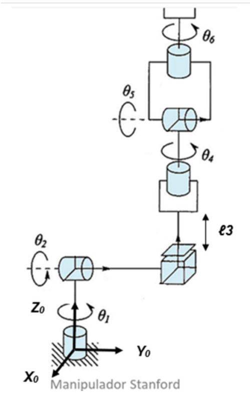


```
%Limpieza de pantalla
clear all
close all
clc
```

Hablamos de un robot manipulador que consiste en que cuenta con un eslabón que rota al rededor de un eje. A continuación se define las variables que usaremos a lo largo del programa, como los ángulos, velocidades, aceletaciones de las articulaciones así como la masa y matriz de inercia. Longitud del eslabon y la distancia al centro de masa que será muy importante para nuestro análisis de la altura y la energía potencial.

```
tic
%DECLARACIÓN DE VARIABLES.
%Declaración de variables simbólicas
syms th1(t)  th2(t) l3(t)  th4(t) th5(t) th6(t) t   %Angulos de cada articulación
syms th1p(t) th2p(t) l3p(t) th4p(t) th5p(t) th6p(t)  %Velocidades de cada
articulación
syms th1pp(t) th2pp(t) l3pp(t) th4pp(t) th5pp(t) th6pp(t)%Aceleraciones de cada
articulación
syms m1 Ixx1 Iyy1 Izz1 m2  Ixx2 Iyy2 Izz2 m3 Ixx3 Iyy3 Izz3  m4  Ixx4 Iyy4 Izz4 m5
Ixx5 Iyy5 Izz5 m6 Ixx6 Iyy6 Izz6  %Masas y matrices de Inercia
syms l1 lc1 l2 lc2 lo3 lc3 l4 lc4 l5 lc5 l6 lc6 %l=longitud de eslabones y
lc=distancia al centro de masa de cada eslabón
syms pi g a cero

 %Creamos el vector de coordenadas articulares
  Q= [th1; th2; l3; th4; th5; th6];
 %disp('Coordenadas generalizadas');
 %pretty (Q);
```

Se crea el velor de coordenadas articulaes del sistema, las cuales explican la posición de las articulaciones del robot. Tambien las velocidades articuales que pueden representar una velocidad angular y por ultimo las aceleraciones articuales para explicar la aceleración angular.

```
 %Creamos el vector de velocidades articulares
  Qp= [th1p; th2p; l3p; th4p; th5p; th6p];
```

```
 %disp('Velocidades generalizadas');
 %pretty (Qp);
 %Creamos el vector de aceleraciones articulares
  Qpp= [th1pp; th2pp; l3pp; th4pp; th5pp; th6pp];
 %disp('Aceleraciones generalizadas');
 %pretty (Qpp);
```

Tenemos la configuración del robot donde gracias al dibujo se sabe de que trata la junta rotacional. RP representa el número de grados de libertad.

```
%Configuración del robot, 0 para junta rotacional, 1 para junta prismática
RP=[0 0 1 0 0 0];

%Número de grado de libertad del robot
GDL= size(RP,2);
GDL_str= num2str(GDL);
```

Para esto, hacemos el análisis de cada articulación del robot, se declara la posición.

```
%Articulación 1
%Posición de la articulación 1 respecto a 0
P(:,:,1)= [0; 0; l1];
%Matriz de rotación de la junta 1 respecto a 0....
R(:,:,1)= [cos(th1) 0 -sin(th1);
           sin(th1) 0 cos(th1);
           0        -1      0];

%Articulación 2
%Posición de la articulación 2 respecto a 1
P(:,:,2)= [l2*cos(th2); l2*sin(th2);0];
%Matriz de rotación de la junta 1 respecto a 0
R(:,:,2)= [cos(th2)  0  -sin(th2);
           sin(th2)  0  -cos(th2);
           0         1         0];

%Articulación 3
%Posición de la articulación 2 respecto a 1
P(:,:,3)= [0;0;lo3];
%Matriz de rotación de la junta 1 respecto a 0
R(:,:,3)= [1 0 0 ;
           0 1 0 ;
           0 0 1];

%Articulación 4
%Posición de la articulación 2 respecto a 1
P(:,:,4)= [0; 0; l4];
%Matriz de rotación de la junta 1 respecto a 0
R(:,:,4)= [cos(th4) 0 -sin(th4);
           sin(th4) 0  cos(th4);
           0       -1        0];
```

```
%Articulación 5
%Posición de la articulación 2 respecto a 1
P(:,:,5)= [l5*cos(th5); l5*sin(th5);0];
%Matriz de rotación de la junta 1 respecto a 0
R(:,:,5)= [cos(th5)  0  -sin(th5);
           sin(th5)  0  -cos(th5);
           0         1         0];

%Articulación 6
%Posición de la articulación 2 respecto a 1
P(:,:,6)= [0; 0; l6];
%Matriz de rotación de la junta 1 respecto a 0
R(:,:,6)= [cos(th6) -sin(th6)  0;
           sin(th6)  cos(th6)  0;
             0         0      1];


%Creamos un vector de ceros
Vector_Zeros= zeros(1, 3);
```

Se inician las matrices de transformación hómogenea local y global donde se asigna la letra A en la ultima posición GDL de estas matrices tridimensionales que tiene de concatenación a la matriz de rotación, la matriz de posición y una fila de ceros seguido de un 1 para indicar toda la matriz de transformación.

```
%Inicializamos las matrices de transformación Homogénea locales
A(:,:,GDL)=simplify([R(:,:,GDL) P(:,:,GDL); Vector_Zeros 1]);
%Inicializamos las matrices de transformación Homogénea globales
T(:,:,GDL)=simplify([R(:,:,GDL) P(:,:,GDL); Vector_Zeros 1]);
```

Se representan los vectores de posición donde se copia la matriz de posición y son vistas desde el marco de referencia inercial, también las matrices de rotación.

```
%Inicializamos las posiciones vistas desde el marco de referencia inercial
PO(:,:,GDL)= P(:,:,GDL);
%Inicializamos las matrices de rotación vistas desde el marco de referencia inercial
RO(:,:,GDL)= R(:,:,GDL);
```

Para comenzar con las matrices de transformación se tiene un ciclo que itera sobre cada grado de libertad, calculando la matriz de transofrmación homogenea local, concatenando la rotación, posiicón y un vector de zeros. Con el uso try y catch se empieza a formar la matriz de transformación global, con el catch se asigna un matriz local a la global quedando la misma para los 2.

```
for i = 1:GDL
    i_str= num2str(i);
   %disp(strcat('Matriz de Transformación local A', i_str));
    A(:,:,i)=simplify([R(:,:,i) P(:,:,i); Vector_Zeros 1]);
   %pretty (A(:,:,i));
```

```
    %Globales
     try
        T(:,:,i)= T(:,:,i-1)*A(:,:,i);
     catch
        T(:,:,i)= A(:,:,i);
     end
%     disp(strcat('Matriz de Transformación global T', i_str));
    T(:,:,i)= simplify(T(:,:,i));
%     pretty(T(:,:,i))

    RO(:,:,i)= T(1:3,1:3,i);
    PO(:,:,i)= T(1:3,4,i);
    %pretty(RO(:,:,i));
    %pretty(PO(:,:,i));
end
%%Velocidades para cada eslabón
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%CALCULAMOS LAS VELOCIDADES PARA CADA ESLABÓN%%%%%%%
%%%%%%%%%%%%%%%%%%%%
```

Se define el cálculo del jacobiano lineal de forma analítica donde primeramente se asigna la posición final del extremo del robot al jacobiano angular y lineal. De ahí va el bucle en el que primero se comprueba si la articulación es de junta rotacional o prismática,

```
%%%%%%%%%%  VELOCIDADES PARA ESLABÓN 6  %%%%%%%%%%

%Calculamos el jacobiano lineal de forma analítica
Jv_a6(:,GDL)=PO(:,:,GDL);
Jw_a6(:,GDL)=PO(:,:,GDL);

for k= 1:GDL
    if RP(k)==0
       %Para las juntas de revolución
        try
            Jv_a6(:,k)= cross(RO(:,3,k-1), PO(:,:,GDL)-PO(:,:,k-1));
            Jw_a6(:,k)= RO(:,3,k-1);
        catch
            Jv_a6(:,k)= cross([0,0,1], PO(:,:,GDL));%Matriz de rotación de 0 con
respecto a 0 es la Matriz Identidad, la posición previa tambien será 0
            Jw_a6(:,k)=[0,0,1];%Si no hay matriz de rotación previa se obtiene la
Matriz identidad
         end
     else
%         %Para las juntas prismáticas
        try
            Jv_a6(:,k)= RO(:,3,k-1);
        catch
            Jv_a6(:,k)=[0,0,1];
        end
            Jw_a6(:,k)=[0,0,0];
     end
```

```
end

%Obtenemos SubMatrices de Jacobianos
Jv_a6= simplify (Jv_a6);
Jw_a6= simplify (Jw_a6);
%disp('Jacobiano lineal obtenido de forma analítica');
%pretty (Jv_a);
%disp('Jacobiano ángular obtenido de forma analítica');
%pretty (Jw_a);

%Matriz de Jacobiano Completa
%disp('Matriz de Jacobiano');
Jac6= [Jv_a6;
      Jw_a6];
Jacobiano6= simplify(Jac6);
% pretty(Jacobiano);

%Obtenemos vectores de Velocidades Lineales y Angulares para el eslabón 2
 disp('Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 6');
```

```
Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 6
```

```
V6=simplify (Jv_a6*Qp);
pretty(V6)
```

```
/                                                         th5p(t) ((#7 - #4) (#20 - #19 - #17 + #18) - sin(th2(t))
|
|                                                         th5p(t) ((#6 + #5) (#20 - #19 - #17 + #18) + sin(th2(t))
|
\ th2p(t) (#9 - #10 + #8 + l6 cos(th5(t)) sin(th2(t)) - l5 sin(th2(t)) sin(th5(t)) - l5 cos(th2(t)) cos(th4(t)) cos(

where

   #1 == l5 cos(th5(t)) - l6 sin(th5(t))

   #2 == l5 sin(th1(t)) sin(th2(t)) sin(th5(t))

   #3 == l5 cos(th1(t)) sin(th2(t)) sin(th5(t))

   #4 == cos(th2(t)) sin(th1(t)) sin(th4(t))

   #5 == cos(th1(t)) cos(th2(t)) sin(th4(t))

   #6 == cos(th4(t)) sin(th1(t))

   #7 == cos(th1(t)) cos(th4(t))

   #8 == lo3 sin(th2(t))

   #9 == l4 sin(th2(t))

   #10 == l2 cos(th2(t))

                        2
   #11 == cos(th2(t))

   #12 == l4 cos(th2(t)) + lo3 cos(th2(t)) - l2 sin(th2(t)) + #20 - #19 - #17 + #18
```

```
#13 == l6 (sin(th5(t)) #22 + cos(th5(t)) sin(th1(t)) sin(th2(t)))

#14 == l6 (sin(th5(t)) #21 - cos(th1(t)) cos(th5(t)) sin(th2(t)))

#15 == l5 cos(th5(t)) #21

#16 == l5 cos(th5(t)) #22

#17 == l5 cos(th4(t)) cos(th5(t)) sin(th2(t))

#18 == l6 cos(th4(t)) sin(th2(t)) sin(th5(t))

#19 == l5 cos(th2(t)) sin(th5(t))

#20 == l6 cos(th2(t)) cos(th5(t))

#21 == sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(t))

#22 == cos(th1(t)) sin(th4(t)) + cos(th2(t)) cos(th4(t)) sin(th1(t))
```

```
disp('Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 6');
```

```
Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 6
```

```
W6=simplify (Jw_a6*Qp);
pretty(W6)
```

```
/ th6p(t) (sin(th5(t)) (sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(t))) - cos(th1(t)) cos(th5(t)) sin
|
| th5p(t) (cos(th1(t)) cos(th4(t)) - cos(th2(t)) sin(th1(t)) sin(th4(t))) - th6p(t) (sin(th5(t)) (cos(th1(t)) sin(th
|
\                                                   th1p(t) + th6p(t) (cos(th2(t)) cos(th5(t)) + cos(th4(t)
```

```
%%%%%%%%%%  VELOCIDADES PARA ESLABÓN 5  %%%%%%%%%%

%Calculamos el jacobiano lineal y angular de forma analítica
Jv_a5(:,GDL-1)=PO(:,:,GDL-1);
Jw_a5(:,GDL-1)=PO(:,:,GDL-1);

for k= 1:GDL-1
    if RP(k)==0
       %Para las juntas de revolución
        try
            Jv_a5(:,k)= cross(RO(:,3,k-1), PO(:,:,GDL-1)-PO(:,:,k-1));
            Jw_a5(:,k)= RO(:,3,k-1);
        catch
            Jv_a5(:,k)= cross([0,0,1], PO(:,:,GDL-1));%Matriz de rotación de 0 con
respecto a 0 es la Matriz Identidad, la posición previa tambien será 0
            Jw_a5(:,k)=[0,0,1];%Si no hay matriz de rotación previa se obtiene la
Matriz identidad
         end
     else
%         %Para las juntas prismáticas
```

```
        try
            Jv_a5(:,k)= RO(:,3,k-1);
        catch
            Jv_a5(:,k)=[0,0,1];
        end
            Jw_a5(:,k)=[0,0,0];
     end
 end

%Obtenemos SubMatrices de Jacobianos
Jv_a5= simplify (Jv_a5);
Jw_a5= simplify (Jw_a5);
%disp('Jacobiano lineal obtenido de forma analítica');
%pretty (Jv_a);
%disp('Jacobiano ángular obtenido de forma analítica');
%pretty (Jw_a);

%Matriz de Jacobiano Completa
%disp('Matriz de Jacobiano');
Jac5= [Jv_a5;
      Jw_a5];
Jacobiano5= simplify(Jac5);
% pretty(Jacobiano);

%Obtenemos vectores de Velocidades Lineales y Angulares para el eslabón 2
disp('Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 5');
```

```
Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 5
```

```
Qp=Qp(t);
V5=simplify (Jv_a5*Qp(1:5));
pretty(V5)
```

```
/ - th1p(t) (l5 cos(th5(t)) (cos(th1(t)) sin(th4(t)) + cos(th2(t)) cos(th4(t)) sin(th1(t))) - sin(th1(t)) (#4 - #5 -
|
| - th1p(t) (cos(th1(t)) (#4 - #5 + #3) + l5 cos(th5(t)) (sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(
|
\

where

   #1 == l2 sin(th2(t)) - lo3 cos(th2(t)) - l4 cos(th2(t)) + l5 cos(th2(t)) sin(th5(t)) + l5 cos(th4(t)) cos(th5(t)

                     2
   #2 == cos(th2(t))

   #3 == lo3 sin(th2(t))

   #4 == l4 sin(th2(t))

   #5 == l2 cos(th2(t))
```

```
 disp('Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 5');
```

```
Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 5
```

```
  W5=simplify (Jw_a5*Qp(1:5));
pretty(W5)
```

```
/ - th5p(t) (cos(th4(t)) sin(th1(t)) + cos(th1(t)) cos(th2(t)) sin(th4(t))) - sin(th1(t)) th2p(t) - cos(th1(t)) sin(
|
|  th5p(t) (cos(th1(t)) cos(th4(t)) - cos(th2(t)) sin(th1(t)) sin(th4(t))) + cos(th1(t)) th2p(t) - sin(th1(t)) sin(t
|
\                                   th1p(t) + cos(th2(t)) th4p(t) + sin(th2(t)) sin(th4(t)) th5p(t)
```

```
%%%%%%%%%%  VELOCIDADES PARA ESLABÓN 4  %%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculamos el jacobiano lineal y angular de forma analítica
Jv_a4(:,GDL-2)=PO(:,:,GDL-2);
Jw_a4(:,GDL-2)=PO(:,:,GDL-2);

for k= 1:GDL-2
    if RP(k)==0
       %Para las juntas de revolución
        try
            Jv_a4(:,k)= cross(RO(:,3,k-1), PO(:,:,GDL-2)-PO(:,:,k-1));
            Jw_a4(:,k)= RO(:,3,k-1);
        catch
            Jv_a4(:,k)= cross([0,0,1], PO(:,:,GDL-2));%Matriz de rotación de 0 con
respecto a 0 es la Matriz Identidad, la posición previa tambien será 0
            Jw_a4(:,k)=[0,0,1];%Si no hay matriz de rotación previa se obtiene la
Matriz identidad
         end
     else
%         %Para las juntas prismáticas
        try
            Jv_a4(:,k)= RO(:,3,k-1);
        catch
            Jv_a4(:,k)=[0,0,1];
        end
            Jw_a4(:,k)=[0,0,0];
     end
 end

%Obtenemos SubMatrices de Jacobianos
Jv_a4= simplify (Jv_a4);
Jw_a4= simplify (Jw_a4);
%disp('Jacobiano lineal obtenido de forma analítica');
%pretty (Jv_a);
%disp('Jacobiano ángular obtenido de forma analítica');
%pretty (Jw_a);

%Matriz de Jacobiano Completa
%disp('Matriz de Jacobiano');
```

```
Jac4= [Jv_a4;
      Jw_a4];
Jacobiano4= simplify(Jac4);
% pretty(Jacobiano);

%Obtenemos vectores de Velocidades Lineales y Angulares para el eslabón 1
 disp('Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 4');
```

```
Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 4
```

```
V4 = simplify (Jv_a4*Qp(1:4));
pretty(V4)
```

```
/ cos(th1(t)) th2p(t) #2 + sin(th1(t)) th1p(t) #1 - cos(th1(t)) sin(th2(t)) l3p(t) \
|                                                                                  |
| sin(th1(t)) th2p(t) #2 - cos(th1(t)) th1p(t) #1 - sin(th1(t)) sin(th2(t)) l3p(t) |
|                                                                                  |
\                          th2p(t) #1 + cos(th2(t)) l3p(t)                         /

where

   #1 == l4 sin(th2(t)) - l2 cos(th2(t)) + lo3 sin(th2(t))

   #2 == l4 cos(th2(t)) + lo3 cos(th2(t)) - l2 sin(th2(t))
```

```
 disp('Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 4');
```

```
Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 4
```

```
W4=simplify (Jw_a4*Qp(1:4));
pretty(W4)
```

```
/ - sin(th1(t)) th2p(t) - cos(th1(t)) sin(th2(t)) th4p(t) \
|                                                         |
|  cos(th1(t)) th2p(t) - sin(th1(t)) sin(th2(t)) th4p(t)  |
|                                                         |
\             th1p(t) + cos(th2(t)) th4p(t)               /
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%  VELOCIDADES PARA ESLABÓN 3  %%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculamos el jacobiano lineal y angular de forma analítica
Jv_a3(:,GDL-3)=PO(:,:,GDL-3);
Jw_a3(:,GDL-3)=PO(:,:,GDL-3);

for k= 1:GDL-3
    if RP(k)==0
       %Para las juntas de revolución
        try
            Jv_a3(:,k)= cross(RO(:,3,k-1), PO(:,:,GDL-3)-PO(:,:,k-1));
            Jw_a3(:,k)= RO(:,3,k-1);
        catch
            Jv_a3(:,k)= cross([0,0,1], PO(:,:,GDL-3));%Matriz de rotación de 0 con
respecto a 0 es la Matriz Identidad, la posición previa tambien será 0
```

```
            Jw_a3(:,k)=[0,0,1];%Si no hay matriz de rotación previa se obtiene la
Matriz identidad
         end
     else
%         %Para las juntas prismáticas
        try
            Jv_a3(:,k)= RO(:,3,k-1);
        catch
            Jv_a3(:,k)=[0,0,1];
        end
            Jw_a3(:,k)=[0,0,0];
     end
 end

%Obtenemos SubMatrices de Jacobianos
Jv_a3= simplify (Jv_a3);
Jw_a3= simplify (Jw_a3);
%disp('Jacobiano lineal obtenido de forma analítica');
%pretty (Jv_a);
%disp('Jacobiano ángular obtenido de forma analítica');
%pretty (Jw_a);

%Matriz de Jacobiano Completa
%disp('Matriz de Jacobiano');
Jac3= [Jv_a3;
      Jw_a3];
Jacobiano3= simplify(Jac3);
% pretty(Jacobiano);

%Obtenemos vectores de Velocidades Lineales y Angulares para el eslabón 1
 disp('Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 3');
```

```
Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 3
```

```
V3=simplify (Jv_a3*Qp(1:3));
pretty(V3)
```

```
/ cos(th1(t)) th2p(t) #2 - sin(th1(t)) th1p(t) #1 - cos(th1(t)) sin(th2(t)) l3p(t) \
|                                                                                   |
| cos(th1(t)) th1p(t) #1 + sin(th1(t)) th2p(t) #2 - sin(th1(t)) sin(th2(t)) l3p(t) |
|                                                                                   |
\                          cos(th2(t)) l3p(t) - th2p(t) #1                          /

where

   #1 == l2 cos(th2(t)) - lo3 sin(th2(t))

   #2 == lo3 cos(th2(t)) - l2 sin(th2(t))
```

```
 disp('Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 3');
```

```
Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 3
```

```
W3=simplify (Jw_a3*Qp(1:3));
```

```
pretty(W3)
```

```
/ -sin(th1(t)) th2p(t) \
|                      |
|  cos(th1(t)) th2p(t) |
|                      |
\        th1p(t)       /
```

```
%%%%%%%%%%  VELOCIDADES PARA ESLABÓN 2  %%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculamos el jacobiano lineal y angular de forma analítica
Jv_a2(:,GDL-4)=PO(:,:,GDL-4);
Jw_a2(:,GDL-4)=PO(:,:,GDL-4);

for k= 1:GDL-4
    if RP(k)==0
       %Para las juntas de revolución
        try
            Jv_a2(:,k)= cross(RO(:,3,k-1), PO(:,:,GDL-4)-PO(:,:,k-1));
            Jw_a2(:,k)= RO(:,3,k-1);
        catch
            Jv_a2(:,k)= cross([0,0,1], PO(:,:,GDL-4));%Matriz de rotación de 0 con
respecto a 0 es la Matriz Identidad, la posición previa tambien será 0
            Jw_a2(:,k)=[0,0,1];%Si no hay matriz de rotación previa se obtiene la
Matriz identidad
         end
     else
%         %Para las juntas prismáticas
        try
            Jv_a2(:,k)= RO(:,3,k-1);
        catch
            Jv_a2(:,k)=[0,0,1];
        end
            Jw_a2(:,k)=[0,0,0];
     end
 end

%Obtenemos SubMatrices de Jacobianos
Jv_a2= simplify (Jv_a2);
Jw_a2= simplify (Jw_a2);
%disp('Jacobiano lineal obtenido de forma analítica');
%pretty (Jv_a);
%disp('Jacobiano ángular obtenido de forma analítica');
%pretty (Jw_a);

%Matriz de Jacobiano Completa
%disp('Matriz de Jacobiano');
Jac2= [Jv_a2;
      Jw_a2];
Jacobiano2= simplify(Jac2);
% pretty(Jacobiano);
```

```
%Obtenemos vectores de Velocidades Lineales y Angulares para el eslabón 1
 disp('Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 2');
```

```
Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 2
```

```
V2=simplify (Jv_a2*Qp(1:2));
pretty(V2)
```

```
/ - l2 cos(th2(t)) sin(th1(t)) th1p(t) - l2 cos(th1(t)) sin(th2(t)) th2p(t) \
|                                                                           |
|  l2 cos(th1(t)) cos(th2(t)) th1p(t) - l2 sin(th1(t)) sin(th2(t)) th2p(t)  |
|                                                                           |
\                          -l2 cos(th2(t)) th2p(t)                          /
```

```
 disp('Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 2');
```

```
Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 2
```

```
W2=simplify (Jw_a2*Qp(1:2));
pretty(W2)
```

```
/ -sin(th1(t)) th2p(t) \
|                      |
|  cos(th1(t)) th2p(t) |
|                      |
\        th1p(t)       /
```

```


%%%%%%%%%%  VELOCIDADES PARA ESLABÓN 1  %%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculamos el jacobiano lineal y angular de forma analítica
Jv_a1(:,GDL-5)=PO(:,:,GDL-5);
Jw_a1(:,GDL-5)=PO(:,:,GDL-5);

for k= 1:GDL-5
    if RP(k)==0
       %Para las juntas de revolución
        try
            Jv_a1(:,k)= cross(RO(:,3,k-1), PO(:,:,GDL-5)-PO(:,:,k-1));
            Jw_a1(:,k)= RO(:,3,k-1);
        catch
            Jv_a1(:,k)= cross([0,0,1], PO(:,:,GDL-5));%Matriz de rotación de 0 con respecto a 0 es la Matriz Identidad, la posición previa tambien será 0
            Jw_a1(:,k)=[0,0,1];%Si no hay matriz de rotación previa se obtiene la Matriz identidad
         end
     else
%         %Para las juntas prismáticas
        try
            Jv_a1(:,k)= RO(:,3,k-1);
        catch
            Jv_a1(:,k)=[0,0,1];
        end
```

```
            Jw_a1(:,k)=[0,0,0];
     end
 end

%Obtenemos SubMatrices de Jacobianos
Jv_a1= simplify (Jv_a1);
Jw_a1= simplify (Jw_a1);
%disp('Jacobiano lineal obtenido de forma analítica');
%pretty (Jv_a);
%disp('Jacobiano ángular obtenido de forma analítica');
%pretty (Jw_a);

%Matriz de Jacobiano Completa
%disp('Matriz de Jacobiano');
Jac1= [Jv_a1;
      Jw_a1];
Jacobiano1= simplify(Jac1);
% pretty(Jacobiano);

%Obtenemos vectores de Velocidades Lineales y Angulares para el eslabón 1
 disp('Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 1');
```

```
Velocidad lineal obtenida mediante el Jacobiano lineal del Eslabón 1
```

```
V1=simplify (Jv_a1*Qp(1:1));
pretty(V1)
```

```
/ 0 \
|   |
| 0 |
|   |
\ 0 /
```

```
 disp('Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 1');
```

```
Velocidad angular obtenida mediante el Jacobiano angular del Eslabón 1
```

```
W1=simplify (Jw_a1*Qp(1:1));
pretty(W1)
```

```
/    0    \
|         |
|    0    |
|         |
\ th1p(t) /
```

```


%Energía Cinética
%%%%%%%%%%%%%%%%%%%%%%%%%% Omitimos la división de cada lc %%%%%%%%%%%%%%%
%Distancia del origen del eslabón a su centro de masa
%Vectores de posición respecto al centro de masa
 P01=subs(P(:,:,1), l1, lc1);%La función subs sustituye l1 por lc1 en
 P12=subs(P(:,:,2), l2, lc2); %la expresión P(:,:,1)/2
 P23=subs(P(:,:,3), lo3, lc3);
```

```
 P34=subs(P(:,:,4), l4, lc4);
 P45=subs(P(:,:,5), l5, lc5);
 P56=subs(P(:,:,6), l6, lc6);

%Creamos matrices de inercia para cada eslabón

I1=[Ixx1 0 0;
    0 Iyy1 0;
    0 0 Izz1];

I2=[Ixx2 0 0;
    0 Iyy2 0;
    0 0 Izz2];

I3=[Ixx3 0 0;
    0 Iyy3 0;
    0 0 Izz3];

I4=[Ixx4 0 0;
    0 Iyy4 0;
    0 0 Izz4];

I5=[Ixx5 0 0;
   0 Iyy5 0;
    0 0 Izz5];

I6=[Ixx6 0 0;
    0 Iyy6 0;
    0 0 Izz6];

%Funciones para la energía cinemática.
%Función de energía cinética

%Calculamos la energía cinética para cada uno de los eslabones%%%%%%%%%

%Eslabón 1
V1_Total= V1+cross(W1,P01);
K1= (1/2*m1*(V1_Total))'*((V1_Total)) + (1/2*W1)'*(I1*W1); % Formula general de la energía cinetica 1/2 * m*v^2
disp('Energía Cinética en el Eslabón 1');
```

```
Energía Cinética en el Eslabón 1
```

```
K1= simplify (K1);
pretty (K1);
```

```
                2
Izz1 |th1p(t)|
---------------
       2
```

```
%Eslabón 2
V2_Total= V2+cross(W2,P12);
K2= (1/2*m2*(V2_Total))'*((V2_Total)) + (1/2*W2)'*(I2*W2);
disp('Energía Cinética en el Eslabón 2');
```

```
Energía Cinética en el Eslabón 2
```

```
K2= simplify (K2);
pretty (K2);
```

```
                                                                                                                        /
          __                                                                                                            | #
          m2 (lc2 cos(th2(t)) th1p(t) + l2 cos(th1(t)) cos(th2(t)) th1p(t) - l2 sin(th1(t)) sin(th2(t)) th2p(t)) | -
Izz2 #2                                                                                                                 \
------- + ----------------------------------------------------------------------------------------------------------------
   2                                                                                       2

where

                2
   #1 == |th2p(t)|

                2
   #2 == |th1p(t)|

             ______
   #3 == cos(th2(t))

             ______
   #4 == sin(th2(t))

             ______
   #5 == sin(th1(t))

             ______
   #6 == cos(th1(t))
```

```
%Eslabón 3
V3_Total= V3+cross(W3,P23);
K3= (1/2*m3*(V3_Total))'*((V3_Total)) + (1/2*W3)'*(I3*W3);
disp('Energía Cinética en el Eslabón 3');
```

```
Energía Cinética en el Eslabón 3
```

```
K3= simplify (K3);
pretty (K3);
```

```
__   ______           _______                                            __   ______            _______      ___    _______
m3 (l3p(t) #8 - th2p(t) #5) (cos(th2(t)) l3p(t) - th2p(t) #1)   m3 (l3p(t) #3 #7 - th2p(t) #3 lc3 - th2p(t) #3 #6 +
------------------------------------------------------------- - -----------------------------------------------------------
                              2

where

   #1 == l2 cos(th2(t)) - lo3 sin(th2(t))

             ______
   #2 == sin(th1(t))
```

```
#3 == cos(th1(t))

#4 == lo3 cos(th2(t)) - l2 sin(th2(t))

#5 == #8 l2 - #7 lo3

#6 == #8 lo3 - #7 l2

#7 == sin(th2(t))

#8 == cos(th2(t))
```

```
%Eslabón 4
V4_Total= V4+cross(W4,P34);
K4= (1/2*m4*(V4_Total))'*((V4_Total)) + (1/2*W4)'*(I4*W4);
disp('Energía Cinética en el Eslabón 4');
```

```
Energía Cinética en el Eslabón 4
```

```
K4= simplify (K4);
pretty (K4);
```

```
m4 (lc4 (#9 - #8) - l3p(t) #13 #12 + th2p(t) #13 #6 + th1p(t) #14 #5) (lc4 #2 + cos(th1(t)) th2p(t) #4 + sin(th1(t)
-----------------------------------------------------------------------------------------------------------------------
                                                                                2

where

   #1 == l4 sin(th2(t)) - l2 cos(th2(t)) + lo3 sin(th2(t))

   #2 == cos(th1(t)) th2p(t) - sin(th1(t)) sin(th2(t)) th4p(t)

   #3 == sin(th1(t)) th2p(t) + cos(th1(t)) sin(th2(t)) th4p(t)

   #4 == l4 cos(th2(t)) + lo3 cos(th2(t)) - l2 sin(th2(t))

   #5 == #12 l4 - #11 l2 + #12 lo3

   #6 == #11 l4 + #11 lo3 - #12 l2

   #7 == th4p(t) #13 #12

   #8 == th4p(t) #14 #12

   #9 == th2p(t) #13

   #10 == th2p(t) #14
```

```
                ______
   #11 == cos(th2(t))

                ______
   #12 == sin(th2(t))

                ______
   #13 == cos(th1(t))

                ______
   #14 == sin(th1(t))
```

```
%Eslabón 5
V5_Total= V5+cross(W5,P45);
K5= (1/2*m5*(V5_Total))'*((V5_Total)) + (1/2*W5)'*(I5*W5);
disp('Energía Cinética en el Eslabón 5');
```

```
Energía Cinética en el Eslabón 5
```

```
K5= simplify (K5);
pretty (K5);
```

```
     / _______             \       __  _______       __       __       ___          __              __     ______
     | th1p(t)   #25   #24 |      m5 (th2p(t) (#31 l2 - #29 l4 - #29 lo3 + #29 #22 l5 + #31 #26 #23 l5) - l3p(t) #3
Izz5 | ------- + --- + --- | #2 + --------------------------------------------------------------------------------
     \    2       2     2  /

where

   #1 == l2 sin(th2(t)) - lo3 cos(th2(t)) - l4 cos(th2(t)) + l5 cos(th2(t)) sin(th5(t)) + l5 cos(th4(t)) cos(th5(t)

   #2 == th1p(t) + cos(th2(t)) th4p(t) + sin(th2(t)) sin(th4(t)) th5p(t)

                    2
   #3 == cos(th2(t))

   #4 == lo3 sin(th2(t))

   #5 == l4 sin(th2(t))

   #6 == l2 cos(th2(t))

   #7 == th5p(t) (#19 - #18) + cos(th1(t)) th2p(t) - sin(th1(t)) sin(th2(t)) th4p(t)

   #8 == th5p(t) (#21 + #20) + sin(th1(t)) th2p(t) + cos(th1(t)) sin(th2(t)) th4p(t)

              __       ___       __           __             __
   #9 == #29 l2 - #31 lo3 - #31 l4 + #31 #22 l5 + #26 #23 #29 l5

          _______
   #10 == th1p(t) + #25 + #24

          _______
   #11 == th5p(t) (#27 #26 - #31 #28 #30)

          _______
   #12 == th5p(t) (#26 #28 + #27 #31 #30)

              __       __       ___
   #13 == #29 l4 - #31 l2 + #29 lo3
```

```
       _______
#14 == th4p(t) #27 #29

       _______
#15 == th4p(t) #28 #29

       _______
#16 == th2p(t) #27

       _______
#17 == th2p(t) #28

#18 == cos(th2(t)) sin(th1(t)) sin(th4(t))

#19 == cos(th1(t)) cos(th4(t))

#20 == cos(th1(t)) cos(th2(t)) sin(th4(t))

#21 == cos(th4(t)) sin(th1(t))

           ______
#22 == sin(th5(t))

           ______
#23 == cos(th5(t))

       _______
#24 == th5p(t) #29 #30

       _______
#25 == th4p(t) #31

           ______
#26 == cos(th4(t))

           ______
#27 == cos(th1(t))

           ______
#28 == sin(th1(t))

           ______
#29 == sin(th2(t))

           ______
#30 == sin(th4(t))

           ______
#31 == cos(th2(t))
```

```
%Eslabón 6
V6_Total= V6+cross(W6,P56);
K6= (1/2*m6*(V6_Total))'*((V6_Total)) + (1/2*W6)'*(I6*W6);
disp('Energía Cinética en el Eslabón 6');
```

```
Energía Cinética en el Eslabón 6
```

```
K6= simplify (K6);
pretty (K6);
```

Izz6 (th1p(t) + th6p(t) (cos(th2(t)) cos(th5(t)) + cos(th4(t)) sin(th2(t)) sin(th5(t))) + cos(th2(t)) th4p(t) + sin(

where

#1 == l5 cos(th5(t)) - l6 sin(th5(t))

#2 == l5 sin(th1(t)) sin(th2(t)) sin(th5(t))

#3 == l5 cos(th1(t)) sin(th2(t)) sin(th5(t))

#4 == lo3 sin(th2(t))

#5 == l4 sin(th2(t))

#6 == l2 cos(th2(t))

#7 == $\cos(th2(t))^2$

#8 == th5p(t) (#29 - #28) - th6p(t) #27 + cos(th1(t)) th2p(t) - sin(th1(t)) sin(th2(t)) th4p(t)

#9 == th5p(t) (#32 + #31) - th6p(t) #30 + sin(th1(t)) th2p(t) + cos(th1(t)) sin(th2(t)) th4p(t)

#10 == l4 cos(th2(t)) + lo3 cos(th2(t)) - l2 sin(th2(t)) + #36 - #35 - #33 + #34

#11 == #50 $\overline{l4}$ + #50 $\overline{lo3}$ - #46 $\overline{l2}$ + #50 #48 $\overline{l6}$ - #50 #47 $\overline{l5}$ - #37 + #38

#12 == l5 cos(th5(t)) #44

#13 == l5 cos(th5(t)) #43

#14 == #50 #48 $\overline{l6}$ - #50 #47 $\overline{l5}$ - #37 + #38

#15 == $\overline{th6p(t)}$ #39

#16 == $\overline{th6p(t)}$ #40

#17 == #48 $\overline{l5}$ - #47 $\overline{l6}$

#18 == $\overline{th5p(t)}$ #41

#19 == $\overline{th5p(t)}$ #42

#20 == #46 $\overline{l4}$ - #50 $\overline{l2}$ + #46 $\overline{lo3}$

#21 == $\overline{th4p(t)}$ #53 #46

#22 == $\overline{th4p(t)}$ #52 #46

#23 == #53 #46 #47 $\overline{l5}$

#24 == #52 #46 #47 $\overline{\texttt{l5}}$

#25 == $\overline{\texttt{th2p(t)}}$ #53

#26 == $\overline{\texttt{th2p(t)}}$ #52

#27 == sin(th5(t)) #43 + cos(th5(t)) sin(th1(t)) sin(th2(t))

#28 == cos(th2(t)) sin(th1(t)) sin(th4(t))

#29 == cos(th1(t)) cos(th4(t))

#30 == sin(th5(t)) #44 - cos(th1(t)) cos(th5(t)) sin(th2(t))

#31 == cos(th1(t)) cos(th2(t)) sin(th4(t))

#32 == cos(th4(t)) sin(th1(t))

#33 == l5 cos(th4(t)) cos(th5(t)) sin(th2(t))

#34 == l6 cos(th4(t)) sin(th2(t)) sin(th5(t))

#35 == l5 cos(th2(t)) sin(th5(t))

#36 == l6 cos(th2(t)) cos(th5(t))

#37 == #51 #48 #46 $\overline{\texttt{l5}}$

#38 == #51 #46 #47 $\overline{\texttt{l6}}$

#39 == #47 #45 - #53 #48 #46

#40 == #47 #49 + #48 #52 #46

#41 == #53 #51 - #50 #52 #54

#42 == #51 #52 + #53 #50 #54

#43 == cos(th1(t)) sin(th4(t)) + cos(th2(t)) cos(th4(t)) sin(th1(t))

#44 == sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(t))

#45 == #52 #54 - #53 #50 #51

#46 == sin($\overline{\texttt{th2(t)}}$)

#47 == sin($\overline{\texttt{th5(t)}}$)

#48 == cos($\overline{\texttt{th5(t)}}$)

#49 == #53 #54 + #50 #51 #52

#50 == cos($\overline{\texttt{th2(t)}}$)

```
                   ______
   #51 == cos(th4(t))

                   ______
   #52 == sin(th1(t))

                   ______
   #53 == cos(th1(t))

                   ______
   #54 == sin(th4(t))
```

```
K_Total= simplify (K1+K2+K3+K4+K5+K6);
disp('Energía Cinética Total');
```

```
Energía Cinética Total
```

```
pretty (K_Total);
```

```
                    __
                    m2 (lc2 cos(th2(t)) th1p(t) + l2 cos(th1(t)) cos(th2(t)) th1p(t) - l2 sin(th1(t)) sin(th2(t)) th
Izz1 #1   Izz2 #1
------- + ------- + ----------------------------------------------------------------------------------------------------
   2         2                                                                                  2

where

                    2
   #1 == |th1p(t)|

   #2 == sin(th1(t)) sin(th2(t)) l3p(t)

   #3 == cos(th1(t)) sin(th2(t)) l3p(t)

                    2
   #4 == |th2p(t)|

   #5 == l5 cos(th5(t)) - l6 sin(th5(t))

   #6 == l5 sin(th1(t)) sin(th2(t)) sin(th5(t))

   #7 == l5 cos(th1(t)) sin(th2(t)) sin(th5(t))

   #8 == cos(th2(t)) cos(th5(t)) + cos(th4(t)) sin(th2(t)) sin(th5(t))

   #9 == cos(th2(t)) l3p(t)

   #10 == l5 cos(th2(t)) cos(th4(t)) cos(th5(t))

          _______
          th1p(t)
   #11 == -------
             2

   #12 == l5 sin(th2(t)) sin(th5(t))

   #13 == th5p(t) #57 - th6p(t) #52 + #58 - #56

   #14 == th5p(t) #60 - th6p(t) #53 + #61 + #59

   #15 == #66 + #65 - #64 + #55 - #63 - #62 + #54
```

#16 == l5 cos(th5(t)) #82

#17 == l5 cos(th5(t)) #81

#18 == th5p(t) #57 + #58 - #56

#19 == th5p(t) #60 + #61 + #59

#20 == #64 - #65 - #66 + #63 + #62

#21 == th1p(t) + #67 + #68

#22 == 2 cos(th4(t)) sin(th1(t)) #69 + cos(th1(t)) sin(th4(t)) cos(th2(t)) - #84

#23 == - 2 cos(th1(t)) cos(th4(t)) #69 + sin(th1(t)) sin(th4(t)) cos(th2(t)) + #83

#24 == #92 $\overline{\mathrm{l4}}$ + #92 $\overline{\mathrm{lo3}}$ - #89 $\overline{\mathrm{l2}}$ + #92 #87 $\overline{\mathrm{l6}}$ - #92 #86 $\overline{\mathrm{l5}}$ - #72 + #73

#25 == #89 $\overline{\mathrm{l4}}$ - #92 $\overline{\mathrm{l2}}$ + #89 $\overline{\mathrm{lo3}}$

#26 == $\overline{\mathrm{th5p(t)}}$ #70

#27 == $\overline{\mathrm{th5p(t)}}$ #71

#28 == #89 $\overline{\mathrm{l2}}$ - #92 $\overline{\mathrm{lo3}}$ - #92 $\overline{\mathrm{l4}}$ + #92 #86 $\overline{\mathrm{l5}}$ + #72

#29 == #92 #87 $\overline{\mathrm{l6}}$ - #92 #86 $\overline{\mathrm{l5}}$ - #72 + #73

#30 == $\overline{\mathrm{th4p(t)}}$ #91 #89

#31 == $\overline{\mathrm{th4p(t)}}$ #90 #89

#32 == $\overline{\mathrm{l3p(t)}}$ #91 #89

#33 == $\overline{\mathrm{l3p(t)}}$ #90 #89

#34 == $\overline{\mathrm{th2p(t)}}$ #91

#35 == $\overline{\mathrm{th2p(t)}}$ #90

#36 == $\overline{\mathrm{l3p(t)}}$ #92

#37 == $\overline{\mathrm{th6p(t)}}$ #74

#38 == $\overline{\mathrm{th6p(t)}}$ #75

#39 == - 2 #91 #94 $\#92^{2}$ + #90 #93 #92 + #91 #94

$\#40 == 2\ \#94\ \#90\ \#92^{2} + \#91\ \#93\ \#92 - \#94\ \#90$

$\#41 == \#92\ \overline{l2} - \#89\ \overline{lo3}$

$\#42 == \#87\ \overline{l5} - \#86\ \overline{l6}$

$\#43 == \overline{th1p(t)} + \#77 + \#76$

$\#44 == \#91\ \#89\ \#86\ \overline{l5}$

$\#45 == \#90\ \#89\ \#86\ \overline{l5}$

$\#46 == \#92\ \overline{l4} + \#92\ \overline{lo3} - \#89\ \overline{l2}$

$\#47 == \sin(th1(t))\ (\#80 - \#78 + \#79)$

$\#48 == \cos(th1(t))\ (\#80 - \#78 + \#79)$

$\#49 == \#92\ \overline{lo3} - \#89\ \overline{l2}$

$\#50 == \#92\ \#94\ \#87\ \overline{l5}$

$\#51 == \#92\ \#87 + \#94\ \#89\ \#86$

$\#52 == \sin(th5(t))\ \#81 + \cos(th5(t))\ \sin(th1(t))\ \sin(th2(t))$

$\#53 == \sin(th5(t))\ \#82 - \cos(th1(t))\ \cos(th5(t))\ \sin(th2(t))$

$\#54 == l6\ \cos(th4(t))\ \sin(th2(t))\ \sin(th5(t))$

$\#55 == l6\ \cos(th2(t))\ \cos(th5(t))$

$\#56 == \sin(th1(t))\ \sin(th2(t))\ th4p(t)$

$\#57 == \#83 - \cos(th2(t))\ \sin(th1(t))\ \sin(th4(t))$

$\#58 == \cos(th1(t))\ th2p(t)$

$\#59 == \cos(th1(t))\ \sin(th2(t))\ th4p(t)$

$\#60 == \#84 + \cos(th1(t))\ \cos(th2(t))\ \sin(th4(t))$

$\#61 == \sin(th1(t))\ th2p(t)$

$\#62 == l5\ \cos(th4(t))\ \cos(th5(t))\ \sin(th2(t))$

$\#63 == l5\ \cos(th2(t))\ \sin(th5(t))$

$\#64 == l2\ \sin(th2(t))$

$\#65 == lo3\ \cos(th2(t))$

$\#66 == l4\ \cos(th2(t))$

```
#67 == cos(th2(t)) th4p(t)

#68 == sin(th2(t)) sin(th4(t)) th5p(t)

                  2
#69 == cos(th2(t))

#70 == #91 #94 - #92 #90 #93

#71 == #94 #90 + #91 #92 #93

                   __
#72 == #94 #87 #89 l5

                   __
#73 == #94 #89 #86 l6

#74 == #86 #85 - #91 #87 #89

#75 == #86 #88 + #87 #90 #89

       _______
#76 == th5p(t) #89 #93

       _______
#77 == th4p(t) #92

#78 == l2 cos(th2(t))

#79 == lo3 sin(th2(t))

#80 == l4 sin(th2(t))

#81 == cos(th1(t)) sin(th4(t)) + cos(th2(t)) cos(th4(t)) sin(th1(t))

#82 == sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(t))

#83 == cos(th1(t)) cos(th4(t))

#84 == cos(th4(t)) sin(th1(t))

#85 == #90 #93 - #91 #92 #94

           ______
#86 == sin(th5(t))

           ______
#87 == cos(th5(t))

#88 == #91 #93 + #92 #94 #90

           ______
#89 == sin(th2(t))

           ______
#90 == sin(th1(t))

           ______
#91 == cos(th1(t))

           ______
#92 == cos(th2(t))

           ______
```

$\#93 == \sin(\mathrm{th4}(t))$

$\#94 == \cos(\overline{\mathrm{th4}(t)})$

```
%Energia Potencial  p=mgh

%Obtenemos las alturas respecto a la gravedad
 h1= P01(3); %Tomo la altura paralela al eje z
 h2= P12(2); %Tomo la altura paralela al eje y
 h3= P23(3); %Tomo la altura paralela al eje y
 h4= P34(3);
 h5= P45(2);
 h6= P56(3);

 U1=m1*g*h1
```

U1 = $g\, \mathrm{lc}_1\, m_1$

```
U2=m2*g*h2
```

U2 = $g\, \mathrm{lc}_2\, m_2 \sin(\mathrm{th}_2(t))$

```
U3=m3*g*h3
```

U3 = $g\, \mathrm{lc}_3\, m_3$

```
U4=m4*g*h4
```

U4 = $g\, \mathrm{lc}_4\, m_4$

```
U5=m5*g*h5
```

U5 = $g\, \mathrm{lc}_5\, m_5 \sin(\mathrm{th}_5(t))$

```
U6=m6*g*h6
```

U6 = $g\, \mathrm{lc}_6\, m_6$

```
%Calculamos la energía potencial total
U_Total= U1 + U2 +U3 +U4 +U5 +U6;
```

Para el calcuo delo lagrangiano se hace una resta entre la energía cinética total y la energía potencial total de todo el sistema. Para obtener una correcta formulación de las ecuaciones de movimiento de todo el sistema debido a que se utiliza para poder derivar las ecuaciones.

```
%Obtenemos el Lagrangiano
Lagrangiano= simplify (K_Total-U_Total);
pretty (Lagrangiano);
```

```
                    __
                    m2 (lc2 cos(th2(t)) th1p(t) + l2 cos(th1(t)) cos(th2(t)) th1p(t) - l2 sin(th1(t)) sin(th2(t)) th
Izz1 #1   Izz2 #1
------- + ------- + -----------------------------------------------------------------------------------------------
   2         2                                                                                     2

where

                  2
   #1 == |th1p(t)|

   #2 == sin(th1(t)) sin(th2(t)) l3p(t)

   #3 == cos(th1(t)) sin(th2(t)) l3p(t)

                  2
   #4 == |th2p(t)|

   #5 == l5 cos(th5(t)) - l6 sin(th5(t))

   #6 == l5 sin(th1(t)) sin(th2(t)) sin(th5(t))

   #7 == l5 cos(th1(t)) sin(th2(t)) sin(th5(t))

   #8 == cos(th2(t)) cos(th5(t)) + cos(th4(t)) sin(th2(t)) sin(th5(t))

   #9 == cos(th2(t)) l3p(t)

   #10 == l5 cos(th2(t)) cos(th4(t)) cos(th5(t))

          _______
          th1p(t)
   #11 == -------
             2

   #12 == l5 sin(th2(t)) sin(th5(t))

   #13 == th5p(t) #57 - th6p(t) #52 + #58 - #56

   #14 == th5p(t) #60 - th6p(t) #53 + #61 + #59

   #15 == #68 + #67 - #66 + #55 - #65 - #64 + #54

   #16 == l5 cos(th5(t)) #82

   #17 == l5 cos(th5(t)) #81

   #18 == th5p(t) #57 + #58 - #56

   #19 == th5p(t) #60 + #61 + #59

   #20 == th1p(t) + #62 + #63

   #21 == #66 - #67 - #68 + #65 + #64

   #22 == 2 cos(th4(t)) sin(th1(t)) #69 + cos(th1(t)) sin(th4(t)) cos(th2(t)) - #84

   #23 == - 2 cos(th1(t)) cos(th4(t)) #69 + sin(th1(t)) sin(th4(t)) cos(th2(t)) + #83

              __       __       ___
   #24 == #89 l4 - #92 l2 + #89 lo3
```

$\#25 == \#92\ l4 + \#92\ lo3 - \#89\ l2 + \#92\ \#87\ l6 - \#92\ \#88\ l5 - \#74 + \#75$

$\#26 == \overline{th5p(t)}\ \#70$

$\#27 == \overline{th5p(t)}\ \#71$

$\#28 == \#89\ \overline{l2} - \#92\ \overline{lo3} - \#92\ \overline{l4} + \#92\ \#88\ \overline{l5} + \#74$

$\#29 == \overline{th6p(t)}\ \#72$

$\#30 == \overline{th6p(t)}\ \#73$

$\#31 == \#92\ \#87\ \overline{l6} - \#92\ \#88\ \overline{l5} - \#74 + \#75$

$\#32 == \overline{th4p(t)}\ \#91\ \#89$

$\#33 == \overline{th4p(t)}\ \#90\ \#89$

$\#34 == \overline{l3p(t)}\ \#91\ \#89$

$\#35 == \overline{l3p(t)}\ \#90\ \#89$

$\#36 == \overline{th2p(t)}\ \#91$

$\#37 == \overline{th2p(t)}\ \#90$

$\#38 == \overline{l3p(t)}\ \#92$

$\#39 == - 2\ \#91\ \#94\ \#92^{2} + \#90\ \#93\ \#92 + \#91\ \#94$

$\#40 == 2\ \#94\ \#90\ \#92^{2} + \#91\ \#93\ \#92 - \#94\ \#90$

$\#41 == \#92\ \overline{l2} - \#89\ \overline{lo3}$

$\#42 == \#87\ \overline{l5} - \#88\ \overline{l6}$

$\#43 == \overline{th1p(t)} + \#77 + \#76$

$\#44 == \#91\ \#89\ \#88\ \overline{l5}$

$\#45 == \#90\ \#89\ \#88\ \overline{l5}$

$\#46 == \#92\ \overline{l4} + \#92\ \overline{lo3} - \#89\ \overline{l2}$

#47 == sin(th1(t)) (#80 - #78 + #79)

#48 == cos(th1(t)) (#80 - #78 + #79)

#49 == #92 $\overline{lo3}$ - #89 $\overline{l2}$

#50 == #92 #94 #87 $\overline{l5}$

#51 == #92 #87 + #94 #89 #88

#52 == sin(th5(t)) #81 + cos(th5(t)) sin(th1(t)) sin(th2(t))

#53 == sin(th5(t)) #82 - cos(th1(t)) cos(th5(t)) sin(th2(t))

#54 == l6 cos(th4(t)) sin(th2(t)) sin(th5(t))

#55 == l6 cos(th2(t)) cos(th5(t))

#56 == sin(th1(t)) sin(th2(t)) th4p(t)

#57 == #83 - cos(th2(t)) sin(th1(t)) sin(th4(t))

#58 == cos(th1(t)) th2p(t)

#59 == cos(th1(t)) sin(th2(t)) th4p(t)

#60 == #84 + cos(th1(t)) cos(th2(t)) sin(th4(t))

#61 == sin(th1(t)) th2p(t)

#62 == cos(th2(t)) th4p(t)

#63 == sin(th2(t)) sin(th4(t)) th5p(t)

#64 == l5 cos(th4(t)) cos(th5(t)) sin(th2(t))

#65 == l5 cos(th2(t)) sin(th5(t))

#66 == l2 sin(th2(t))

#67 == lo3 cos(th2(t))

#68 == l4 cos(th2(t))

#69 == $\mathrm{cos(th2(t))}^2$

#70 == #91 #94 - #92 #90 #93

#71 == #94 #90 + #91 #92 #93

#72 == #88 #85 - #91 #87 #89

#73 == #88 #86 + #87 #90 #89

#74 == #94 #87 #89 $\overline{l5}$

#75 == #94 #89 #88 $\overline{l6}$

```
            _______
  #76 == th5p(t) #89 #93

            _______
  #77 == th4p(t) #92

  #78 == l2 cos(th2(t))

  #79 == lo3 sin(th2(t))

  #80 == l4 sin(th2(t))

  #81 == cos(th1(t)) sin(th4(t)) + cos(th2(t)) cos(th4(t)) sin(th1(t))

  #82 == sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(t))

  #83 == cos(th1(t)) cos(th4(t))

  #84 == cos(th4(t)) sin(th1(t))

  #85 == #90 #93 - #91 #92 #94

  #86 == #91 #93 + #92 #94 #90

                ______
  #87 == cos(th5(t))

                ______
  #88 == sin(th5(t))

                ______
  #89 == sin(th2(t))

                ______
  #90 == sin(th1(t))

                ______
  #91 == cos(th1(t))

                ______
  #92 == cos(th2(t))

                ______
  #93 == sin(th4(t))

                ______
  #94 == cos(th4(t))
```

El modelo de energia se define con H la cual es la suma de la energía total con la energía potencial total del sistema, proporciona una representación de la energía del sistema y se utiliza para analizar el comportamiento.

```
%Modelo de Energía
disp('Modelo de energía total');
```

```
Modelo de energía total
```

```
 H= simplify (K_Total+U_Total);
  pretty (H)
```

```
                 __
                 m2 (lc2 cos(th2(t)) th1p(t) + l2 cos(th1(t)) cos(th2(t)) th1p(t) - l2 sin(th1(t)) sin(th2(t)) t
```

```
Izz1 #1   Izz2 #1
------- + ------- + -----------------------------------------------------------------------------------------
   2         2                                                                                      2

where

                  2
   #1 == |th1p(t)|

   #2 == sin(th1(t)) sin(th2(t)) l3p(t)

   #3 == cos(th1(t)) sin(th2(t)) l3p(t)

                  2
   #4 == |th2p(t)|

   #5 == l5 cos(th5(t)) - l6 sin(th5(t))

   #6 == l5 sin(th1(t)) sin(th2(t)) sin(th5(t))

   #7 == l5 cos(th1(t)) sin(th2(t)) sin(th5(t))

   #8 == cos(th2(t)) cos(th5(t)) + cos(th4(t)) sin(th2(t)) sin(th5(t))

   #9 == cos(th2(t)) l3p(t)

   #10 == l5 cos(th2(t)) cos(th4(t)) cos(th5(t))

          _______
          th1p(t)
   #11 == -------
             2

   #12 == l5 sin(th2(t)) sin(th5(t))

   #13 == th5p(t) #57 - th6p(t) #52 + #58 - #56

   #14 == th5p(t) #60 - th6p(t) #53 + #61 + #59

   #15 == #68 + #67 - #66 + #55 - #65 - #64 + #54

   #16 == l5 cos(th5(t)) #82

   #17 == l5 cos(th5(t)) #81

   #18 == th5p(t) #57 + #58 - #56

   #19 == th5p(t) #60 + #61 + #59

   #20 == th1p(t) + #62 + #63

   #21 == #66 - #67 - #68 + #65 + #64

   #22 == 2 cos(th4(t)) sin(th1(t)) #69 + cos(th1(t)) sin(th4(t)) cos(th2(t)) - #84

   #23 == - 2 cos(th1(t)) cos(th4(t)) #69 + sin(th1(t)) sin(th4(t)) cos(th2(t)) + #83

              __       __       ___
   #24 == #89 l4 - #92 l2 + #89 lo3

              __       ___       __           __           __
   #25 == #92 l4 + #92 lo3 - #89 l2 + #92 #87 l6 - #92 #88 l5 - #74 + #75

          _______
```

#26 == th5p(t) #70

#27 == $\overline{\text{th5p(t)}}$ #71

#28 == #89 $\overline{\text{l2}}$ - #92 $\overline{\text{lo3}}$ - #92 $\overline{\text{l4}}$ + #92 #88 $\overline{\text{l5}}$ + #74

#29 == $\overline{\text{th6p(t)}}$ #72

#30 == $\overline{\text{th6p(t)}}$ #73

#31 == #92 #87 $\overline{\text{l6}}$ - #92 #88 $\overline{\text{l5}}$ - #74 + #75

#32 == $\overline{\text{th4p(t)}}$ #91 #89

#33 == $\overline{\text{th4p(t)}}$ #90 #89

#34 == $\overline{\text{l3p(t)}}$ #91 #89

#35 == $\overline{\text{l3p(t)}}$ #90 #89

#36 == $\overline{\text{th2p(t)}}$ #91

#37 == $\overline{\text{th2p(t)}}$ #90

#38 == $\overline{\text{l3p(t)}}$ #92

#39 == - 2 #91 #94 $\#92^{2}$ + #90 #93 #92 + #91 #94

#40 == 2 #94 #90 $\#92^{2}$ + #91 #93 #92 - #94 #90

#41 == #92 $\overline{\text{l2}}$ - #89 $\overline{\text{lo3}}$

#42 == #87 $\overline{\text{l5}}$ - #88 $\overline{\text{l6}}$

#43 == $\overline{\text{th1p(t)}}$ + #77 + #76

#44 == #91 #89 #88 $\overline{\text{l5}}$

#45 == #90 #89 #88 $\overline{\text{l5}}$

#46 == #92 $\overline{\text{l4}}$ + #92 $\overline{\text{lo3}}$ - #89 $\overline{\text{l2}}$

#47 == sin(th1(t)) (#80 - #78 + #79)

#48 == cos(th1(t)) (#80 - #78 + #79)

#49 == #92 $\overline{\mathrm{lo3}}$ - #89 $\overline{\mathrm{l2}}$

#50 == #92 #94 #87 $\overline{\mathrm{l5}}$

#51 == #92 #87 + #94 #89 #88

#52 == sin(th5(t)) #81 + cos(th5(t)) sin(th1(t)) sin(th2(t))

#53 == sin(th5(t)) #82 - cos(th1(t)) cos(th5(t)) sin(th2(t))

#54 == l6 cos(th4(t)) sin(th2(t)) sin(th5(t))

#55 == l6 cos(th2(t)) cos(th5(t))

#56 == sin(th1(t)) sin(th2(t)) th4p(t)

#57 == #83 - cos(th2(t)) sin(th1(t)) sin(th4(t))

#58 == cos(th1(t)) th2p(t)

#59 == cos(th1(t)) sin(th2(t)) th4p(t)

#60 == #84 + cos(th1(t)) cos(th2(t)) sin(th4(t))

#61 == sin(th1(t)) th2p(t)

#62 == cos(th2(t)) th4p(t)

#63 == sin(th2(t)) sin(th4(t)) th5p(t)

#64 == l5 cos(th4(t)) cos(th5(t)) sin(th2(t))

#65 == l5 cos(th2(t)) sin(th5(t))

#66 == l2 sin(th2(t))

#67 == lo3 cos(th2(t))

#68 == l4 cos(th2(t))

#69 == $\cos(\mathrm{th2}(t))^2$

#70 == #91 #94 - #92 #90 #93

#71 == #94 #90 + #91 #92 #93

#72 == #88 #85 - #91 #87 #89

#73 == #88 #86 + #87 #90 #89

#74 == #94 #87 #89 $\overline{\mathrm{l5}}$

#75 == #94 #89 #88 $\overline{\mathrm{l6}}$

#76 == $\overline{\mathrm{th5p}(t)}$ #89 #93

#77 == $\overline{\mathrm{th4p}(t)}$ #92

#78 == l2 cos(th2(t))

#79 == lo3 sin(th2(t))

#80 == l4 sin(th2(t))

#81 == cos(th1(t)) sin(th4(t)) + cos(th2(t)) cos(th4(t)) sin(th1(t))

#82 == sin(th1(t)) sin(th4(t)) - cos(th1(t)) cos(th2(t)) cos(th4(t))

#83 == cos(th1(t)) cos(th4(t))

#84 == cos(th4(t)) sin(th1(t))

#85 == #90 #93 - #91 #92 #94

#86 == #91 #93 + #92 #94 #90

#87 == cos($\overline{\mathrm{th5}(t)}$)

#88 == sin($\overline{\mathrm{th5}(t)}$)

#89 == sin($\overline{\mathrm{th2}(t)}$)

#90 == sin($\overline{\mathrm{th1}(t)}$)

#91 == cos($\overline{\mathrm{th1}(t)}$)

#92 == cos($\overline{\mathrm{th2}(t)}$)

#93 == sin($\overline{\mathrm{th4}(t)}$)

#94 == cos($\overline{\mathrm{th4}(t)}$)