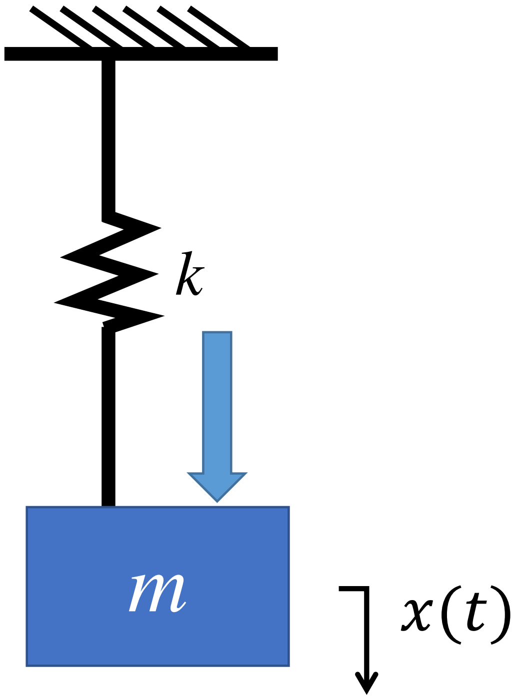
Dinâmica de Sistemas Navais e Oceânicos

PNV3314 Dinâmica de Sistemas

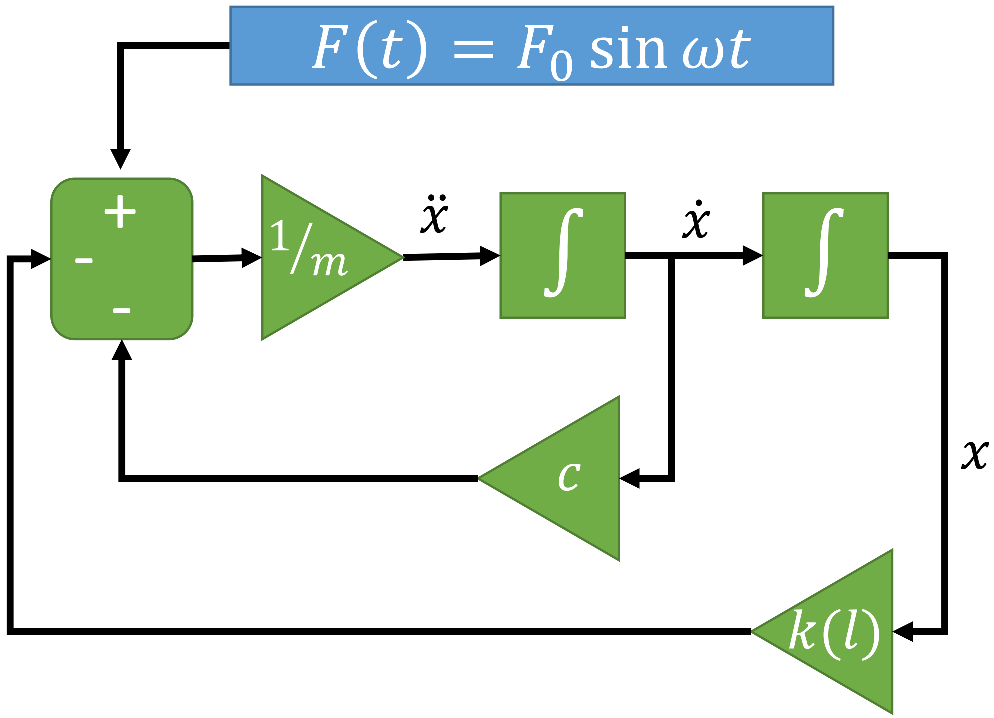
Aula 21

Relembre...

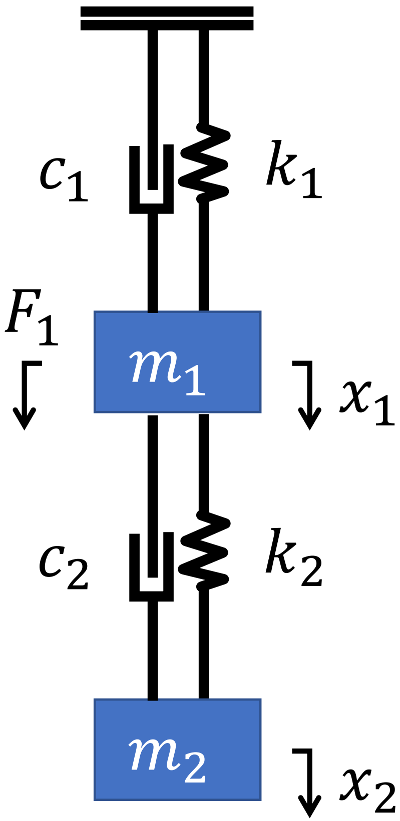


$$m\ddot{x} + c\dot{x} + kx = F_0 \sin \omega t$$

$$\ddot{x} = \frac{1}{m}([F_0 \sin \omega t] - c\dot{x} - kx)$$



Simulação de sistema de dois graus de liberdade



As equações do movimento

$$m_1\ddot{x}_1 + c_1\dot{x}_1 + c_2(\dot{x}_1 - \dot{x}_2) + k_1x_1 + k_2(x_1 - x_2) = F_1$$

$$m_2\ddot{x}_2 + c_2(\dot{x}_2 - \dot{x}_1) + k_2(x_2 - x_1) = 0$$

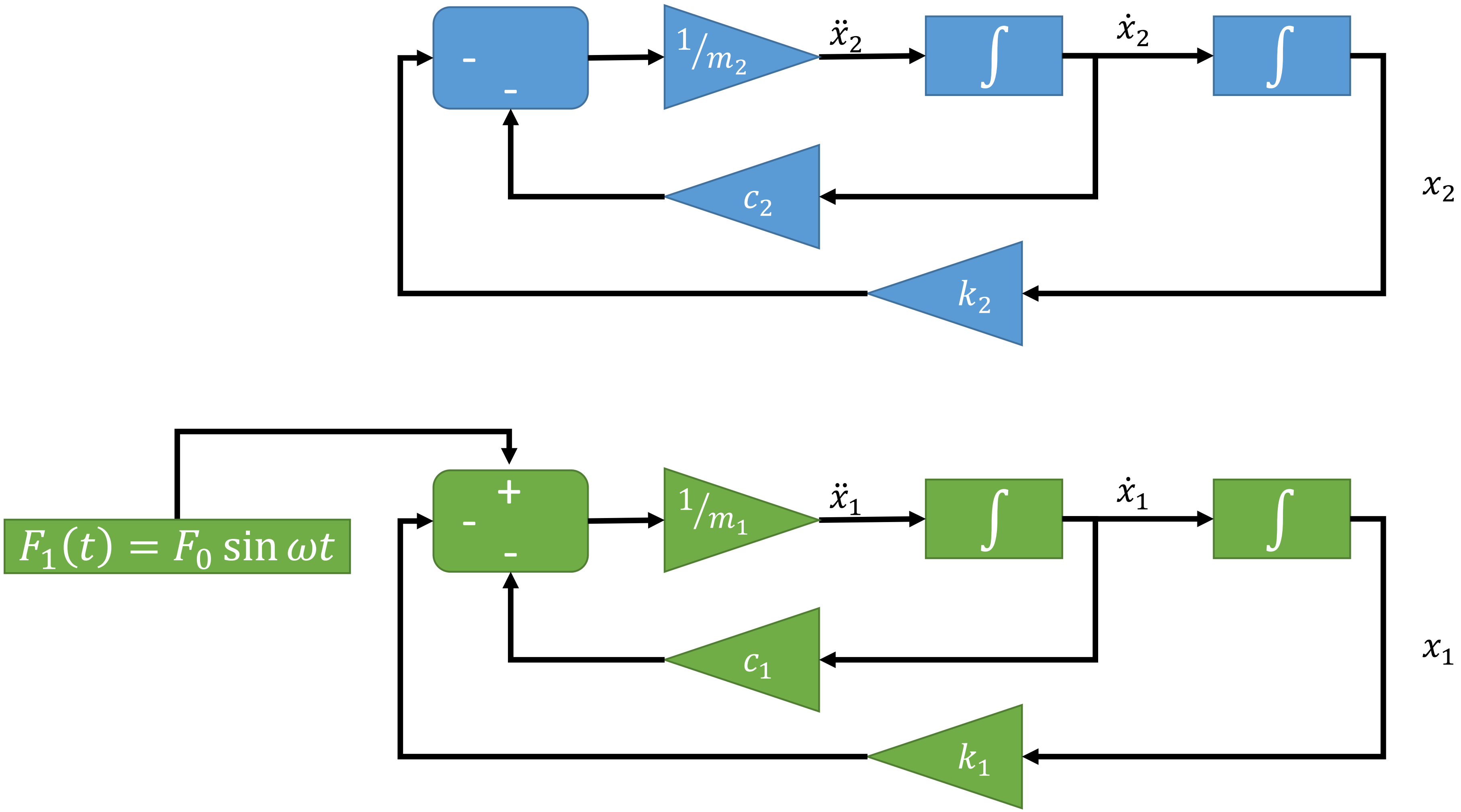
Na forma matricial

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{Bmatrix} + \begin{bmatrix} c_1 + c_2 & -c_2 \\ -c_2 & c_2 \end{bmatrix} \begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} + \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ 0 \end{Bmatrix}$$

Ou rearranjadas isolando c e k

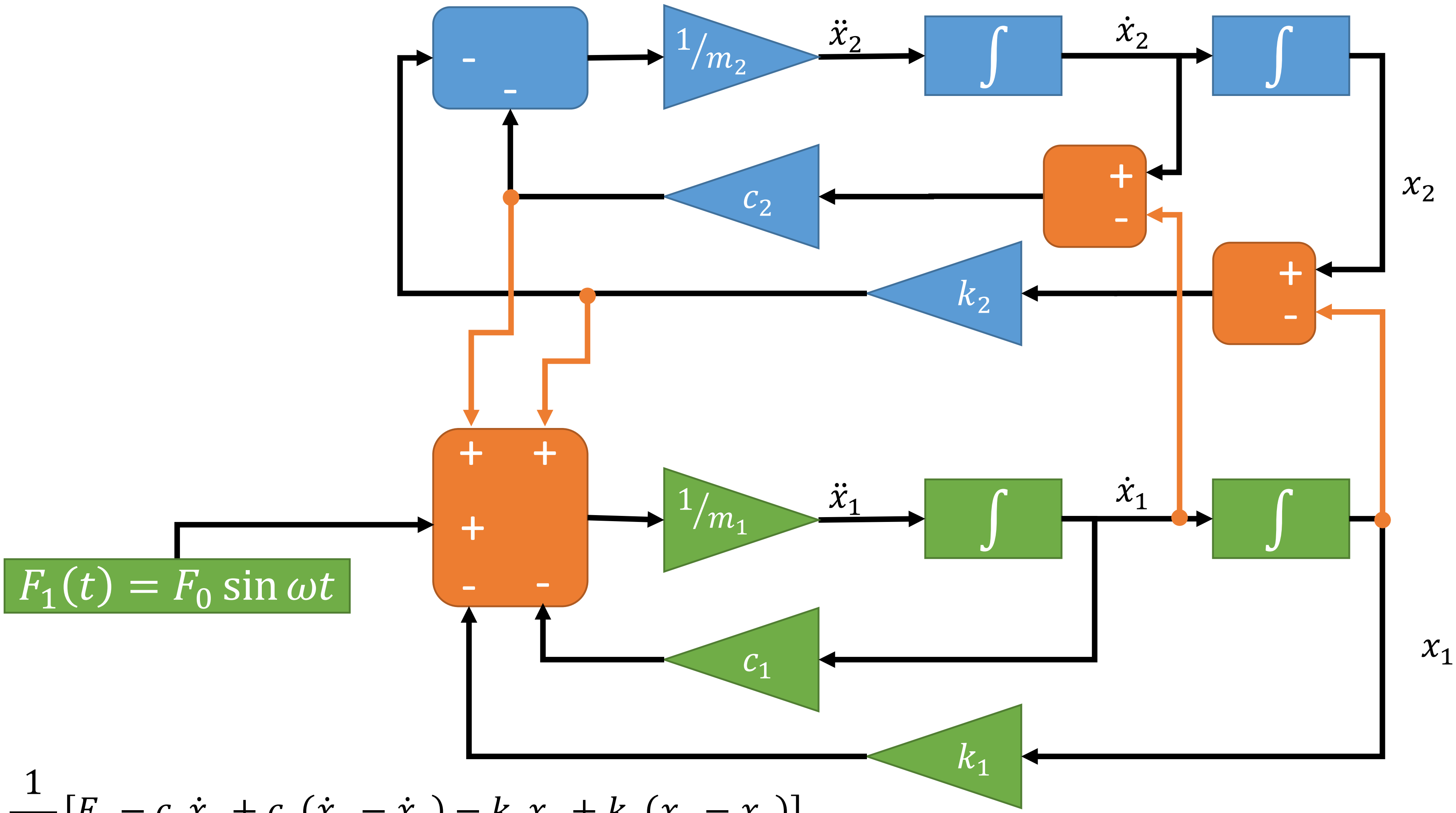
$$\ddot{x}_1 = \frac{1}{m_1}\left[F_1 - c_1\dot{x}_1 + \underline{c_2(\dot{x}_2 - \dot{x}_1)} - k_1x_1 + \underline{k_2(x_2 - x_1)}\right]$$

$$\ddot{x}_2 = \frac{1}{m_2}\left[\underline{-c_2(\dot{x}_2 - \dot{x}_1)} - \underline{k_2(x_2 - x_1)}\right]$$

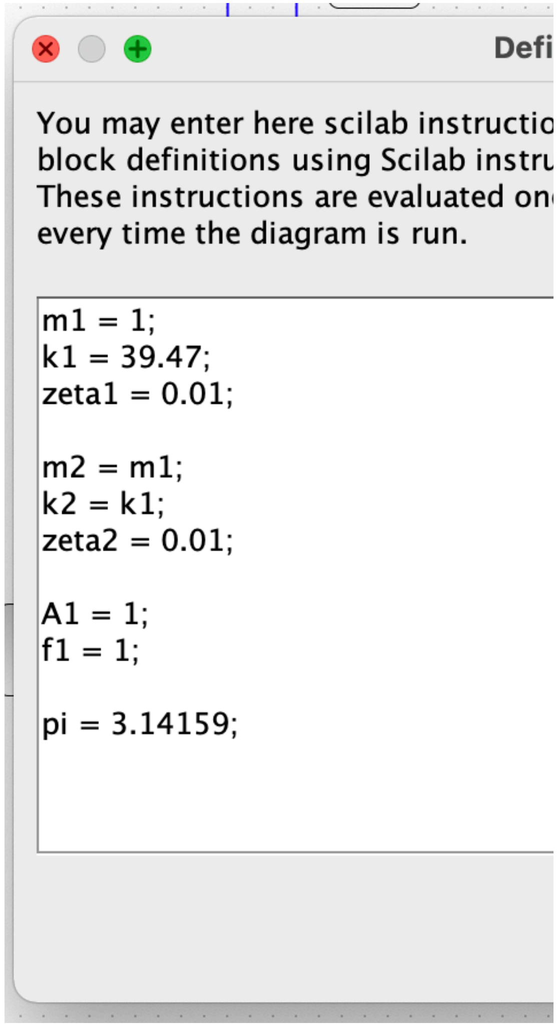


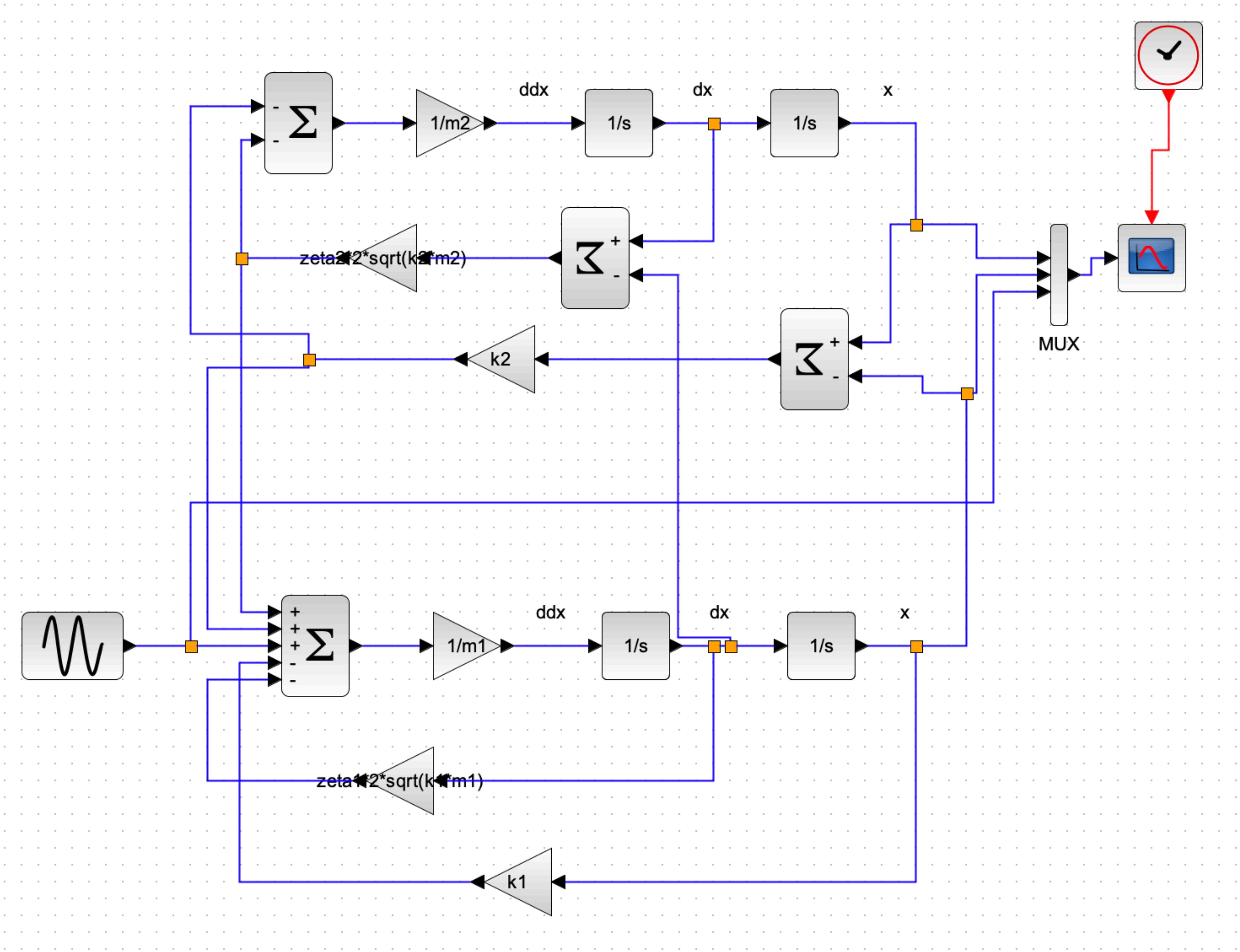
-
-
$1/m_2$
$\ddot{x}_2$
$\int$
$\dot{x}_2$
$\int$
x_2
c_2
k_2
$F_1(t) = F_0 \sin \omega t$
+
-
-
$1/m_1$
$\ddot{x}_1$
$\int$
$\dot{x}_1$
$\int$
x_1
c_1
k_1

$$\ddot{x}_2 = \frac{1}{m_2}[-c_2(\dot{x}_2 - \dot{x}_1) - k_2(x_2 - x_1)]$$



$$\ddot{x}_1 = \frac{1}{m_1}[F_1 - c_1\dot{x}_1 + c_2(\dot{x}_2 - \dot{x}_1) - k_1x_1 + k_2(x_2 - x_1)]$$

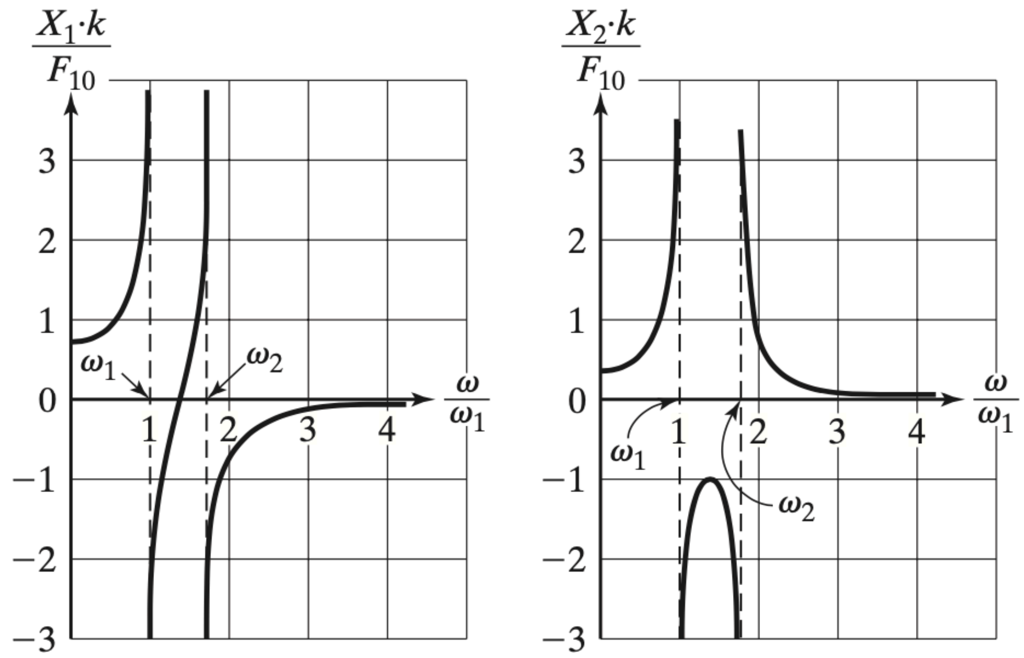




mck_aula21_b.zcos

Exercício de simulação

Implemente a simulação no modo BATCH para produzir as curvas de resposta em função de Zeta e Omega.



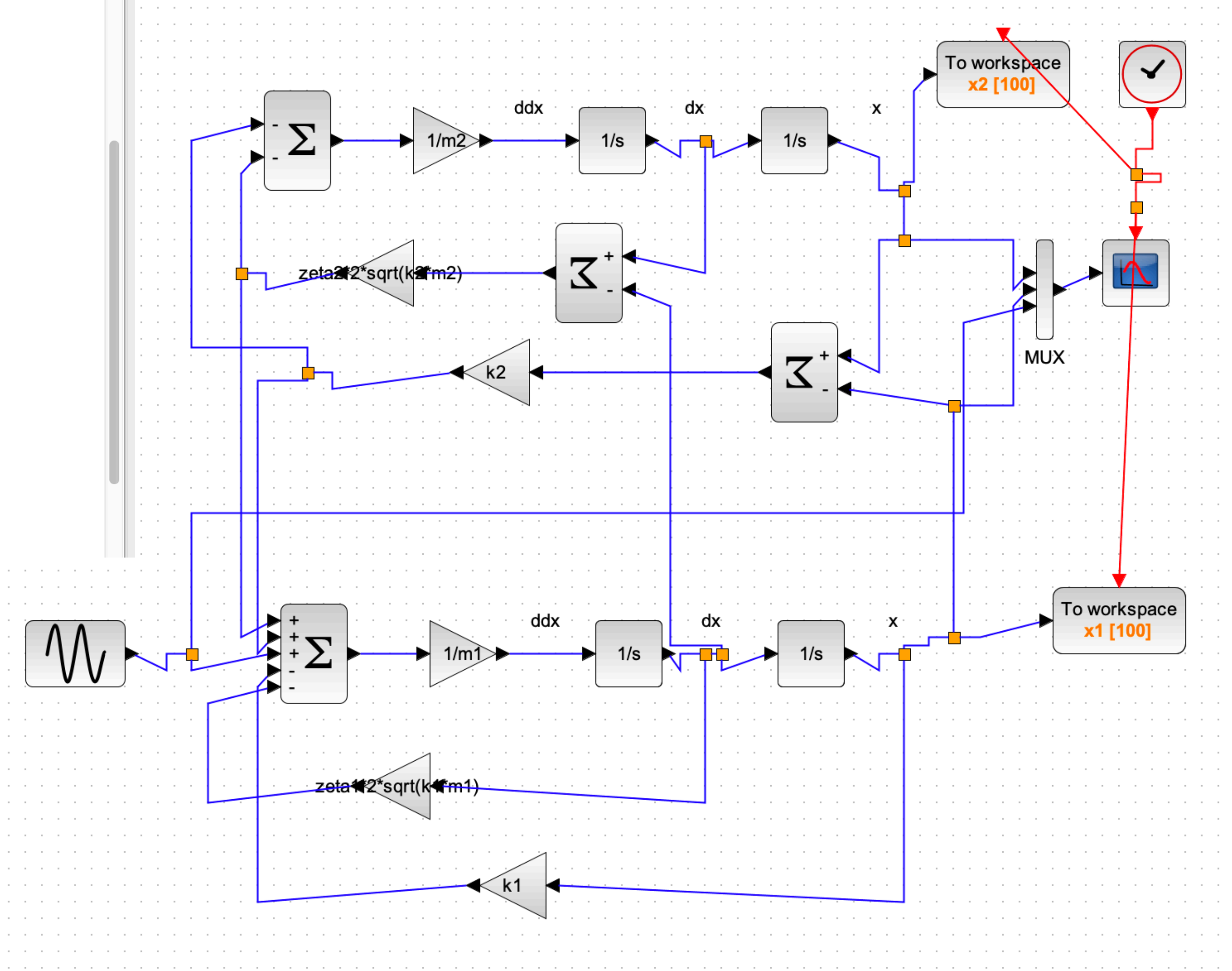
```
A1 = 1;

// Define o comprimento do cabo em m
zeta1 = [0.01,0.05, 0.1];

// Atribua as constantes ao contexto
//Context.Tf=100;
Context.A1 = A1;

// Loop de T
for m = 1:size(zeta1,2);
    
    // Loop de lc
    for n = 1:size(fr1,2);
        
        // Atribui as variáveis ao contexto
        Context.zeta1 = zeta1(m);
        Context.f1 = fr1(n);
        
        // Executa a simulação com o contexto
        scicos_simulate(scs_m,Context);
        
        // Recebe as variáveis de resposta enviadas ao "workspace"
        //t(:,n) = x1.time;
        //x1(:,n) = x1.values;
        //x2(:,n) = x2.values;
        
        // Calcula a amplitude da resposta a partir do RMS do sinal
        // multiplicado por raiz de 2 (amplitude harmônica).
        Ax1(m,n) = sqrt(2)*sqrt(sum(x1.values.*conj(x1.values))/size(x1.values,1));
        Ax2(m,n) = sqrt(2)*sqrt(sum(x2.values.*conj(x2.values))/size(x2.values,1));
        
    end
end

// Calcula o fator de amplificação
//AMP = Ax./Ay;
```



mck_aula21_batch.zcos, SIM_batch_aula21.sce