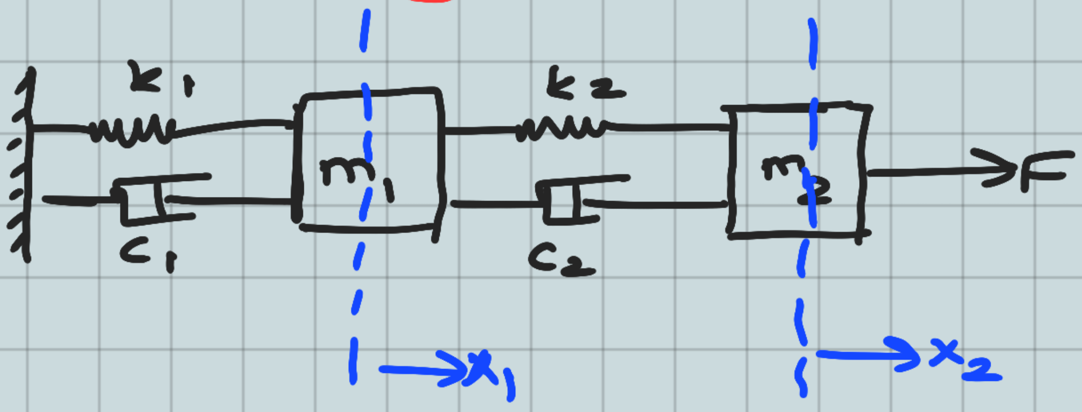
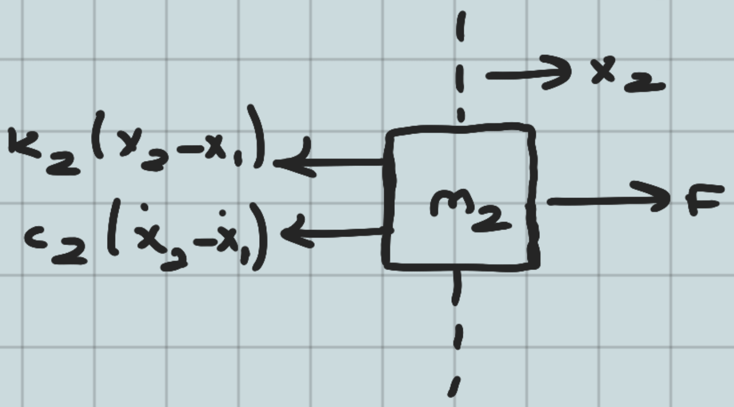
Mass-spring-damper modeling ⇒

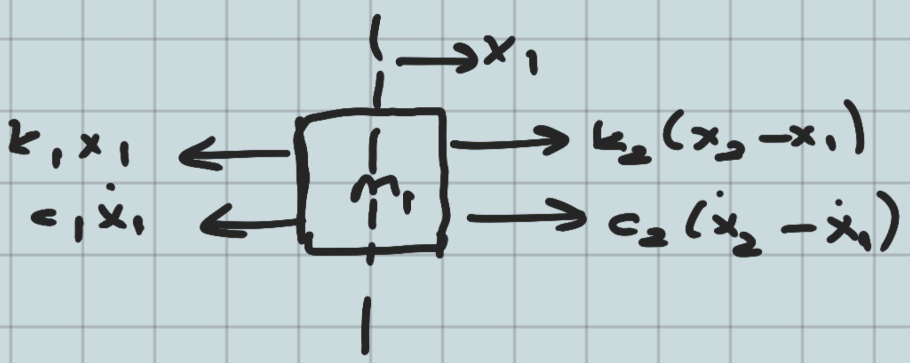


* for mass 2 m_2;



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

* for mass 1 m_1;



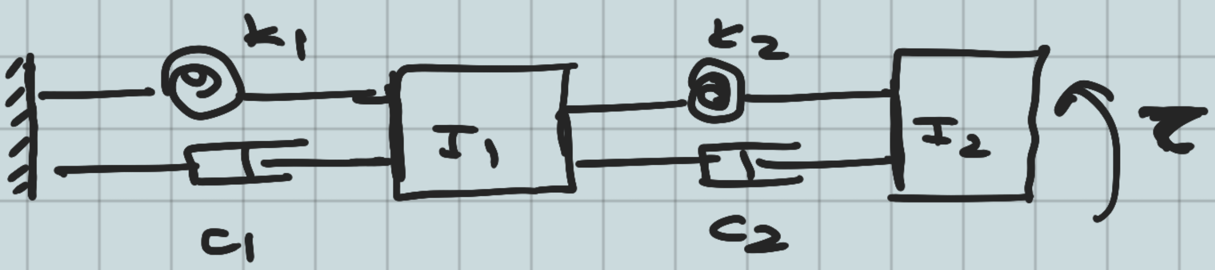
$$m_1\ddot{x}_1 = k_2(x_2 - x_1) + c_2(\dot{x}_2 - \dot{x}_1) - k_1 x_1 - c_1\dot{x}_1$$

for our case ⇒ * $F \rightarrow$ Torque

* $x \rightarrow \theta$

* $m \rightarrow I$

Hence; using torsional spring & damping



equations become;

(I) $I_2 \ddot{\theta}_2 = \tau - k_2(\theta_2 - \theta_1) - c_2(\dot{\theta}_2 - \dot{\theta}_1)$

(II) $I_1 \ddot{\theta}_1 = k_2(\theta_2 - \theta_1) + c_2(\dot{\theta}_2 - \dot{\theta}_1) - k_1\theta_1 - c_1\dot{\theta}_1$

leaving $\ddot{\theta}_2$ and $\ddot{\theta}_1$ alone;

(I) $$\ddot{\theta}_2 = \frac{\tau - k_2(\theta_2 - \theta_1) - c_2(\dot{\theta}_2 - \dot{\theta}_1)}{I_2}$$

(II) $$\ddot{\theta}_1 = \frac{k_2(\theta_2 - \theta_1) + c_2(\dot{\theta}_2 - \dot{\theta}_1) - k_1\theta_1 - c_1\dot{\theta}_1}{I_1}$$

to solve (I) and (II) with ode45 on matlab; defining;

(to put these in a state-variable form)

$$y = \begin{Bmatrix} \theta_1 \\ \dot{\theta}_1 \\ \theta_2 \\ \dot{\theta}_2 \end{Bmatrix} \longrightarrow \dot{y} = \begin{Bmatrix} \dot{\theta}_1 \\ \ddot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_2 \end{Bmatrix}$$

State variable

$$\dot{y}_1 = \dot{\theta}_1 = y(2)$$

$$\dot{y}_2 = \ddot{\theta}_1 = \frac{k_2\big(y(3)-y(1)\big) + c_2\big(y(4)-y(2)\big) - k_1 y(1) - c_1 y(2)}{I_1}$$

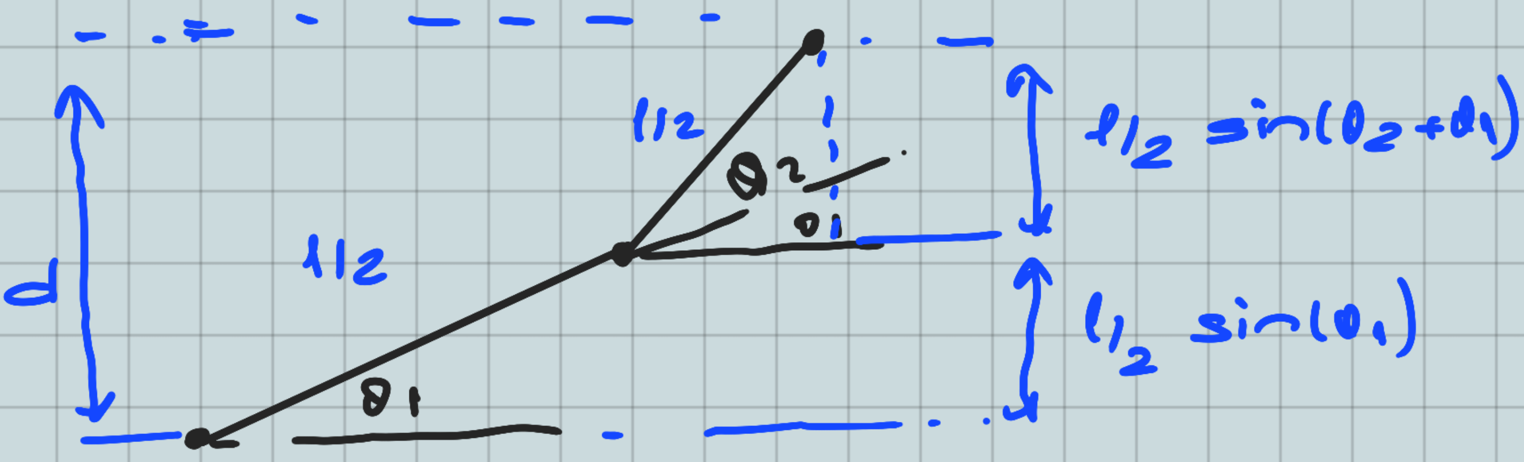
$$\dot{y}_3 = \dot{\theta}_2 = y(4)$$

$$\dot{y}_4 = \ddot{\theta}_2 = \frac{\tau - k_2\big(y(3)-y(1)\big) - c_2\big(y(4)-y(2)\big)}{I_2}$$

matlab code

```
function dy = SpringDamperMass(t,y,τ,k,c)
        y = [   ẏ1
                ẏ2
                ẏ3
                ẏ4]
end
func = @(t,y) SpringDamperMass(t,y,τ,k,c)
[t,y] = ode45(func,[0:0.001],[0 0 0 0])
```

finding the height of the edges $\Rightarrow$



$$d = \frac{l}{2}\left(\sin\theta_1 + \sin(\theta_2+\theta_1) \right)$$

— extra —

— ODEs in a state-variable form —

$$\{\dot{x}\} = A\{x\} + B$$

example $\Rightarrow$ $\ddot{x} + 7\dot{x} + 10x = 20$

$x(0) = 5$

$\dot{x}(0) = 3$

$$\{x\} = \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} x \\ \frac{dx}{dt} \end{Bmatrix}$$ $\Rightarrow$ state variable $\underbrace{}_{2\times1}$

since we have $\ddot{x}$ at most $\to \frac{d^2x}{dt^2} \to \ddot{x}$

(I)

$$\Rightarrow \dot{x}_1 = \frac{dx_1}{dt} = \frac{dx}{dt}$$

(II)

$$\Rightarrow \dot{x}_2 = \frac{dx_2}{dt} = \frac{d}{dt}\left(\frac{dx}{dt}\right) = \frac{d^2x}{dt^2}$$

(I) $\dot{x}_1 = \frac{dx}{dt} = \dot{x} = x_2$

(II) $\dot{x}_2 = 20 - 7\dot{x} - 10x$

$$= 20 - 7x_2 - 10x_1$$

— extra —

matlab code

$y \sim x$

```
→ function dy = SecondOrder(t, y)
      dy = [  y(2)
              20 - 7y(2) - 10y(1) ]
  end

→ func = @(t,y) Secondorder(t,y)
  [t,y] = ode45(func, [0 5], [5 3])
```

$\underbrace{[5\ 3]}_{\text{initial cond.}}$

```
→ plot(t, y(:,1))
```