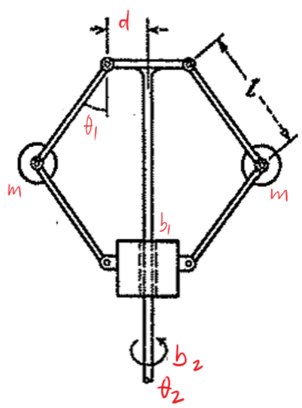



```
> restart
> with(LinearAlgebra):
  with(Physics):
  with(plots):
  with(plottools):
  with(DEtools):
  with(Typesetting):
  interface(typesetting=extended):
  interface(showassumed=0):

> T1:=2*1/2*(J+m*l^2)*diff(theta__1(t),t)^2:
  T2:=2*1/2*(J+m*(l*sin(theta__1(t))+d)^2)*diff(theta__2(t),t)^2:
  V:=2*m*g*l*(1-cos(theta__1(t))):
  R1:=1/2*b__1*diff(l-l*cos(theta__1(t)),t)^2:
  R2:=1/2*b__2*diff(theta__2(t),t)^2:

  printf("\n");

  print(`Kinetic energy of the bar:`);
  print('T'=T1+T2);

  printf("\n");

  print(`Potential energy of the bar:`);
  print('V'=V);

  printf("\n");

  print(`Translational dissipation along the bar:`);
  print('R__1'=R1);

  printf("\n");

  print(`Rotational dissipation disspation:`);
  print('R__2'=R2);
```

Kinetic energy of the bar:

$$T = \left(m\,l^2 + J\right)\,\dot{\theta}_1(t)^2 + \left(J + m\left(l\sin\left(\theta_1(t)\right) + d\right)^2\right)\,\dot{\theta}_2(t)^2$$

Potential energy of the bar:

$$V = 2\,m\,g\,l\left(1 - \cos\left(\theta_1(t)\right)\right)$$

Translational dissipation along the bar:

$$R_1 = \frac{b_1 l^2 \dot{\theta}_1(t)^2 \sin(\theta_1(t))^2}{2}$$

Rotational dissipation disspation:

$$R_2 = \frac{b_2 \dot{\theta}_2(t)^2}{2}$$

```
> T:=T1+T2:
  V:=V:
  L:=T-V:
  print('L'=L);

  eq1:=simplify(<diff(L,theta__1(t)), diff(L,theta__2(t))>):
  eq2:=simplify(<diff(L,diff(theta__1(t),t)), diff(L,diff(theta__2(t),t))>):
  eq3:=simplify(<diff(eq2[1],t), diff(eq2[2],t)>):
  eqR1:=simplify(<diff(R1,diff(theta__1(t),t)), diff(R1,diff(theta__2(t),t))>):
  eqR2:=simplify(<diff(R2,diff(theta__1(t),t)), diff(R2,diff(theta__2(t),t))>):

  print('diff(L,theta)'=eq1);
  print('diff(L,diff(theta,t))'=eq2);
  print('diff(R__1,diff(theta,t))'=eqR1);
  print('diff(R__2,diff(theta,t))'=eqR2);
  print('diff(diff(L,diff(theta,t)),t)'=simplify(eq3));
```

$$L = (m l^2 + J)\, \dot{\theta}_1(t)^2 + \left(J + m\,(l \sin(\theta_1(t)) + d)^2\right) \dot{\theta}_2(t)^2 - 2\, m\, g\, l\, (1 - \cos(\theta_1(t)))$$

$$\frac{\partial}{\partial \theta} L = \begin{bmatrix} 2\, l \left(\cos(\theta_1(t))\, (l \sin(\theta_1(t)) + d)\, \dot{\theta}_2(t)^2 - g \sin(\theta_1(t))\right) m \\ 0 \end{bmatrix}$$

$$\frac{\partial}{\partial \frac{\partial}{\partial t}\theta} L = \begin{bmatrix} 2\,(m l^2 + J)\, \dot{\theta}_1(t) \\ 2\left(J + m\,(l \sin(\theta_1(t)) + d)^2\right) \dot{\theta}_2(t) \end{bmatrix}$$

$$\frac{\partial}{\partial \frac{\partial}{\partial t}\theta} R_1 = \begin{bmatrix} b_1 l^2 \dot{\theta}_1(t) \sin(\theta_1(t))^2 \\ 0 \end{bmatrix}$$

$$\frac{\partial}{\partial \frac{\partial}{\partial t}\theta} R_2 = \begin{bmatrix} 0 \\ b_2 \dot{\theta}_2(t) \end{bmatrix}$$

$$\frac{\partial^2}{\partial \frac{\partial}{\partial t}\theta \partial t} L = \left[\left[2\,(m l^2 + J)\, \ddot{\theta}_1(t)\right],\right.$$

$$\left[\left(-2 \cos(\theta_1(t))^2 l^2 m + 4 \sin(\theta_1(t))\, d\, l\, m + (2 d^2 + 2 l^2)\, m + 2 J\right) \ddot{\theta}_2(t) + 4\, m\, (l \sin(\theta_1(t))\right.$$

$$+ d) \, l \, \dot{\theta}_1(t) \cos(\theta_1(t)) \, \dot{\theta}_2(t) \Big] \Big]$$

```
> printf("\n");
print(`Thus,`);
print('diff(diff(L,diff(theta,t)),t)-diff(L,theta)+diff(R__1,diff(theta(t),t))+diff
(R__2,diff(theta(t),t))'=<0, tau(t)>);
eq4:=simplify({(eq3[1]-eq1[1]+eqR1[1]+eqR2[1]=0), (eq3[2]-eq1[2]+eqR1[2]+eqR2[2]=tau(t)
)}):
print(<eq4[1],eq4[2]>);
eq5:=simplify(solve(eq4,{diff(theta__1(t),t,t), diff(theta__2(t),t,t)})):
print(<eq5[1],eq5[2]>);
```

Thus,

$$\frac{\partial^2}{\partial \frac{\partial}{\partial t} \theta \partial t} L - \frac{\partial}{\partial \theta} L + \frac{\partial}{\partial \dot{\theta}(t)} R_1 + \frac{\partial}{\partial \dot{\theta}(t)} R_2 = \begin{bmatrix} 0 \\ \tau(t) \end{bmatrix}$$

$$\Bigg[\Bigg[\left(-2 \cos(\theta_1(t))^2 l^2 m + 4 \sin(\theta_1(t)) \, d \, l \, m + (2 d^2 + 2 l^2) \, m + 2 J\right) \ddot{\theta}_2(t) + 4 \Big(m \, \big(l \sin(\theta_1(t)) + d \big) \, l \, \dot{\theta}_1(t) \cos(\theta_1(t)) + \frac{b_2}{4} \Big) \dot{\theta}_2(t) = \tau(t) \Bigg],$$

$$\Big[(2 m l^2 + 2 J) \, \ddot{\theta}_1(t) - \Big(l \, b_1 \, (\cos(\theta_1(t)) - 1) \, (\cos(\theta_1(t)) + 1) \, \dot{\theta}_1(t) + 2 m \, \Big(\cos(\theta_1(t)) \, \big(l \sin(\theta_1(t)) + d \big) \, \dot{\theta}_2(t)^2 - g \sin(\theta_1(t)) \Big) \Big) \, l = 0 \Big] \Bigg]$$

$$\Bigg[\Bigg[\ddot{\theta}_1(t) = \frac{1}{2 m l^2 + 2 J} \Big(\Big(l \, b_1 \, (\cos(\theta_1(t)) - 1) \, (\cos(\theta_1(t)) + 1) \, \dot{\theta}_1(t) + 2 m \, \Big(\cos(\theta_1(t)) \, \big(l \sin(\theta_1(t)) + d \big) \, \dot{\theta}_2(t)^2 - g \sin(\theta_1(t)) \Big) \Big) \, l \Big) \Bigg],$$

$$\Bigg[\ddot{\theta}_2(t) = \frac{\left(-4 m \, \big(l \sin(\theta_1(t)) + d \big) \, l \, \dot{\theta}_1(t) \cos(\theta_1(t)) - b_2\right) \dot{\theta}_2(t) + \tau(t)}{-2 \cos(\theta_1(t))^2 l^2 m + 4 \sin(\theta_1(t)) \, d \, l \, m + (2 d^2 + 2 l^2) \, m + 2 J} \Bigg] \Bigg]$$

```
> m:=0.1:
J:=0.01:
l:=0.2:
g:=9.8:

b__1:=5:
b__2:=0.1:
d:=0.02:
RPM:=100*0.10472: #converting RPM to rad/s
```

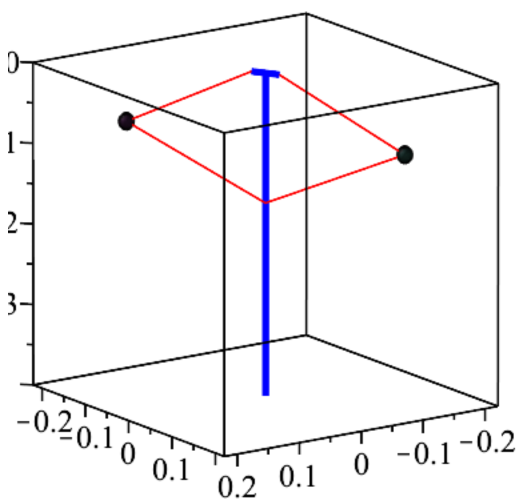
```
> ode:=eval(eq5[1]):
ode:=subs([diff(theta__2(t),t)=RPM, diff(theta__2(t),t,t)=0],ode):
```

```
> sol:=dsolve({ode, theta__1(0)=0, D(theta__1)(0)=0}, numeric, method=rkf45, output=
listprocedure):
```

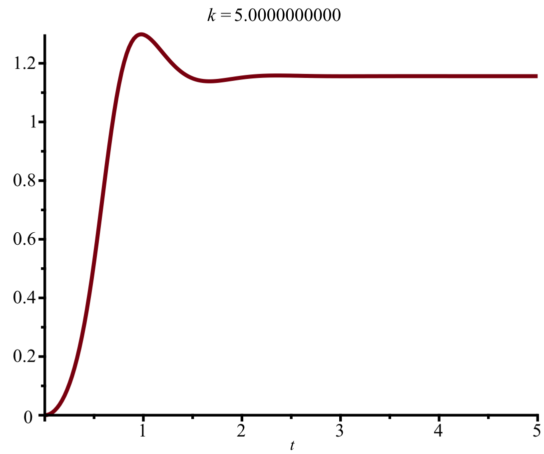
```
th1_array:=eval(theta__1(t),sol):
th2_array(t):=RPM*t:
```

```
> A:=[0,0,0]:
A1:=[-d*cos(th2_array(t)),-d*sin(th2_array(t)),0]:
A2:=[ d*cos(th2_array(t)), d*sin(th2_array(t)),0]:
B:=[-(l*sin(th1_array(t))+d)*cos(th2_array(t)), -(l*sin(th1_array(t))+d)*sin(th2_array
(t)), -l*cos(th1_array(t))]:
C:=[ (l*sin(th1_array(t))+d)*cos(th2_array(t)),  (l*sin(th1_array(t))+d)*sin(th2_array
(t)), -l*cos(th1_array(t))]:
E:=[ 0,0, -2*l]:
F:=[ 0,0, -l*cos(th1_array(t))*2]:
anim1:=animate(line, [A1, B, color=red, scaling=constrained], t=0..5,frames=100):
anim2:=animate(line, [A2, C, color=red, scaling=constrained], t=0..5,frames=100):
anim3:=animate(line, [A, E, color=blue,thickness=5, scaling=constrained], t=0..5,
frames=100):
anim4:=animate(line, [A1, A2, color=blue,thickness=5, scaling=constrained], t=0..5,
frames=100):
anim5:=animate(sphere, [B, 0.01,scaling=constrained], t=0..5,frames=100):
anim6:=animate(sphere, [C, 0.01,scaling=constrained], t=0..5,frames=100):
anim7:=animate(line, [B, F, color=red, scaling=constrained], t=0..5,frames=100):
anim8:=animate(line, [C, F, color=red, scaling=constrained], t=0..5,frames=100):



h1:=display(anim1, anim2, anim3, anim4, anim5, anim6, anim7, anim8);
```

$t = 5.0000000000$




```
> animate(plot,[th1_array(t),t=0..k],k=0..5)
```



k = 5.0000000000
1.2
1
0.8
0.6
0.4
0.2
0
1
2
3
4
5
t