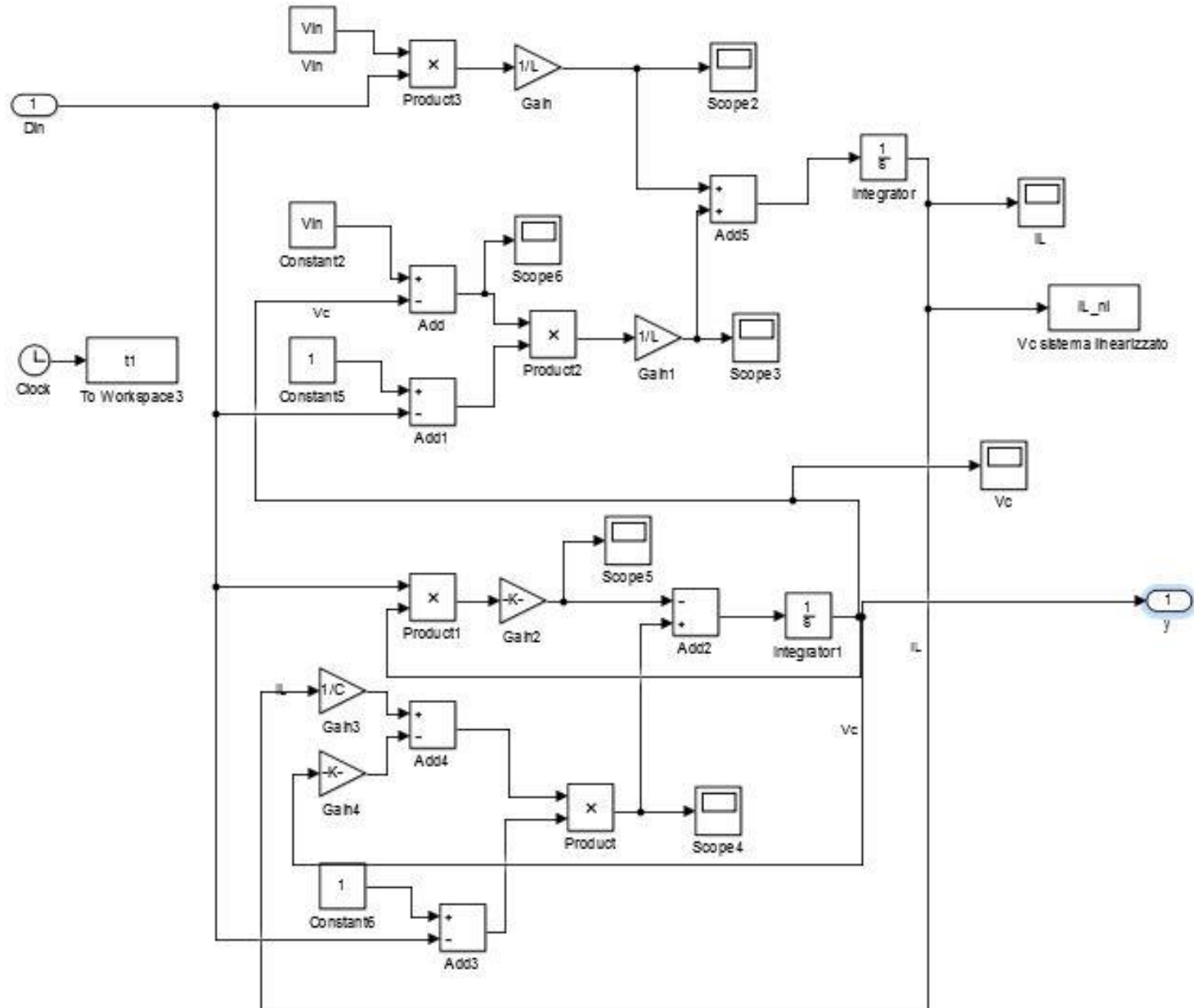
nello scrivere le equazioni corrispondenti ad ogni stato di lavoro del convertitore (in questo caso due stati: on e off e poi moltiplicarle per il tempo che il convertitore spende in ogni stato). Il sistema non lineare cambia:

$$\begin{cases} \dot{I}_L = \frac{V_i D}{L T} + \left(\frac{V_i - V_C}{L}\right) \cdot \left(1 - \frac{t_{ON}}{T}\right) \\ \dot{V}_C = -\frac{V_C D}{R \cdot C T} + \left(\frac{I_L}{C} - \frac{V_C}{R \cdot C}\right) \cdot \left(1 - \frac{t_{ON}}{T}\right) \cdot \\ y = V_C \end{cases}$$

Definiamo una variabile  $\hat{u} = \frac{t_{ON}}{T} = D$ , dove  $\hat{u} \in [0, 1]$ ; sostituendo, avremo il **modello mediato**:

$$\begin{cases} \dot{I}_L = \frac{V_i}{L} \hat{u} + \left(\frac{V_i - V_C}{L}\right) \cdot (1 - \hat{u}) \\ \dot{V}_C = -\frac{V_C}{R \cdot C} \hat{u} + \left(\frac{I_L}{C} - \frac{V_C}{R \cdot C}\right) \cdot (1 - \hat{u}) \cdot \\ y = V_C \end{cases}$$

# Modello non lineare mediato (Simulink)



# LINEARIZZAZIONE

Scriviamo il modello mediato, visto in precedenza, nella **forma generale di ISU** :

$$\begin{cases} \dot{x}(t) = f(t, x(t), u(t)) = A(t) \cdot x(t) + B(t) \cdot u(t) \\ y(t) = \eta(t, x(t), u(t)) = C(t) \cdot x(t) + D(t) \cdot u(t) \end{cases}$$

Linearizziamo il sistema non lineare e lo studiamo con le tecniche dei sistemi lineari. Pertanto, si supponga di conoscere l'evoluzione  $\hat{x}$  corrispondente allo stato iniziale  $\widehat{x}_0$  e all'ingresso  $\hat{u}$  del sistema di sopra. Per uno stato iniziale  $x_0 = \widehat{x}_0 + \delta x_0$  ed un ingresso  $u = \hat{u} + \delta u$ , si passa all'applicazione delle formule:

$$\begin{aligned} \blacktriangleright \dot{x}(t) &= \hat{x}(t) + \delta \dot{x}(t) = f(t, \hat{x}(t) + \delta x(t), \hat{u}(t) + \delta u(t)) = \\ &= f(t, \hat{x}(t), \hat{u}(t)) + \frac{\partial f}{\partial x \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}}} (\delta x(t)) + \frac{\partial f}{\partial u \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}}} (\delta u(t)) + h_f(t) \end{aligned}$$



$$\begin{aligned} \blacktriangleright y(t) &= \hat{y}(t) + \delta y(t) = \eta(t, \hat{x}(t) + \delta x(t), \hat{u}(t) + \delta u(t)) = \\ &= \eta(t, \hat{x}(t), \hat{u}(t)) + \frac{\partial \eta}{\partial x} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}} (\delta x(t)) + \frac{\partial \eta}{\partial u} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}} (\delta u(t)) + h_\eta(t) \end{aligned}$$

dove  $h_f(t)$  e  $h_\eta(t)$  sono infinitesimi di ordine superiore rispetto a  $\begin{pmatrix} \delta x \\ \delta u \end{pmatrix}$ . Nell'ipotesi che i due infinitesimi si possano trascurare (condizione necessaria perché ciò possa essere fatto è che siano sufficientemente piccole le perturbazioni rispettivamente dello stato iniziale  $\delta x_0$  e dell'ingresso  $\delta u$ ), si ottiene il **sistema ISU linearizzato**:

$$\begin{cases} \delta \dot{x}(t) = A(t) \cdot \delta x(t) + B(t) \cdot \delta u(t) \\ \delta y(t) = C(t) \cdot \delta x(t) + D(t) \cdot \delta u(t) \end{cases} \text{ in cui:}$$

$$A(t) = \frac{\partial f}{\partial x} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}}, \quad B(t) = \frac{\partial f}{\partial u} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}}, \quad C(t) = \frac{\partial \eta}{\partial x} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}}, \quad D(t) = \frac{\partial \eta}{\partial u} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}}.$$

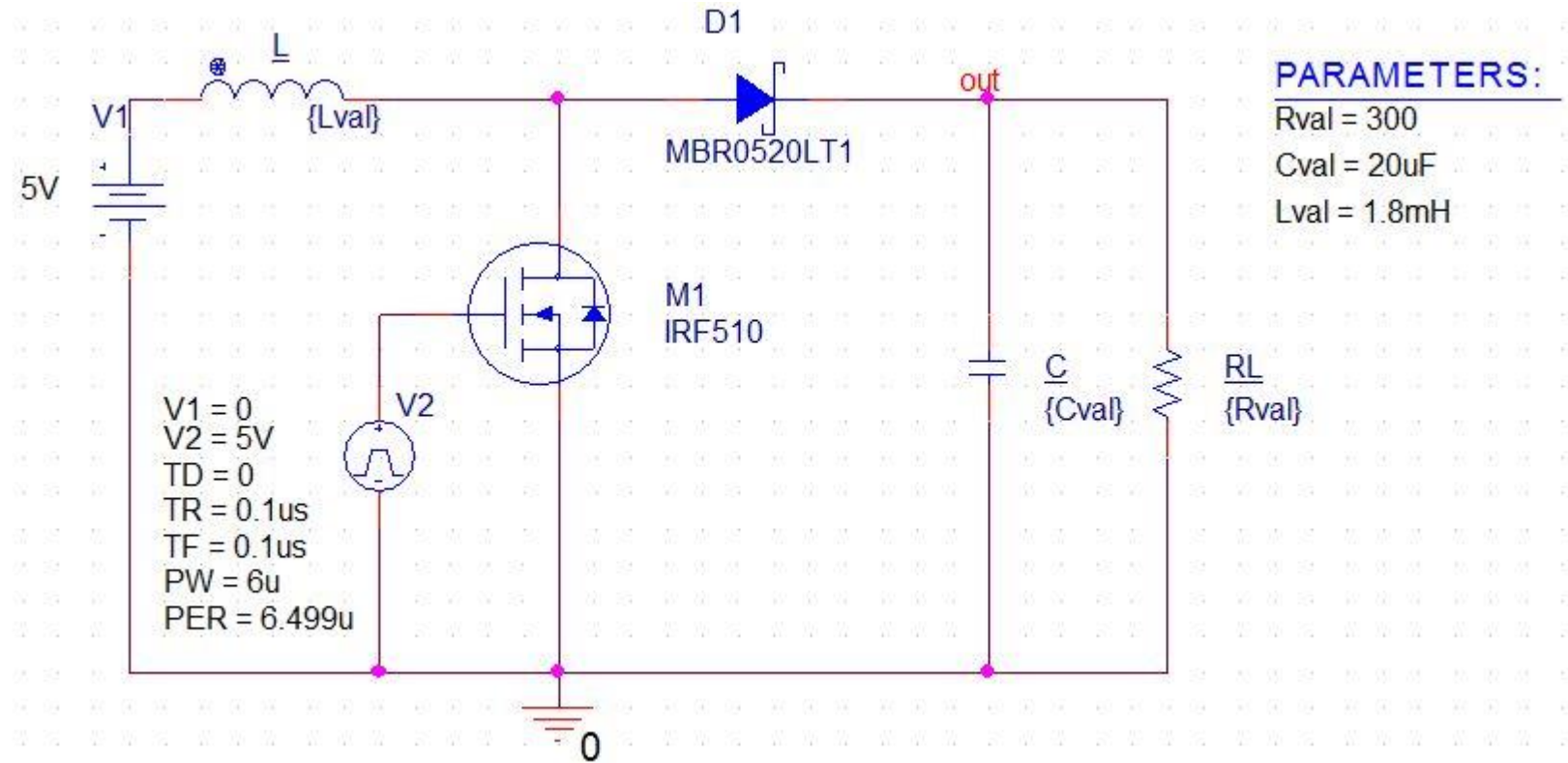
# SCELTA DELLE CONDIZIONI DI RIFERIMENTO

$$I_{L,max} = 300 \text{ mA} \rightarrow I_{L,rif} = \widehat{x}_1 = \frac{1}{2} \cdot I_{L,max} = 150 \text{ mA}$$

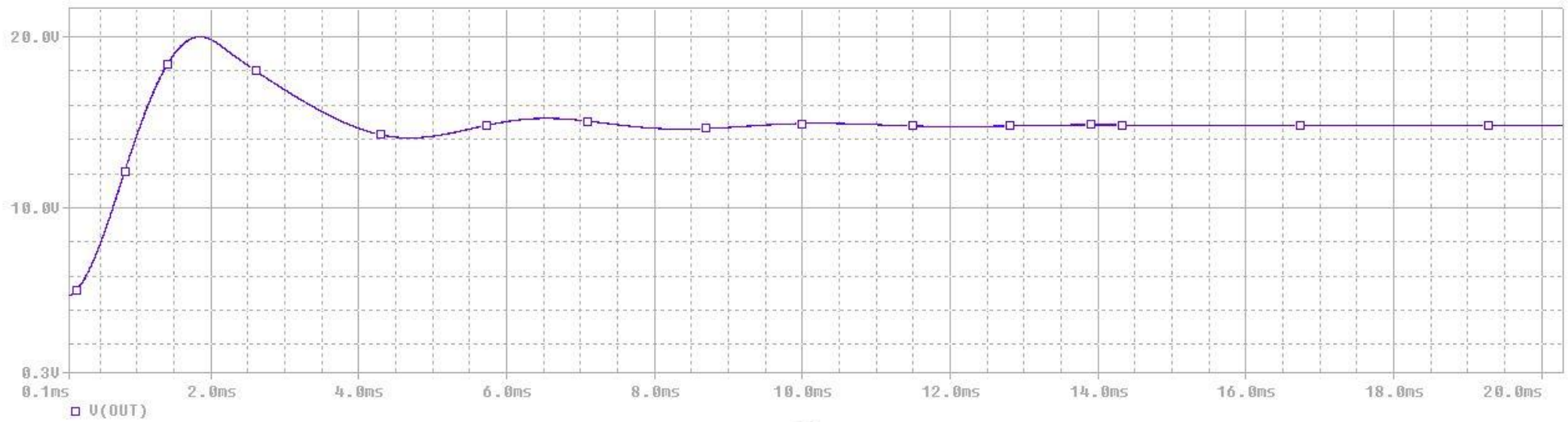
$$V_{C,rif} = \widehat{x}_2 = V_o = 15 \text{ V}$$

$$R_{rif} = \frac{V_o}{I_{L,rif} \cdot (1 - D)} = 300 \ \Omega$$

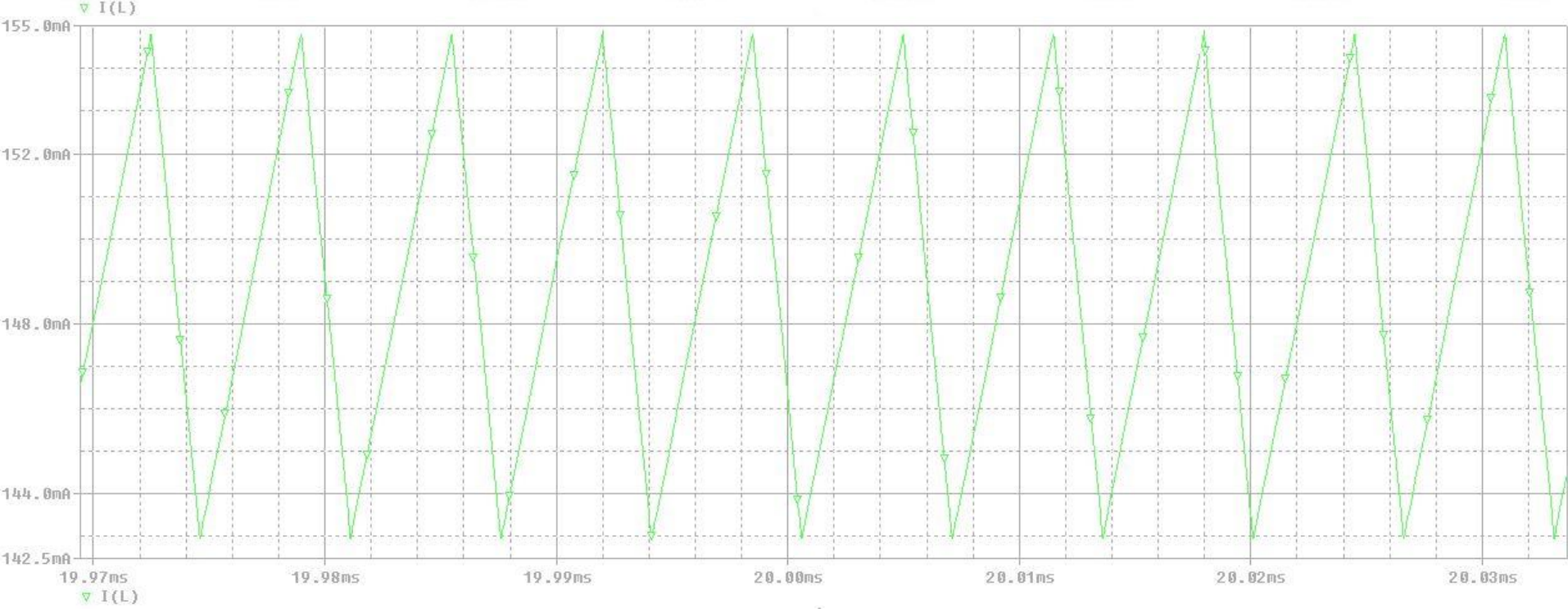
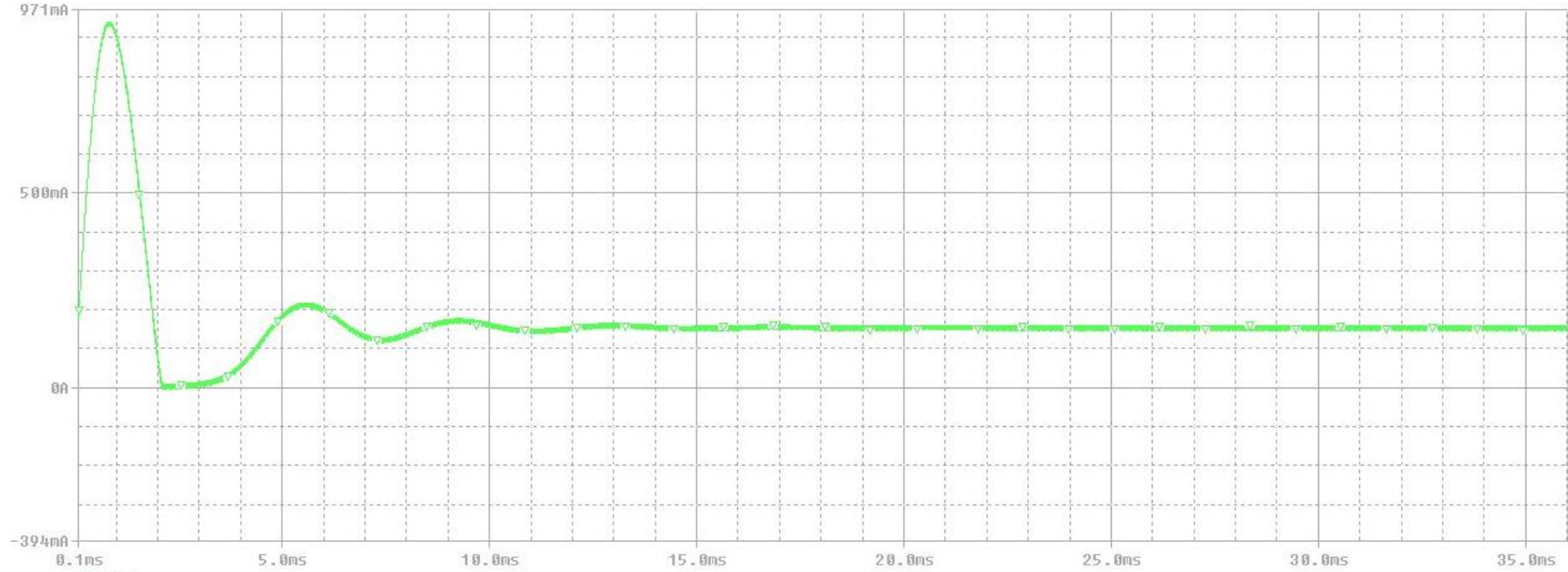
# VERIFICHE P-SPICE DELLE CONDIZIONI DI RIFERIMENTO



**VERIFICHE DELLA  $V_c$**



**VERIFICHE DELLA  $I_L$**



Nel nostro caso, sapendo che  $f = (I_L, V_C)$  ed  $\eta = V_C = y$ , dopo aver verificato le condizioni di riferimento, le quattro matrici sono infine

$$\begin{aligned}
 A(t) &= \frac{\partial f}{\partial x} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{L}(1-\hat{u}) \\ \frac{1}{C}(1-\hat{u}) & -\frac{1}{R \cdot C}\hat{u} - \frac{1}{R \cdot C}(1-\hat{u}) \end{bmatrix} = \\
 &= \begin{bmatrix} 0 & -\frac{1}{L}(1-\hat{u}) \\ \frac{1}{C}(1-\hat{u}) & -\frac{1}{R \cdot C} \end{bmatrix},
 \end{aligned}$$

$$\begin{aligned}
 B(t) &= \frac{\partial f}{\partial u} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}} = \begin{bmatrix} \frac{\partial f_1}{\partial u} \\ \frac{\partial f_2}{\partial u} \end{bmatrix} = \begin{bmatrix} \frac{V_{in}}{L} - \frac{V_{in}-x_2}{L} \\ -\frac{x_2}{R \cdot C} - \left( \frac{x_1}{C} - \frac{x_2}{R \cdot C} \right) \end{bmatrix} = \begin{bmatrix} \frac{x_2}{L} \\ -\frac{x_1}{C} \end{bmatrix},
 \end{aligned}$$



$$C(t) = \frac{\partial \eta}{\partial x} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}} = \begin{bmatrix} \frac{\partial V_C}{\partial x_1} & \frac{\partial V_C}{\partial x_2} \end{bmatrix} = [0 \quad 1],$$

$$D(t) = \frac{\partial \eta}{\partial u} \Big|_{\substack{x=\hat{x}(t) \\ u=\hat{u}(t)}} = \frac{\partial V_C}{\partial u} = 0.$$

Alle condizioni di riferimento, il sistema è descritto da una ISU (state space, che tiene conto delle quattro matrici); per ricavare la f.d.t. del boost, su Matlab abbiamo usato la funzione **"G=ss2tf(A,B,C,D)"**:

$$G(s) = C(sI - A)^{-1}B + D \rightarrow G(s) = \frac{-7500s + 1.389 \cdot 10^{-8}}{s^2 + 166.7s + 3.086 \cdot 10^{-6}}.$$

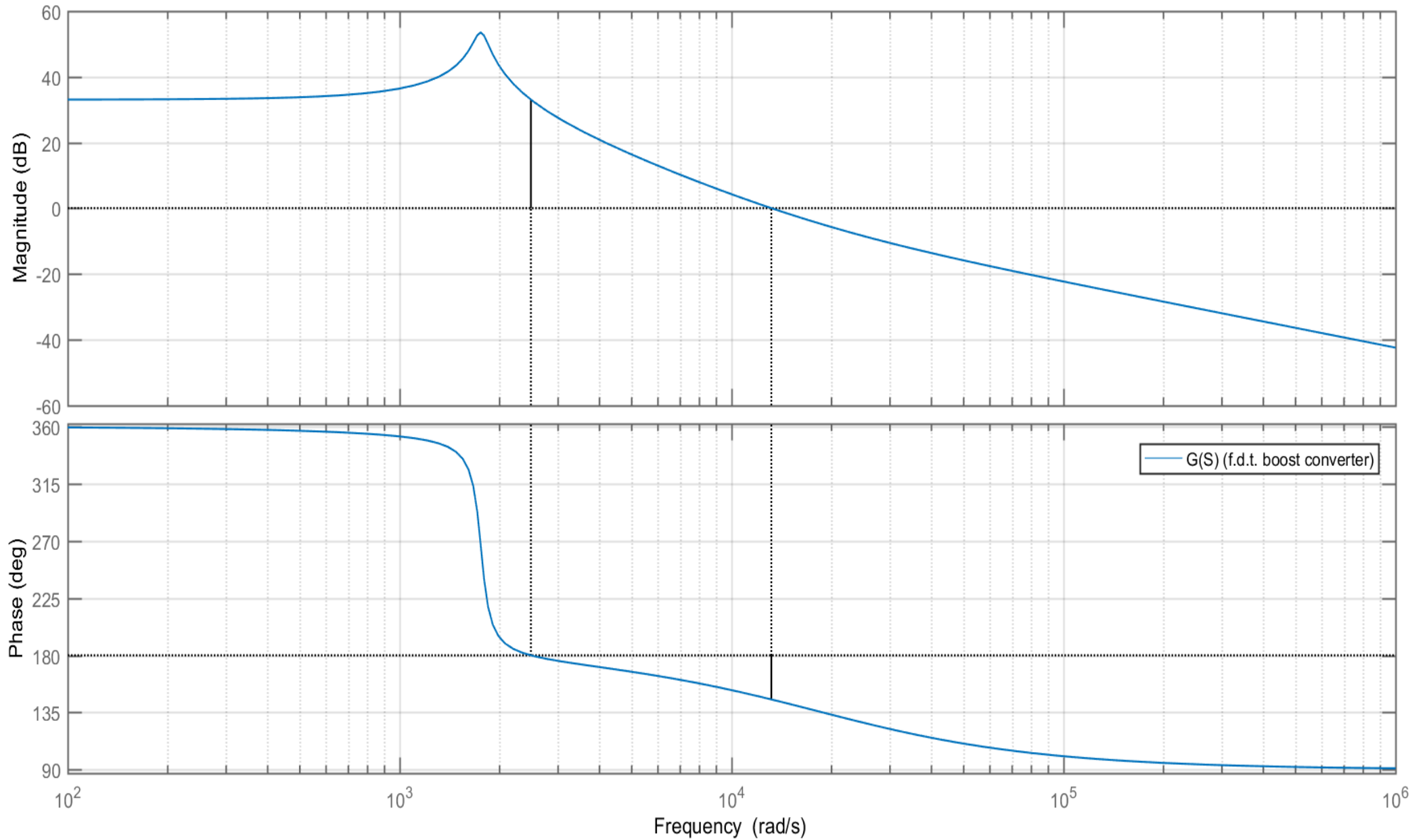
```

40 % 2 - VALUTAZIONE DELLA FDT DEL PROCESSO DA CONTROLLARE (G(S))
41
42 % 2.1 CALCOLO DELLA FDT G(S) TEMPO CONTINUO SU CUI APPLICARE IL CONTROLLO
43 %settaggio parametri fissi boost converter (no input tastiera)
44 - X1rif=0.150 %ILrif
45 - X2rif=15 %Vcrif
46 - XR=[X1rif; X2rif]
47 - L=1.8e-3
48 - C=20e-6
49 - Vin=5
50 - R=300
51 - D=0.6666667
52 %dU=0.001
53 - Dp=D*100%duty percentuale
54 - f=153850
55 % modello ISU linearizzato %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
56 - Am=[0 -(1/L)*(1-D) ; (1/C)*(1-D) -1/(R*C)]
57 - Bm=[X2rif/L ; -X1rif/C] % X1rif= stato X1 di riferimento; X2rif= stato X2 di riferimento
58 - Cm=[0 1]
59 - Dm=[0]
60 % funzione di trasferimento del modello linearizzato%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
61 - [num, den]=ss2tf(Am,Bm,Cm,Dm)
62 %calcolo funzione di trasferimento, dai coefficienti determinati con ss2tf
63 - G=tf(num, den)

```

### Bode Diagram

Gm = -33.1 dB (at 2.48e+03 rad/s) , Pm = -34.7 deg (at 1.32e+04 rad/s)



## MATRICE DI RAGGIUNGIBILITA'

$$H_R = [B \quad A \cdot B] = \begin{bmatrix} \frac{x_2}{L} & \frac{x_1}{C \cdot L} (1 - D) \\ -\frac{x_1}{C} & \frac{x_2}{C \cdot L} (1 - D) + \frac{1}{R \cdot C^2} x_1 \end{bmatrix}$$

$$\det(H_R) = \frac{x_2^2}{C \cdot L^2} (1 - D) + \frac{x_1 \cdot x_2}{RLC^2} + \frac{x_1^2}{C^2 \cdot L} (1 - D)$$

## MATRICE DI OSSERVABILITA'

$$H_O = \begin{bmatrix} H \\ H \cdot F \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{(D - 1)}{C} & -\frac{1}{C \cdot R} \end{bmatrix}$$

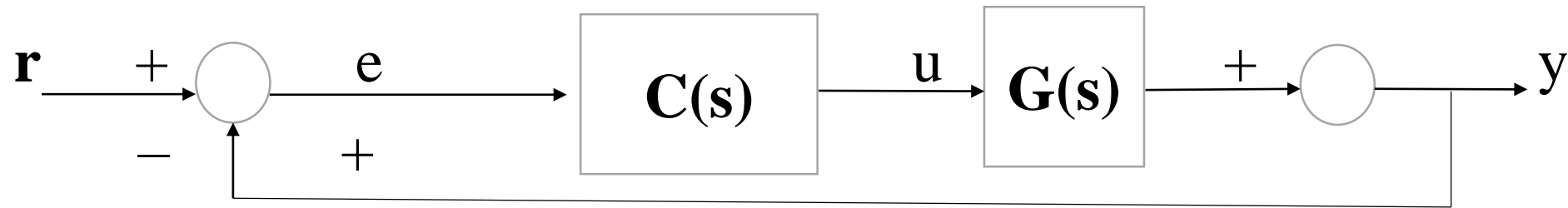
$$\det(H_O) = \frac{D - 1}{C}$$

### **3. SINTESI E VERIFICHE DEL CONTROLLO**

# CONTROLLO DIGITALE SUL BOOST CONVERTER

- Lo scopo è ottenere una tensione d'uscita che sia “indipendente” dal valore di resistenza assunto dal carico (il boost, senza controllo digitale, presenta una tensione d'uscita il cui valore è influenzato dal carico collegato a valle)
- Il circuito di controllo calcola l'errore sulla tensione d'uscita e tramite l'algoritmo di controllo progettato ed eseguito dal microcontrollore, regola in modo automatico il duty cycle del segnale PWM in ingresso al boost, in maniera tale da ottenere in uscita la tensione desiderata (di riferimento).

## 3.1 Loop shaping



$\mathbf{C}(s)$  è la f.d.t. del controllore da progettare e  $\mathbf{G}(s)$  è la f.d.t. nota del processo.  $\mathbf{C}(s)$  è il regolatore che confronta  $\mathbf{r}$  con  $\mathbf{y}$ , ottenendo la differenza  $\mathbf{e}$ ; stabilendo poi la  $\mathbf{u}$  da dare alla  $\mathbf{G}(s)$  affinché esso generi un'uscita  $\mathbf{y}$  più possibile simile ad  $\mathbf{r}$ .

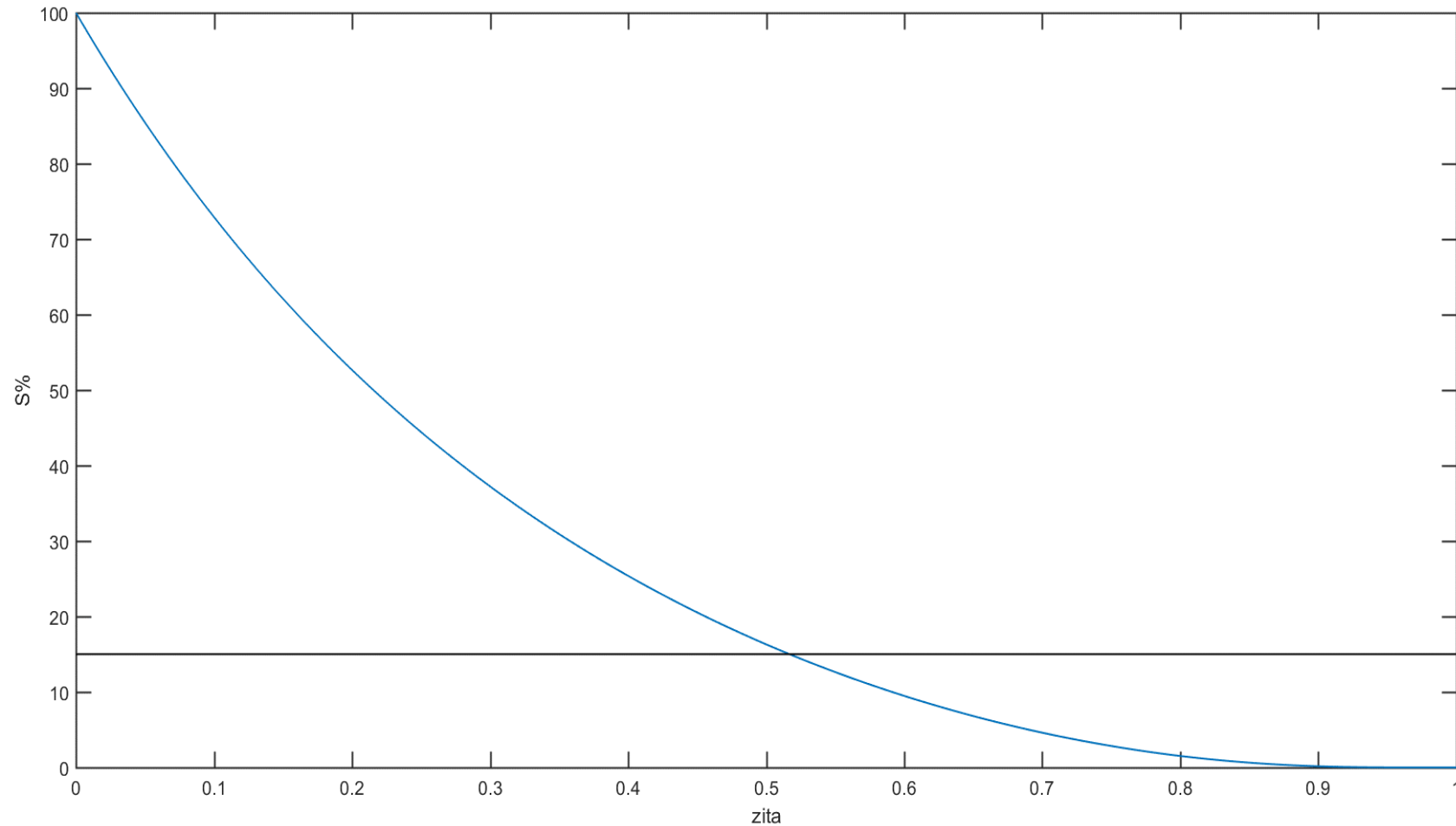


# LOOP SHAPING

- Lo scopo del loop shaping è di progettare un controllo tale che la risposta in frequenza del sistema a ciclo aperto sia quella desiderata.
- Si impongono dei vincoli sulla risposta in frequenza a ciclo aperto in base alle specifiche di progetto; ogni specifica influirà su particolari parametri della funzione d'anello.
- Le specifiche sono:
  - 1)  $S\% = 15\%$
  - 2)  $t_a = 40 \text{ ms}$  al 10%
  - 3) Errore a regime nulloche si traducono nei vincoli:
  - $S\% \rightarrow$  vincolo su  $\xi \rightarrow$  vincolo su margine di fase  $\varphi_M$
  - $t_a \rightarrow$  vincolo su banda passante  $\omega_C$
  - Errore a regime nullo  $\rightarrow$  vincolo su numero di poli nell'origine.

# TRADUZIONE DELLE SPECIFICHE DI PROGETTO

$$\blacktriangleright S\% = \frac{|y_{max} - y_{\infty}|}{|y_{\infty}|} = 100 \cdot e^{-\frac{\pi \cdot \xi}{\sqrt{1 - \xi^2}}} \rightarrow \xi_{min} = 0,5150.$$



$$\triangleright t_a = -\frac{\ln(0.01 \cdot \varepsilon)}{\xi_{min} \cdot \omega_C} \rightarrow \omega_C = -\frac{\ln(0.01 \cdot \varepsilon)}{\xi_{min} \cdot t_a} = 111,776 \text{ rad/s}$$

$$\triangleright \varphi_M = 2 \cdot \arcsin(\xi_{min}) = 61,9949^\circ$$

- Volendo garantire la stabilità del sistema a ciclo chiuso, anche a fronte di perturbazioni di  $G(s)$  determinate da variazioni del carico, abbiamo scelto di aggiungere ulteriori  $10^\circ$  oltre ai consueti  $10^\circ$ , quindi  $MR = 20$ :

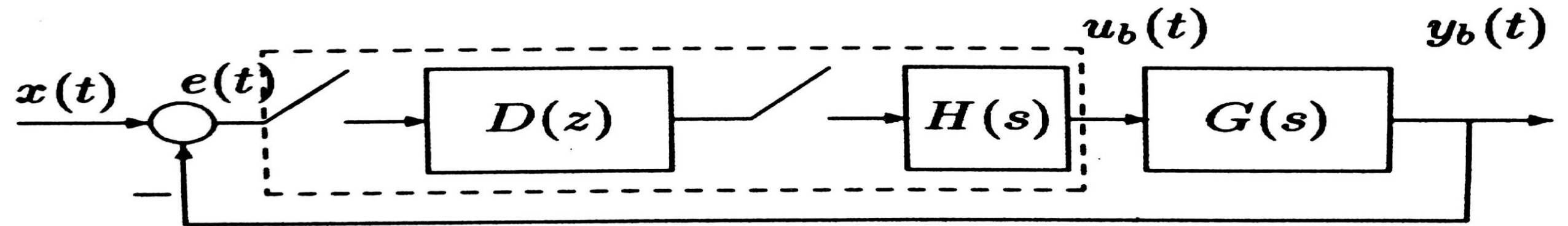
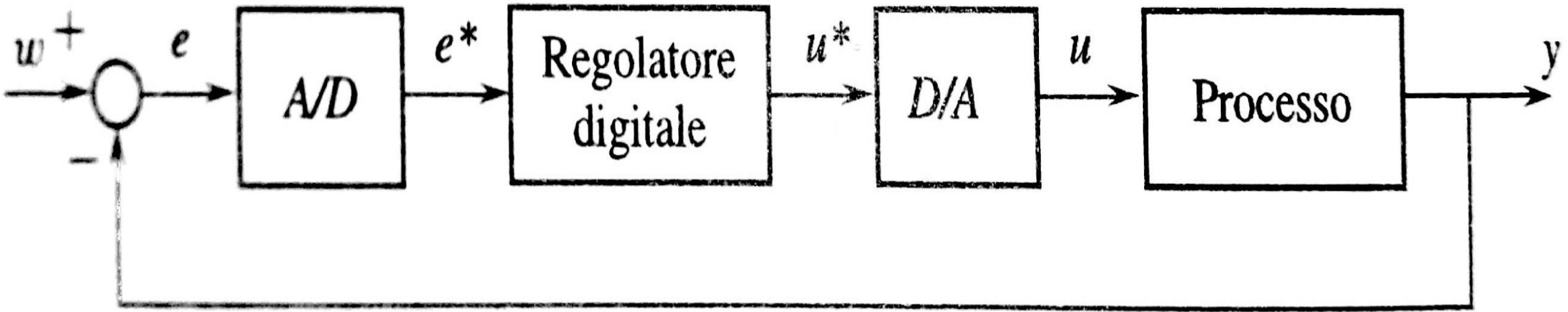
$$\varphi_M = 81,9949^\circ$$
$$\omega_C = 111,776 \text{ rad/s}$$



Da garantire sulla risposta in frequenza di  $L(s)$ , per ottenere le prestazioni in termini di  $t_a$  e  $S\%$  desiderati

```
33 %1.2 CALCOLO DELLA PULSAZIONE DI ATTRAVERSAMENTO RICHIESTA
34 -      wc=-(log(0.01*epsilonP))/(zitaMin*ta) %rad/s
35 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
36 %1.3 CALCOLO DEL MARGINE DI FASE RICHIESTO
37 -      pm=2*asind(zitaMin) %degrees
38 -      pm=pm+MR %aggiunta del margine di robustezza
```

# APPLICAZIONE DELLA F.D.T. $H(s)$



Si definiscono frequenza e pulsazione di campionamento rispettivamente:

$$f_s = \frac{1}{T_s} \quad e \quad \omega_s = 2\pi \cdot f_s = \frac{2\pi}{T_s}.$$

Affinchè il regolatore digitale lavori bene, è fondamentale che rispetti il **Teorema di Shannon**:

$$\omega_{s, min} = 2 \cdot \omega_C = 223,552 \text{ rad/s} \rightarrow f_{s, min} = 35,597 \text{ Hz.}$$

Abbiamo lavorato a una  $f_s = 500 \text{ Hz}$ , quindi un  $T_s = 2 \text{ ms}$ .

Può essere conveniente sostituire al ritardo un'approssimante costruita come una funzione razionale di variabile complessa, detta **Approssimante di Padè**. Avendo considerato in precedenza, un ricostruttore di ordine zero, con questa approssimazione, si ha che:  $H(s) = \frac{1-e^{-Ts}}{s} \approx \frac{T}{\frac{T}{2}s+1}$ . Per l'analisi della discretizzazione,

si considera  $H(s) = \frac{1}{\frac{T}{2}s+1}$ , con T che si omette poichè esso è compensato da  $\frac{1}{T}$  del campionatore.

```
79 % 2.3 APPLICAZIONE DELLA F.D.T H(S), CHE TIENE CONTO DEL CAMPIONATORE +
80 % RICOSTRUTTORE. PASSAGGIO NECESSARIO PER LA PROGETTAZIONE DEL
81 % CONTROLLO DIGITALE
82 -     fs=500
83 -     Ts= 1/fs
84 -     H=tf([1],[Ts/2 1])
```

$$\blacktriangleright \mathbf{G}(s) = \frac{-7500s + 1.389 \cdot 10^{-8}}{s^2 + 166.7s + 3.086 \cdot 10^{-6}}$$

$$\blacktriangleright \mathbf{H}(s) = \frac{1}{0.001 \cdot s + 1}$$

$$\blacktriangleright \mathbf{G}(s) \cdot \mathbf{H}(s) = \frac{-7500 \cdot s + 1.389 \cdot 10^{-8}}{0.001 \cdot s^3 + 1.167 \cdot s^2 + 3253 \cdot s + 3.086 \cdot 10^{-6}}$$



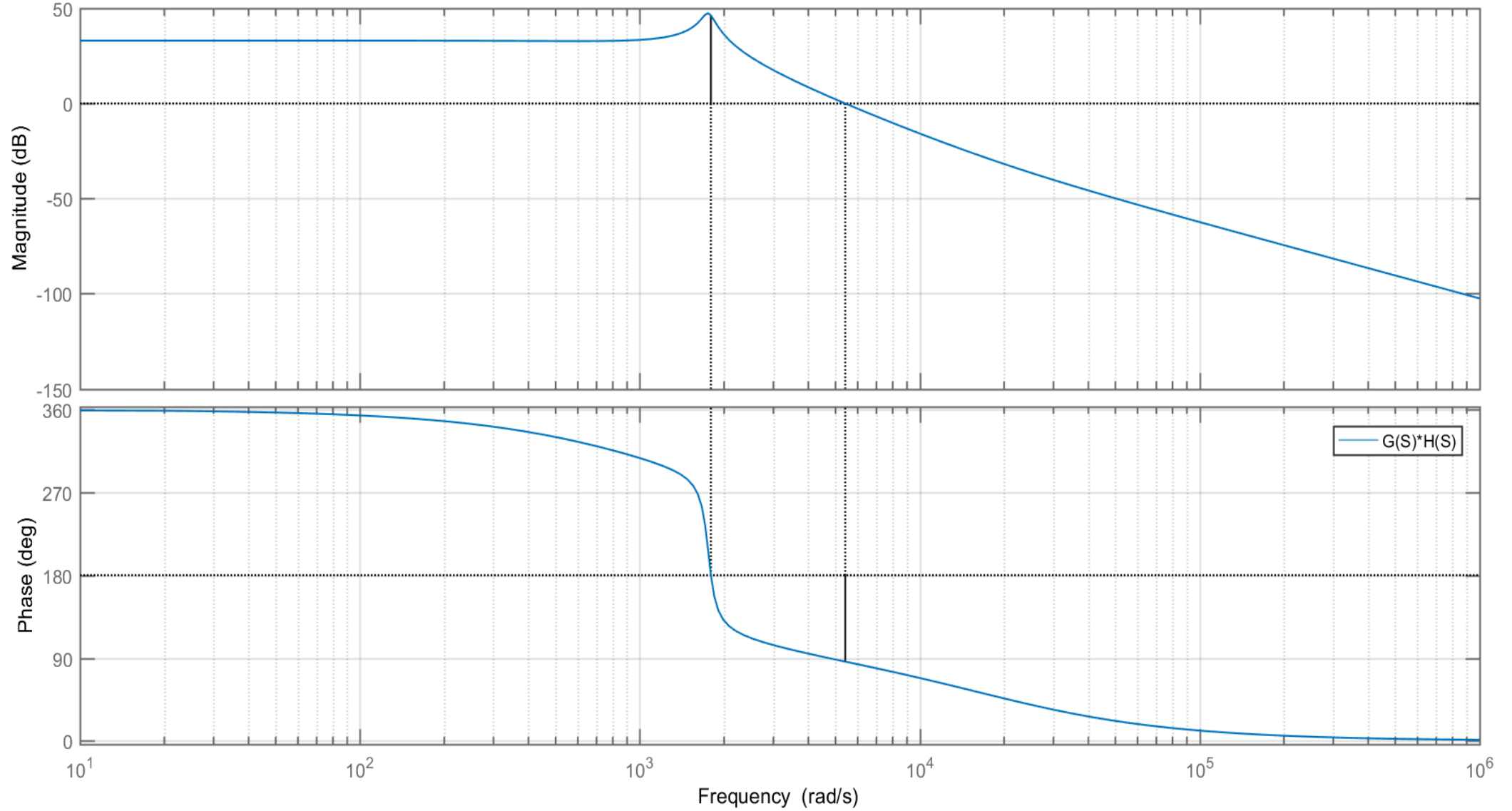
# PLOT DELLE RISPOSTE IN FREQUENZA



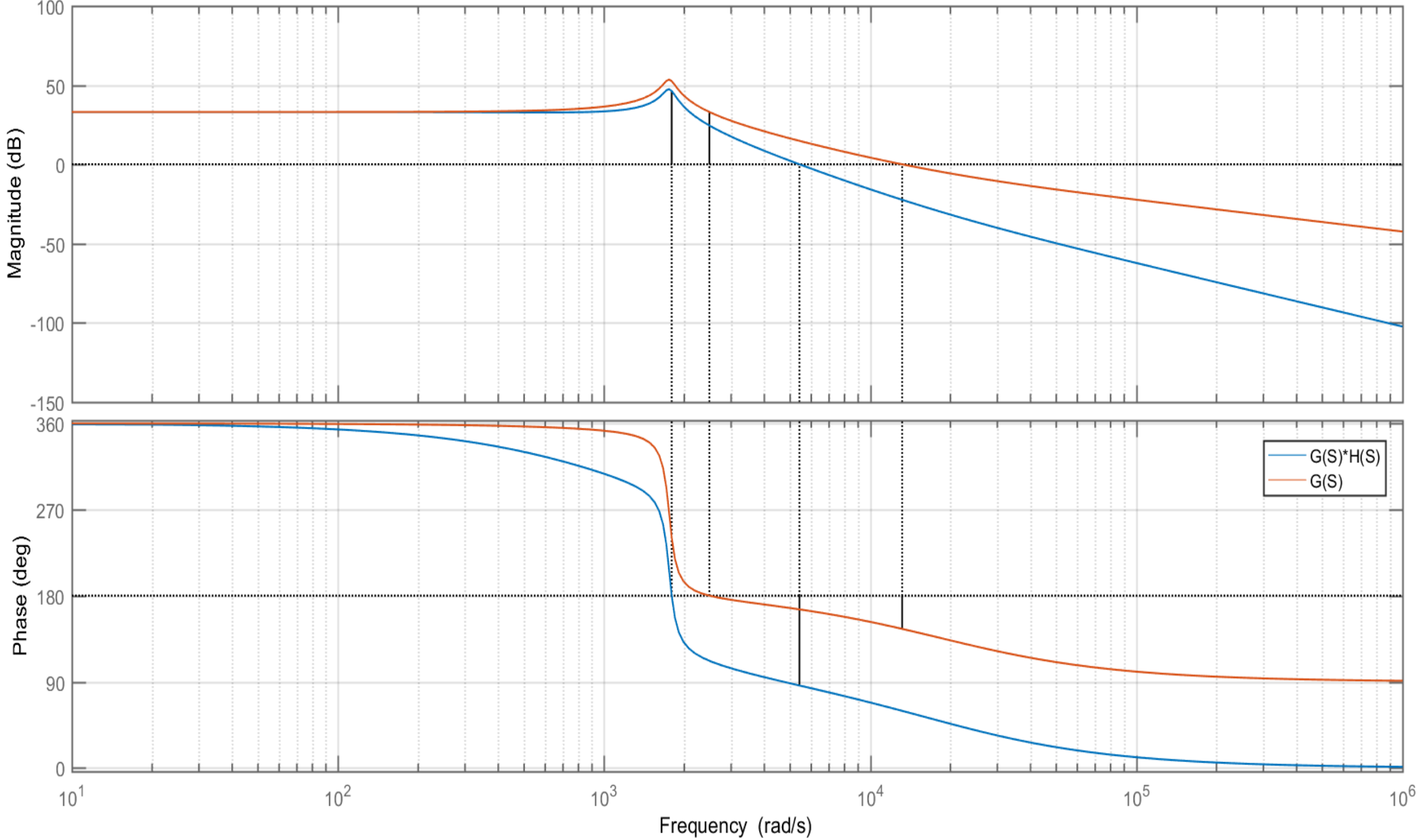
# $G(s) \cdot H(s)$

Bode Diagram

Gm = -46.4 dB (at 1.79e+03 rad/s) , Pm = -93.9 deg (at 5.42e+03 rad/s)



confronto tra  $G(S)$  e  $G(S)*H(S)$

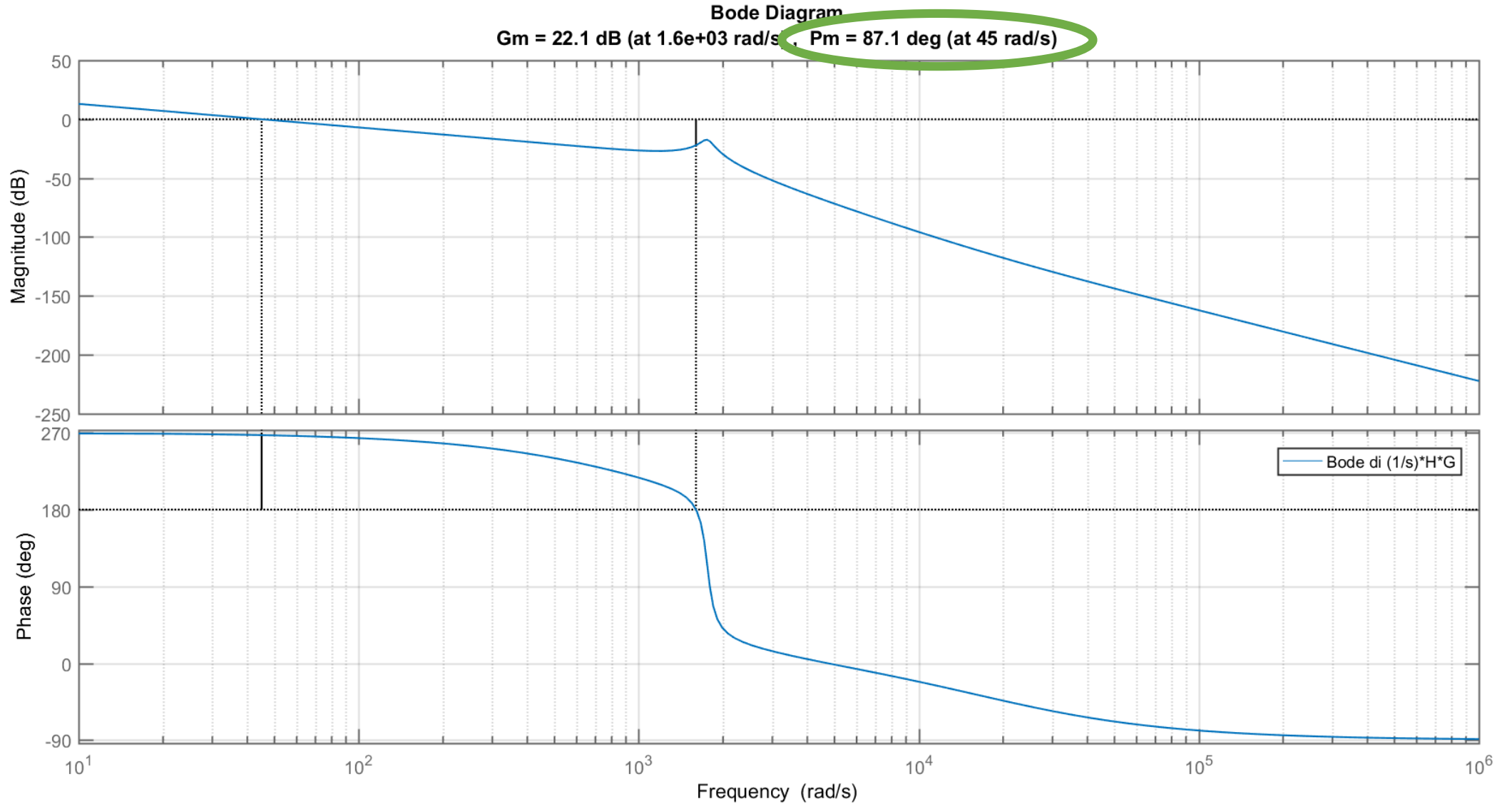


# I tentativo Loop Shaping

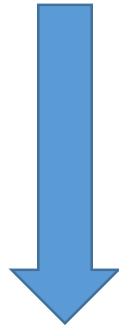
- Polo di primo ordine nell'origine



-20 dB/dec  
-90°



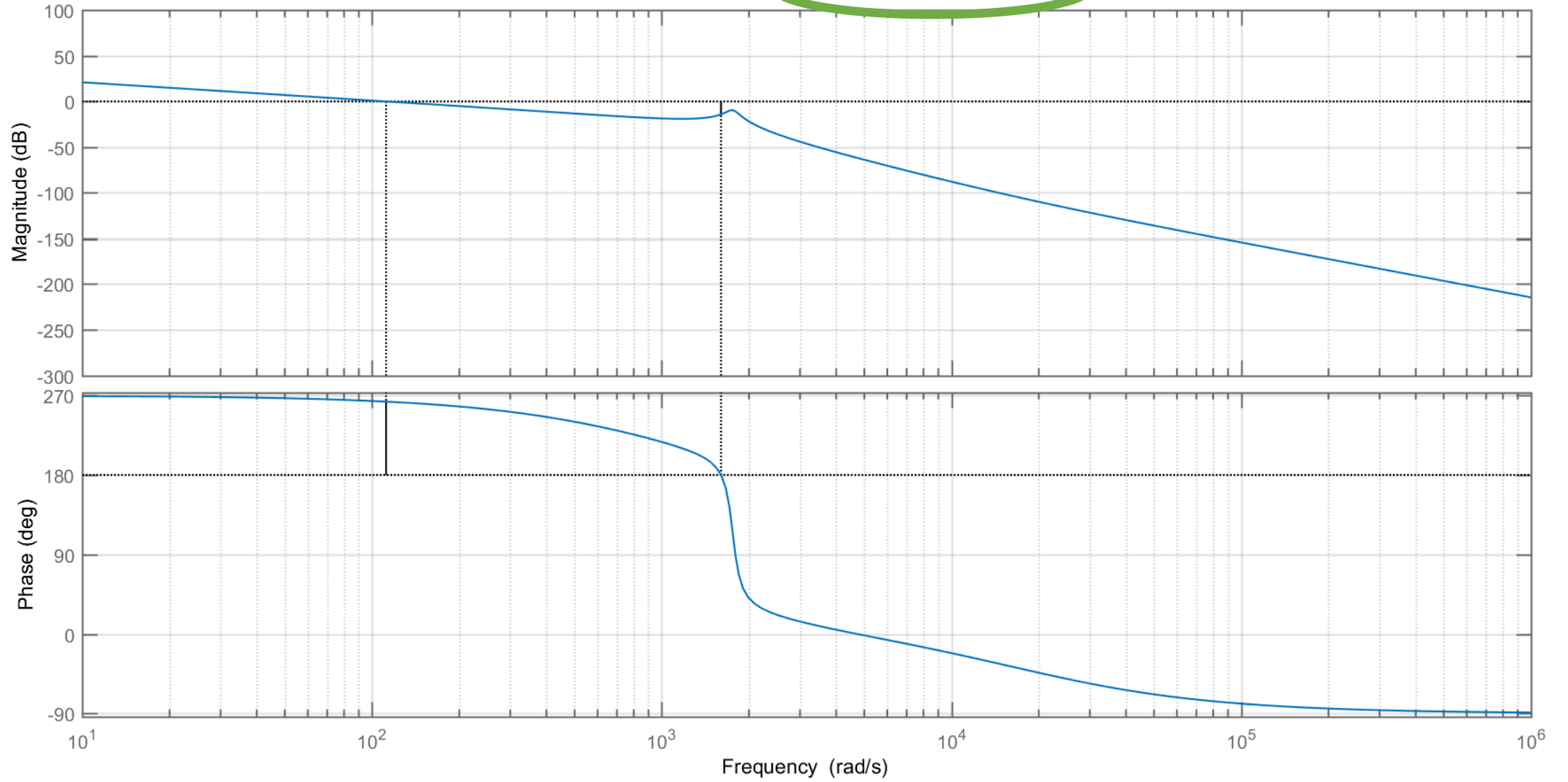
Necessità di aumentare  $\omega_c$



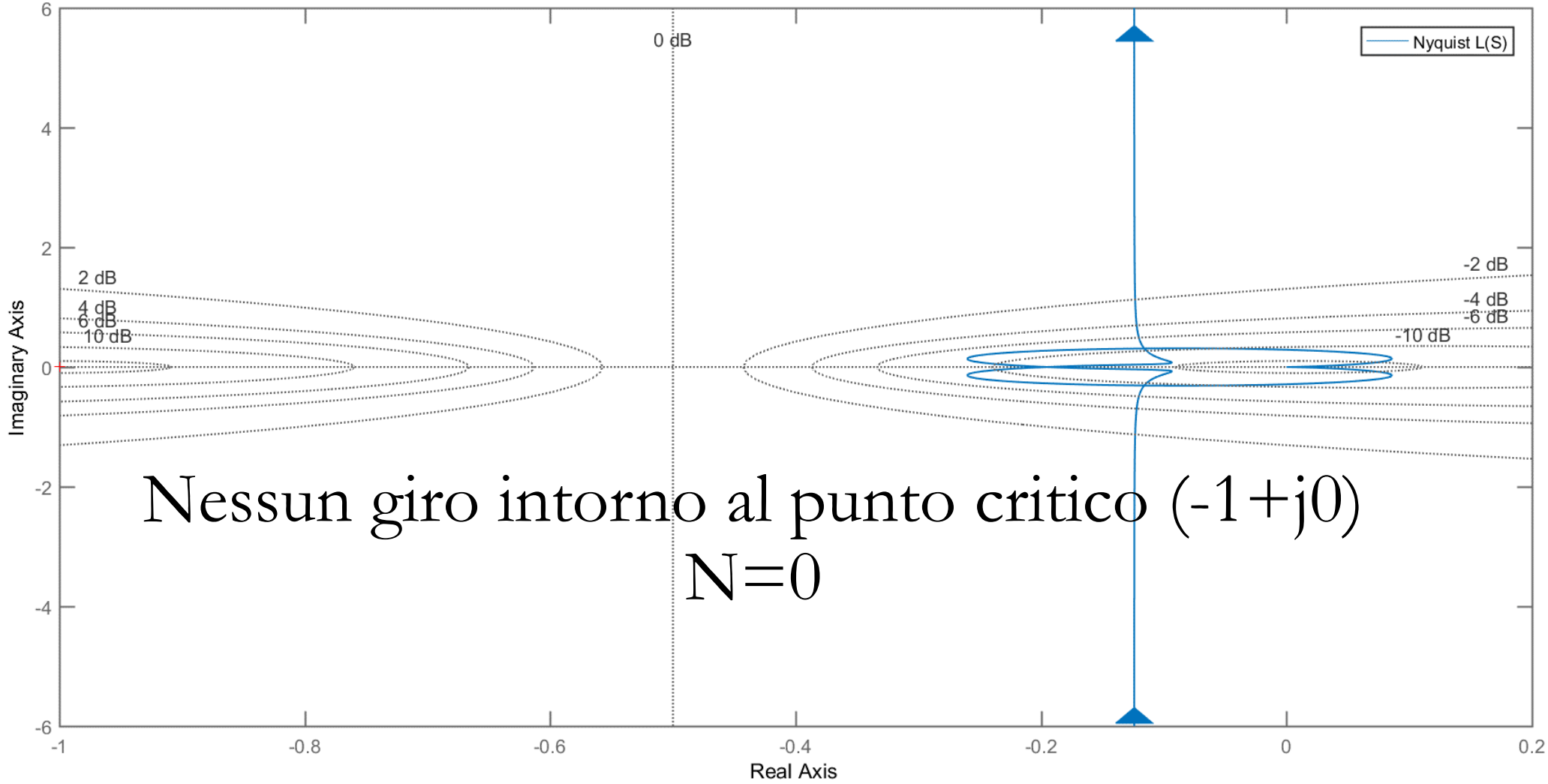
**Guadagno M**

```
98 % 3.2 GUADAGNO STATICO PER MODIFICARE WC
99 -     wcA=wc           %pulsazione di attraversamento desiderata
100 -     [m,p]=bode(G1, wcA) %calcolo di modulo e fase della G1 alla wc desiderata
101 -     M=1/m         %guadagno (che garantisce la wc voluta)
102
103
104 % 3.3 FDT AD ANELLO APERTO FINALE -> L=(C*G)
105 -     Ls=G1*M
```

**Bode Diagram**  
**Gm = 14.2 dB (at 1.6e+03 rad/s) , Pm = 82.9 deg (at 112 rad/s)**



Nyquist Diagram



Poli della L(S):

1.0e+03 \*  
0.0000 + 0.0000i  
-0.0833 + 1.7548i  
-0.0833 - 1.7548i  
-1.0000 + 0.0000i



Nessun polo a  
parte reale positiva:  
 $P=0$

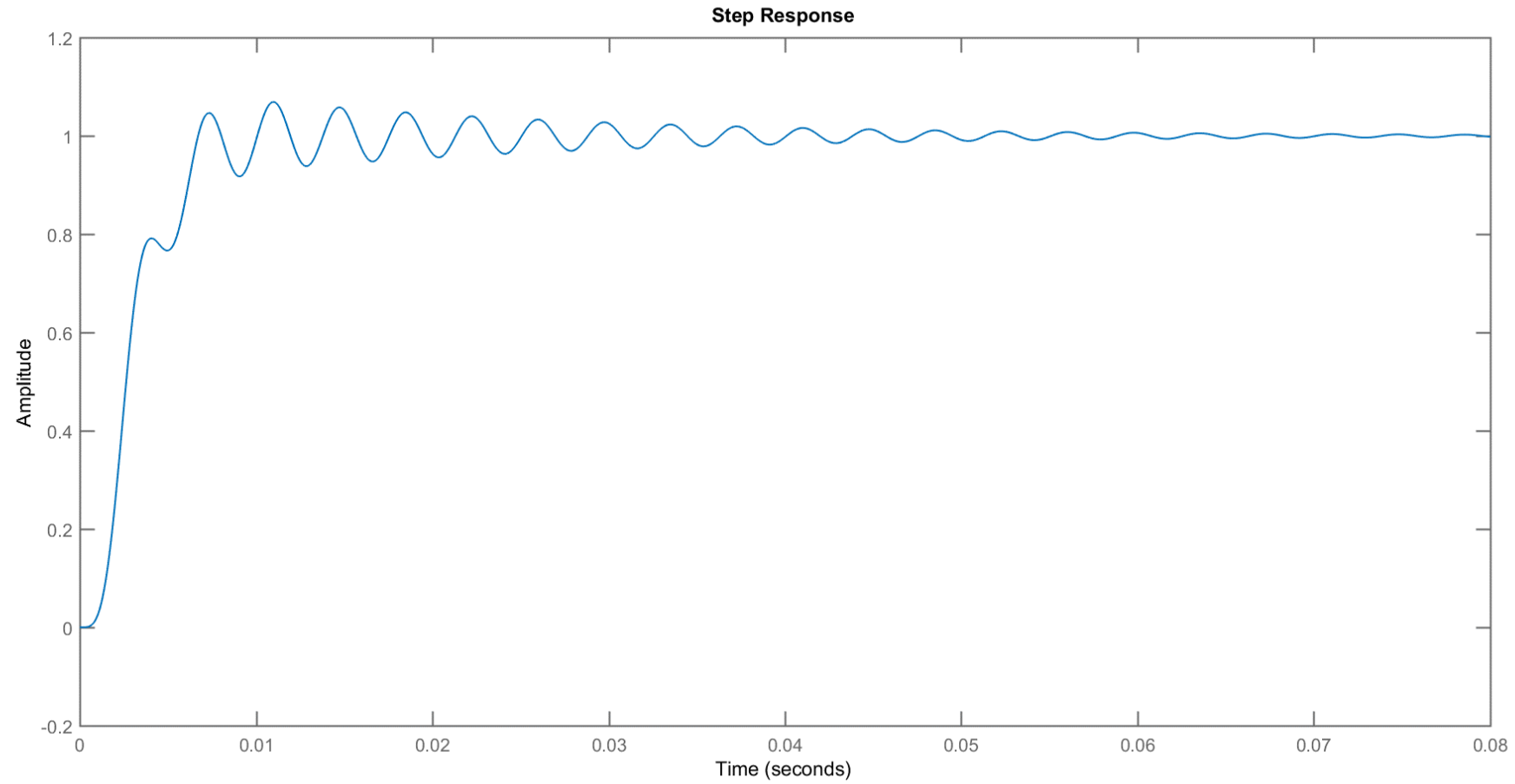
$N=P=0$

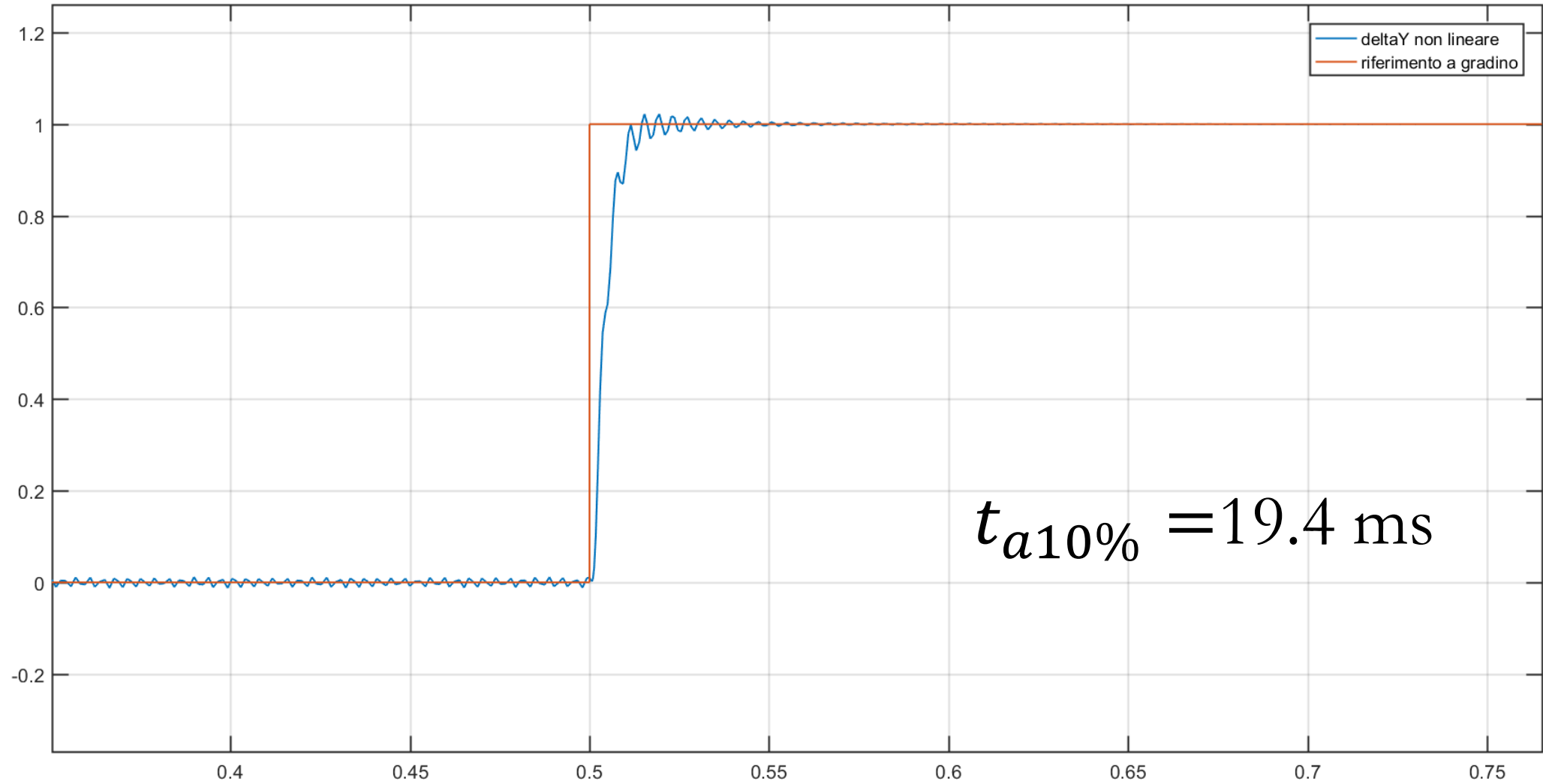


Il sistema risulta  
semplicemente stabile a ciclo  
chiuso, per il criterio di  
Nyquist

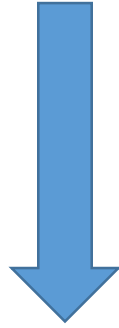


# Risposta I tentativo Loop Shaping





Risposta al gradino non soddisfacente



## **II tentativo Loop Shaping**

- Polo di secondo ordine nell'origine

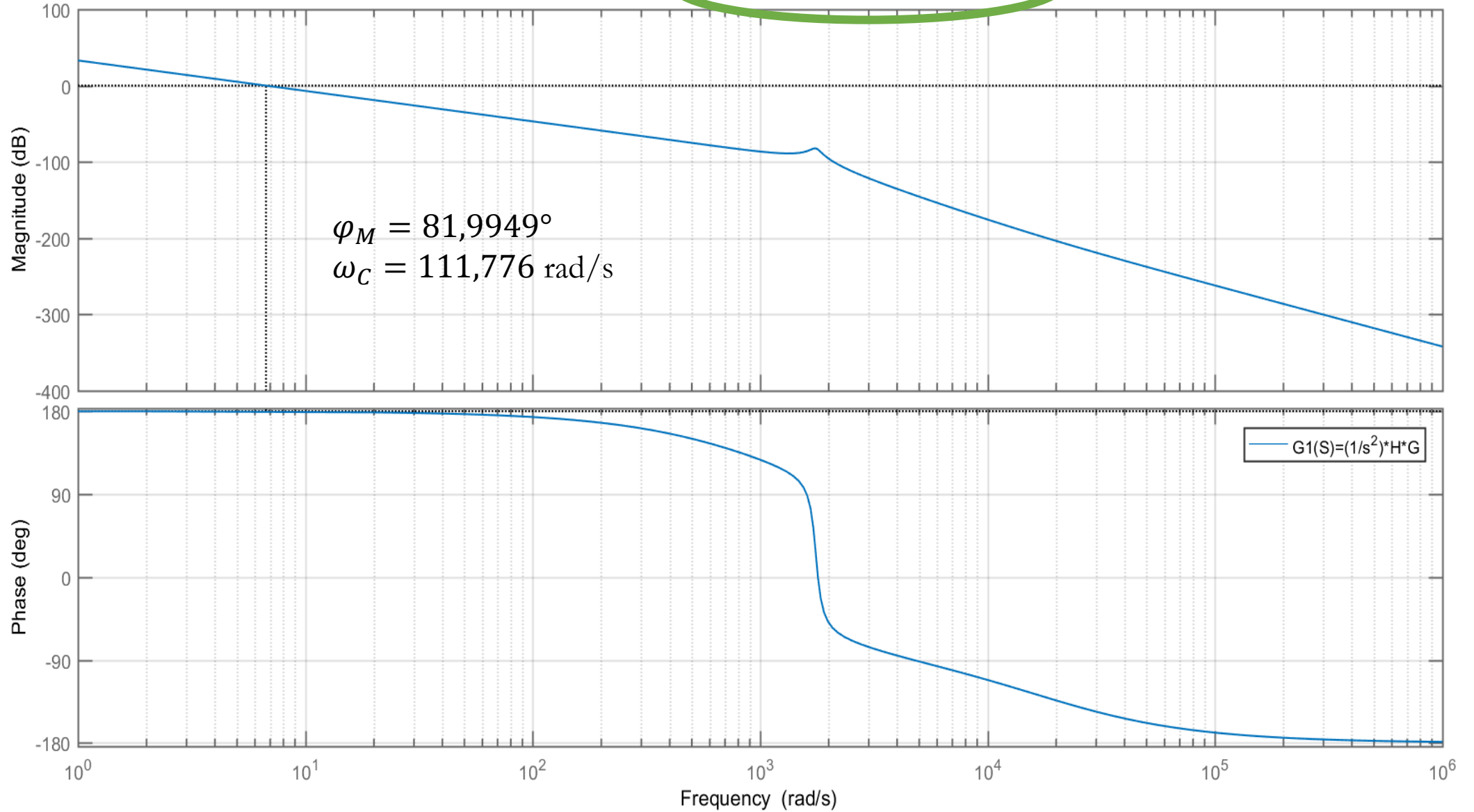
- Polo di secondo ordine nell'origine  -40 dB/dec  
-180°

$$G_1(s) = \frac{1}{s^2} \cdot G(s) \cdot H(s) =$$

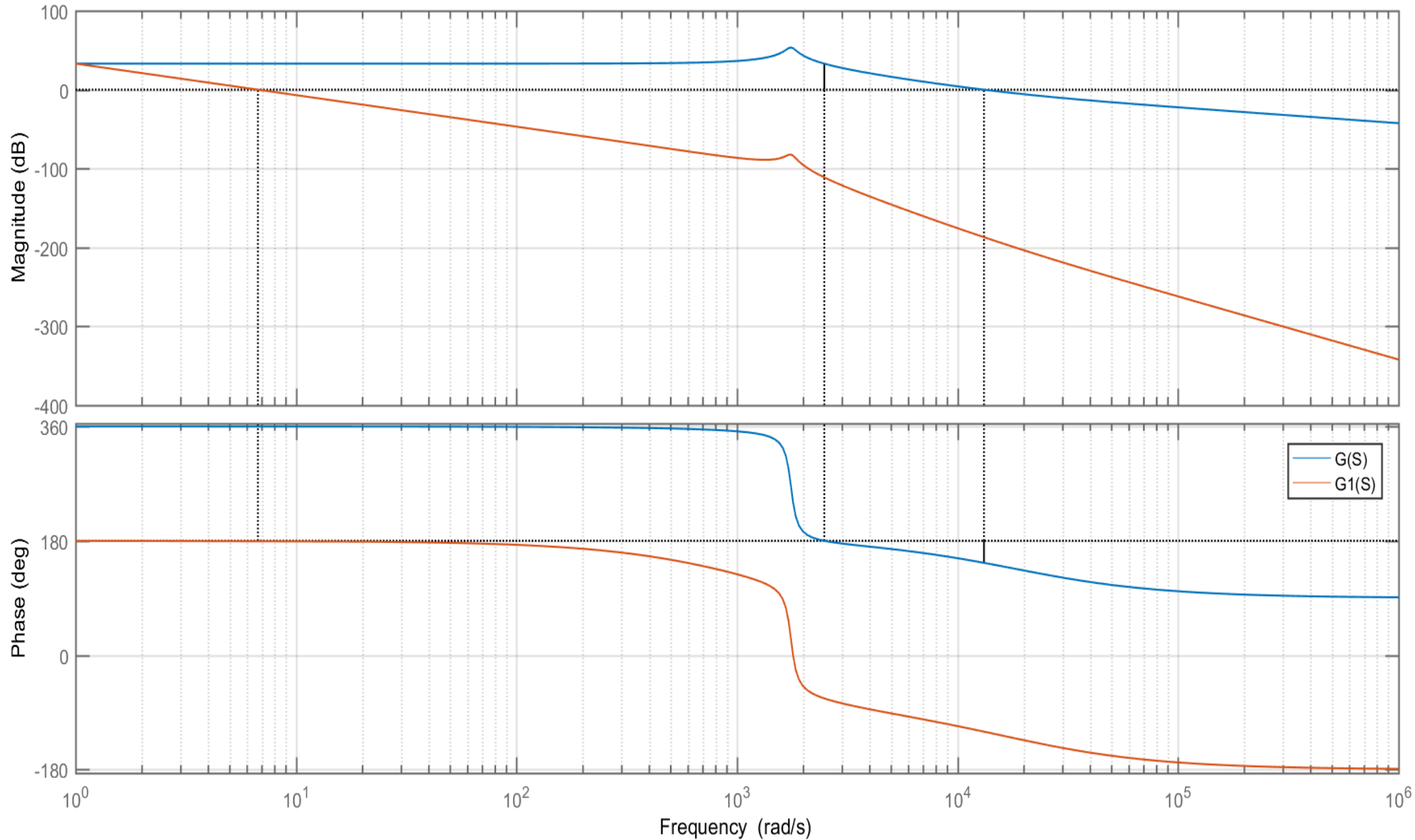
$$= \frac{-7500 \cdot s + 1.389 \cdot 10^{-8}}{0.001 \cdot s^5 + 1.167 \cdot s^4 + 3253 \cdot s^3 + 3.086 \cdot 10^{-6} \cdot s^2}$$

Bode Diagram

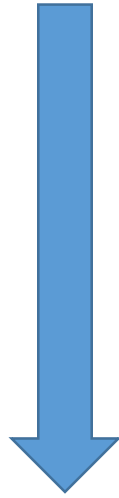
Gm = -Inf dB (at 0 rad/s) , Pm = -0.426 deg (at 6.71 rad/s)



Confronto  $G(S)$  e  $G1(S)=(1/s^2 \cdot H \cdot G)$



Necessità di aumentare  $\omega_C$  e  $\varphi_M$



**Rete anticipatrice**

# ***PROGETTO DELLA RETE ANTICIPATRICE***

$$C_{lead}(s) = \frac{1 + s \cdot T}{1 + s \cdot \tau}$$

- $m = 0.0036$  in  $\omega_C = 111.776$  rad/s  $\longrightarrow M = \frac{1}{m} = 278.2397$
- $\varphi_M = 81,9949^\circ$   $\longrightarrow \phi = 81,9949^\circ$
- Calcolo dei parametri  $T$  e  $\tau$  è:

$$\frac{1 + j \cdot \omega_C \cdot T}{1 + j \cdot \omega_C \cdot \tau} = M \cdot e^{j\phi} \rightarrow \begin{cases} \tau = \frac{M \cdot \cos\phi - 1}{M \cdot \omega_C \cdot \sin\phi} \\ T = \frac{M - \cos\phi}{\omega_C \cdot \sin\phi} \end{cases}$$



```

% 3.2 RETE ANTICIPATRICE (PER GUADAGNARE FASE E AUMENTARE LA PULSAZIONE DI ATTRAVERSAMENTO
wcA=wc          %pulsazione di attraversamento desiderata
[m,p]=bode(G1, wcA) %calcolo di modulo e fase della G1 alla wc desiderata
M=1/m          %guadagno della rete anticipatrice (che garantisce la wc voluta)
phiA=pm        %margine di fase desiderato
%CALCOLO PARAMETRI DELLA RETE
T=(M-cosd(phiA))/(wcA*sind(phiA))
tau=(M*cosd(phiA)-1)/(M*wcA*sind(phiA))
%costruzione della rete
Clead=tf([T 1],[tau 1])

```

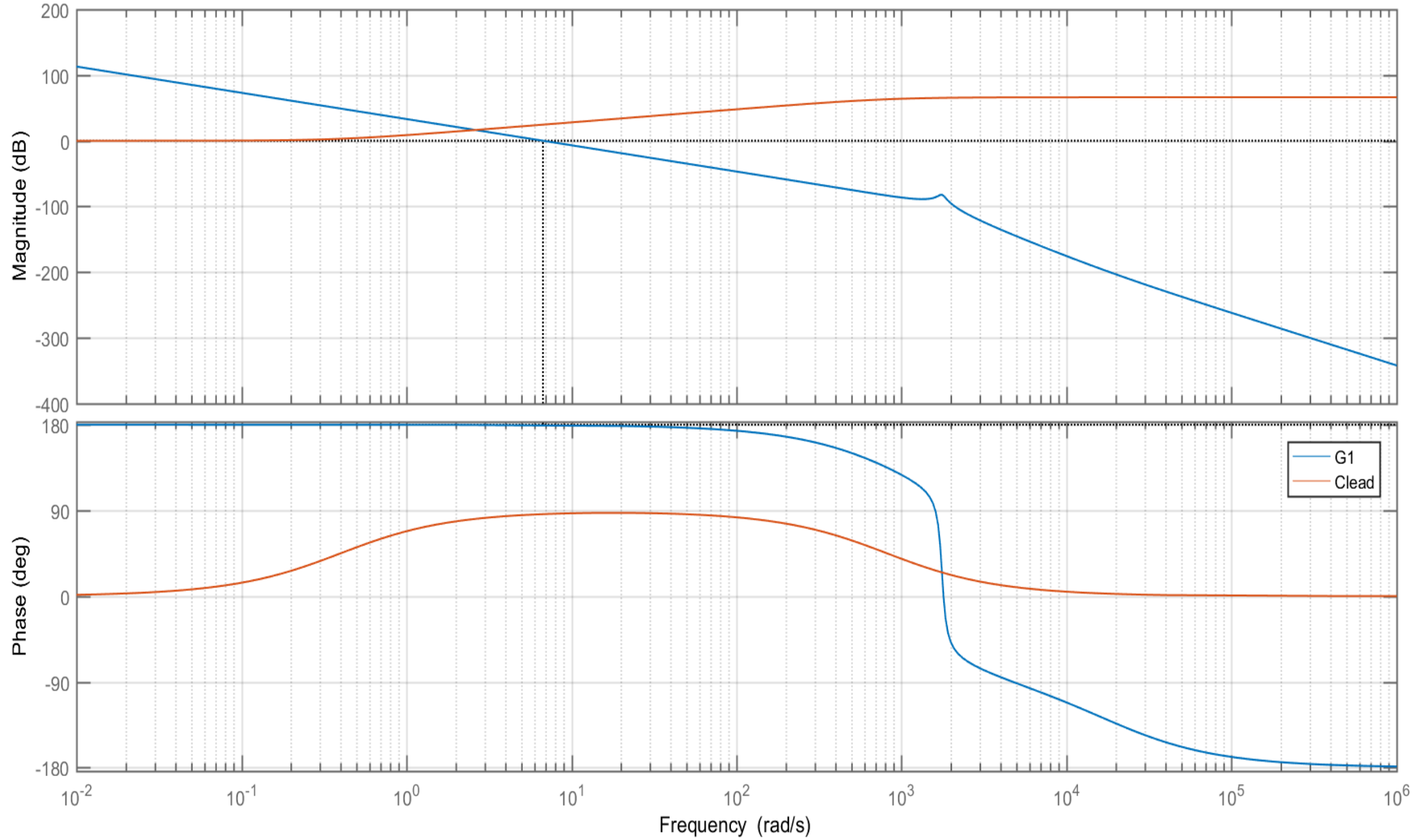
Otteniamo:

1.  $T = 2.5125 \text{ s}$
2.  $\tau = 0.0012 \text{ s}$

$$C_{lead}(s) = \frac{1+2.5125 \cdot s}{1+0.0012 \cdot s},$$

con  $T$  e  $\tau$  reali positivi (quindi la rete è fisicamente realizzabile). In seguito, si vedranno gli effetti della rete anticipatrice che andranno a sommarsi alla relazione della  $G_1(s)$ .

### Applicazione rete anticipatrice



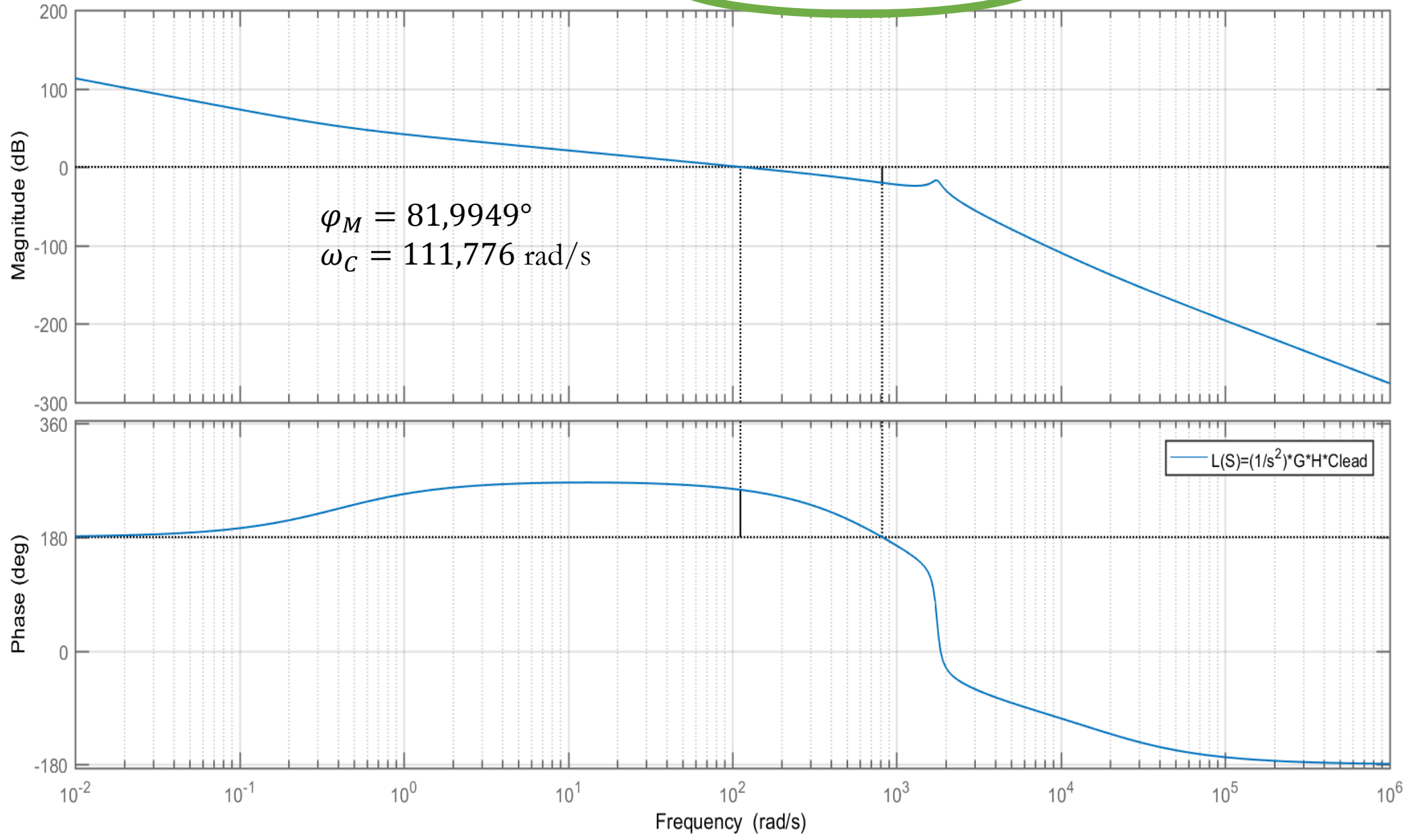
➤ Calcolo della f.d.t.  $L(s)$  ad anello aperto:

$$L(s) = \frac{1}{s^2} \cdot C_{lead}(s) \cdot G(s) \cdot H(s) =$$

$$= \frac{-1.884 \cdot 10^{-4} \cdot s^2 + 3.49 \cdot 10^{-8} \cdot s + 1.389 \cdot 10^{-8}}{1.226 \cdot 10^{-6} \cdot s^6 + 0.00243 \cdot s^5 + 5.154 \cdot s^4 + 7036 \cdot s^3 + 3.086 \cdot 10^{-6} \cdot s^2} \cdot$$

Bode Diagram

Gm = 20.3 dB (at 816 rad/s) , Pm = 74.9 deg (at 112 rad/s)



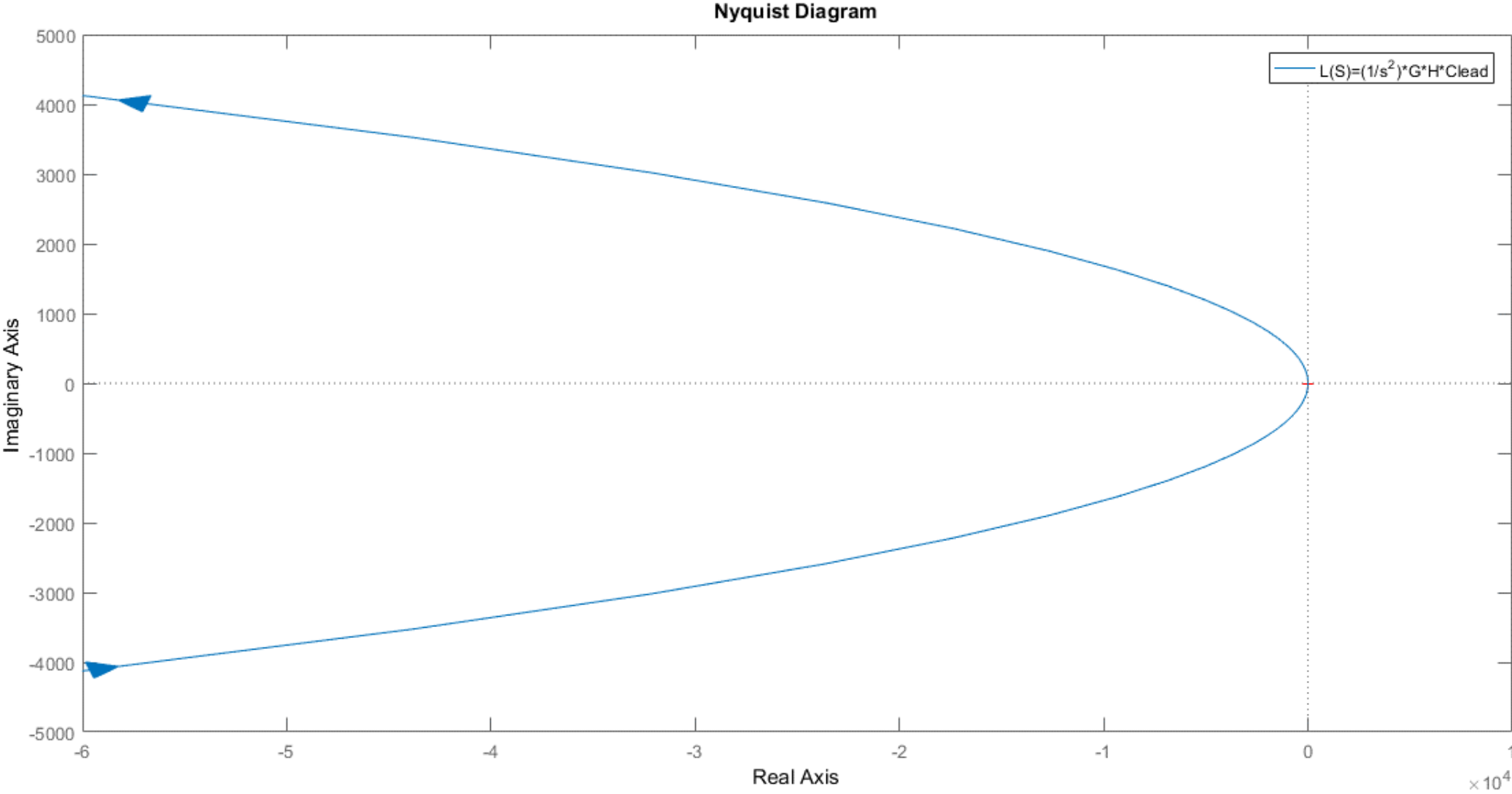
# Calcolo della f.d.t del controllo

$$C_{fin}(s) = \frac{1}{s^2} \cdot C_{lead}(s) = \frac{1+2.512 \cdot s}{0.001226 \cdot s^3 + s^2},$$

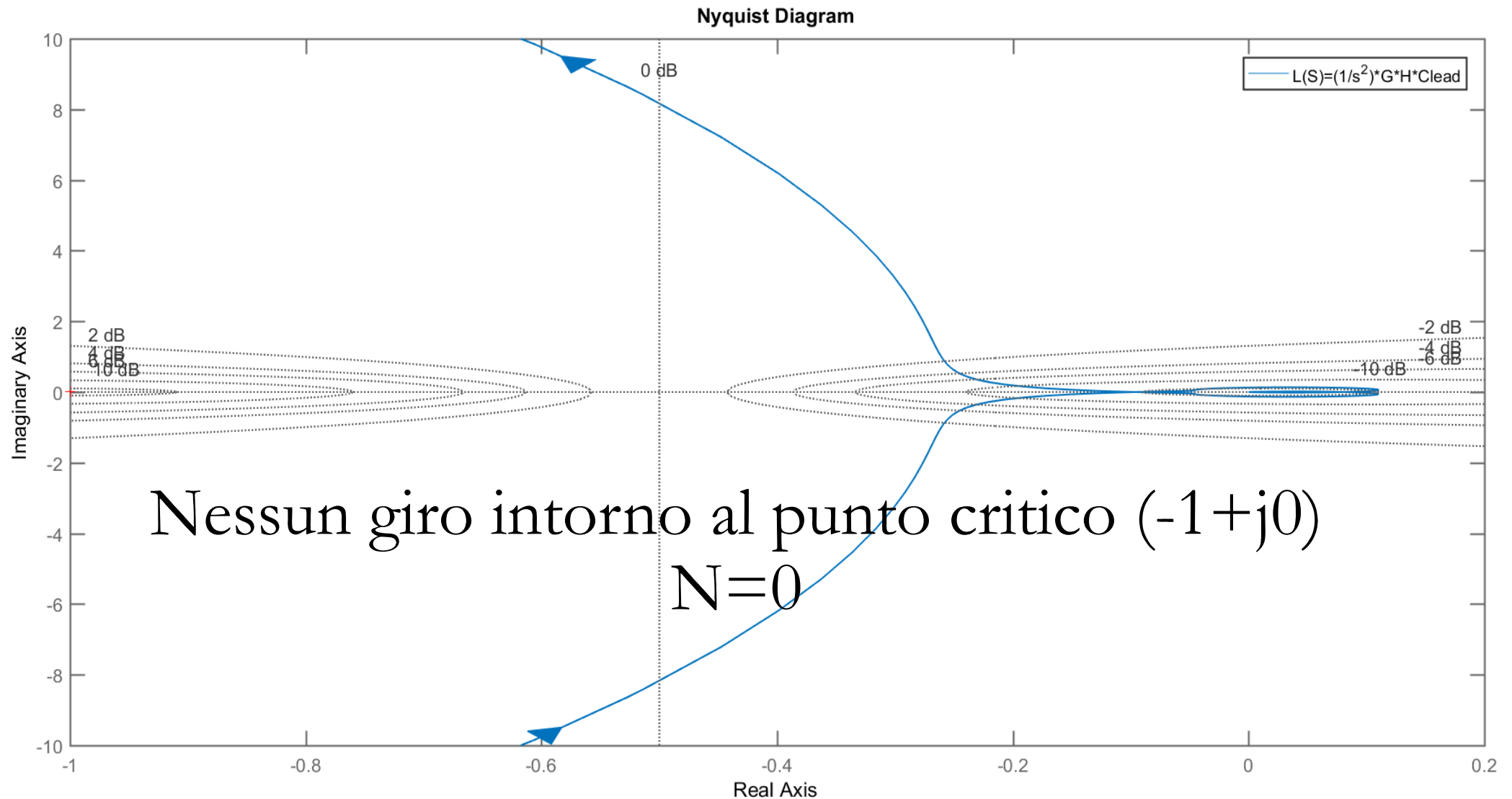
che ingloba i poli nell'origine e la rete anticipatrice.

## **3.2 Verifiche del controllo tempo continuo**

# STABILITA' – DIAGRAMMA DI NYQUIST DELLA $L(s)$



# ZOOM DEL DIAGRAMMA DI NYQUIST DELLA $L(s)$





Poli della L(s):

1.0e+03 \*

0.000000000000000000 + 0.000000000000000000i

0.000000000000000000 + 0.000000000000000000i

-0.083333333333333333 + 1.754843209907384i

-0.083333333333333333 - 1.754843209907384i

-1.000000000000000000 + 0.000000000000000000i

-0.815870891786337 + 0.000000000000000000i



Nessun polo a  
parte reale positiva:  
P=0

N=P=0



Il sistema risulta  
semplicemente stabile a ciclo  
chiuso, per il criterio di  
Nyquist

# TEST DELLE PRESTAZIONI

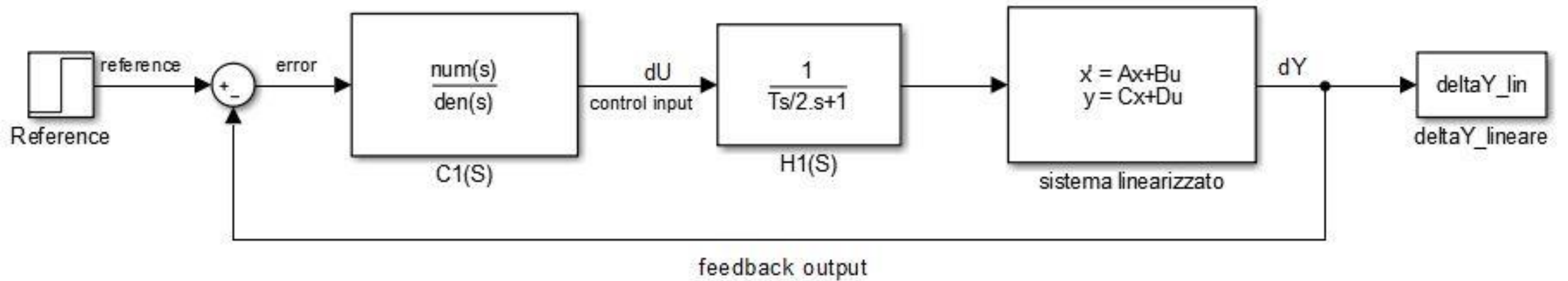
## ➤ Risposta al riferimento di tipo gradino unitario:

1. Su sistema linearizzato, con controllo applicato
2. Su sistema non lineare, con controllo applicato

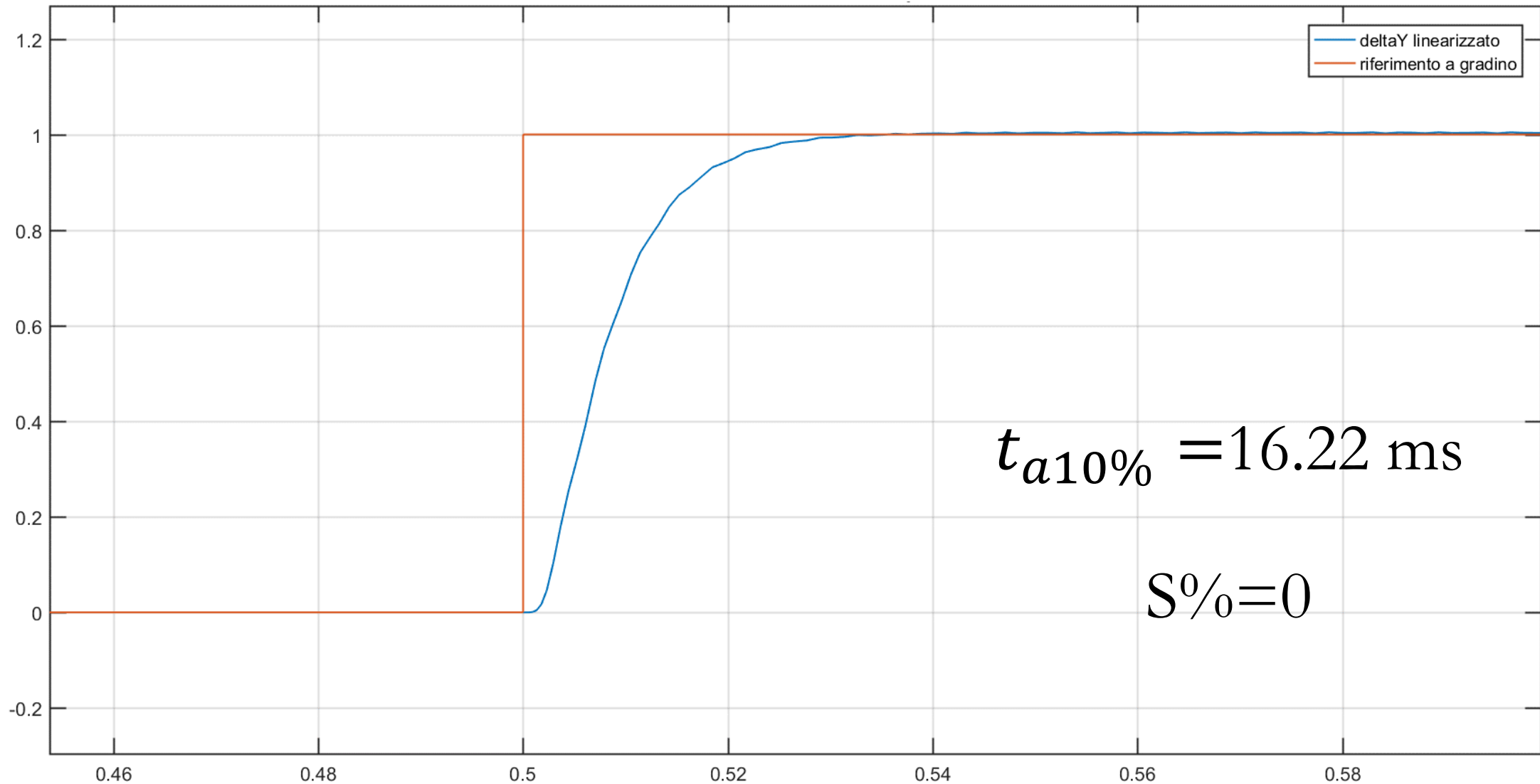


Modelli Simulink

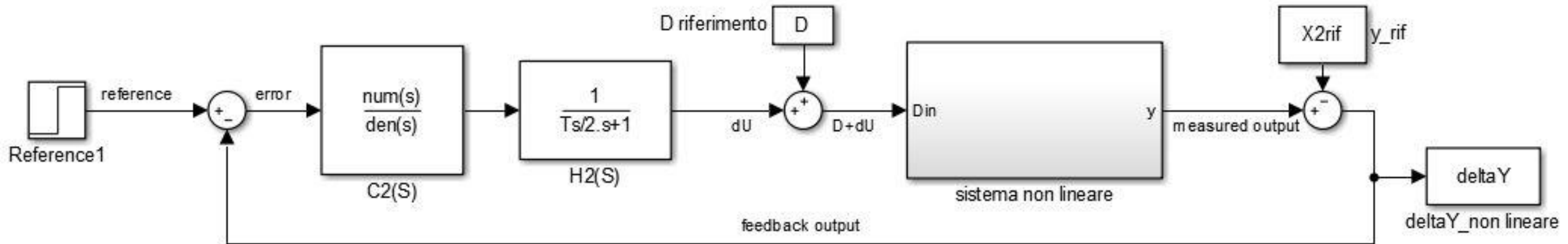
# 1. Sistema linearizzato con controllo applicato



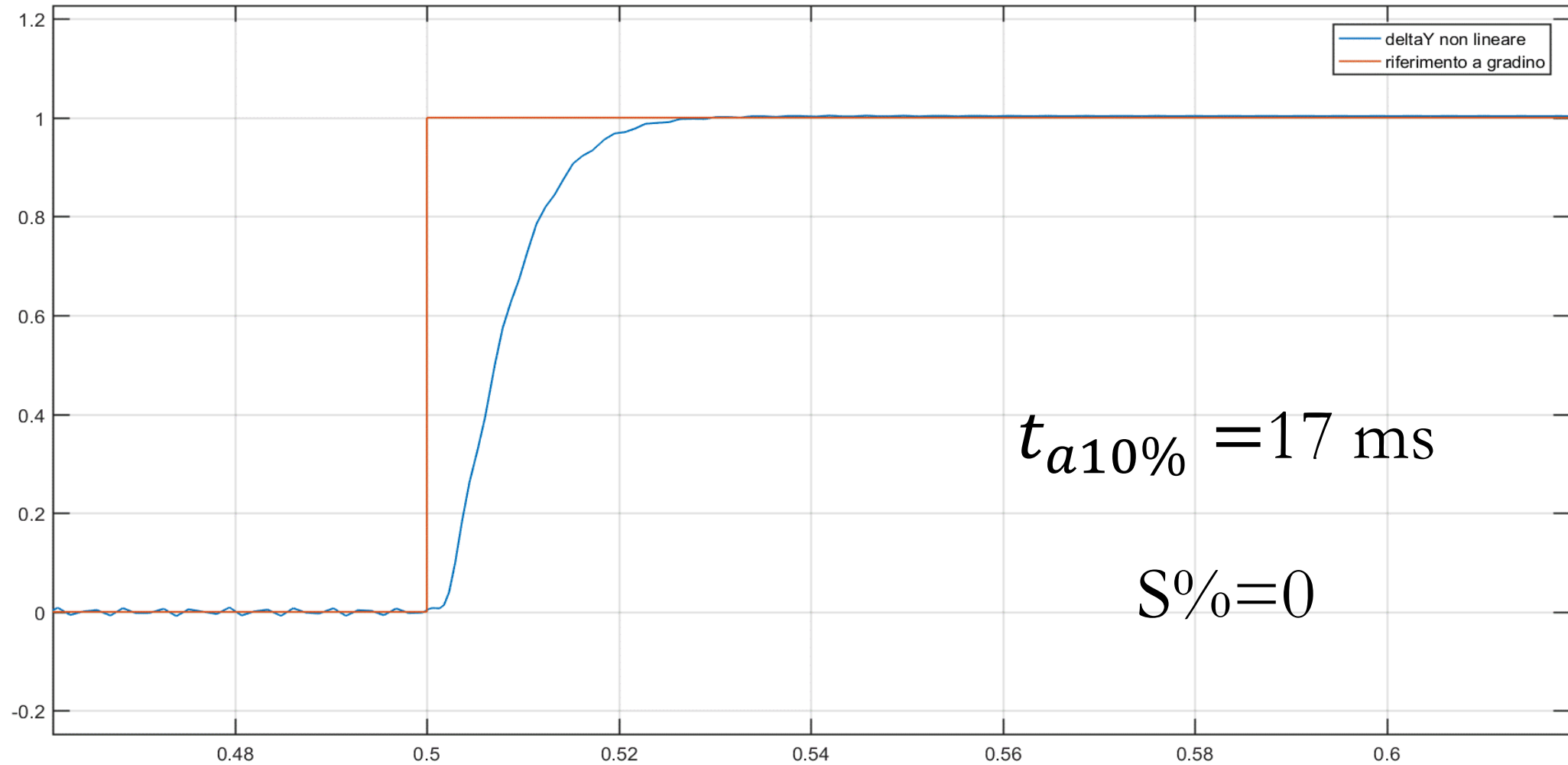
# Risposta al gradino unitario



## 2. Sistema non lineare con controllo applicato

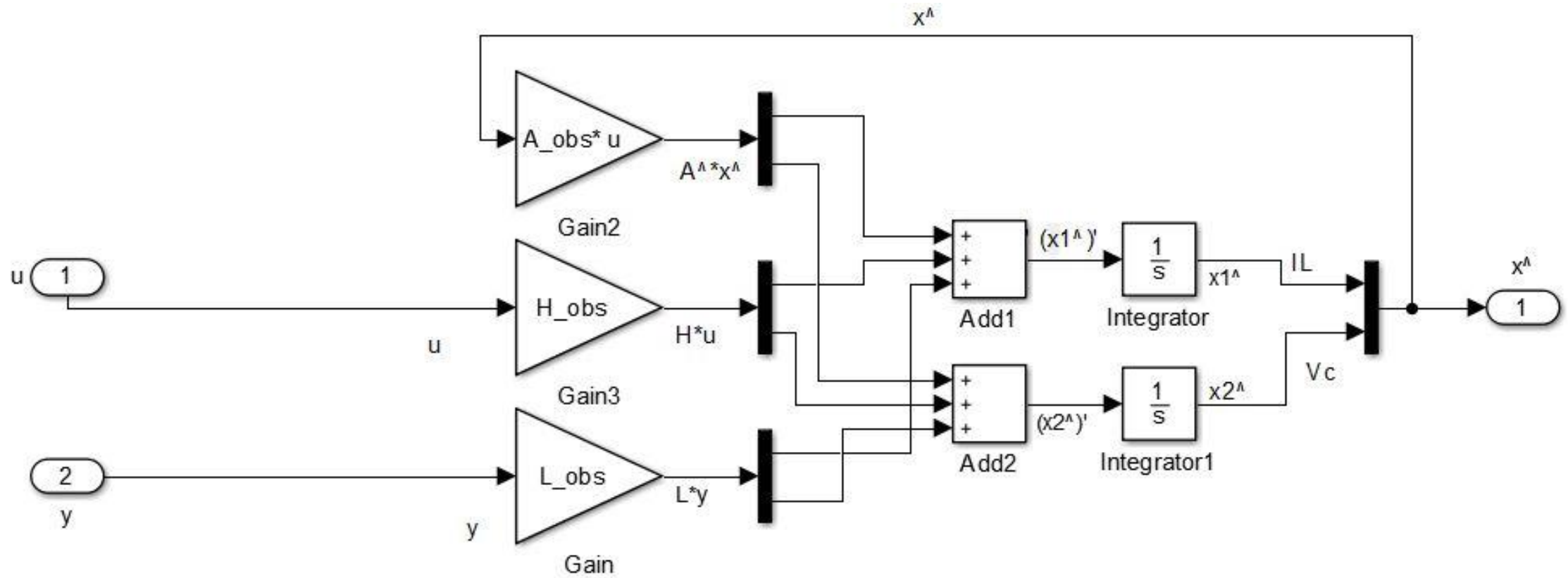


# Risposta al gradino unitario



### **3.3 Alternativa: Osservatore asintotico**

$$\dot{\hat{x}} = \hat{A} \cdot \hat{x} + H \cdot u + L \cdot y$$





$$H = B$$

$$\hat{A} = A - L \cdot C$$

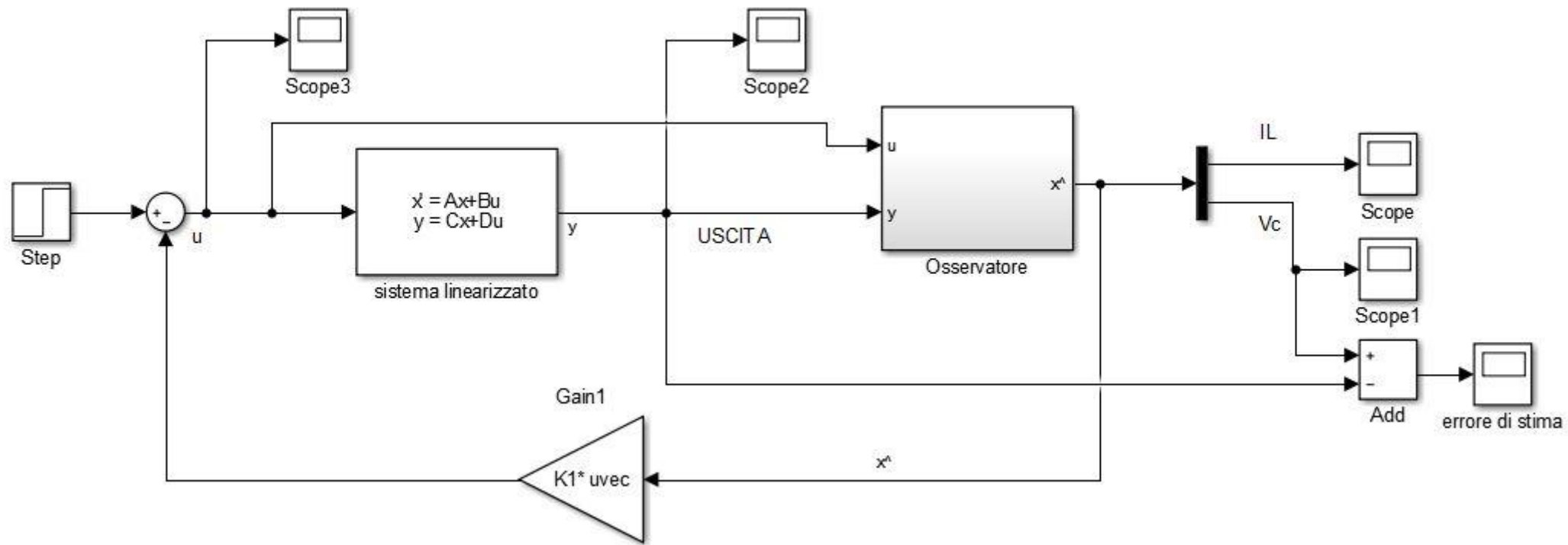
$\hat{A}$  asintoticamente stabile   $L$

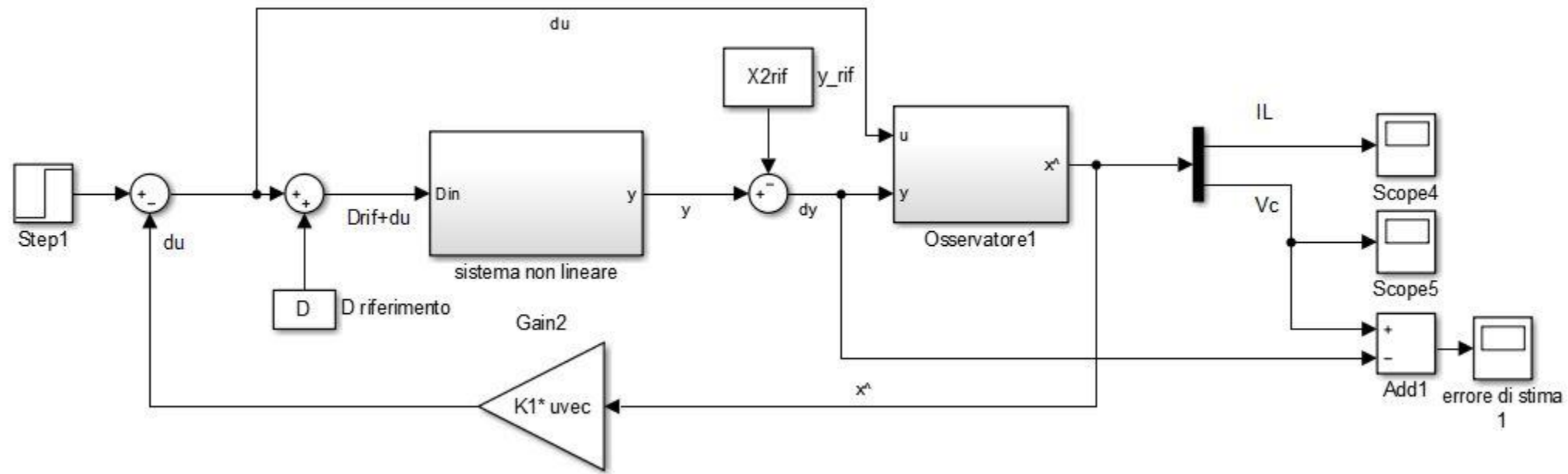
*place*  *Formula di Ackermann*

```

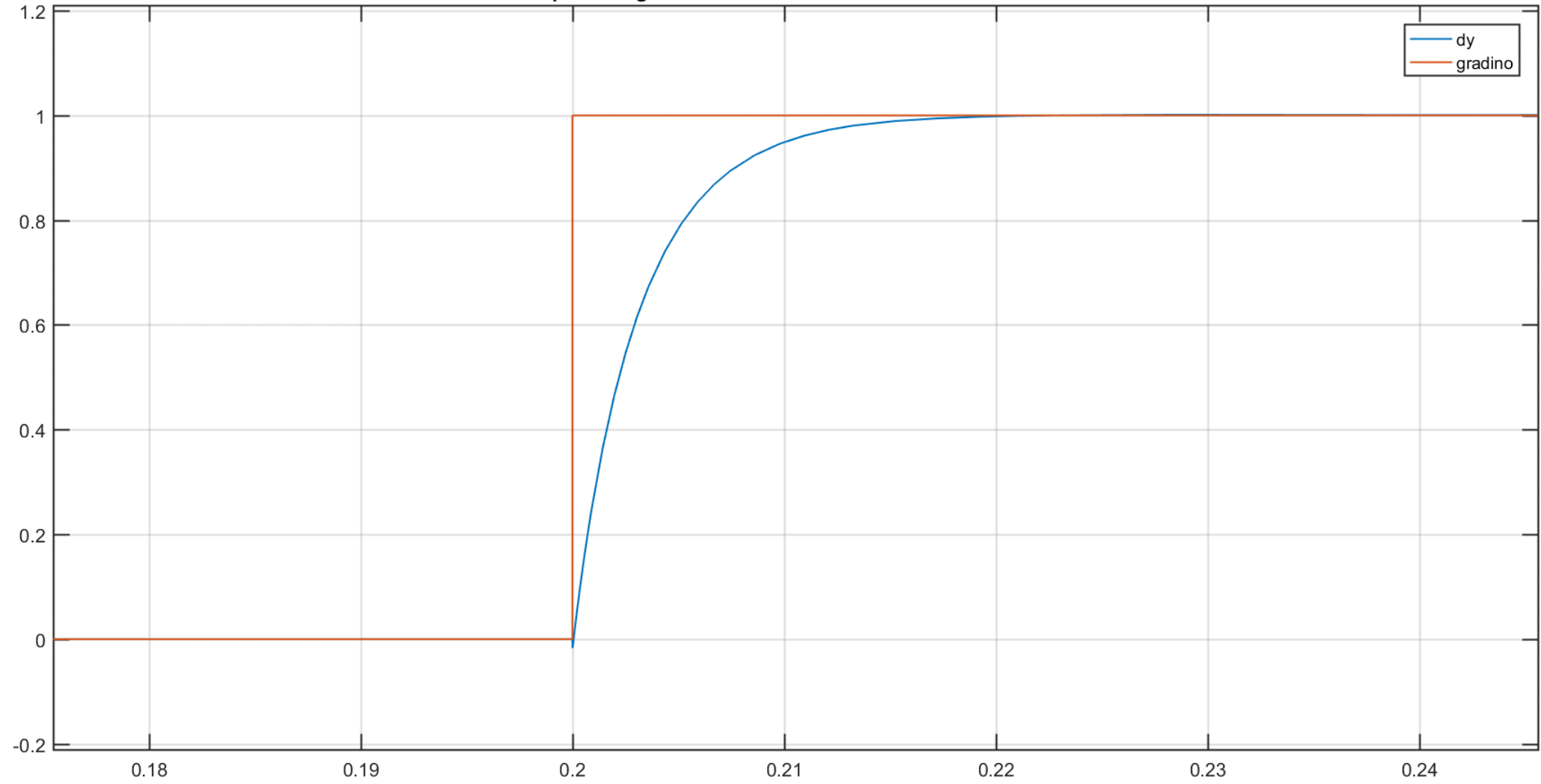
61 - K1=place (Am, Bm, pK)
62 - eig (Am-Bm*K1)
77 - p=[-10*lambda1 -10*lambda2]
--
82 %formula di assegnamento degli autovalori p
83 - LT=place (Am', Cm', p)
84
85 - L_obs=LT'
86
87 %matrice A osservatore (A^)|
88 - A_obs=Am-L_obs*Cm
89 - eig (A_obs)
90 - H_obs=Bm
91

```





risposta al gradino unitario - controllo con osservatore



## **4. DISCRETIZZAZIONE DEL CONTROLLO**

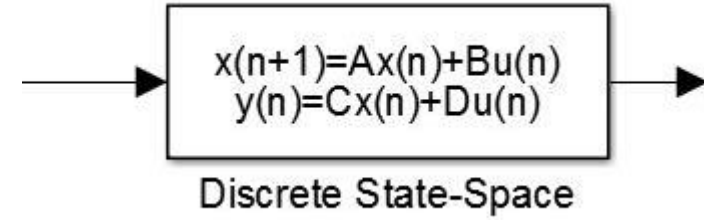
## **4.1 Trasformazione di Tustin**

La trasformazione di Tustin trasforma una  $D(s)$  analogica stabile (secondo e terzo quadrante, nella parte immaginaria) in una  $D(z)$  discreta stabile (circonferenza nell'origine di raggio unitario). Nel nostro caso, abbiamo calcolato su Matlab mediante l'istruzione **Dz=c2d(Cfin, Ts, 'tustin')**:

$$D(z) = \frac{0.001129 \cdot z^3 + 0.00113 \cdot z^2 - 0.001128 \cdot z - 0.001128}{z^3 - 2.101 \cdot z^2 + 1.203 \cdot z - 0.1014}$$



$$D(z) = \frac{0.001129 \cdot z^3 + 0.00113 \cdot z^2 - 0.001128 \cdot z - 0.001128}{z^3 - 2.101 \cdot z^2 + 1.203 \cdot z - 0.1014}$$



```

147 % 5 - DISCRETIZZAZIONE DEL SISTEMA CONTINUO
148 - Dz=c2d(Cfin,Ts,'tustin')
149 - [numDz, denDz]=tfdata(Dz,'v')
150 - [Ad, Bd, Cd, Dd]=tf2ss(numDz, denDz)

```

Ad =

```

2.1014 -1.2028 0.1014
1.0000 0 0
0 1.0000 0

```

Bd =

```

1
0
0

```

Cd =

```

0.0035 -0.0025 -0.0010

```

Dd =

```

0.0011

```

## **4.2 Alternativa: metodo con ZOH**

§ 5.1 - DISCRETIZZAZIONE DEL SISTEMA CONTINUO CON ZOH

```
DzZ=c2d(Cfin,Ts,'zoh')  
[numDzZ, denDzZ]=tfdata(DzZ,'v')  
[Ad_zoh,Bd_zoh,Cd_zoh,Dd_zoh]=tf2ss(numDzZ,denDzZ)
```



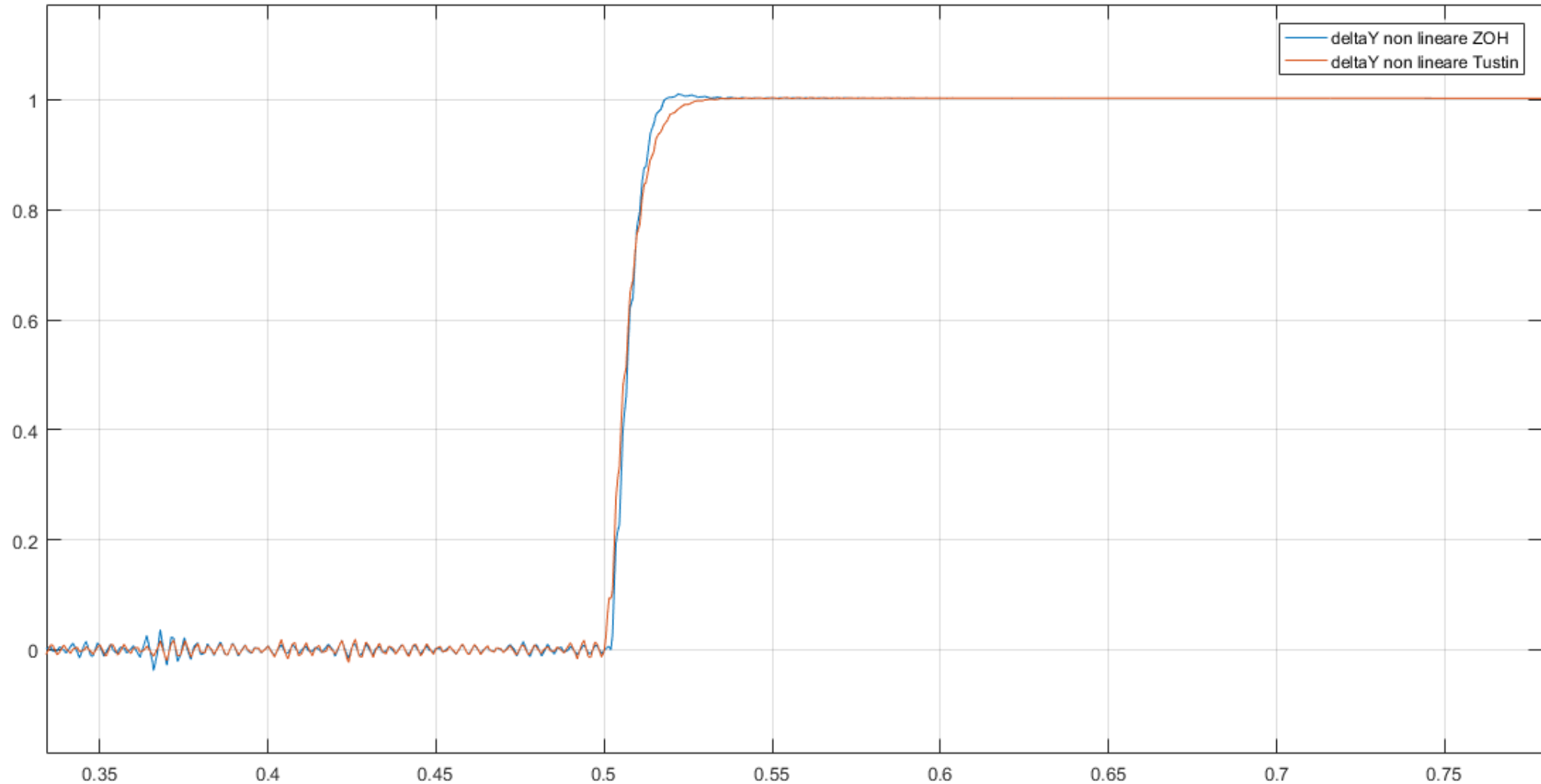
```
Ad_zoh =  
    2.1956   -1.3912    0.1956  
    1.0000         0         0  
         0    1.0000         0
```

```
Bd_zoh =  
    1  
    0  
    0
```

```
Cd_zoh =  
    0.0025   -0.0011   -0.0015
```

```
Dd_zoh =  
    0
```

# ➤ Confronto discretizzazione ZOH e Tustin: risposta al gradino del sistema non lineare



## **4.3 Verifica del controllo tempo discreto**

**Stabilità**



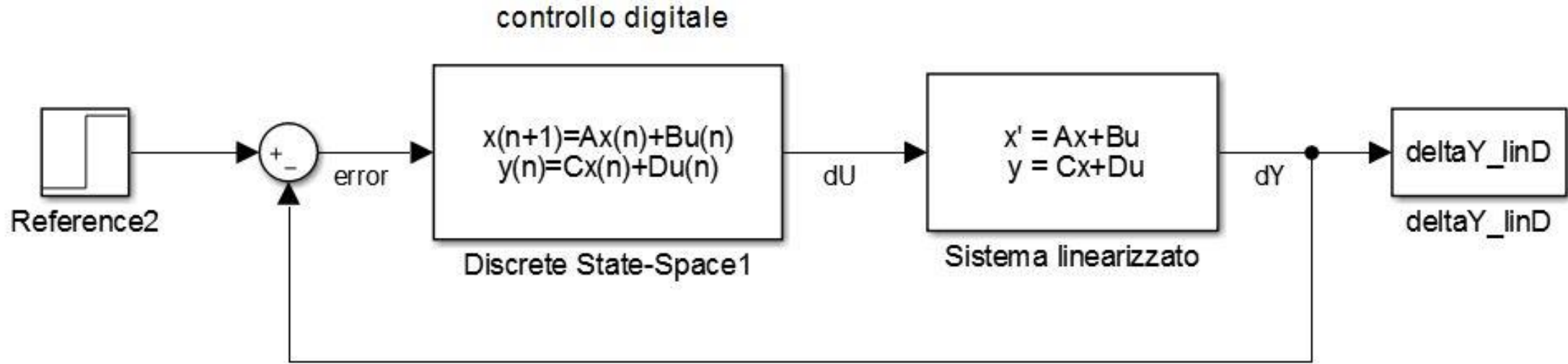
Le proprietà in tempo continuo si mantengono in tempo discreto, usando il metodo di discretizzazione di Tustin

## **TEST DELLE PRESTAZIONI**

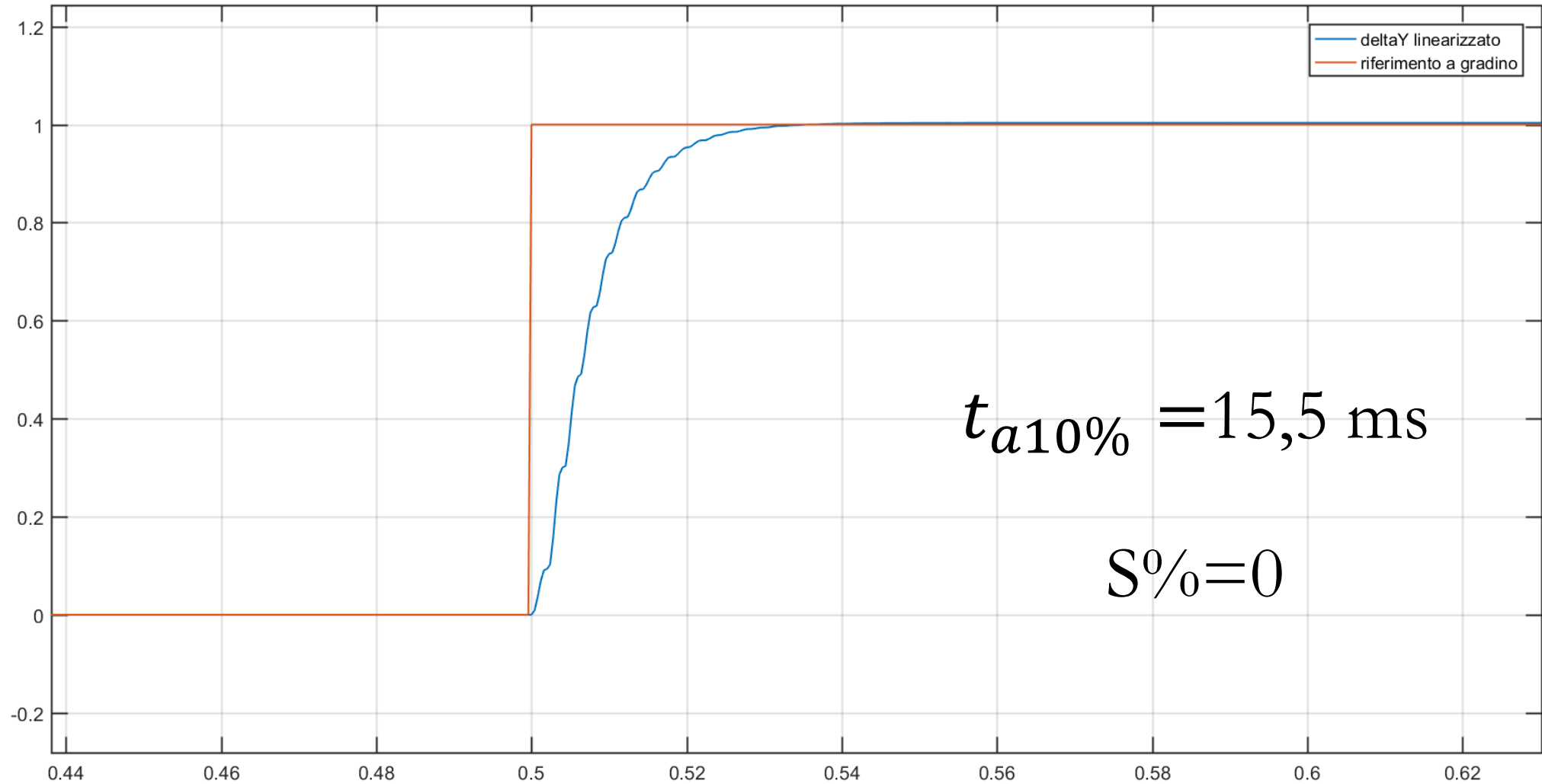
➤ **Risposta al riferimento di tipo gradino unitario:**

1. Su sistema linearizzato, con controllo digitale applicato
2. Su sistema non lineare, con controllo digitale applicato

# 1. Sistema linearizzato con controllo digitale applicato

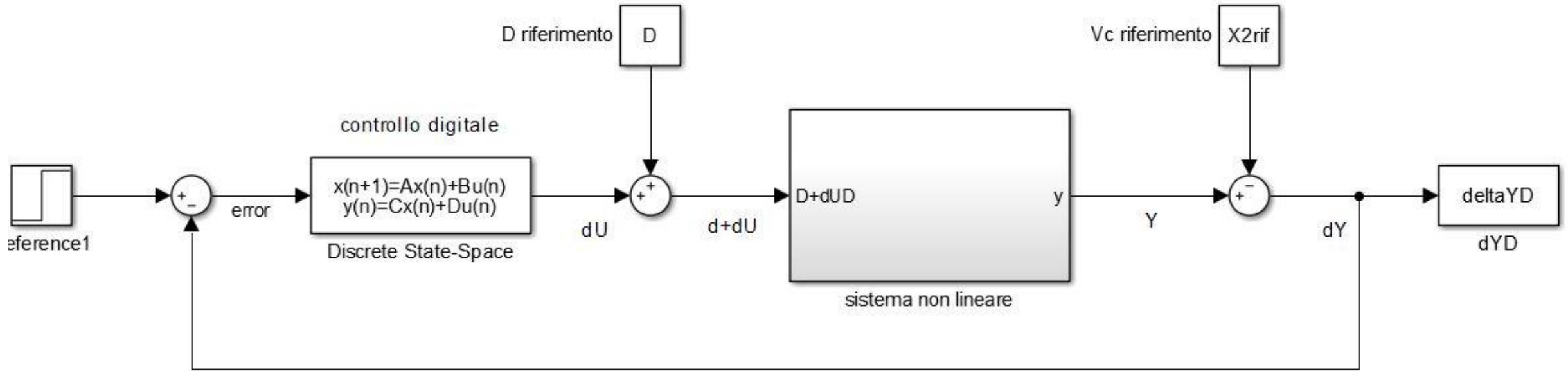


# Risposta al gradino unitario

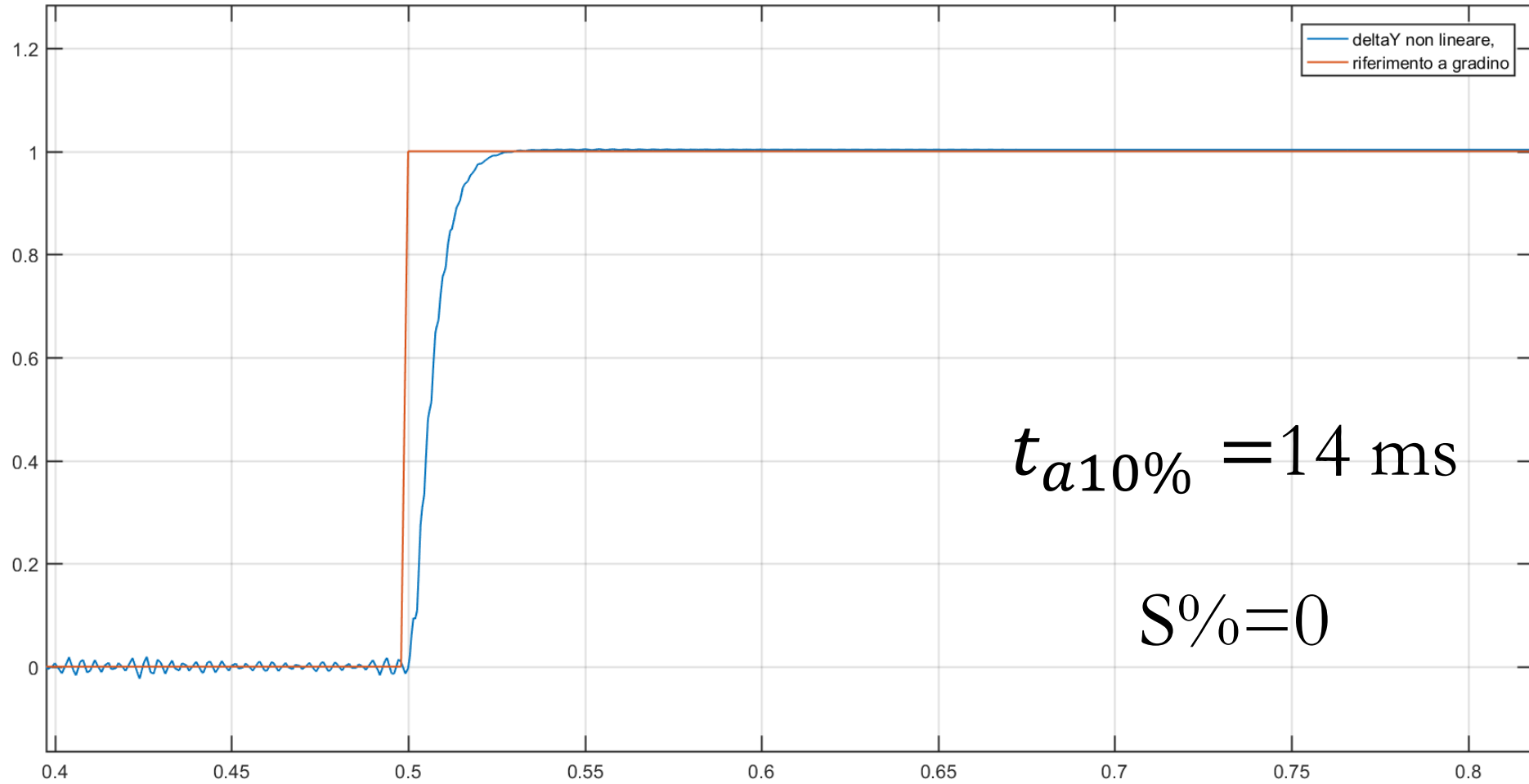




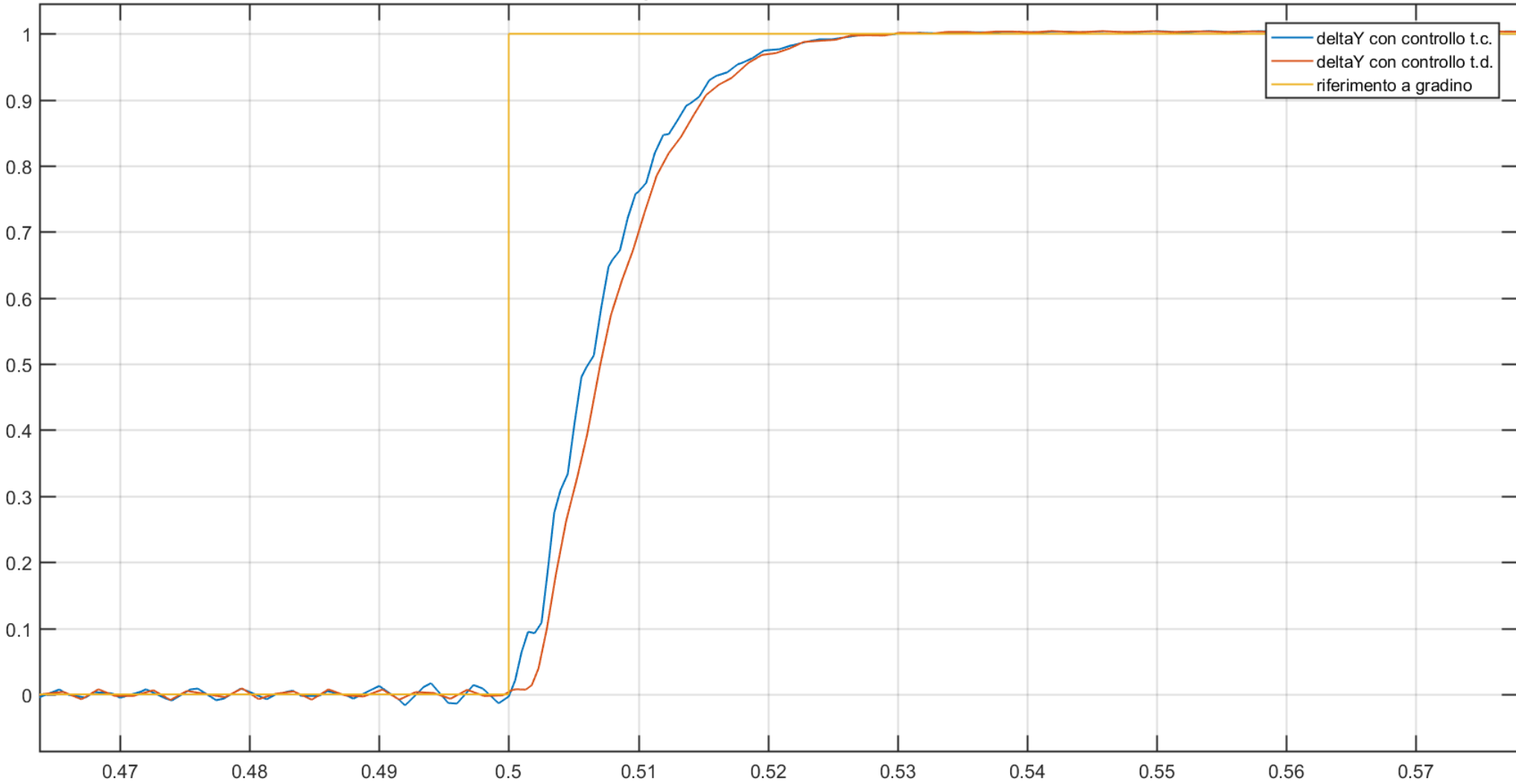
## 2. Sistema non lineare con controllo digitale applicato



# Risposta al gradino unitario

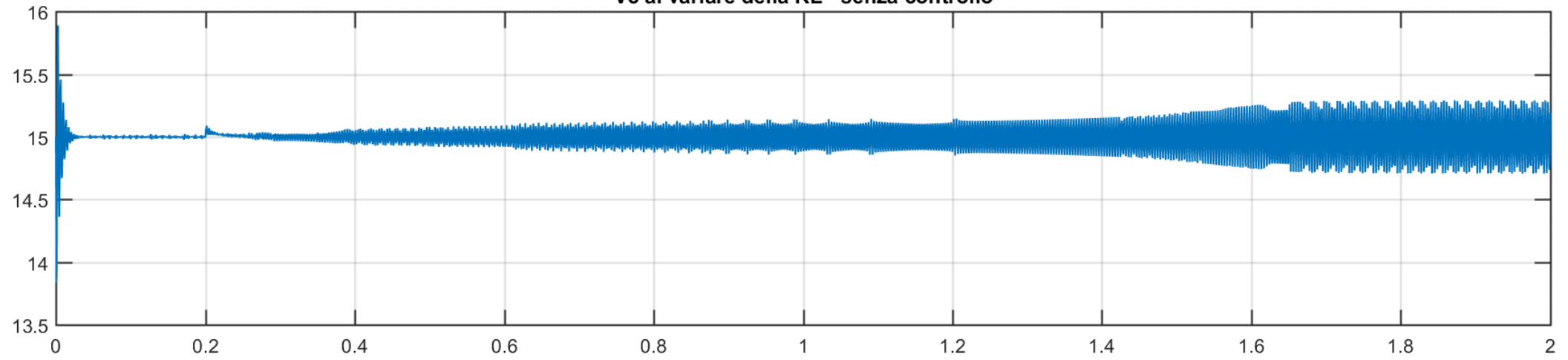


# Confronto tra le risposte con controllo t.c. e t.d.

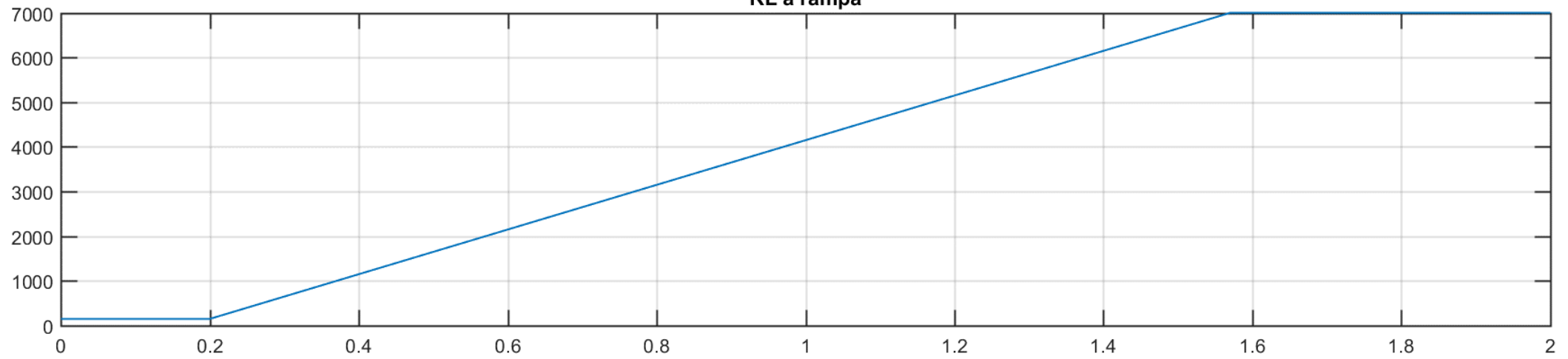


## **5. VERIFICHE AL VARIARE DEL CARICO (SIMULINK)**

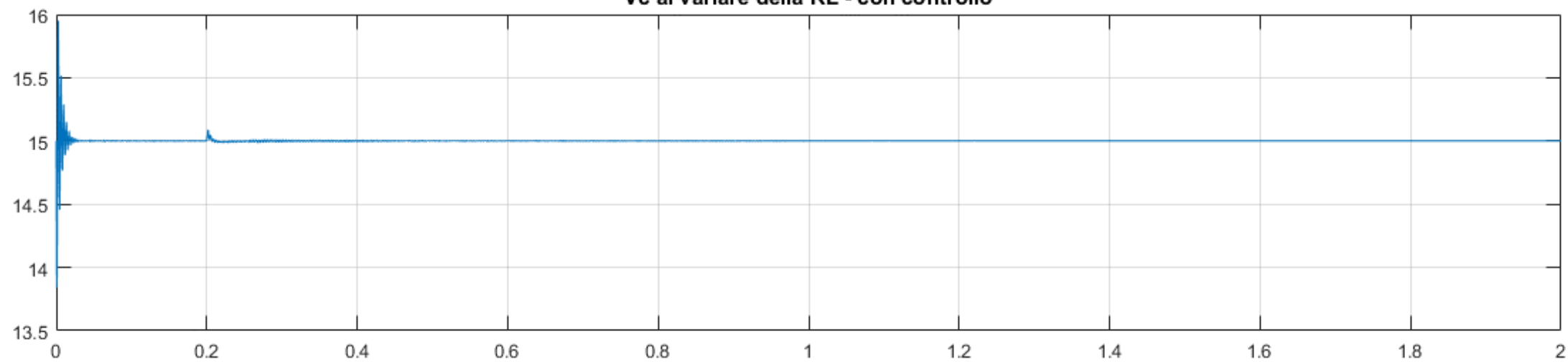
Vc al variare della RL - senza controllo



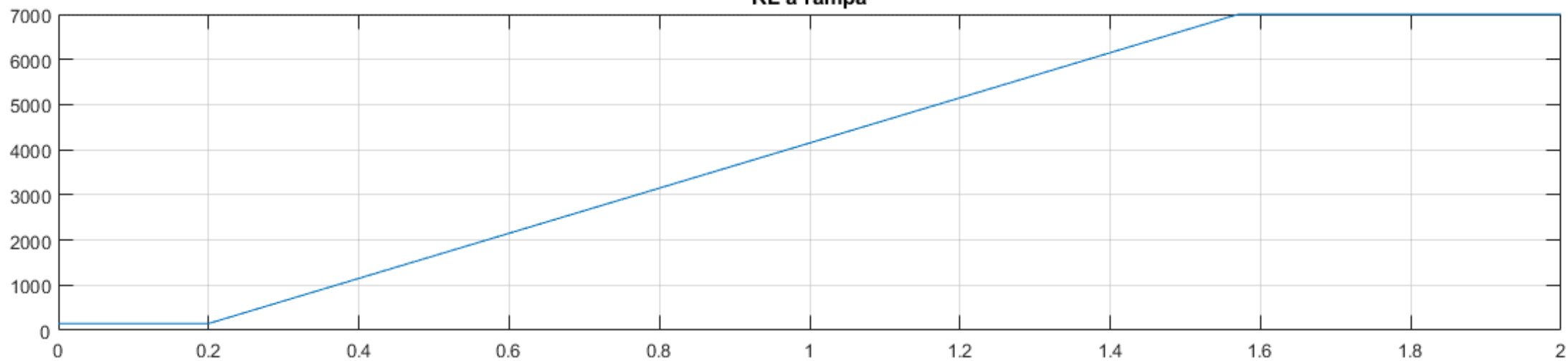
RL a rampa



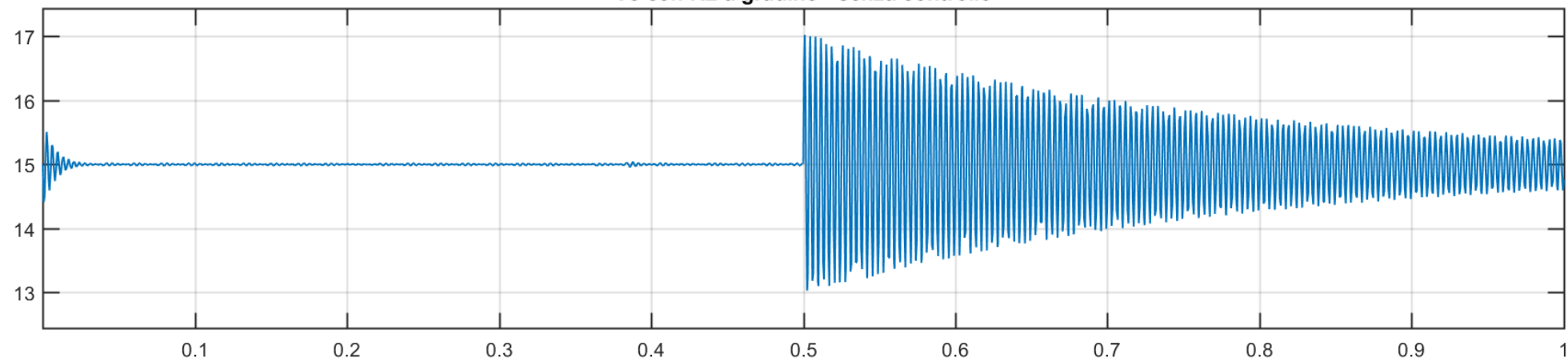
Vc al variare della RL - con controllo



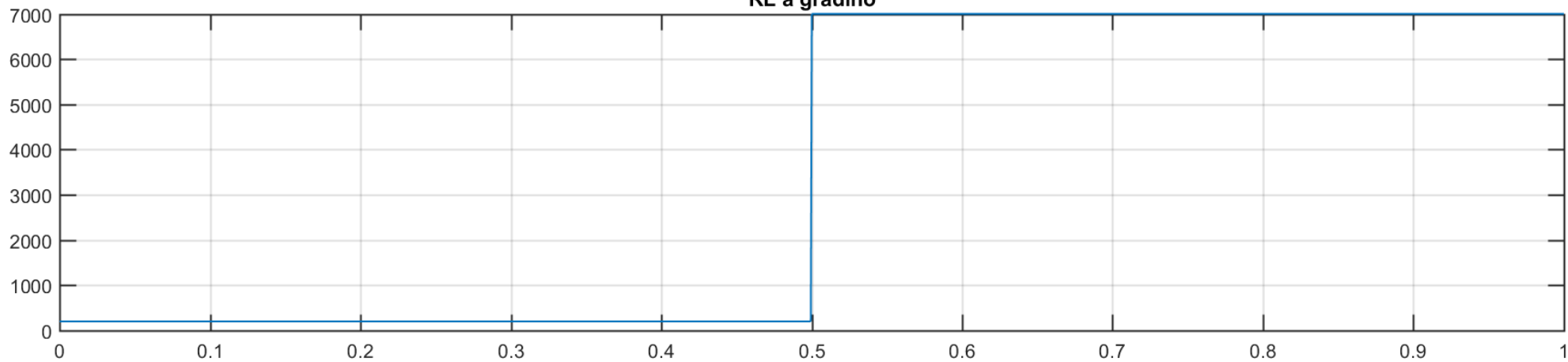
RL a rampa



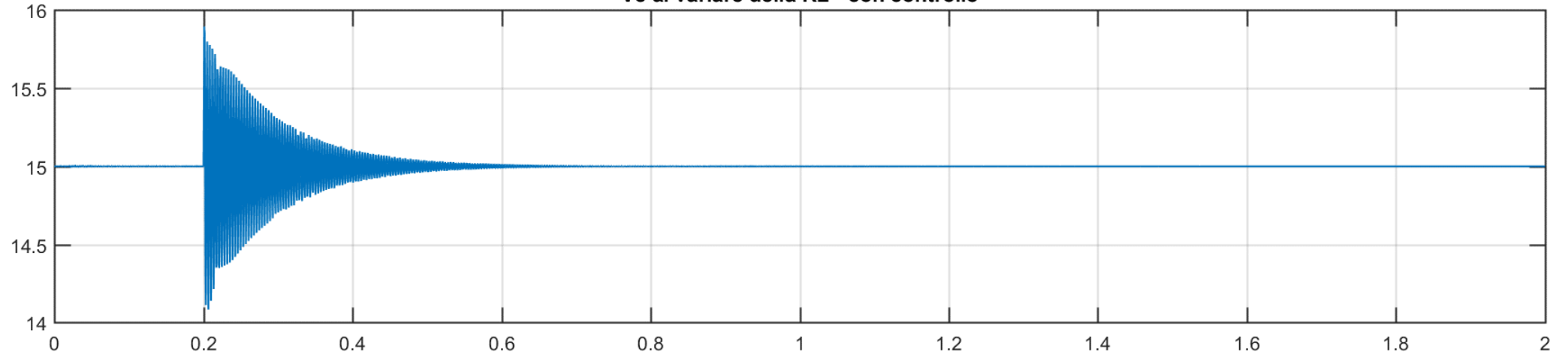
Vc con RL a gradino - senza controllo



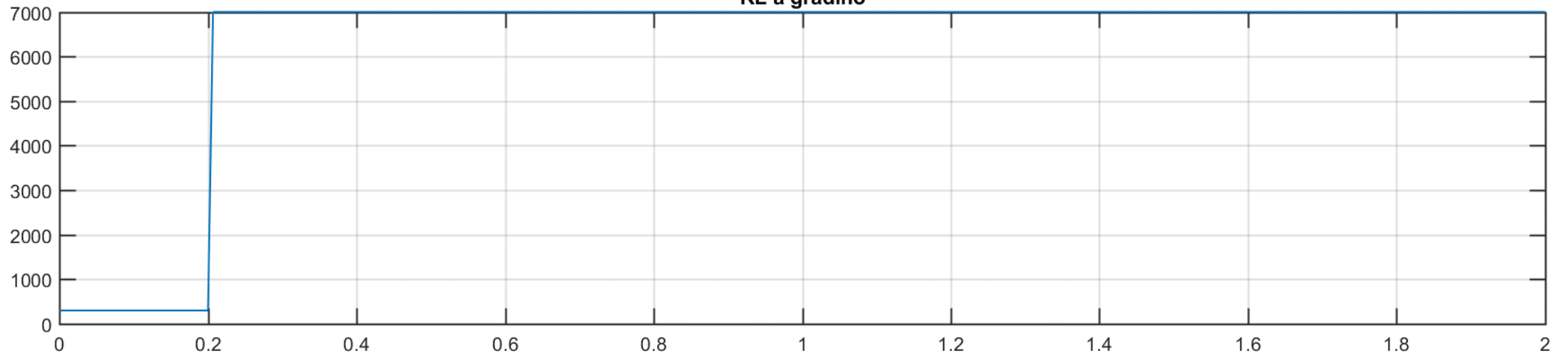
RL a gradino



Vc al variare della RL - con controllo



RL a gradino

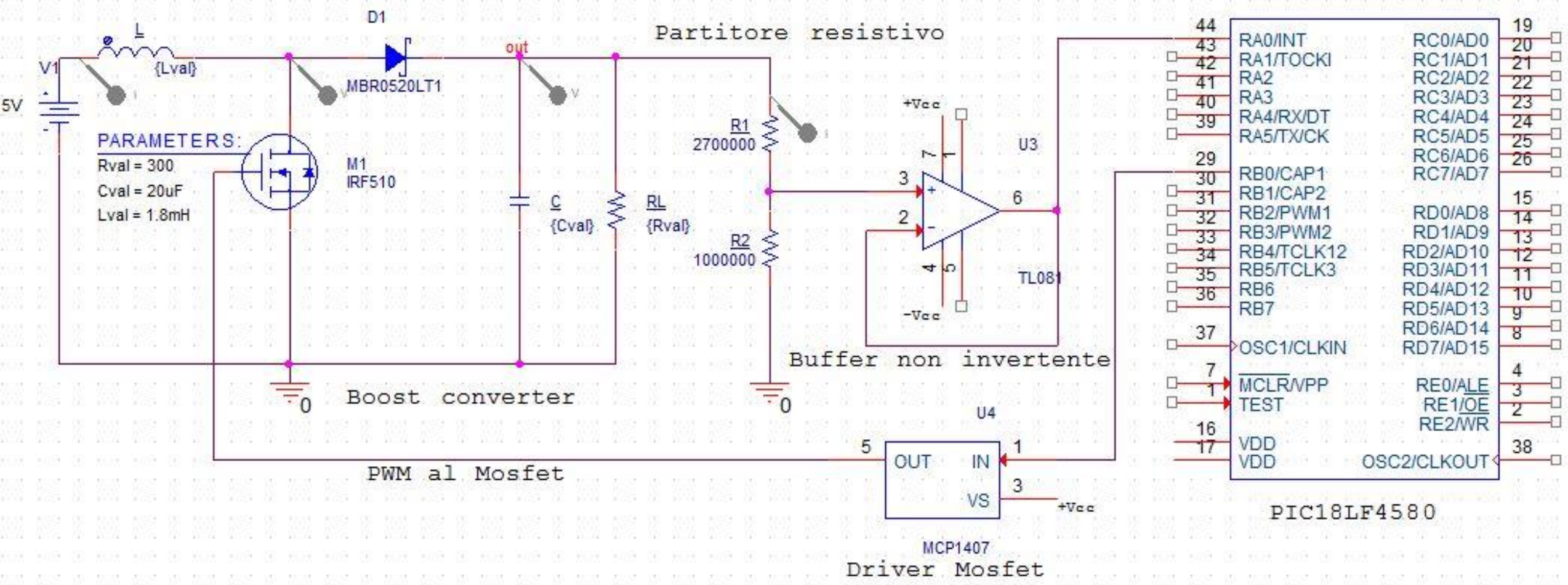


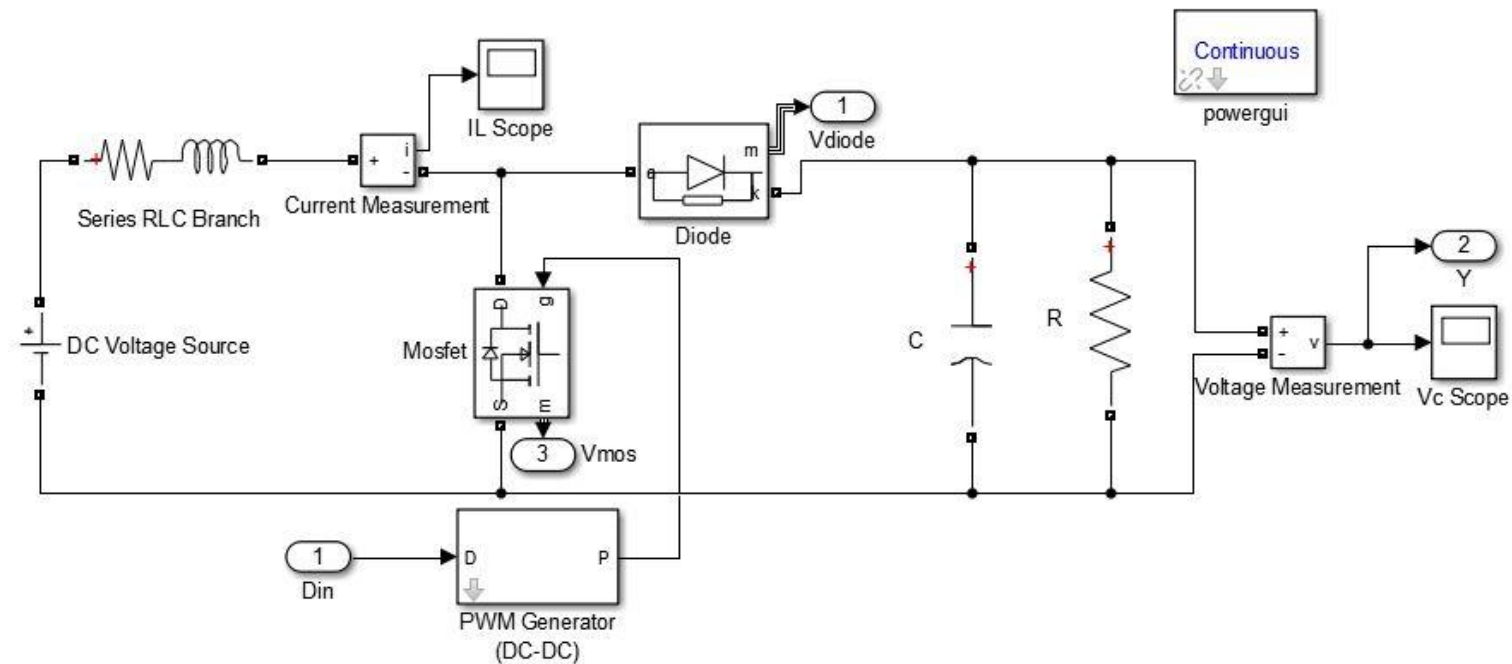
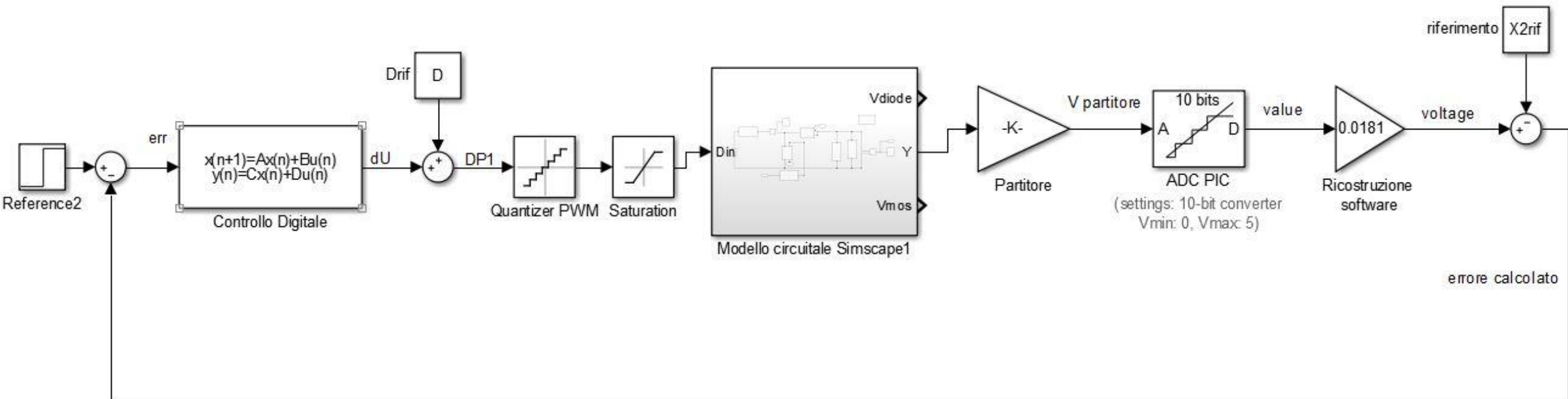


## **6. IMPLEMENTAZIONE E VERIFICHE SPERIMENTALI**

## Hardware:

- PIC18LF4580 come sistema di calcolo digitale
- Partitore di tensione per scalare la tensione di uscita di 15 o più volt;
- Buffer con amplificatore operazionale per disaccoppiamento tra uscita e ADC del PIC;
- Driver di corrente per il Mosfet
- Printed Circuit Board del sistema (boost più controllo digitale)





# Software PIC: algoritmo di controllo progettato

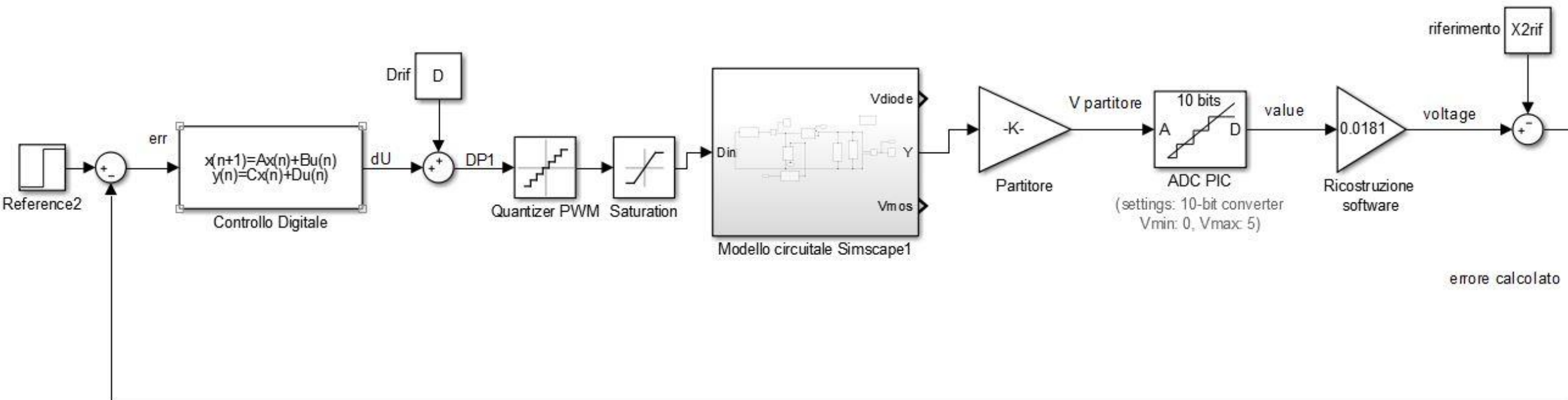


- Uso di interrupt software da timer0 (ogni 500ms);
- Uso di ADC interno;
- Uso del massimo clock disponibile

```

31 #INT_RTCC //interrupt da timer per le funzioni di controllo (fs=500Hz)
32 void RTCC_isr(void)
33 {
34 //pin di verifica
35 output_high(PIN_C4);
36
37 read_adc(ADC_READ_ONLY); //lettura ADC
38 value = read_adc(); //salvataggio valore letto
39 voltage=k*value; //conversione del valore numerico letto in una tensione (Vc)
40 err=-voltage+Vrif; //calcolo dell'errore rispetto al riferimento, cambiato di segno
41
42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)==> calcolo del dU da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;

```

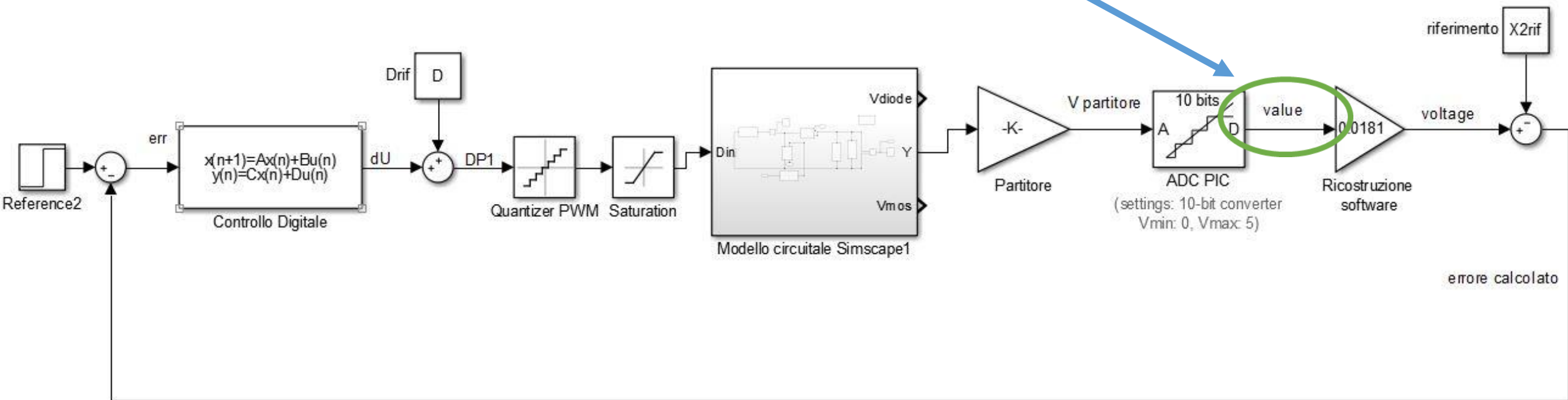




```

prova8.c
31 #INT_RTCC //interrupt da timer per le funzioni di controllo (fs=500Hz)
32 void RTCC_isr(void)
33 {
34 //pin di verifica
35 output_high(PIN_C4);
36
37 read_adc(ADC_READ_ONLY); //lettura ADC
38 value = read_adc(); //salvataggio valore letto
39 voltage=k*value; //conversione del valore numerico letto in una tensione (Vc)
40 err=-voltage+Vrif; //calcolo dell'errore rispetto al riferimento, cambiato di segno
41
42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del dU da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;

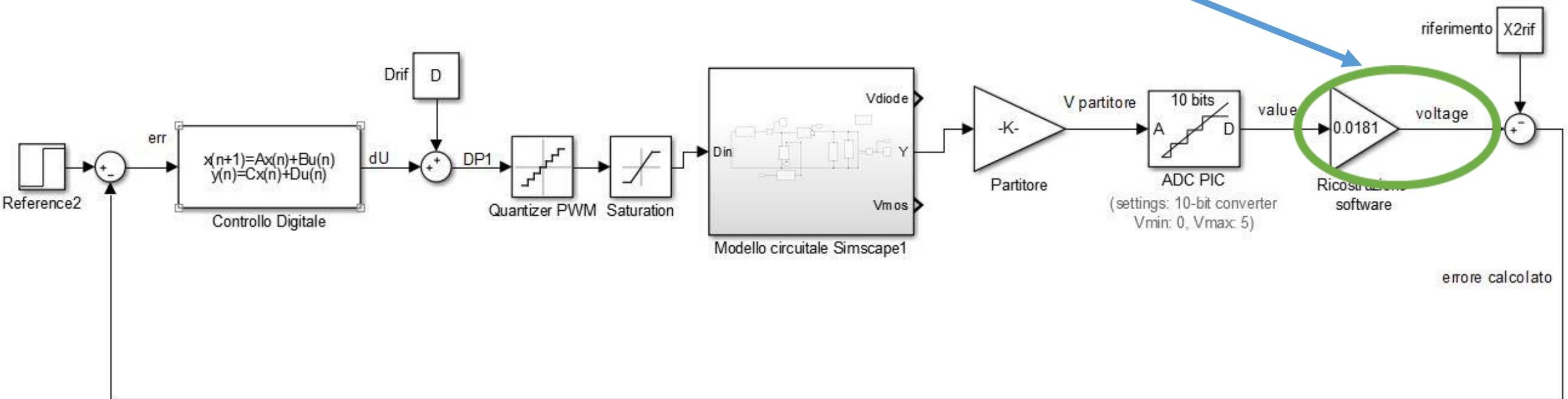
```



```

prova8.c
31 #INT_RTCC //interrupt da timer per le funzioni di controllo (fs=500Hz)
32 void RTCC_isr(void)
33 {
34 //pin di verifica
35 output_high(PIN_C4);
36
37 read_adc(ADC_READ_ONLY); //lettura ADC
38 value = read_adc(); //salvataggio valore letto
39 voltage=k*value; //conversione del valore numerico letto in una tensione (Vc)
40 err=-voltage+Vrif; //calcolo dell'errore rispetto al riferimento, cambiato di segno
41
42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del dU da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;

```

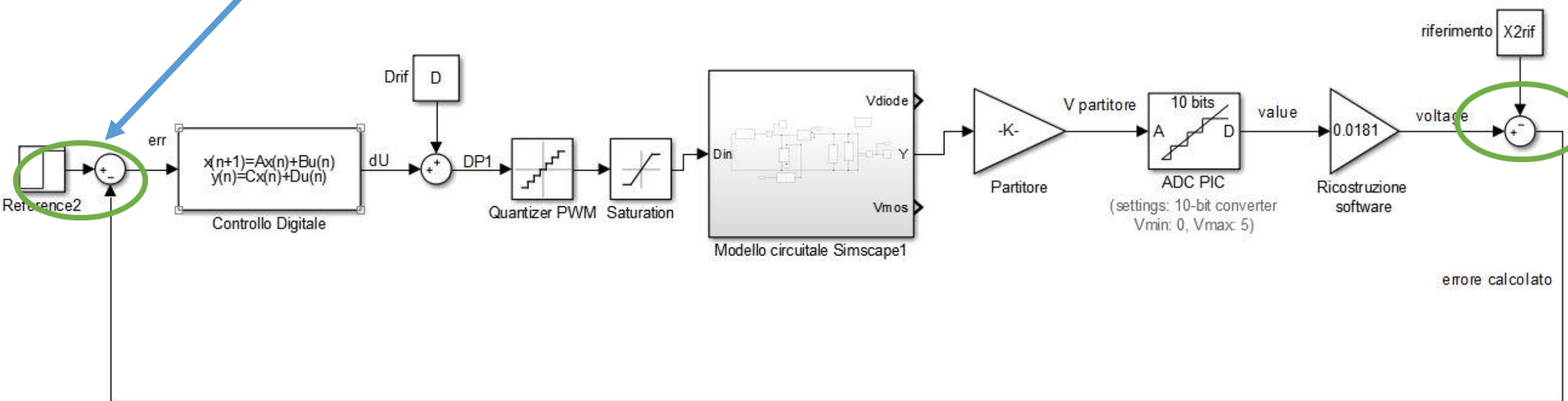




```

prova8.c
31 #INT_RTCC //interrupt da timer per le funzioni di controllo (fs=500Hz)
32 void RTCC_isr(void)
33 {
34 //pin di verifica
35 output_high(PIN_C4);
36
37 read_adc(ADC_READ_ONLY); //lettura ADC
38 value = read_adc(); //salvataggio valore letto
39 voltage=k*value; //conversione del valore numerico letto in una tensione (Vc)
40 err=-voltage+Vrif; //calcolo dell'errore rispetto al riferimento, cambiato di segno
41
42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del dU da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;

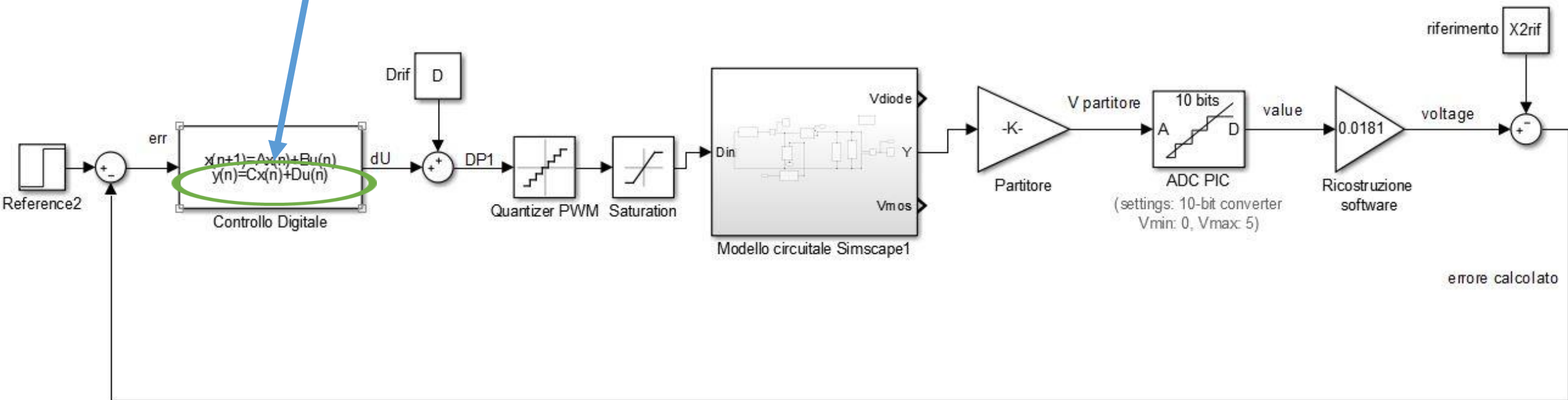
```



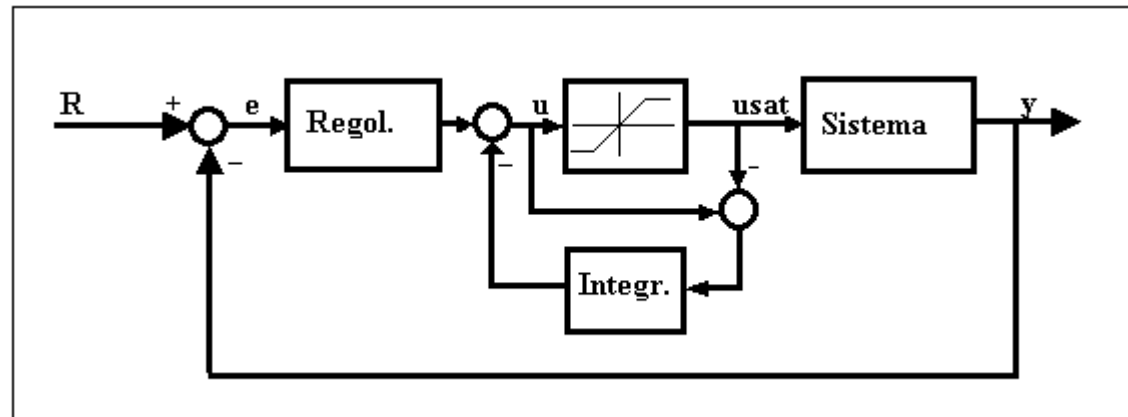
```

42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del dU da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;
53 Y=sCX+Derr;
54
55 //ANTI WIND-UP
56 if(Y>=DD1){sCX=sCX+((DD1)-Y); Y=sCX+Derr;}
57 else if(Y<=-Drif){sCX=sCX+((-Drif)-Y); Y=sCX+Derr;}
58
59 //AGGIORNAMENTO DEL PWM DUTY
60 du= Y; //uscita del controllo(Y)=du (correzione sull'ingresso pwm del boost)
61 DP1=(Drif+du); //somma tra duty di riferimento e du da applicare
62 DP=(int16)(DP1*1000); //moltiplicazione per 1000(funzione set duty percent accetta valori tra 0 e 1000), e casting
63 pwm_set_duty_percent(DP); //applicazione del duty calcolato
64

```



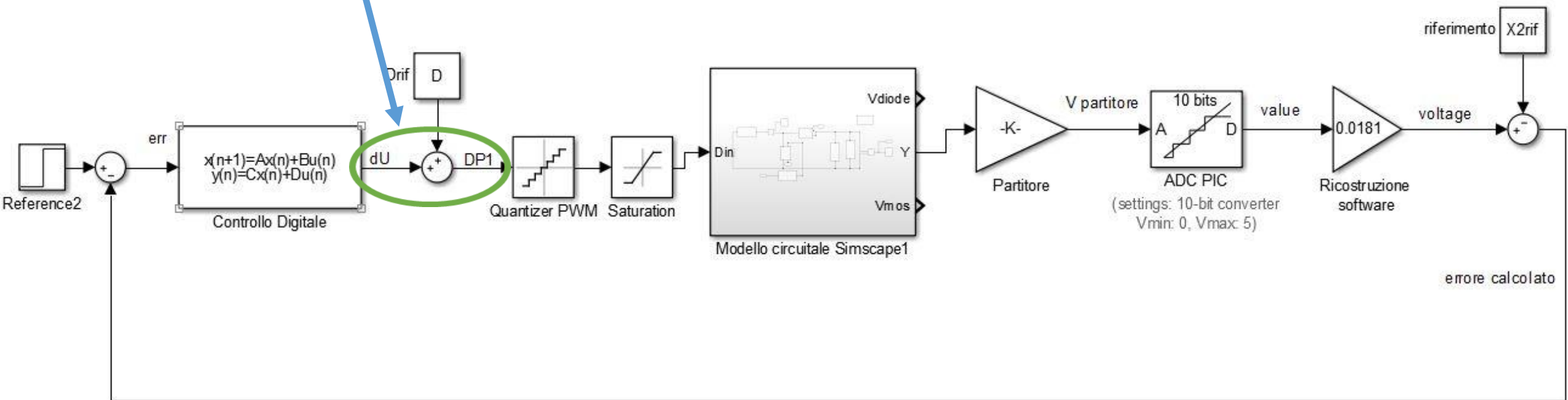
```
42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del du da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;
53 Y=sCX+Derr;
54
55 //ANTI WIND-UP
56 if(Y>=DD1){sCX=sCX+((DD1)-Y); Y=sCX+Derr;}
57 else if(Y<=-Drif){sCX=sCX+((-Drif)-Y); Y=sCX+Derr;}
58
59 //AGGIORNAMENTO DEL PWM DUTY
60 du=Y; //uscita del controllo(Y)=du (correzione sull'ingresso pwm del boost)
61 DP1=(Drif+du); //somma tra duty di riferimento e du da applicare
62 DP=(int16)(DP1*1000); //moltiplicazione per 1000(funzione set duty percent accetta valori tra 0 e 1000), e casting
63 pwm_set_duty_percent(DP); //applicazione del duty calcolato
64
```



```

42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del du da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;
53 Y=sCX+Derr;
54
55 //ANTI WIND-UP
56 if(Y>DD1){sCX=sCX+((DD1)-Y); Y=sCX+Derr;}
57 else if(Y<=-Drif){sCX=sCX+((-Drif)-Y); Y=sCX+Derr;}
58
59 //AGGIORNAMENTO DEL PWM DUTY
60 du= Y; //uscita del controllo(Y)=du (correzione sull'ingresso pwm del boost)
61 DP1=(Drif+du); //somma tra duty di riferimento e du da applicare
62 DP=(int16)(DP1*1000); //moltiplicazione per 1000(funzione set duty percent accetta valori tra 0 e 1000), e casting
63 pwm_set_duty_percent(DP); //applicazione del duty calcolato
64

```

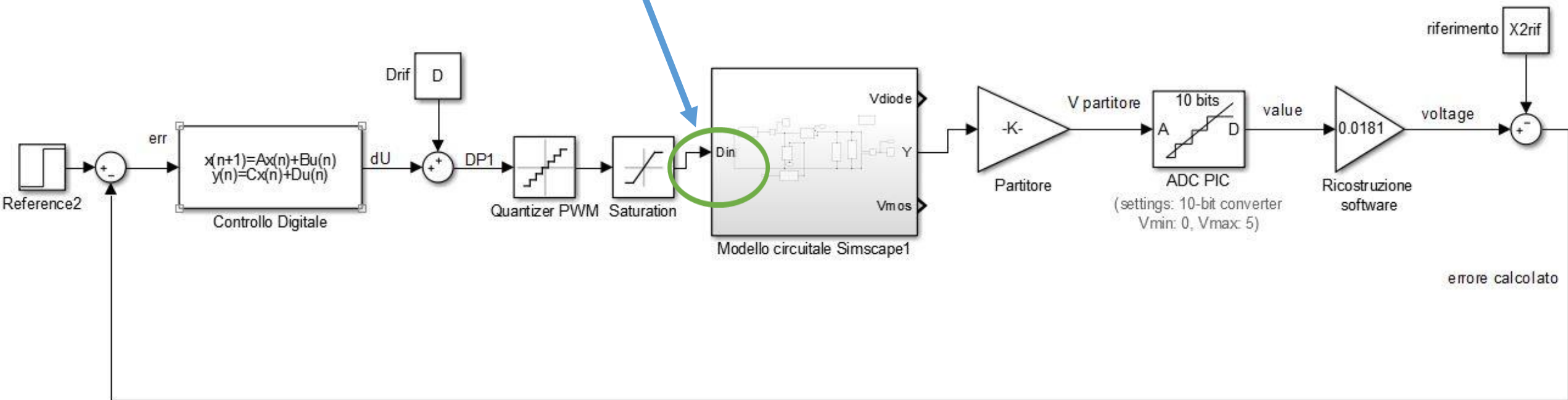




```

42 //FIRMWARE DEL CONTROLLO DIGITALE
43
44 //CALCOLO DELL'USCITA, NOTO ERRORE E STATO ATTUALE
45 //calcolo dell'uscita Y(k) all'istante k (attuale)===> calcolo del du da applicare
46 CX[0]=C11*XA[0];
47 CX[1]=C12*XA[1];
48 CX[2]=C13*XA[2];
49 //calcolo della somma dei prodotti di C*X
50 sCX=CX[0]+CX[1]+CX[2];
51 //Y=C*x(k)+D*u (u=err)
52 Derr=Dd*err;
53 Y=sCX+Derr;
54
55 //ANTI WIND-UP
56 if(Y>=DD1){sCX=sCX+((DD1)-Y); Y=sCX+Derr;}
57 else if(Y<=-Drif){sCX=sCX+((-Drif)-Y); Y=sCX+Derr;}
58
59 //AGGIORNAMENTO DEL PWM DUTY
60 du= Y; //uscita del controllo(Y)=du (correzione sull'ingresso pwm del boost)
61 DP1=(Drif+du); //somma tra duty di riferimento e du da applicare
62 DP=(int16)(DP1*1000); //moltiplicazione per 1000(funzione set duty percent accetta valori tra 0 e 1000), e casting
63 pwm_set_duty_percent(DP); //applicazione del duty calcolato
64

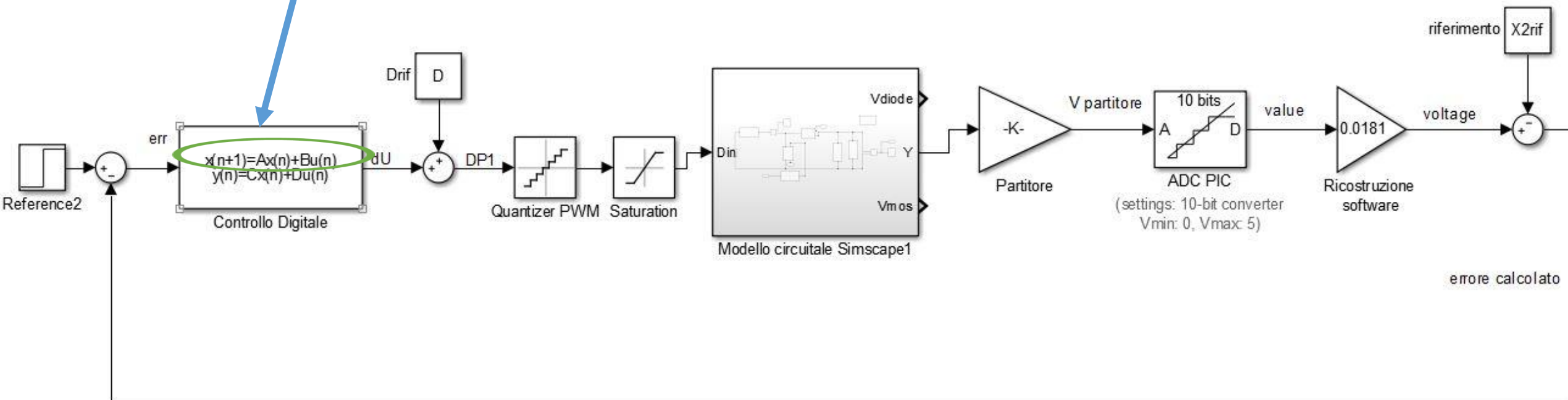
```



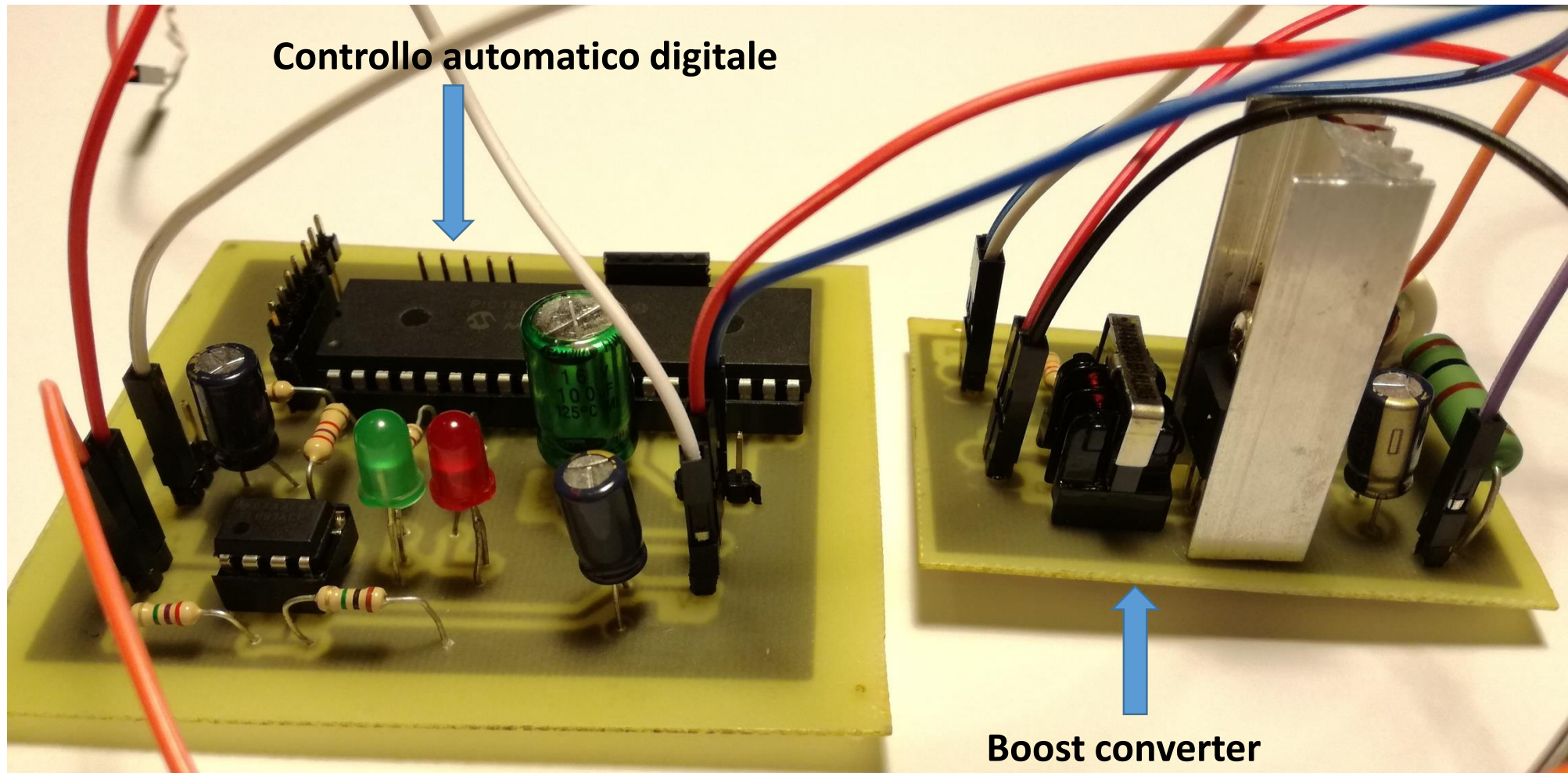
```

64
65 //CALCOLO DELLO STATO SUCCESSIVO X(K+1), NOTO ERRORE E STATO ATTUALE (=STATO SUCCESSIVO, AL PASSO PRECEDENTE)
66 //prodotto A*XA
67 //prodotto riga1
68 p_riga1[0]= A11*XA[0]; p_riga1[1]=A12*XA[1]; p_riga1[2]=A13*XA[2];
69 //somma dei prodotti
70 AXA11=p_riga1[0]+p_riga1[1]+p_riga1[2];
71 //riempimento altre celle diverse da 0
72 AXA21=XA[0];
73 AXA31=XA[1];
74 //prodotto B*U (un solo elemento diverso da 0)
75 //BU=err;
76 //somma A*X(K)+B*u(k) = AXa+BU ==> x(k+1)=A*X(k)+B*u(k) uscita all'istante k+1 (successivo)
77 XS[0]=AXA11+err;
78 XS[1]=AXA21;
79 XS[2]=AXA31;
80 //X attuale (PER IL PASSO SUCCESSIVO) = X successivo VALUTATO A QUESTO PASSO , reset di X successivo per nuovo calcolo
81 //XA=XS; XS=0;
82 XA[0]=XS[0];
83 XA[1]=XS[1];
84 XA[2]=XS[2];
85
86
87 //accensione led di segnalazione
88 if((err>=0.5)|| (err<=-0.5)){ output_high(PIN_A5); output_low(PIN_A3);} //accensione led verde se l'errore è minore di 0.5V
89 else {output_high(PIN_A3); output_low(PIN_A5);}

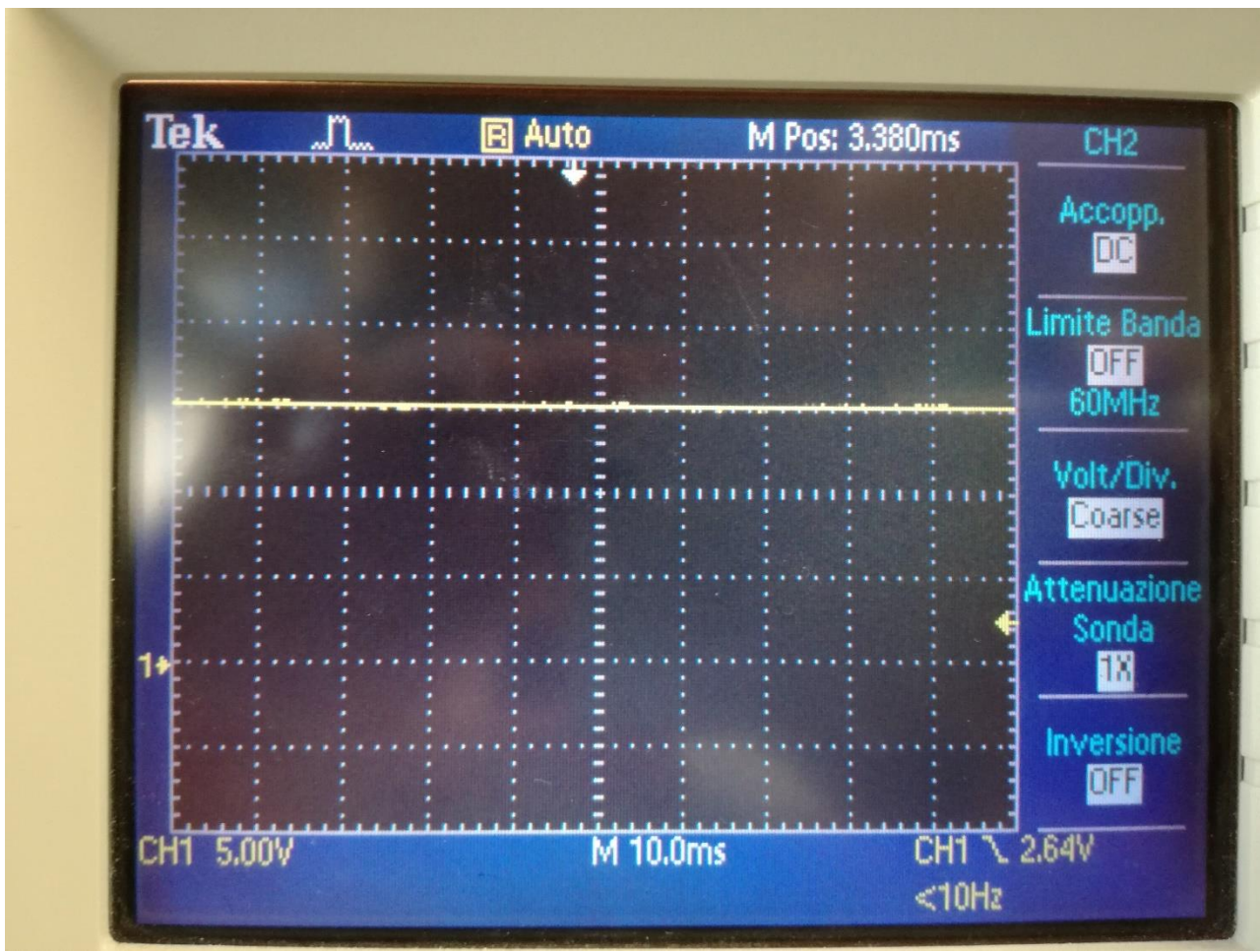
```



# Realizzazione del sistema

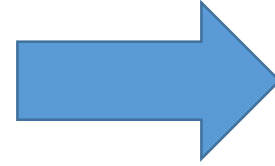






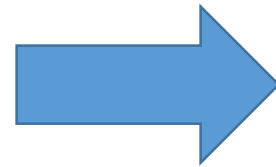


Sistema senza controllo



Variazione  
dell'uscita tra 14V e  
16V, variando RL

Sistema con controllo



Variazione dell'uscita  
tra 15.3V e 14.8V,  
variando RL