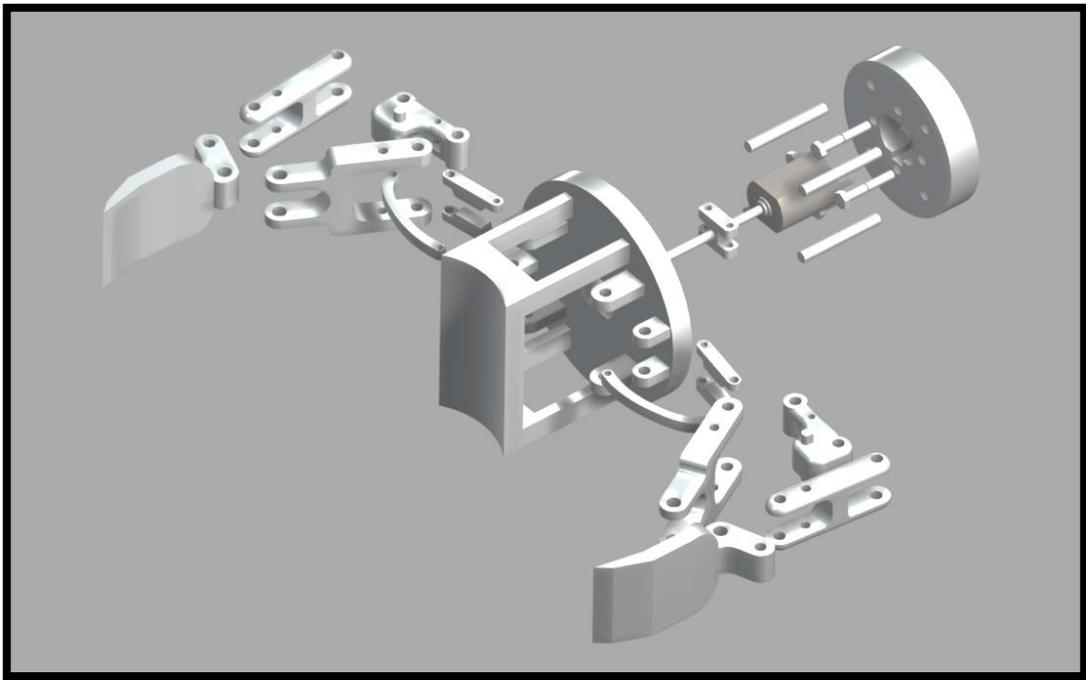


---

# DISEÑO DE UN MANIPULADOR

SISTEMAS DE FABRICACIÓN AUTOMATIZADOS Y ROBÓTICA INDUSTRIAL



IÑIGO MARTÍNEZ A903541

CORO CELIGÜETA A903107

MAITE GÁRATE A903485

JESÚS PAREDES A903237

---

---

# Contenidos

1.	<u>INTRODUCCIÓN</u>	1-3
2.	<u>METODOLOGÍA</u>	2-3
3.	<u>GEOMETRÍA</u>	3-4
3.1.	MÉTODOS NUMÉRICOS EN MATLAB	3-4
3.2.	DETERMINACIÓN DE LOS PUNTOS F Y H DE ANCLAJE	3-6
3.3.	ANÁLISIS CINEMÁTICO Y ESTÁTICO DEL DISEÑO POR GIM	3-6
4.	<u>DISEÑO EN CREO PARAMETRIC 3.0.</u>	4-10
4.1.	ESTUDIO CINEMÁTICO	4-10
4.2.	ESTUDIO DINÁMICO	4-11
5.	<u>ELECCIÓN DE MOTOR</u>	5-12
6.	<u>ANEXOS:</u>	6-14
6.1.	RENDERIZADOS EN CREO PARAMETRIC	6-14
6.2.	CÓDIGO MATLAB	6-14
6.3.	DOCUMENTACIÓN	6-14

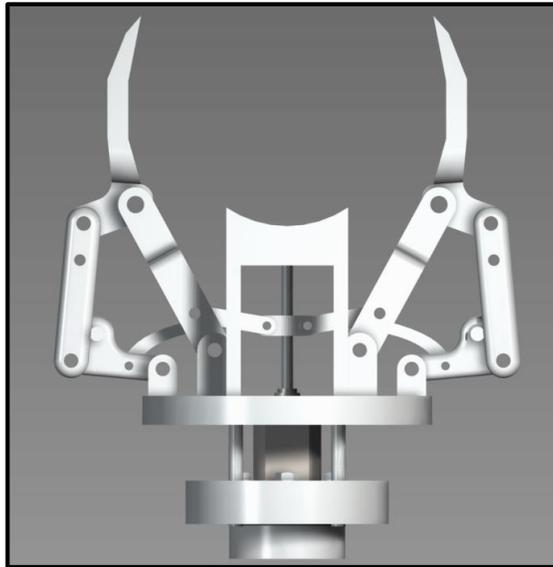
---

## 1. INTRODUCCIÓN

Por el presente documento se pretende diseñar un manipulador para el amarre de piezas cilíndricas. En este caso, las piezas abarcarán diámetros de entre 50 y 80 milímetros, con una longitud de 150 milímetros y un peso de 1 kilogramo.

En el diseño de este amarre, se ha considerado irrelevante la posición inicial de la pieza. El diseño se ha realizado de tal manera que la pieza se puede sujetar tanto si se encuentra situada sobre su base plana, como si se encuentra tumbada sobre una arista lateral.

El robot propuesto para el amarre es el Fanuc LR Mate 200iB, con una conexión eléctrica de 24V y 2A. Estos parámetros serán críticos para la elección del motor que accionará las garras del dispositivo.



## 2. METODOLOGÍA

En primer lugar, se han realizado pequeños bocetos para dar con un diseño básico del manipulador. En esta fase se han determinado los elementos del manipulador, su geometría y las dimensiones básicas para el estudio cinemático.

En segundo lugar, se ha procedido a realizar un cálculo preliminar para la cinemática de la estructura. En este aspecto se ha hecho uso de un programa comercial llamado GIM, que ha aportado resultados satisfactorios. Del mismo modo, se ha utilizado MATLAB y se ha realizado una simulación por métodos numéricos. Mediante este cálculo se ha verificado que la geometría y las dimensiones eran adecuadas para el correcto agarre de la pieza cilíndrica.

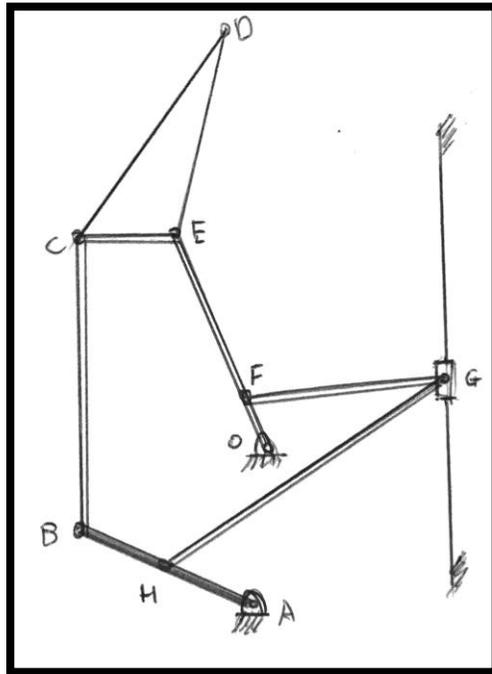
Una vez comprobada la validez del diseño, se ha procedido al diseño asistido por ordenador en el software CREO PARAMETRIC 3.0. Se ha diseñado cada pieza teniendo en cuenta el proceso de montaje y las restricciones del proceso de fabricación.

Finalmente, desde el propio software CAD, se ha realizado un estudio cinemático y un estudio dinámico. Con el estudio de fuerzas, se ha determinado el paso del husillo y par máximo del motor. Con todos los datos, se ha procedido a la elección de un motor ofertado en el mercado.

---

### 3. GEOMETRÍA

En una primera fase de diseño, se han determinado el número de elementos del manipulador y su geometría. El origen del movimiento proviene de una deslizadera o tuerca, que será desplazada por la rotación de un husillo. El husillo tendrá acoplado el motor de corriente continua que se escogerá más adelante.



Como se puede apreciar en la figura, el mecanismo consta de 6 barras articuladas. La pieza cilíndrica se agarrará con el sólido CDE y su simétrico. El plano de simetría es el formado por la guía de la deslizadera. Las barras OE y AB tienen uno de los extremos articulados en la base. Para provocar el movimiento de estas barras, se hace uso de otras dos barras (FG y HG) que se unen con la deslizadera.

En este mecanismo, la distancia entre garras es crítica, puesto que debe ser posible agarrar piezas de diámetros dispares, del de 50 mm hasta 80 mm.

#### 3.1. Métodos numéricos en MATLAB

Una vez definida la geometría, se ha procedido a realizar una simulación por métodos numéricos en MATLAB, para estudiar la cinemática de la geometría. Para ello se ha resuelto los problemas de posición, velocidad, aceleración del mecanismo. En este simulación se han omitido las barras FG y GH, por motivos de simplicidad.

En primer lugar, se ha definido el vector de coordenadas dependientes  $q$  y el vector de coordenadas independientes o grados de libertad  $z$  (que se supone conocido):

$$q = [ x_B; y_B; x_C; y_C; x_E; y_E; x_D; y_D ] \quad ; \quad z = [ \varphi; \theta ]$$

Los ángulos  $\varphi$  y  $\theta$  corresponden a los ángulos que forman las barras OE y AB con la horizontal.

En segundo lugar, se han enumerado las ecuaciones de restricción del mecanismo, que son de dos tipos. Por un lado, ecuaciones de restricción de sólido rígido (para las barras) y de coordenada relativa (para la definición de los ángulos). A modo de ejemplo se incluyen las correspondientes a la barra AB:

$$(x_B - x_A)^2 + (y_B - y_A)^2 - AB^2 = 0$$

$$x_B - AB \cdot \cos \varphi = 0$$

$$y_B - AB \cdot \sin \theta = 0$$

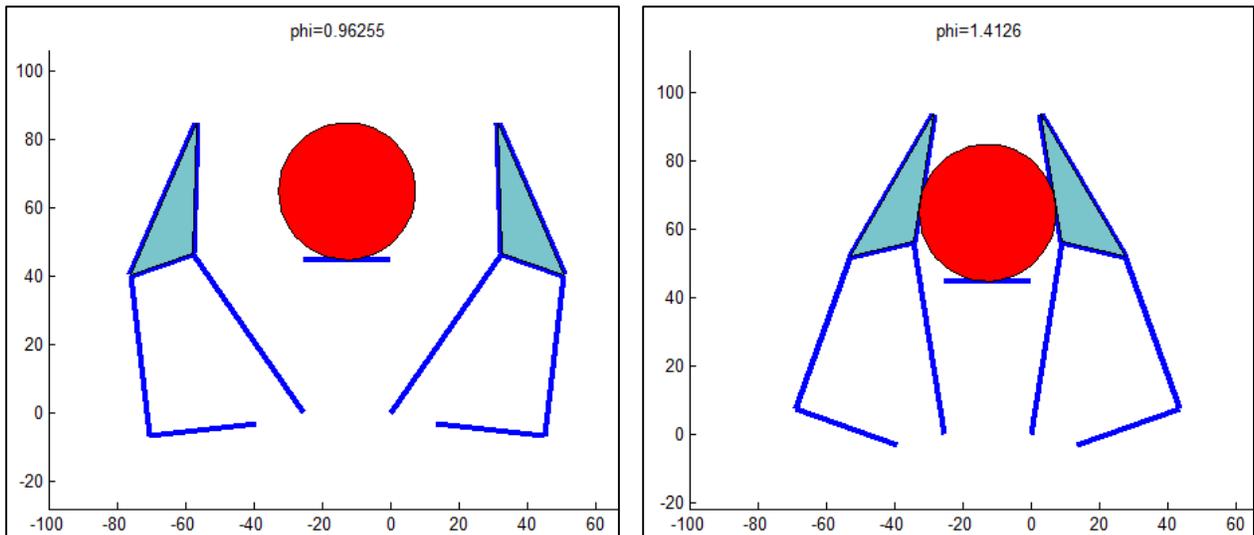
Para realizar la simulación cinemática, es necesario calcular el valor de las ecuaciones de restricción y la matriz jacobiana. Con ello, se sigue el método de Newton-Raphson para resolver la posición del sistema. Posteriormente, para calcular las velocidades se realiza el producto de la jacobiana por el vector  $q$  y se iguala dicho producto a un vector nulo, teniendo en cuenta el valor de los grados de libertad. Finalmente, realizando un planteamiento análogo en aceleraciones, se obtienen las aceleraciones en los puntos indicados en  $q$ .

Cabe señalar que para la simulación es necesario conocer las condiciones iniciales de los grados de libertad y las longitudes de las barras, que se estimaron del boceto inicial:

$$z = [55^\circ; -7^\circ] ; \dot{z} = [2\pi; 0] ; \ddot{z} = [0; 0] ;$$

$$OE = 55; AB = 30; BC = 45; CE = 20; DE = 40; CD = 50;$$

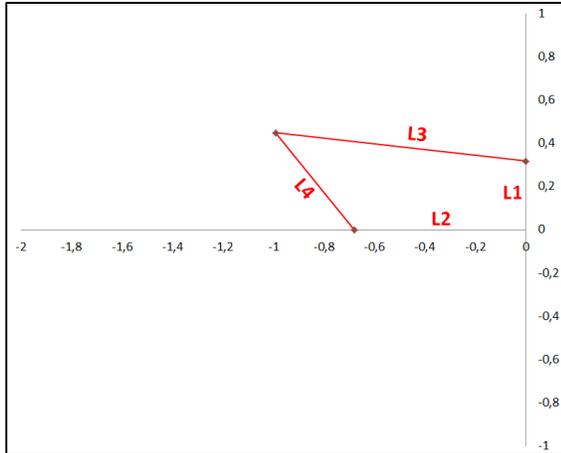
Se recomienda fervientemente que se ejecute el archivo "Mecanismo.m", para visualizar el movimiento del sistema. En este apartado se ha resumido el procedimiento seguido, aunque en el código adjunto se ilustran todos los pasos seguidos. Para esta simulación se ha utilizado el método explicado en el libro "Teoría de Máquinas" 2ª Ed., de Alejo Avello Iturriagoitia, páginas 130-133.



Con esta simulación, se ha visto que la geometría es adecuada para el amarre de la pieza cilíndrica, sea cual sea su diámetro. Además, se ha visto que los ángulos de las barras no varían en gran medida, por lo que el sistema de barras y la deslizadera se consideran una opción factible.

### 3.2. Determinación de los puntos F y H de anclaje

Para determinar los puntos ideales de anclaje sobre las barras OE y AB, se realizó un pequeño estudio cinemático, que se representó por medio de EXCEL. El eje Y representa la guía de la deslizadera y la barra L4 está articulada en el eje X.



La ecuación que rige la cinemática de este sencillo sistema es:

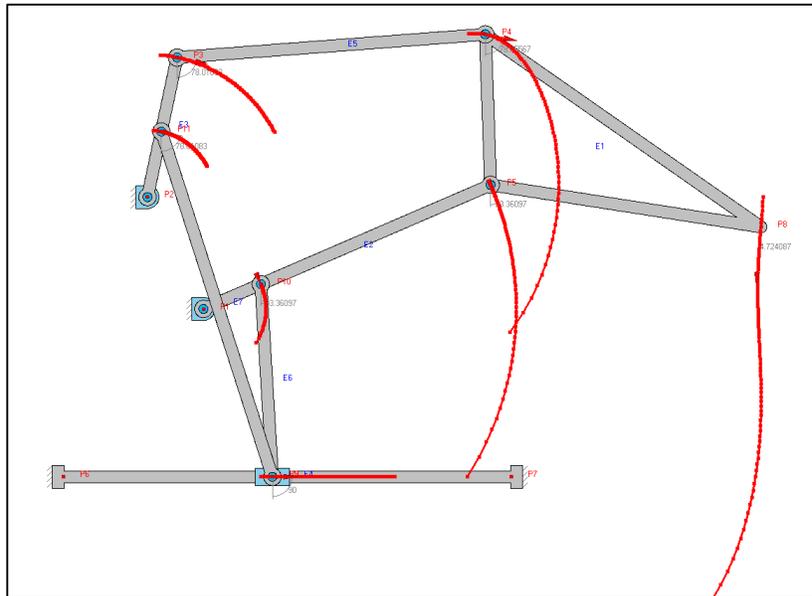
$$\vec{w}_4 \cdot \vec{L}_4 + \vec{w}_3 \cdot \vec{L}_3 = \vec{v}$$

Proyectando esta ecuación en los ejes X e Y, se obtienen los valores de las velocidades angulares  $\vec{w}_3$  y  $\vec{w}_4$  para diferentes valores de la velocidad lineal. Este cálculo se mostrará más adelante, en el apartado de elección del motor. Este cálculo se ha realizado para la juntas en los puntos F y H.

### 3.3. Análisis cinemático y estático del diseño por GIM

Por medio del software libre GIM, se ha pretendido ampliar y concretar la geometría estudiada en el apartado anterior. En este software se trabaja por medio de puntos, por lo que se introdujeron los puntos analizados en MATLAB. Posteriormente se crearon los diferentes elementos: barras, articulaciones y deslizadera.

Una vez definida la geometría se estudió la cinemática del mecanismo, aplicando un desplazamiento conocido en la deslizadera. En la siguiente imagen se pueden visualizar las trayectorias seguidas por los diferentes puntos definidos. Como se puede comprobar en el video "Recording #4", el mecanismo funciona correctamente, tal y como se ha visto en la simulación de MATLAB.

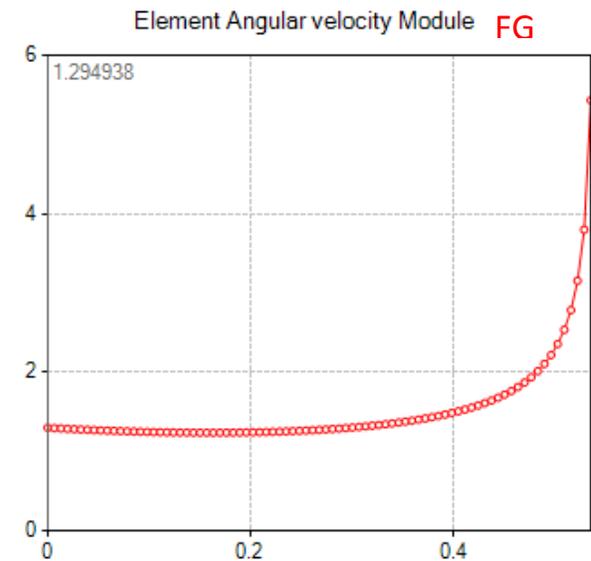
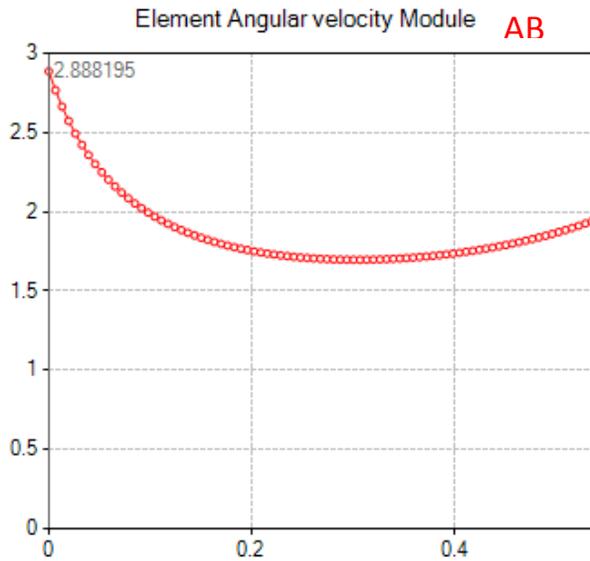
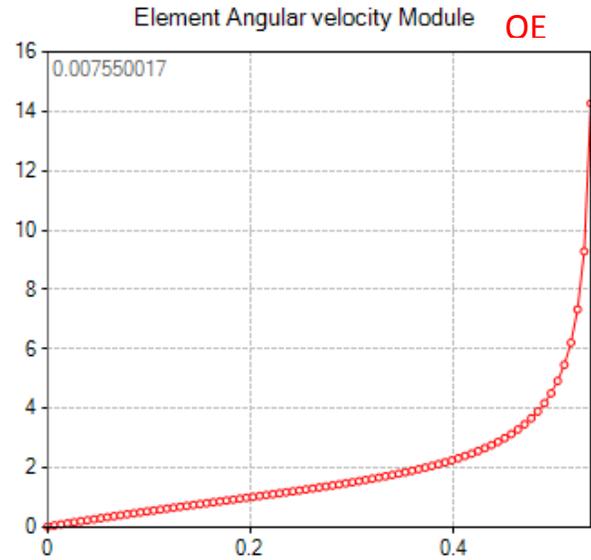
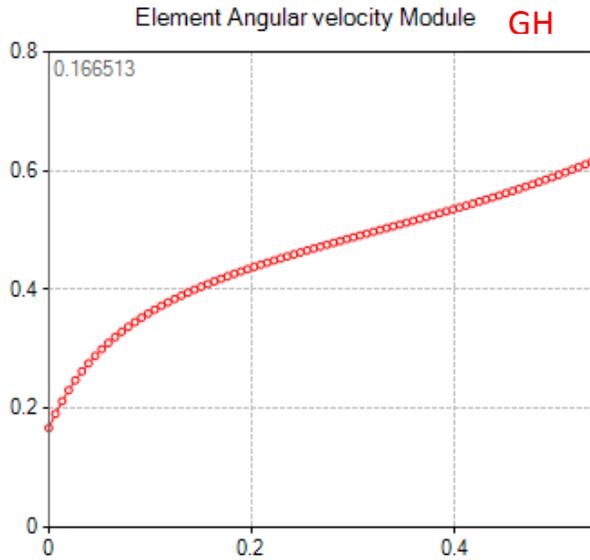


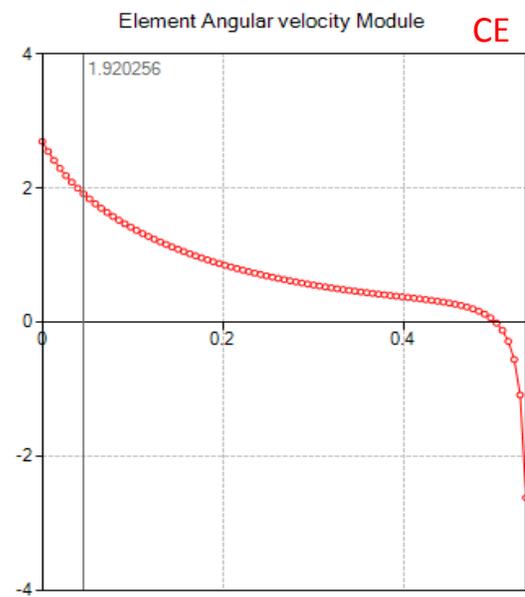
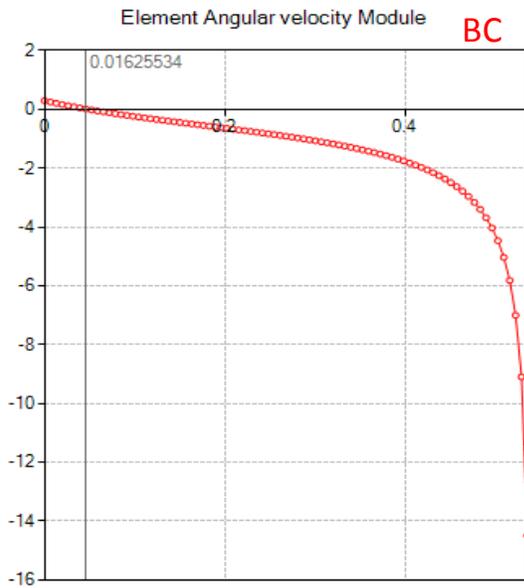
Con la geometría mostrada, se fijaron las dimensiones finales del mecanismo (en mm):

OE=57.15; AB=31.75; BC=47.50; CE=19.00; DE=38.00; CD=48.30;

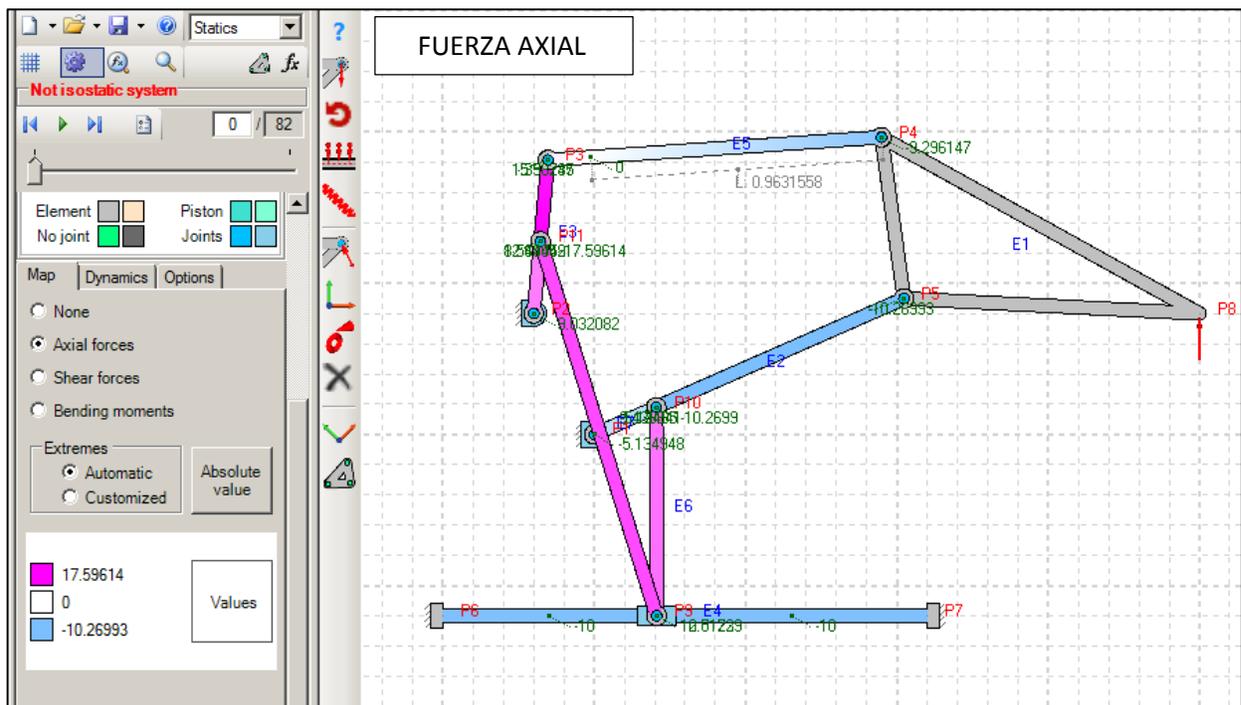
AH=11.75 OF=20

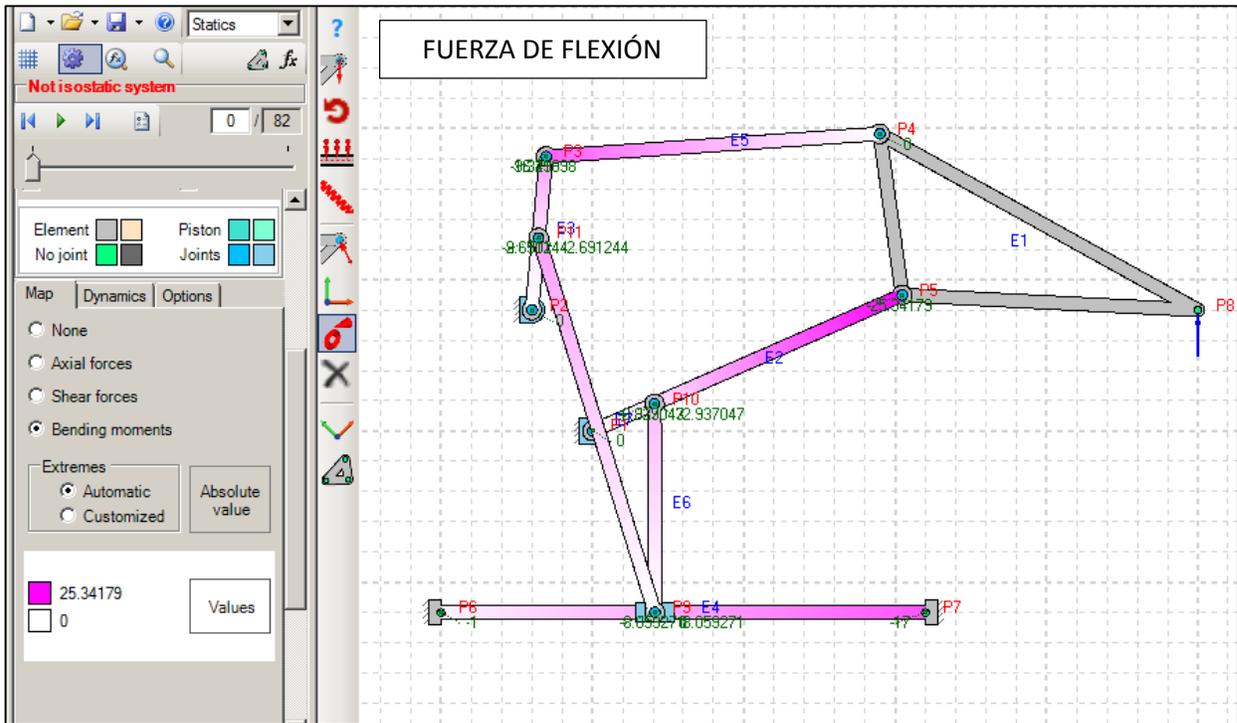
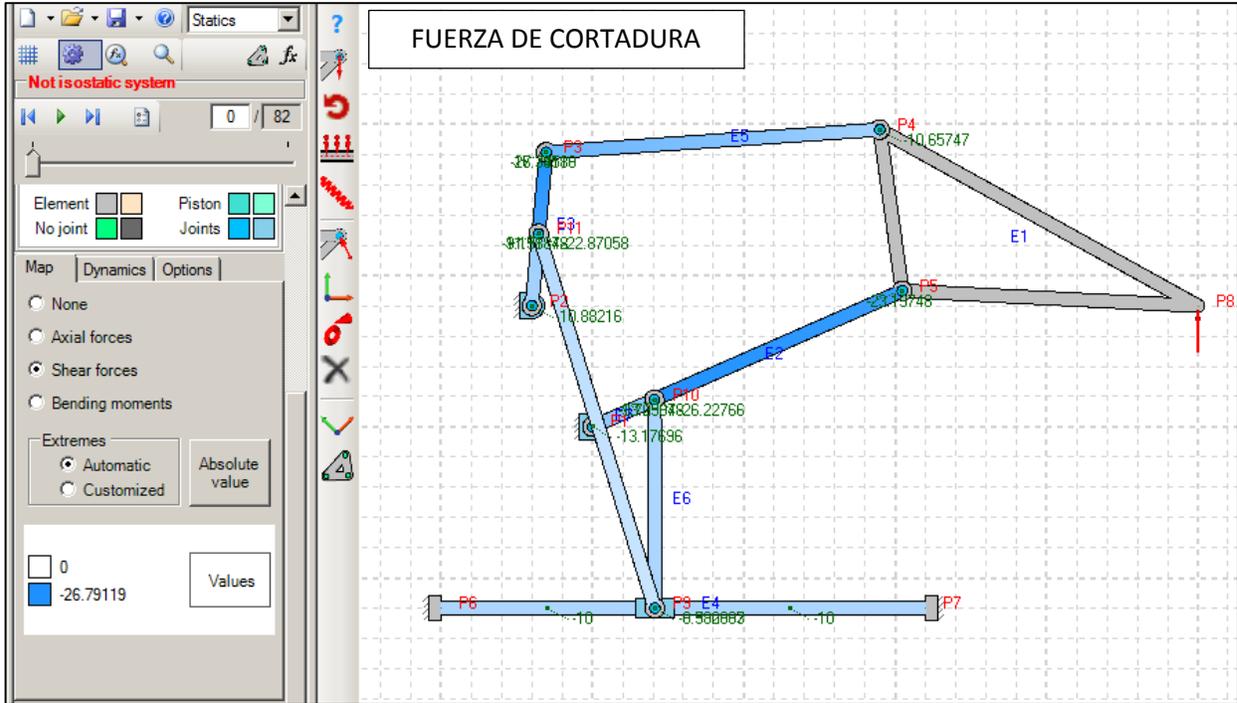
A continuación también se han incluido las velocidades angulares de las barras AB, OE, FG y GH, para comparar con los valores posteriores obtenidos en CREO PARAMETRIC.





Del mismo modo, se han realizado análisis estáticos de la estructura ante una fuerza de 10 N en la punta de la pinza, representando la reacción que le produce la pieza cilíndrica. A continuación se han incluido diagramas de fuerzas axiales, de cortadura y de flexión. Nótese los valores máximos de las fuerzas en la parte izquierda de las imágenes.





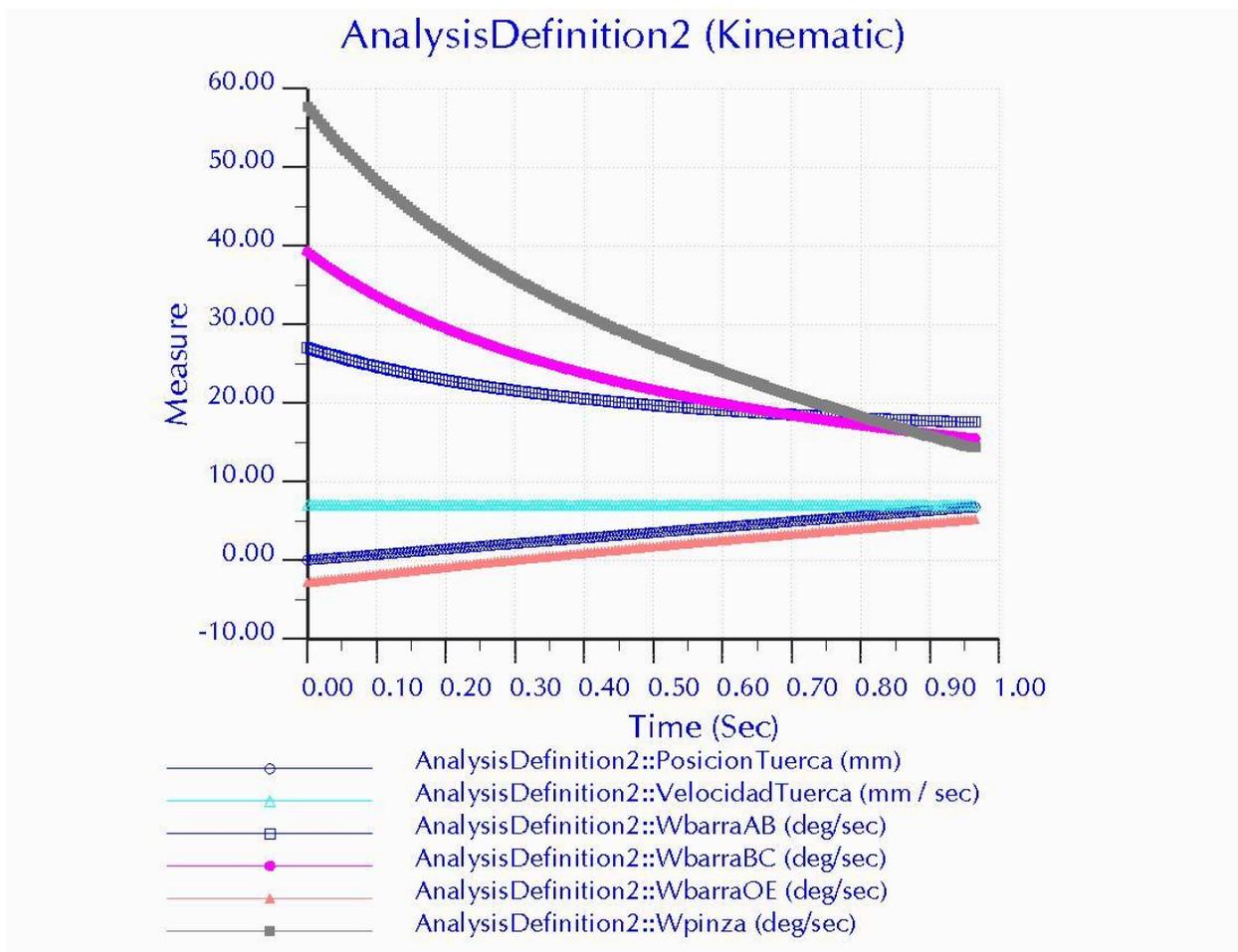
## 4. DISEÑO EN CREO PARAMETRIC 3.0.

Una vez realizados los estudios previos, se procedió al diseño en CAD del manipulador. En primer lugar se introdujo la geometría del robot, para realizar un correcto amarre del manipulador con el robot.

Se decidió diseñar una pieza de unión con agujeros pasante, que alojará la base de motor dc y se atornillará al motor por medio de 4 tornillos M4x25. Sobre esta pieza de unión se ha diseñado una base sobre la que se sitúan las uniones articuladas y el apoyo del husillo. En esta misma pieza se ha colocado un pequeño apoyo circular de  $\varnothing 80$  para la pieza cilíndrica, de manera que al agarrar el cilindro, éste se apoyará en esta base. En las imágenes y planos adjuntos, así como en el assembly "amarre.asm" de CREO PARAMETRIC se pueden comprobar todos los detalles del diseño. En los archivos adjuntos se puede visualizar el movimiento del mecanismo en los videos "Recording #1, #2 y #3".

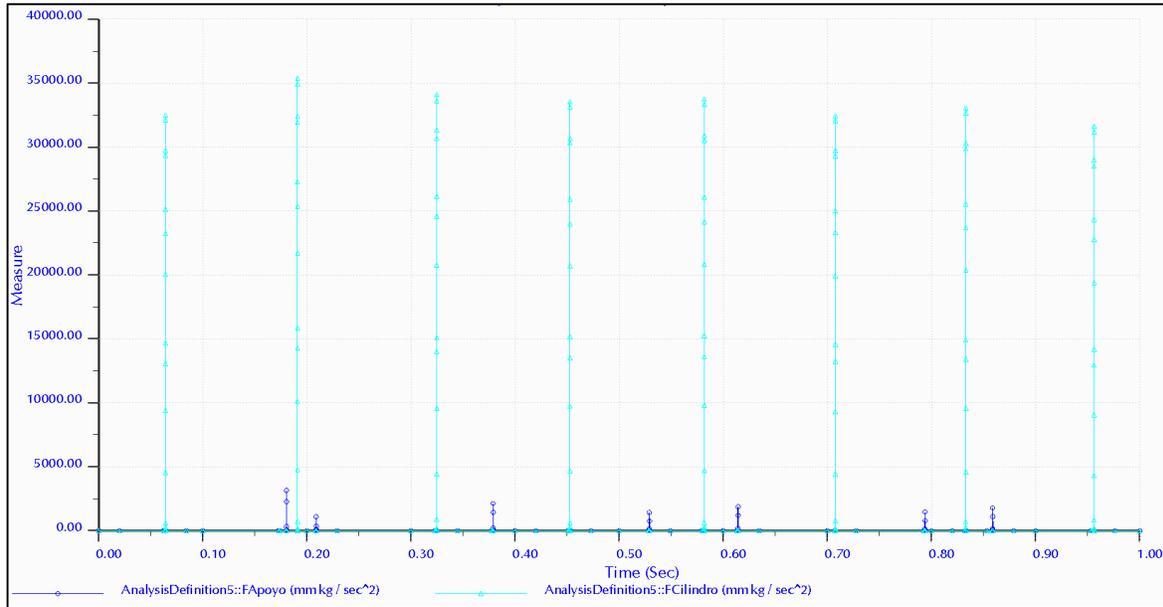
### 4.1. Estudio Cinemático

Desde CREO PARAMETRIC se permite utilizar el módulo de mecanismos, para realizar estudios cinemáticos, estáticos y dinámicos. A continuación se incluye el análisis cinemático de la estructura desde que comienza el movimiento hasta que se contacta con la pieza. A la hora de comparar con las gráficas obtenidas por GIM, cabe considerar únicamente los 0.4 segundos (las velocidades de avance y las escalas no son las mismas, aunque las formas de las curvas son muy similares).

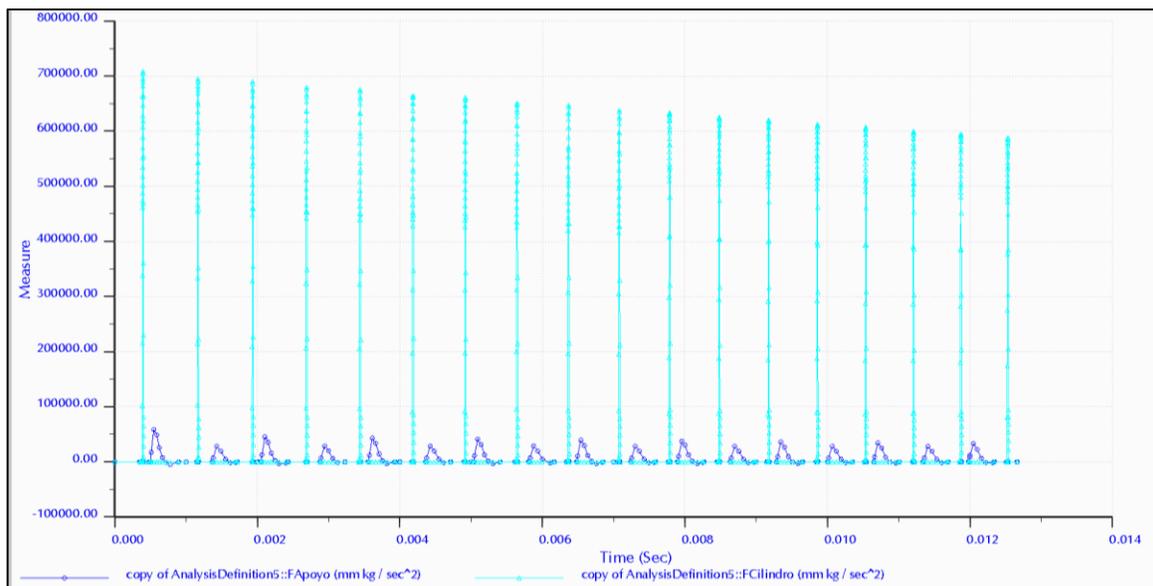


## 4.2. Estudio Dinámico

Para el estudio dinámico, se colocaron las garras del mecanismo en posición muy cercana a la pieza cilíndrica, y se actuó el motor, aplicando una fuerza axial en la deslizadera, y provocando reacciones entre la pieza y el cilindro y entre la pieza y la base circular. A continuación se visualizan las reacciones para el caso de no tener gravedad y para el caso con gravedad aplicada en dirección opuesta al motor, en el caso de que se aplique una fuerza de 42N.



Las reacciones sin gravedad se sitúan en torno a los 35 N entre pieza y garras y en 3N entre pieza y base cilíndrica. En el caso de aplicar la gravedad, estos valores se disparan a 700 N y 50 N respectivamente. Se ha visto que efectivamente la pieza queda perfectamente agarrada entre los tres puntos de unión. La conclusión principal, es que mediante este cálculo, se ha visto que es necesaria al menos una fuerza de **40N** sobre la deslizadera para sujetar correctamente la pieza.



---

## 5. ELECCIÓN DE MOTOR

Con el dato de 40N, ya es posible dimensionar el par necesario en el motor, en función del paso que se escoja en el husillo. Esto viene dado por las siguientes fórmulas:

$$T = \frac{1}{\eta} \cdot F \cdot \frac{p}{2\pi} ; \eta = 0.8$$

Se ha visto que los pares nominales en motores DC de 24V ronda los 5-10 Nmm. Desde el fabricante Bühler se ofrece las siguientes opciones:

<b>Fabricante</b>	Bühler Motor	Bühler Motor
<b>Designación</b>	Motor DC 22 x 48, 1.13.078.0XX, 12 - 24 V	Motor DC 24 flat, 1.16.011.5XX, 12 - 24 V
<b>Referencia</b>	1.13.078.012	1.16.011.532
<b>Par nominal ( N mm)</b>	10	5

<b>T (N*mm)</b>	<b>p (mm/rev)</b>	<b>F (N)</b>
4	0,5	40,21
4	1	20,11
4	2	10,05
4	4	5,03
4	5	4,02
10	0,5	100,5
10	1	50,27
10	2	25,13
10	4	12,57
10	5	10,05

A continuación se incluyen los diferentes valores de las velocidades angulares en función de la velocidad lineal que se aplique a la deslizadera, teniendo en cuenta que:

$$v = w \cdot \frac{p}{2\pi}$$

<b>v</b>	<b>Union H</b>		<b>Union F</b>	
	w4	w3	w4	w3
<b>1</b>	0,27	0,12	3,71	0,01
<b>2</b>	0,54	0,24	7,41	0,01
<b>3</b>	0,81	0,37	11,12	0,02
<b>4</b>	1,08	0,49	14,82	0,03
<b>5</b>	1,35	0,61	18,53	0,03
<b>6</b>	1,62	0,73	22,23	0,04
<b>7</b>	1,89	0,85	25,94	0,05
<b>8</b>	2,16	0,97	29,64	0,05
<b>9</b>	2,43	1,10	33,35	0,06
<b>10</b>	2,70	1,22	37,06	0,07
<b>11</b>	2,97	1,34	40,76	0,08
<b>12</b>	3,24	1,46	44,47	0,08
<b>13</b>	3,51	1,58	48,17	0,09
<b>14</b>	3,78	1,70	51,88	0,10
<b>15</b>	4,05	1,83	55,58	0,10
<b>16</b>	4,32	1,95	59,29	0,11
<b>17</b>	4,59	2,07	62,99	0,12
<b>18</b>	4,86	2,19	66,70	0,12
<b>19</b>	5,13	2,31	70,41	0,13
<b>20</b>	5,40	2,43	74,11	0,14

Por tanto, en este manipulador, se ha escogido el motor DC 22 x 48, 1.13.078.0XX, 12 - 24 V del fabricante Bühler Motor, que nos aporta una fuerza de 50,27 N, suficiente para agarrar correctamente la pieza cilíndrica propuesta.

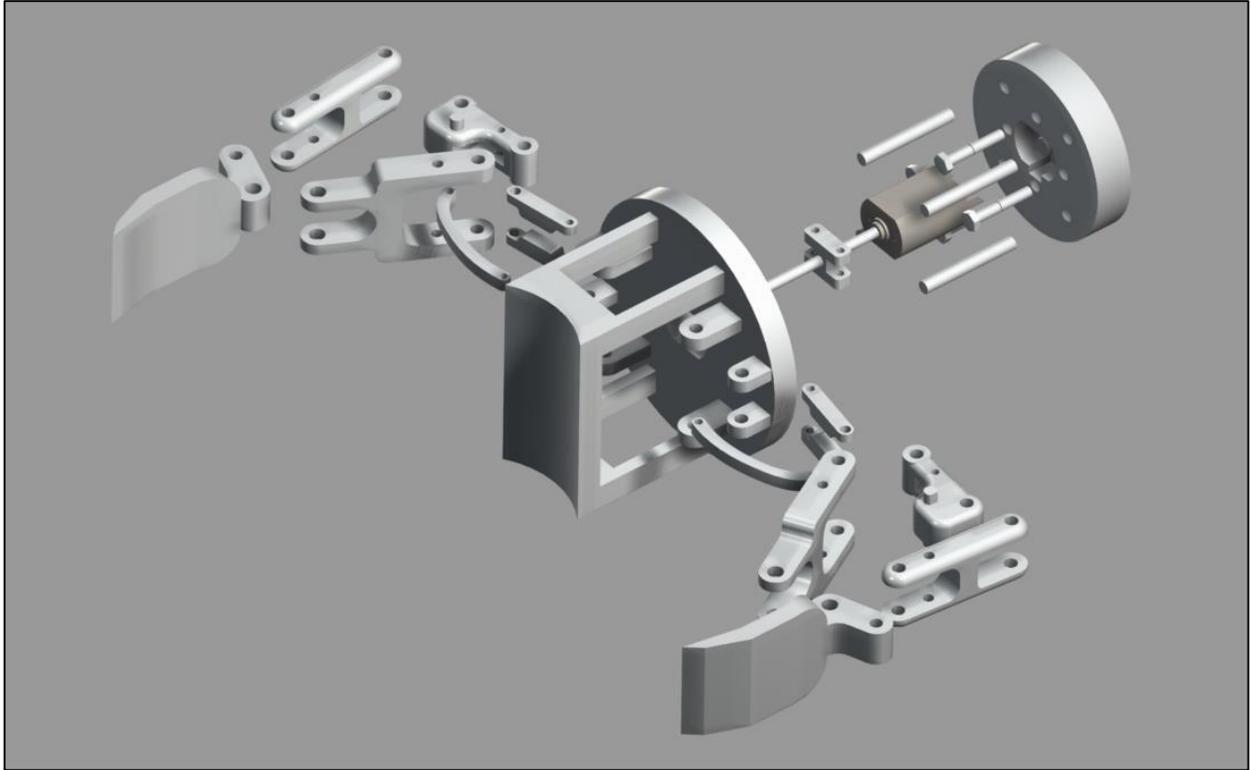
---

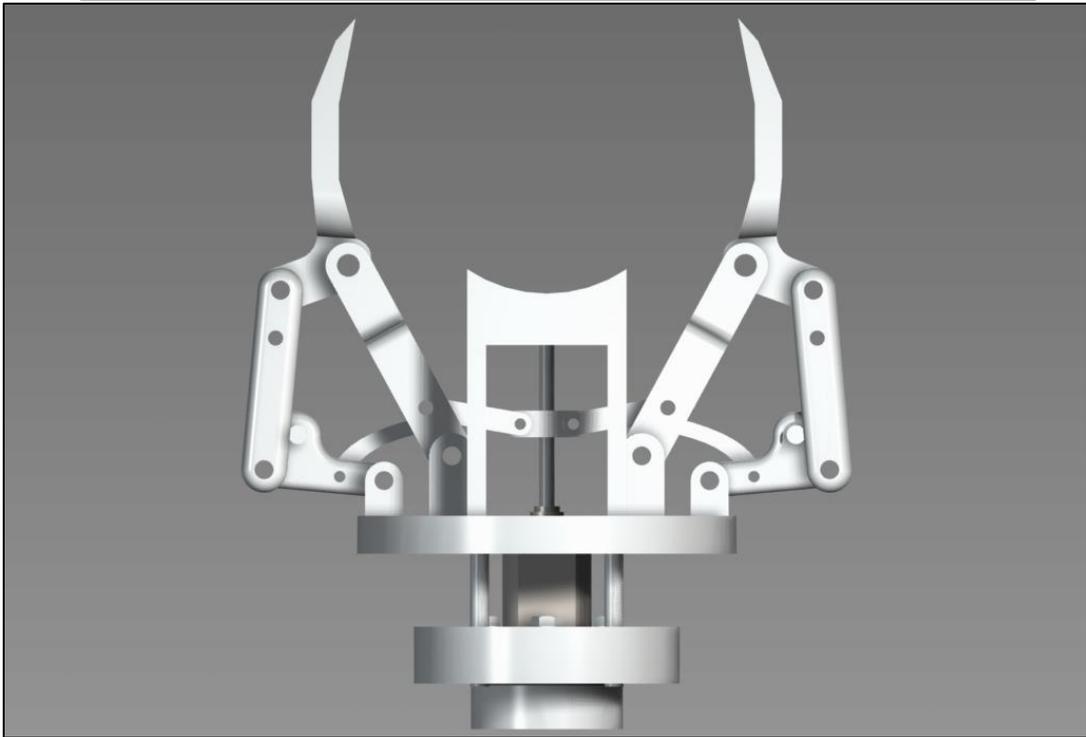
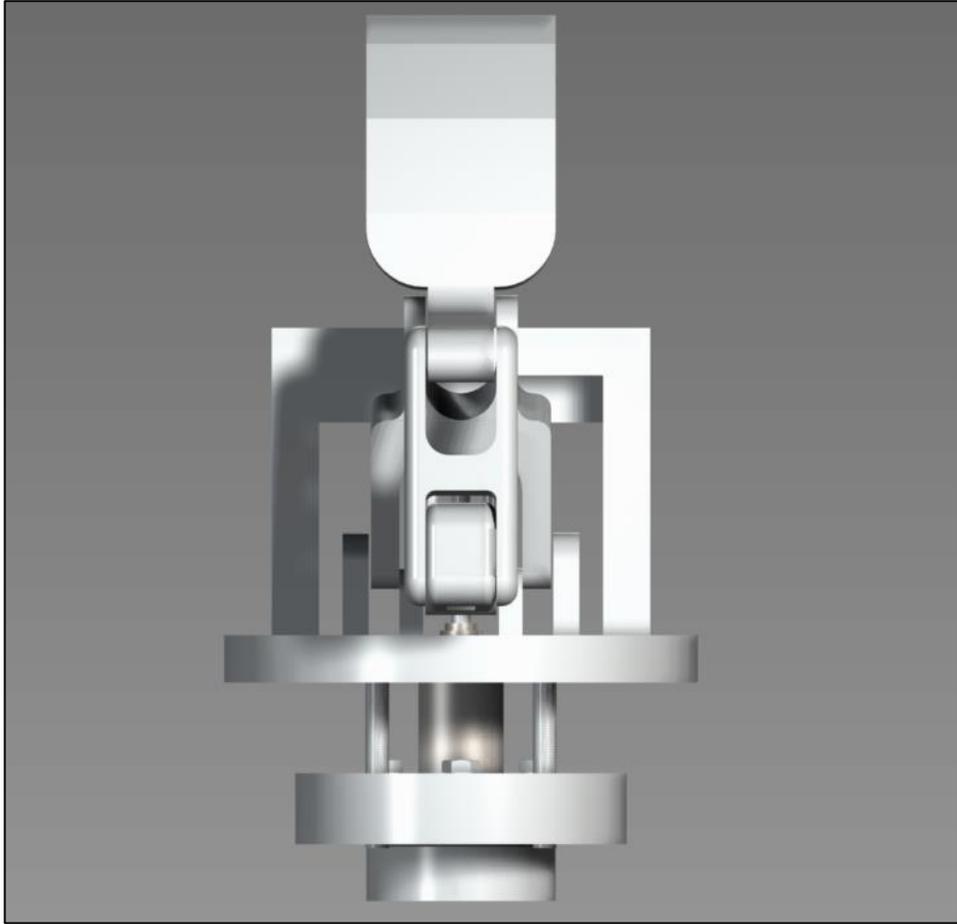
## 6. ANEXOS:

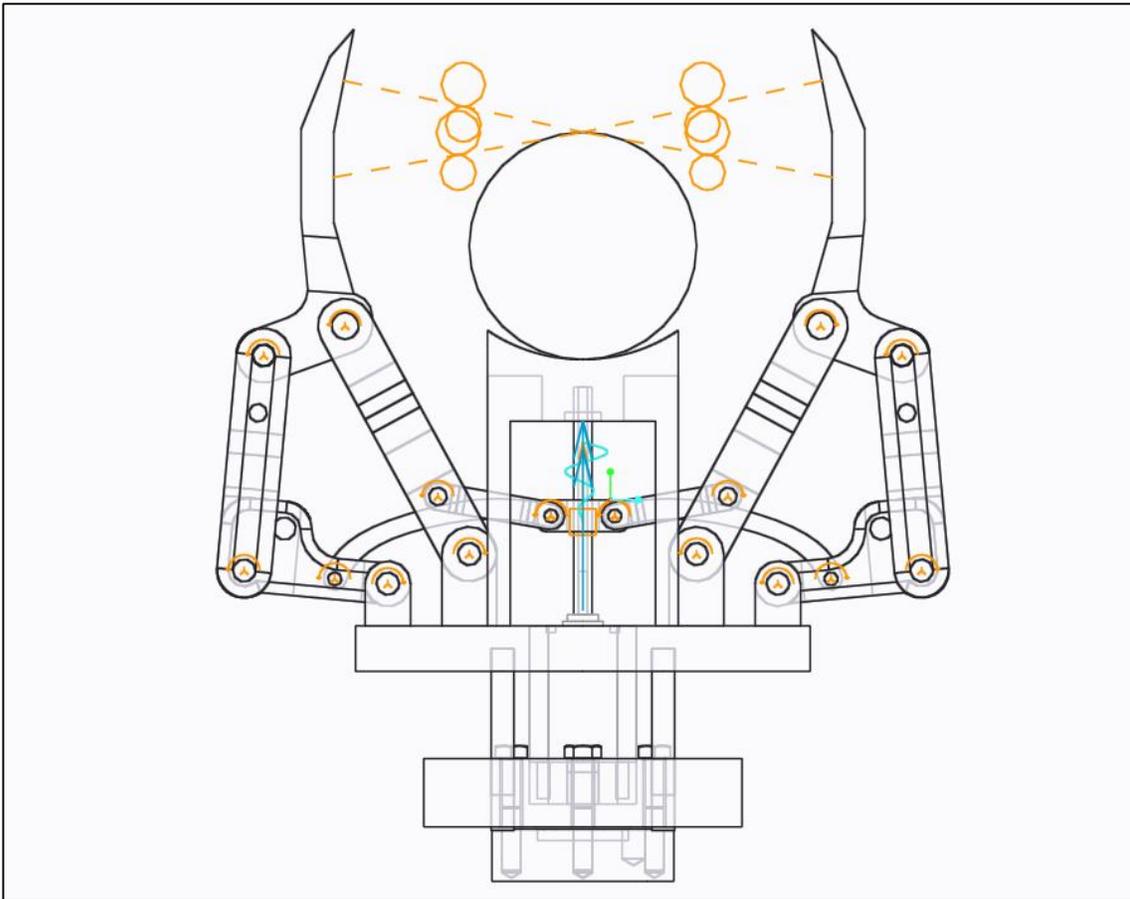
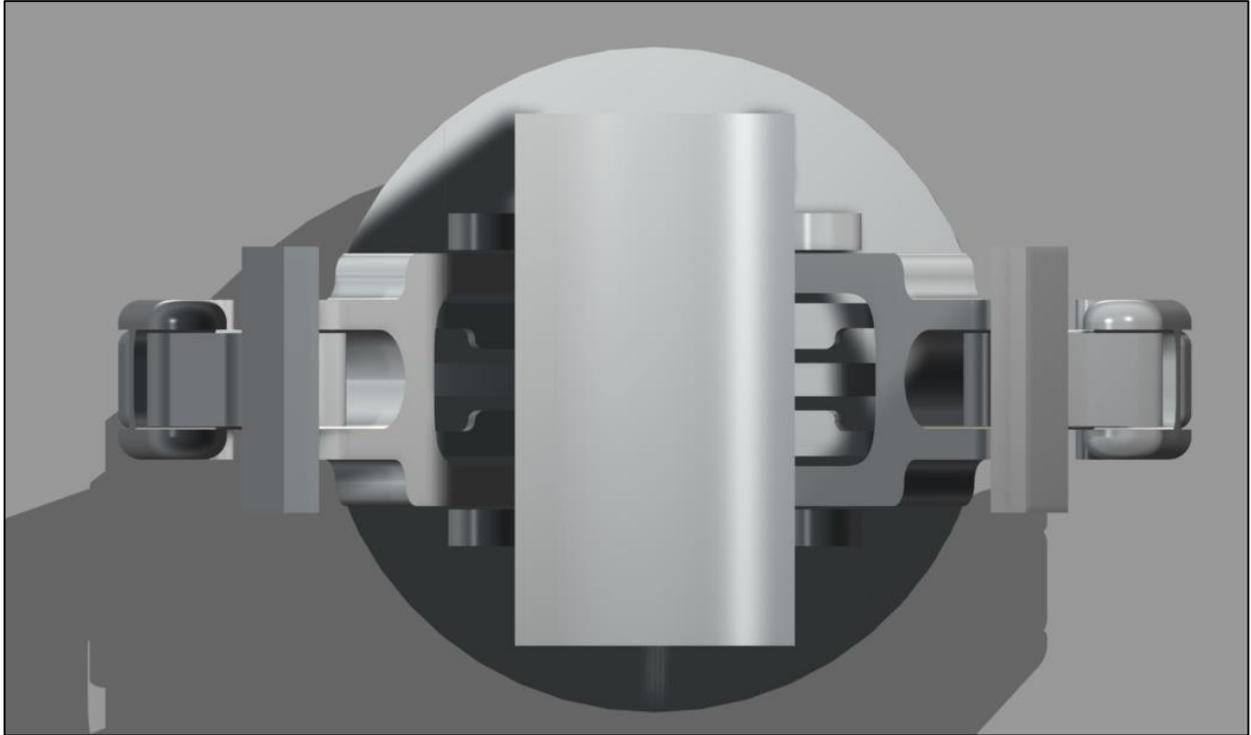
6.1. *Renderizados en CREO PARAMETRIC*

6.2. *Código MATLAB*

6.3. *Documentación*







## Mecanismo.m

```
function Mecanismo
    clear all
    clc
    close all

    global q qini L OE AB BC CE DE CD alpha_1 alpha_2 alpha_3 alpha_4 x0 y0
    xA yA

    L=12.7;
    OE=57.15; AB=31.75; BC=47.52; CE=19.05; DE=38.013; CD=48.36;
    alpha_1=0.11865; alpha_2=1.45643; alpha_3=0.34906; alpha_4=0.96255;

    x0=0; y0=0;
    xA=OE*cos(alpha_4)+CE*cos(alpha_3)-BC*cos(alpha_2)-AB*cos(alpha_1);
    yA=OE*sin(alpha_4)-CE*sin(alpha_3)-BC*sin(alpha_2)+AB*sin(alpha_1);
    xB=xA+AB*cos(alpha_1); yB=yA-AB*sin(alpha_1);
    xC=xB+BC*cos(alpha_2); yC=yB-BC*sin(alpha_2);
    xE=x0+OE*cos(alpha_4); yE=y0+OE*sin(alpha_4);
    xD=xC-CD*cos(47+alpha_3); yD=yC+CD*sin(47+alpha_3);
    phi=alpha_4;
    theta=pi/2;

    qini=[xB;yB;xC;yC;xE;yE;xD;yD;phi;theta];

    q=qini;
    q=ProblemaPosicion(phi,theta);
    scatter([x0 xA q(1) q(3) q(5) q(7)], [y0 yA q(2) q(4) q(6) q(8)]);
    q=qini;
    SimulacionCinematica;

end
```

### ProblemaPosicion.m

```
function qnew=ProblemaPosicion(phi,theta)
    global q qini L OE AB BC CE DE CD alpha_1 alpha_2 alpha_3 alpha_4 x0 y0
    xA yA
    error=1e+10;
    epsilon=1e-7;

    qnew=q;

    mJacob=zeros(12,10);
    vectorEc=zeros(12,1);

    while (error>epsilon)
        qnew(9)=phi;
        qnew(10)=theta;
        mJacob(1:10,:)=CalculaJacobiana(qnew);
        mJacob(11,9)=1;
        mJacob(12,10)=1;

        vectorEc(1:10,1)=CalculaEcRestriccion(qnew);
        vectorEc(11)=0;
        vectorEc(12)=0;

        delta_q=-mJacob\vectorEc;
        qnew=qnew + delta_q;
        error=norm(delta_q);
    end
end
```

### ProblemaVelocidad.m

```
function qp = ProblemaVelocidad(q,phip,thetap)

    mJacob = zeros(12,10);

    mJacob(1:10,:)=CalculaJacobiana(q);
    mJacob(11,9) = 1;
    mJacob(12,10) = 1;

    b=zeros(12,1);

    b(11)=phip;
    b(12)=thetap;
    qp=mJacob\b;
end
```

### ProblemaAceleracion.m

```
function qpp = ProblemaAceleracion(q, qp, phipp,thetapp)

    mJacob = zeros(12,10);
    mJacob(1:10,:) = CalculaJacobiana(q);
    mJacob(11,9) = 1;
    mJacob(12,10) = 1;

    b=zeros(12,1);
    b(1:10) = CalculaFiqupp(q, qp);
    b(11)=-phipp;
    b(12)=-thetapp;

    qpp=-mJacob\b;
end
```

### CalculaEcRestriccion.m

```
function r=CalculaEcRestriccion(q)

    global L OE AB BC CE DE CD alpha_1 alpha_2 alpha_3 alpha_4 x0 y0 xA yA

    L=12.7;
    OE=57.15; AB=31.75; BC=47.52; CE=19.05; DE=38.013; CD=48.36;
    alpha_1=0.11865; alpha_2=1.45643; alpha_3=0.34906; alpha_4=0.96255;

    x0=0; y0=0;
    xA=OE*cos(alpha_4)+CE*cos(alpha_3)-BC*cos(alpha_2)-AB*cos(alpha_1);
    yA=OE*sin(alpha_4)-CE*sin(alpha_3)-BC*sin(alpha_2)+AB*sin(alpha_1);
    phi=alpha_4;
    theta=(pi-alpha_1-alpha_2);

    xB=q(1); yB=q(2);
    xC=q(3); yC=q(4);
    xE=q(5); yE=q(6);
    xD=q(7); yD=q(8);
    phi=q(9); theta=q(10);

    r=zeros(10,1);

    r(1)=(xB-xA)^2+(yB-yA)^2-AB^2;
    r(2)=(xC-xB)^2+(yC-yB)^2-BC^2;
    r(3)=(xE-x0)^2+(yE-y0)^2-OE^2;
    r(4)=(xC-xE)^2+(yC-yE)^2-CE^2;
    r(5)=(xD-xE)^2+(yD-yE)^2-DE^2;
    r(6)=(xC-xD)^2+(yC-yD)^2-CD^2;

    r(7)=(xE-x0)-OE*cos(phi);
    r(8)=(xA-xB)*(xC-xB)+(yA-yB)*(yC-yB)-AB*BC*cos(theta);
    r(9)=(yE-y0)-OE*sin(phi);
    r(10)=(yA-yB)*(xC-xB)-(xA-xB)*(yC-yB)-AB*BC*sin(theta);
end
```

## CalculaJacobian.m

```
function fiq=CalculaJacobian(q)

global L OE AB BC CE DE CD alpha_1 alpha_2 alpha_3 alpha_4 x0 y0 xA yA

L=12.7;
OE=57.15; AB=31.75; BC=47.52; CE=19.05; DE=38.013; CD=48.36;
alpha_1=0.11865; alpha_2=1.45643; alpha_3=0.34906; alpha_4=0.96255;
x0=0; y0=0;
xA=OE*cos(alpha_4)+CE*cos(alpha_3)-BC*cos(alpha_2)-AB*cos(alpha_1);
yA=OE*sin(alpha_4)-CE*sin(alpha_3)-BC*sin(alpha_2)+AB*sin(alpha_1);
phi=alpha_4;
theta=(pi-alpha_1-alpha_2);

xB=q(1); yB=q(2);
xC=q(3); yC=q(4);
xE=q(5); yE=q(6);
xD=q(7); yD=q(8);
phi=q(9); theta=q(10);

fiq=zeros(10,10);

fiq(1,1)=2*(xB-xA);
fiq(1,2)=2*(yB-yA);

fiq(2,1)=-2*(xC-xB);
fiq(2,2)=-2*(yC-yB);
fiq(2,3)=2*(xC-xB);
fiq(2,4)=2*(yC-yB);

fiq(3,5)=2*(xE-x0);
fiq(3,6)=2*(yE-y0);

fiq(4,3)=2*(xC-xE);
fiq(4,4)=2*(yC-yE);
fiq(4,5)=-2*(xC-xE);
fiq(4,6)=-2*(yC-yE);

fiq(5,5)=-2*(xD-xE);
fiq(5,6)=-2*(yD-yE);
fiq(5,7)=2*(xD-xE);
fiq(5,8)=2*(yD-yE);

fiq(6,3)=2*(xC-xD);
fiq(6,4)=2*(yC-yD);
fiq(6,7)=-2*(xC-xD);
fiq(6,8)=-2*(yC-yD);

fiq(7,5)=1;
fiq(7,9)=(OE)*(sin(phi));

fiq(8,1)=-xC-xA+2*xB;
fiq(8,2)=-yC-yA+2*yB;
fiq(8,3)=xA-xB;
fiq(8,4)=yA-yB;
fiq(8,10)=AB*BC*sin(theta);

fiq(9,6)=1;
fiq(9,9)=-OE*cos(phi);

fiq(10,1)=yC-yA;
fiq(10,2)=xA-xC;
fiq(10,3)=yA-yB;
fiq(10,4)=xB-xA;
fiq(10,10)=-AB*BC*cos(theta);

end
```

## CalculaFiqppp.m

```
function fiqppp=CalculaFiqppp(q,qp)

    global L OE AB BC CE DE CD alpha_1 alpha_2 alpha_3 alpha_4 x0 y0 xA yA

    L=12.7;
    OE=57.15; AB=31.75; BC=47.52; CE=19.05; DE=38.013; CD=48.36;
    alpha_1=0.11865; alpha_2=1.45643; alpha_3=0.34906; alpha_4=0.96255;
    x0=0; y0=0;
    xA=OE*cos(alpha_4)+CE*cos(alpha_3)-BC*cos(alpha_2)-AB*cos(alpha_1);
    yA=OE*sin(alpha_4)-CE*sin(alpha_3)-BC*sin(alpha_2)+AB*sin(alpha_1);
    phi=alpha_4;
    theta=(pi/2);

    xBp=qp(1); yBp=qp(2);
    xCp=qp(3); yCp=qp(4);
    xEp=qp(5); yEp=qp(6);
    xDp=qp(7); yDp=qp(8);
    phip=qp(9); thetap=qp(10);

    fiqp=zeros(10,10);

    fiqp(1,1)=2*(xBp);
    fiqp(1,2)=2*(yBp);

    fiqp(2,1)=-2*(xCp-xBp);
    fiqp(2,2)=-2*(yCp-yBp);
    fiqp(2,3)=2*(xCp-xBp);
    fiqp(2,4)=2*(yCp-yBp);

    fiqp(3,5)=2*(xEp);
    fiqp(3,6)=2*(yEp);

    fiqp(4,3)=2*(xCp-xEp);
    fiqp(4,4)=2*(yCp-yEp);
    fiqp(4,5)=-2*(xCp-xEp);
    fiqp(4,6)=-2*(yCp-yEp);

    fiqp(5,5)=-2*(xDp-xEp);
    fiqp(5,6)=-2*(yDp-yEp);
    fiqp(5,7)=2*(xDp-xEp);
    fiqp(5,8)=2*(yDp-yEp);

    fiqp(6,3)=2*(xCp-xDp);
    fiqp(6,4)=2*(yCp-yDp);
    fiqp(6,7)=-2*(xCp-xDp);
    fiqp(6,8)=-2*(yCp-yDp);

    fiqp(7,5)=0;
    fiqp(7,9)=OE*phip*sin(phi);

    fiq(8,1)=-xCp+2*xBp;
    fiq(8,2)=-yCp+2*yBp;
    fiq(8,3)=-xBp;
    fiq(8,4)=-yBp;

    fiq(8,10)=AB*BC*thetap*cos(theta);

    fiqp(9,6)=0;
    fiqp(9,9)=-OE*phip*cos(phi);

    fiq(10,1)=yCp;
    fiq(10,2)=-xCp;
    fiq(10,3)=-yBp;
    fiq(10,4)=xBp;

    fiq(10,10)=AB*BC*thetap*sin(theta);

    fiqppp=fiqp*qp;
end
```

## DibujaMecanismo.m

```
function DibujaMecanismo(q)

    global L OE AB BC CE DE CD alpha_1 alpha_2 alpha_3 alpha_4 x0 y0 xA yA

    L=12.7;
    OE=57.15; AB=31.75; BC=47.52; CE=19.05; DE=38.013; CD=48.36;
    alpha_1=0.11865; alpha_2=1.45643; alpha_3=0.34906; alpha_4=0.96255;

    x0=0; y0=0;
    xA=OE*cos(alpha_4)+CE*cos(alpha_3)-BC*cos(alpha_2)-AB*cos(alpha_1);
    yA=OE*sin(alpha_4)-CE*sin(alpha_3)-BC*sin(alpha_2)+AB*sin(alpha_1);
    phi=alpha_4;
    theta=(pi/2);

    xB=q(1); yB=q(2);
    xC=q(3); yC=q(4);
    xE=q(5); yE=q(6);
    xD=q(7); yD=q(8);
    phi=q(9); theta=q(10);

    hold on;

    %axis([11, 25, -5, 1]);
    %axis([0, 10, -3, 2]);
    axis equal;
    % axis manual;
    % axis off;
    line([xE, x0],[yE, y0], 'lineWidth',3);
    line([xB, xA],[yB, yA], 'lineWidth',3);
    line([xB, xC],[yB, yC], 'lineWidth',3);
    line([xC, xE],[yC, yE], 'lineWidth',5);
    line([xC, xD],[yC, yD], 'lineWidth',5);
    line([xD, xE],[yD, yE], 'lineWidth',5);

    %Triangulo
    fill([xC, xD, xE], [yC, yD, yE], ([122,197,205]/256) );
    h=45;
    %Apoyo
    line([0, -2*L],[h,h], 'lineWidth',3);

    %Mirror
    x0m=-x0-2*L;
    xAm=-xA-2*L;
    xBm=-xB-2*L;
    xCm=-xC-2*L;
    xDm=-xD-2*L;
    xEm=-xE-2*L;
    line([xE, x0m],[yE, y0], 'lineWidth',3);
    line([xB, xAm],[yB, yA], 'lineWidth',3);
    line([xB, xCm],[yB, yC], 'lineWidth',3);
```

```
line([xCm, xEm], [yC, yE], 'linewidth', 5);
line([xCm, xDm], [yC, yD], 'linewidth', 5);
line([xDm, xEm], [yD, yE], 'linewidth', 5);

%Triangulo
fill([xCm, xDm, xEm], [yC, yD, yE], ([122,197,205]/256) );

r=20;
t=linspace(0,2*pi,100);
fill(-L+r*cos(t),h+r+r*sin(t),([255,0,0]/256));

set(gcf, 'color', [255 255 255]/256)
drawnow;
end
```

## 4.6. Problemas cinemáticos

El estudio cinemático de un mecanismo pretende conocer su movimiento independientemente de las fuerzas actuantes. Los problemas cinemáticos son de naturaleza puramente geométrica, y se dirigen exclusivamente al análisis del movimiento en términos de posición, velocidad y aceleración, sin considerar las causas que lo producen.

Para definir de forma más precisa los problemas cinemáticos, es preciso definir previamente el vector  $\mathbf{q}$ , de  $n \times 1$ , que contiene las coordenadas dependiente empleadas en la modelización del mecanismo, y el vector  $\mathbf{z}$ , de dimensión  $g \times 1$ , que contiene las coordenadas de los grados de libertad. Normalmente,  $\mathbf{z}$  es un subconjunto del vector  $\mathbf{q}$ , obtenido al tomar tantas componentes de  $\mathbf{q}$  como grados de libertad.

Empleando estos dos vectores, se está en condiciones de definir los tres problemas cinemáticos principales: de *posición*, de *velocidad* y de *aceleración*. La tabla 4.1 resume los datos e incógnitas de cada problema.

Problema	Datos	Incógnitas
De posición	$\mathbf{z}$	$\mathbf{q}$
De velocidad	$\mathbf{q}, \dot{\mathbf{z}}$	$\dot{\mathbf{q}}$
De aceleración	$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{z}}$	$\ddot{\mathbf{q}}$

Tabla 4.1: Clasificación de los problemas cinemáticos.

En el *problema de posición* se trata de hallar la posición de todas las coordenadas naturales a partir de las posiciones de los grados de libertad. En el *problema de velocidad* se trata de hallar la velocidad de todas las coordenadas naturales, conocida la velocidad de los grados de libertad. Y en el *problema de aceleración* se trata de hallar la aceleración de todas las coordenadas naturales conocida la aceleración de los grados de libertad. Para resolver el problema de velocidad es necesario resolver previamente el de posición. Para resolver el problema de aceleración es preciso resolver previamente los problemas de posición y velocidad.

### 4.6.1. Problema de posición

El problema de posición, también conocido como problema *de montaje*, consiste en calcular la posición  $\mathbf{q}$  de todos los elementos del mecanismo, a partir de la posición de los grados de libertad  $\mathbf{z}$ . La solución de este problema es relativamente sencilla en el caso de mecanismos de cadena abierta, debido a que la solución es única y puede calcularse de forma

explícita. En cambio, en los mecanismos con cadena cinemática cerrada el problema es bastante más complejo, normalmente con soluciones múltiples y frecuentemente sin solución explícita.

Como ya se ha dicho, las  $n$  incógnitas del vector  $\mathbf{q}$  se encuentran ligadas por  $m$  ecuaciones de restricción, que agruparemos en el denominado vector de restricciones  $\Phi$ , de dimensión  $m \times 1$ . Puesto que las ecuaciones de restricción son, en principio, funciones de las coordenadas naturales y del tiempo, podemos escribir de forma compacta el conjunto de todas las ecuaciones de restricción como:

$$\Phi(\mathbf{q}, t) = \begin{Bmatrix} \Phi_1(\mathbf{q}, t) \\ \dots \\ \Phi_m(\mathbf{q}, t) \end{Bmatrix} \quad (4.48)$$

En definitiva, nuestro problema consiste en resolver dicho conjunto de ecuaciones no lineales. Para abordar su resolución se puede recurrir al *método de Newton-Raphson*, también conocido como *método de la tangente*, basado en linealizar la ecuación anterior tomando los dos primeros términos de su serie de Taylor,

$$\Phi(\mathbf{q} + \Delta\mathbf{q}, t) \approx \Phi(\mathbf{q}, t) + \Phi_{\mathbf{q}}(\mathbf{q}, t)\Delta\mathbf{q} \approx \mathbf{0} \quad (4.49)$$

donde  $\Phi_{\mathbf{q}}$  es la matriz jacobiana  $m \times n$  de las ecuaciones de restricción, cuya expresión explícita es:

$$\Phi_{\mathbf{q}} = \begin{bmatrix} \frac{\partial\phi_1}{\partial q_1} & \frac{\partial\phi_1}{\partial q_2} & \dots & \frac{\partial\phi_1}{\partial q_n} \\ \frac{\partial\phi_2}{\partial q_1} & \frac{\partial\phi_2}{\partial q_2} & \dots & \frac{\partial\phi_2}{\partial q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial\phi_m}{\partial q_1} & \frac{\partial\phi_m}{\partial q_2} & \dots & \frac{\partial\phi_m}{\partial q_n} \end{bmatrix} \quad (4.50)$$

La ecuación 4.49 se puede tomar como referencia para establecer el siguiente proceso iterativo:

$$\Phi_{\mathbf{q}}(\mathbf{q}_i, t)(\mathbf{q}_{i+1} - \mathbf{q}_i) = -\Phi(\mathbf{q}_i, t) \quad (4.51)$$

En cada iteración es necesario resolver la ecuación 4.51, que representa un sistema lineal de  $m$  ecuaciones con otras tantas incógnitas. Aunque el vector  $\mathbf{q}$  tiene  $n$  componentes, hay que recordar que  $g$  de estas componentes son conocidas de antemano, puesto que representan la posición de los grados de libertad. Para comenzar el proceso iterativo necesitamos partir de una estimación de la solución  $\mathbf{q}_0$ , suficientemente próxima a la solución real

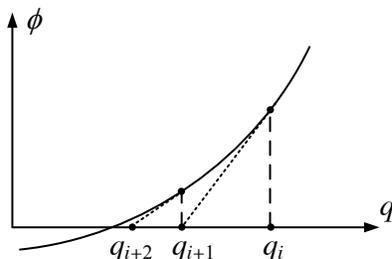


Figura 4.26

para que las sucesivas iteraciones converjan. Aunque no siempre es trivial, en la práctica la obtención de dicha estimación no suele representar una gran dificultad. El proceso iterativo termina cuando éste converge a una solución próxima a la real, lo que sucede cuando la diferencia entre dos iteraciones sucesivas es menor que una tolerancia prefijada,

$$|\mathbf{q}_{i+1} - \mathbf{q}_i| < \epsilon \quad (4.52)$$

Se puede demostrar que el método de Newton Raphson goza de convergencia cuadrática en el entorno de la solución, lo que hace que habitualmente sean precisas muy pocas iteraciones hasta obtener la solución.

En el caso de una sola variable, la interpretación geométrica del método de Newton-Raphson se muestra en la figura 4.26. Como se puede ver, al linealizar la función no lineal estamos acercándonos a la solución por medio de las tangentes en las sucesivas soluciones parciales, cada vez más próximas a la solución verdadera. Lo que en el caso de una función de una variable es la pendiente de cada recta tangente, en el caso general de una función vectorial de varias variables es la matriz jacobiana. Entender bien esta interpretación permitirá descubrir posibles errores en la matriz jacobiana o en el vector de ecuaciones de restricción, a la vista del comportamiento del proceso iterativo.

Se denomina *simulación cinemática* a la resolución sucesiva del problema de posición para una determinada ley de movimiento de los grados de libertad  $\mathbf{z}(t)$ . Mediante ella, se pueden determinar trayectorias, curvaturas y se pueden generar animaciones en el computador. Puesto que la simulación cinemática requiere la resolución del problema de posición en múltiples etapas, la estimación inicial  $\mathbf{q}_0$  requerida al comienzo de cada etapa se puede tomar como la solución final de la etapa anterior.

### 4.6.2. Problema de velocidad

El problema de velocidad consiste en determinar las velocidades  $\dot{\mathbf{q}}$  de todas las variables del mecanismo, conocida su posición  $\mathbf{q}$  y la velocidad  $\dot{\mathbf{z}}$  de los grados de libertad. Derivando la ecuación 4.48 mediante la regla de la cadena, se obtiene:

$$\frac{d}{dt}\Phi(\mathbf{q}, t) = \Phi_{\mathbf{q}}\dot{\mathbf{q}} + \Phi_t = \mathbf{0} \quad (4.53)$$

donde  $\Phi_t$  es la derivada parcial de las ecuaciones de restricción respecto al tiempo. En los casos vistos hasta el momento, esta derivada es nula pues las ecuaciones de restricción de sólido rígido, de par cinemático y de coordenada relativa son independientes del tiempo. Sin embargo, en otros casos este término puede ser no nulo. Despejando en la ecuación anterior:

$$\Phi_{\mathbf{q}}\dot{\mathbf{q}} = -\Phi_t \quad (4.54)$$

Aparentemente, la ecuación 4.54 representa un sistema lineal de  $m$  ecuaciones y  $n$  incógnitas, pero hay que recordar que las velocidades de los grados de libertad son conocidas. El valor de dichas velocidades conocidas se puede incluir añadiendo  $g$  filas triviales al sistema de ecuaciones, cada una de las cuales especifique el valor de la velocidad de un grado de libertad. Este punto se aclarará en un ejemplo posterior.

### 4.6.3. Problema de aceleración

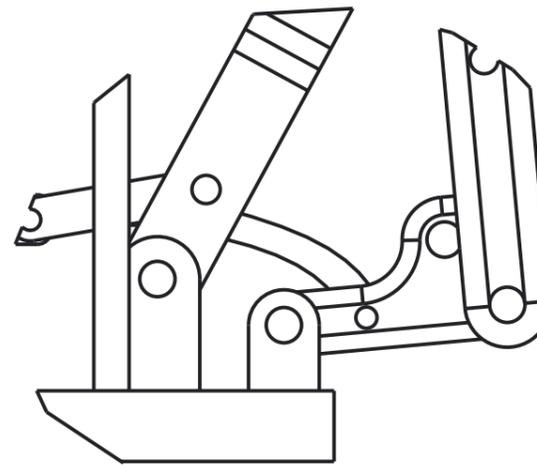
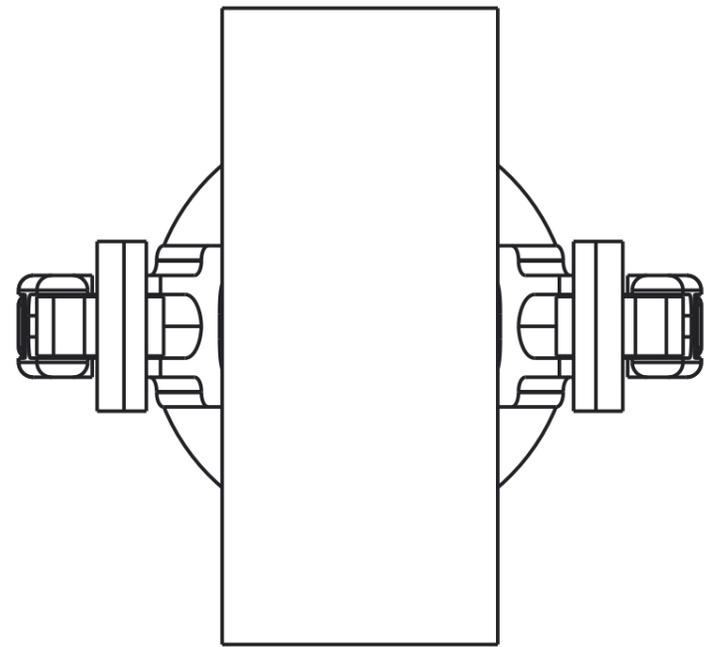
En el problema de aceleración se tratan de determinar las aceleraciones  $\ddot{\mathbf{q}}$  de todas las variables del mecanismo, conocida la posición  $\mathbf{q}$ , la velocidad  $\dot{\mathbf{q}}$ , y las aceleraciones de los grados de libertad  $\ddot{\mathbf{z}}$ . Derivando respecto al tiempo la ecuación de velocidades 4.53, se obtiene

$$\frac{d^2}{dt^2}\Phi(\mathbf{q}, t) = \Phi_{\mathbf{q}}\ddot{\mathbf{q}} + \dot{\Phi}_{\mathbf{q}}\dot{\mathbf{q}} + \ddot{\Phi}_t = \mathbf{0} \quad (4.55)$$

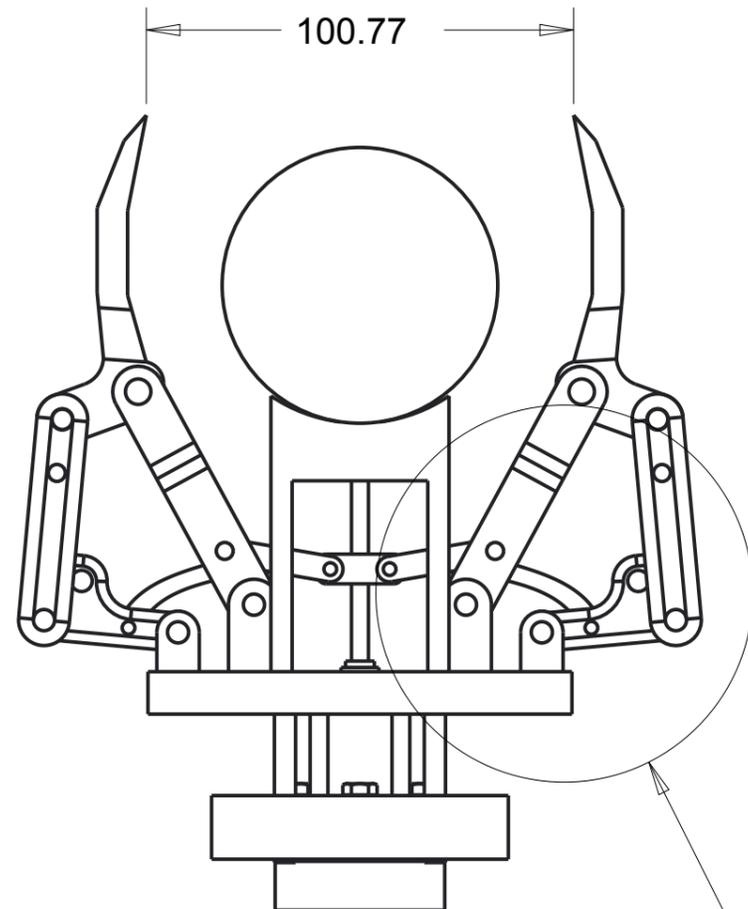
Reordenando los términos, se puede escribir

$$\Phi_{\mathbf{q}}\ddot{\mathbf{q}} = -\dot{\Phi}_{\mathbf{q}}\dot{\mathbf{q}} - \ddot{\Phi}_t \quad (4.56)$$

El miembro de la derecha es conocido en función de la posición, la velocidad y el tiempo. Como en el problema de velocidad, se ha obtenido un sistema lineal de  $m$  ecuaciones con  $n$  incógnitas, en el que se conocen las aceleraciones de los grados de libertad. Dichas aceleraciones conocidas se pueden introducir mediante  $g$  ecuaciones adicionales, cada una de las cuales especifique el valor de la aceleración de un grado de libertad.

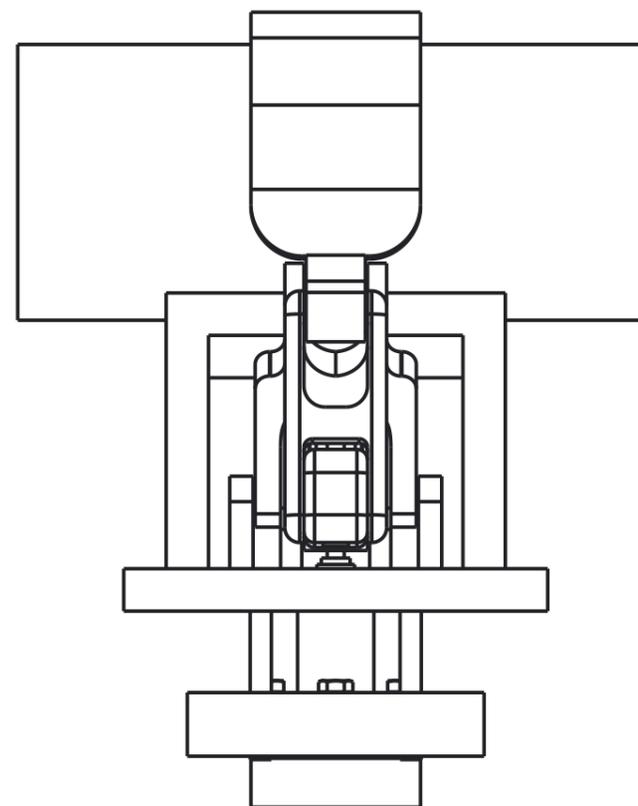


DETAIL A  
SCALE 1.000

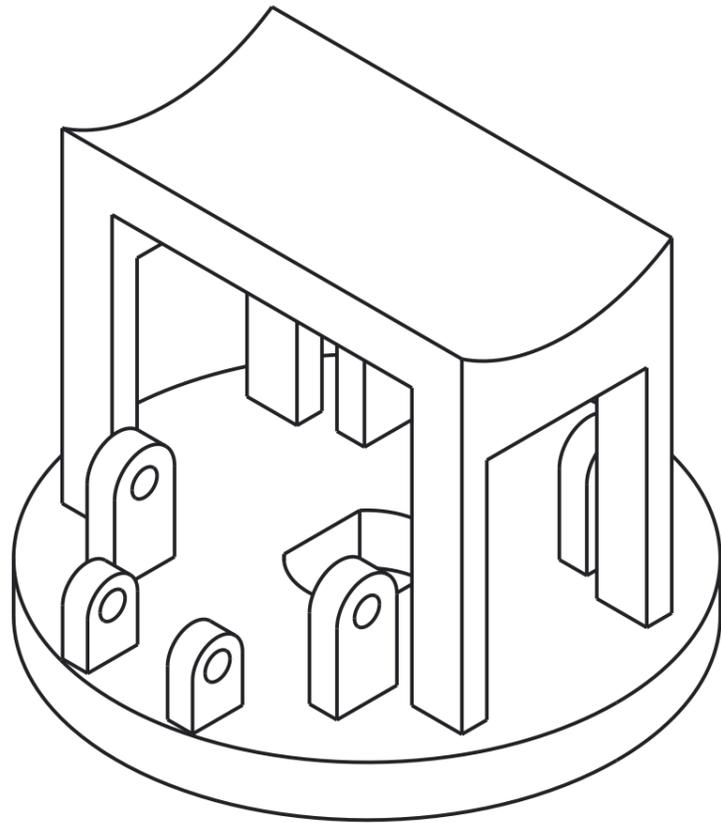


SCALE 0.600

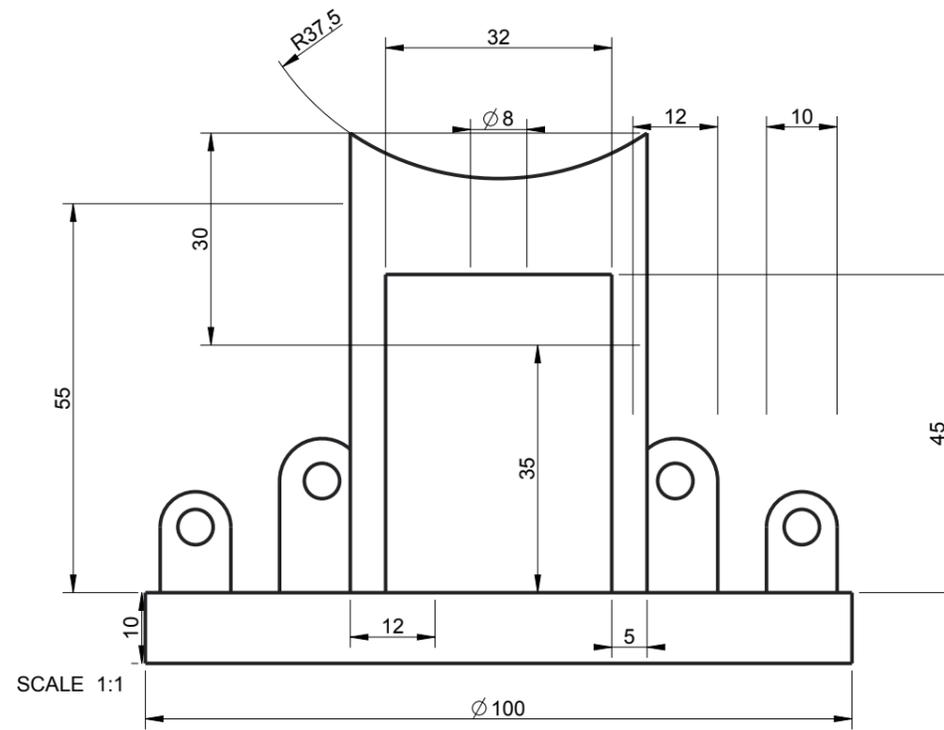
SEE DETAIL A



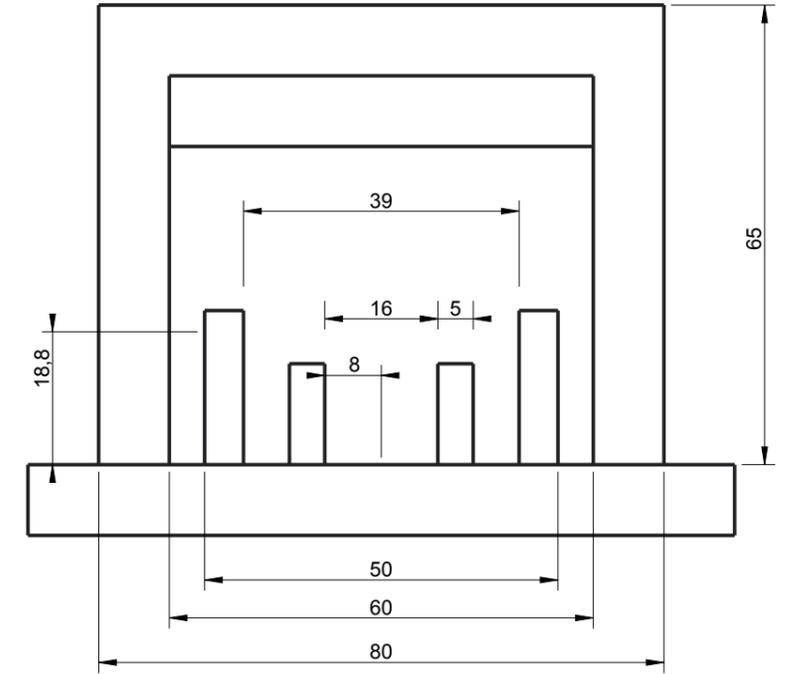
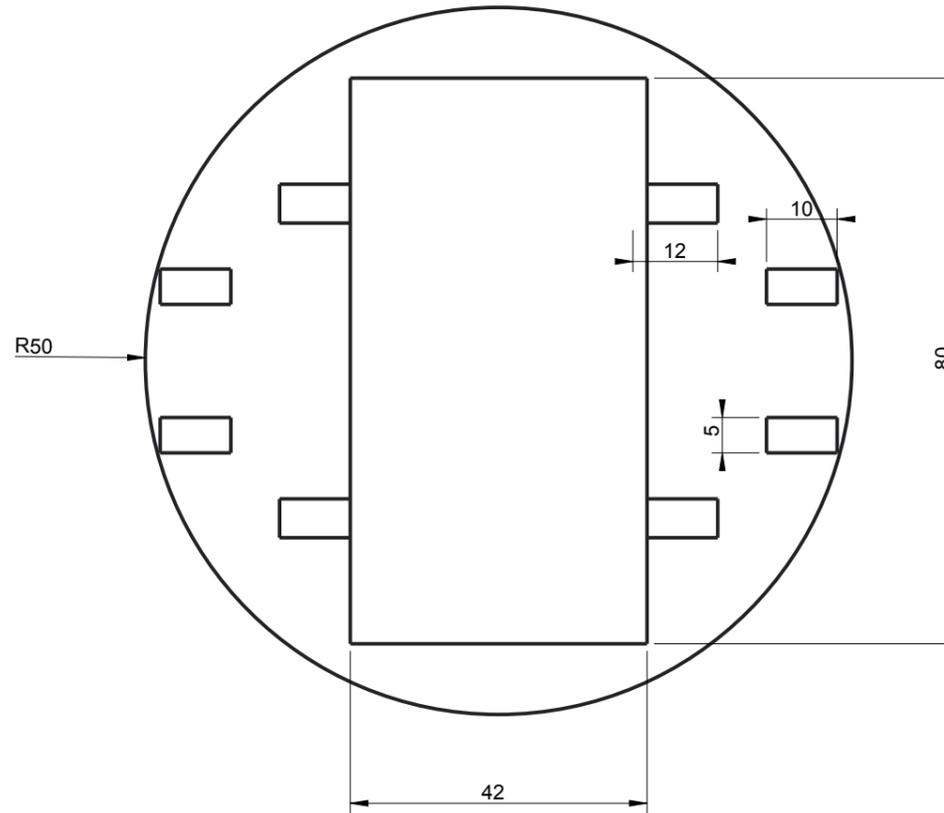
1		3	DES_STEEL	
2		2	DES_ALUMINUM7075	
2		1	DES_ABS	
Nº Piezas	Descripción	Marca	Material	Fichero
	mm.	ISO2768-m	Nov-14-15	Alumos: Grupo5
0.333	UNE 1037	ISO2768-K UNE A3	Fecha plantilla Oct-02 2014	
Denominación:				
<b>AGARRE</b>				
ESCUELA SUPERIOR DE INGENIEROS			Nº Plano:	
<b>TECNUN</b> SAN SEBASTIAN			Sustituye a:	
			Sustituido por:	



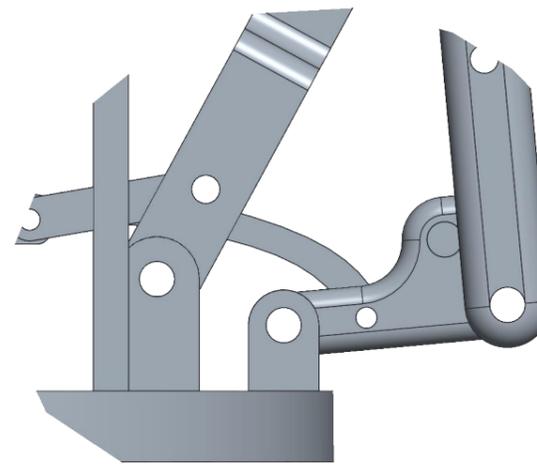
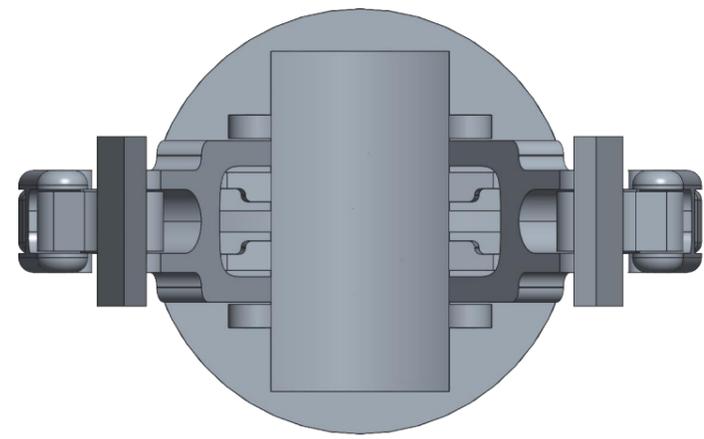
SCALE 1:1



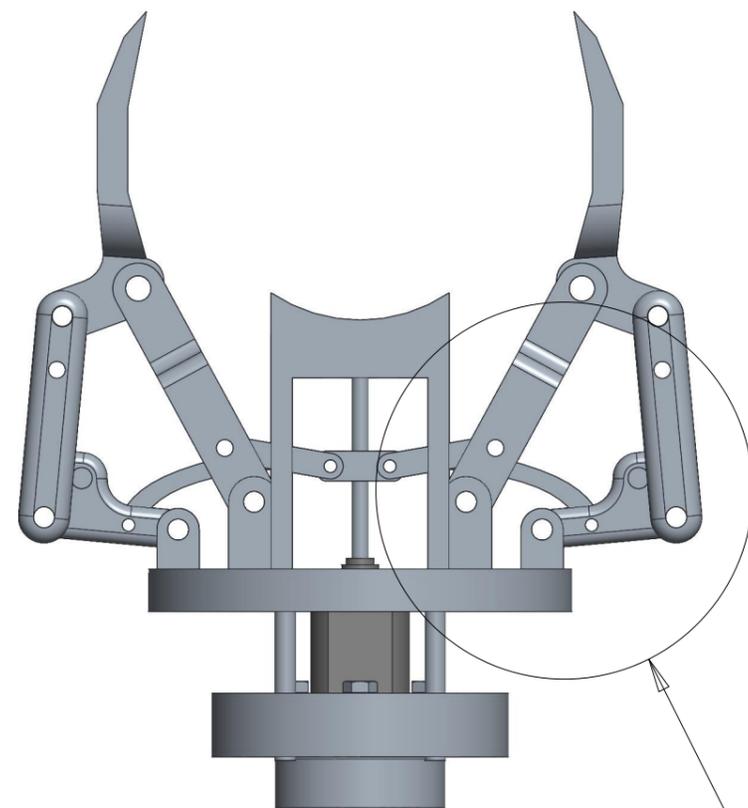
SCALE 1:1



Nº Piezas	Descripción			Marca	Material	Fichero
Metodo Proyección	Unidad dimens.	Toler. geom.	Fecha	Alumnos:		
	mm.	ISO2768-m	Nov-14-15	Grupo5		
Escala	Estado superficial	Toler. general	Fecha plantilla			
1:2	UNE 1037	ISO2768-K	Oct-02 2014			
Denominación:						
PRT0008						
ESCUELA SUPERIOR DE INGENIEROS				Nº Plano:		
<b>TECNUN</b> SAN SEBASTIAN				Sustituye a:		
				Sustituido por:		

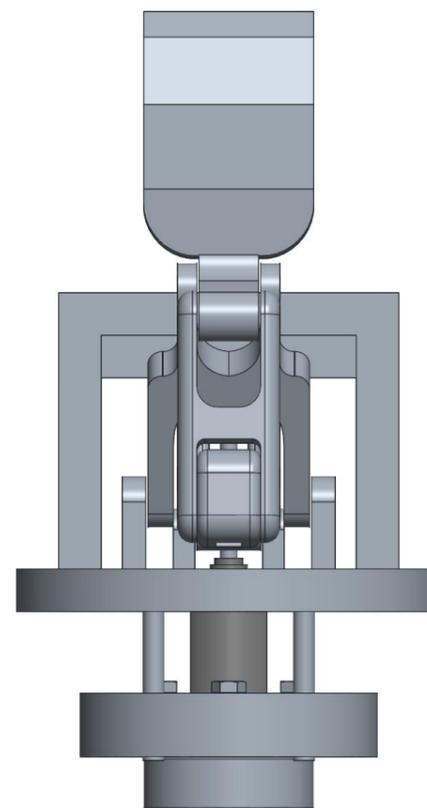


DETAIL A  
SCALE 1.000

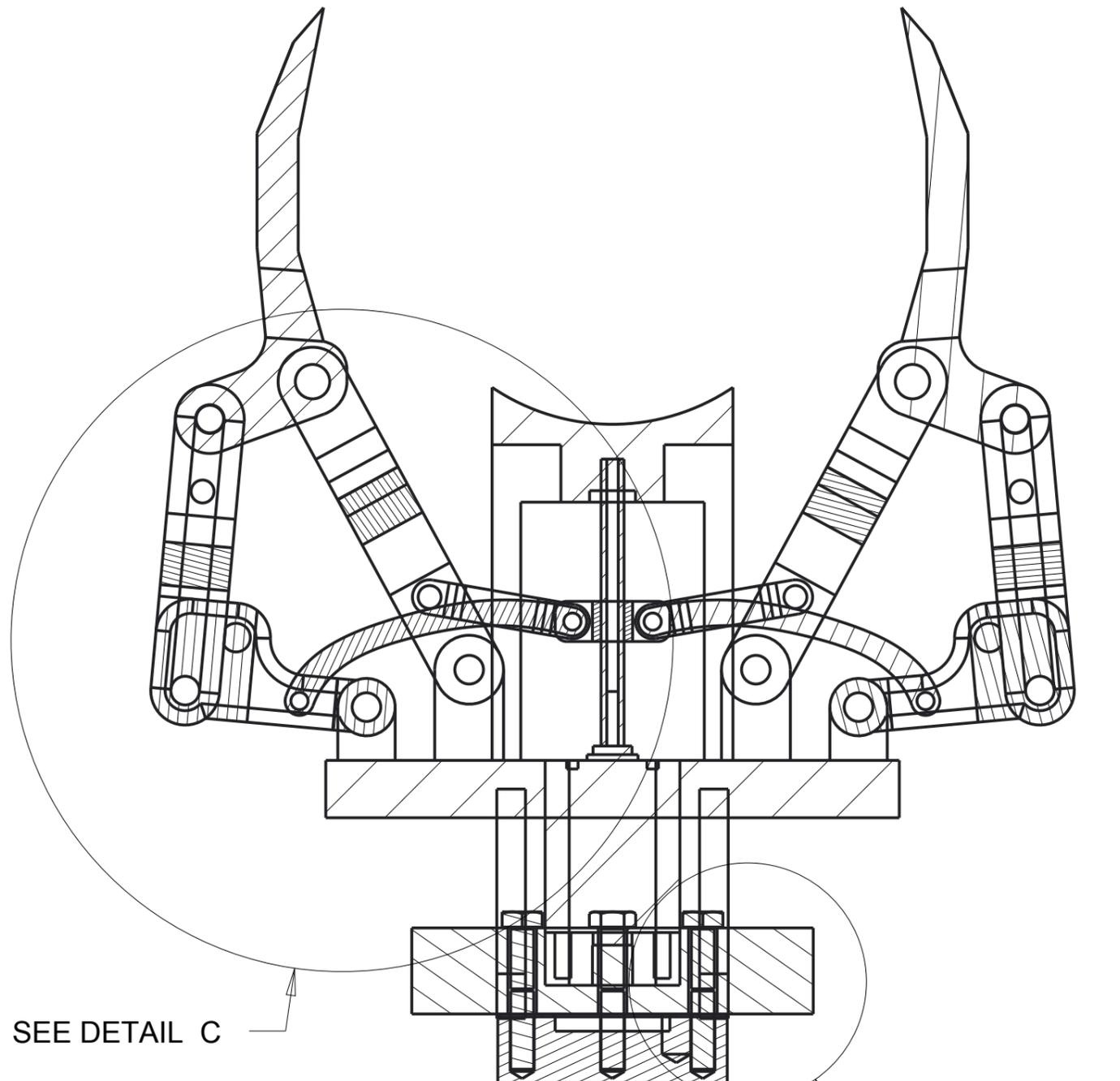


SCALE 0.600

SEE DETAIL A



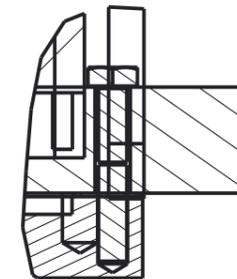
1		3	DES_STEEL	
2		2	DES_ALUMINUM7075	
2		1	DES_ABS	
Nº Piezas	Descripción	Marca	Material	Fichero
Metodo Proyección	Unidad dimens.	Toler. geom.	Fecha	Alumnos:
	mm.	ISO2768-m	Nov-14-15	Grupo5
Escala	Estado superficial	Toler. general	Fecha plantilla	
0.333	UNE 1037	ISO2768-K	Oct-02 2014	
Denominación:				
<b>AGARRE</b>				
ESCUELA SUPERIOR DE INGENIEROS			Nº Plano:	
<b>TECNUN</b> SAN SEBASTIAN			Sustituye a:	
			Sustituido por:	



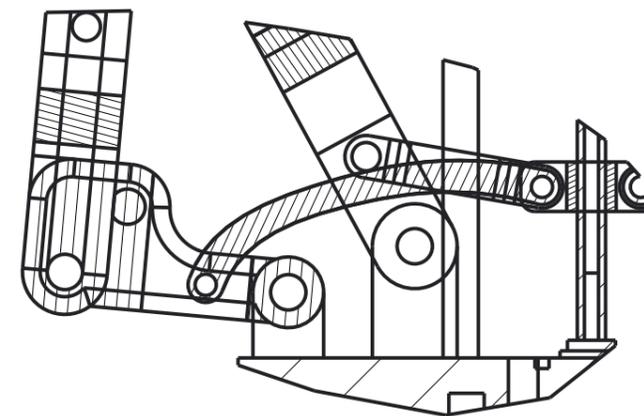
SEE DETAIL C

SECTION XSEC0001-XSEC0001  
SCALE 1.000

SEE DETAIL B

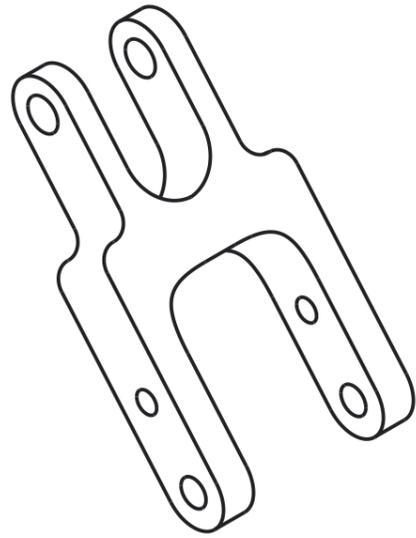


DETAIL B  
SCALE 1.000

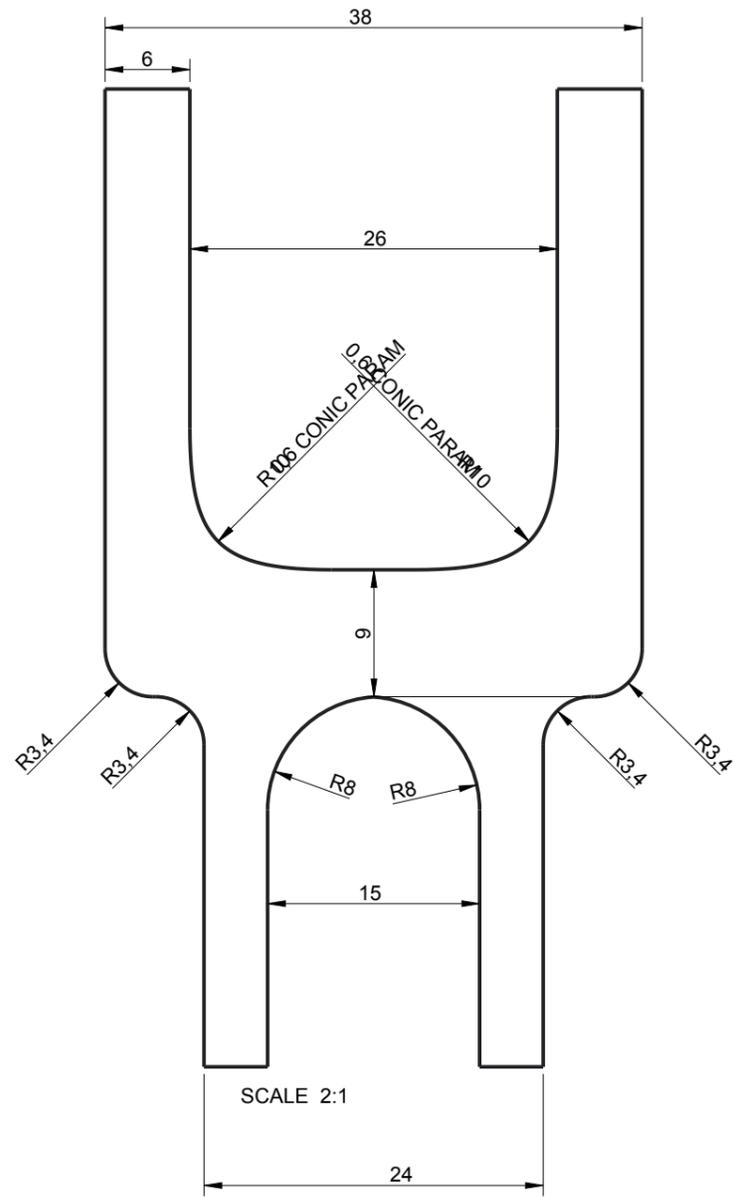


DETAIL C  
SCALE 1.000

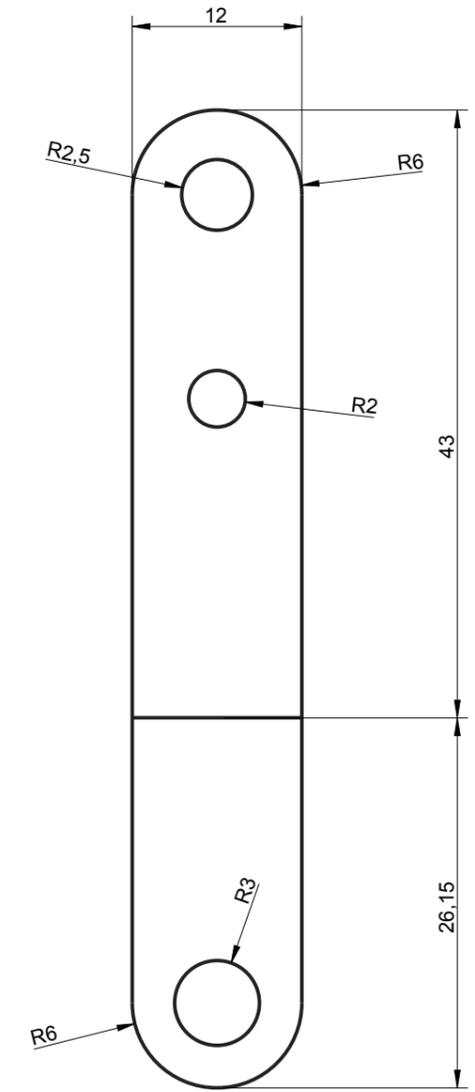
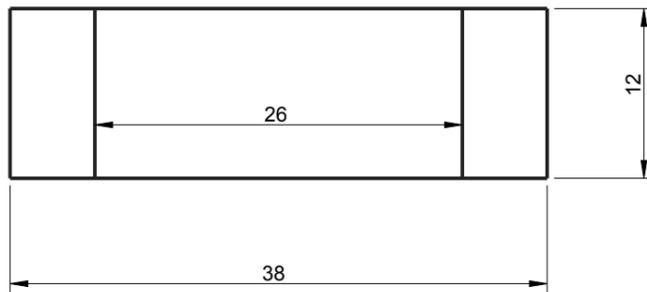
1		3	DES_STEEL	
2		2	DES_ALUMINUM7075	
2		1	DES_ABS	
Nº Piezas	Descripción	Marca	Material	Fichero
Metodo Proyección	Unidad dimens.	Toler. geom.	Fecha	Alumnos:
	mm.	ISO2768-m	Nov-14-15	Grupo5
Escala	Estado superficial	Toler. general	Fecha plantilla	
0.333	UNE 1037	ISO2768-K UNE A3	Oct-02 2014	
Denominación:				
<b>AGARRE</b>				
ESCUELA SUPERIOR DE INGENIEROS			Nº Plano:	
<b>TECNUN</b> SAN SEBASTIAN			Sustituye a:	
			Sustituido por:	



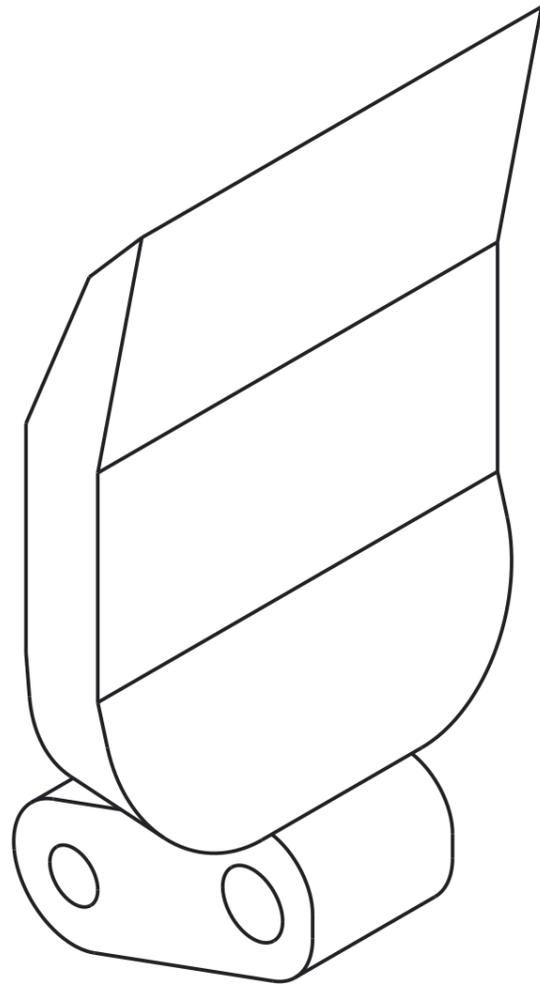
SCALE 1:1



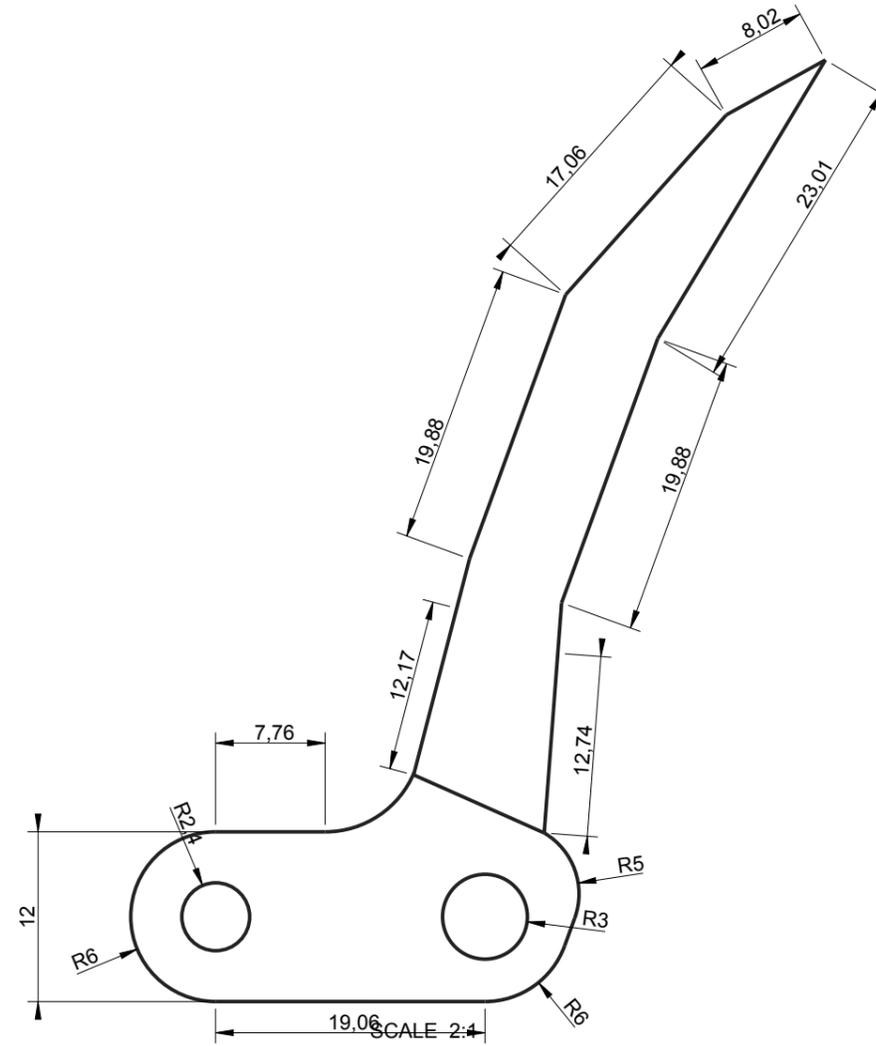
SCALE 2:1



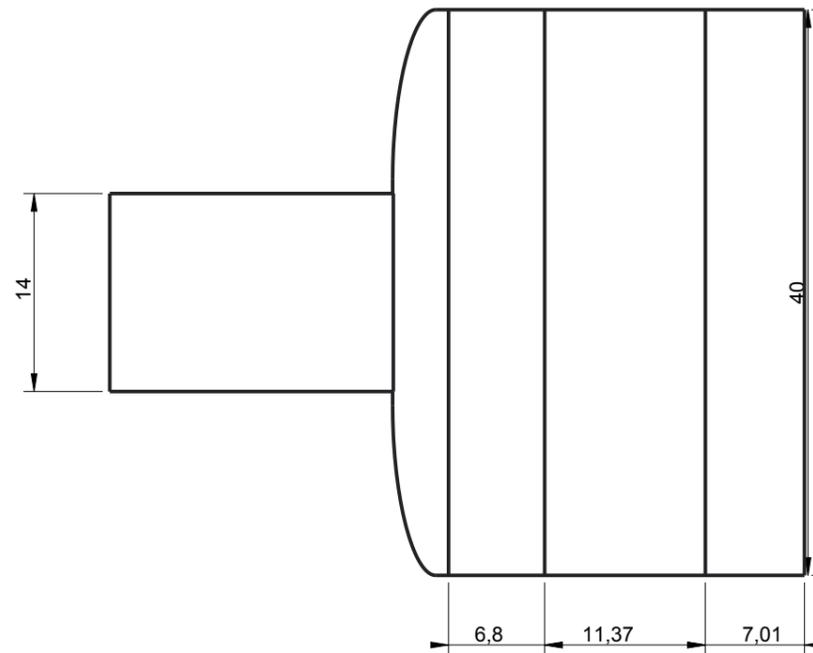
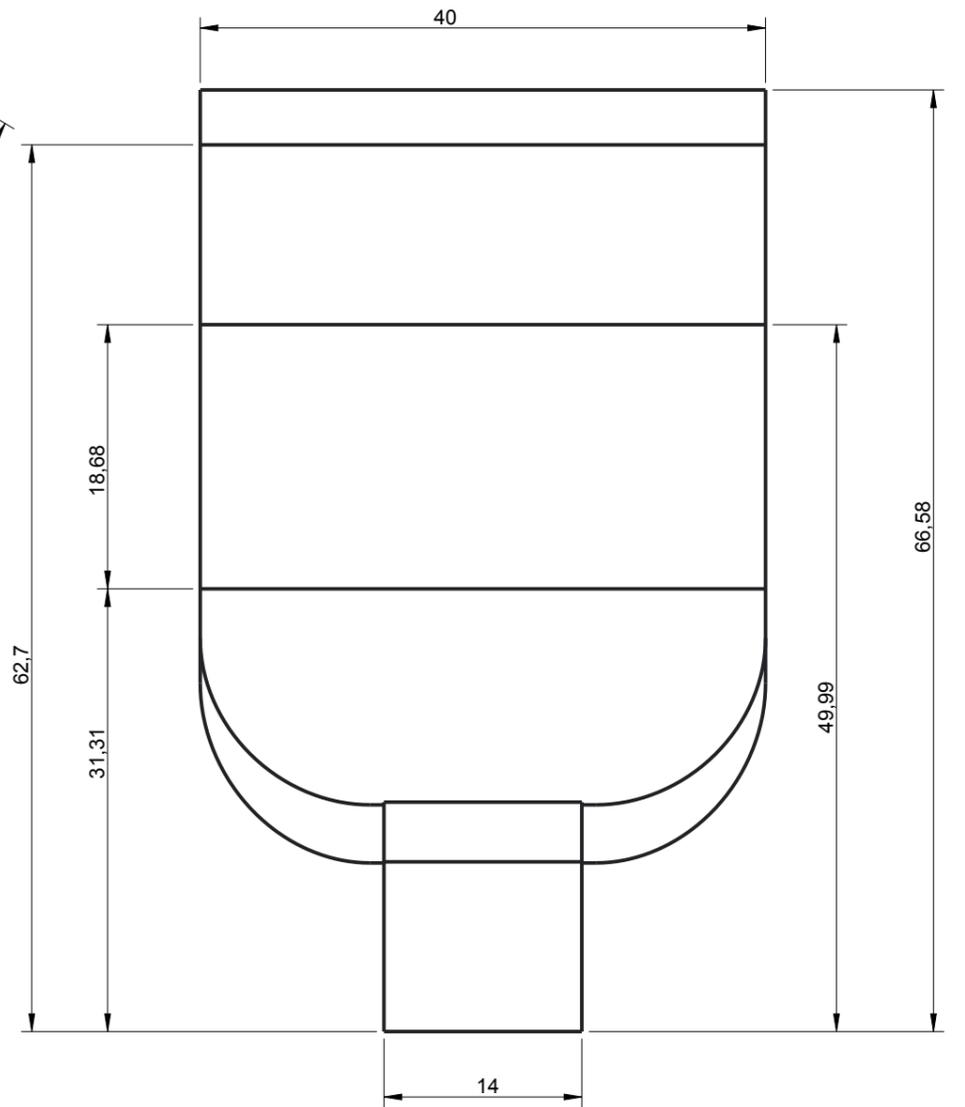
Nº Piezas	Descripción			Marca	Material	Fichero
	Unidad dimens. mm.	Toler. geom. ISO2768-m	Fecha Nov-15-15	Alumnos: Grupo5		
Escala 1:1	Estado superficial UNE 1037	Toler. general ISO2768-K UNE A3	Fecha plantilla Oct-02 2014			
Denominación:  PRT0001						
ESCUELA SUPERIOR DE INGENIEROS				Nº Plano:		
TECNUN SAN SEBASTIAN				Sustituye a:		
				Sustituido por:		



SCALE 2:1

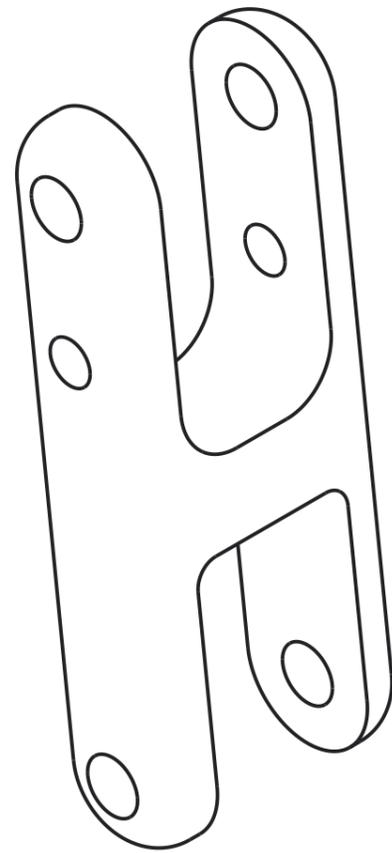


SCALE 2:1

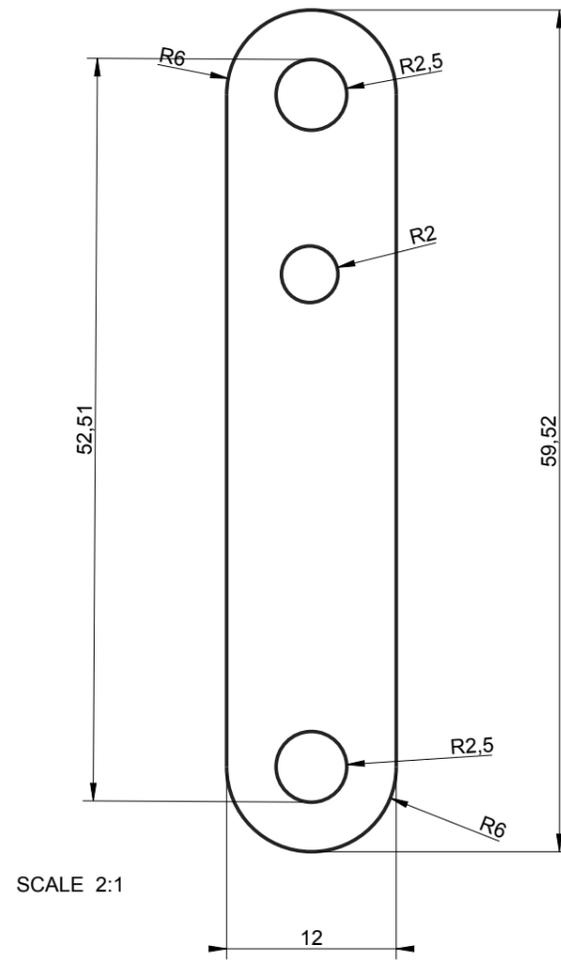


Nº Piezas	Descripción			Marca	Material	Fichero
	Unidad dimens. mm.	Toler. geom. ISO2768-m	Fecha Nov-15-15	Alumnos: Grupo5		
	Estado superficial UNE 1037	Toler. general ISO2768-K	Fecha plantilla Oct-02 2014			
Escala 1:1		UNE A3		Denominación:  PRT0002		
ESCUELA SUPERIOR DE INGENIEROS  <b>TECNUN</b> SAN SEBASTIAN					Nº Plano: Sustituye a: Sustituido por:	

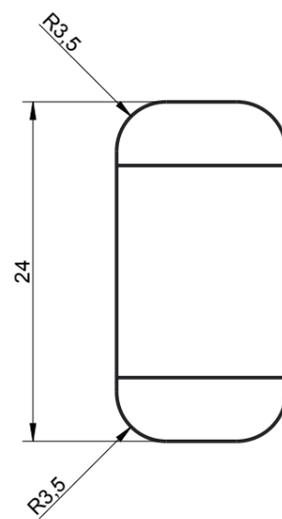
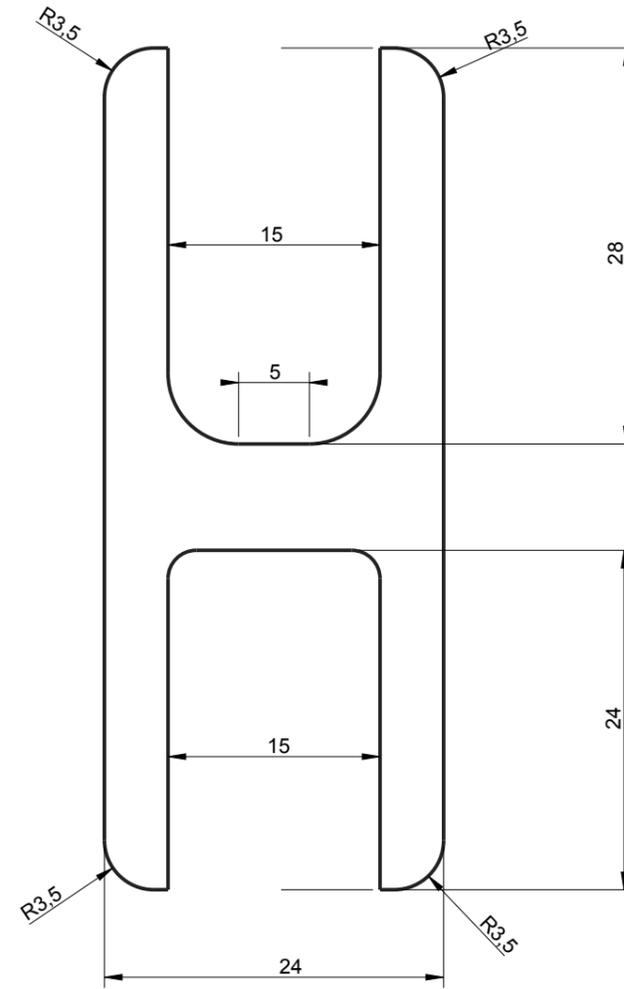




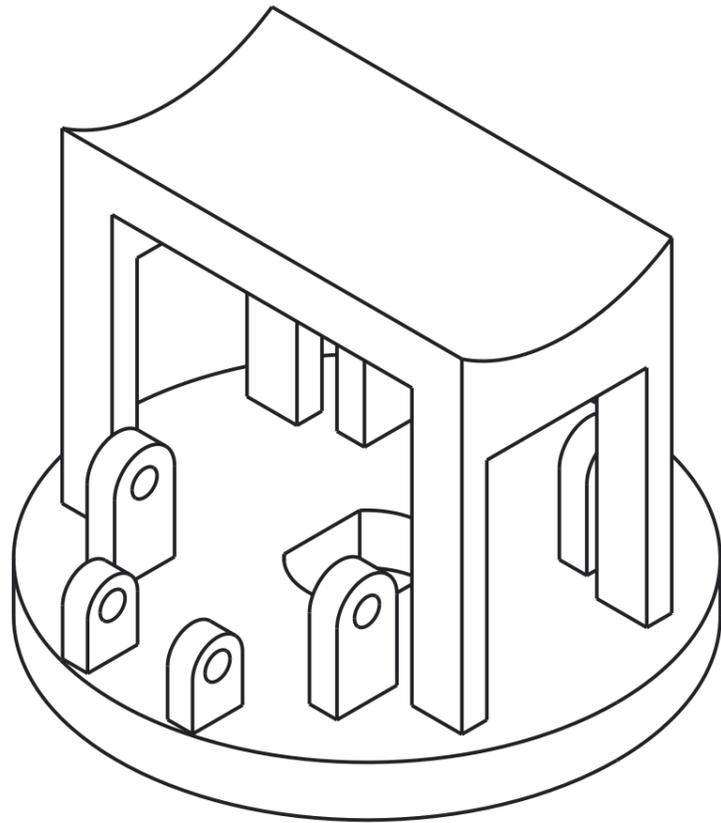
SCALE 2:1



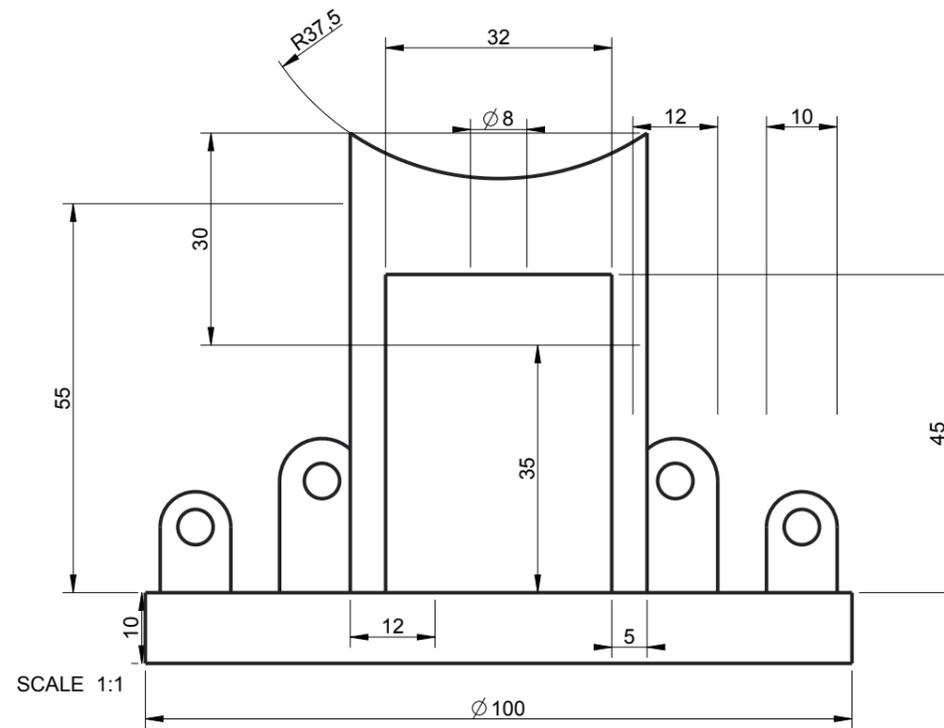
SCALE 2:1



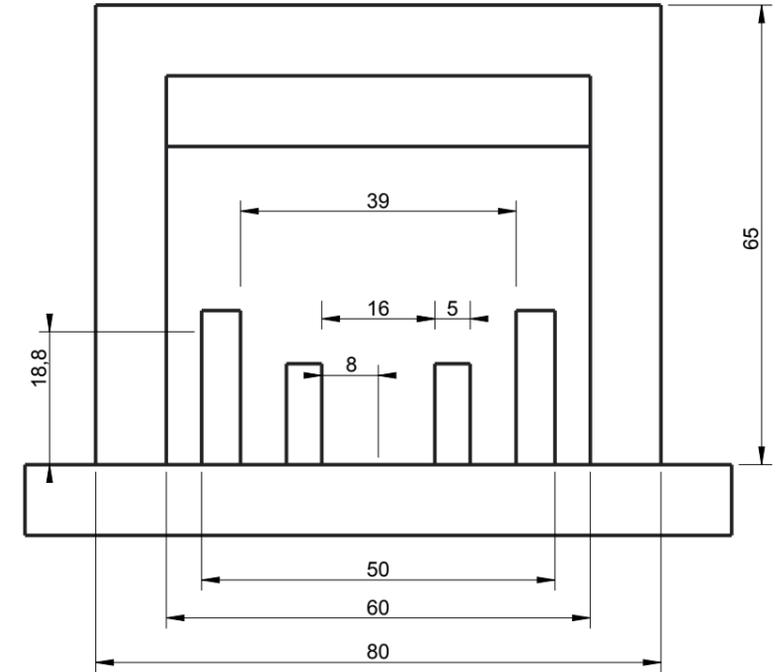
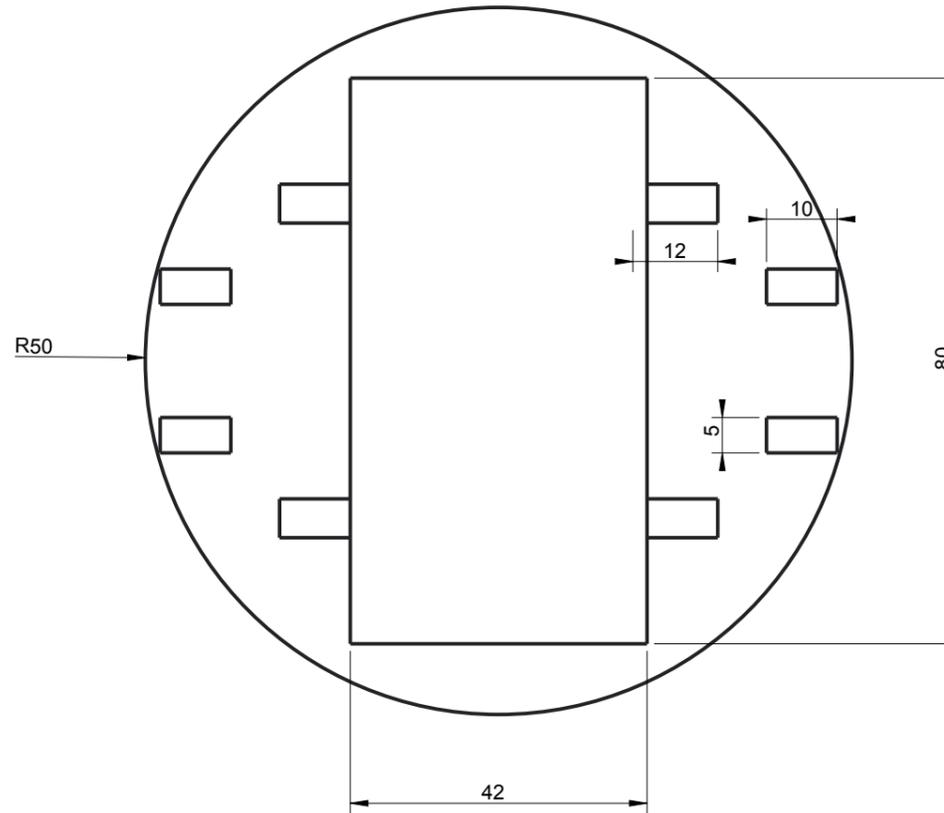
Nº Piezas	Descripción			Marca	Material	Fichero
	Metodo Proyección mm.	Toler. geom. ISO2768-m Toler. general	Fecha Nov-14-15	Alumnos: Grupo5		
Escala 1:1	Estado superficial UNE 1037	ISO2768-K UNE A3	Fecha plantilla Oct-02 2014			
Denominación: <p style="text-align: center;"><b>PRT0004</b></p>						
ESCUELA SUPERIOR DE INGENIEROS <p style="text-align: center;"><b>TECNUN</b></p>					SAN SEBASTIAN	
					Nº Plano: Sustituye a: Sustituido por:	



SCALE 1:1

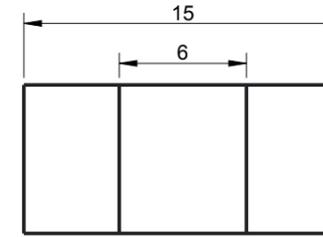
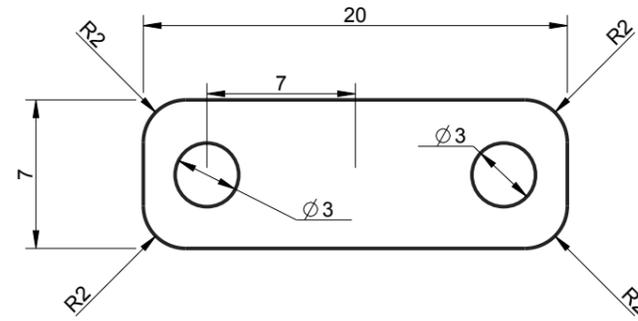
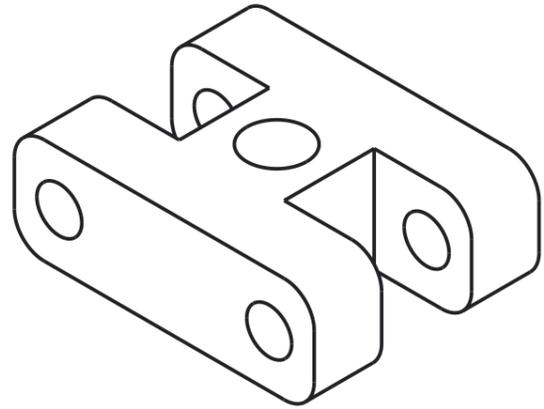


SCALE 1:1

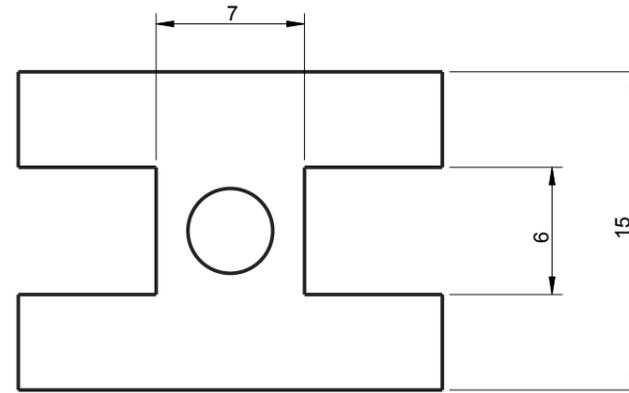


Nº Piezas	Descripción			Marca	Material	Fichero
Metodo Proyección	Unidad dimens.	Toler. geom.	Fecha	Alumnos:		
	mm.	ISO2768-m	Nov-14-15	Grupo5		
Escala	Estado superficial	Toler. general	Fecha plantilla			
1:2	UNE 1037	ISO2768-K	Oct-02 2014			
Denominación:						
PRT0008						
ESCUELA SUPERIOR DE INGENIEROS				Nº Plano:		
<b>TECNUN</b> SAN SEBASTIAN				Sustituye a:		
				Sustituido por:		

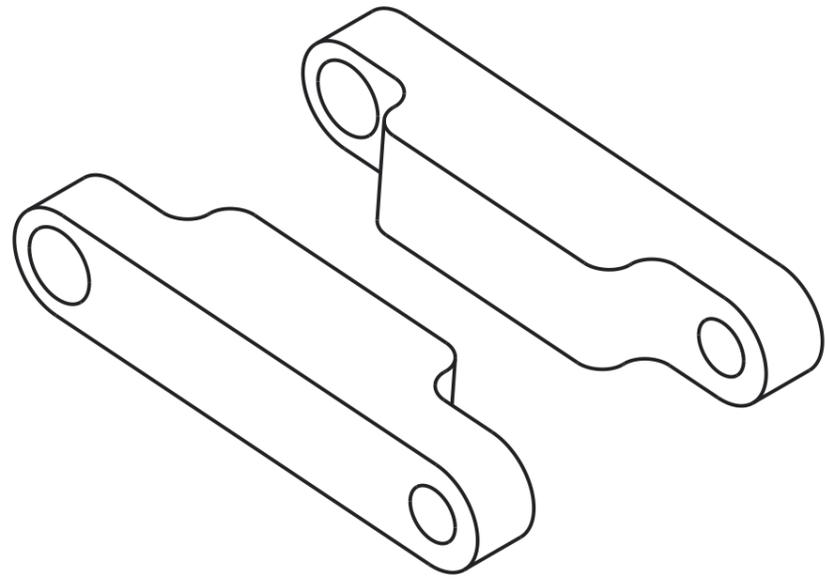




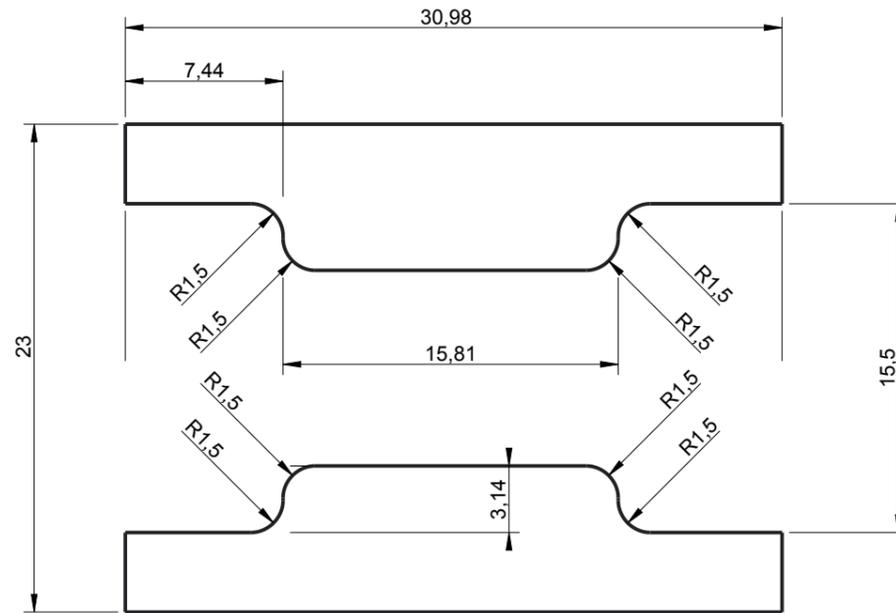
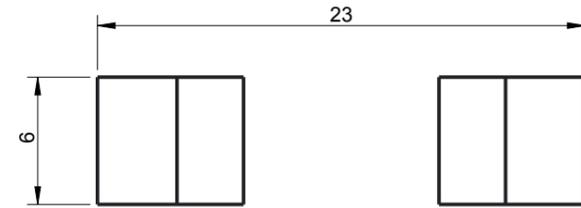
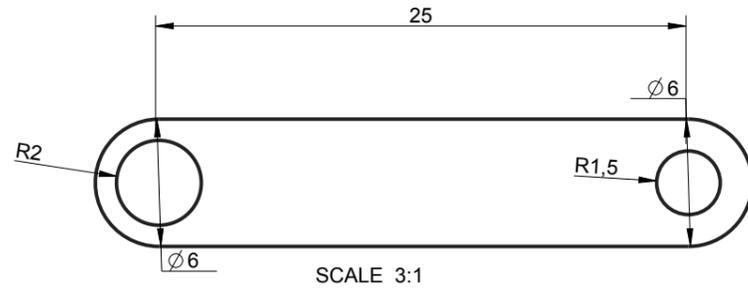
SCALE 3:1



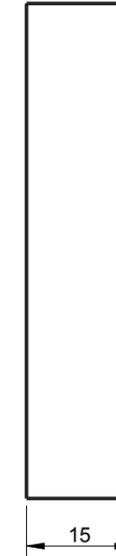
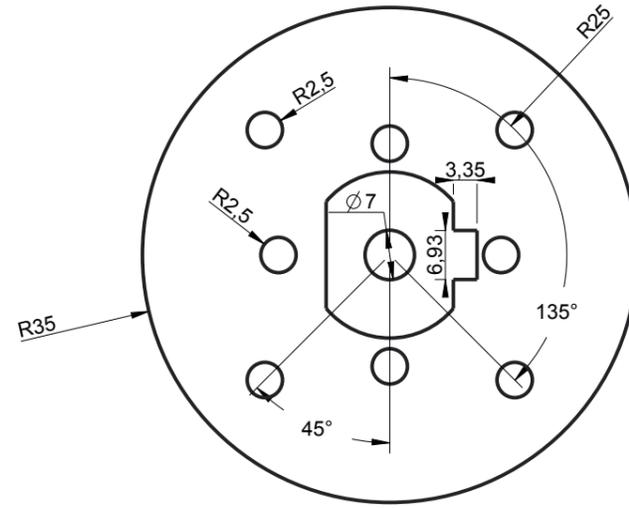
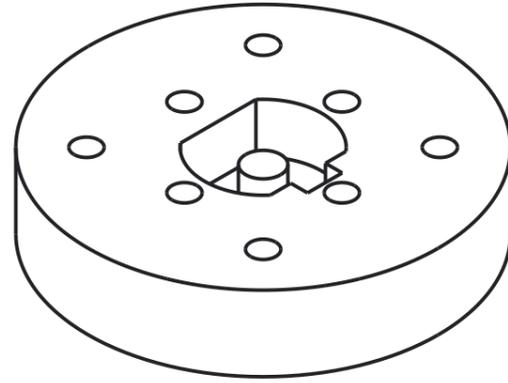
Nº Piezas	Descripción			Marca	Material	Fichero
	Unidad dimens.	Toler. geom.	Fecha	Alumnos: Grupo5		
	mm.	ISO2768-m	Nov-16-15			
Escala	Estado superficial	Toler. general	Fecha plantilla			
3:1	UNE 1037	ISO2768-K	Oct-02 2014			
Denominación:						
PRT0005						
ESCUELA SUPERIOR DE INGENIEROS					Nº Plano:	
TECNUN SAN SEBASTIAN					Sustituye a:	
					Sustituido por:	



SCALE 3:1



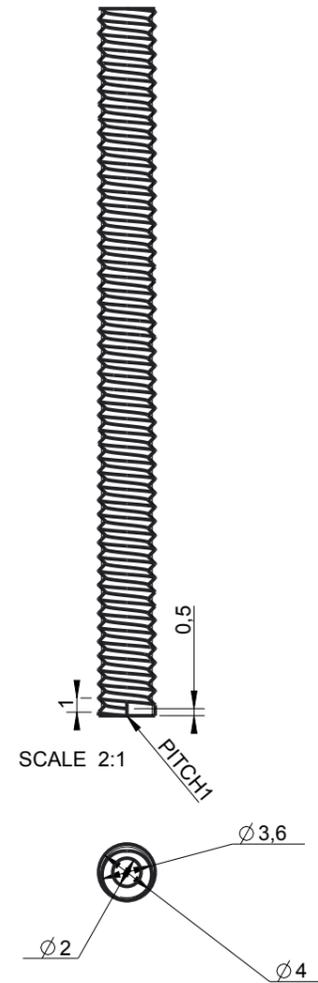
Nº Piezas	Descripción			Marca	Material	Fichero
	Unidad dimens. mm.	Toler. geom. ISO2768-m	Fecha Nov-16-15	Alumnos: Grupo5		
Escala 2:1	Estado superficial UNE 1037	Toler. general ISO2768-K UNE A3	Fecha plantilla Oct-02 2014			
Denominación:  PRT0006						
ESCUELA SUPERIOR DE INGENIEROS					Nº Plano:	
TECNUN SAN SEBASTIAN					Sustituye a:	
					Sustituido por:	



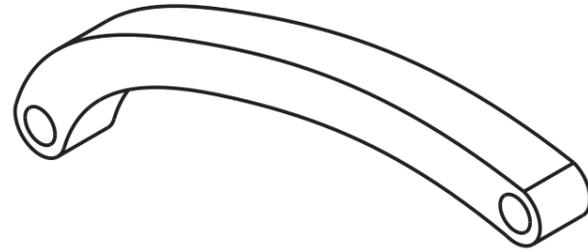
Nº Piezas	Descripción			Marca	Material	Fichero
	Unidad dimens.	Toler. geom.	Fecha	Alumnos: Grupo5		
	mm.	ISO2768-m	Nov-16-15			
Escala	Estado superficial	Toler. general	Fecha plantilla			
1:1	UNE 1037	ISO2768-K	Oct-02 2014			
Denominación:						
PRT0007						
ESCUELA SUPERIOR DE INGENIEROS					Nº Plano:	
<b>TECNUN</b> SAN SEBASTIAN					Sustituye a:	
					Sustituido por:	



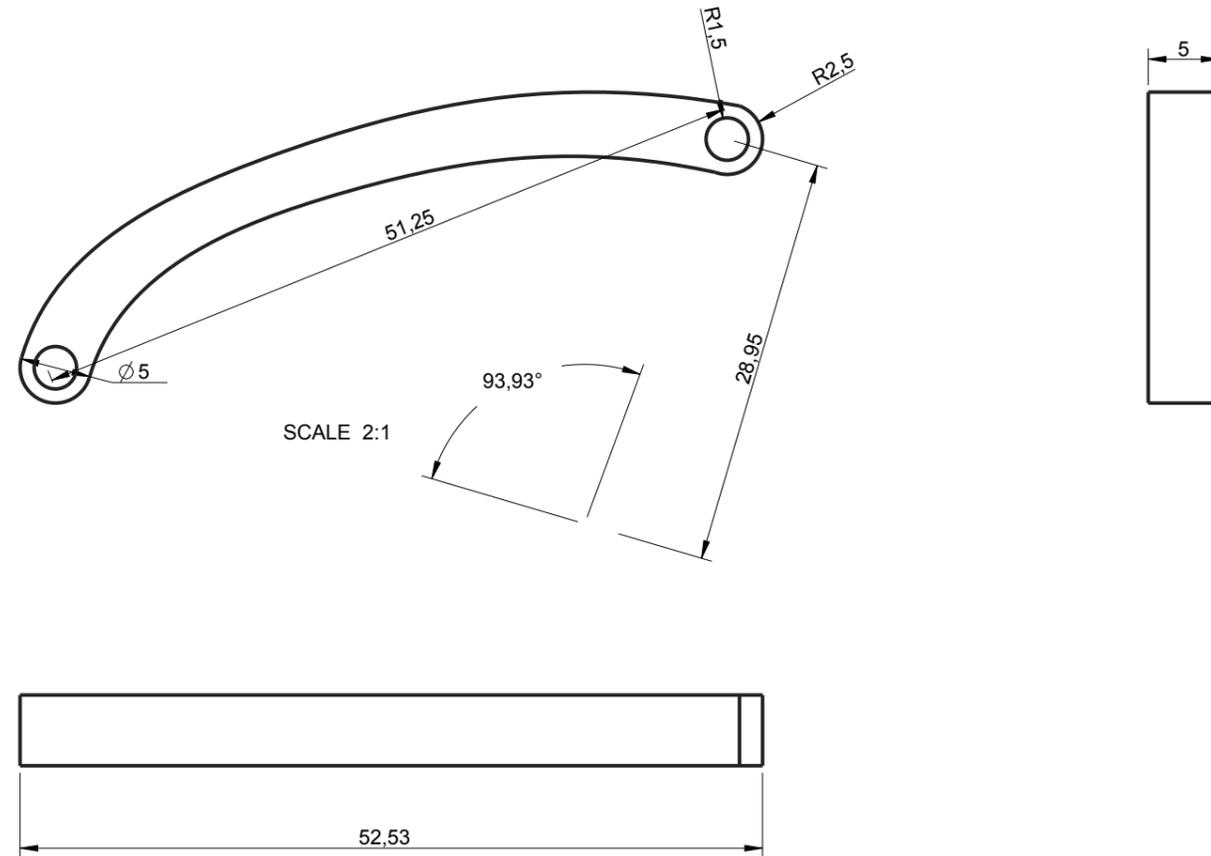
SCALE 2:1



Nº Piezas		Descripción		Marca	Material	Fichero
Metodo Proyección	Unidad dimens.	Toler. geom.	Fecha	Alumnos:		
	mm.	ISO2768-m	Nov-16-15	Grupo5		
Escala	Estado superficial	Toler. general	Fecha plantilla			
1:1	UNE 1037	ISO2768-K	Oct-02 2014			
Denominación:						
PRT0010						
ESCUELA SUPERIOR DE INGENIEROS					Nº Plano:	
<b>TECNUN</b> SAN SEBASTIAN					Sustituye a:	
					Sustituido por:	



SCALE 2:1



SCALE 2:1

Nº Piezas	Descripción			Marca	Material	Fichero
	Unidad dimens. mm.	Toler. geom. ISO2768-m	Fecha Nov-16-15	Alumnos: Grupo5		
Escala 1:1	Estado superficial UNE 1037	Toler. general ISO2768-K UNE A3	Fecha plantilla Oct-02 2014			
Denominación:  PRT0011						
ESCUELA SUPERIOR DE INGENIEROS  TECNUN SAN SEBASTIAN					Nº Plano: Sustituye a: Sustituido por:	