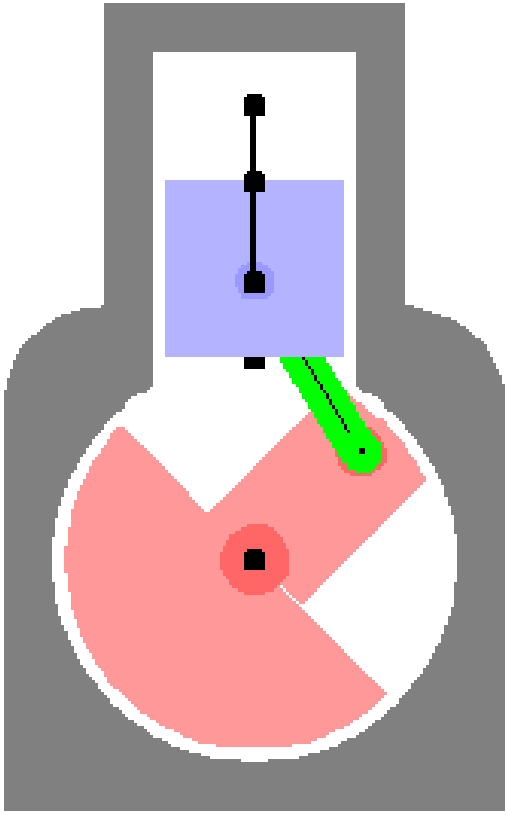
RESTRICCIONES EN "MECHANICA" (1)
COMPRESOR 2D, 4 CUERPOS



3 CUERPOS MOVILES -> 9 COORDENADAS

$$\mathbf{q} \equiv [q_1, \; q_2, \; \cdots, \; q_{nc}]^T$$

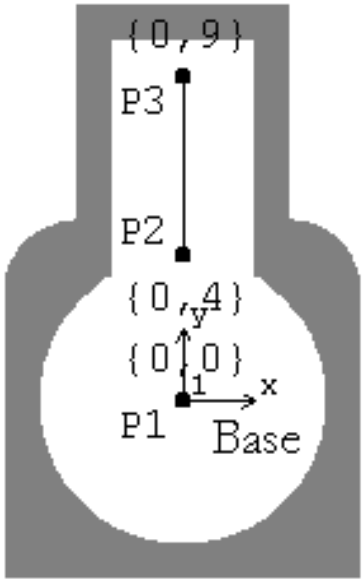
EJEMPLO COMPRESOR - 2D:

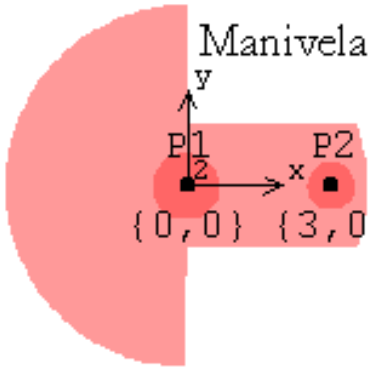
$$nc = 9$$

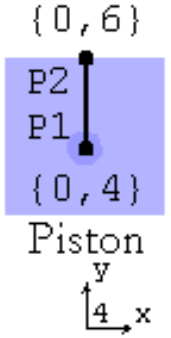
$$\mathbf{q} \equiv [q_1 \;\; q_2 \;\; q_3 \;\; q_4 \;\; q_5 \;\; q_6 \;\; q_7 \;\; q_8 \;\; q_9]^T$$

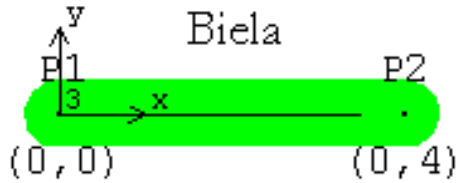
$$\mathbf{q} \equiv [X2 \;\; Y2 \;\; Th2 \;\; X3 \;\; Y3 \;\; Th3 \;\; X4 \;\; Y4 \;\; Th4]^T$$

RESTRICCIONES EN “MECHANICA” (2)
COMPRESOR 2D, 4 CUERPOS





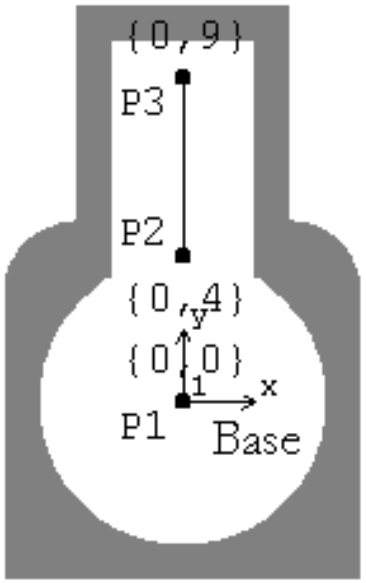


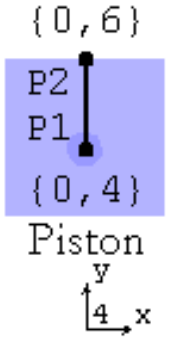


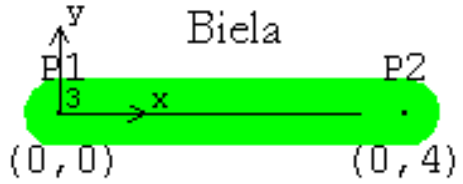
CUERPOS QUE COMPONEN EL SISTEMA MECANICO:

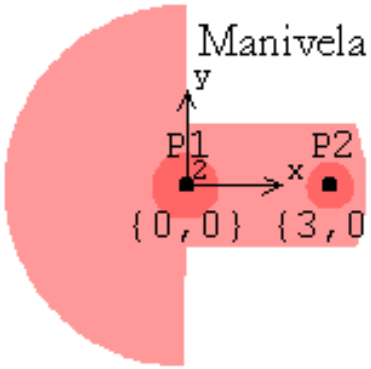
```
base = 1;
manivela = 2;
biela  = 3;
piston = 4;
```

RESTRICCIONES EN "MECHANICA" (3)
COMPRESOR 2D, 4 CUERPOS









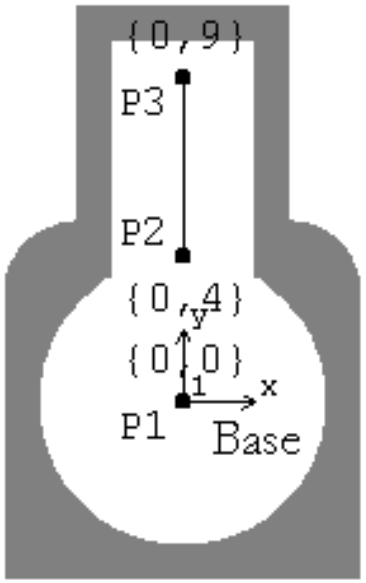
DEFINICION DE CADA CUERPO MEDIANTE PUNTOS:

```
bd[1] = Body[base, PointList->
  {(*P1*){0., 0.},
   (*P2*){0., 4.},
   (*P3*){0., 9.}}]
bd[2] = Body[manivela, PointList->
  {(*P1*){0., 0.},
   (*P2*){3., 0.}}];
bd[3] = Body[biela, PointList->
  {(*P1*){0.,0.},
   (*P2*){4.,0.}}];
bd[4] = Body[piston, PointList->
  {(*P1*){0., 4.},
   (*P2*){0., 6.}}];
```

```
SetBodies[ bd[1], bd[2], bd[3], bd[4] ];

SetBodies[Body[biela, InitialGuess->{{2,2},1}]];
```

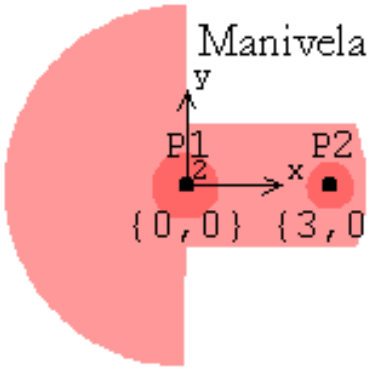
RESTRICCIONES EN “MECHANICA” (4)
Revolute2, Translate2, RotationLock1

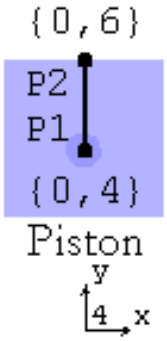


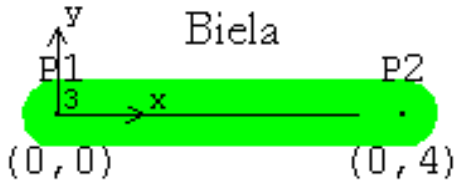
LA DEFINICION DE LAS RESTRICCIONES IMPUESTAS POR LOS PARES EXISTENTES SE REALIZA UTILIZANDO LOS PUNTOS DEFINIDOS EN CADA CUERPO:

```
cs[1] = Revolute2[1,Point[manivela,1], Point[base,1]];
cs[2] = Revolute2[2,Point[biela,1],Point[manivela,2]];
cs[3] = Revolute2[3,Point[piston,1],Point[biela,2]];
cs[4] = Translate2[4,Line[base,2,3], Line[piston,1,2]];
cs[5] = RotationLock1[5,manivela, 2 Pi T];
```

```
SetConstraints[ cs[1], cs[2], cs[3], cs[4], cs[5]];
```







```
cs[1] = Revolute2[1,Point[manivela,2], Point[base,1]];
cs[2] = Revolute2[2,Point[biela,1],Point[manivela,1]];
cs[3] = Revolute2[3,Point[piston,1],Point[biela,2]];
cs[4] = Translate2[4,Line[base,2,3], Line[piston,1,2]];
cs[5] = RotationLock1[5,manivela, 2 Pi T];


cs[1] = Revolute2[1,Point[manivela,1], Point[base,1]];
cs[2] = Revolute2[2,Point[biela,1],Point[manivela,2]];
cs[3] = Revolute2[3,Point[piston,2],Point[biela,1]];
cs[4] = Translate2[4,Line[base,2,3], Line[piston,1,2]];
cs[5] = RotationLock1[5,manivela, 2 Pi T];

 cs[1] = Revolute2[1,Point[manivela,1], Point[base,1]];
 cs[2] = Revolute2[2,Point[biela,1],Point[manivela,2]];
 cs[3] = Revolute2[3,Point[piston,1],Point[biela,2]];
 cs[4] = Translate2[4,Line[base,2,3], Line[biela,1,2]];
 cs[5] = RotationLock1[5,manivela, 2 Pi T];


SetConstraints[ cs[1], cs[2], cs[3], cs[4] ];
```

RESTRICCIONES EN “MECHANICA” (5)
Revolute2, Translate2, RotationLock1

UTILIZANDO METODOS NUMERICOS SE PODRA RESOLVER EL SISTEMA DE ECUACIONES DE RESTRICCION (INCLUYENDO LAS CINEMATICAS Y LAS DE CONDUCCION) PARA OBTENER EL VECTOR DE COORDENADAS GENERALIZADAS EN CADA INSTANTE:

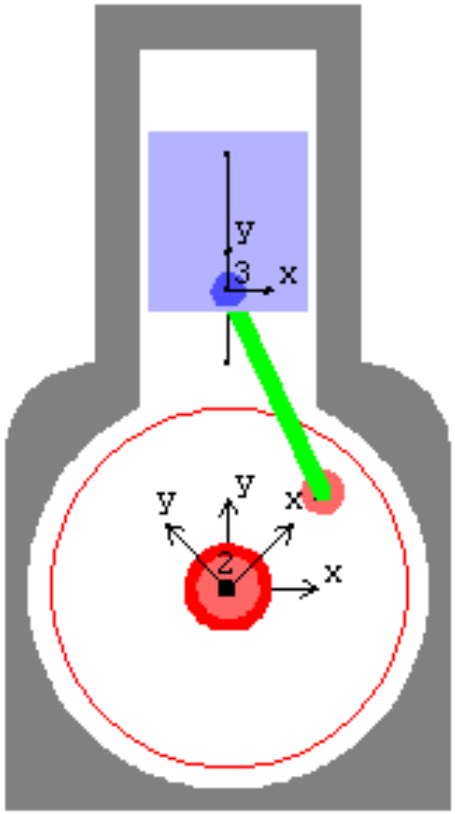
MatrixForm[Constraints[All]]

```
X2 = 0
Y2 = 0
-X2 + X3 - 3. Cos[Th2] = 0
-Y2 + Y3 - 3. Sin[Th2] = 0
-X3 + X4 - 4. Cos[Th3] - 4. Sin[Th4] = 0
-Y3 + Y4 + 4. Cos[Th4] - 4. Sin[Th3] = 0
10. Sin[Th4] = 0
2. Cos[Th4] (X4 - 4. Sin[Th4]) - 2. (4. - Y4 - 4. Cos[Th4]) Sin[Th4] = 0
-2 Pi T + Th2 = 0
```

SolveMech[0.]

```
{T -> 0., X2 -> 0., Y2 -> 0., Th2 -> 0., X3 -> 3., Y3 -> 0., Th3 -> 2.41886,
                    -22                                        -23
 X4 -> -3.83377 10   , Y4 -> -1.35425, Th4 -> -5.29396 10   }
```

RESTRICCIONES EN "MECHANICA" (6)
COMPRESOR 2D, 3 CUERPOS



2 CUERPOS MOVILES -> 6 COORDENADAS

$$\mathbf{q} \equiv [q_1, \quad q_2, \quad \cdots, \quad q_{nc}]^T$$

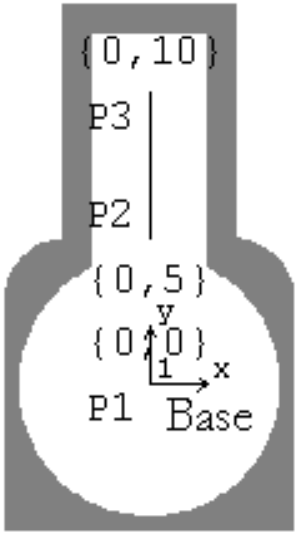
EJEMPLO COMPRESOR - 2D:

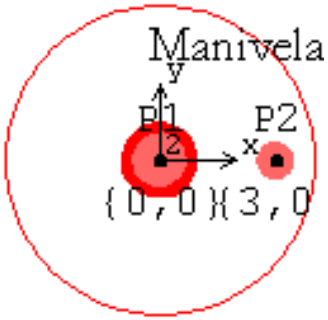
$$nc = 6$$

$$\mathbf{q} \equiv [q_1 \quad q_2 \quad q_3 \quad q_4 \quad q_5 \quad q_6]^T$$

$$\mathbf{q} \equiv [X2 \quad Y2 \quad Th2 \quad X3 \quad Y3 \quad Th3]^T$$

RESTRICCIONES EN “MECHANICA” (7)
COMPRESOR 2D, 3 CUERPOS

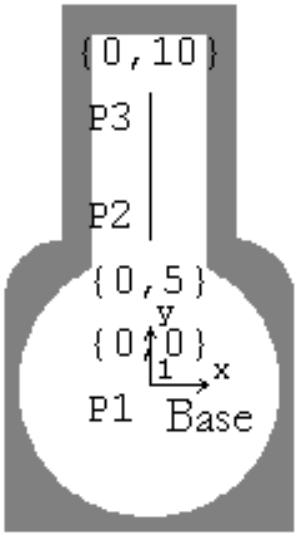


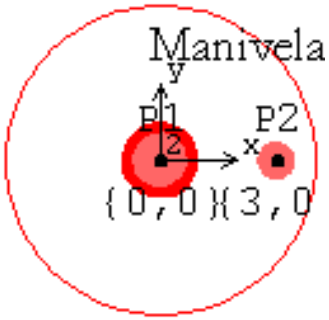


CUERPOS QUE COMPONEN EL SISTEMA MECANICO:

```
base        = 1;
manivela    = 2;
deslizadera = 3;
```

RESTRICCIONES EN "MECHANICA" (8)
COMPRESOR 2D, 3 CUERPOS



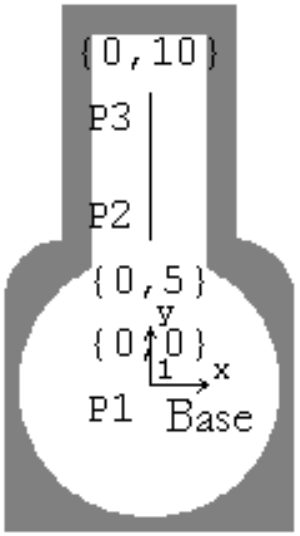


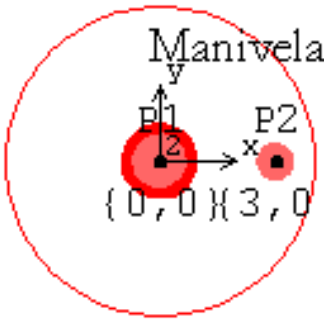
DEFINICION DE CADA CUERPO MEDIANTE PUNTOS:

```
bd[base] = Body[base, PointList->{
  (*P1*) { 0, 0 },
  (*P2*) { 0, 5 },
  (*P3*) { 0, 10}}];
bd[manivela] = Body[manivela, PointList->{
  (*P1*) { 0, 0 },
  (*P2*) { 3, 0 }}];
bd[deslizadera] = Body[deslizadera, PointList->{
  (*P1*) { 0, 0 },
  (*P2*) { 0, 3 }},
  InitialGuess->{{0,5},0}];
```

```
SetBodies[bd[base],bd[manivela],bd[deslizadera]]
```

RESTRICCIONES EN "MECHANICA" (9)
Revolute2, Translate2, RotationLock1





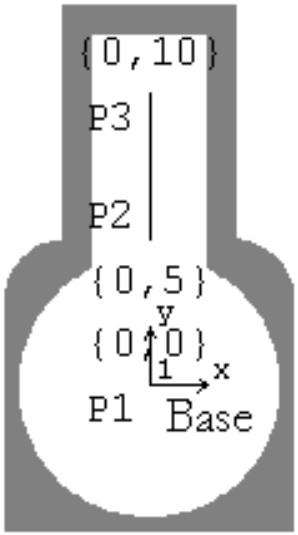
?Revolute2
Revolute2[cnum, point1, point2] models a rotational joint. point1 is constrained to be coincident with point2. Constrains 2 DOF.
cs[1] = Revolute2[1, Point[manivela,1], Point[base,1]];
?RelativeDistance1
RelativeDistance1[cnum, point1, point2, dist] models a connecting link. The absolute distance from point1 to point2 is constrained to be equal to dist. Constrains 1 DOF.
cs[2] = RelativeDistance1[2, Point[manivela,2], Point[deslizadera,1], 5.0];
?Translate2
Translate2[cnum, axis1, axis2] constrains axis1 to be parallel and coincident with axis2. Constrains 2 DOF.
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2]];
?RotationLock1
RotationLock1[cnum, bnum1, bnum2, ang] constrains the angular coordinate of body bnum1 to be ang units greater than the angular coordinate of body bnum2. If bnum2 is omitted, bnum1 is oriented relative to the global coordinate system. ang defaults to zero. Constrains 1 DOF.
cs[4] = RotationLock1[4, manivela, 2 Pi T];

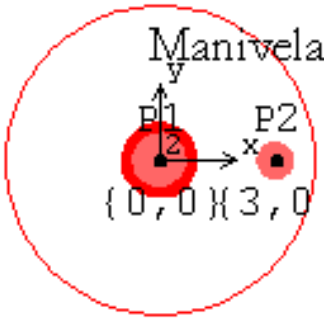
RESTRICCIONES EN "MECHANICA" (10)
Revolute2, RelativeDistance1, Translate2, RotationLock1

DEFINICION DE LAS RESTRICCIONES:

```
cs[1] = Revolute2[1, Point[manivela,1], Point[base,1] ];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                             Point[deslizadera,1], 5.0 ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];

SetConstraints[ cs[1], cs[2], cs[3], cs[4] ];
```





```
?Revolute2
Revolute2[cnum, point1, point2] models a rotational
   joint.  point1 is constrained to be coincident
   with point2.  Constrains 2 DOF.
cs[1] = Revolute2[1, Point[deslizadera,1], Point[base,1] ];
?RelativeDistance1
RelativeDistance1[cnum, point1, point2, dist] models
   a connecting link.  The absolute distance from
   point1 to point2 is constrained to be equal to
   dist.  Constrains 1 DOF.
cs[2] = RelativeDistance1[2, Point[manivela,2],
                             Point[deslizadera,1], 5.0 ];
?Translate2
Translate2[cnum, axis1, axis2] constrains axis1 to
   be parallel and coincident with axis2.
   Constrains 2 DOF.
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
?RotationLock1
RotationLock1[cnum, bnum1, bnum2, ang] constrains
   the angular coordinate of body bnum1 to be ang
   units greater than the angular coordinate of body
   bnum2.  If bnum2 is omitted, bnum1 is oriented
   relative to the global coordinate system.  ang
   defaults to zero.  Constrains 1 DOF.
cs[4] = RotationLock1[4, manivela, 2 Pi T];
```

RESTRICCIONES EN “MECHANICA” (11)
Revolute2, RelativeDistance1, Translate2, RotationLock1

SISTEMA DE ECUACIONES DE RESTRICCION (INCLUYENDO LAS CINEMATICAS Y LAS DE CONDUCCION) A RESOLVER PARA OBTENER EL VECTOR DE COORDENADAS GENERALIZADAS EN CADA INSTANTE:

MatrixForm[Constraints[All]]

X2 = 0

Y2 = 0

$$-25. + (X2 - X3 + 3\,\text{Cos}[Th2])^2 + (Y2 - Y3 + 3\,\text{Sin}[Th2])^2 = 0$$

15 Sin[Th3] = 0

3 X3 Cos[Th3] - 3 (5 - Y3) Sin[Th3] = 0

-2 Pi T + Th2 = 0

SolveMech[.0]

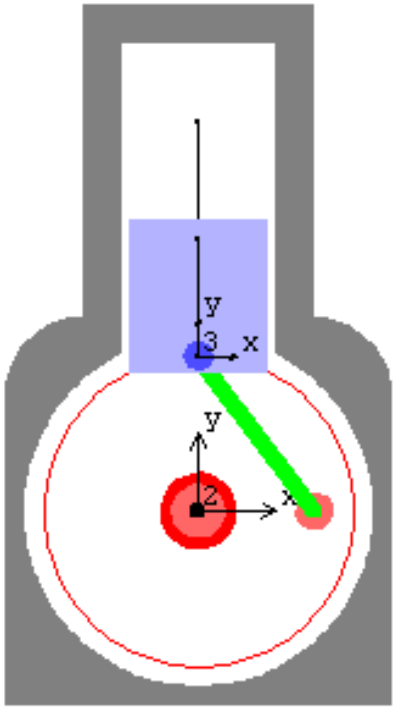
{T -> 0., X2 -> 0., Y2 -> 0., Th2 -> 0., X3 -> 0., Y3 -> 4., Th3 -> 0.}

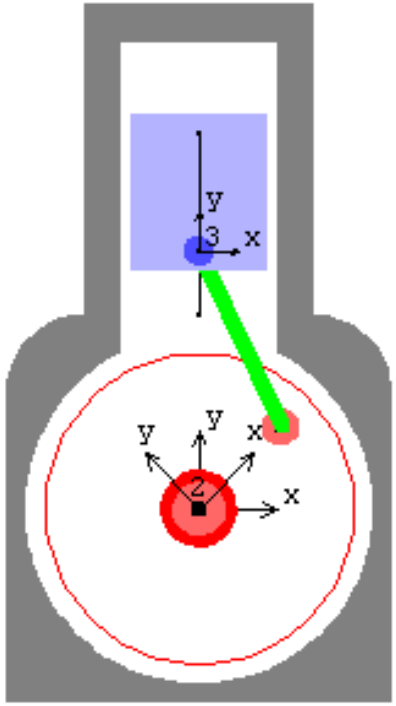
RESTRICCIONES EN “MECHANICA” (12)
Revolute2, RelativeDistance1, Translate2, RotationLock1

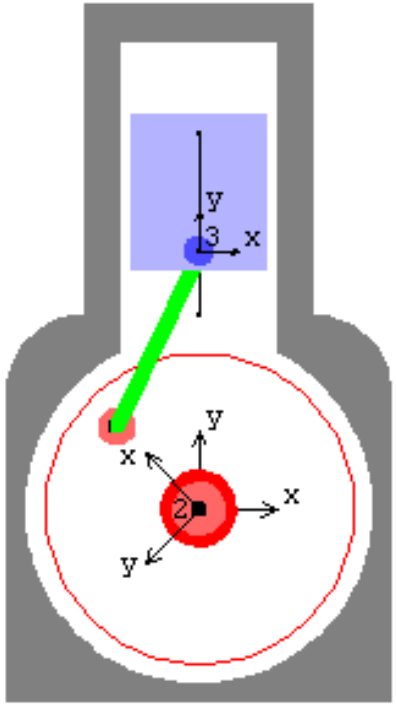
MatrixForm[SolveMech[.0]]
T -> 0.
X2 -> 0.
Y2 -> 0.
Th2 -> 0.
X3 -> 0.
Y3 -> 4.
Th3 -> 0.

MatrixForm[SolveMech[.125]]
T -> 0.125
X2 -> 0.
Y2 -> 0.
Th2 -> 0.785398
X3 -> 0.
Y3 -> 6.64901
Th3 -> 0.

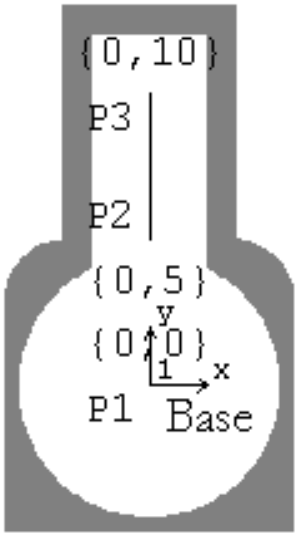
MatrixForm[SolveMech[.175]]
T -> 0.175
X2 -> 0.
Y2 -> 0.
Th2 -> 1.09956
X3 -> 0.
Y3 -> 7.48395
Th3 -> 0.







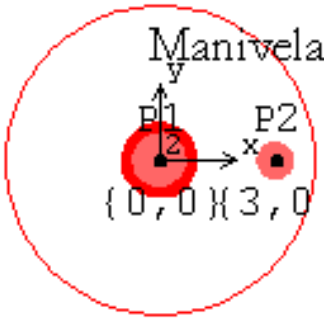
RESTRICCIONES EN "MECHANICA" (13)
OriginLock2, RelativeDistance1, Translate2, RotationLock1



?OriginLock2

OriginLock2[cnum, bnum1, bnum2] locks the position of body bnum1 to that of body bnum2. The local origins of the two bodies are coincident. If bnum2 is omitted, bnum1 is located relative to the global origin. Constrains 2 DOF.

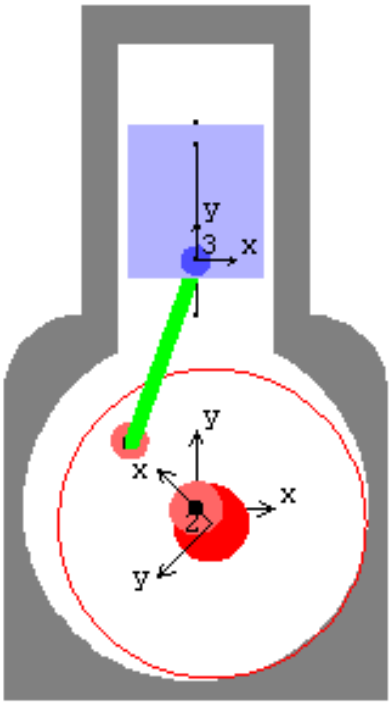
```
xoff =.
yoff =.
cs[1] = OriginLock2[1, manivela, {xoff, yoff}];
cs[1] = Revolute2[1, Point[manivela,1], Point[base,1] ];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                             Point[deslizadera,1], 5.0 ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] ];
MatrixForm[Constraints[1]]
-xoff + X2 = 0
-yoff + Y2 = 0
```



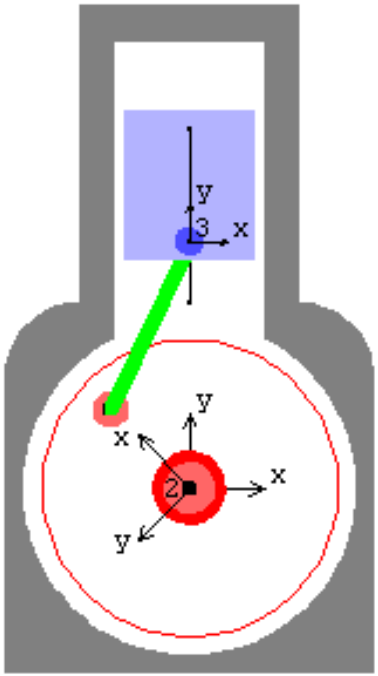
```
?OriginLock2
OriginLock2[cnum, bnum1, bnum2] locks the position of body bnum1
   to that of body bnum2.  The local origins of the two bodies
   are coincident.  If bnum2 is omitted, bnum1 is located
   relative to the global origin.  Constrains 2 DOF.
xoff =0.4.
yoff =-0.4
cs[1] = OriginLock2[1, manivela, {xoff, yoff}];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                             Point[deslizadera,1], 5.0 ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
```

RESTRICCIONES EN “MECHANICA” (14) OriginLock2, RelativeDistance1, Translate2, RotationLock1

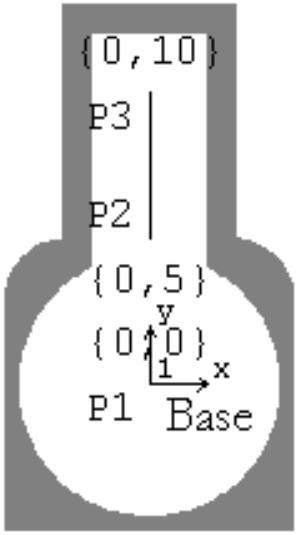
```
xoff=0.4;
yoff=-0.4;
MatrixForm[SolveMech[0.375]]
T -> 0.375
X2 -> 0.4
Y2 -> -0.4
Th2 -> 2.35619
X3 -> 0.
Y3 -> 6.41568
Th3 -> 0.
```




```
xoff=0.0;
yoff=0.0;
MatrixForm[SolveMech[0.375]]
T -> 0.375
X2 -> 0.
Y2 -> 0.
Th2 -> 2.35619
X3 -> 0.
Y3 -> 6.64901
Th3 -> 0.
```



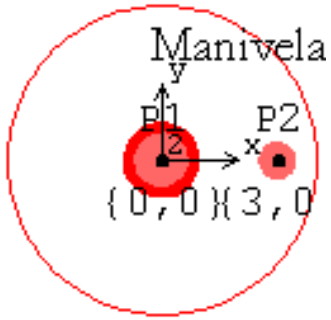
RESTRICCIONES EN "MECHANICA" (15)
DirectedPosition2, RelativeDistance1, Translate2, RotationLock1



?DirectedPosition2

DirectedPosition2[cnum, point1, point2, vector, dist] constrains the location of point1 to be equal to the location of point2 plus a vector that is parallel to the given vector and dist units long. Constrains 2 DOF.

```
xoff =.
yoff =.
cs[1] = DirectedPosition2[1, Point[manivela, 1], {xoff,yoff}];
cs[1] = Revolute2[1, Point[manivela,1], Point[base,1] ];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                             Point[deslizadera,1], 5.0 ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] ];
MatrixForm[Constraints[1]]
-xoff + X2 = 0
-yoff + Y2 = 0
```



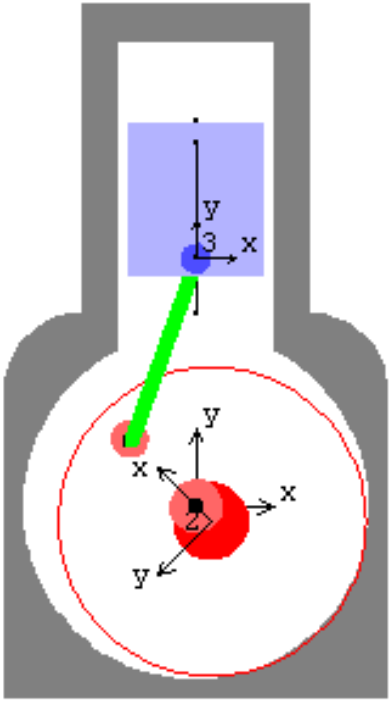
?DirectedPosition2

DirectedPosition2[cnum, point1, point2, vector, dist] constrains the location of point1 to be equal to the location of point2 plus a vector that is parallel to the given vector and dist units long. Constrains 2 DOF.

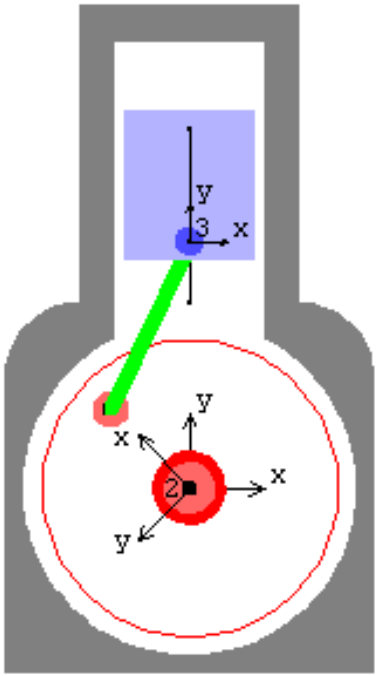
```
xoff =.
yoff =.
cs[1] = DirectedPosition2[1, Point[manivela, 1], {xoff,yoff}];
cs[2] = RelativeDistance1[2, Point[base,2],
                             Point[deslizadera,1], 5.0 ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
```

RESTRICCIONES EN “MECHANICA” (16) DirectedPosition2, RelativeDistance1, Translate2, RotationLock1

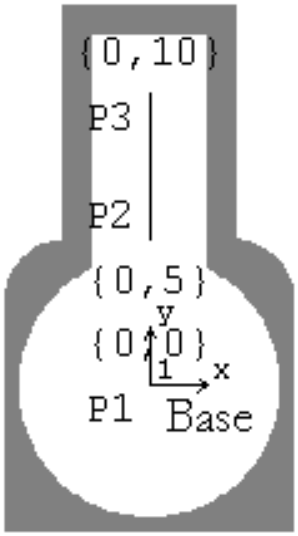
```
xoff=0.4;
yoff=-0.4;
MatrixForm[SolveMech[0.375]]
T -> 0.375
X2 -> 0.4
Y2 -> -0.4
Th2 -> 2.35619
X3 -> 0.
Y3 -> 6.41568
Th3 -> 0.
```



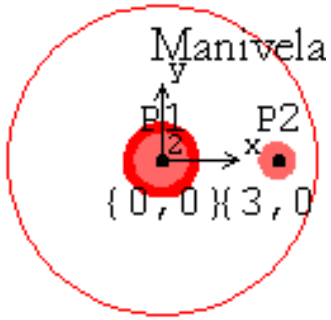

```
xoff=0.0;
yoff=0.0;
MatrixForm[SolveMech[0.375]]
T -> 0.375
X2 -> 0.
Y2 -> 0.
Th2 -> 2.35619
X3 -> 0.
Y3 -> 6.64901
Th3 -> 0.
```



RESTRICCIONES EN "MECHANICA" (17)
DirectedPosition2, RelativeDistance1, Translate2, RotationLock1

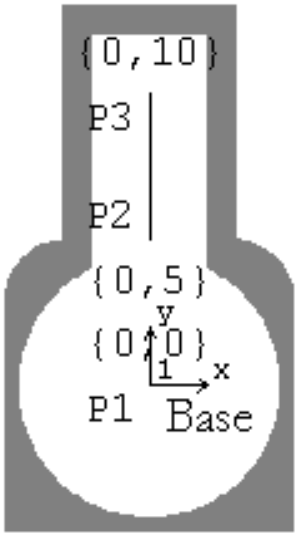



```
xoff =.
yoff =.
cs[1] = DirectedPosition2[1, Point[manivela, 1], {xoff,yoff}];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                             Point[deslizadera,1], 5.0 ];
?Translate2
Translate2[cnum, axis1, axis2] constrains axis1 to be parallel
   and coincident with axis2.  Constrains 2 DOF.
cs[3] = Translate2[3, Line[Point[base, 2], Point[base, 3]],
            Axis[Point[deslizadera, 1], Vector[deslizadera, 2]] ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] ];
MatrixForm[Constraints[3]]
15 Sin[Th3] = 0
3 X3 Cos[Th3] - 3 (5 - Y3) Sin[Th3] = 0
```



```
xoff =.
yoff =.
cs[1] = DirectedPosition2[1, Point[manivela, 1], {xoff,yoff}];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                           Point[deslizadera,1], 5.0 ];
?Translate2
Translate2[cnum, axis1, axis2] constrains axis1 to be parallel
   and coincident with axis2.  Constrains 2 DOF.
cs[3] = Translate2[3, Line[Point[base, 2], Point[base, 3]],
            Axis[Point[deslizadera, 1], Vector[deslizadera, 2]] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
```

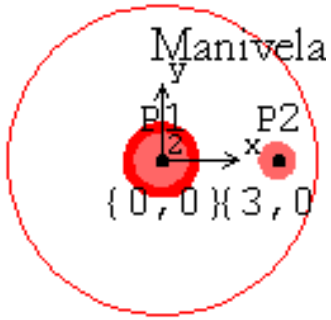
RESTRICCIONES EN "MECHANICA" (18)
PointOnLines2, RelativeDistance1, Translate2, RotationLock1



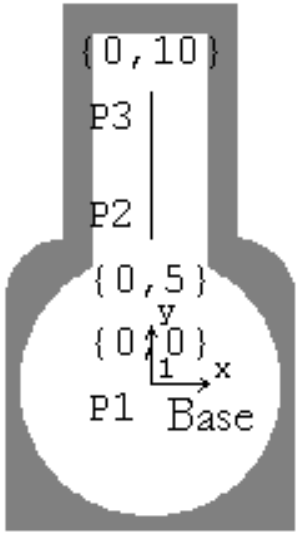
?PointOnLines2

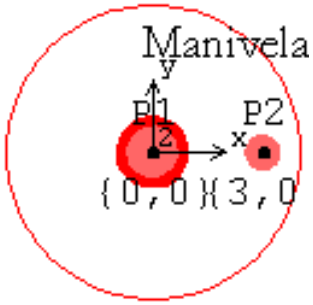
PointOnLines2[cnum, point, axis1, axis2] constrains the point to lie at the intersection of axis1 and axis2. Constrains 2 DOF.

```
cs[1] = PointOnLines2[1, Point[manivela, 1],Line[base, 1, {2.0, 0}],
                                                Line[base, 1, {0, 2.0}]];
cs[1] = Revolute2[1, Point[manivela,1], Point[base,1] ];
cs[2] = RelativeDistance1[2, Point[manivela,2],
                                Point[deslizadera,1], 5.0 ];
cs[3] = Translate2[3, Line[Point[base, 2], Point[base, 3]],
            Axis[Point[deslizadera, 1], Vector[deslizadera, 2]] ];
cs[4] = RotationLock1[4, manivela, 2 Pi T];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] ];
MatrixForm[Constraints[1]]
-2. Y2 = 0
 2. X2 = 0
```



RESTRICCIONES EN "MECHANICA" (19)
OriginLock2, RotationLock1,RelativeDistance1, PointOnLine1, RotationLock1





?PointOnLine1

PointOnLine1[cnum, point, axis] models a planar slider in which the point is constrained to lie on the axis. The axis may be a Mech2D Line or Axis object. Constrains 1 DOF.

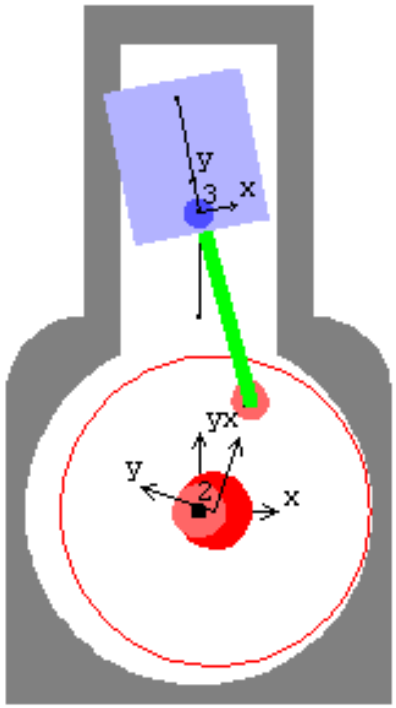
?RotationLock1

RotationLock1[cnum, bnum1, bnum2, ang] constrains the angular coordinate of body bnum1 to be ang units greater than the angular coordinate of body bnum2. If bnum2 is omitted, bnum1 is oriented relative to the global coordinate system. ang defaults to zero. Constrains 1 DOF.

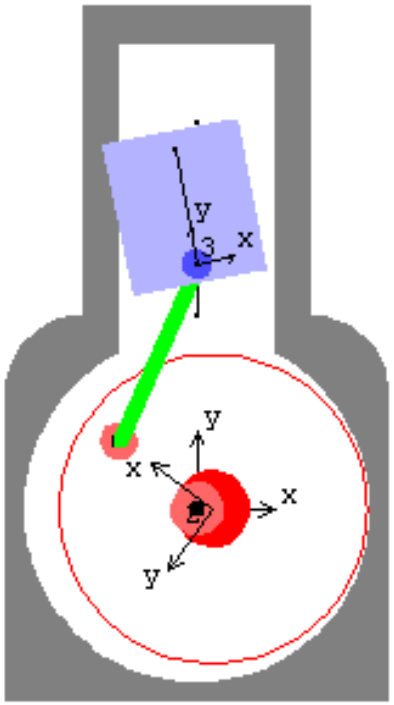
```
xoff =.
yoff =.
inclinacion =.
cs[1] = OriginLock2[1, manivela, {xoff, yoff} ];
cs[2] = RotationLock1[2, manivela, 2 Pi T];
cs[1] = Revolute2[1, Point[manivela,1], Point[base,1] ];
cs[3] = RelativeDistance1[3, Point[manivela, 2],
                             Point[deslizadera, 1], 5.0 ];
cs[4] = PointOnLine1[4, Point[deslizadera, 1], Line[base, 2, 3] ];
cs[3] = Translate2[3, Line[base,2,3], Line[deslizadera,1,2] ];
cs[5] = RotationLock1[5, deslizadera, inclinacion ];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] , cs[5]];
MatrixForm[Constraints[2]]
-2 Pi T + Th2
MatrixForm[Constraints[4]]
-5 X3
```

RESTRICCIONES EN “MECHANICA” (20)
OriginLock2, RotationLock1,RelativeDistance1, PointOnLine1, RotationLock1

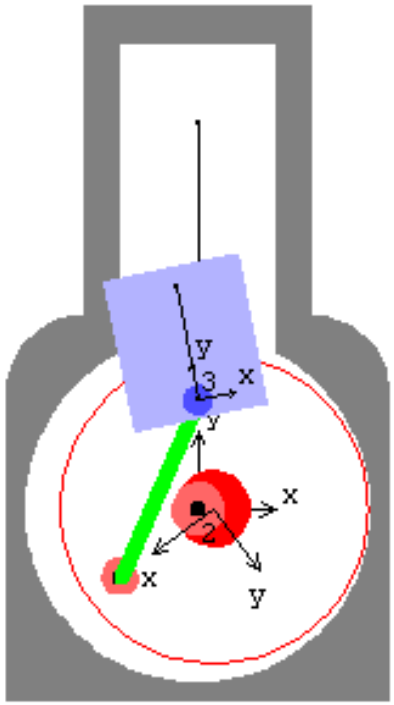
```
xoff = 0.4;
yoff = 0.0;
inclinacion = 0.2;
MatrixForm[SolveMech[0.2]]
T -> 0.2
X2 -> 0.4
Y2 -> 0.
Th2 -> 1.25664
X3 -> 0.
Y3 -> 7.67385
Th3 -> 0.2
```



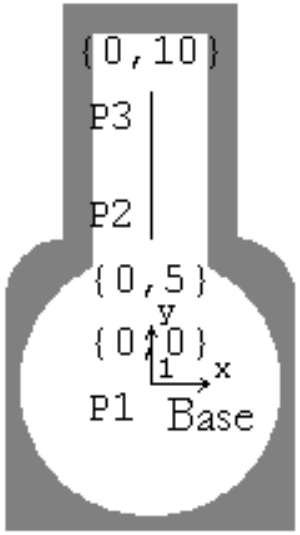

```
xoff = 0.4;
yoff = 0.0;
inclinacion = 0.2;
MatrixForm[SolveMech[0.4]]
T -> 0.4
X2 -> 0.4
Y2 -> 0.
Th2 -> 2.51327
X3 -> 0.
Y3 -> 6.33403
Th3 -> 0.2
```

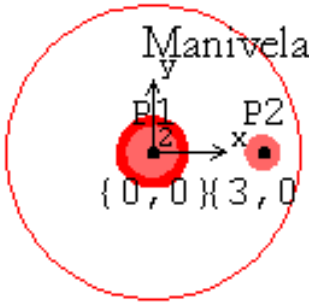



```
xoff = 0.4;
yoff = 0.0;
inclinacion = 0.2;
MatrixForm[SolveMech[0.6]]
T -> 0.6
X2 -> 0.4
Y2 -> 0.
Th2 -> 3.76991
X3 -> 0.
Y3 -> 2.80732
Th3 -> 0.2
```



RESTRICCIONES EN "MECHANICA" (21)
RelativeX1, RelativeY1, RotationLock1,RelativeDistance1, PointOnLine1, RotationLock1





?RelativeX1

RelativeX1[cnum, point1, point2, dist] constrains the X
coordinate of point1 to be dist units greater than the X
coordinate of point2. Constrains 1 DOF.

?RelativeY1

RelativeY1[cnum, point1, point2, dist] constrains the Y
coordinate of point1 to be dist units greater than the Y
coordinate of point2. Constrains 1 DOF.

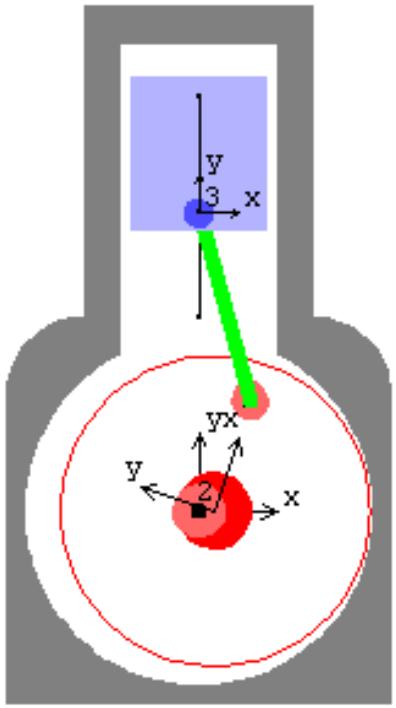
```
xoff =.
yoff =.
cs[1] = RelativeX1[1, Point[manivela, 1], xoff];
cs[2] = RelativeY1[2, Point[manivela, 1], yoff];
cs[1] = OriginLock2[1, manivela, {xoff, yoff} ];
cs[3] = RotationLock1[3, manivela, 2 Pi*T ];
cs[4] = RelativeDistance1[4, Point[manivela, 2],
                             Point[deslizadera, 1], 5];
cs[5] = PointOnLine1[5, Point[deslizadera, 1], Line[base, 2, 3] ];
cs[6] = RotationLock1[6, deslizadera, 0 ];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] , cs[5] , cs[6]];
MatrixForm[Constraints[1]]
-xoff + X2
MatrixForm[Constraints[2]]
-yoff + Y2
```

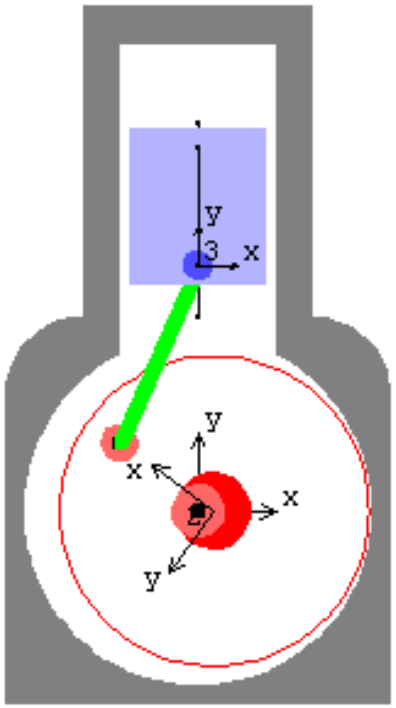
RESTRICCIONES EN “MECHANICA” (22)
RelativeX1, RelativeY1, RotationLock1,RelativeDistance1, PointOnLine1, RotationLock1

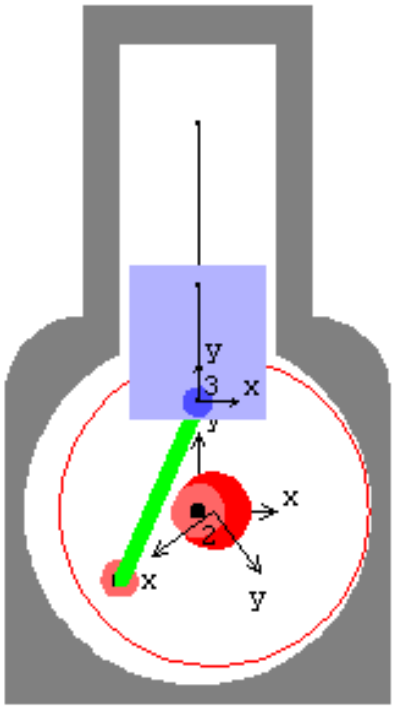
```
xoff = 0.4;
yoff = 0.0;
MatrixForm[SolveMech[0.2]]
T -> 0.2
X2 -> 0.4
Y2 -> 0.
Th2 -> 1.25664
X3 -> 0.
Y3 -> 7.67385
Th3 -> 0.
```

```
xoff = 0.4;
yoff = 0.0;
MatrixForm[SolveMech[0.4]]
T -> 0.4
X2 -> 0.4
Y2 -> 0.
Th2 -> 2.51327
X3 -> 0.
Y3 -> 6.33403
Th3 -> 0.
```

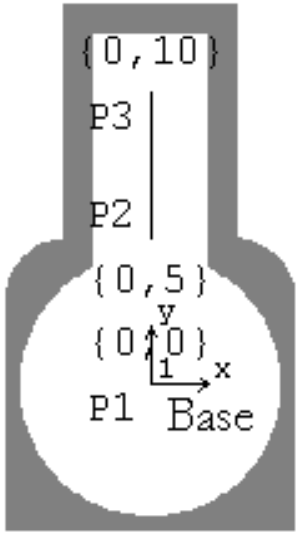
```
xoff = 0.4;
yoff = 0.0;
MatrixForm[SolveMech[0.6]]
T -> 0.6
X2 -> 0.4
Y2 -> 0.
Th2 -> 3.76991
X3 -> 0.
Y3 -> 2.80732
Th3 -> 0.
```

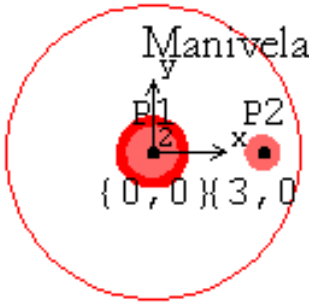






RESTRICCIONES EN "MECHANICA" (23)
OriginLock2, RotationLock1,RelativeDistance1, PointOnLine1, RelativeAngle1

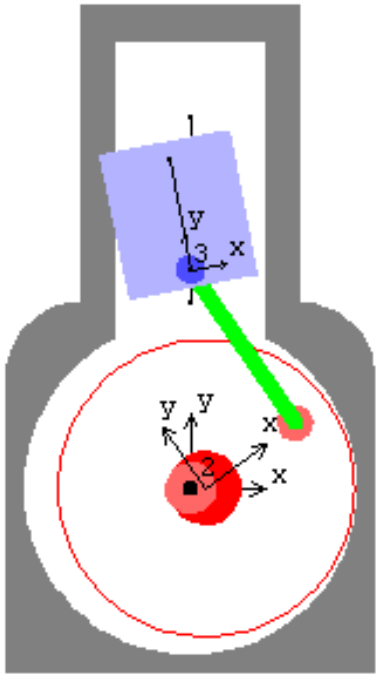




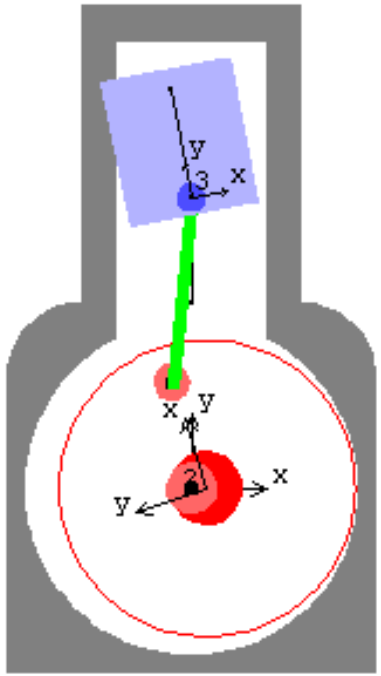

```
?RelativeAngle1
RelativeAngle1[cnum, vector1, vector2 , ang] constrains the angle between
vector1 and vector2 to be equal to ang. Constrains 1 DOF.
xoff =.
yoff =.
inclinacion =.
cs[1] = OriginLock2[1, manivela, {xoff, yoff} ];
cs[2] = RotationLock1[2, manivela, 2 Pi T];
cs[3] = RelativeDistance1[3, Point[manivela, 2],
                             Point[deslizadera, 1], 5];
cs[4] = PointOnLine1[4, Point[deslizadera, 1], Line[base, 2, 3] ];
cs[5] = RelativeAngle1[5, Vector[deslizadera, 2], Line[base, 3, 2],
        inclinacion ];
cs[5] = RotationLock1[5, deslizadera, inclinacion ];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] , cs[5]];
MatrixForm[Constraints[5]]
15 Cos[Th3] Sin[inclinacion] - 15 Cos[inclinacion] Sin[Th3]
```

RESTRICCIONES EN “MECHANICA” (24)
OriginLock2, RotationLock1,RelativeDistance1, PointOnLine1, RelativeAngle1

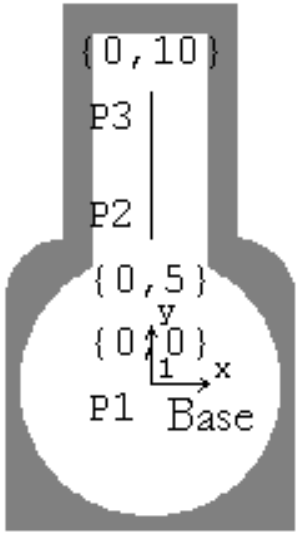
```
xoff = 0.4;
yoff = 0.0;
inclinacion = 0.2;
MatrixForm[SolveMech[0.1]]
T -> 0.1
X2 -> 0.4
Y2 -> 0.
Th2 -> 0.628319
X3 -> 0.
Y3 -> 5.88741
Th3 -> 0.2
```

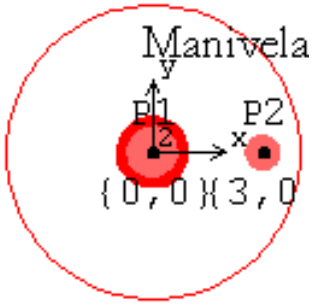



```
xoff = 0.4;
yoff = 0.0;
inclinacion = 0.2;
MatrixForm[SolveMech[0.3]]
T -> 0.3
X2 -> 0.4
Y2 -> 0.
Th2 -> 1.88496
X3 -> 0.
Y3 -> 7.82531
Th3 -> 0.2
```



RESTRICCIONES EN “MECHANICA” (25)
Revolute2, RelativeDistance1, PointOnLine1, Parallel1, RotationLock1





?Parallel1

Parallel1[cnum, vector1, vector2] constrains vector1 to be parallel to vector2. Constrains 1 DOF.

?Orthogonal1

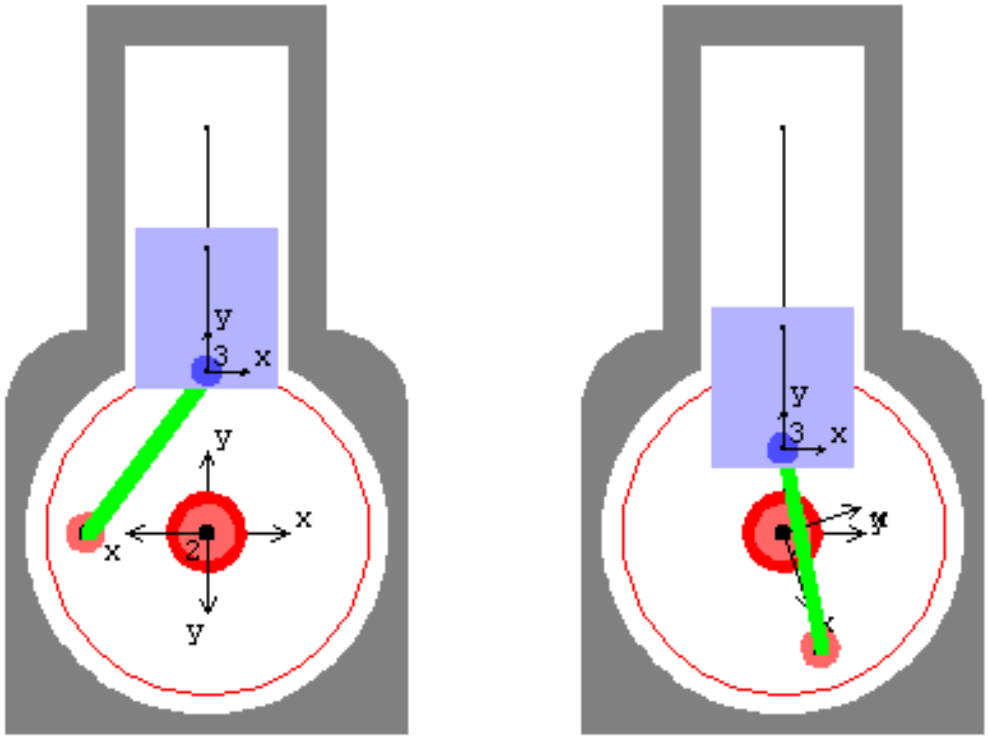
Orthogonal1[cnum, vector1, vector2] constrains vector1 to be orthogonal to vector2. Constrains 1 DOF.

```
cs[1] = Revolute2[1, Point[manivela, 1] ,Point[base, 1]];
cs[1] = OriginLock2[1, manivela, {xoff, yoff} ];
cs[2] = RotationLock1[2, manivela, 2 Pi T];
cs[2] = RelativeDistance1[2, Point[manivela, 2],
                             Point[deslizadera, 1], 5.0 ];
cs[3] = PointOnLine1[3, Point[deslizadera, 1], Line[base, 2, 3] ];
cs[4] = Parallel1[4, Line[deslizadera, 2, 1], Line[base, 3, 2] ];
cs[5] = RotationLock1[5, manivela, base, 2 Pi T ];
cs[5] = RelativeAngle1[5, Vector[deslizadera, 2], Line[base, 3, 2],
        inclinacion ];
SetConstraints[ cs[1], cs[2], cs[3], cs[4] , cs[5]];
MatrixForm[Constraints[4]]
-15 Sin[Th3]
```

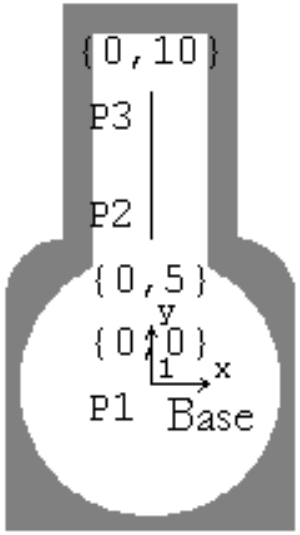
RESTRICCIONES EN “MECHANICA” (26)
Revolute2, RelativeDistance1, PointOnLine1, Parallel1, RotationLock1

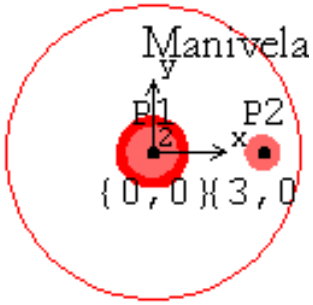
MatrixForm[SolveMech[0.5]]
T -> 0.5
X2 -> 0.
Y2 -> 0.
Th2 -> 3.14159
X3 -> 0.
Y3 -> 4.
Th3 -> 0.

MatrixForm[SolveMech[0.8]]
T -> 0.8
X2 -> 0.
Y2 -> 0.
Th2 -> 5.02655
X3 -> 0.
Y3 -> 2.06014
Th3 -> 0.



RESTRICCIONES EN "MECHANICA" (27)
Revolute2, RelativeDistance1, PointOnLine1, Parallel1, RotationLock1





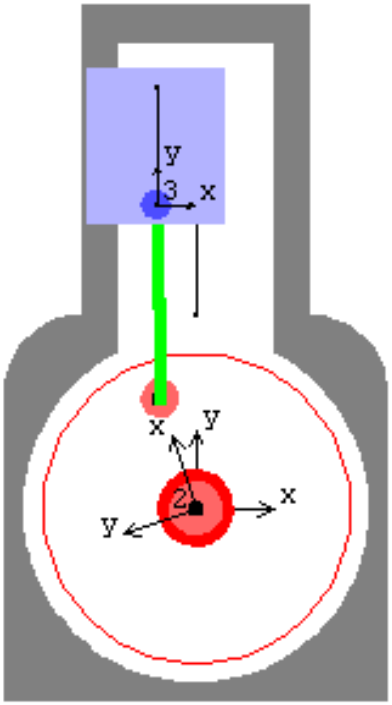
?PointOnLine1

PointOnLine1[cnum, point, axis] models a planar slider in which the point is constrained
to lie on the axis. The axis may be a Mech2D Line or Axis object.
Constrains 1 DOF.

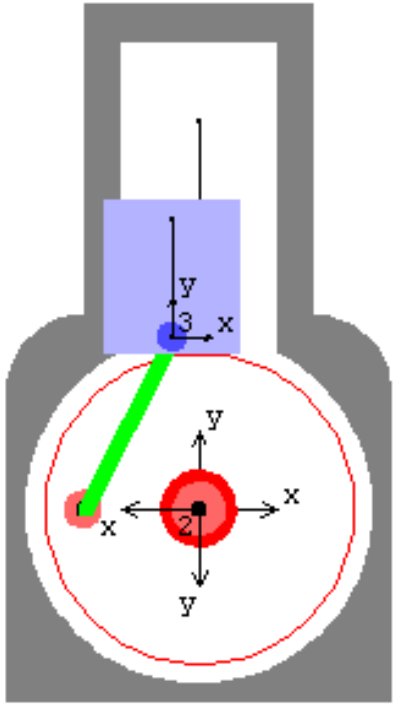
axisoff =.
axisangle =.
cs[1] = Revolute2[1, Point[manivela, 1], Point[base, 1]];
cs[2] = RelativeDistance1[2, Point[manivela, 2],
Point[deslizadera, 1], 5];
cs[3] = PointOnLine1[3, Point[deslizadera, 1], Line[base, 2, 3],
axisoff, axisangle];
cs[4] = Parallel1[4, Line[deslizadera, 2, 1], Line[base, 3, 2]];
cs[4] = Parallel1[4, Line[deslizadera, 2, 1], Line[base, 3, 2]];
cs[5] = RotationLock1[5, manivela, base, 2 Pi T];
SetConstraints[cs[1], cs[2], cs[3], cs[4] , cs[5]];
MatrixForm[Constraints[4]]
-15 Sin[Th3]

RESTRICCIONES EN “MECHANICA” (28)
Revolute2, RelativeDistance1, PointOnLine1, Parallel1, RotationLock1

```
axisoff = 0.75;
axisangle  = 0.10;
MatrixForm[SolveMech[0.3]]
T -> 0.3
X2 -> 0.
Y2 -> 0.
Th2 -> 1.88496
X3 -> -1.03991
Y3 -> 7.8519
Th3 -> 0.
```




```
axisoff = 0.75;
axisangle  = 0.10;
MatrixForm[SolveMech[0.5]]
T -> 0.5
X2 -> 0.
Y2 -> 0.
Th2 -> 3.14159
X3 -> -0.697402
Y3 -> 4.43825
Th3 -> 0.
```




```
axisoff = 0.75;
axisangle  = 0.10;
MatrixForm[SolveMech[0.7]]
T -> 0.7
X2 -> 0.
Y2 -> 0.
Th2 -> 4.39823
X3 -> -0.46535
Y3 -> 2.12547
Th3 -> 0.
```