



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

Diseño, implementación y control de un vehículo
holonómico de seis ruedas con suspensión tipo rocker-
boogie

Trabajo Fin de Máster

Máster Universitario en Ingeniería Mecatrónica

AUTOR/A: Martínez Esteve, Pedro

Tutor/a: Casanova Calvo, Vicente Fermín

CURSO ACADÉMICO: 2022/2023



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE

TRABAJO DE FIN DE MÁSTER
MÁSTER EN INGENIERÍA MECATRÓNICA

Autor:

Pedro Martínez Esteve

Tutor:

Vicente Fermín Casanova Calvo

RESUMEN

El presente proyecto se basa en el desarrollo y la implementación de un vehículo similar al *Rover Curiosity* empleado por la *NASA* para recoger muestras en Marte.

Este vehículo está diseñado mecánicamente para sortear obstáculos tanto frontal como lateralmente, pudiendo hacerlo incluso en aquellas ocasiones en las que algunas de sus ruedas no está en contacto con el suelo, por lo que es óptimo para la exploración en terrenos irregulares.

El diseño se obtendrá en formato *STL* y posteriormente será tratado en *Solidworks* para ser compatible con la simulación.

Después, se realizará dicha simulación en *Simscape Multibody*, ensamblando el conjunto y tratando de recrear las condiciones más realistas, para verificar el correcto funcionamiento y diseñar los bucles de control de los motores. En esta fase se creará un entorno con obstáculos e irregularidades que el vehículo deberá superar con solvencia.

En el paso siguiente, se fabricarán algunos componentes y se comprarán los restantes, se ensamblará el vehículo manualmente y se realizará todo el montaje de la parte eléctrica y electrónica.

Y, por último, se programará e implementará la comunicación *Bluetooth* y el control manual, todo mediante un *Arduino*, para poder manejarlo en tiempo real con una *gamepad*.

ABSTRACT

This project is based on the development and implementation of a vehicle similar to the Rover Curiosity used by NASA to collect samples on Mars.

This vehicle is mechanically designed to avoid obstacles both frontally and laterally, being able to do so even on those occasions when some of its wheels are not in contact with the ground, making it optimal for exploring uneven terrain.

The design will be obtained in STL format and will later be processed in Solidworks to be compatible with the simulation.

Afterwards, said simulation will be carried out in Simscape Multibody, assembling the set and trying to recreate the most realistic conditions, to verify the correct operation and design the motor control loops. In this phase, an environment will be created with obstacles and irregularities that the vehicle must overcome with solvency.

In the next step, some components will be manufactured and the rest will be purchased, the vehicle will be manually assembled and all the assembly of the electrical and electronic parts will be carried out.

And finally, Bluetooth communication and speed control will be programmed and implemented, all through an Arduino, to be able to handle it in real time with a gamepad.

RESUM

El present projecte es basa en el desenvolupament i la implementació d'un vehicle similar al *Rover Curiosity* emprat per la NASA per a recollir mostres en Mart.

Aquest vehicle està dissenyat mecànicament per a superar obstacles tant frontal com lateralment, podent fer-ho fins i tot en aquelles ocasions en les quals algunes de les seues rodes no estiguen en contacte amb el sòl, per la qual cosa és òptim per a l'exploració en terrenys irregulars.

El disseny s'obtindrà en format *STL* i posteriorment serà tractat en *Solidworks* per a ser compatible amb la simulació.

Després, es realitzarà aquesta simulació en *Simscape Multibody*, assemblant el conjunt i tractant de recrear les condicions més realistes, per a verificar el correcte funcionament i dissenyar els bucles de control dels motors. En aquesta fase es crearà un entorn amb obstacles i irregularitats que el vehicle haurà de superar amb solvència.

En el pas següent, es fabricaran alguns components i es compraran els restants, s'assemblarà el vehicle manualment i es realitzarà tot el muntatge de la part elèctrica i electrònica.

I, finalment, es programarà i implementarà la comunicació *Bluetooth* i el control de velocitat, tot mitjançant un *Arduino*, per a poder manejar-ho en temps real amb una *gamepad*.

ÍNDICE

RESUMEN	1
ABSTRACT	2
RESUM	3
1. INTRODUCCIÓN	8
1.1. OBJETIVOS.....	8
1.2. ESTRUCTURA DE LA MEMORIA.....	9
1.3. ANTECEDENTES.....	10
1.3.1. ROVER CURIOSITY.....	10
1.3.1.1. MISIÓN.....	11
1.3.1.2. CARACTERÍSTICAS CONSTRUCTIVAS.....	13
2. ESTUDIO TEÓRICO	15
2.1. SUSPENSIÓN ROCKER-BOGIE.....	15
2.2. CINEMÁTICA.....	16
2.2.1. CINEMÁTICA DIRECTA.....	17
2.2.2. CINEMÁTICA INVERSA.....	19
2.3. ODOMETRÍA.....	19
2.4. MOVIMIENTOS DEL VEHÍCULO.....	21
2.4.1. DESPLAZAMIENTO FRONTAL.....	21
2.4.2. DESPLAZAMIENTO LATERAL.....	22
2.4.3. TRAZO DE CURVAS.....	22
2.4.4. ROTACIÓN.....	23
2.5. CONTROL AUTOMÁTICO.....	24
2.5.1. COMPONENTES DE UN BUCLE DE CONTROL.....	24
2.5.1.1. CONTROLADOR.....	25
2.5.1.2. SISTEMA, ENTRADA Y SALIDA.....	25
2.5.1.3. SENSOR.....	26
2.5.1.4. REFERENCIA Y ERROR MEDIDO.....	27
2.5.2. COMPONENTES DE UN CONTROL PID.....	27
2.5.2.1. ACCIÓN PROPORCIONAL.....	27
2.5.2.2. ACCIÓN INTEGRAL.....	28
2.5.2.3. ACCIÓN DERIVADA.....	28

3. DISEÑO: SOLIDWORKS.....	29
3.1. LISTADO DE PIEZAS.....	29
3.2. CUERPO CENTRAL.....	31
3.2.1. CHASIS.....	32
3.2.2. ELEMENTO DE CONSTRUCCIÓN.....	33
3.2.3. ELEMENTO DE CONSTRUCCIÓN 2.....	33
3.2.4. SUJECIÓN RODAMIENTO PRINCIPAL.....	34
3.3. SUSPENSIONES LATERALES.....	35
3.3.1. TUBOS PVC.....	36
3.3.2. SERVOMOTOR.....	36
3.3.3. SUJECIÓN SUSPENSIÓN.....	37
3.3.4. ENLACE SUSPENSIÓN-CONSTRUCCIÓN.....	38
3.3.5. ACOPLAMIENTO RUEDA DELANTERA.....	39
3.3.6. ACOPLAMIENTO RUEDA TRASERA.....	39
3.3.7. ALOJAMIENTO RODAMIENTO TRASERO.....	40
3.3.8. ACOPLAMIENTO RODAMIENTO TRASERO.....	40
3.4. RUEDAS Y ARCOS.....	41
3.4.1. ALOJAMIENTO MOTOR.....	42
3.4.2. MOTOR.....	43
3.4.3. ARCO ESTRUCTURAL.....	44
3.4.4. ACOPLAMIENTO RODAMIENTO SERVO.....	45
3.4.4. RUEDA.....	45
3.4.5. ACOPLA INTERIOR RUEDA.....	46
4. SIMULACIÓN: SIMSCAPE MULTIBODY.....	48
4.1. PIEZAS PARA SIMULACIÓN.....	48
4.1.1. SIMPLIFICACIÓN DE LA GEOMETRÍA.....	48
4.1.2. PREPARACIÓN DE LOS ARCHIVOS PARA SIMSCAPE MULTIBODY.....	50
4.2. ENSAMBLAJE DEL VEHÍCULO EN SIMSCAPE MULTIBODY.....	53
4.2.1. RELACIONES ENTRE PIEZAS.....	55
4.2.2. ESTABLECIMIENTO DE COEFICIENTES DE ROZAMIENTO.....	57
4.3. DISEÑO DEL ENTORNO.....	58
4.3.1. CREACIÓN DE OBSTÁCULOS.....	59
4.3.1.1. ESCALONES FRONTALES.....	60
4.3.1.2. ESCALONES CON TRES RUEDAS.....	61

4.3.1.3. RAMPA LATERAL.....	61
4.3.1.4. RAMPA FRONTAL.....	62
4.4. RELACIÓN ENTRE VEHÍCULO Y ENTORNO.....	63
4.4.1. DEFINICIÓN DE GRADOS DE LIBERTAD CON EL MUNDO.....	64
4.4.2. DEFINICIÓN DE LA POSICIÓN INICIAL DEL VEHÍCULO.....	65
4.4.3. CREACIÓN DE FUERZAS DE CONTACTO.....	66
4.4.3.1. DEFINICIÓN DE COEFICIENTES.....	67
4.4.4. SENSORIZACIÓN DEL VEHÍCULO.....	69
4.5. DISEÑO DE LA TRAYECTORIA.....	70
4.6. GENERACIÓN DE SEÑALES DE LOS MOTORES.....	72
4.6.1. SERVOS.....	72
4.6.2. MOTORES DE TRACCIÓN.....	76
4.7. CONTROL DE LOS MOTORES DE TRACCIÓN.....	79
4.7.1. BUCLE DE CONTROL E IMPLEMENTACIÓN.....	79
4.7.2. DISEÑO DEL CONTROLADOR.....	80
4.8. RESULTADOS DE LA SIMULACIÓN.....	82
4.8.1. VELOCIDAD ANTE REFERENCIA.....	83
4.8.2. POSICIÓN DE LOS SERVOMOTORES.....	83
4.8.3. POSICIÓN DEL CENTRO DE MASAS DE VEHÍCULO.....	84
4.8.4. ÁNGULO DE CABECEO DEL CHASIS DEL VEHÍCULO.....	86
5. IMPLEMENTACIÓN REAL.....	87
5.1. FABRICACIÓN.....	87
5.1.1. SINTERIZADO SELECTIVO POR LÁSER (SLS).....	87
5.1.2. CORTE POR HAZ LÁSER (LBC).....	88
5.1.3. MODELADO POR DEPOSICIÓN FUNDIDA (FDM).....	88
5.1.4. MODIFICACIONES REALIZADAS.....	89
5.1.4.1. CHASIS.....	89
5.1.4.2. ACOPLÉ INTERIOR DE LA RUEDA.....	91
5.1.4.3. ARCO ESTRUCTURAL.....	92
5.1.5. RELACIÓN DE PIEZAS Y PROCESOS DE FABRICACIÓN.....	93
5.2. ENSAMBLAJE.....	94
5.3. ELÉCTRICA Y ELECTRÓNICA.....	103
5.3.1. ARDUINO MEGA 2560 REV3.....	103
5.3.2. SERVOMOTOR MG996R.....	103

5.3.3. MOTOR DC 12V 100RPM.....	104
5.3.4. MÓDULO BLUETOOTH HC-06.....	105
5.3.5. DRIVER L298N.....	106
5.3.6. DRIVER PCA9685.....	108
5.4. CONTROL MANUAL.....	109
6. PLIEGO DE CONDICIONES.....	111
6.1. DEFINICIÓN Y ALCANCE.....	111
6.1.2. OBJETO.....	111
6.2. CONDICIONES Y NORMAS DE CARÁCTER GENERAL.....	111
6.3. CONDICIONES DE ESPECIFICACIONES TÉCNICAS.....	112
6.3.1. NORMATIVA APLICABLE A LA GENERACIÓN DE PLANOS.....	112
6.3.2. NORMATIVA APLICABLE A LA GESTIÓN DE LA CALIDAD.....	112
6.3.3. NORMATIVA APLICABLE A PROYECTOS ELÉCTRICOS.....	113
6.3.4. NORMATIVA SOBRE FABRICACIÓN Y COMPONENTES.....	113
7. PRESUPUESTO.....	114
7.1. DESGLOSE DE COSTES MATERIALES.....	114
7.1.1. MATERIALES.....	114
7.2. DESGLOSE DE COSTES REPERCUTIDOS.....	115
7.2.1. PERSONAL.....	115
7.2.2. FABRICACIÓN E INSTALACIONES.....	116
7.2.3. SOFTWARE Y LICENCIAS.....	117
7.3. PRESUPUESTO FINAL.....	118
7.3.1. PRESUPUESTO DE COSTES MATERIALES.....	118
7.3.2. PRESUPUESTO DE COSTES TOTAL.....	118
8. PROPUESTAS DE MEJORA.....	119
9. CONCLUSIONES.....	120
10. BIBLIOGRAFÍA.....	122
ÍNDICE DE FIGURAS Y TABLAS.....	125
ANEXO 1: PLANOS.....	130
ANEXO 2: CÓDIGO.....	152

1. INTRODUCCIÓN

En este punto se aporta al lector, de forma breve y concisa, un resumen de los objetivos que se pretenden abordar en este trabajo, así como una corta descripción de la memoria del mismo.

1.1. OBJETIVOS

El objetivo del trabajo es diseñar un vehículo holonómico, es decir, que todos sus grados de libertad sean controlables. Dichos grados son el desplazamiento en los ejes "X", "Y" y "Z", así como la rotación respecto a éste último.

Además, el vehículo está diseñado para poder sortear obstáculos y desniveles, con todas las ruedas al mismo tiempo o únicamente con algunas de ellas, pero siempre de forma que no pierda la verticalidad ni quede encallado.

Por lo tanto, los motivos por los que ha sido escogido este vehículo para realizar el Trabajo de Fin de Máster son los siguientes:

- La complejidad del sistema de suspensiones y la necesidad de que el vehículo supere obstáculos sin volcar hace que desde el punto de vista mecánico suponga un reto.
- Antes de realizar la implementación se debe hacer una simulación con Simscape Multibody, ampliando conocimientos sobre herramientas profesionales de simulación de sistemas mecatrónicos.
- Las piezas se fabricarán por métodos de fabricación aditiva, ámbito del que cada día es más importante ser conocedor.
- La programación e implementación final se realizará con Arduino Due, añadiendo un gamepad con el que se controlará el vehículo.

-
- Y por último, el especial interés del autor por la astronomía y los vehículos terrestres.

1.2. ESTRUCTURA DE LA MEMORIA

El documento se estructura en diez apartados, cada uno de los cuales consta de una breve introducción.

- El primer apartado, es decir, el actual, no es más que una breve introducción que pretende servir de prólogo para ayudar al lector a comprender mejor las motivaciones, los objetivos y la estructura del trabajo en cuestión.
- El segundo apartado también tiene como objetivo auxiliar al lector en la comprensión del texto, pero en este caso no desde el punto de vista motivacional, sino teóricamente, introduciendo conceptos que pueden resolver dudas futuras o mejorar el grado de entendimiento del documento.
- El tercer apartado describe los componentes mecánicos del vehículo, desde el punto de vista de su geometría y su funcionalidad.
- El cuarto apartado, y también el más extenso, ilustra los pasos seguidos para la creación de la simulación en el software Simscape Multibody. Estos van desde el modelado del entorno, el ensamblaje del vehículo y la relación entre ambos, hasta la generación de señales y el diseño de los bucles de control de los motores, finalizando con un análisis de los resultados obtenidos.
- El quinto apartado documenta la implementación real, incluyendo la fabricación y el ensamblaje de las piezas del vehículo, el montaje eléctrico y el diseño y la implementación del control manual.
- El sexto apartado, o pliego de condiciones, recoge las delimitaciones generales, técnicas, económicas y legales que determinan la viabilidad, seguridad y correctitud del proyecto.

-
- El séptimo apartado describe el presupuesto del proyecto, desglosando cada uno de los recursos necesarios y sus correspondientes costes.
 - El octavo apartado enumera las propuestas de mejora del diseño del vehículo, descubiertas a raíz de la realización del propio trabajo, tras muchas horas de ensayos, errores y estudio.
 - El noveno apartado, o conclusiones, refleja la experiencia vivida, el aprendizaje adquirido y las lecciones aprendidas por el autor durante el desarrollo del proyecto.
 - El décimo apartado es un listado de todas las fuentes de las que el autor se ha hecho servir durante la realización del proyecto, tanto a modo de aprendizaje, como de inspiración o apoyo para encontrar información.

Para finalizar, la información vital para el desarrollo del proyecto, pero que puede resultar poco trascendente o demasiado cargante para el lector, se ha recogido en dos anexos:

- El primer anexo es una recopilación de los planos de todas las piezas que integran el vehículo.
- El segundo anexo contiene todo el código que ha sido utilizado para programar correctamente el microcontrolador y conseguir un funcionamiento óptimo del sistema.

1.3. ANTECEDENTES

El vehículo que se desarrolla en este proyecto está claramente inspirado en los astromóviles marcianos de seis ruedas rover *Curiosity* y rover *Perseverance*. El diseño de ambos es prácticamente idéntico, así como su misión, por lo que únicamente entraremos en detalle con el *Curiosity*.

1.3.1. ROVER *CURIOSITY*

El *Curiosity* fue enviado a Marte el 26 de noviembre de 2011 por la NASA, aterrizando en el planeta rojo, concretamente en el cráter Gale, el 6 de agosto de 2012, tras más de ocho meses de vuelo.

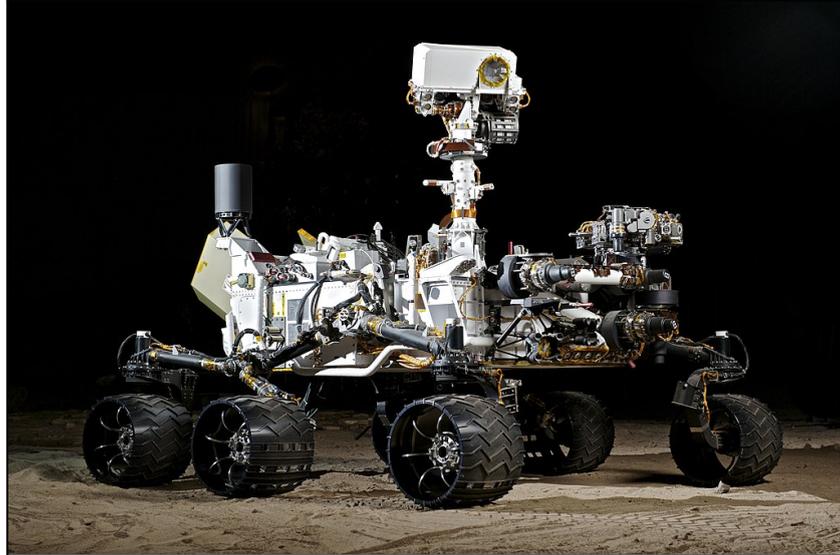


Figura 1. Rover *Curiosity*.

1.3.1.1. MISIÓN

El lugar del aterrizaje, anteriormente mencionado, no fue casual, como es de esperar, ya que este cráter es uno de los puntos en los que la NASA estima que hay más probabilidades de que exista o haya existido vida, por haber señales de que ha albergado agua en su superficie.



Figura 2. Cráter de Gale, lugar de aterrizaje del *Curiosity*.

Las principales tareas que se le encomendaban al robot eran: la investigación del clima y la geología marciana, la recogida de información sobre si el cráter de Gale en algún momento había ofrecido condiciones favorables para la vida microbiana y el análisis del entorno para aportar datos relevantes en el estudio de la habitabilidad de Marte y la accesibilidad para los seres humanos en futuras misiones.

Afortunadamente, cumplió con creces la labor que se esperaba de él, encontrando, aproximadamente un año después de su aterrizaje, restos de ondas y surcos formados hace millones de años por agua, al parecer perteneciente a un lago formado dentro del cráter. Por ello, el *Curiosity* es el más famoso de todos los vehículos que han transitado el planeta vecino.

A pesar de que su estancia activa en Marte estaba estipulada en aproximadamente dos años, a día de hoy, casi once años después, el vehículo sigue activo y explorando la superficie marciana, habiendo recorrido ya más de treinta kilómetros.

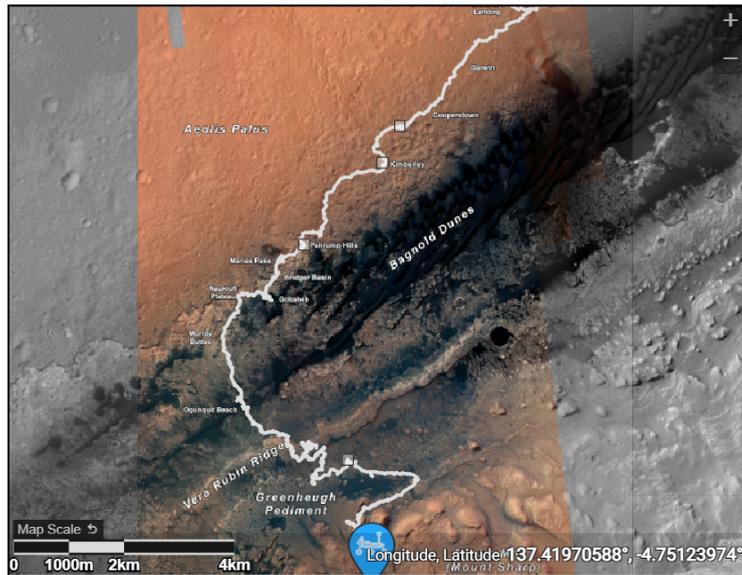


Figura 3. Recorrido y ubicación actual del *Curiosity*.

1.3.1.2. CARACTERÍSTICAS CONSTRUCTIVAS

El *Curiosity* mide 2,9 metros de largo, 2,7 metros de ancho y 2,2 metros de alto, con un peso de 899 kilogramos, 80 de los cuales corresponden a instrumentos científicos.



Figura 4. Dos ingenieros del JPL junto al *Curiosity*.

En cuanto a la alimentación, incluye un generador termoeléctrico de radioisótopos, que consiste en una batería que convierte el calor desprendido por un material radioactivo en electricidad por el efecto Seebeck.

Además, equipa instrumentos de telecomunicaciones muy avanzados, dos ordenadores y cuatro procesadores, así como los siguientes instrumentos científicos:

- Sistema de cámaras "MastCam".
- Centro de cámaras de análisis químico "ChemCam".
- Cámaras de navegación "NavCams".
- Estación de monitorización medioambiental "REMS".
- Cámaras de prevención de peligros "HazCams".
- Captador de imágenes del brazo robótico "MAHLI".
- Espectrómetro de rayos X y partículas alfa "APXS".
- Instrumento de química y mineralogía "CheMin".
- Analizador de muestras "SAM".
- Herramienta eliminadora del polvo "DRT".
- Detector de radiación "RAD".
- Albedo dinámico de neutrones "DAN".
- Captador de imágenes del descenso "MARDI".
- Brazo Robótico.

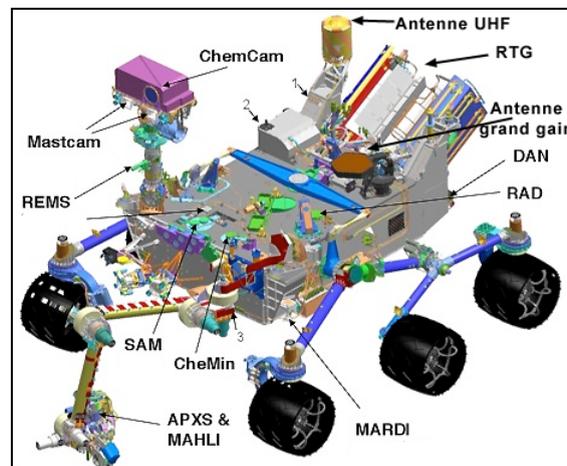


Figura 5. Esquema de instrumentos científicos del Curiosity.

2. ESTUDIO TEÓRICO

En esta sección se introducen conceptos teóricos que son de necesario conocimiento para adquirir una comprensión completa del trabajo.

2.1. SUSPENSIÓN ROCKER-BOGIE

Este sistema de suspensiones fue diseñado específicamente por la NASA para este tipo de misiones, con el objetivo de crear un vehículo excelentemente estable en terrenos irregulares, evitando así accidentes o pérdidas de tracción extremas que puedan llevar a un atascamiento, lo cual puede llegar a suponer un fracaso inmediato de la misión espacial.

El mecanismo está formado por seis ruedas agrupadas en dos conjuntos simétricos, cada una de las cuales está accionada por un motor independiente.

Dentro de cada uno de los conjuntos podemos encontrar una barra angulada llamada *rocker*, que une una de las ruedas, generalmente la delantera, con un balancín, llamado *bogie*, que enlaza las dos ruedas restantes.

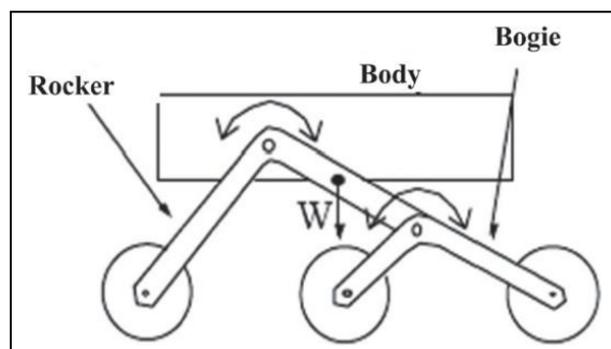


Figura 6. Esquema de un Sistema *Rocker-Bogie*.

El *rocker* se une al chasis del vehículo mediante un mecanismo diferencial, de modo que si uno de los *rockers* sube, el otro baja, asegurando el contacto con el suelo en mayor medida.

Así, gracias a dicho mecanismo y a la doble articulación de bisagra (chasis-rocker y rocker-bogie) conseguimos que por acción de la gravedad el vehículo se adapte perfectamente a elevaciones, depresiones o sinuosidades en el relieve del terreno, manteniendo siempre el mayor número de ruedas en contacto con el suelo, y permaneciendo el chasis paralelo al suelo.

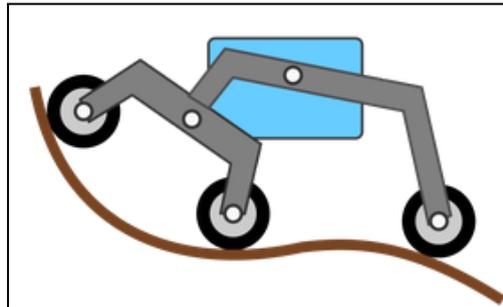


Figura 7. Mecanismo Rocker-Bogie en un Terreno Irregular.

2.2. CINEMÁTICA

En este apartado se explicará la cinemática del vehículo, siendo esta la descripción detallada del movimiento del mismo.

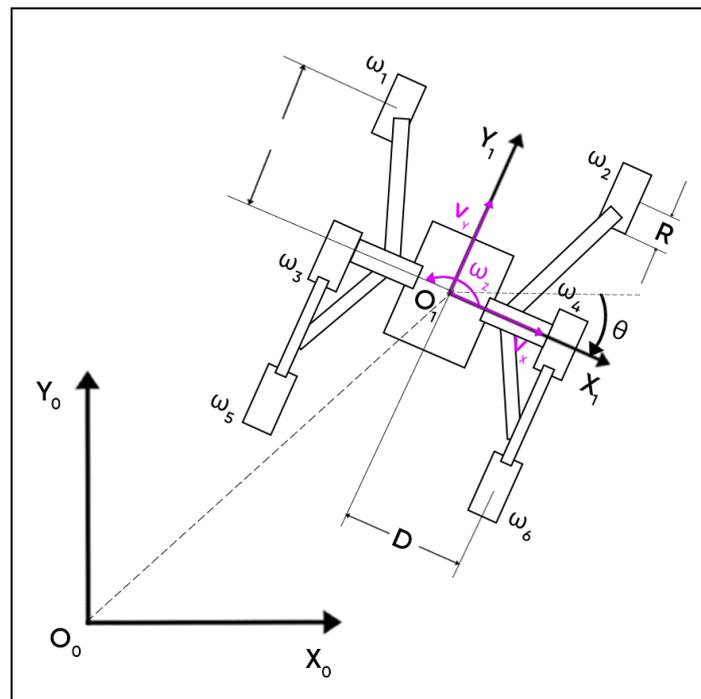


Figura 8. Variables de la Cinemática del Vehículo.

Definimos las siguientes variables en el mecanismo:

- x , y , θ : Siendo x e y las posiciones del centro de masas del vehículo longitudinal y transversalmente, y θ la rotación respecto al eje vertical.
- O_0 , X_0 , Y_0 : Como sistema de referencia global, siendo invariable con el movimiento del vehículo.
- O_1 , X_1 , Y_1 : Como sistema de referencia local, moviéndose y girando siempre solidario con el centro de masas del vehículo.
- L , D : Como distancias entre el centro de masas y el centro de las ruedas.
- R : Como radio de las ruedas.
- ω_1 , ω_2 , ω_3 , ω_4 , ω_5 , ω_6 : Como velocidades angulares de las ruedas [rad/s].
- v_x , v_y , ω_z : Como velocidades del centro de masas del vehículo, tanto en traslación lateral y frontal [m/s], como en rotación respecto a la vertical [rad/s].

A continuación, se analizará la cinemática directa e inversa del vehículo.

2.2.1. CINEMÁTICA DIRECTA

Es el estudio que nos permitirá conocer la velocidad del centro de masas del vehículo a partir de las velocidades angulares de las ruedas.

Respecto a la velocidad lateral del vehículo (v_x), debemos saber qué ruedas equilibrarían el vehículo si giraran en sentidos contrarios tres a tres, para conocer cuales se anulan entre ellas.

Determinamos que estas deben ser las ruedas 1, 4 y 5 frente a las ruedas 2, 3 y 6, ya que si giraran con la misma velocidad pero en sentidos opuestos, el vehículo permanecería inmóvil respecto al eje "X".

Respecto a la velocidad frontal (v_y), es lógico deducir que todas las ruedas portarán el mismo signo, ya que cuantas más ruedas rueden en el mismo sentido, mayor velocidad tendremos en el sentido positivo del eje "Y".

Y finalmente, en cuanto a la rotación respecto a la vertical (ω_z), lo primero que podemos deducir es que las ruedas centrales no aportan, por lo que la tercera y cuarta columna de la matriz tendrán un 0.

Seguidamente, solo nos queda averiguar qué ruedas girarán en el mismo sentido si queremos hacer rotar al vehículo sobre su eje vertical, deduciendo que deberán dividirse en lateral derecho y lateral izquierdo, ya que si las ruedas de la derecha tienen sentido positivo, las de la izquierda deberán tener sentido negativo, y viceversa.

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} = \frac{R}{6} \begin{bmatrix} -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ \frac{1}{-(D+L)} & \frac{1}{(D+L)} & 0 & 0 & \frac{1}{-(D+L)} & \frac{1}{(D+L)} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \\ \omega_5 \\ \omega_6 \end{bmatrix}$$

Figura 9. Planteamiento Cinemática Directa del Vehículo.

De esta forma, a partir de las seis variables de entrada, es decir, las velocidades angulares de las ruedas, se pueden calcular las tres variables de salida, que son las velocidades de traslación y rotación del centro de masas del vehículo.

$$\begin{aligned}
 v_x(t) &= \frac{R}{6} \cdot (-\omega_1 + \omega_2 + \omega_3 - \omega_4 - \omega_5 + \omega_6) \\
 v_y(t) &= \frac{R}{6} \cdot (\omega_1 + \omega_2 + \omega_3 + \omega_4 + \omega_5 + \omega_6) \\
 \omega_z(t) &= \frac{R}{6 \cdot (D+L)} \cdot (-\omega_1 + \omega_2 - \omega_5 + \omega_6)
 \end{aligned}$$

Figura 10. Expresiones Cinemática Directa del Vehículo.

2.2.2. CINEMÁTICA INVERSA

Como su nombre indica, es el proceso inverso al explicado en el punto anterior, es decir, nos permitirá conocer las velocidades de las ruedas a partir de las velocidades del centro de masas del vehículo.

Este estudio es especialmente útil, ya que los parámetros que se desea controlar y conocer son la posición y la velocidad finales del vehículo, pero los que realmente podemos controlar por accionamiento son la orientación y la velocidad de las ruedas.

Además, habiendo calculado la cinemática directa, únicamente debemos despejar las velocidades angulares de las expresiones anteriores, por lo que las ecuaciones que describen la cinemática inversa son las siguientes:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \\ \omega_5 \\ \omega_6 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} -1 & 1 & -(D+L) \\ 1 & 1 & (D+L) \\ 1 & 1 & 0 \\ -1 & 1 & 0 \\ -1 & 1 & -(D+L) \\ 1 & 1 & (D+L) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}$$

Figura 11. Cinemática Inversa del Vehículo.

2.3. ODOMETRÍA

La odometría trata de estudiar la estimación de la posición de los vehículos con ruedas durante la navegación.

Básicamente, es la aplicación del estudio de la cinemática directa para obtener las posiciones del centro de masas del vehículo en un instante determinado.

Para ello, en primer lugar debemos tener claro cómo funciona la variación de una posición respecto del tiempo, y esto se resume en la siguiente expresión: “La posición de un cuerpo en un instante es el resultado de sumar la posición en el instante inmediatamente anterior más la variación de posición experimentada entre ambos instantes”.

O lo que es lo mismo:

$$\Delta T = t_1 - t_0$$

$$x_1 = x_0 + v_{x(t)} \cdot \Delta T$$

Figura 12. Variación de Posición Respecto al Tiempo.

Por lo tanto, las posiciones en un instante “t” del centro de masas del vehículo serán las siguientes:

$$x(t) = x_{(t-1)} + v_{x(t)} \cdot \Delta T = x_{(t-1)} + \frac{R}{6} \cdot (-\omega_1 + \omega_2 + \omega_3 - \omega_4 - \omega_5 + \omega_6) \cdot \Delta T$$

$$y(t) = y_{(t-1)} + v_{y(t)} \cdot \Delta T = y_{(t-1)} + \frac{R}{6} \cdot (\omega_1 + \omega_2 + \omega_3 + \omega_4 + \omega_5 + \omega_6) \cdot \Delta T$$

$$\theta(t) = \theta_{(t-1)} + \omega_z(t) \cdot \Delta T = \theta_{(t-1)} + \frac{R}{6 \cdot (D+L)} \cdot (-\omega_1 + \omega_2 - \omega_5 + \omega_6) \cdot \Delta T$$

Figura 13. Odometría del Vehículo.

Este estudio es imperfecto, ya que es posible que suceda una acumulación de errores por factores no contemplados tales como:

- Inexactitud en el tamaño de las ruedas.
- Mal alineamiento de las mismas.
- Patinaje o deslizamiento del vehículo por derrapes, fuerzas externas o mal contacto con el terreno.

Por esto mismo, un sistema de posicionamiento basado en odometría únicamente es posible en condiciones ideales, pero en condiciones reales necesitamos sensores de posición que nos den información veraz en tiempo real.

2.4. MOVIMIENTOS DEL VEHÍCULO

En este proyecto se desarrolla un vehículo holonómico, es decir, que debemos poder direccionarlo controladamente en todos los sentidos, es decir, generar traslación en "X" e "Y", y rotación en "Z".

Para optimizar la maniobrabilidad en espacios reducidos y mejorar la precisión de movimientos se ha diseñado el mecanismo para que pueda ejecutar cuatro tipos distintos de desplazamientos.

2.4.1. DESPLAZAMIENTO FRONTAL

Es el tipo de movimiento más simple, permitiendo al vehículo moverse hacia delante y hacia detrás en línea recta, sin rotación alguna en cualquiera de las seis ruedas, o lo que es lo mismo, desplazarse en el eje "X" en sentido positivo y negativo.

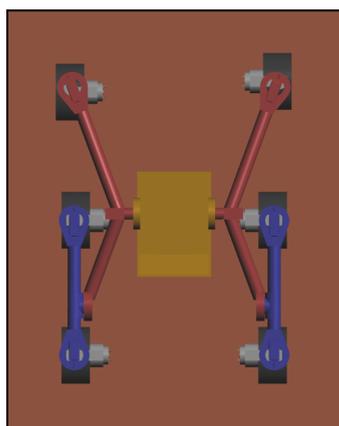


Figura 14. Vehículo en Posición de Desplazamiento Frontal.

Se obtiene accionando los motores de las seis ruedas a la vez, en el mismo sentido y a la misma velocidad.

2.4.2. DESPLAZAMIENTO LATERAL

Es el otro tipo de movimiento rectilíneo que puede ejecutar el vehículo, y le permite moverse hacia la izquierda o la derecha, o lo que es lo mismo, desplazarse en el eje "Y" en sentido positivo y negativo.

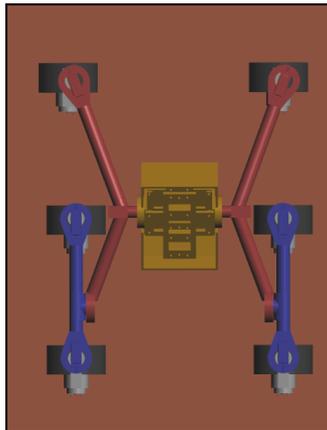
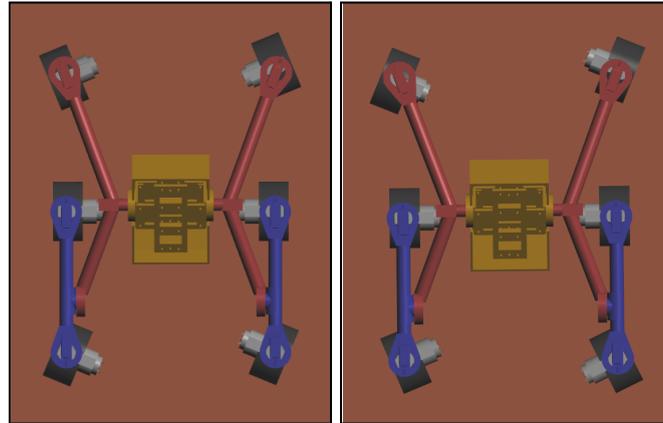


Figura 15. Vehículo en Posición de Desplazamiento Lateral.

Se obtiene accionando los motores de las seis ruedas a la vez, en el mismo sentido y a la misma velocidad, y haciendo girar todas las ruedas 90° exactos respecto a su posición inicial.

2.4.3. TRAZO DE CURVAS

Este tipo de movimiento es de tipo compuesto, ya que el vehículo se desplaza en los ejes "X" e "Y" simultáneamente.



Figuras 16(a) y 16(b). Vehículo en Posición de Girar a Izquierda y Derecha.

Se obtiene direccionando las dos ruedas delanteras un ángulo determinado en sentido horario o antihorario respecto a la posición inicial, y las dos ruedas traseras el mismo ángulo en sentido contrario. Las ruedas motrices se accionan todas a la vez, en el mismo sentido y a la misma velocidad.

2.4.4. ROTACIÓN

El último movimiento no genera desplazamiento del centro de masas del vehículo, sino que consiste en la rotación del mismo respecto del eje vertical o de cota, también llamado "Z".

Este movimiento se obtiene direccionando las ruedas delantera derecha y trasera izquierda 45° en sentido antihorario y las ruedas delantera izquierda y trasera derecha otros 45° en sentido horario, de modo que la proyección radial de las ruedas forme lo que se asemejaría a un rombo.

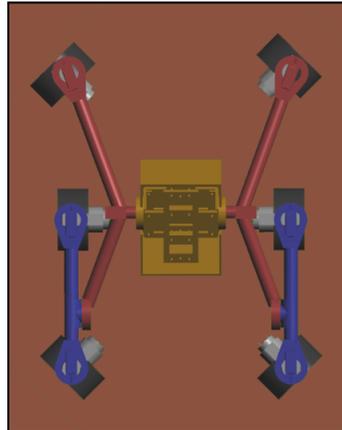


Figura 17. Vehículo en Posición de Giro Sobre el Eje Vertical.

Dependiendo del sentido de giro que deseemos, las parejas de ruedas mencionadas anteriormente deberán accionarse a la vez, a la misma velocidad, pero siempre en sentidos opuestos.

2.5. CONTROL AUTOMÁTICO

En este capítulo se explicará brevemente en qué consiste un control PID, siendo este uno de los métodos más utilizados para dirigir sistemas hacia posiciones o niveles determinados.

Supone un componente casi obligatorio en cualquier proceso industrial o ingenieril en el que se necesite controlar variables o parámetros sin intervención manual.

Sus usos varían desde el diseño de limitadores de velocidad y controles de temperatura hasta la consecución de sistemas de dirección automática de barcos.

2.5.1. COMPONENTES DE UN BUCLE DE CONTROL

A continuación, procederemos a explicar los elementos de un bucle de control automático, los cuales se pueden apreciar en la siguiente figura:

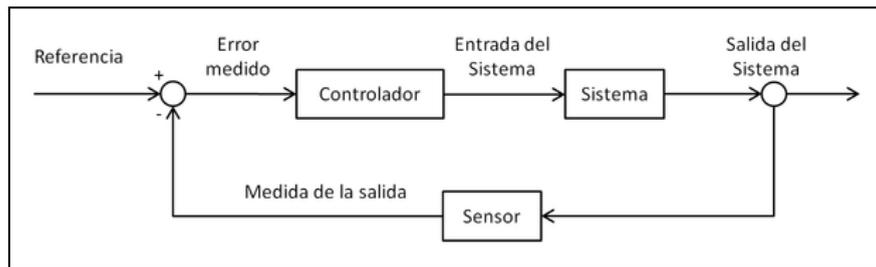


Figura 18. Bucle de Control Automático.

2.5.1.1. CONTROLADOR

En resumidas cuentas, un controlador es un mecanismo que a través de un lazo de retroalimentación permite regular variables de un proceso en general. El controlador PID calcula la diferencia entre nuestra variable real y la variable deseada, utilizando esta diferencia, también llamada error, para calcular la acción de control que deberá ejercer sobre el sistema.

Es necesario al menos un sensor, un controlador y un actuador.

2.5.1.2. SISTEMA, ENTRADA Y SALIDA

Es el proceso industrial o ingenieril que se pretende controlar. Debemos conocer con exactitud el funcionamiento de dicho sistema, para poder modelarlo y extraer las fórmulas que describen su comportamiento.

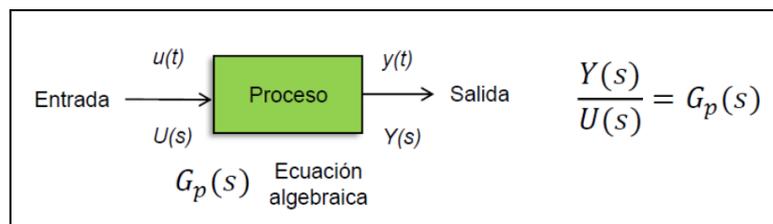


Figura 19. Representación de un Proceso Industrial.

Estas fórmulas, también llamadas funciones de transferencia, que relacionan la entrada con la salida del sistema, serán necesarias para diseñar el controlador, y un error en las mismas comportará que no podamos controlar correctamente el proceso.

En ocasiones, en el proceso hay perturbaciones que no podemos predecir o modelar, tales como el viento, deformaciones mecánicas o vibraciones por fuerzas externas, que pueden comprometer la efectividad del control.

La entrada, o entradas en casos de sistemas multivariables, es el valor para el que diseñamos todo el bucle, es decir, todo el trabajo se realiza para que este parámetro sea correcto, ya que será lo que propicie que el funcionamiento del sistema también lo sea.

La salida del sistema es la información real que podemos extraer de su comportamiento y la mediremos para poder mejorarlo.

2.5.1.3. SENSOR

Es un instrumento de medida que nos aporta información en tiempo real de la salida del proceso.



Figura 20. Sensor de Efusividad Térmica.

La función del sensor es la de aportar retroalimentación al sistema, de modo que, funcionando como un lazo cerrado, podamos calcular el error que existe en cada momento.

Su precisión es crucial, ya que los cálculos del controlador dependen de ella.

2.5.1.4. REFERENCIA Y ERROR MEDIDO

Es la entrada inicial al bucle de control, es decir, es el valor que deseamos que tenga la salida, por lo que, en un sistema ideal, ambos valores serían idénticos, pero en la realidad este nunca es el caso.

El error es la diferencia entre el valor deseado y el real, es decir, se obtiene restando la información del sensor o salida del sistema a la referencia.

2.5.2. COMPONENTES DE UN CONTROL PID

Un control de tipo PID puede tener tres acciones de control, correspondientes a las iniciales de su nombre, y son:

- Proporcional
- Integral
- Derivada

En función del proceso a controlar se escogerá cuáles de las acciones se integrarán y cuáles no, ya que no tienen por qué estar todas presentes, pudiendo existir controles:

- Proporcionales (P)
- Proporcionales-Integrales (PI)
- Proporcionales-Derivativos (PD)
- Proporcionales-Integrales-Derivativos (PID)

A continuación, se describirán las acciones y sus aportaciones al comportamiento del sistema.

2.5.2.1. ACCIÓN PROPORCIONAL

Se obtiene calculando el producto de la señal de error (E) por la constante proporcional (K_p).

$$U_P = K_P \cdot e(t)$$

Figura 21. Acción Proporcional.

Basado en el presente, mejora el comportamiento transitorio y permanente, modificando la salida en función del error, pero a pesar de lograr que el error en estado estacionario se aproxime a cero, no se puede eliminar por completo.

2.5.2.2. ACCIÓN INTEGRAL

Se obtiene calculando la integral de la función del error en muestreos anteriores y multiplicándola por la constante integral (K_I)

$$U_I = K_I \int_0^t e(t) dt$$

Figura 22. Acción Integral.

Basado en el pasado, permite conseguir un error nulo en régimen permanente, sumando los errores a lo largo del tiempo.

2.5.2.3. ACCIÓN DERIVADA

Se obtiene derivando la función del error y multiplicándola por la constante derivativa (K_D).

$$U_D = K_D \frac{de}{dt}$$

Figura 23. Acción Derivada.

Basada en el futuro, mejora el comportamiento transitorio, mediante la predicción del error, estimando en base a datos anteriores.

3. DISEÑO: SOLIDWORKS

En este capítulo se describirá detalladamente el mecanismo constructivo del vehículo en cuestión.

Lo dividiremos en cuatro apartados fundamentales, comenzando con un listado de todas las piezas, continuando con el cuerpo central del astromóvil, pasando con los sistemas laterales de suspensiones tipo *rocker-bogie* y finalizando con el subensamblaje que engloba los arcos y las ruedas.

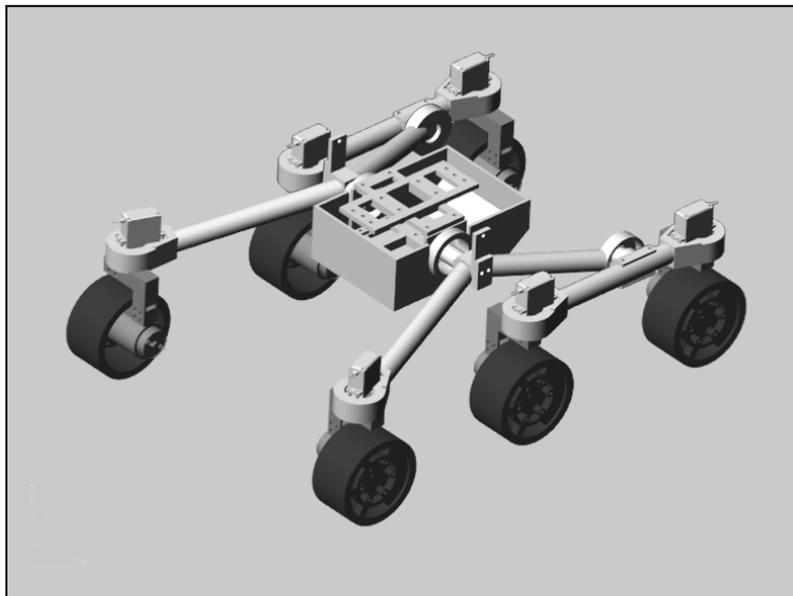


Figura 24. Ensamblaje Completo del Vehículo.

Cabe destacar que en este capítulo se describe el vehículo que será fabricado finalmente y volveremos a ver en el apartado de la implementación real, pero no será exactamente el mismo vehículo que se utilizará en la simulación.

3.1. LISTADO DE PIEZAS

Las piezas del vehículo han sido adquirido a través de la plataforma *Gumroad*, concretamente al usuario *kamilkosi*, el cual diseñó un Rover basado en el *Curiosity* de la NASA y decidió comercializarlo en dicha plataforma.

Por ello, los ficheros obtenidos están en formato STL, y son los que aparecen en la siguiente tabla.

Archivo STL	Descripción
AntirollBarLink.stl	Enlace Constructivo
Bearing.stl	Rodamiento
Chassis.stl	Chasis
ConstructionElement.stl	Elemento de Construcción
ConstrucionPart2.stl	Elemento de Construcción 2
FrontWheelCoupler.stl	Acoplamiento Rueda Delantera
MainAntirollBar.stl	Barra Constructiva
MainBearingHolder.stl	Sujeción Rodamiento Principal
MainSuspentionHolder.stl	Sujeción Suspensión
NEW_MotorHolder_for_37mm.stl	Alojamiento Motor
RearSuspentionBearingCoupler.stl	Acoplamiento Rodamiento Trasero
RearSuspentionBearingHolder.stl	Alojamiento Rodamiento Trasero
RearWheelCoupler.stl	Acoplamiento Rueda Trasera
SuspentionRollBarLink.stl	Enlace Suspensión-Construcción
Wheel.stl	Rueda
WheelArc.stl	Arco Estructural
WheelBearingHub.stl	Acoplamiento Rodamiento Servo
WheelInsidePart.stl	Acople Interior Rueda

Tabla 1. Relación de Archivos STL.

Además, para completar el mecanismo, debemos crear y añadir tres tamaños de tubos de PVC, así como añadir los motores y servos correspondientes:

Archivo STL	Descripción
PVC 130x25x21	Tubo de 130x25
PVC 150x25x21	Tubo de 150x25
PVC 180x25x21	Tubo de 180x25
Gearmotor	Motor
Sunfounder_20kg	Servomotor

Tabla 2. Relación de Piezas Adicionales.

A partir de ahora, y para simplificar el documento, llamaremos a las piezas por el nombre que aparece en la columna de la derecha de las dos tablas anteriores.

3.2. CUERPO CENTRAL

Es el elemento central de nuestro ensamblaje, ya que tiene la misión de mantener unidos los dos trenes laterales de ruedas, y además es el alojamiento del microprocesador, los *drivers* y demás electrónica del vehículo, a la que debe servir también de protección.

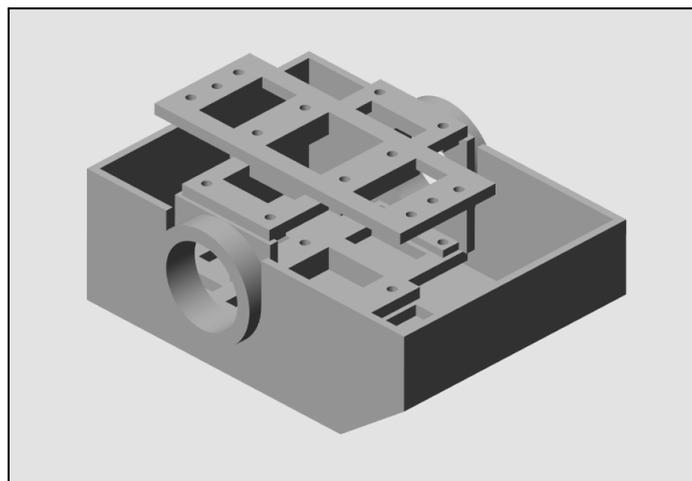


Figura 25. Cuerpo Central.

Este conjunto está formado por las siguientes piezas:

Pieza	Cantidad
Chasis	1
Elemento de Construcción	2
Elemento de Construcción 2	2
Sujeción Rodamiento Principal	2

Tabla 3. Listado de Piezas del Cuerpo Central.

A continuación, se detallarán las características constructivas de cada una de ellas.

3.2.1. CHASIS

El chasis es la parte principal del cuerpo central, encargada de albergar a la electrónica y al resto de partes del subensamblaje. Es la pieza sobre la que se ensamblará el resto del mecanismo.

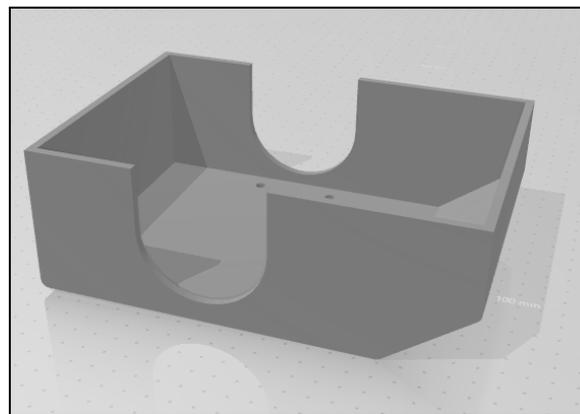


Figura 26. Chasis.

Tiene cuatro taladros en la parte inferior para atornillar los elementos constructivos y las sujeciones de los rodamientos principales, así como dos recortes en forma de arco para que estos últimos encajen.

Mide 70 mm de alto, 200 mm largo y 138 mm de ancho, con un espesor de pared de 4 mm.

3.2.2. ELEMENTO DE CONSTRUCCIÓN

Es uno de los dos elementos de construcción que se alojan dentro del chasis, para aportar robustez al conjunto y proporcionar un lugar donde fijar los componentes electrónicos.

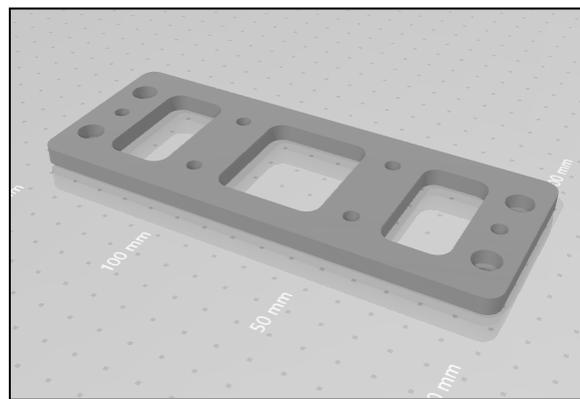


Figura 27. Elemento de Construcción.

Tiene diez taladros en su superficie, para unirse con el chasis, con las sujeciones de los rodamientos principales, con los otros elementos de construcción y para atornillar la electrónica.

Mide 155 mm de largo y 60 mm de ancho, con un espesor de 6 mm.

3.2.3. ELEMENTO DE CONSTRUCCIÓN 2

Es muy similar a la pieza anterior, teniendo esta seis taladros en lugar de diez, con el objetivo de unirse al resto de elementos constructivos y a la electrónica.

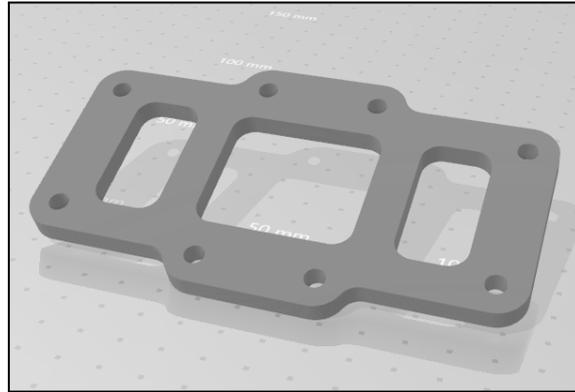


Figura 28. Elemento de Construcción 2.

Mide 125 mm largo y 70 mm de ancho, con un espesor de 6 mm.

3.2.4. SUJECIÓN RODAMIENTO PRINCIPAL

Esta pieza se inserta en el chasis y sirve para alojar el rodamiento que unirá el cuerpo central con la suspensión del lado pertinente.

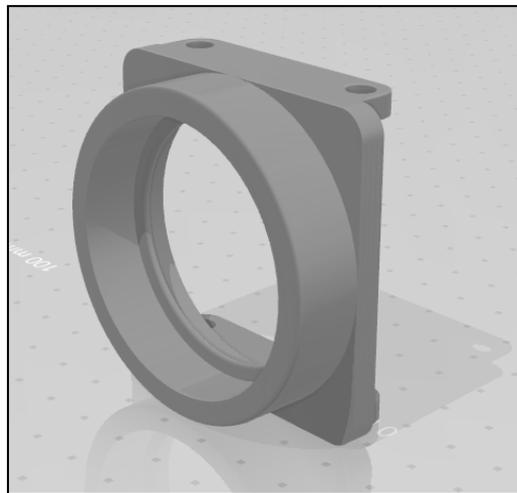


Figura 29. Sujeción Rodamiento Principal.

Tiene cuatro taladros para unirse con el chasis y los elementos de construcción.

Mide 28 mm de alto, 60 mm largo y 60 mm de ancho, con un espesor de pared de 6 mm. El diámetro interior del alojamiento es de 50 mm.

3.3. SUSPENSIONES LATERALES

Las suspensiones laterales, como se ha explicado en el estudio teórico, están compuestas por dos barras cada una, una de las cuales hace de *rocker* y la otra de *bogie*.

La función del mecanismo es de hacer de doble balancín, con dos articulaciones rotatorias estilo bisagra, que aseguran el contacto de las ruedas con el suelo.

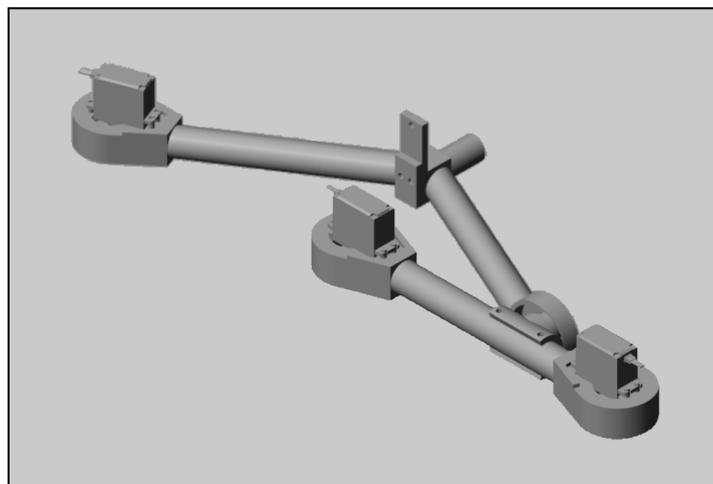


Figura 30. Suspensión Lateral Izquierda.

Este conjunto está formado por las siguientes piezas:

Pieza	Cantidad
Tubo 180x25	2
Tubo 130x25	2
Tubo 150x25	2
Servomotor	6
Sujeción Suspensión	2
Enlace Suspensión-Construcción	2

Acoplamiento Rueda Delantera	2
Acoplamiento Rueda Trasera	4
Alojamiento Rodamiento Trasero	2
Acoplamiento Rodamiento Trasero	2

Tabla 4. Listado de Piezas de las Suspensiones Laterales.

A continuación, se detallarán las características constructivas de cada una de ellas.

3.3.1. TUBOS PVC

Cada una de las suspensiones laterales cuenta con tres tubos de PVC de 25mm de diámetro exterior y 21mm de diámetro interior, siendo su longitud de 180, 150 y 130mm respectivamente.

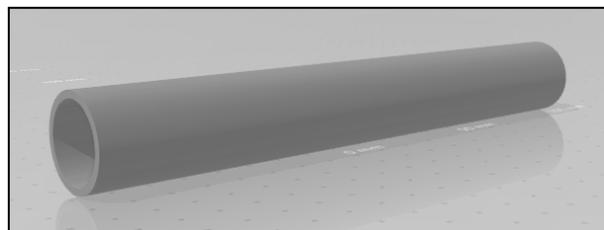


Figura 31. Tubo PVC 180x25x21.

El tubo más largo y el más corto se emplean en el *rocker*, uniendo la sujeción de la suspensión con el acoplamiento de la rueda delantera y el alojamiento del rodamiento trasero respectivamente.

A su vez, el tubo mediano une los dos acoplamientos de las ruedas traseras.

3.3.2. SERVOMOTOR

En cada una de las suspensiones laterales hay tres servomotores que se encargan de rotar los arcos que contienen las ruedas. En este apartado únicamente

hablaremos de las características constructivas del servomotor, y de cómo se ensambla en el diseño.

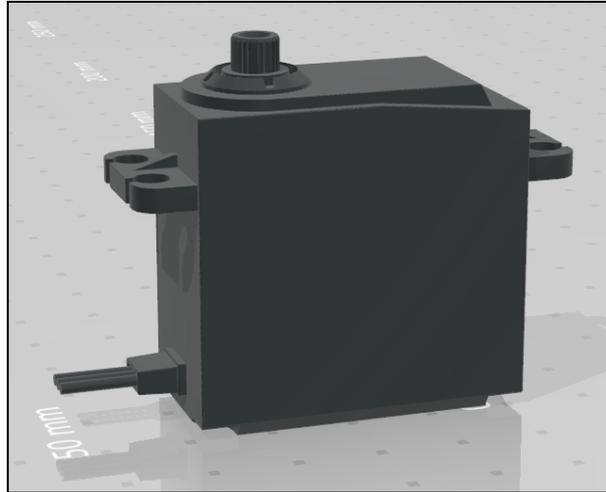


Figura 32. Servomotor.

Cada uno de los servomotores va ubicado encima de uno de los acoplamientos de las ruedas, teniendo cuatro taladros en los laterales para encajarlos en los salientes de estos.

Mide 62 mm de alto, 47 mm largo y 20 mm de ancho, siendo el tamaño estándar del modelo MG996R. Los cuatro taladros que tiene para colocar insertos son de 7 mm de diámetro.

3.3.3. SUJECIÓN SUSPENSIÓN

Es la pieza principal de la suspensión lateral, sobre la que se ensambla el resto. Tiene dos salientes en los laterales que se introducen en los tubos de PVC, y un saliente en el centro que se introduce en el rodamiento que se aloja en el cuerpo central.

Además tiene dos taladros que permitirán unirla al enlace suspensión-construcción.

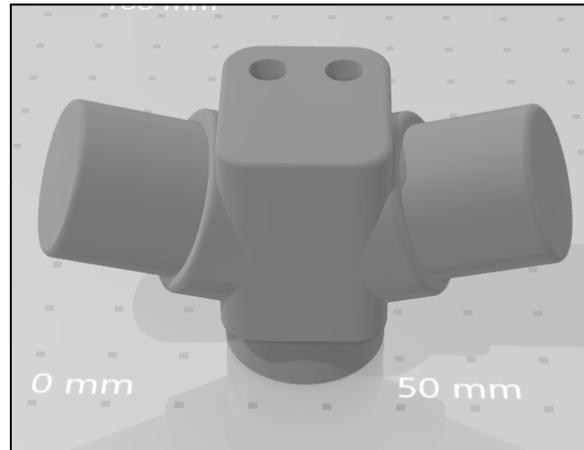


Figura 33. Sujeción Suspensión.

Tiene unas medidas totales de 40 mm de alto, 70 mm largo y 29 mm de ancho, con un espesor de pared de mm. Los dos salientes que se unen con los tubos de PVC son de 21 mm de diámetro, mientras que el que se introduce en el rodamiento es de 22 mm.

3.3.4. ENLACE SUSPENSIÓN-CONSTRUCCIÓN

Se encarga de unir la suspensión lateral con el sistema de equilibrado del vehículo, uniéndose a la sujeción de la suspensión con los dos taladros inferiores y a la pletina lateral con el taladro superior. Está formada por dos rectángulos superpuestos por las caras.

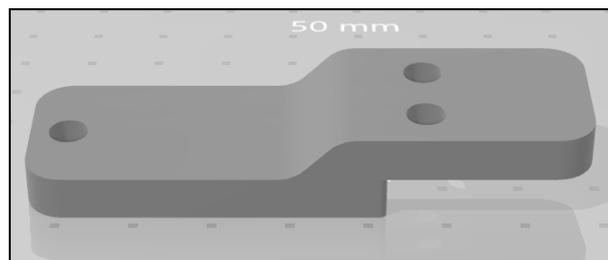


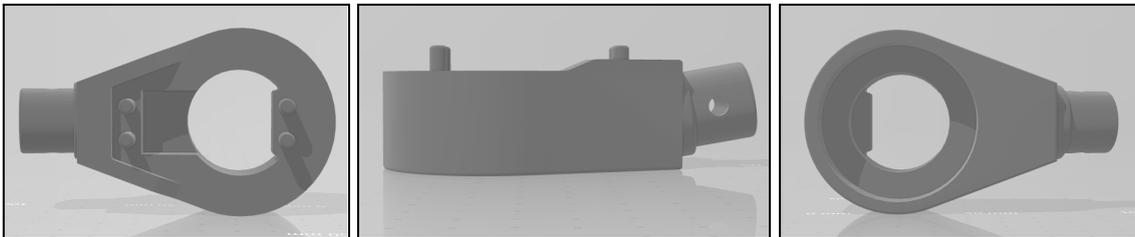
Figura 34. Enlace Suspensión-Construcción.

Mide 73 mm de largo y 22 mm de ancho, con un espesor de pared de 6 mm, siendo 12 mm en el tramo en el que se superponen ambos planos.

3.3.5. ACOPLAMIENTO RUEDA DELANTERA

Esta pieza se ubica encima del arco de la rueda, y se une a ella mediante un rodamiento que se ensambla en su interior. Además, también es la pieza encargada de alojar los servomotores, mediante los cuatro insertos que tiene en su parte superior.

Asimismo, también tiene un saliente cilíndrico, con cierta angulación, que se introduce en el tubo de PVC más largo de los tres que hay en el sistema de suspensión.



Figuras 35(a) , 35(b) y 35(c). Acoplamiento Rueda Delantera.

Mide 33 mm de alto, 100 mm largo y 58 mm de ancho. Los salientes que se insertan en el servomotor tienen un diámetro de 7 mm, y el que se inserta en el tubo de PVC tiene un diámetro de 21 mm y una angulación de 17° respecto a la horizontal. A su vez, el diámetro exterior del alojamiento del rodamiento es de 50 mm.

3.3.6. ACOPLAMIENTO RUEDA TRASERA

Su función es exactamente la misma que la del acoplamiento de la rueda delantera, con la diferencia de que el tubo de PVC al que se une es al mediano, y no habiendo en el saliente la angulación que sí había en el anterior.

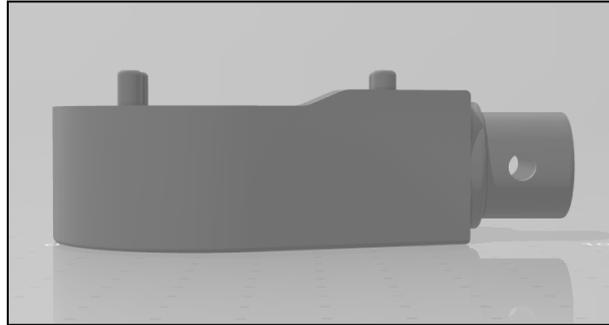


Figura 36. Acoplamiento Rueda Trasera.

Las medidas también son las mismas que en el acoplamiento de la rueda delantera, exceptuando la angulación del saliente, que en este caso es completamente paralelo a la horizontal.

3.3.7. ALOJAMIENTO RODAMIENTO TRASERO

Su función es la de alojar el rodamiento que actuará de bisagra entre el *rocker* y el *bogie*, haciendo que el sistema de suspensión pueda autoequilibrarse gracias a la gravedad.

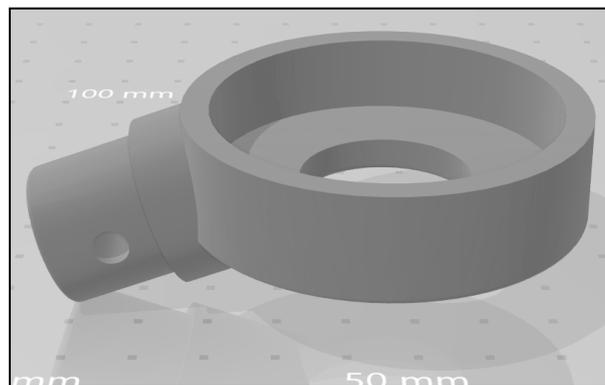


Figura 37. Alojamiento Rodamiento Trasero.

Mide 37 mm de alto, 80 mm largo y 58 mm de ancho, con un espesor de pared de mm. El diámetro del alojamiento es de 50 mm, y cuenta con un saliente de 22 mm de diámetro, con una angulación de 17°.

3.3.8. ACOPLAMIENTO RODAMIENTO TRASERO

El acoplamiento del rodamiento trasero es la pieza que une el *rocker* y el *bogie*, concretamente uniendo el tubo de PVC mediano, mediante cuatro taladros en sus laterales, con el alojamiento del rodamiento, insertándose en el mismo gracias al saliente cilíndrico que tiene.

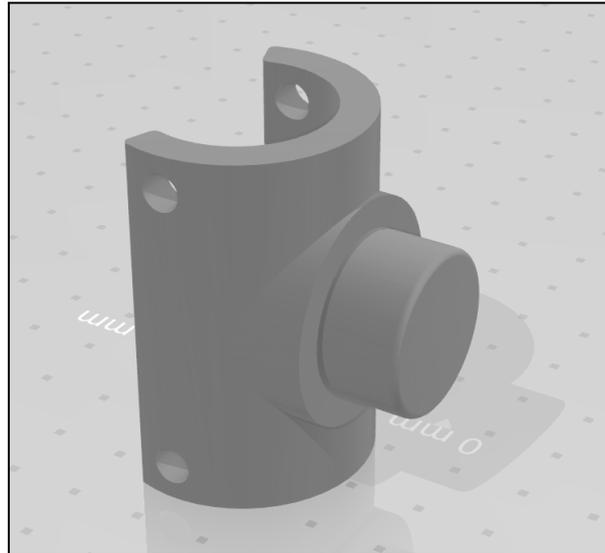


Figura 38. Acoplamiento Rodamiento Trasero.

Mide 50 mm de alto, 35 mm largo y 35 mm de ancho, con un espesor de pared de 5 mm, por lo que el diámetro interior de la cavidad es de 25 mm para acoplarse en el exterior del tubo de PVC, al que se une mediante los cuatro taladros que podemos observar. Incluye un saliente con un diámetro de 22 mm, para ser insertado en uno de los rodamientos.

3.4. RUEDAS Y ARCOS

En esta ocasión, tenemos seis ensamblajes iguales, que consisten en unos arcos que giran respecto al sistema de suspensiones por la acción de seis servomotores, para orientar las ruedas en la dirección deseada.

Estas ruedas están a su vez unidas a seis motores de corriente continua, que serán los que doten de movimiento al vehículo.

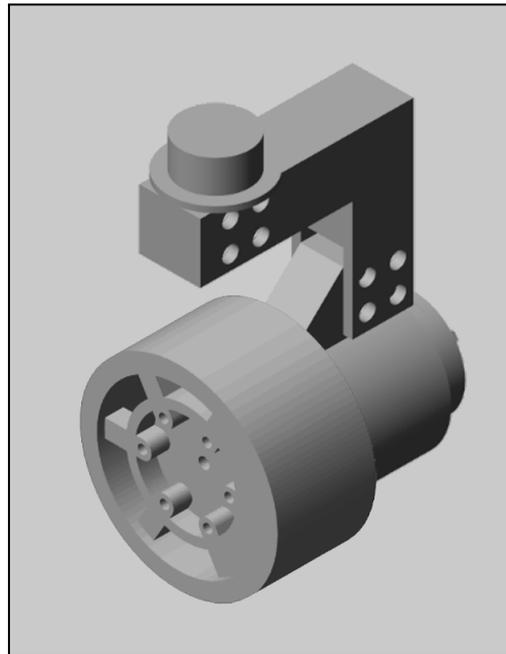


Figura 39. Rueda y Arco.

Este conjunto está formado por las siguientes piezas:

Pieza	Cantidad
Alojamiento Motor	6
Motor	6
Arco Estructural	6
Acoplamiento Rodamiento Servo	6
Rueda	6
Acople Interior Rueda	6

Tabla 5. Listado de Piezas de Ruedas y Arco.

A continuación, se detallarán las características constructivas de cada una de ellas.

3.4.1. ALOJAMIENTO MOTOR

En esta cavidad se introduce el motor de corriente continua que propulsará el vehículo, quedando unido a ésta mediante dos tornillos que se introducen en las dos ranuras que podemos observar en forma de coliso.

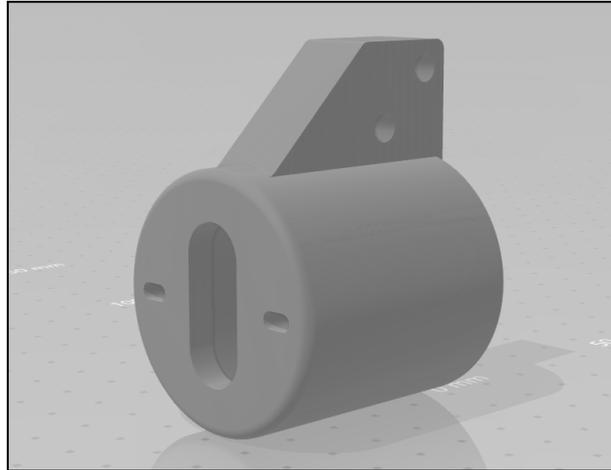


Figura 40. Alojamiento Motor.

Así, también se une al arco estructural mediante los dos taladros que tiene en el saliente de la parte superior.

Mide 45 mm de alto, 65 mm largo y 49 mm de ancho, con un espesor de pared de 6 mm, por lo que el diámetro interior de la cavidad es de 37 mm, perfecto para alojar el motor de corriente continua en su interior.

3.4.2. MOTOR

En este apartado, únicamente hablaremos de las características geométricas del motor y de cómo se relaciona con el resto de piezas, como ya hemos hecho con los servomotores.

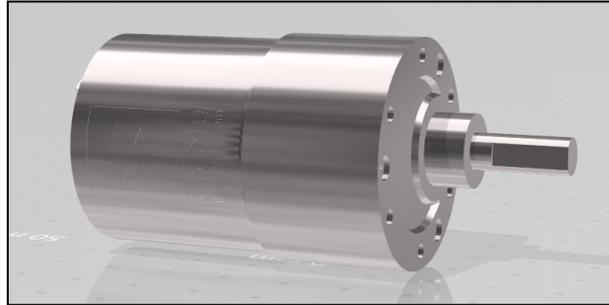


Figura 41. Motor.

Como acabamos de ver, el motor se introduce dentro del alojamiento del motor, unido a este por los dos taladros anteriormente mencionados, y uniéndose también a la rueda mediante la introducción del eje en el acople interior de ésta.

Tiene una longitud de 80 mm, incluyendo el eje, y un diámetro exterior de 37 mm.

3.4.3. ARCO ESTRUCTURAL

Esta pieza tiene la misión de transmitir el giro del servomotor a la rueda, uniendo el acoplamiento del rodamiento del servo y el alojamiento del motor, ambos gracias a los taladros que se pueden ver en la siguiente figura:

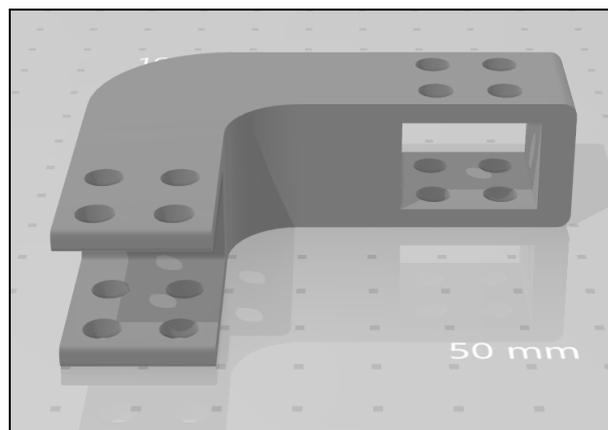


Figura 42. Arco Estructural.

Mide 20 mm de alto, 70 mm largo y 60 mm de ancho, con un espesor de pared de 6 mm.

3.4.4. ACOPLAMIENTO RODAMIENTO SERVO

Esta pieza es un inserto que se introduce en el rodamiento ubicado dentro del alojamiento de la rueda, y a su vez, se une mediante los taladros al recién descrito arco estructural. Su misión es traducir el movimiento del servomotor en el giro de la rueda.

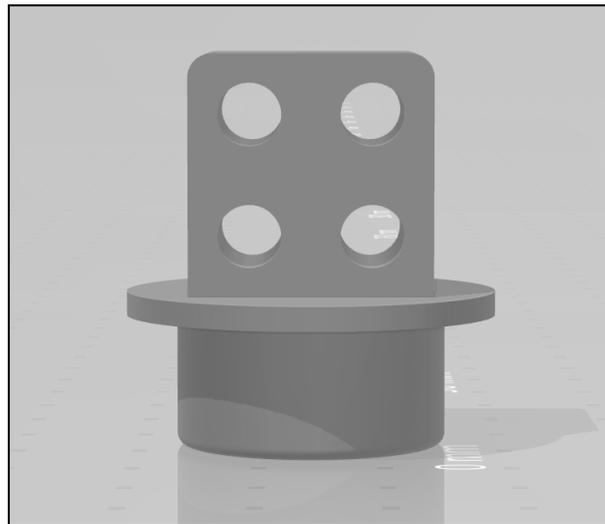


Figura 43. Acoplamiento Rodamiento Servo.

Mide 33 mm de alto, y en cuanto al largo y al ancho máximo viene determinado por el cilindro central, de 30 mm de diámetro. Asimismo, el diámetro del saliente cilíndrico de la parte inferior es de 22 mm, ya que va insertado en un rodamiento.

3.4.4. RUEDA

Las ruedas, como es evidente, son las piezas encargadas de dirigir y mover el vehículo. Éstas se unirán a los motores mediante los acoples interiores, y además, por su geometría, albergarán a los alojamientos de los motores en su interior.

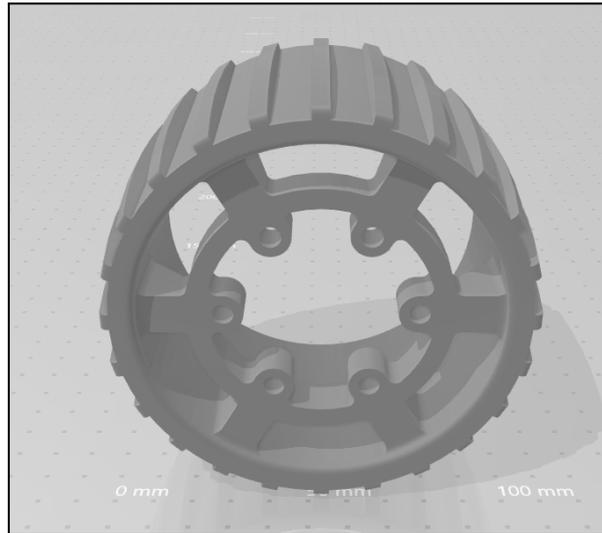


Figura 44. Rueda.

A su misma vez, estas ruedas tienen salientes en dirección axial en su zona perimetral, para asegurar una mejor adherencia con el terreno y por lo tanto tener una mayor tracción y ser más eficiente y estable.

Cuentan con un diámetro exterior de 107 mm, y tienen un ancho de 64 mm.

3.4.5. ACOPLER INTERIOR RUEDA

Estas piezas circulares se atornillan a las ruedas mediante los seis taladros exteriores, y se unirán a los motores alojando su eje en el taladro central, es decir, son el intermediario en la transmisión del movimiento entre el motor y la rueda.

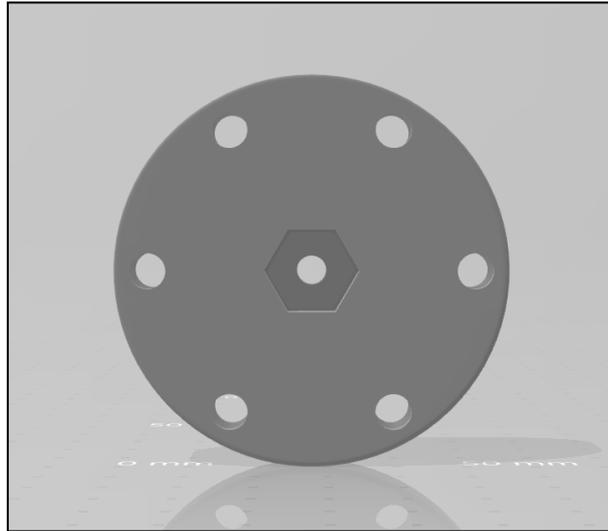


Figura 45. Acople Interior Rueda.

En su interior podemos observar un rebaje en forma hexagonal, dentro del cual se introducirán unos insertos que aportarán fiabilidad a la unión con el eje del motor mencionada anteriormente.

Su diámetro es de 57 mm, y su espesor de 10 mm.

4. SIMULACIÓN: SIMSCAPE MULTIBODY

En este apartado se definirán todos los pasos realizados en materia de simulación de comportamiento del vehículo, pasando por la simplificación de las piezas, su ensamblaje, el modelado del entorno, el modelado de las relaciones entorno-vehículo y el diseño de trayectorias y control, hasta llegar al análisis de resultados.

El software escogido para la simulación es Simulink, el entorno de programación visual de MATLAB, de la compañía Mathworks. Aunque más concretamente, la mayoría del trabajo estará realizado empleando la librería Simscape Multibody.

Muchos de los términos específicos referentes al software y a la librería pueden ser desconocidos para el lector, y serán definidos y explicados en los subapartados a medida que vayan a aparecer y a ser necesarios.

4.1. PIEZAS PARA SIMULACIÓN

En ocasiones, por muy fiel que trate ser el entorno virtual al mundo físico, nos encontramos con limitaciones o requerimientos especiales que deben ser tenidos en cuenta, es por ello que en este apartado se explicará cómo y por qué se han modificado las piezas para mejorar la experiencia en la simulación.

4.1.1. SIMPLIFICACIÓN DE LA GEOMETRÍA

Uno de los parámetros más importantes para realizar la simulación es la cantidad de geometría de las piezas, dependiendo de ésta en gran parte los recursos que consume el ordenador, y por lo tanto, el tiempo que tarda cada vez que deseamos simular.

El objeto de este trabajo es un mecanismo con muchas piezas con alto nivel de detalle y muchas interacciones entre ellas y el entorno, por lo que si tratamos de simular empleando los componentes finales que serán impresos, probablemente el

tiempo que necesitemos cada vez que queramos hacer una modificación o iteración será elevado.

Además, una de las materias que predominan en el proyecto es el diseño de los bucles de control, lo cual hace que vayan a ser necesarias numerosas pruebas hasta alcanzar el resultado deseado.

Por todo esto, se ha optado por rediseñar las piezas originales exclusivamente para la simulación, pero siempre siguiendo dos premisas:

- Se debe eliminar todo detalle innecesario geoméricamente que no aporte en la funcionalidad del diseño, incluyendo todos los redondeos de aristas, por generar un número elevado de facetas, a pesar de no aportar al funcionamiento.
- La geometría principal de las piezas debe respetarse en la medida de lo posible, para conseguir la mayor fidelidad posible con el prototipo real, especialmente las partes de las piezas que intervienen en la unión con otras, que deben ser exactamente iguales.

A continuación, se muestra un ejemplo de comparativa entre el componente que formará parte del prototipo físico y el que se emplea en la simulación, en concreto la rueda.

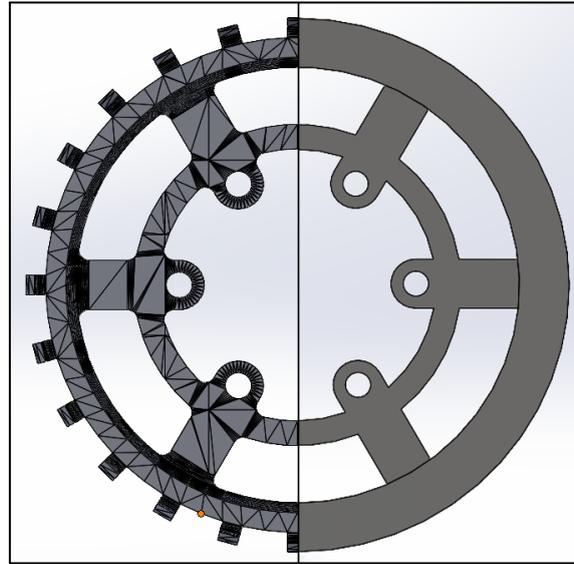


Figura 46. Comparativa Rueda Prototipo vs Rueda Simulación.

En la imagen se puede observar como la rueda que se incluirá en el diseño final tiene en su parte externa unos salientes que mejorarán la tracción, y en toda su constitución tiene redondeos para hacer más resistente la pieza y evitar aristas vivas. En cambio, la rueda que se utilizará en la simulación es un cilindro simple y las uniones entre piezas son aristas definidas.

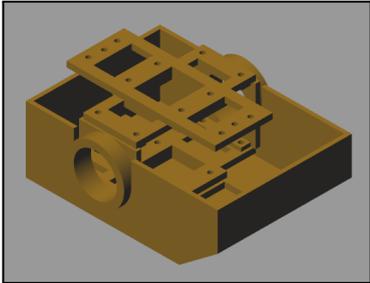
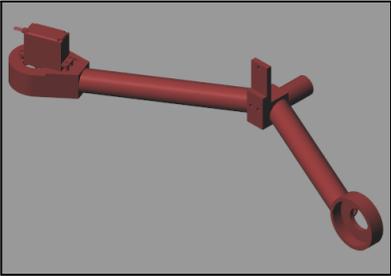
Así mismo, si nos fijamos bien en el componente rediseñado, podemos advertir que también se cumple la segunda premisa, ya que se han respetado en buena medida tanto la cantidad de material, como el diámetro exterior de la rueda y el diámetro y la disposición de los taladros que servirán de unión con otro componente. Gracias a esto conseguimos que el peso en la simulación sea similar y que el diseño del control en Simulink sea adecuado para el tamaño de rueda final.

4.1.2. PREPARACIÓN DE LOS ARCHIVOS PARA SIMSCAPE MULTIBODY

Por último, también es importante modificar el formato de los archivos, guardando los que vayan a ser empleados en la simulación como STEP AP214 en lugar de STL, ya que al relacionar piezas necesitaremos poder seleccionar circunferencias y aristas como geometrías de referencia.

Además, para poder realizar el ensamblaje correctamente en Simscape Multibody, desde el programa de diseño 3D no debemos exportar los componentes por separado, sino que lo debemos hacer por subconjuntos, exportando como archivos únicos las agrupaciones de piezas que tengan uniones rígidas entre sí, es decir, que no tengan movimiento relativo entre ellas.

En la siguiente tabla se muestra la equivalencia entre las piezas únicas y los subconjuntos en formato STEP AP214 exportados para ser utilizados en la simulación, junto con una imagen de las mismas:

SUBCONJUNTOS RÍGIDOS	PIEZAS ÚNICAS	IMAGEN
Chasis Completo	<ul style="list-style-type: none"> ● Chasis ● 2x Elemento de Construcción ● 2x Elemento de Construcción 2 ● 2x Sujeción Rodamiento Principal 	
Suspensión Central Izquierda	<ul style="list-style-type: none"> ● Sujeción Suspensión ● PVC 180x25x21 ● PVC 130x25x21 ● Servomotor ● Enlace Suspensión-Construcción ● Acoplamiento Rueda Delantera ● Alojamiento Rodamiento Trasero 	

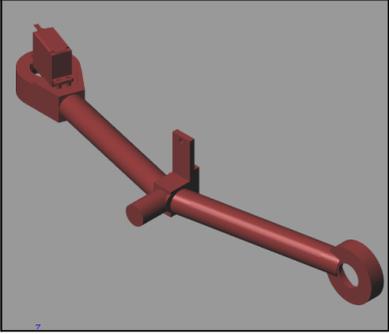
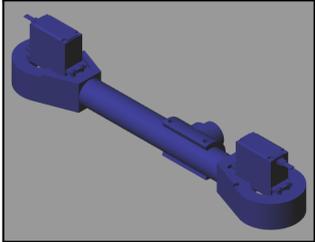
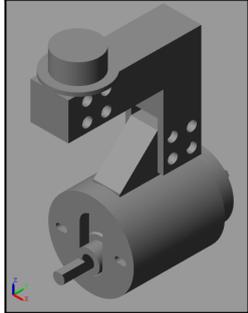
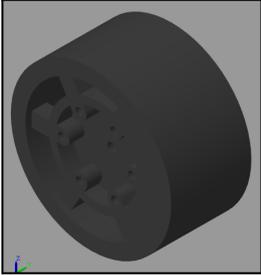
<p>Suspensión Central Derecha</p>	<ul style="list-style-type: none"> • Sujeción Suspensión • PVC 180x25x21 • PVC 130x25x21 • Servomotor • Enlace Suspensión-Construcción • Acoplamiento Rueda Delantera • Alojamiento Rodamiento Trasero 	
<p>2x Eje Trasero</p>	<ul style="list-style-type: none"> • PVC 150x25x21 • 2x Acoplamiento Rueda Trasera • 2x Servomotor • Acoplamiento Rodamiento Trasero 	
<p>6x Arco</p>	<ul style="list-style-type: none"> • Alojamiento Motor • Motor • Arco Estructural • Acoplamiento Rodamiento Servo 	
<p>6 x Rueda</p>	<ul style="list-style-type: none"> • Rueda • Acople Interior Rueda 	

Tabla 6. Relación de Subconjuntos Para Simulación.

4.2. ENSAMBLAJE DEL VEHÍCULO EN SIMSCAPE MULTIBODY

En este apartado se explicará cómo se importan los archivos CAD a la simulación mediante Simscape Multibody, cómo se establecen las relaciones entre ellos y cómo se definen los coeficientes de rozamiento que guían el comportamiento de las articulaciones.

Para ello, en primer lugar introduciremos ciertos conceptos, bloques y funcionalidades del programa.

- **FILE SOLID:** Es el bloque de la librería Simscape Multibody que utilizaremos para importar nuestros archivos CAD a Simulink.

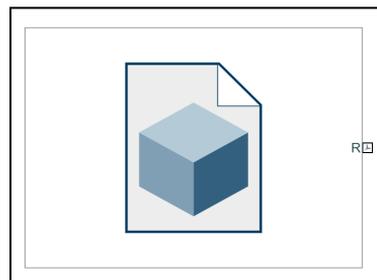


Figura 47. Bloque File Solid.

Permite que importemos los ficheros STEP mencionados en el punto anterior y nos da la opción de modificar la apariencia (color) de la pieza, asignarle una densidad y crear sistemas de referencia, que serán los que posibilitarán la existencia de relaciones físicas entre dicha pieza y otras o el entorno.

- **REVOLUTE JOINT:** Este bloque nos permite establecer relaciones de rotación entre piezas, formando una bisagra.

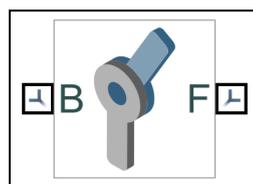


Figura 48. Bloque Revolute Joint.

Además, existe la opción de que la junta sea un accionamiento, por lo que es también la relación que se establece entre un motor y el elemento sobre el que actúa, pudiendo incluso añadir sensorización al bloque para medir la velocidad angular.

- **MODEL PROPERTIES:** En ocasiones, debemos asignar un valor numérico a parámetros de la simulación. El menú de model properties nos proporciona la opción de declarar y almacenar variables, de modo que podemos introducir las variables en lugar de los valores en cada uno de los elementos de la simulación.

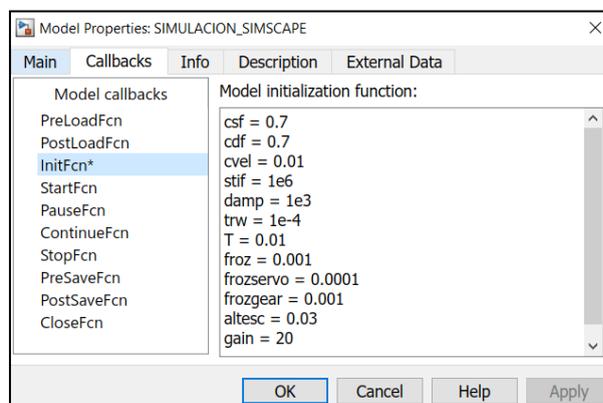


Figura 49. Model Properties.

Esto es especialmente útil cuando debemos iterar o probar valores hasta que la simulación tenga un funcionamiento correcto, pudiendo modificar el valor de la variable en lugar de tener que cambiar los parámetros dentro de los elementos afectados, ahorrando así mucho tiempo.

- **SUBSYSTEMS Y CONNECTION PORTS:** En ensamblajes complejos, con muchas piezas y muchos niveles, la creación de subsistemas es crucial. Estos no son más que agrupaciones de bloques y sus relaciones, representados por un único bloque que los engloba, para facilitar el orden del espacio de trabajo.

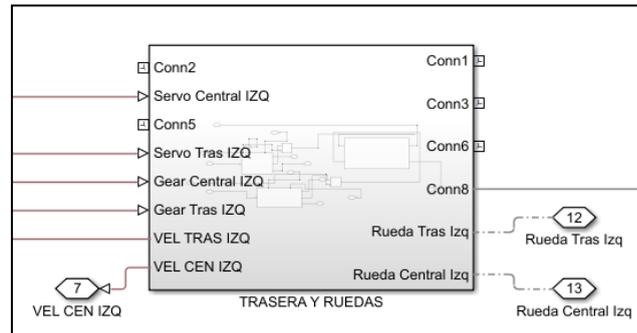


Figura 50. Subsistema y Puertos de Conexión.

A las salidas del subsistema que se pueden apreciar en la figura se les conecta un puerto de conexión, que permitirá que en el nivel inmediatamente superior al subsistema en cuestión podamos conectar señales a las entradas a las referencian dichos puertos.

4.2.1. RELACIONES ENTRE PIEZAS

Por las características constructivas del vehículo, las relaciones entre articulaciones son todas de concentricidad, es decir, el movimiento relativo de unas piezas respecto a otras es únicamente de rotación en uno de los tres ejes, como en una bisagra.

Cabe destacar que las relaciones se establecen superponiendo dos sistemas de referencia, uno de ellos de cada una de las piezas que se desean unir, por lo que hay que crear dichos sistemas orientados de forma que las direcciones y los sentidos de los ejes coincidan según queramos que se ensamblen las piezas, teniendo en cuenta que el eje de rotación debe ser siempre el eje cota o eje "Z".

A pesar de ser todas en principio del mismo tipo, las podemos dividir en simples y accionadas, y dentro de estas últimas, distinguimos entre las que tienen entrada de posición o de par.

En la tabla se detallarán todas las relaciones existentes en el conjunto:

UNIÓN	SUBCONJUNTOS	TIPO DE UNIÓN
-------	--------------	---------------

Lateral Izquierdo	<ul style="list-style-type: none"> ● Chasis Completo ● Suspensión Central Izquierda 	Revolute Joint (Simple)
Lateral Derecho	<ul style="list-style-type: none"> ● Chasis Completo ● Suspensión Central Derecha 	Revolute Joint (Simple)
Permiten que los laterales completos del vehículo oscilen respecto al chasis por acción de la gravedad, aumentando su capacidad de sortear obstáculos y mejorando su estabilidad.		
Trasera Izquierda	<ul style="list-style-type: none"> ● Suspensión Central Izquierda ● Eje Trasero 	Revolute Joint (Simple)
Trasera Derecha	<ul style="list-style-type: none"> ● Suspensión Central Derecha ● Eje Trasero 	Revolute Joint (Simple)
Son uniones similares a las dos anteriores, en este caso permitiendo que cada par de ruedas traseras oscile, de nuevo por acción de la gravedad, para añadir aún más estabilidad y capacidad de tracción al vehículo.		
4x Servos Traseros	<ul style="list-style-type: none"> ● Eje Trasero ● Arco 	Revolute Joint (Entrada de Posición)
Servo Delantero Izquierdo	<ul style="list-style-type: none"> ● Suspensión Central Izquierda ● Arco 	Revolute Joint (Entrada de Posición)
Servo Delantero Derecho	<ul style="list-style-type: none"> ● Suspensión Central Derecha ● Arco 	Revolute Joint (Entrada de Posición)
Los servomotores hacen girar a los arcos, y por lo tanto a las ruedas, controlando la dirección del vehículo.		
6x Rueda	<ul style="list-style-type: none"> ● Arco ● Rueda 	Revolute Joint (Entrada de Par y Salida de Velocidad)
Los motores hacen girar a las ruedas mediante una entrada de par. Además, tenemos la lectura de la velocidad, que la emplearemos en el control.		

Tabla 7. Relaciones de Piezas en Ensamblaje.

Tras el establecimiento de todas las uniones mencionadas, tenemos nuestro ensamblaje completamente formado y con un comportamiento físico a priori lógico.

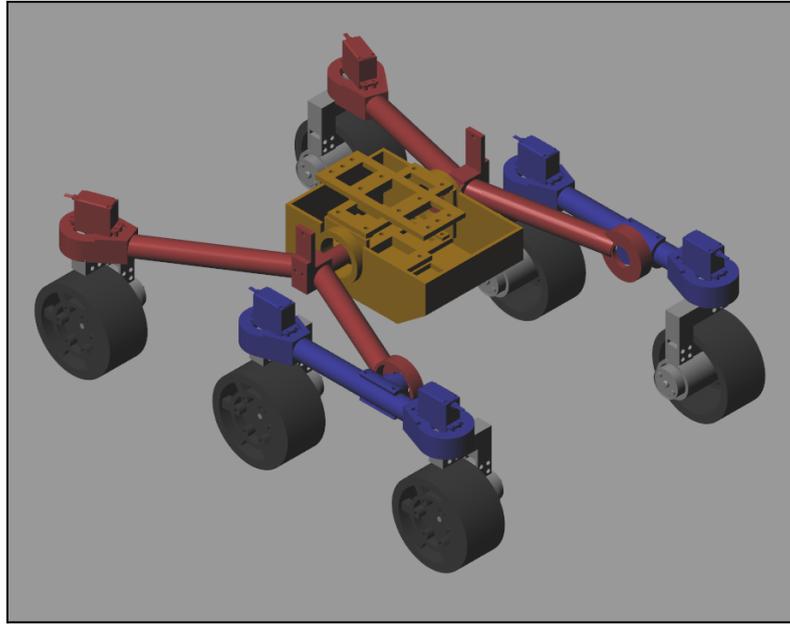


Figura 51. Ensamblaje en Simscape Multibody.

4.2.2. ESTABLECIMIENTO DE COEFICIENTES DE ROZAMIENTO

Para dotar de realismo y sentido a la simulación, es necesario que exista fricción entre las piezas, ya que de lo contrario, tendríamos situaciones inverosímiles, tales como que, por ejemplo, una rueda que haya sido impulsada en algún momento por un motor, siga girando indefinidamente aunque ya no esté recibiendo dicho impulso, por no existir una fuerza que la frene.

Como ya se ha visto anteriormente, los coeficientes de rozamiento se declaran como variables en el menú "model properties". A pesar de que exista fricción en todas las uniones y que en el mundo real cada una sea distinta, en la simulación las englobamos en los siguientes grupos:

- **Rozamiento del Rodamiento:** Es el rozamiento existente en las uniones no accionadas, en las que ambas piezas se unen únicamente por medio de un

rodamiento, y el movimiento es consecuencia de fuerzas externas. El valor final asignado es de $0,001 \text{ N}\cdot\text{m}/(\text{deg}/\text{s})$.

- **Rozamiento del Servo:** Es el rozamiento que existe dentro del propio servo, haciendo que frene una vez se corta la señal. El valor final asignado es de $0,0001 \text{ N}\cdot\text{m}/(\text{deg}/\text{s})$.
- **Rozamiento del Motor:** Es el rozamiento interno del eje del motor. Sin esta fricción, el movimiento una vez originado por el motor tendría un efecto ininterrumpido, haciendo que la rueda girase hasta que una fuerza externa la detuviera. El valor final asignado es de $0,001 \text{ N}\cdot\text{m}/(\text{deg}/\text{s})$.

La decisión de los valores finales se ha tomado mediante iteraciones en la simulación, suspendiendo el vehículo en el aire, fijado por el centro de gravedad del mismo, y aplicándole fuerzas externas que moviesen las articulaciones, con el fin de encontrar la combinación de parámetros que hiciesen que las reacciones a dichas fuerzas fuesen proporcionadas, lógicas y realistas.

4.3. DISEÑO DEL ENTORNO

Como en toda simulación física, necesitamos un entorno en el que nuestro mecanismo va a ser sometido a pruebas y mediciones, con el fin de mejorar su desarrollo.

En este caso, por ser un vehículo con una finalidad tan concreta, el escenario va a ser lo más parecido posible al terreno marciano, por lo que debemos crear un terreno irregular en el que “entrenar” a nuestro Rover.

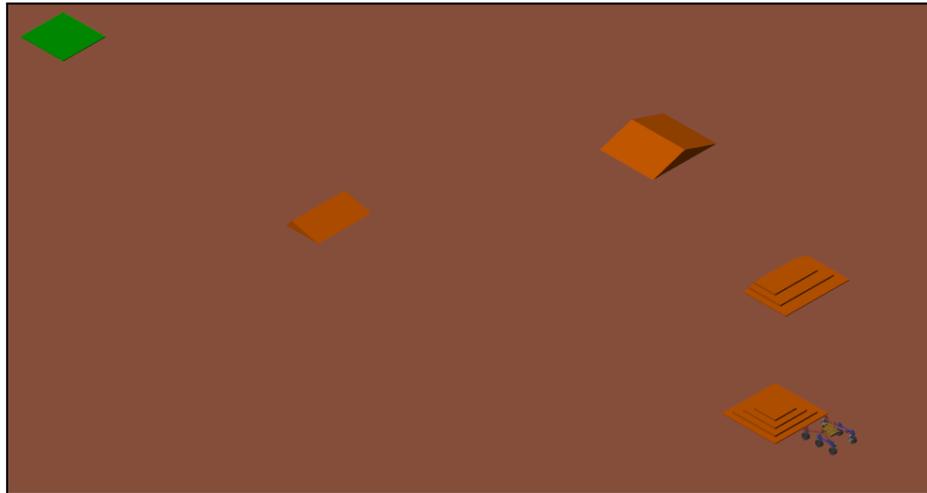
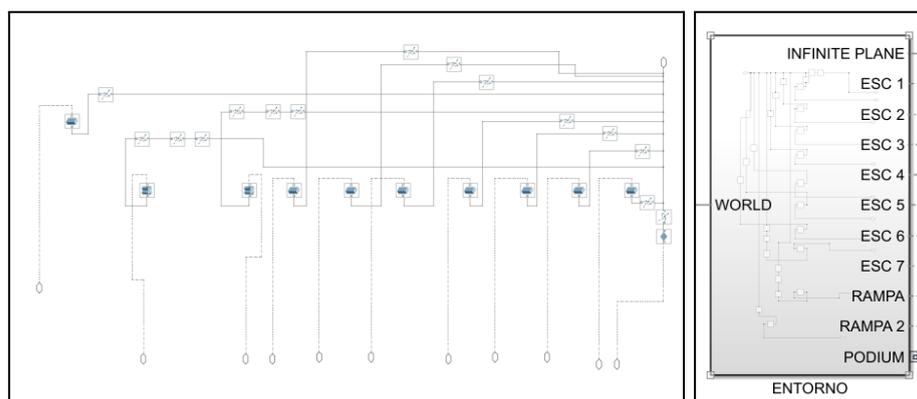


Figura 52. Entorno de Simulación en Simscape Multibody.

4.3.1. CREACIÓN DE OBSTÁCULOS

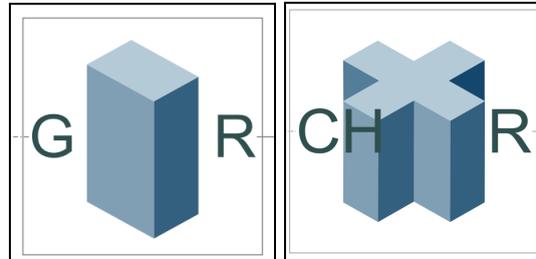
En este capítulo se describirá brevemente la creación de sólidos que servirán de obstáculos en la trayectoria del vehículo en la simulación, con el fin de evaluar cómo de estable es el mismo en un terreno hostil e irregular.

Además, este añadido nos ayudará a diseñar un control de mayor calidad, observando cómo rinde el mecanismo en situaciones demandantes como rampas u obstáculos en los que hay varias ruedas en el aire.



Figuras 53(a) y 53(b). Creación de Obstáculos en Simscape Multibody y Subsistema "Entorno".

Los obstáculos serán escalones y rampas, creados con los bloques de brick solid y extruded solid de Simscape respectivamente, y a continuación se describirá su geometría y su función.



Figuras 54(a) y 54(b). Bloques "Brick Solid" y "Extruded Solid".

4.3.1.1. ESCALONES FRONTALES

El primer obstáculo al que se enfrentará nuestro Rover será un grupo de escalones con una forma similar a una pirámide maya, el cual tendrá que superar en desplazamiento completamente frontal y con todas sus ruedas a la vez.

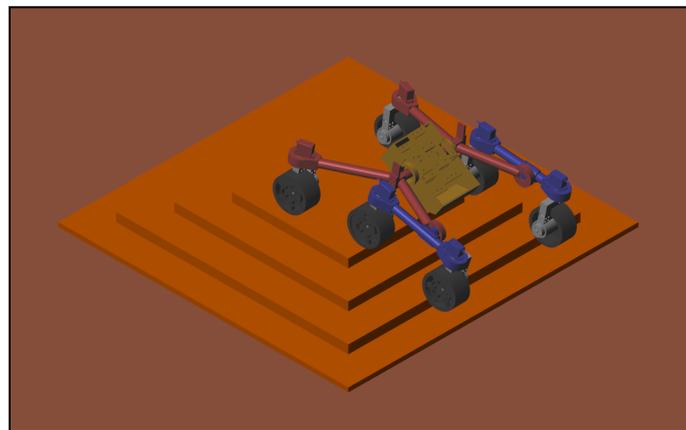


Figura 55. Vehículo Superando Escalones Frontales.

Serán cuatro escalones concéntricos con forma de cuadrado, midiendo lado del mayor 1 m y el lado del menor 40 cm, y teniendo cada uno una altura de 3 cm.

La mayor dificultad del obstáculo no reside únicamente en la altura total de 12 cm que tiene, sino que la mayor traba que tendrá el vehículo será el cambio de altura al que se enfrentará cada 10 cm, haciendo que las ruedas tengan que traccionar

varias veces seguidas en poco espacio y en ocasiones sin estar todas en contacto completo simultáneo con el suelo.

4.3.1.2. ESCALONES CON TRES RUEDAS

El segundo obstáculo que se encontrará será muy similar al primero en forma y tamaño, con la diferencia de que en esta ocasión serán tres en vez de cuatro escalones, con la misma altura que los anteriores, y cortados por la mitad.

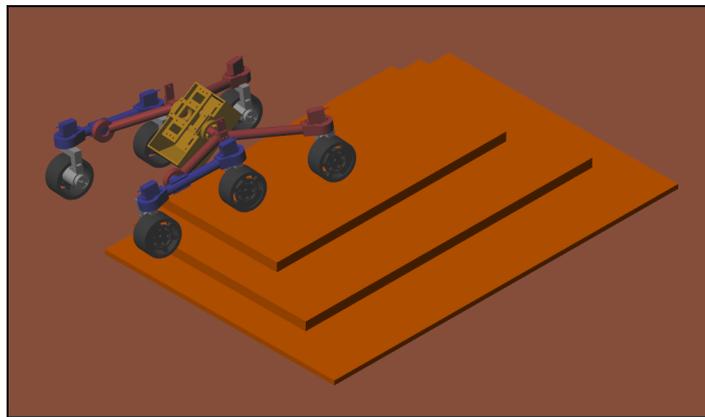


Figura 56. Vehículo Superando Escalones Laterales.

Este obstáculo es el más crítico, ya que el Rover lo debe superar únicamente con tres ruedas, llegando a haber un desnivel de 9 cm entre un tren de ruedas y el otro.

Aquí se pone a prueba de verdad la estabilidad del vehículo, y su capacidad para traccionar en situaciones difíciles, así como la capacidad de las articulaciones que hacen de bisagra en las suspensiones para autoequilibrar las ruedas por acción de la gravedad.

4.3.1.3. RAMPA LATERAL

En este obstáculo se pone a prueba uno de los movimientos característicos de este vehículo, que no suele poder darse en el resto de automóviles, siendo éste el desplazamiento lateral.

La rampa tendrá una altura de 15 cm y una inclinación a ambos lados de 20°.

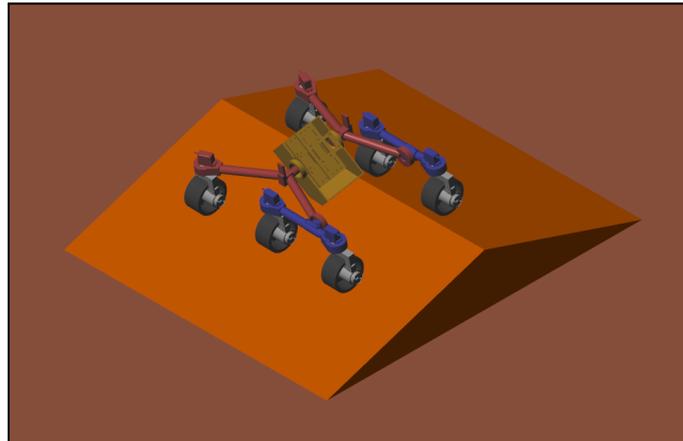


Figura 57. Vehículo Superando Rampa Lateral.

Como ya hemos mencionado anteriormente, el Rover puede girar todas sus ruedas 90° para orientarlas perpendiculares a la dirección frontal y desplazarse de forma completamente lateral, tanto a derecha como a izquierda. En este caso, queremos ver la eficacia con la que supera obstáculos y mantiene la estabilidad en esta posición.

4.3.1.4. RAMPA FRONTAL

El último obstáculo es una rampa a priori bastante simple, pero en este caso tiene dos dificultades añadidas.

La primera de ellas es que se realizará marcha atrás, lo cual no debería ser un problema por la geometría constructiva del vehículo, pero también debe ser probado.

Y la segunda viene dada por el excesivo desnivel en la bajada, ya que podría hacer volcar al Rover, pero se debe simular por la probabilidad de la existencia de salientes abruptos en el terreno del planeta rojo.

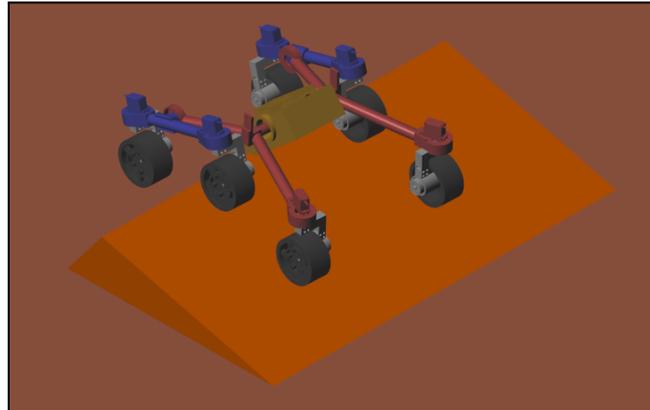


Figura 58. Vehículo Superando Rampa Frontal.

La rampa tiene una altura de 10 cm, con una inclinación de 15° en la subida y 60° en la bajada.

4.4. RELACIÓN ENTRE VEHÍCULO Y ENTORNO

En este apartado se definirá la relación de nuestro mecanismo con el entorno creado en Simscape Multibody, el cual ha sido explicado en el capítulo anterior.

Esta relación determina cómo van a interactuar vehículo y entorno, lo cual limitará los movimientos que el primero puede realizar respecto al segundo.

El capítulo en cuestión engloba:

- La definición de grados de libertad, que serán cruciales para dotar de movimiento al mecanismo.
- La posición inicial del vehículo respecto del centro del entorno o "mundo".
- La creación de fuerzas de contacto entre el vehículo y el entorno, para que la interacción entre ellos sea fidedigna al comportamiento real.
- La sensorización añadida al vehículo para después poder medir resultados y graficarlos, además de ser necesaria para el diseño del control de velocidad.

4.4.1. DEFINICIÓN DE GRADOS DE LIBERTAD CON EL MUNDO

Para definir cómo se puede mover el vehículo respecto al entorno, es necesario dotarle de grados de libertad, ya que por defecto, la unión es rígida.

En este caso, al ser un vehículo que no tiene ningún tipo de restricción y puede desplazarse en todos los sentidos, debemos asignarle los seis grados de libertad posibles, siendo estos:

- Rotación respecto al eje X
- Rotación respecto al eje Y
- Rotación respecto al eje Z
- Traslación respecto al eje X
- Traslación respecto al eje Y
- Traslación respecto al eje Z

En Simscape Multibody existe un bloque llamado "6-DOF Joint", el cual dota de los seis grados de libertad a la unión, pero existe un problema, y es que a la hora de leer mediante sensorización los datos de rotación, los extrae en formato de cuaterniones, con parte real e imaginaria, lo que dificultaría futuramente el diseño del control, debiendo añadir una estructura de bloques a modo de conversión.

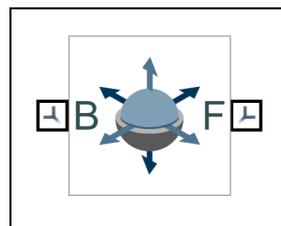


Figura 59. 6-DOF Joint.

Por lo tanto, la solución pasa por emplear una combinación de bloques que aporten todos los grados de libertad sumándolos entre ellos, por lo que se emplearán los bloques "Cartesian Joint" y "Gimbal Joint".

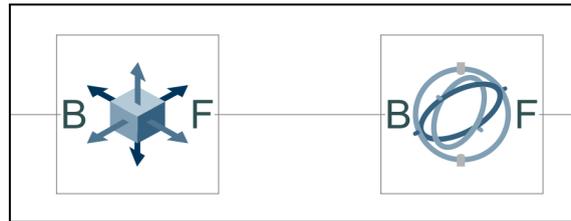


Figura 60. *Cartesian Joint* y *Gimbal Joint*.

Como se puede deducir de la figura, el bloque de “*Cartesian Joint*” nos permitirá habilitar la traslación respecto a los tres ejes, mientras que el bloque “*Gimbal Joint*” nos permitirá hacer lo mismo con la rotación en este caso.

4.4.2. DEFINICIÓN DE LA POSICIÓN INICIAL DEL VEHÍCULO

Al introducir el vehículo en la simulación, si no especificamos dónde queremos que se sitúe, lo hará con su centro de masas en la misma ubicación que el centro de masas del “*Infinite Plane*” que representa el suelo.

Si añadimos propiedades físicas a la simulación y el vehículo y el suelo se intersectan, el vehículo quedará atrapado dentro de éste, sin poder liberarse.

Para evitar esta situación, la solución es levantar el vehículo respecto del suelo en el instante inicial con el bloque “*Rigid Transform*”, el cual aplica una matriz de transformación respecto del centro de masas del objeto al cual lo enlacemos.

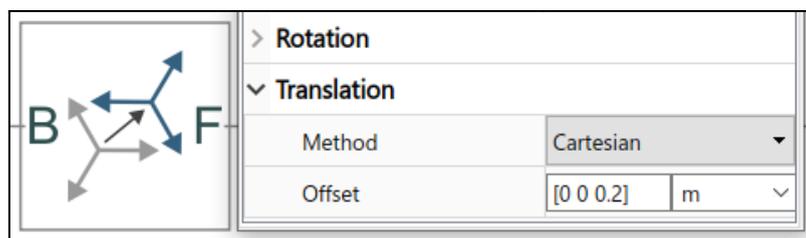


Figura 61. *Rigid Transform*.

En este caso, la matriz de rotación está deshabilitada, y en la matriz de traslación podemos observar que únicamente se le ha dado valor al eje Z o vertical, por lo que, en el instante inicial, el centro de masas del vehículo está 20 cm por encima del suelo, cayendo a continuación por la fuerza de la gravedad.

4.4.3. CREACIÓN DE FUERZAS DE CONTACTO

Siguiendo con la explicación anterior, en el momento en el que "soltamos" el vehículo y cae sobre el entorno, debe haber una serie de fuerzas que definan cómo van a apoyarse las ruedas sobre el suelo, en lugar de atravesarlo.

Estas fuerzas serán creadas mediante el bloque "Spatial Contact Force" de la librería de Simscape Multibody, el cual podemos observar en la siguiente figura:

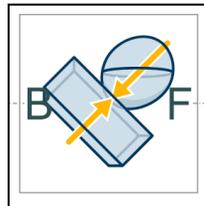


Figura 62. Spatial Contact Force.

Este método de asignación de propiedades físicas nos permite modelar la interacción entre elementos individualmente, lo cual puede ser una ventaja en muchos mecanismos, pero en nuestro caso resulta un gran inconveniente, ya que debemos crear una fuerza de contacto entre cada una de las ruedas y cada uno de los elementos que vayan a interactuar con las mismas.



Figura 63. Subsistema "Contact Forces".

En total, teniendo en cuenta que tenemos seis ruedas en el vehículo, y que el entorno está formado por el suelo, dos rampas y siete escalones, el número final de

fuerzas de contacto que debemos crear será de sesenta, resultando una estructura intrincada que deberá representar un subsistema en sí misma.

4.4.3.1. DEFINICIÓN DE COEFICIENTES

Para regular la relación física mencionada anteriormente, debemos establecer una serie de parámetros dentro de los bloques de las fuerzas de contacto.

En este caso, al ser todas las ruedas iguales y todos los elementos del entorno del mismo material, se ha optado por establecer los mismos coeficientes en todas las fuerzas de contacto, por lo que la interacción de cada una de las ruedas con cada uno de los elementos del entorno estarán regidas por los mismos parámetros físicos.

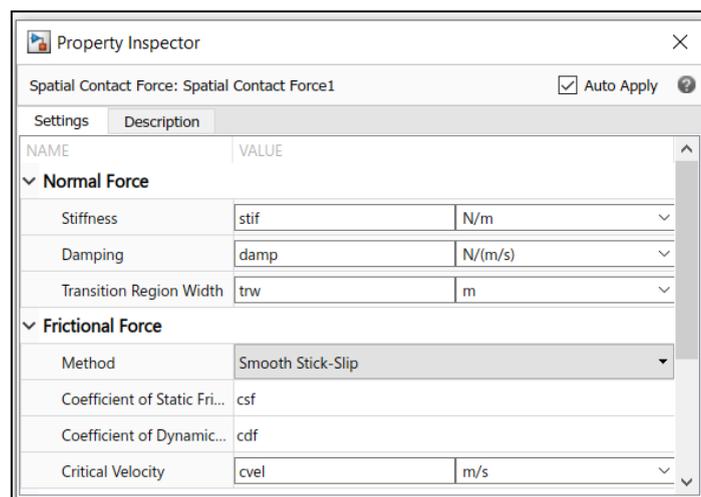


Figura 64. Coeficientes de las "Spatial Contact Forces".

Existen tres parámetros que definen la fuerza normal entre sólidos, y son los siguientes:

- **Stiffness (Rigidez):** Representa la resistencia del contacto a la penetración geométrica, es decir, cuanto mayor es el valor, más duro es el contacto entre las superficies. En este caso el valor asignado ha sido el valor por defecto para contactos no deslizantes, de 10^6 N/m, evitando así que pueda haber cualquier tipo de intersección entre geometrías.

- **Damping (Amortiguación):** Define la resistencia de amortiguación mientras dos geometrías están en contacto. Cuanto mayor es el valor, más energía se pierde en el contacto, por lo que un valor de 0 representaría una colisión perfectamente elástica, en la que toda la energía se conserva en forma de movimiento. En este caso el valor asignado ha sido el valor por defecto para contactos no deslizantes, de 10^3 N/(m/s).
- **Transition Region Width (Ancho de la Región de Transición):** Representa la región en la que la fuerza de contacto es máxima. Cuanto menor es el valor, más precisa es la simulación, pero menos rápida. En este caso el valor asignado ha sido el valor por defecto para contactos no deslizantes, de 10^{-4} m.

Además, existen otros tres parámetros que regulan el comportamiento de las fuerzas de rozamiento entre sólidos, siendo los que se definen a continuación:

- **Coefficient of Static Friction (Coeficiente de Rozamiento Estático):** Es el ratio de magnitud entre fuerza de rozamiento y fuerza normal cuando la velocidad tangencial tiende a cero. En otras palabras, determina la cantidad de rozamiento que ejercerá la rueda contra el suelo en el momento del arranque desde parado.

El valor escogido es de 0,7, ya que las ruedas debemos simular que son de un material elástico como el caucho, para que tengan un buen agarre y generen una tracción idónea.

- **Coefficient of Dynamic Friction (Coeficiente de Rozamiento Dinámico):** Es el ratio de magnitud entre fuerza de rozamiento y fuerza normal cuando la velocidad tangencial es elevada. En otras palabras, determina la cantidad de rozamiento que ejercerá la rueda contra el suelo cuando el vehículo ya se encuentre en movimiento.

El valor escogido es de nuevo de 0,7. Teóricamente debería ser menor que el coeficiente de fricción estático, pero tras iterar en la simulación, el comportamiento más realista al ojo humano se da con este valor.

- **Critical Velocity (Velocidad Crítica):** Es la velocidad que determina el cambio de comportamiento de estático a dinámico. En nuestro caso no es determinante, ya que ambos coeficientes de rozamiento son idénticos, por lo que hemos dejado el valor por defecto de 0,01 m/s.

4.4.4. SENSORIZACIÓN DEL VEHÍCULO

En el campo de la ingeniería, la mejora y la medición van unidas de la mano, y es por esto que una de las partes más importantes en el desarrollo de cualquier vehículo es la sensorización del mismo.

En este caso, tenemos tres objetivos muy definidos:

- Debemos conocer en todo momento la **posición del centro de masas del vehículo** para observar si sigue correctamente la trayectoria, sin experimentar desviaciones indeseadas.

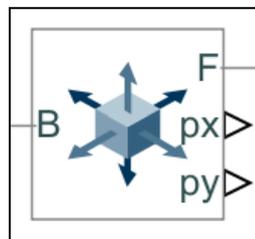


Figura 65. Cartesian Joint con Sensorización de Posición en X e Y.

Estas mediciones las extraeremos activando directamente las salidas sensorizadas en el bloque "Cartesian Joint", midiendo la posición en los ejes X e Y.

- Necesitamos controlar el ángulo de cabeceo del chasis, para evitar que parte de los instrumentos que se alojan en su interior puedan resultar dañados.

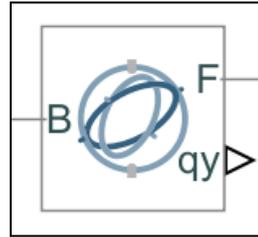


Figura 66. Gimbal Joint con Sensorización de Rotación Respecto a Y.

Esta medición la extraeremos activando directamente la salida sensorizada en el bloque "Gimbal Joint", midiendo la rotación respecto al eje Y.

- Es imprescindible medir la velocidad angular de las ruedas, para poder diseñar un buen control de velocidad del vehículo.

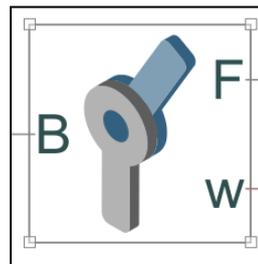


Figura 67. Revolute Joint con Sensorización de Velocidad.

Esta medición la extraeremos activando directamente la salida sensorizada de velocidad en los bloques "Revolute Joint" que unen cada una de las ruedas con los motores que las accionan.

4.5. DISEÑO DE LA TRAYECTORIA

En este caso, por tratarse de una simulación enfocada a la superación de obstáculos, la trayectoria que debe seguir el vehículo ha sido diseñada a partir del entorno, con el fin de superar el mayor número de éstos de la forma más rápida y óptima posible.

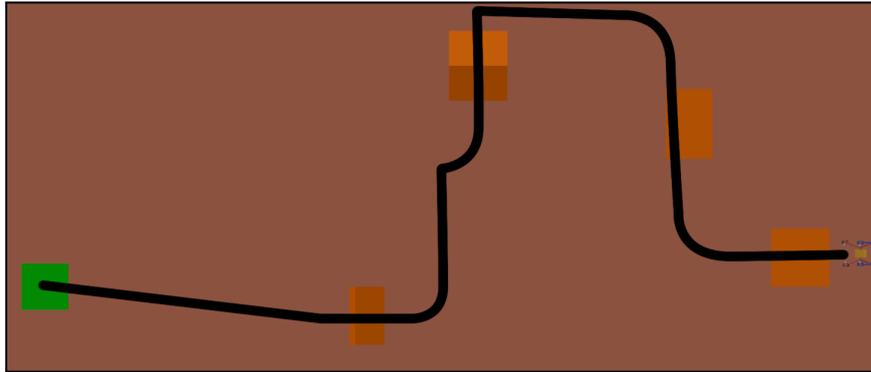


Figura 68. Trayectoria Deseada del Vehículo.

La duración de la trayectoria es de 36 segundos, por lo que vamos a descomponerla en tramos de tiempo, según las acciones que debe realizar el vehículo en dicho tramo:

Inicio (s)	Fin (s)	Descripción
0	1	Aterrizaje y arranque.
1	5	Superación de un conjunto de cuatro escalones de subida y bajada con todas las ruedas en desplazamiento rectilíneo frontal.
5	6,5	Giro de 90° hacia la derecha.
6,5	9	Superación de un conjunto de tres escalones de subida y bajada con las tres ruedas del tren derecho en desplazamiento rectilíneo frontal.
9	10,5	Giro de 90° hacia la izquierda.
10,5	13	Desplazamiento rectilíneo frontal de aproximación a la primera rampa.
13	13,5	Rotación de 90° de todas las ruedas del vehículo.
13,5	20	Superación de una rampa de subida y bajada extensa pero de pendiente poco pronunciada con todas las ruedas en desplazamiento lateral.

20	23,5	Giro de 90° hacia la izquierda marcha atrás.
23,5	25	Desplazamiento rectilíneo marcha atrás de aproximación a la segunda rampa.
25	26,5	Giro de 90° hacia la derecha marcha atrás.
26,5	29	Superación de una rampa de subida y bajada de pendiente pronunciada con todas las ruedas en desplazamiento rectilíneo marcha atrás.
29	33	Desplazamiento rectilíneo marcha atrás de aproximación a la posición de destino.
33	33,5	Rotación de 45° de las cuatro ruedas externas, formando un rombo.
33,5	36	Giro del vehículo sobre su propio eje vertical.

Tabla 8. Descripción de Acciones de Trayectoria y Relación con Tiempos.

4.6. GENERACIÓN DE SEÑALES DE LOS MOTORES

Para que el vehículo siga la trayectoria deseada y sea capaz de superar todos los obstáculos, debemos generar un tipo de señal concreta en los motores, con el fin de que la velocidad y la orientación de las ruedas sea la deseada en todo momento.

Esta tarea se dividirá en dos subtareas, que a su vez deberán estar interrelacionadas entre ellas, y serán las siguientes:

- Generación de señales en los servos para controlar la dirección del vehículo.
- Generación de señales en los motores de tracción para controlar la velocidad del vehículo.

4.6.1. SERVOS

El vehículo cuenta con seis servomotores, uno por cada rueda, los cuales tienen la misión de hacer rotar el conjunto formado por la rueda y el arco estructural, con el fin de permitir variedad de desplazamientos en el Rover.

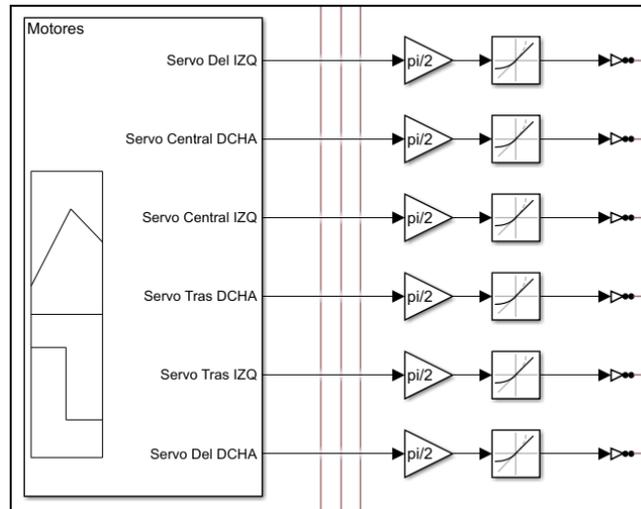


Figura 69. Sistema de Señales de los Servomotores.

Como se puede observar en la figura, cada uno de los servomotores tendrá su señal independiente, ya que, debido a la variabilidad de movimientos que el vehículo puede registrar, no hay dos ruedas que giren en el mismo sentido en todos ellos.

Además, empleamos otros tres bloques adicionales entre la señal generada y la unión de revolución a la que se la adjudicamos, siendo estos los siguientes:

- **Ganancia:** Es un multiplicador. En este caso lo escogemos de $\pi/2$, para conseguir que un valor de 1 en la señal sea equivalente a un giro de 90° en la rueda, con el fin de hacer la tarea más intuitiva.
- **Rate Limiter:** Este bloque limita el paso de la primera derivada, o lo que es lo mismo, se utiliza para evitar que el cambio de la señal que le llega a la revolución sea demasiado brusco. Empleamos un rango de valores estándar de entre -10 y 10.

- **Simulink-PS Converter:** Convierte la señal de un bloque o sistema de Simulink en una señal física que puede ser utilizada como entrada en una simulación de Simscape.

Ahora sí, conociendo toda la información necesaria, vamos a proceder a desglosar cada una de las señales de los servomotores.

En este caso, la descompondremos en tramos de tiempo, como ya hicimos en el apartado anterior con el diseño de la trayectoria.

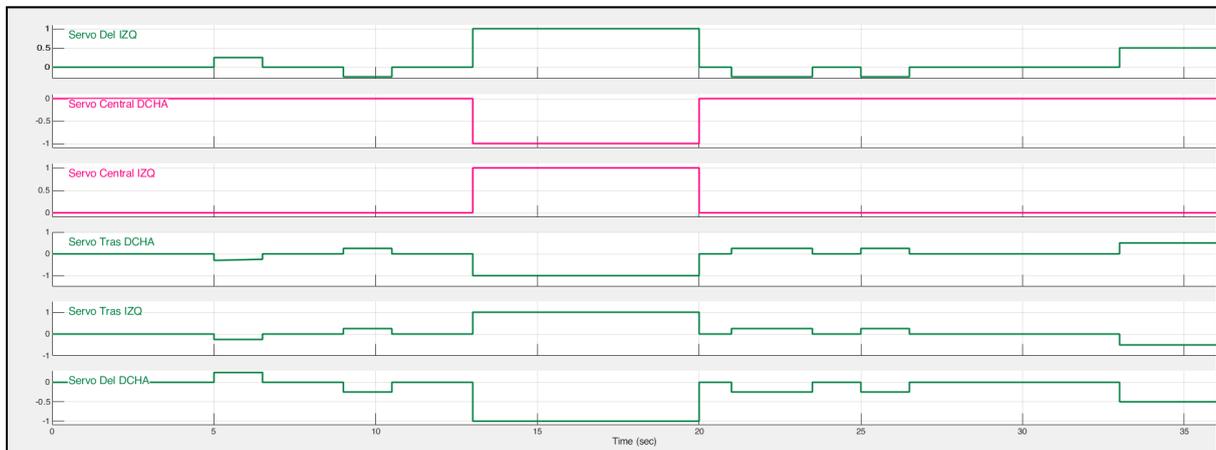


Figura 70. Gráfica de las Señales de los Servomotores.

En la siguiente tabla se especificará el valor de cada una de las señales en cada tramo, junto con una breve explicación del resultado sobre el comportamiento del vehículo:

Inicio (s)	Fin (s)	Servo Rueda Delantera Derecha	Servo Rueda Delantera Izquierda	Servo Rueda Central Derecha	Servo Rueda Central Izquierda	Servo Rueda Trasera Derecha	Servo Rueda Trasera Izquierda
0	5	0	0	0	0	0	0
El vehículo está preparado para un desplazamiento rectilíneo.							
5	6,5	0,25	0,25	0	0	-0,25	-0,25

Los servos de las ruedas delanteras giran $22,5^\circ$ en sentido horario, los servos centrales se mantienen en la posición original, y los servos de las ruedas traseras giran $22,5^\circ$ en sentido antihorario.

El vehículo está preparado para hacer un giro hacia la derecha marcha adelante o un giro hacia la izquierda marcha atrás.

6,5	9	0	0	0	0	0	0
-----	---	---	---	---	---	---	---

El vehículo está preparado para un desplazamiento rectilíneo.

9	10,5	-0,25	-0,25	0	0	0,25	0,25
---	------	-------	-------	---	---	------	------

Los servos de las ruedas delanteras giran $22,5^\circ$ en sentido antihorario, los servos centrales se mantienen en la posición original, y los servos de las ruedas traseras giran $22,5^\circ$ en sentido horario.

El vehículo está preparado para hacer un giro hacia la izquierda marcha adelante o un giro hacia la derecha marcha atrás.

10,5	13	0	0	0	0	0	0
------	----	---	---	---	---	---	---

El vehículo está preparado para un desplazamiento rectilíneo.

13	20	-1	1	-1	1	-1	1
----	----	----	---	----	---	----	---

Los servos de las ruedas de la mitad derecha del vehículo giran 90° en sentido antihorario, y los servos de las ruedas de la mitad izquierda giran 90° en sentido horario.

El vehículo está preparado para realizar un desplazamiento lateral.

20	21	0	0	0	0	0	0
----	----	---	---	---	---	---	---

El vehículo está preparado para un desplazamiento rectilíneo.

21	23,5	-0,25	-0,25	0	0	0,25	0,25
----	------	-------	-------	---	---	------	------

<p>Los servos de las ruedas delanteras giran 22,5^º en sentido antihorario, los servos centrales se mantienen en la posición original, y los servos de las ruedas traseras giran 22,5^º en sentido horario.</p> <p>El vehículo está preparado para hacer un giro hacia la izquierda marcha adelante o un giro hacia la derecha marcha atrás.</p>							
23,5	25	0	0	0	0	0	0
<p>El vehículo está preparado para un desplazamiento rectilíneo.</p>							
25	26,5	-0,25	-0,25	0	0	0,25	0,25
<p>Los servos de las ruedas delanteras giran 22,5^º en sentido antihorario, los servos centrales se mantienen en la posición original, y los servos de las ruedas traseras giran 22,5^º en sentido horario.</p> <p>El vehículo está preparado para hacer un giro hacia la izquierda marcha adelante o un giro hacia la derecha marcha atrás.</p>							
26,5	33	0	0	0	0	0	0
<p>El vehículo está preparado para un desplazamiento rectilíneo.</p>							
33	36	-0,5	0,5	0	0	0,5	-0,5
<p>El servo delantero derecho y el servo trasero izquierdo giran 45^º en sentido antihorario, mientras que el servo delantero izquierdo y el servo trasero derecho giran 45^º en sentido horario.</p> <p>El vehículo está preparado para hacer un giro sobre sí mismo sin desplazarse.</p>							

Tabla 9. Descripción de Señales de Servomotores y Relación con Tiempos.

4.6.2. MOTORES DE TRACCIÓN

El vehículo cuenta con seis motores de tracción, al igual que como sucedía con los servomotores, uno por cada rueda, los cuales tienen la misión de hacer rotar las

ruedas respecto al eje horizontal, con el fin de propulsar los desplazamientos del Rover.

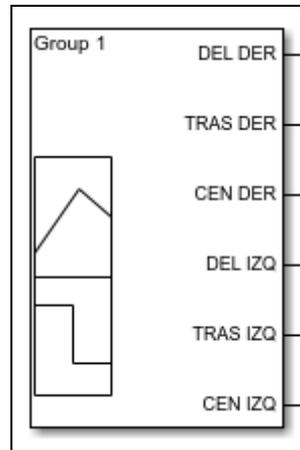


Figura 71. Subsistema de Señales de los Motores.

De nuevo, cada uno de los motores tiene su señal independiente del resto. En este caso, para el simple desplazamiento del vehículo no sería estrictamente necesario, ya que sí que hay ruedas que tienen comportamientos idénticos en toda la simulación. Pero en este caso es necesario, ya que aquí entra el diseño de control, el cuál explicaremos en el siguiente capítulo.

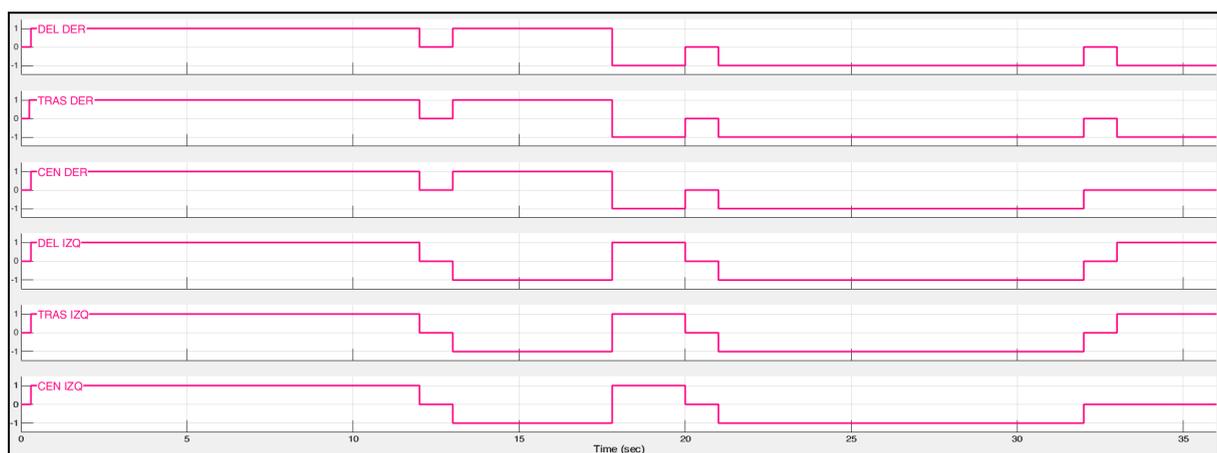


Figura 72. Gráfica de las Señales de los Servomotores.

Como con los servomotores, descompondremos la simulación en tramos de tiempo, indicando las señales y la descripción del movimiento en la siguiente tabla:

Inicio (s)	Fin (s)	Motores Delantero y Trasero Derecho	Motores Delantero y Trasero Izquierdo	Motor Central Derecho	Motor Central Izquierdo
0	0,3	0	0	0	0
El vehículo se encuentra parado.					
0,3	12	1	1	1	1
El vehículo es propulsado marcha adelante.					
12	13	0	0	0	0
El vehículo se encuentra parado.					
13	17,8	1	-1	1	-1
El vehículo es propulsado lateralmente.					
17,8	20	-1	1	-1	1
El vehículo es propulsado lateralmente en el sentido contrario al anterior.					
20	21	0	0	0	0
El vehículo se encuentra parado.					
21	32	-1	-1	-1	-1
El vehículo es propulsado marcha atrás.					
32	33	0	0	0	0
El vehículo se encuentra parado.					
33	36	-1	1	0	0
El vehículo rota sobre sí mismo.					

Tabla 10. Descripción de Señales de Motores y Relación con Tiempos.

4.7. CONTROL DE LOS MOTORES DE TRACCIÓN

Como mencionábamos en el apartado anterior, dos ruedas con comportamientos idénticos deberían compartir su señal de entrada, pero en este caso, el diseño de control requiere que cada rueda tenga su propia entrada.

Esto viene dado por el siguiente motivo, y es que, si decidimos diseñar un control de velocidad, como es el caso, cada rueda debe tener una señal de entrada independiente, ya que las irregularidades del terreno pueden propiciar que una rueda necesite más par que otra a pesar de estar girando ambas a la misma velocidad.

En este capítulo explicaremos la estructura del bucle de control, el diseño del controlador y los valores finales resultantes.

4.7.1. BUCLE DE CONTROL E IMPLEMENTACIÓN

En este caso, nuestro bucle de control será de tipo Proporcional-Integral (PI), ya que es el idóneo para sistemas de primer orden como el nuestro, en el que la acción derivada no es necesaria, por no haber una subamortiguación o un sobreoscilamiento severo en el sistema causado por el propio motor.

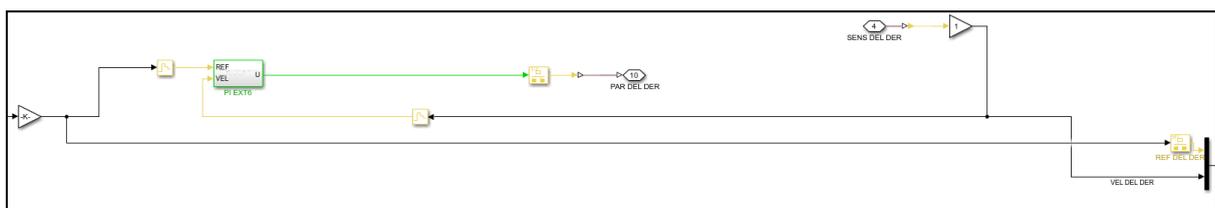


Figura 73. Bucle de Control de una de las Ruedas.

En la figura se puede apreciar el bucle de control completo perteneciente a una de las ruedas, pero procederemos a continuación a desglosarlo y describirlo:

- **Entrada del Sistema:** En este caso la entrada al sistema es la señal generada.

- **Ganancia:** Es un multiplicador que aplicamos a las entradas del sistema. En este caso, al ser el vehículo simétrico, para que una señal positiva signifique una rueda girando hacia delante, las ruedas del lado derecho deben llevar la ganancia positiva, y las ruedas del lado izquierdo deben llevar el mismo valor pero en negativo.
- **Controlador:** Es el elemento que regula la variable de salida mediante el cálculo de la acción de control a aplicar. En el siguiente subapartado explicaremos el controlador con más profundidad.
- **Salida del Bucle:** En este caso, la salida del bucle será el par que deberemos aplicar a la unión de revolución de la rueda para que traccione.
- **Realimentación:** La realimentación de este bucle es la lectura de velocidad que extraemos directamente de cada una de las ruedas.
- **Graficado:** Finalmente, extraemos en una gráfica la velocidad de referencia y la lectura real de velocidad, con el fin de poder compararlas para diseñar correctamente el controlador.

4.7.2. DISEÑO DEL CONTROLADOR

En este subapartado se va a explicar el proceso seguido para escoger los valores finales del controlador, así como las pruebas realizadas. No se explicará en qué consiste un controlador ni sus partes, ya que está explicado en el estudio teórico.

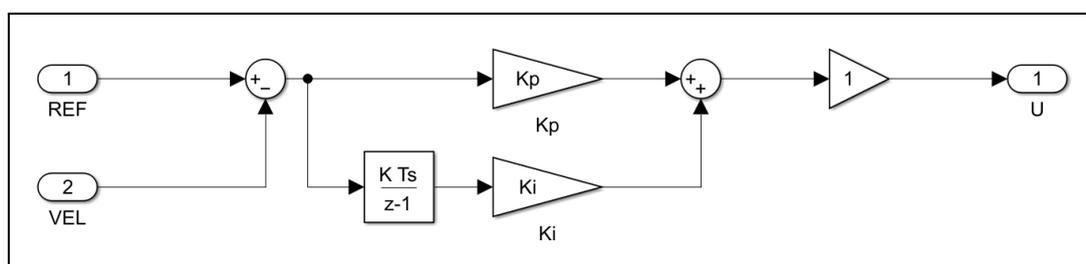


Figura 74. Controlador sin Valores Definidos.

Para la observación de las gráficas emplearemos figuras únicamente de una de las ruedas, ya que las seis son prácticamente idénticas y así se puede apreciar mejor la relación entre referencia y señal real.

- **Primera Prueba ($K_p = 0,001$; $K_i = 0,025$):** Para la prueba inicial escogemos valores estándar, para determinar si el orden de magnitud es el correcto.

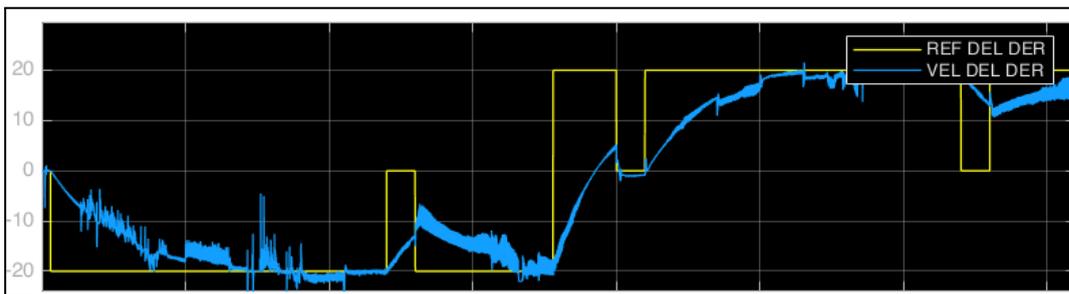


Figura 75. Primera Prueba ($K_p = 0,001$; $K_i = 0,025$).

Como se puede apreciar, el sistema es excesivamente lento, aunque al final consigue llegar a la velocidad de referencia, por lo que aumentaremos ambas ganancias, pero anotaremos el valor de K_i como posible solución final.

Las sobreoscilaciones que se observan son error del programa, propiciado por el propio rozamiento de las ruedas con el suelo, por lo que no es necesario considerarlo para el diseño.

- **Segunda Prueba ($K_p = 0,01$; $K_i = 0,05$):** Como se puede apreciar en la figura, al aumentar ambas ganancias el resultado es algo mejor, aunque el arranque en transitorio sigue siendo bastante lento.

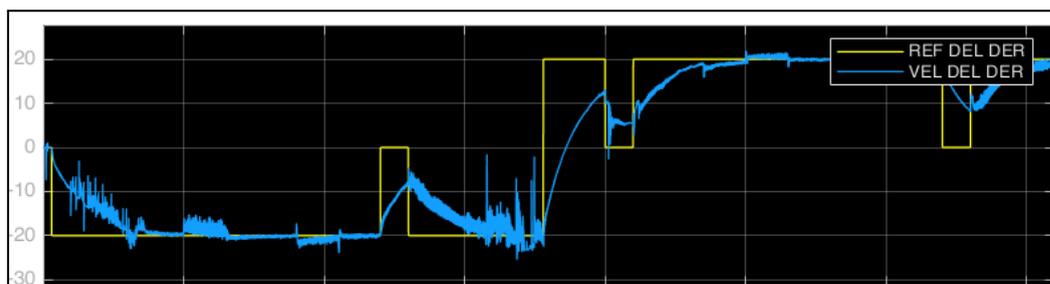


Figura 76. Segunda Prueba ($K_p = 0,01$; $K_i = 0,05$).

- **Tercera Prueba ($K_p = 0,03$; $K_i = 0,08$):** En esta ocasión observamos como la respuesta ya es algo aceptable para un vehículo, aunque siga lejos de ser óptima, pero ya es algo con lo que podríamos empezar a trabajar.

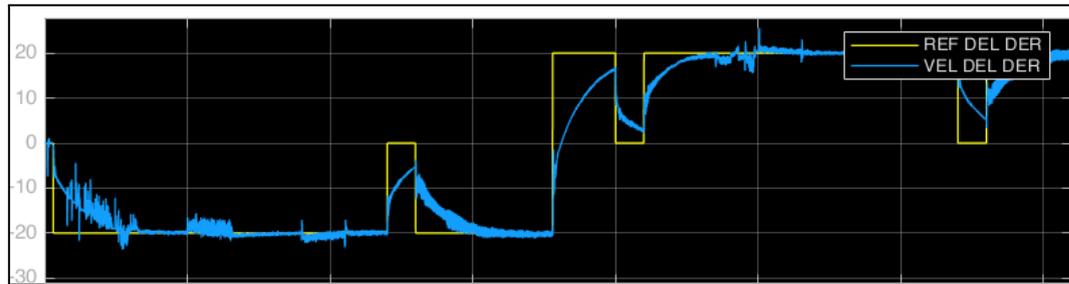


Figura 77. Tercera Prueba ($K_p = 0,03$; $K_i = 0,08$).

- **Cuarta Prueba ($K_p = 0,05$; $K_i = 0,08$):** Finalmente será la configuración que escojamos, ya que si seleccionamos valores superiores comienza a tener un comportamiento demasiado inestable.

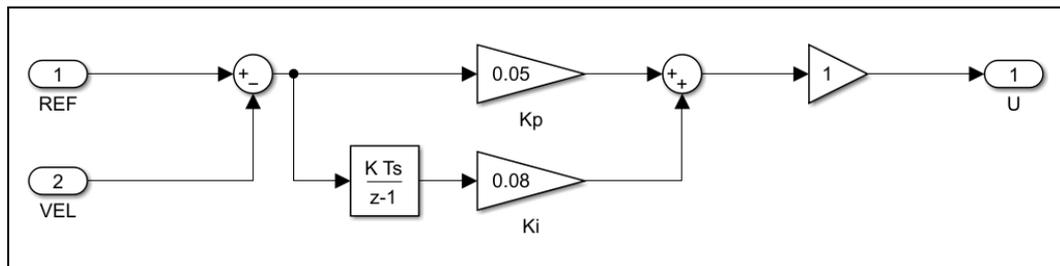


Figura 78. Controlador Final.

La gráfica la observaremos en el siguiente capítulo, en el que se muestran los resultados de la simulación.

4.8. RESULTADOS DE LA SIMULACIÓN

En este capítulo se aportarán los datos y las gráficas necesarias para evaluar el comportamiento del vehículo en la simulación, el cual es el objetivo de la misma.

4.8.1. VELOCIDAD ANTE REFERENCIA

Finalmente, como se ha indicado en el apartado anterior, se ha seleccionado el controlador realizado en la cuarta prueba, por haber llegado ya al punto en el que no existían mejoras significativas, y nos encontrábamos a las puertas de la inestabilidad.

Como se puede apreciar, el control está lejos de ser perfecto, pero en la mayoría de los tramos, especialmente en los largos, se alcanza la velocidad deseada y se mantiene sin apenas error, por lo que daremos por válido el control.

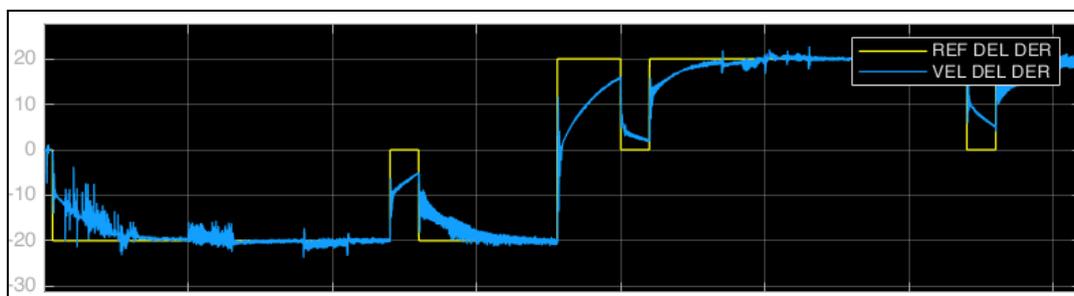


Figura 79. Velocidad ante Referencia con el Controlador Final.

En el caso de una aplicación en la que se requiriese más precisión, deberíamos optar por métodos de control más sofisticados, los cuales propondremos en el capítulo de posibles mejoras.

4.8.2. POSICIÓN DE LOS SERVOMOTORES

En cuanto a los servomotores, sí es cierto que necesitamos una precisión absoluta, para poder dirigir al Rover de la forma más óptima posible hacia los obstáculos, por lo que, si la señal original no tiene un comportamiento prácticamente perfecto, deberemos añadir control también a este componente.

Al ser las seis señales igual de precisas, únicamente graficaremos una de ellas, en concreto la del servomotor delantero izquierdo.

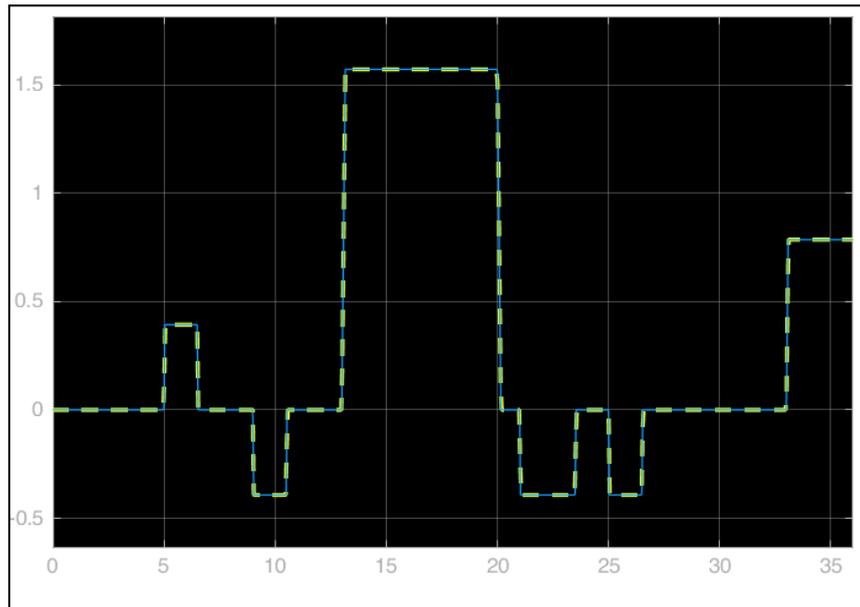


Figura 80. Señal vs Referencia del Servomotor Delantero Izquierdo.

Como se puede apreciar en la figura, ambas señales coinciden a la perfección, incluso en los momentos en los que se superan obstáculos, por lo que no es necesario añadir un control adicional.

4.8.3. POSICIÓN DEL CENTRO DE MASAS DE VEHÍCULO

Para este caso concreto, la forma de graficar será distinta, ya que en *Simulink* el eje X de las gráficas es por defecto el tiempo, por lo que tendremos que extraer los datos de la posición en X y en Y como dos vectores independientes.

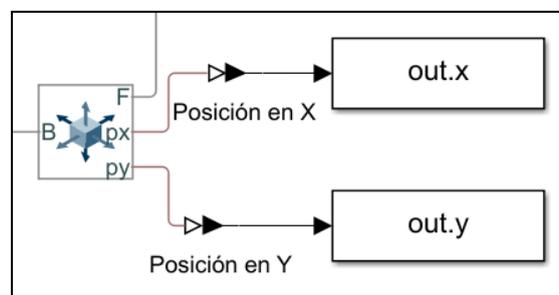


Figura 81. Extracción de Datos de Posición en Vectores.

A continuación, únicamente debemos indicar a MATLAB que deseamos una figura en la que el eje X esté representado por la evolución de la posición del vehículo en X, y lo mismo con el eje Y.

```
plot(out.x,out.y)
xlim([-15 0])
ylim([-1.5 4.5])
title('Trayectoria del Vehículo')
xlabel('Posición en X')
ylabel('Posición en Y')
```

Figura 82. Código de Generación de Figura en MATLAB.

En este caso, como se puede observar, la trayectoria que dibujan los sensores de posición es casi idéntica a la trayectoria diseñada. Únicamente existen ciertas desviaciones debidas a movimientos indeseados al subir y bajar de los obstáculos, por ser estos irregulares.

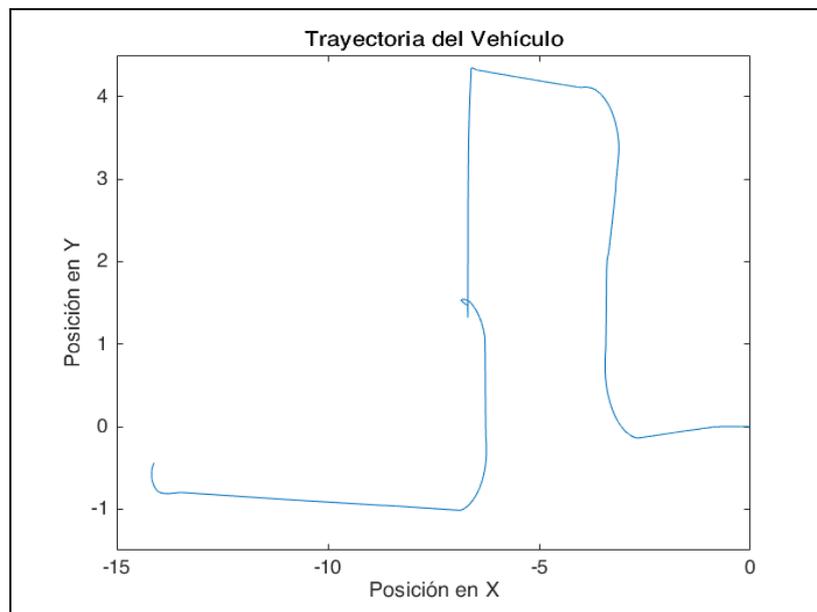


Figura 83. Trayectoria del Vehículo.

Esto ya podía ser observado a simple vista en la simulación, pero para misiones remotas es necesario comprobar que la sensorización se corresponde con la realidad.

4.8.4. ÁNGULO DE CABECEO DEL CHASIS DEL VEHÍCULO

En este caso, el ángulo de cabeceo lo medimos para comprobar que el propio chasis, sin fijaciones adicionales, nunca vuelca más de 45° , que es el ángulo a partir del cual se podrían dañar los instrumentos que se alojan en su interior.

En la gráfica, el ángulo viene expresado en radianes, por lo que nunca deberemos superar el valor absoluto de 0,785.

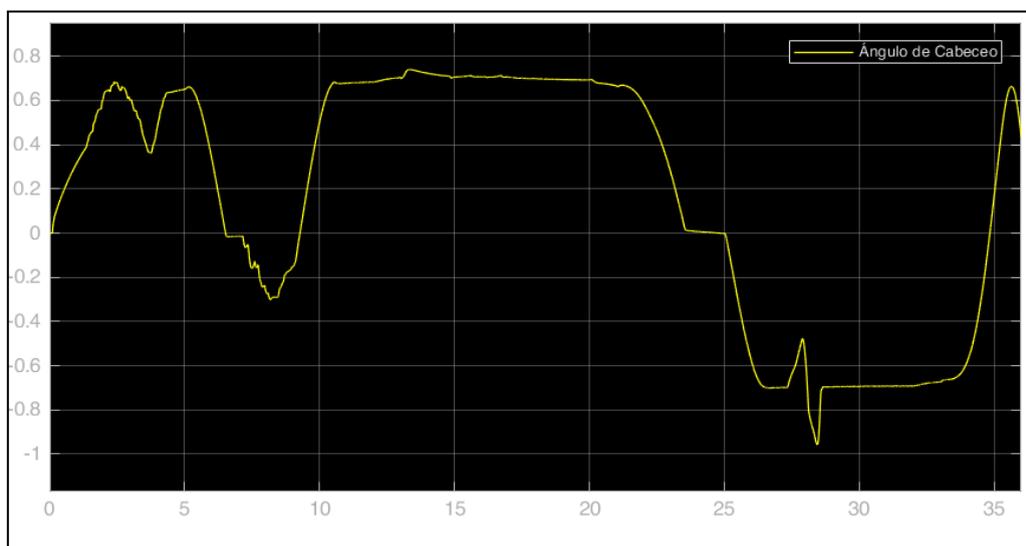


Figura 84. Ángulo de Cabeceo.

Como se puede apreciar, a pesar de que en la mayoría del tiempo estamos dentro de límites aceptables, existe un tramo en el segundo 28 en el que sobrepasamos el valor crítico.

Además, cabe destacar que sí existe un balanceo continuo, el cual no es para nada deseable en este tipo de misiones, por lo que se le añadirá al prototipo final una fijación adicional.

5. IMPLEMENTACIÓN REAL

En este capítulo del trabajo se explicará todo el proceso de implementación física del vehículo, abarcando desde la fabricación de los componentes y su ensamblaje hasta el diseño del control manual y de las conexiones del microprocesador.

5.1. FABRICACIÓN

En este apartado definiremos todos los procesos de fabricación de los componentes mecánicos del ensamblaje, así como las modificaciones que se han realizado a dichos componentes para poder ser fabricados por estos procesos.

5.1.1. SINTERIZADO SELECTIVO POR LÁSER (SLS)

Esta técnica de fabricación está basada en la impresión por adición, ya que se añade polvo a una cuba calentada a una temperatura ligeramente inferior al punto de fusión de este material.

A continuación, un láser de alta potencia apunta a los lugares donde deseamos que se solidifique el material, y se produce la sinterización, que no es más que la unión de partículas por fusión.



Figura 85. Limpieza de una Pieza Fabricada por SLS.

Este proceso de fabricación será el más empleado para este vehículo, obteniendo piezas ligeras y compactas, con aristas regulares y caras lisas.

5.1.2. CORTE POR HAZ LÁSER (LBC)

Este método de fabricación se basa en el corte de planchas mediante la aplicación de calor localizado, gracias a un haz láser.

El proceso físico que ocurre es la fundición o vaporización del material, pero al ser altamente localizada y muy precisa, obtenemos cortes limpios y rectos.



Figura 86. Máquina de Corte Láser en Acción.

En nuestro ensamblaje, las piezas fabricadas por corte láser serán las que tengan un espesor reducido y uniforme, y el material empleado será el PMMA, comúnmente conocido como metacrilato.

5.1.3. MODELADO POR DEPOSICIÓN FUNDIDA (FDM)

Por último, esta técnica consiste, como su nombre indica, en la deposición de material fundido. Más concretamente, lo que sucede exactamente en el proceso es que una boquilla calienta un filamento de material termoplástico hasta el punto de fusión y lo deposita formando capas de una altura determinada.

El material, al entrar en contacto con el aire, se solidifica, creando una capa de material sobre la que se depositará la siguiente capa, y así sucesivamente, hasta crear una pieza o componente completo.

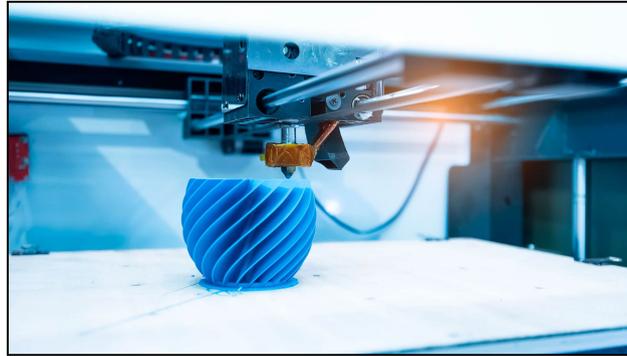


Figura 87. Máquina de FDM en Proceso de Fabricación.

Al obtener piezas de peor calidad por este proceso de fabricación, en nuestro caso únicamente fabricaremos las ruedas, por tener un tamaño excesivo para las máquinas de SLS de las que disponemos.

El material empleado será el PLA, un termoplástico de uso común en este tipo de técnicas de fabricación.

5.1.4. MODIFICACIONES REALIZADAS

En origen, el vehículo completo fue diseñado para ser fabricado por deposición de material fundido o FDM, pero finalmente se ha optado por emplear también el corte y el sinterizado láser.

Estos procesos de fabricación tienen limitaciones, por lo que las piezas deben ser diseñadas especialmente pensadas para ellos. Por esto mismo, a algunas piezas se le ha realizado ligeras modificaciones, las cuales serán explicadas a continuación.

5.1.4.1. CHASIS

El chasis en origen iba a ser un cuerpo único con forma de prisma al que se le ha añadido una cara inclinada.

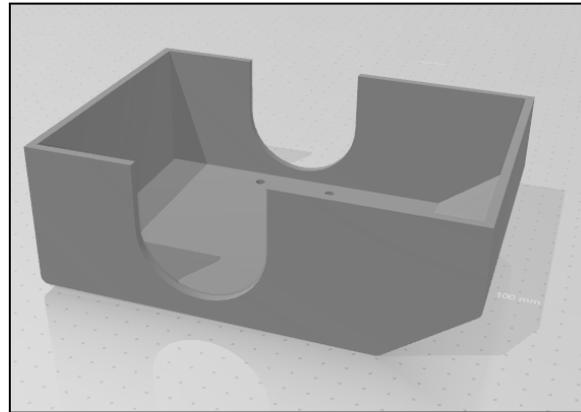


Figura 88. Diseño Original del Chasis.

Por su excesivo tamaño y la geometría de sus piezas, se decidió fabricarlo por corte láser, modificando su diseño de las siguientes formas:

- Supresión de la cara inclinada.
- División del cuerpo único en cinco caras independientes.
- Modelado de entrantes y salientes en las caras.

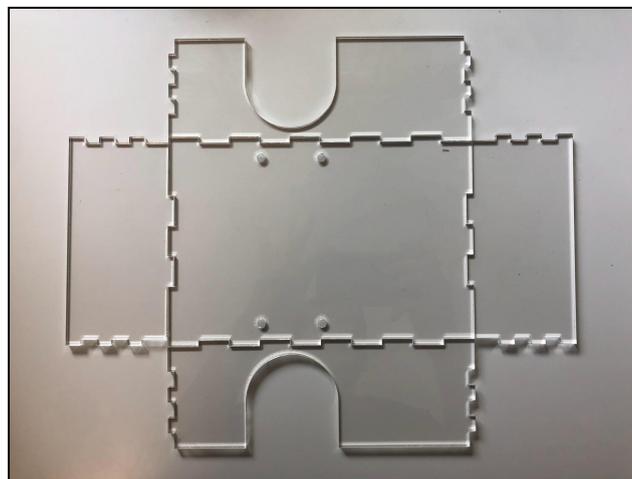


Figura 89. Despiece del Chasis.

Finalmente, las caras se unen mediante la aplicación de adhesivo y presión para formar el cuerpo completo deseado.

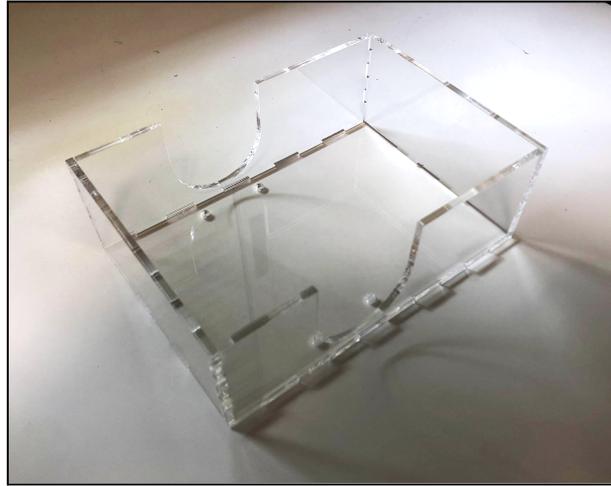


Figura 90. Chasis Completo Fabricado por LBC.

5.1.4.2. ACOPLER INTERIOR DE LA RUEDA

De nuevo, como sucede con el chasis, el acople interior de la rueda estaba pensado para ser una pieza única, pero por su excesivo espesor se decidió fabricar el doble de piezas mediante corte láser, para después unirlos.

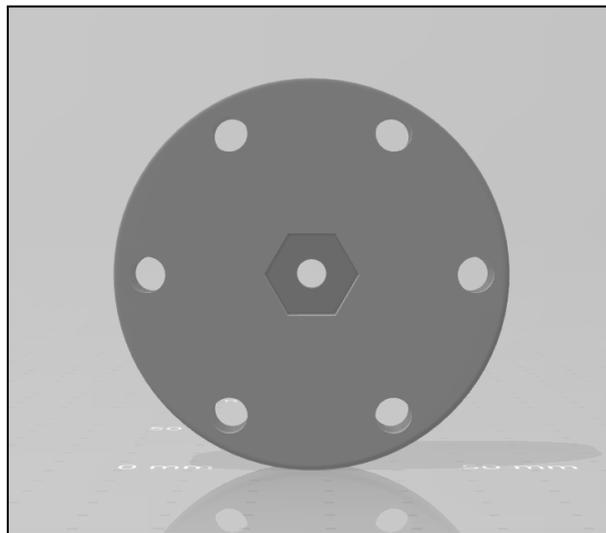


Figura 91. Diseño Original del Acople Interior de la Rueda.

Como se puede apreciar, en el centro hay un taladro y un rebaje de forma hexagonal, por lo que cada una de las mitades deberá tener uno de los taladros.

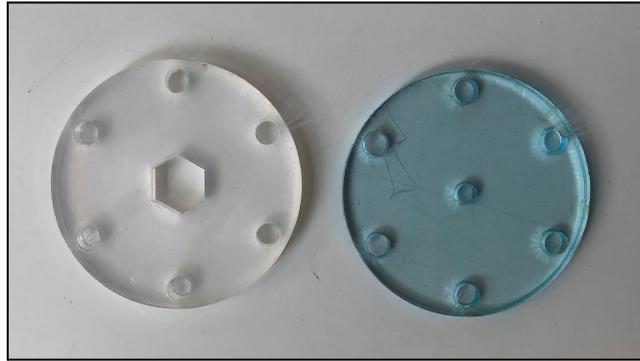


Figura 92. Mitades del Acople Interior de la Rueda Fabricadas con LBC.

Estas dos piezas se unirán posteriormente en el ensamblaje mediante uniones roscadas, quedando fijas y ejerciendo la misma función que el diseño original.

5.1.4.3. ARCO ESTRUCTURAL

En el caso del arco estructural, el problema se encuentra en el centro de la pieza, el cual es completamente hueco y está delimitado por paredes de espesor uniforme.

En el caso de la fabricación por deposición de material fundido esto no supondría un problema, pero por calidad y precisión geométrica se ha optado por fabricar la pieza por sinterización láser, y aquí sí supone un inconveniente.

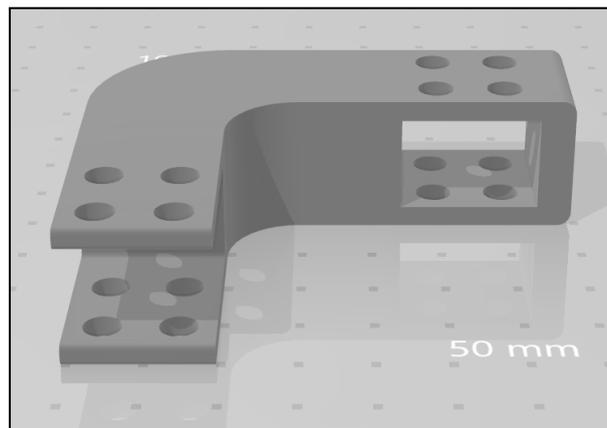


Figura 93. Diseño Original del Arco Estructural.

Como hemos explicado anteriormente, en el proceso de sinterización láser se solidifica polvo dentro de una cubeta. El polvo no sinterizado se limpia a

continuación, pero en el caso de haber una oquedad dentro de un cuerpo sin que ésta tenga ninguna salida al exterior, el polvo queda atrapado dentro.

Por esto mismo, se ha modificado la pieza para que en lugar de existir un hueco rodeado por paredes haya dos caras unidas por nervios en dirección radial, lo cual también añadirá una mayor resistencia mecánica.



Figura 94. Arco Estructural Fabricado por SLS.

5.1.5. RELACIÓN DE PIEZAS Y PROCESOS DE FABRICACIÓN

A continuación se adjunta una tabla en la que se relacionarán los componentes mecánicos del ensamblaje y el método por el que han sido fabricados:

Método de Fabricación	Pieza
Sinterizado Láser	<ul style="list-style-type: none"> ● Enlace Constructivo ● Rodamiento ● Acoplamiento Rueda Delantera ● Barra Constructiva ● Sujeción Rodamiento Principal ● Sujeción Suspensión ● Alojamiento Motor ● Acoplamiento Rodamiento Trasero ● Alojamiento Rodamiento Trasero ● Acoplamiento Rueda Trasera ● Enlace Suspensión-Construcción ● Arco Estructural ● Acoplamiento Rodamiento Servo

Corte Láser	<ul style="list-style-type: none"> ● Acople Interior Rueda ● Chasis ● Elemento de Construcción ● Elemento de Construcción 2
Deposición de Material Fundido	<ul style="list-style-type: none"> ● Rueda

Tabla 11. Relación de Piezas y Procesos de Fabricación.

5.2. ENSAMBLAJE

En este capítulo se ilustrará el proceso de ensamblaje manual llevado a cabo para conformar el mecanismo completo, paso a paso.

Se omitirá la enumeración y explicación de las piezas individualmente, ya que es idéntica a la desarrollada en el capítulo 3, en el que se describe el diseño de las piezas en Solidworks.

El primer paso será ensamblar cada una de las seis ruedas. Para ello debemos ensamblar la rueda junto con las dos mitades que forman el acople interior de la misma, mediante seis tornillos M5X40 y sus correspondientes tuercas, empleando la mano y una llave fija.

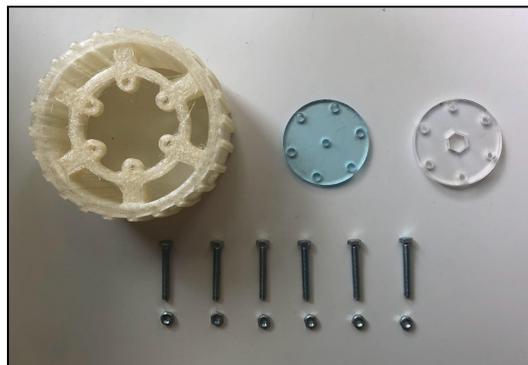


Figura 95. Despiece Ensamblaje Rueda.

En este paso es muy importante que la mitad del acople interior de la rueda que quede orientada hacia la cavidad interior sea la que tiene el recorte hexagonal,

para poder unir el motor a la rueda al final del ensamblaje, obteniendo el resultado final que se puede observar en la siguiente figura.



Figura 96. Ensamblaje Rueda.

El próximo paso será unir todos los componentes que conforman cada uno de los seis arcos estructurales de las ruedas.

Para ello uniremos el arco estructural por una de sus partes al alojamiento del motor, mediante dos tornillos M5x40, empleando el mismo método que en el paso anterior.

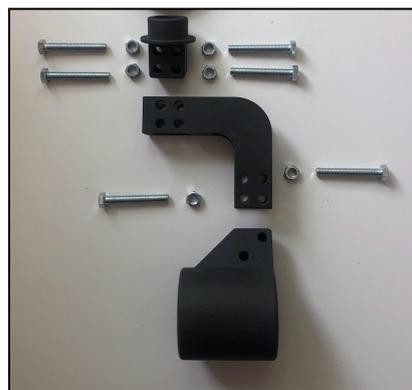


Figura 97. Despiece Ensamblaje Arco Estructural.

Además, también uniremos el arco por su parte superior al acoplamiento del rodamiento del servo, utilizando cuatro tornillos como los anteriores y el mismo método de llave fija y trabajo manual.



Figura 98. Ensamblaje Arco Estructural.

A este ensamblaje que conforma el arco estructural le debemos acoplar el motor, introduciéndolo dentro del alojamiento del mismo, haciendo que el eje se introduzca en la ranura que hay al final de éste.

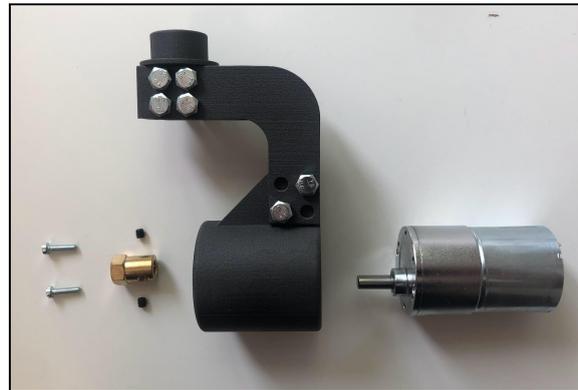


Figura 99. Despiece Ensamblaje Arco Estructural y Motor.

El motor se fijará al alojamiento mediante dos tornillos M3x10, utilizando un destornillador de punta plana, y al eje le fijaremos un tapón metálico mediante dos insertos de M2, que apretaremos con una llave Allen.

Este tapón metálico será la pieza que unirá a continuación al motor con el ensamblaje de la rueda.

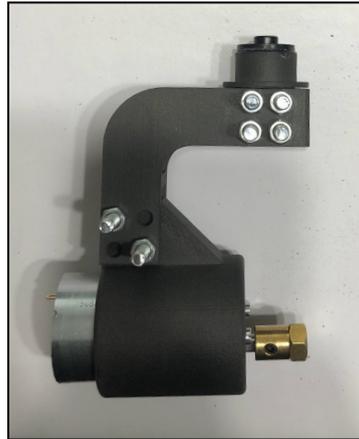


Figura 100. Ensamblaje Arco Estructural y Motor.

Una vez ensamblados los seis motores de corriente continua, procederemos a ensamblar los servomotores, uniéndolos con los acoplamientos de las ruedas mediante los insertos impresos para ello. A su misma vez, un rodamiento se introducirá dentro de cada uno de los acoplamientos de las ruedas, quedando unidos por apriete.



Figuras 101(a) , 101(b) y 101(c). Despiece Ensamblaje Servomotor y Acoplamiento Rueda.

Una vez ensamblado, la pletina circular de plástico que aparece en la figura se unirá al acoplamiento del rodamiento del servomotor mediante adhesivo, utilizando cianoacrilato y presión.



Figura 102. Despiece Ensamblajes Estructurales de la Rueda y Motores.

Esta unión entre la pletina y el acoplamiento del rodamiento del servomotor permitirá que, enlazando el servomotor y dicha pletina, unamos el ensamblaje del arco estructural y el del acoplamiento de la rueda, quedando ambos motores ya englobados en un mismo conjunto.

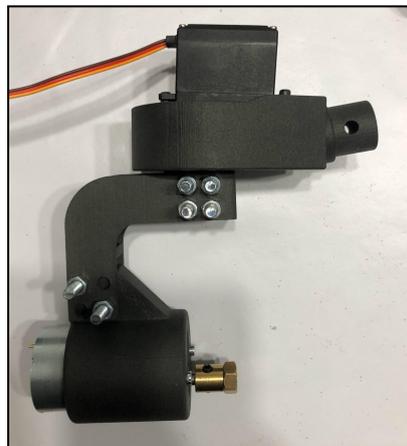


Figura 103. Ensamblajes Estructurales de la Rueda y Motores.

Una vez tenemos ya el ensamblaje de ambos motores, procedemos a operar sobre las suspensiones.

En primer lugar, ensamblamos las dos suspensiones principales, uniendo el enlace construcción-suspensión con la sujeción principal mediante dos tornillos M5x40 y

unión por cianoacrilato. Después, realizamos una doble inserción por apriete, de la sujeción dentro del rodamiento y del rodamiento dentro de su alojamiento.

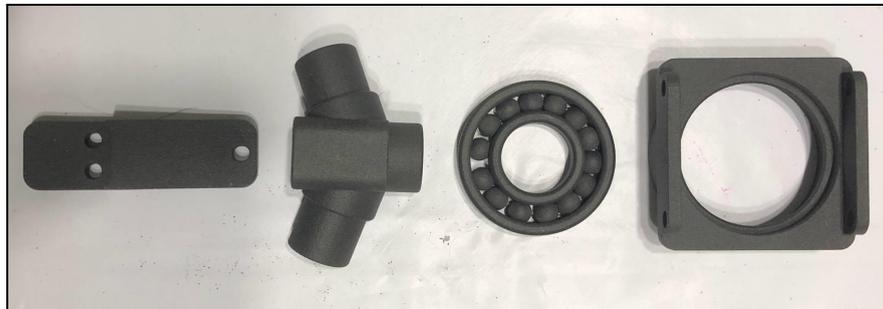


Figura 104. Despiece Ensamblaje Sujeción Suspensión.

El ensamblaje queda como en la figura siguiente:



Figuras 105(a y 105(b). Ensamblaje Sujeción Suspensión Alzado y Planta.

Y ahora debemos proceder a unir ambos con el chasis de metacrilato, mediante intersección y quedando fijados gracias a cuatro tornillos M5x40 y sus consiguientes roscas.

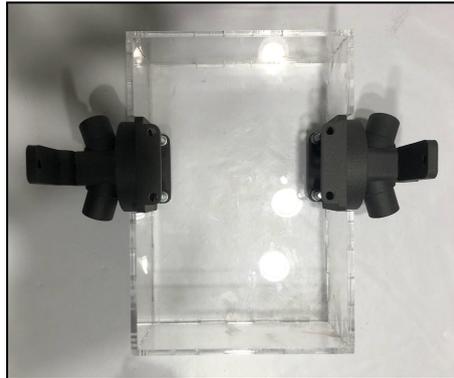


Figura 106. Ensamblaje Chasis y Sujeciones Suspensión.

Una vez tenemos conformado el cuerpo central, procedemos a cortar los tubos de PVC, de los cuales necesitamos dos de 18 cm de longitud, otros dos de 15 cm y otros dos de 13 cm. En la figura se puede apreciar uno de cada tamaño:



Figura 107. Tubos de PVC.

Una vez cortados los tubos, el siguiente paso será proceder a montar los sistemas *rocker-bogie*.



Figura 108. Esquema de Montaje Sistemas Rocker-Bogie.

Para ensamblar el *bogie* debemos unir los dos conjuntos de los motores cuyo acoplamiento de rueda sea para la trasera, es decir, que el pivote saliente sea recto y no oblicuo.



Figura 109. Acoplamiento Rueda Trasera.

Esto lo conseguiremos utilizando el tubo de PVC de longitud mediana, es decir, el de 15 cm, y haciéndole un taladro a cada lado en el que introduciremos un tornillo de M5x40, fijándolo por el otro extremo con su correspondiente tuerca.

Además, aprovecharemos para unir las ruedas, encajando los insertos en los acoples y asegurándolas con un tornillo de M4x10 cada una.

Por último, se hacen dos taladros más en la parte superior del tubo, para unir el acoplamiento del rodamiento trasero con dos tornillos M5x40, al cual le uniremos mediante apriete y adhesivo el alojamiento de dicho rodamiento, con éste dentro.

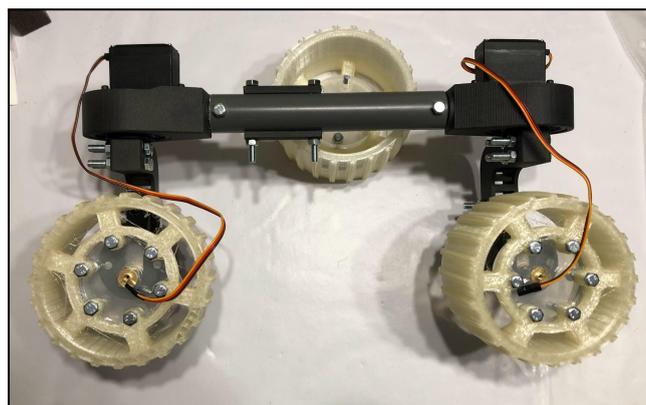


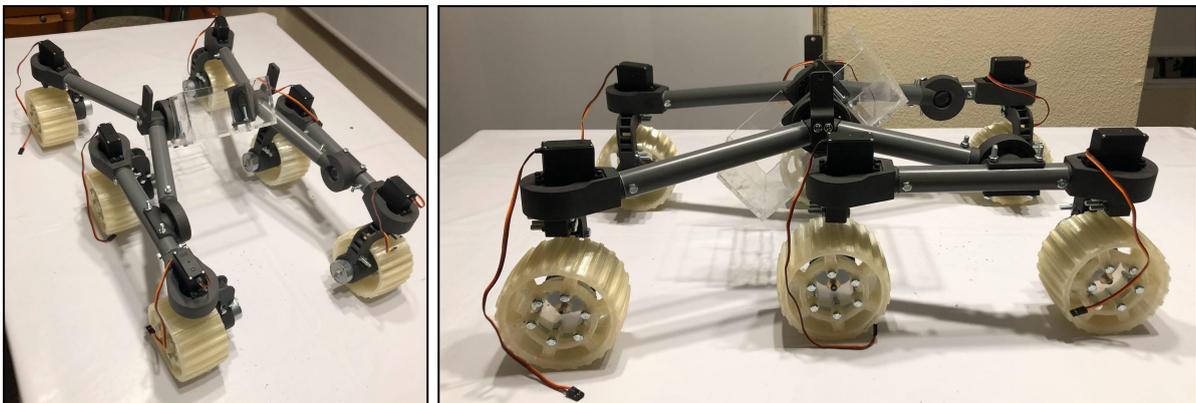
Figura 110. Ensamblaje Bogie.

Después, procedemos a ensamblar el *rocker*, haciendo lo mismo que con el *bogie*, haciendo taladros en los tubos de 18 cm y 13 cm de PVC, y uniendo mediante los tornillos de M5x40 el tubo largo al acoplamiento de la rueda delantera y el tubo corto al alojamiento del rodamiento trasero, uniendo ambos a su vez con la sujeción principal de la suspensión.



Figura 111. Ensamblaje Rocker-Bogie Completo.

Una vez completados los *rocker-bogie* y unidos ambos al cuerpo central, ya tenemos el mecanismo completo.



Figuras 112(a) y 112(b). Mecanismo Completo.

5.3. ELÉCTRICA Y ELECTRÓNICA

En este apartado se describirá la parte eléctrica y electrónica del vehículo.

Se realizará una breve descripción de todos los componentes del mismo, así como de las conexiones entre ellos.

La programación correspondiente a la puesta en funcionamiento de los mismos podrá ser consultada en el anexo dedicado exclusivamente para ello, no siendo descrita en este capítulo.

5.3.1. ARDUINO MEGA 2560 REV3

Es el microprocesador que se empleará para este proyecto. Cuenta con 54 puertos de entrada/salida digital, 15 de los cuales pueden ser utilizados como PWM, y 16 entradas analógicas.

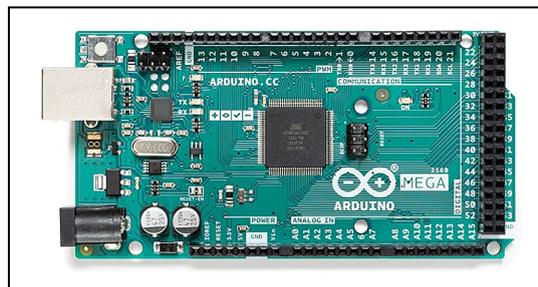


Figura 113. Arduino Mega 2560 Rev3.

A este elemento se conectarán todos los siguientes, por lo que, en vez de hacer un esquema de conexiones propio, se describirán las conexiones componente por componente.

Será alimentado desde una *power bank* que le proporcionará 5V.

5.3.2. SERVOMOTOR MG996R

Es el accionador encargado de rotar las ruedas para orientar el vehículo y dirigirlo en el sentido deseado de la marcha.



Figura 114. Servomotor MG996R.

Los servomotores no se conectan directamente con el microprocesador, sino que lo hacen a través de la placa PCA9685.

5.3.3. MOTOR DC 12V 100RPM

Es el accionador que se encarga de propulsar las ruedas. Funciona con 12V de corriente continua, y tiene una velocidad máxima de 100 rpm.



Figura 115. Motor DC 12V 100RPM.

Los motores tampoco se conectan directamente con el microprocesador, sino que lo hacen a través de los *drivers* L298N.

5.3.4. MÓDULO BLUETOOTH HC-06

El módulo *Bluetooth* HC-06 es el componente que permite la comunicación entre el microprocesador y el dispositivo que emplearemos para su control remoto.

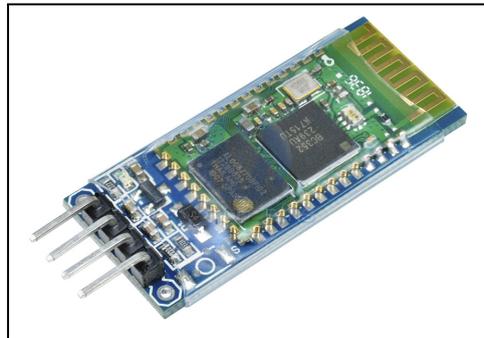


Figura 116. Módulo *Bluetooth* HC-06.

Este módulo se conecta directamente con el microprocesador, empleando cuatro pines, conforme se observa en la figura:

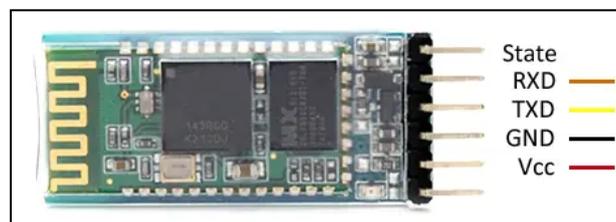


Figura 117. Conexiones Pines HC-06.

Los pines representan lo siguiente:

- **RXD:** Pin de recepción de información.
- **TXD:** Pin de transmisión de información.
- **GND:** Toma de tierra. 0V.
- **VCC:** Pin de alimentación. 5V.

Y se conectan con el microprocesador de la siguiente forma:

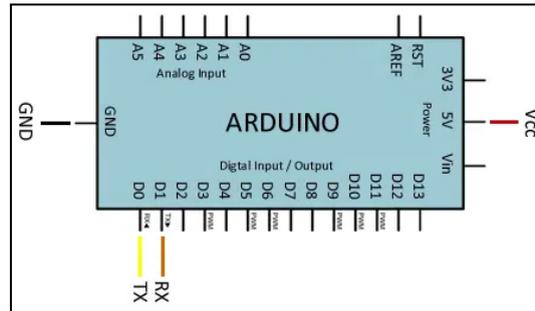


Figura 118. Conexión HC-06 con Arduino.

5.3.5. DRIVER L298N

El *driver* L298N permite alimentar y controlar dos motores de corriente continua simultánea e independientemente, variando la velocidad de giro y la dirección. Será alimentado por una batería externa de 12V.

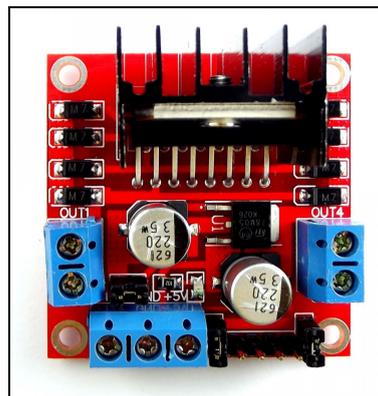


Figura 119. Driver L298N.

Este módulo se conecta directamente con el microprocesador y con los motores de continua, empleando seis pines para el Arduino, y un cable positivo y otro negativo por motor, conforme se observa en la figura:

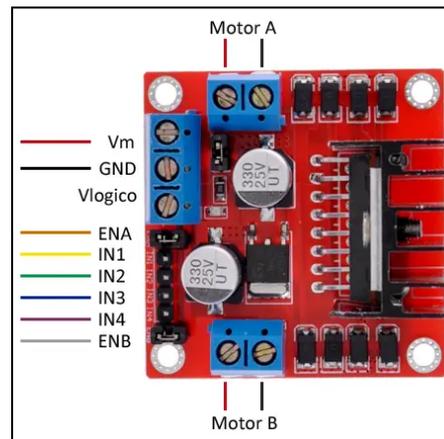


Figura 120. Conexiones Driver L298N.

Los pines tienen las siguientes funciones:

- **ENA:** Recibe la señal PWM que controla la velocidad del motor A.
- **IN1/IN2:** Reciben una señal digital cada uno, siendo una de ellas de 0 y la otra de 1, para indicar al motor A el sentido de giro deseado.
- **IN3/IN4:** Reciben una señal digital cada uno, siendo una de ellas de 0 y la otra de 1, para indicar al motor B el sentido de giro deseado.
- **ENB:** Recibe la señal PWM que controla la velocidad del motor B.
- **VM:** Recibe los 12V directamente de la fuente de alimentación.
- **GND:** Toma de tierra. Común al resto de GND del montaje.

Se conectan con el microprocesador de la forma siguiente:

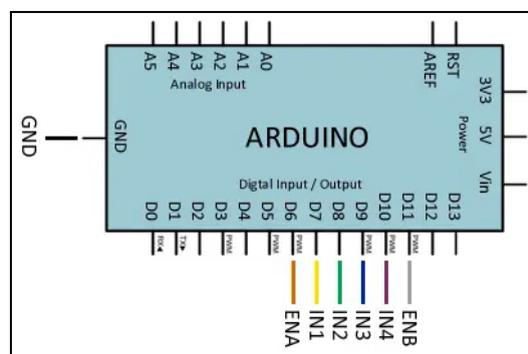


Figura 121. Conexiones Driver L298N con Arduino.

Cabe destacar que en este montaje tenemos seis motores de corriente continua, por lo que emplearemos 3 drivers L298N.

5.3.6. DRIVER PCA9685

El *driver* PCA9685 permite controlar hasta dieciséis servomotores simultánea e independientemente. Una de las grandes ventajas de este componente es que dispone de reloj propio, pudiendo generar sus propias señales PWM, y liberando al microprocesador de dicha tarea. Será alimentado por un portapilas con cuatro pilas AA.

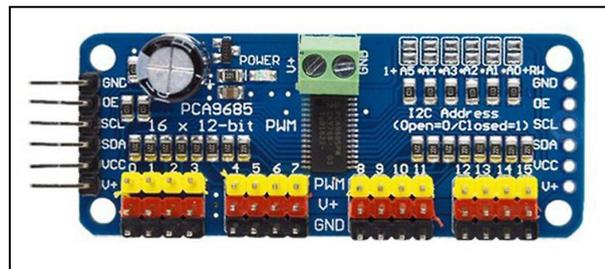


Figura 122. Driver PCA9685.

Este módulo se conecta directamente con el microprocesador y con los servomotores, empleando cuatro pines para el Arduino, y otros tres para cada servomotor, conforme se observa en la figura:

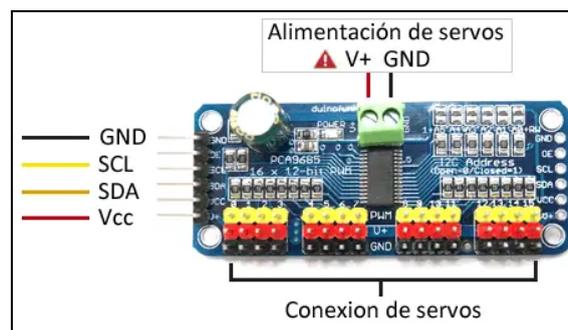


Figura 123. Conexiones Driver PCA9685.

Los pines que conectan con Arduino tienen las siguientes funciones:

- **SCL:** Es el pin correspondiente a la señal de reloj.
- **SDA:** Es el pin correspondiente al sensor de lectura de posición.
- **VCC:** Recibe los 6V directamente del portapilas.
- **GND:** Toma de tierra. Común al resto de GND del montaje.

Se conectan con el microprocesador de la forma siguiente:

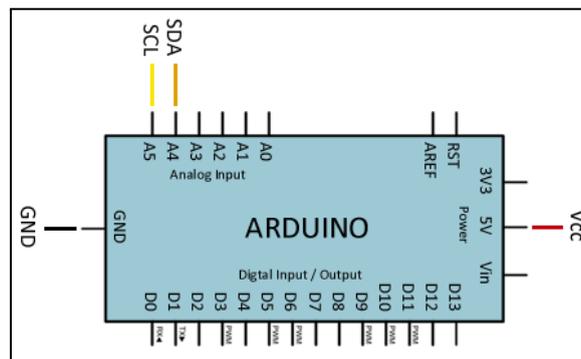


Figura 124. Conexiones Driver PCA9685 con Arduino.

Los pines que conectan con los servos tienen las siguientes funciones:

- **PWM:** Transmite al servomotor la señal modulada que le indicará cuánto girar y en qué sentido.
- **V+:** Alimenta al servomotor.
- **GND:** Toma de tierra. Común al resto de GND del montaje.

Cabe destacar que en este montaje tenemos seis motores de corriente continua, por lo que emplearemos 3 drivers L298N.

5.4. CONTROL MANUAL

En este capítulo se describirá el método de control manual escogido para dirigir el vehículo en tiempo real.

En concreto, se utilizará un *smartphone* con sistema operativo *Android* mediante una aplicación llamada *Bluetooth Electronics*, que permite personalizar interfaces para controlar remotamente dispositivos conectados con *Arduino*.

A continuación, se muestra en la siguiente figura la interfaz que emplearemos para controlar nuestro vehículo:

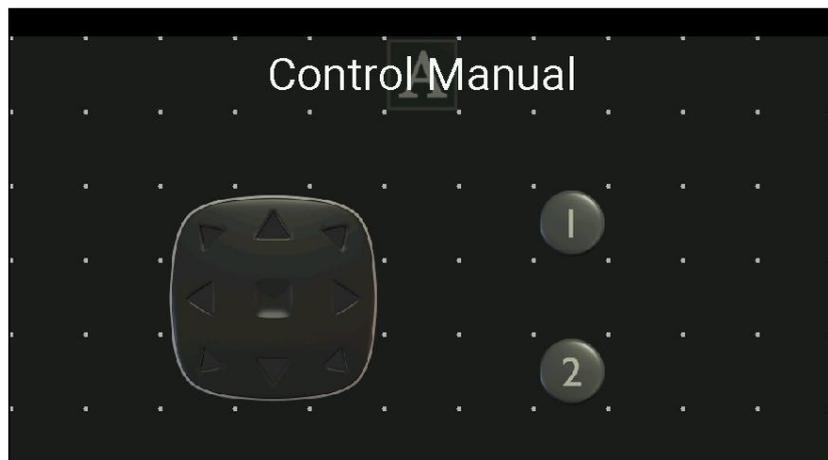


Figura 125. Interfaz *Bluetooth Electronics*.

Los elementos de control tendrán las siguientes funciones:

- **Flechas:** Se encargan de controlar movimientos de desplazamiento rectilíneo y curvilíneo, tanto frontales, como laterales y giros, existiendo ocho tipos de movimiento:
 - Recto hacia delante y atrás.
 - Recto hacia derecha e izquierda.
 - Curvo hacia derecha e izquierda, marcha adelante y marcha atrás.
- **Botones:** El botón número 1 ordena al vehículo que rote sobre su eje vertical en sentido horario, mientras que el botón número 2 ordena al vehículo que rote sobre su eje vertical en sentido antihorario.

6. PLIEGO DE CONDICIONES

6.1. DEFINICIÓN Y ALCANCE

El presente proyecto consta de un vehículo holonómico de seis ruedas, especialmente diseñado para sortear obstáculos manteniendo su centro de gravedad estable en todo momento.

El alcance de dicho proyecto abarca todas las disposiciones técnicas y procesos que integran el mismo, los cuales han sido descritos con exactitud a lo largo de este documento.

6.1.2. OBJETO

El objeto de esta memoria es establecer las condiciones generales y técnicas mínimas requeridas para el desarrollo del proyecto anteriormente mencionado.

En este proyecto no se establecerán condiciones económicas y legales, por ser de tipo académico, y no ser un producto que se vaya a comercializar.

6.2. CONDICIONES Y NORMAS DE CARÁCTER GENERAL

Este proyecto debe cumplir con obligatoriedad todas las condiciones y normas de carácter general por las que se rigen los proyectos industriales.

El objetivo de estas normas tiene como fin la garantía de la seguridad del autor y todos aquellos que participen en cualquier fase del desarrollo del proyecto.

Aquí se destacará la normativa común a todos los proyectos de este tipo, siendo las siguientes:

- UNE 157001:2014. "Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico".

-
- UNE-EN ISO 11442:2006. "Documentación técnica de productos. Gestión de documentos (ISO 11442:2006)".

6.3. CONDICIONES DE ESPECIFICACIONES TÉCNICAS

En lo referente a las condiciones de especificaciones técnicas, estas vienen mucho más definidas por la propia naturaleza del proyecto, correspondiéndose con el aseguramiento de la seguridad y la calidad de los procesos concretos inherentes al mismo.

6.3.1. NORMATIVA APLICABLE A LA GENERACIÓN DE PLANOS

- UNE 1027-95. "Dibujos técnicos. Plegado de planos".
- UNE 1039:1994. "Dibujos técnicos. Acotación. Principios generales, definiciones, métodos de ejecución e indicaciones especiales".
- UNE 1032:1982. "Dibujos técnicos. Principios generales de representación."

6.3.2. NORMATIVA APLICABLE A LA GESTIÓN DE LA CALIDAD

- UNE-EN ISO 8402:1995. "Gestión y aseguramiento de la calidad. (ISO 8402:1994)."
- UNE-EN ISO 10007:1997. "Gestión de la calidad. Directrices para la gestión de la configuración. (ISO 10007:1995)."
- UNE-EN 45020:1998. "Normalización y actividades relacionadas."
- UNE-EN ISO 9001:2000. "Sistemas de gestión de la calidad. Requisitos."
- UNE-EN ISO 9004:2000. "Sistemas de gestión de la calidad. Directrices para la mejora del desempeño"

6.3.3. NORMATIVA APLICABLE A PROYECTOS ELÉCTRICOS

- Reglamento Electrotécnico para Baja Tensión (REBT) (Decreto 2413/1973) y sus Instrucciones Técnicas Complementarias (MITB).

6.3.4. NORMATIVA SOBRE FABRICACIÓN Y COMPONENTES

- UNE-EN ISO 16090-1:2017. "Seguridad de las máquinas herramienta. Centros de mecanizado, centros de fresado, máquinas transfer. Parte 1: Requisitos de seguridad".
- UNE-EN ISO 888:2019. "Elementos de fijación. Pernos, tornillos y espárragos. Longitudes nominales y longitudes roscadas".
- UNE-EN ISO 898-1:2015. "Características mecánicas de los elementos de fijación de acero al carbono y de acero aleado. Parte 1: Pernos, tornillos y bulones con clases de calidad especificadas. Rosca de paso grueso y rosca de paso fino".
- UNE-EN ISO/ASTM 52910:2020 "Fabricación aditiva. Diseño. Requisitos, directrices y recomendaciones (ISO/ASTM 52910:2018)".

7. PRESUPUESTO

En este capítulo se describirán y detallarán los costes de la realización del proyecto presente.

Se deberán tener en cuenta las siguientes consideraciones:

- Tanto la elaboración del proyecto como la fabricación del mismo serán llevados a cabo por una sola persona.
- Se elaborarán dos presupuestos finales, uno de ellos incluyendo únicamente los costes materiales, y otro incluyendo costes repercutidos de personal, fabricación, instalaciones, software y licencias.

7.1. DESGLOSE DE COSTES MATERIALES

Es el coste derivado de los medios directos necesarios para llevar a cabo la fabricación manual del producto en sí mismo.

7.1.1. MATERIALES

ITEM	Coste Unitario (€)	Unidades	Coste (€)
Motor DC12V/100RPM	4,76	6	28,56
Servomotor MG996R	4,77	6	28,62
HC-06 Módulo Bluetooth	11,01	1	11,01
Driver Motor DC L298N	3,9	3	11,7
Driver Servomotores PCA9685	8,99	1	8,99
Cable Speaker 2x0,75 mm ²	6,72	1	6,72
Cables Macho-Hembra 20 cm	0,05	80	4

Tubería PVC 50 cm	1,66	2	3,32
Acoplamiento Hexagonal	0,9	6	5,4
Power Bank Xiaomi 20000 mAh	23,95	1	23,95
Paquete Pilas AA	1,5	1	1,5
Batería 12V 3000 mAh	23,99	1	23,99
Elementos de Fijación (Tornillos y Tuercas)	36,45	1	36,45
PMMA en Láminas	16 €/kg	0,4 kg	6,4
Nylon en Polvo	9,7 €/kg	0,8 kg	7,76
PLA Filamento	18 €/kg	0,3 kg	5,4
Costes de Gastos de Envío			34,7
COSTE TOTAL			248,47 €

Tabla 12. Costes Materiales.

7.2. DESGLOSE DE COSTES REPERCUTIDOS

En este apartado se desglosan los costes que se repercuten al proyecto.

7.2.1. PERSONAL

En este subapartado, como hemos mencionado anteriormente, se va a tener únicamente en cuenta el trabajo del autor, por lo que únicamente deberemos calcular las horas totales de trabajo y multiplicarlas por la tarifa deseada.

Se ha dedicado al trabajo una media de dos horas diarias durante cinco días a la semana en los últimos cuatro meses, por lo que el total de horas asciende a:

- **Horas Trabajadas** = 2 horas * 5 días * 4 semanas * 4 meses = **160 horas**

Si aplicamos la tarifa media del coste horario de un ingeniero en España, que se sitúa en los 15,38€/h, obtenemos el siguiente coste de personal.

- **Coste de Personal** = 160 horas * 15,38€/hora = **2460,8 €**

7.2.2. FABRICACIÓN E INSTALACIONES

Si estimamos el salario medio anual de un técnico en 24000€, podemos deducir que tiene un coste mensual de 2000€, por lo que, si su jornada es de 40 horas semanales, podemos estimar su coste horario en aproximadamente 12,5 euros.

ITEM	Coste Horario (€/h)	Unidades (h)	Coste (€)
Sinterizado Láser	12,5	14	175
Corte Láser	12,5	1,5	18,75
Modelado por Deposición	12,5	17	212,5
COSTE TOTAL			406,25 €

Tabla 13. Costes de Fabricación.

Para el coste de instalaciones repercutiremos la parte proporcional del coste de la instalación ocupada por el proyecto, teniendo en cuenta la cantidad de proyectos similares que se pueden realizar antes de tener que reemplazar el equipamiento.

Los costes de instalaciones son los siguientes:

- Un **ordenador** como el empleado para desarrollar el proyecto tiene un precio de mercado de 2200€, y una vida útil de unas 10000 horas, por lo que las 160 horas empleadas para el trabajo representarán un coste repercutido de **35,2 €**.

Siguiendo este mismo método, calculamos los costes repercutidos de la maquinaria de fabricación:

ITEM	Coste Total (€)	Vida Útil (h)	Uso (h)	Repercutido (€)
Máquina Sinterizado Láser	20000	20000	14	14
Máquina Corte Láser	15000	30000	1,5	0,75
Máquina Modelado por Deposición	1000	5000	17	3,4

Tabla 14. Costes de Instalaciones.

Y finalmente, compilamos todos los costes de fabricación e instalaciones:

ITEM	COSTE (€)
Fabricación	406,25
Ordenador	35,2
Instalaciones	18,15
COSTE TOTAL	459,6 €

Tabla 15. Costes de Fabricación e Instalaciones.

7.2.3. SOFTWARE Y LICENCIAS

Con los gastos de software y licencias seguiremos un método similar al empleado con los gastos de instalaciones, siendo el siguiente:

- Si tenemos en cuenta que una licencia estándar anual de Solidworks cuesta 3158 €, y que aproximadamente se pueden realizar unos 12 proyectos al año

de la duración del presente, el coste repercutido de dicha licencia en el proyecto será de **236,17 €**.

- Una licencia de **Microsoft Office** anual cuesta 69 €, y que, de nuevo, se pueden realizar 12 proyectos similares a éste anualmente, el coste repercutido es de **5,75 €**.

7.3. PRESUPUESTO FINAL

Coste final del proyecto en ambas modalidades.

7.3.1. PRESUPUESTO DE COSTES MATERIALES

El presupuesto de costes materiales es de 248,47 €, es decir, lo que debería invertir una persona que lo fabrica con sus propias manos en casa.

7.3.2. PRESUPUESTO DE COSTES TOTAL

ITEM	COSTE (€)
Materiales	248,47
Personal	2460,8
Fabricación, Instalaciones y Recursos	459,6
Software y Licencias	241,92
COSTE TOTAL	3410,79 €

Tabla 16. Presupuesto de Costes Total.

8. PROPUESTAS DE MEJORA

Como en todo proyecto de ingeniería, al finalizar el mismo debemos hacer un ejercicio de análisis y reflexión, con el fin de poder detectar puntos débiles o a mejorar del producto, para mejorarlos de cara a versiones futuras.

- En primer lugar, la principal mejora aplicable a este vehículo es un control de posición, para poder definir trayectorias y que el astromóvil alcance un mayor grado de autonomía. Esto lo conseguiríamos añadiendo *encoders* a los motores y a los servomotores, y también mejoraríamos el sistema si añadiéramos un localizador *GPS* en el chasis.
- En segundo lugar, sería necesario implementar un control de velocidad como en la simulación, para que el avance del vehículo sea preciso incluso en situaciones en las que una de las ruedas tiene más resistencia al avance que el resto.
- En tercer lugar, por la necesidad del empleo de distintas fuentes de alimentación para los componentes y del uso de *drivers* para alimentar los motores, el tamaño del chasis no es suficiente para albergar toda la electrónica. Asimismo, sería óptimo planificar previo al diseño el esquema de montaje eléctrico, con el fin de diseñar una plancha con taladros y guías en la que ubicar y fijar los componentes.
- En cuarto lugar, después de observar el comportamiento del prototipo real, se ha llegado a la conclusión de que el PLA no es el material idóneo para las ruedas, por lo que en las próximas versiones se tratará de usar un material más elástico, que proporcione mayor agarre y capacidad de tracción.
- Y, por último, una vez comprobado en esta versión que el mecanismo funciona correctamente, en las próximas versiones sería adecuado dotarle de alguna funcionalidad adicional, como por ejemplo, añadirle una cámara para poder grabar o instrumentos meteorológicos para poder hacer estudios climáticos en zonas de difícil acceso.

9. CONCLUSIONES

Habiendo finalizado el presente proyecto, se pueden extraer del mismo las siguientes conclusiones:

I. El sistema de suspensiones *rocker-bogie* empleado en este tipo de vehículos es altamente eficaz para situaciones en las que deben transitar por terrenos irregulares y superar obstáculos.

II. Es de vital importancia planificar adecuadamente y con anticipación el montaje eléctrico de cualquier tipo de proyecto industrial, pero especialmente en vehículos o productos que dispongan de una envolvente para albergar dicho montaje.

En el caso de no actuar de esta forma, podemos diseñar un chasis que no tenga la capacidad de alojar la electrónica, o que no lo haga de una forma óptima.

III. El control automático es una parte indispensable en proyectos industriales, y la fiabilidad del mismo debe ser máxima, así como la exactitud y la precisión de la sensorización.

IV. El diseño de componentes en proyectos industriales debe ser adecuado a la fase del producto en la que nos encontremos. Por ejemplo, en el presente, ciertos componentes han sido modificados en dos ocasiones después de ser creados, con el fin de ser óptimos para la simulación en primer lugar, y con el de ser aptos para la fabricación por métodos aditivos después.

V. Actualmente, con la cantidad de tecnología de la que se dispone, y teniendo en cuenta la importancia de los costes en los proyectos industriales, las simulaciones se han convertido en una herramienta absolutamente indispensable.

Son un método fiable, rápido y poco costoso para realizar pruebas en la fase de validación de producto sin emplear mano de obra y materiales.

VI. En la fase de diseño del producto, los métodos de fabricación aditiva y de prototipado rápido han supuesto un avance notable.

Gracias a estas técnicas, el coste de maquinaria es menor, la libertad en el diseño es mayor, y la variabilidad ha llegado a niveles nunca vistos.

VII. Los costes materiales suponen un bajo porcentaje de los costes totales de un proyecto, ya que la mayoría del coste viene derivado por acciones o elementos que no son tangibles para el usuario o cliente final.

Esto puede hacer que el cliente perciba que se le está cobrando un precio muy por encima del adecuado en cierto tipo de proyectos, por lo que es necesario ejecutar un buen presupuesto que pueda servir de justificación del coste ante posibles consultas.

10. BIBLIOGRAFÍA

- Wikipedia contributors. (2023). Curiosity (rover). Wikipedia. [https://en.wikipedia.org/wiki/Curiosity_\(rover\)](https://en.wikipedia.org/wiki/Curiosity_(rover))
- Wikipedia contributors. (2023b). Radioisotope thermoelectric generator. Wikipedia. https://en.wikipedia.org/wiki/Radioisotope_thermoelectric_generator
- Mars.Nasa.Gov. (s. f.). Where is Curiosity? NASA Mars Exploration. <https://mars.nasa.gov/msl/mission/where-is-the-rover/>
- Colaboradores de Wikipedia. (2021). Rocker-bogie. Wikipedia, la enciclopedia libre. <https://es.wikipedia.org/wiki/Rocker-bogie>
- What are the disadvantages of the rocker bogie suspension system? (s. f.). Quora. <https://www.quora.com/What-are-the-disadvantages-of-the-rocker-bogie-suspension-system>
- Colaboradores de Wikipedia. (2019). Odometría. Wikipedia, la enciclopedia libre. <https://es.wikipedia.org/wiki/Odometr%C3%ADa>
- Kinematic modeling of a high mobility Mars rover. (1999). IEEE Conference Publication | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/772441>
- Colaboradores de Wikipedia. (2023). Controlador PID. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/Controlador_PID
- Colaboradores de Wikipedia. (2023a). Sensor. Wikipedia, la enciclopedia libre. <https://es.wikipedia.org/wiki/Sensor>
- CodeCogs, Z. L. (s. f.). Equation Editor for online mathematics - create, integrate and download. Copyright (C) 2004-2023 CodeCogs, Zyba Ltd. <https://editor.codecogs.com/>
- Kamil Kosi. (2017, 7 octubre). 3D Printed Mars Rover (Based on NASA's Curiosity) - STL Files in description [Video]. YouTube. <https://www.youtube.com/watch?v=PJvhMdysS9s>

-
- RC scaled MARS Rover (advanced). (s. f.). Gumroad.
<https://kamilkosi.gumroad.com//uuKCw>
 - Colaboradores de Wikipedia. (2022). Sinterizado selectivo por láser. Wikipedia, la enciclopedia libre.
https://es.wikipedia.org/wiki/Sinterizado_selectivo_por_l%C3%A1ser
 - Colaboradores de Wikipedia. (2023b). Corte con láser. Wikipedia, la enciclopedia libre. https://es.wikipedia.org/wiki/Corte_con_l%C3%A1ser
 - Colaboradores de Wikipedia. (2022b). Modelado por deposición fundida. Wikipedia, la enciclopedia libre.
https://es.wikipedia.org/wiki/Modelado_por_deposici%C3%B3n_fundida
 - Model contact between two geometries - MATLAB - MathWorks España. (s. f.).
<https://es.mathworks.com/help/sm/ref/spatialcontactforce.html>
 - SolidBi. (2023, 8 febrero). Precio SOLIDWORKS - SolidBI. SolidBI.
<https://solid-bi.es/precio-solidworks/>
 - Llamas, L. (2016). Controlar 16 servos o 16 salidas PWM en Arduino con PCA9685. Luis Llamas.
<https://www.luisllamas.es/controlar-16-servos-o-16-salidas-pwm-en-arduino-con-pca9685/>
 - Llamas, L. (2016a). Controlar motores de corriente continua con Arduino y L298N. Luis Llamas. <https://www.luisllamas.es/arduino-motor-corriente-continua-l298n/>
 - Llamas, L. (2015). Conectar Arduino por Bluetooth con los módulos HC-05 ó HC-06. Luis Llamas.
<https://www.luisllamas.es/conectar-arduino-por-bluetooth-con-los-modulos-hc-05-o-hc-06/>
 - Tutorial Módulo Controlador de servos PCA9685 con Arduino. (s. f.). Naylamp Mechatronics - Perú.

https://naylampmechatronics.com/blog/41_tutorial-modulo-controlador-de-servos-pca9685-con-arduino.html

- INNOVA DOMOTICS. (2021, 16 diciembre). 18/1 Curso Arduino - Control On Off desde App con HC05 y Bluetooth Electronics - Robótica [Vídeo]. YouTube. <https://www.youtube.com/watch?v=HUeiWE5CHs>

ÍNDICE DE FIGURAS Y TABLAS

Figura 1. Rover Curiosity.....	11
Figura 2. Cráter de Gale, lugar de aterrizaje del Curiosity.....	12
Figura 3. Recorrido y ubicación actual del Curiosity.....	13
Figura 4. Dos ingenieros del JPL junto al Curiosity.....	13
Figura 5. Esquema de instrumentos científicos del Curiosity.....	14
Figura 6. Esquema de un Sistema Rocker-Bogie.....	15
Figura 7. Mecanismo Rocker-Bogie en un Terreno Irregular.....	16
Figura 8. Variables de la Cinemática del Vehículo.....	17
Figura 9. Planteamiento Cinemática Directa del Vehículo.....	18
Figura 10. Expresiones Cinemática Directa del Vehículo.....	19
Figura 11. Cinemática Inversa del Vehículo.....	19
Figura 12. Variación de Posición Respecto al Tiempo.....	20
Figura 13. Odometría del Vehículo.....	20
Figura 14. Vehículo en Posición de Desplazamiento Frontal.....	21
Figura 15. Vehículo en Posición de Desplazamiento Lateral.....	22
Figuras 16(a y 16(b). Vehículo en Posición de Girar a Izquierda y Derecha.....	23
Figura 17. Vehículo en Posición de Giro Sobre el Eje Vertical.....	24
Figura 18. Bucle de Control Automático.....	25
Figura 19. Representación de un Proceso Industrial.....	25
Figura 20. Sensor de Efusividad Térmica.....	26
Figura 21. Acción Proporcional.....	28
Figura 22. Acción Integral.....	28
Figura 23. Acción Derivada.....	28
Figura 24. Ensamblaje Completo del Vehículo.....	29
Tabla 1. Relación de Archivos STL.....	30
Tabla 2. Relación de Piezas Adicionales.....	31
Figura 25. Cuerpo Central.....	31
Tabla 3. Listado de Piezas del Cuerpo Central.....	32
Figura 26. Chasis.....	32
Figura 27. Elemento de Construcción.....	33

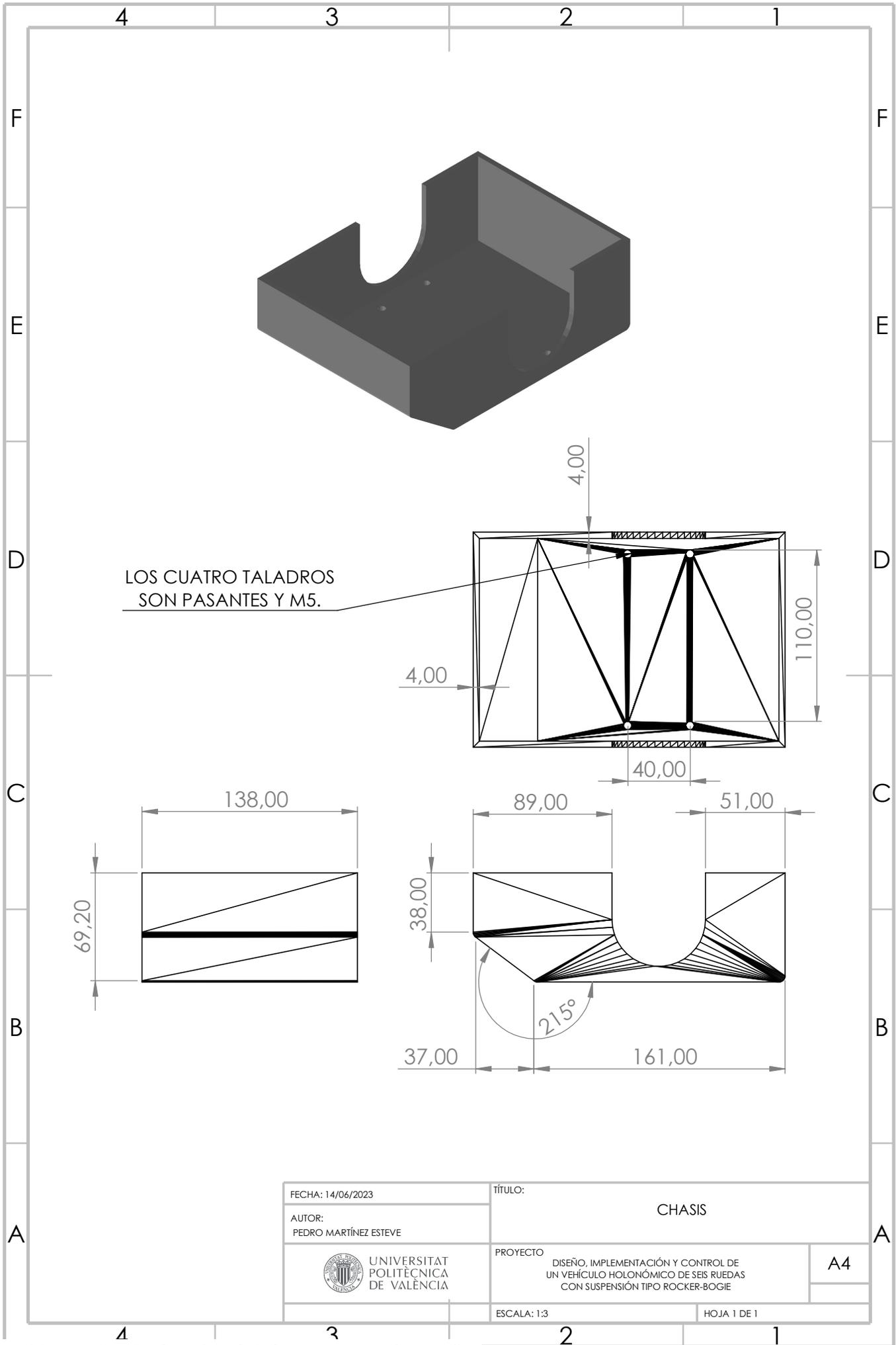
Figura 28. Elemento de Construcción 2.....	34
Figura 29. Sujeción Rodamiento Principal.....	34
Figura 30. Suspensión Lateral Izquierda.....	35
Tabla 4. Listado de Piezas de las Suspensiones Laterales.....	36
Figura 31. Tubo PVC 180x25x21	36
Figura 32. Servomotor.....	37
Figura 33. Sujeción Suspensión.....	38
Figura 34. Enlace Suspensión-Construcción.....	38
Figuras 35(a) , 35(b) y 35(c). Acoplamiento Rueda Delantera.....	39
Figura 36. Acoplamiento Rueda Trasera.....	40
Figura 37. Alojamiento Rodamiento Trasero.....	40
Figura 38. Acoplamiento Rodamiento Trasero.....	41
Figura 39. Rueda y Arco.....	42
Tabla 5. Listado de Piezas de Ruedas y Arco.....	42
Figura 40. Alojamiento Motor.....	43
Figura 41. Motor.....	44
Figura 42. Arco Estructural.....	44
Figura 43. Acoplamiento Rodamiento Servo.....	45
Figura 44. Rueda.....	46
Figura 45. Acople Interior Rueda.....	47
Figura 46. Comparativa Rueda Prototipo vs Rueda Simulación.....	50
Tabla 6. Relación de Subconjuntos Para Simulación.....	52
Figura 47. Bloque File Solid.....	53
Figura 48. Bloque Revolute Joint.....	53
Figura 49. Model Properties.....	54
Figura 50. Subsistema y Puertos de Conexión.....	55
Tabla 7. Relaciones de Piezas en Ensamblaje.....	56
Figura 51. Ensamblaje en Simscape Multibody.....	57
Figura 52. Entorno de Simulación en Simscape Multibody.....	59
Figuras 53(a) y 53(b). Creación de Obstáculos en Simscape Multibody y Subsistema "Entorno".....	59
Figuras 54(a) y 54(b). Bloques "Brick Solid" y "Extruded Solid"	60
Figura 55. Vehículo Superando Escalones Frontales.....	60

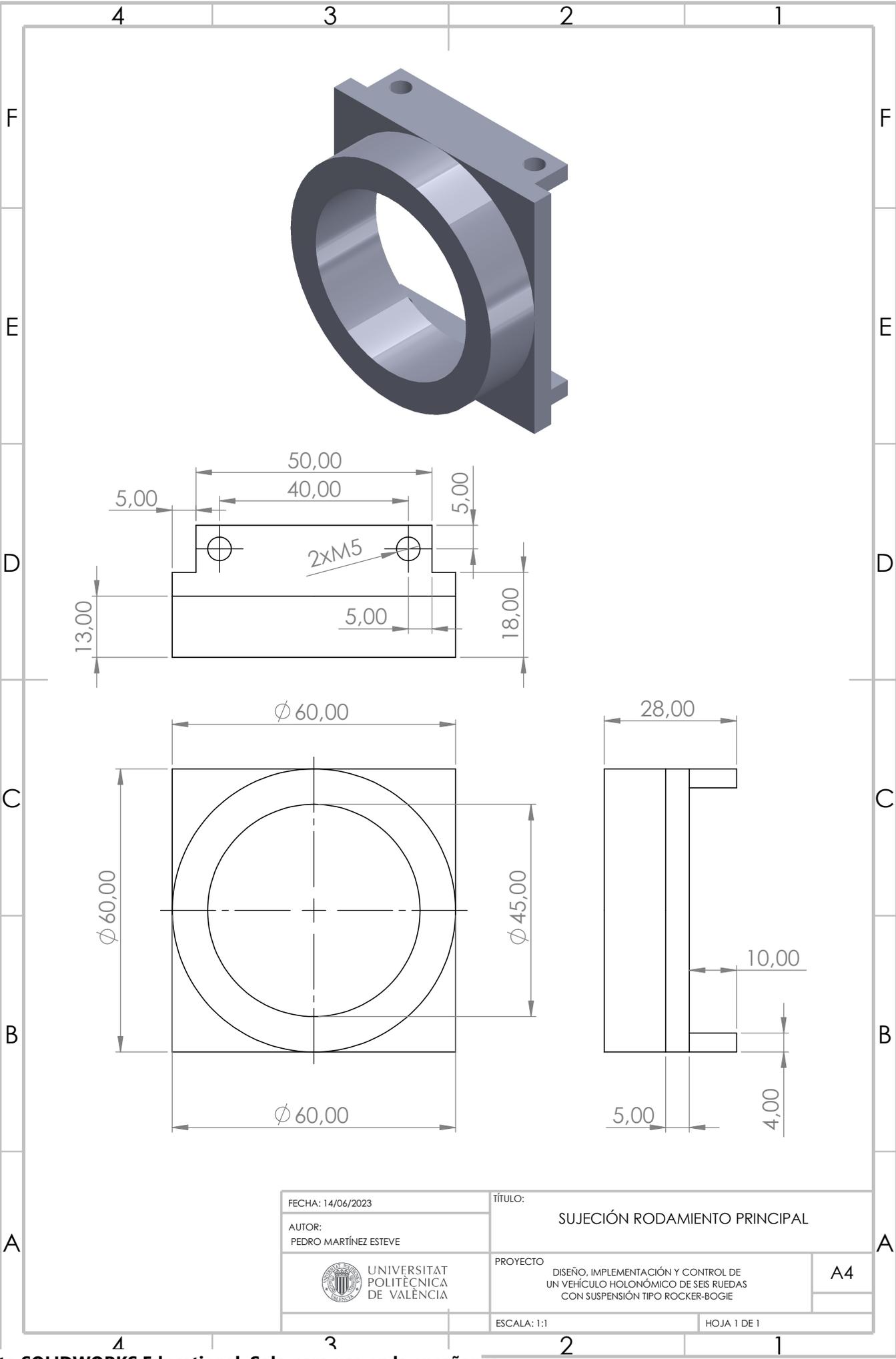
Figura 56. Vehículo Superando Escalones Laterales.....	61
Figura 57. Vehículo Superando Rampa Lateral.....	62
Figura 58. Vehículo Superando Rampa Frontal.....	63
Figura 59. 6-DOF Joint.....	64
Figura 60. Cartesian Joint y Gimbal Joint.....	65
Figura 61. Rigid Transform.....	65
Figura 62. Spatial Contact Force.....	66
Figura 63. Subsistema "Contact Forces".....	66
Figura 64. Coeficientes de las "Spatial Contact Forces".....	67
Figura 65. Cartesian Joint con Sensorización de Posición en X e Y.....	69
Figura 66. Gimbal Joint con Sensorización de Rotación Respecto a Y.....	70
Figura 67. Revolute Joint con Sensorización de Velocidad.....	70
Figura 68. Trayectoria Deseada del Vehículo.....	71
Tabla 8. Descripción de Acciones de Trayectoria y Relación con Tiempos.....	72
Figura 69. Sistema de Señales de los Servomotores.....	73
Figura 70. Gráfica de las Señales de los Servomotores.....	74
Tabla 9. Descripción de Señales de Servomotores y Relación con Tiempos.....	76
Figura 71. Subsistema de Señales de los Motores.....	77
Figura 72. Gráfica de las Señales de los Servomotores.....	77
Tabla 10. Descripción de Señales de Motores y Relación con Tiempos.....	78
Figura 73. Bucle de Control de una de las Ruedas.....	79
Figura 74. Controlador sin Valores Definidos.....	80
Figura 75. Primera Prueba ($K_p = 0,001$; $K_i = 0,025$).....	81
Figura 76. Segunda Prueba ($K_p = 0,01$; $K_i = 0,05$).....	82
Figura 77. Tercera Prueba ($K_p = 0,03$; $K_i = 0,08$).....	82
Figura 78. Controlador Final.....	82
Figura 79. Velocidad ante Referencia con el Controlador Final.....	83
Figura 80. Señal vs Referencia del Servomotor Delantero Izquierdo.....	84
Figura 81. Extracción de Datos de Posición en Vectores.....	84
Figura 82. Código de Generación de Figura en MATLAB.....	85
Figura 83. Trayectoria del Vehículo.....	85
Figura 84. Ángulo de Cabeceo.....	86
Figura 85. Limpieza de una Pieza Fabricada por SLS.....	87

Figura 86. Máquina de Corte Láser en Acción.....	88
Figura 87. Máquina de FDM en Proceso de Fabricación.....	89
Figura 88. Diseño Original del Chasis.....	90
Figura 89. Despiece del Chasis.....	90
Figura 90. Chasis Completo Fabricado por LBC.....	91
Figura 91. Diseño Original del Acople Interior de la Rueda.....	91
Figura 92. Mitades del Acople Interior de la Rueda Fabricadas con LBC.....	92
Figura 93. Diseño Original del Arco Estructural.....	92
Figura 94. Arco Estructural Fabricado por SLS.....	93
Tabla 11. Relación de Piezas y Procesos de Fabricación.....	94
Figura 95. Despiece Ensamblaje Rueda.....	94
Figura 96. Ensamblaje Rueda.....	95
Figura 97. Despiece Ensamblaje Arco Estructural.....	95
Figura 98. Ensamblaje Arco Estructural.....	96
Figura 99. Despiece Ensamblaje Arco Estructural y Motor.....	96
Figura 100. Ensamblaje Arco Estructural y Motor.....	97
Figuras 101(a , 101(b y 101(c. Despiece Ensamblaje Servomotor y Acoplamiento Rueda.....	97
Figura 102. Despiece Ensamblajes Estructurales de la Rueda y Motores.....	98
Figura 103. Ensamblajes Estructurales de la Rueda y Motores.....	98
Figura 104. Despiece Ensamblaje Sujeción Suspensión.....	99
Figuras 105(a y 105(b. Ensamblaje Sujeción Suspensión Alzado y Planta.....	99
Figura 106. Ensamblaje Chasis y Sujeciones Suspensión.....	100
Figura 107. Tubos de PVC.....	100
Figura 108. Esquema de Montaje Sistemas Rocker-Bogie.....	100
Figura 109. Acoplamiento Rueda Trasera.....	101
Figura 110. Ensamblaje Bogie.....	101
Figura 111. Ensamblaje Rocker-Bogie Completo.....	102
Figuras 112(a y 112(b. Mecanismo Completo.....	102
Figura 113. Arduino Mega 2560 Rev3.....	103
Figura 114. Servomotor MG996R.....	104
Figura 115. Motor DC 12V 100RPM.....	104
Figura 116. Módulo Bluetooth HC-06.....	105

Figura 117. Conexiones Pines HC-06.....	105
Figura 118. Conexión HC-06 con Arduino.....	106
Figura 119. Driver L298N.....	106
Figura 120. Conexiones Driver L298N.....	107
Figura 121. Conexiones Driver L298N con Arduino.....	107
Figura 122. Driver PCA9685.....	108
Figura 123. Conexiones Driver PCA9685.....	108
Figura 124. Conexiones Driver PCA9685 con Arduino.....	109
Figura 125. Interfaz Bluetooth Electronics.....	110
Tabla 12. Costes Materiales.....	115
Tabla 13. Costes de Fabricación.....	116
Tabla 14. Costes de Instalaciones.....	117
Tabla 15. Costes de Fabricación e Instalaciones.....	117
Tabla 16. Presupuesto de Costes Total.....	118

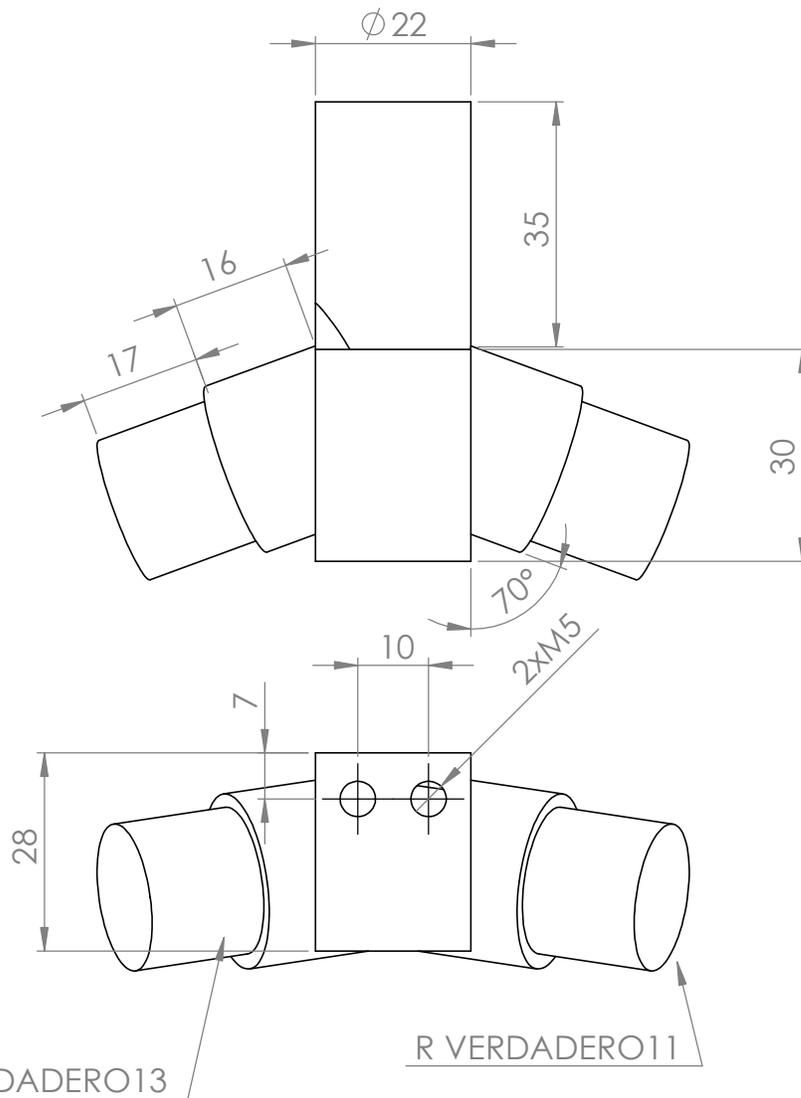
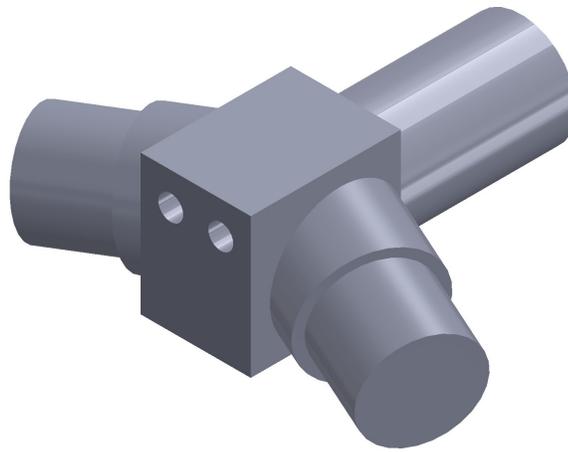
ANEXO 1: PLANOS



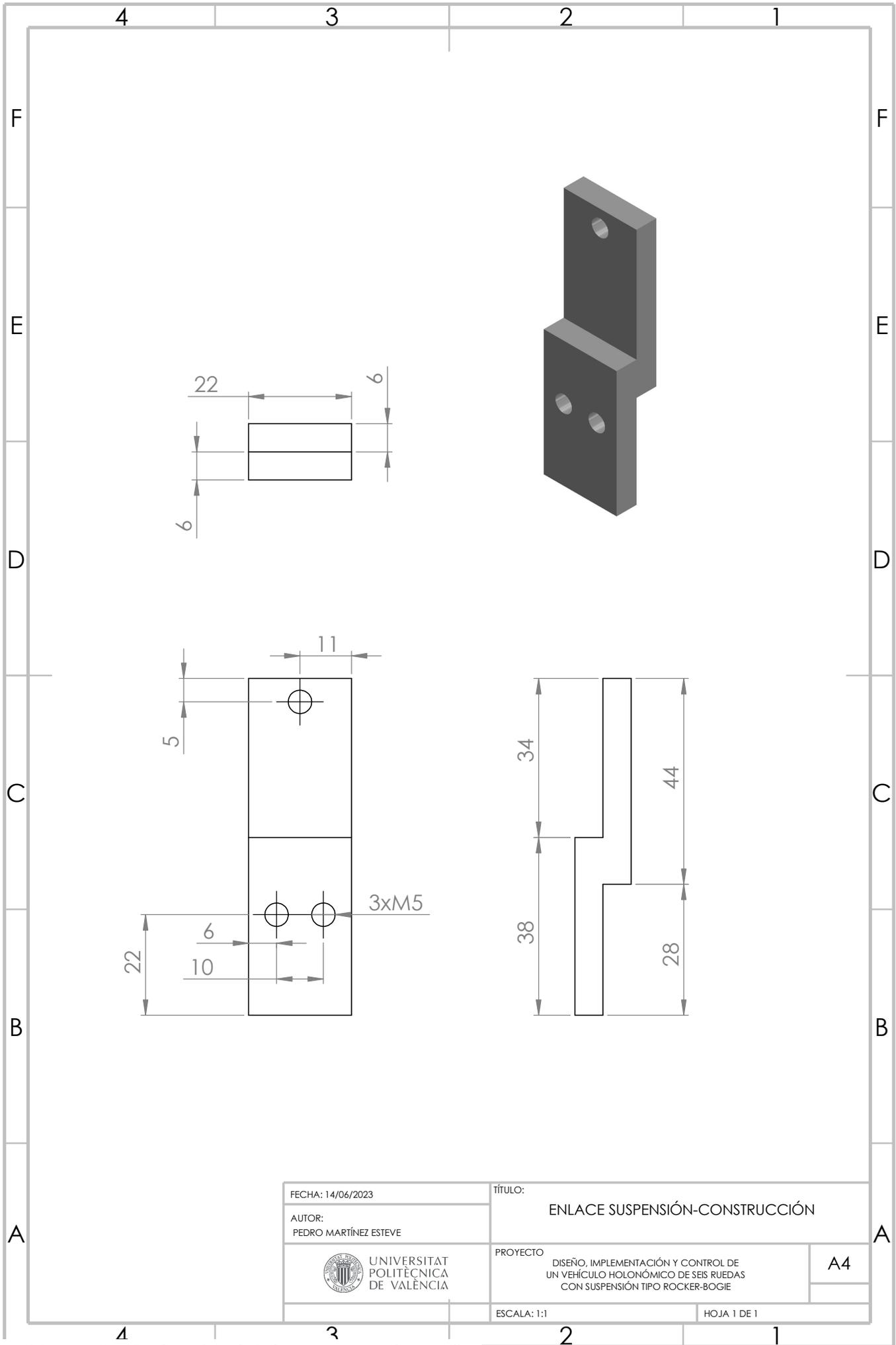


FECHA: 14/06/2023	TÍTULO:
AUTOR: PEDRO MARTÍNEZ ESTEVE	SUJECIÓN RODAMIENTO PRINCIPAL
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO
	DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE
ESCALA: 1:1	HOJA 1 DE 1

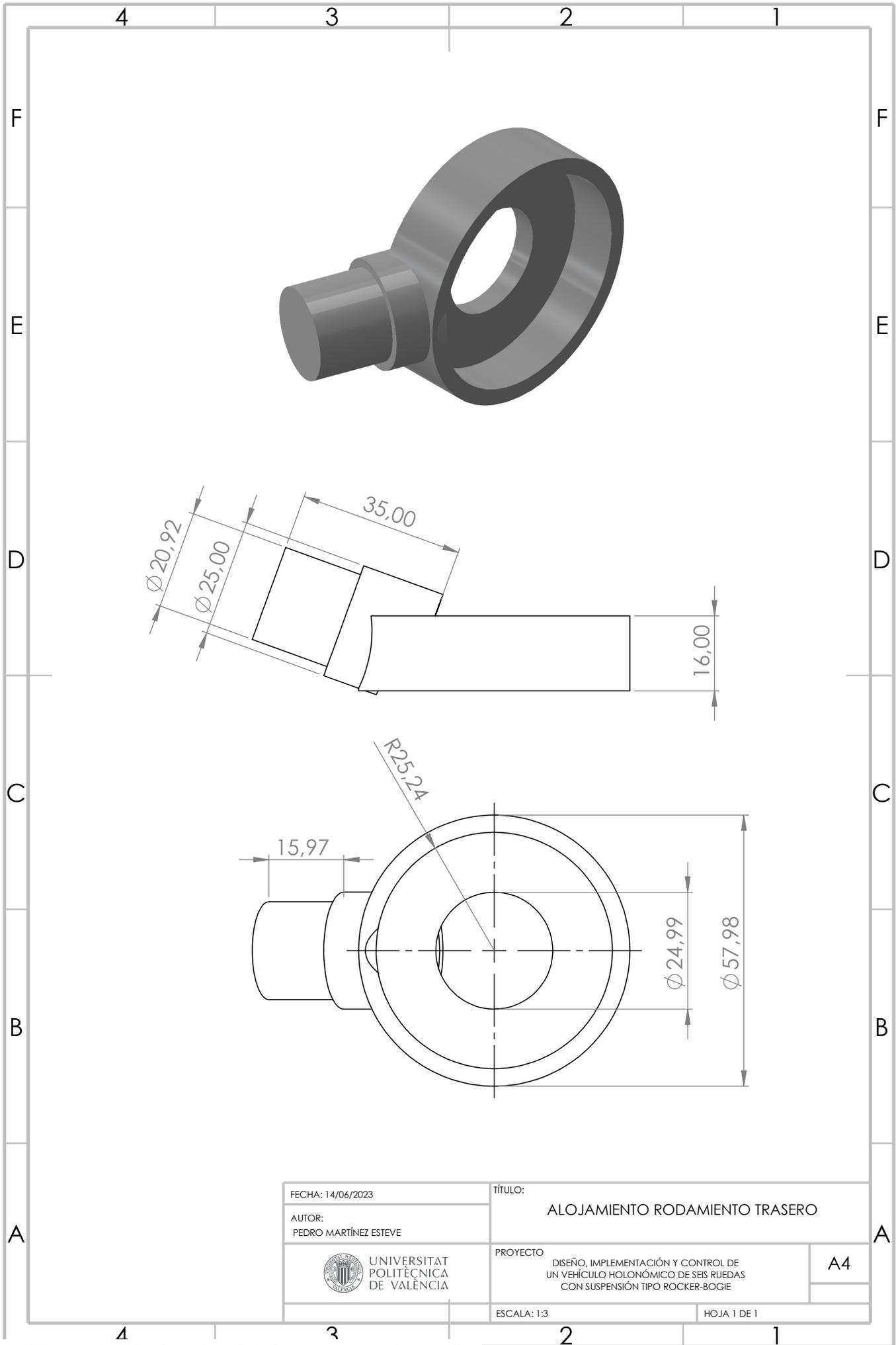
A4	
----	--



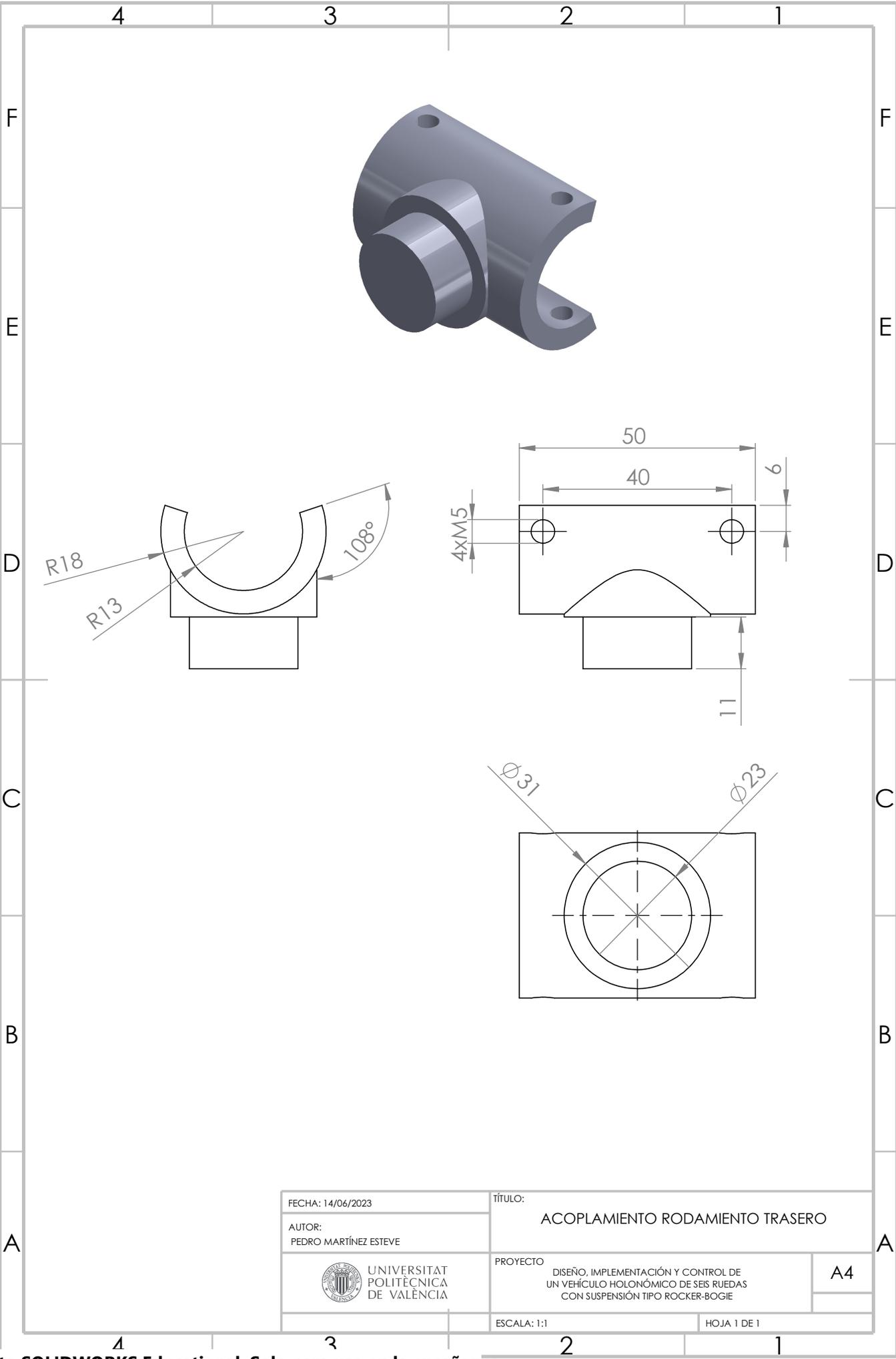
FECHA: 14/06/2023	TÍTULO:	SUJECIÓN SUSPENSIÓN	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO	DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:1	HOJA 1 DE 1	A4



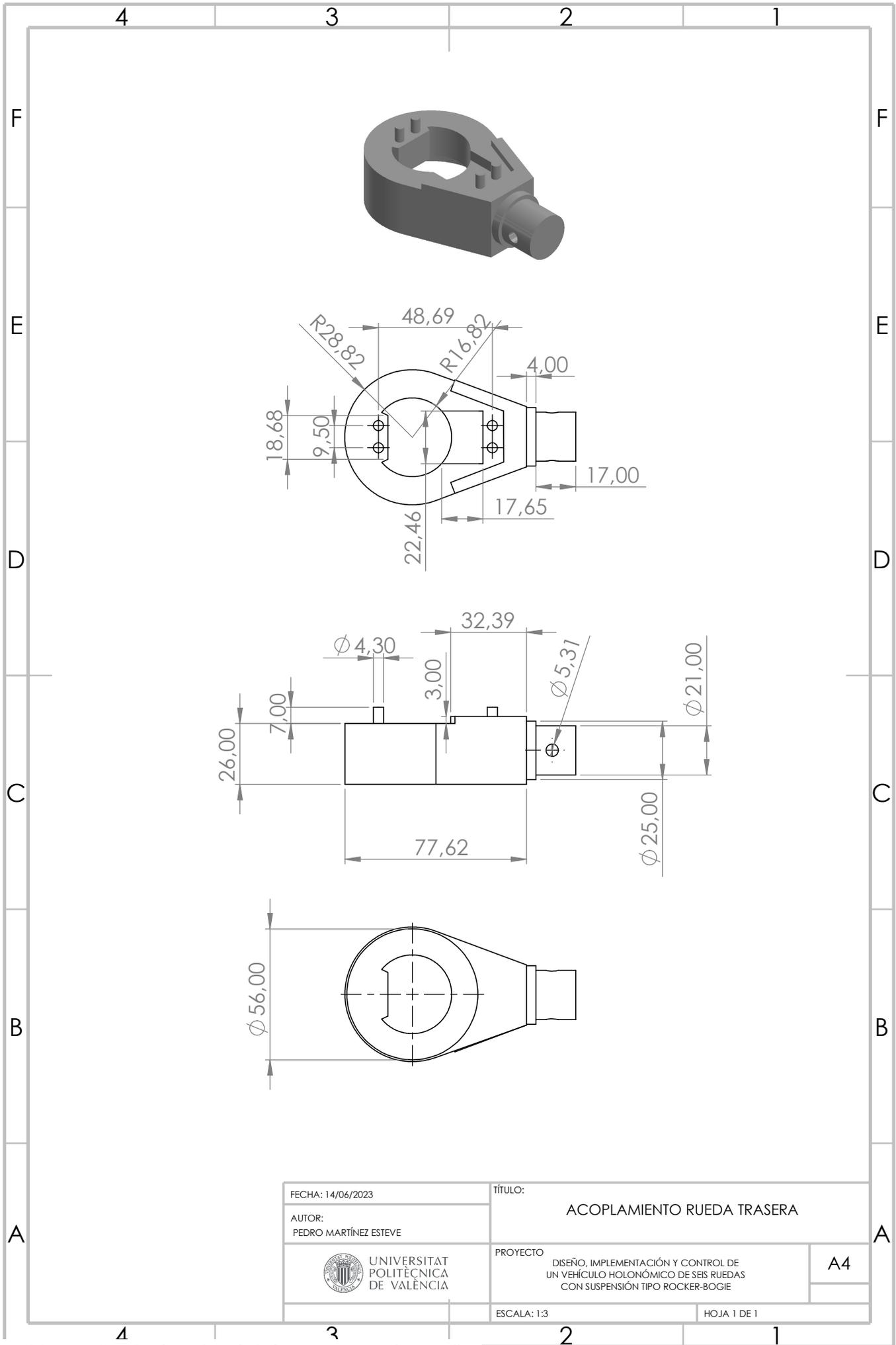
FECHA: 14/06/2023		TÍTULO:	
AUTOR: PEDRO MARTÍNEZ ESTEVE		ENLACE SUSPENSIÓN-CONSTRUCCIÓN	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO		A4
	DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE		
ESCALA: 1:1		HOJA 1 DE 1	



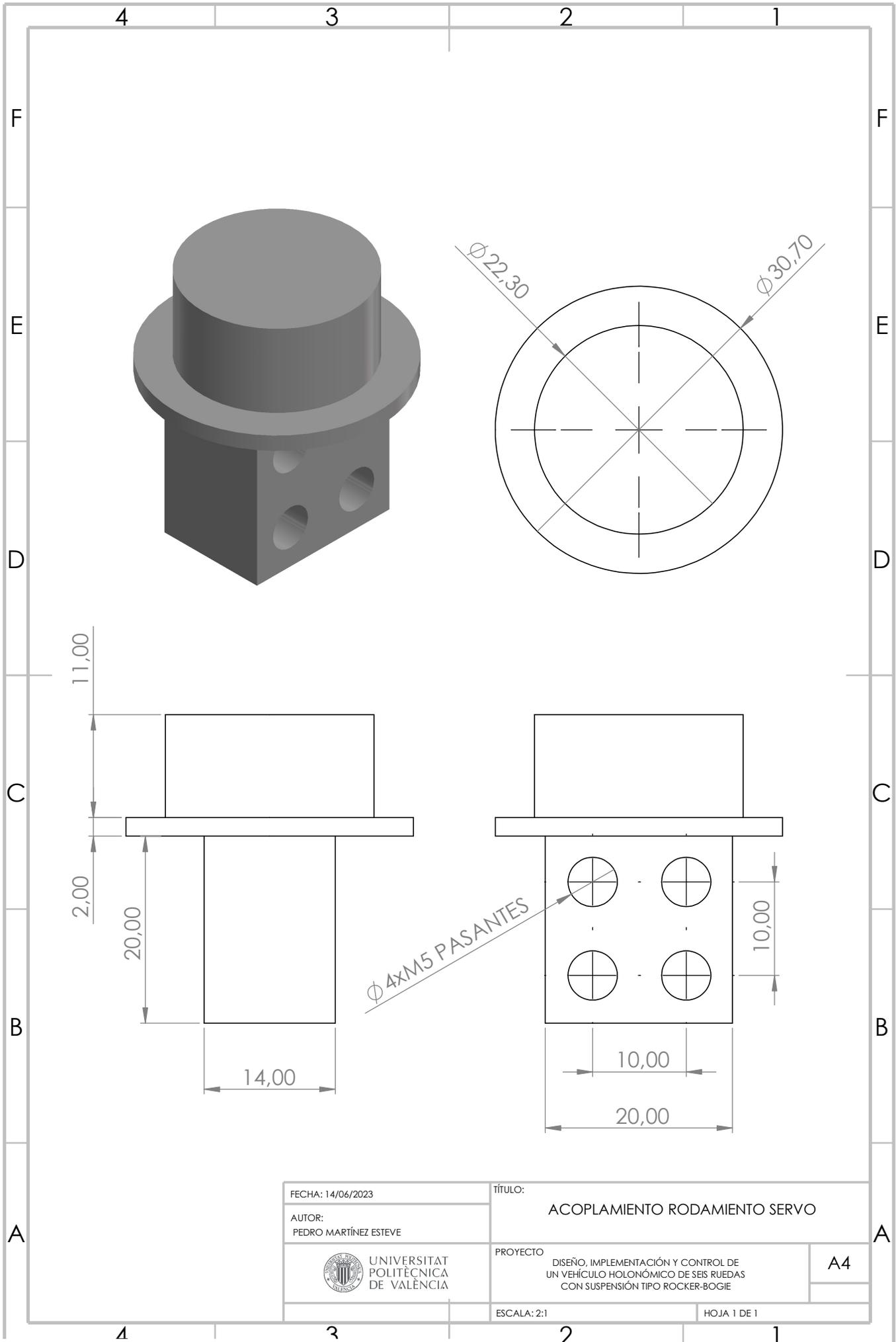
FECHA: 14/06/2023	TÍTULO: ALOJAMIENTO RODAMIENTO TRASERO	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:3	HOJA 1 DE 1



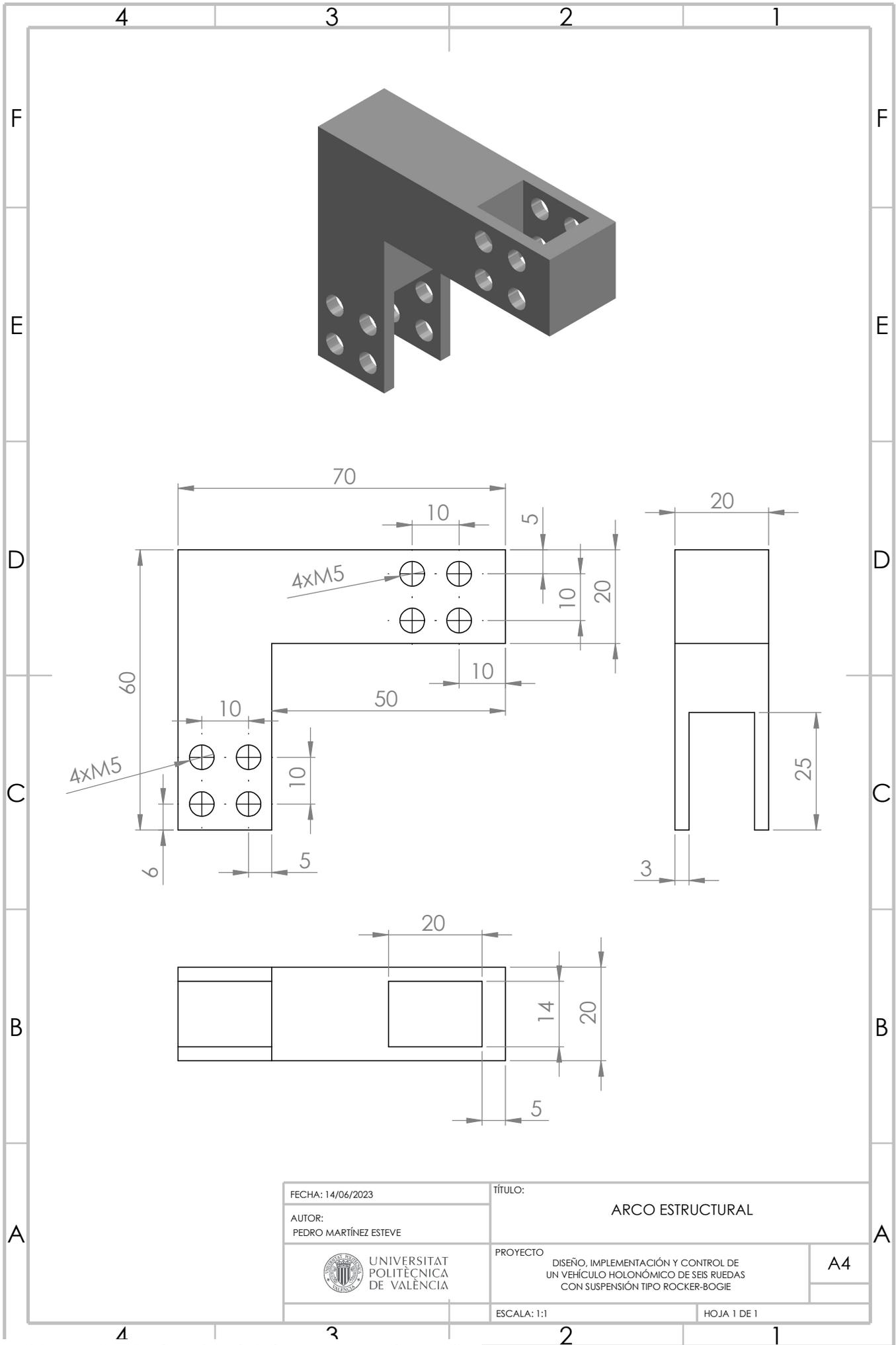
FECHA: 14/06/2023		TÍTULO:	
AUTOR: PEDRO MARTÍNEZ ESTEVE		ACOPLAMIENTO RODAMIENTO TRASERO	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO	A4	
	DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE		
ESCALA: 1:1		HOJA 1 DE 1	



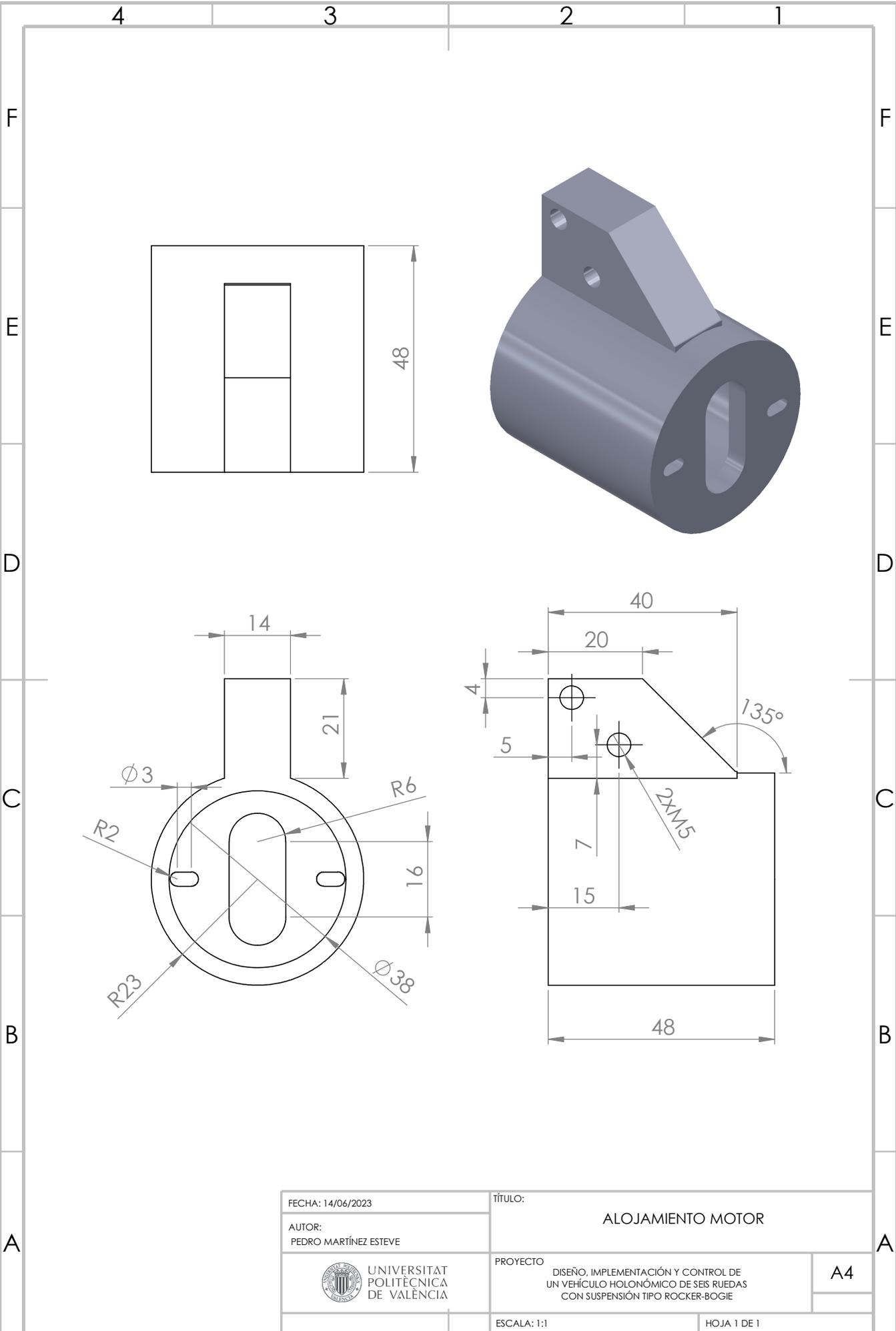
FECHA: 14/06/2023	TÍTULO: ACOPLAMIENTO RUEDA TRASERA	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:3	A4 HOJA 1 DE 1



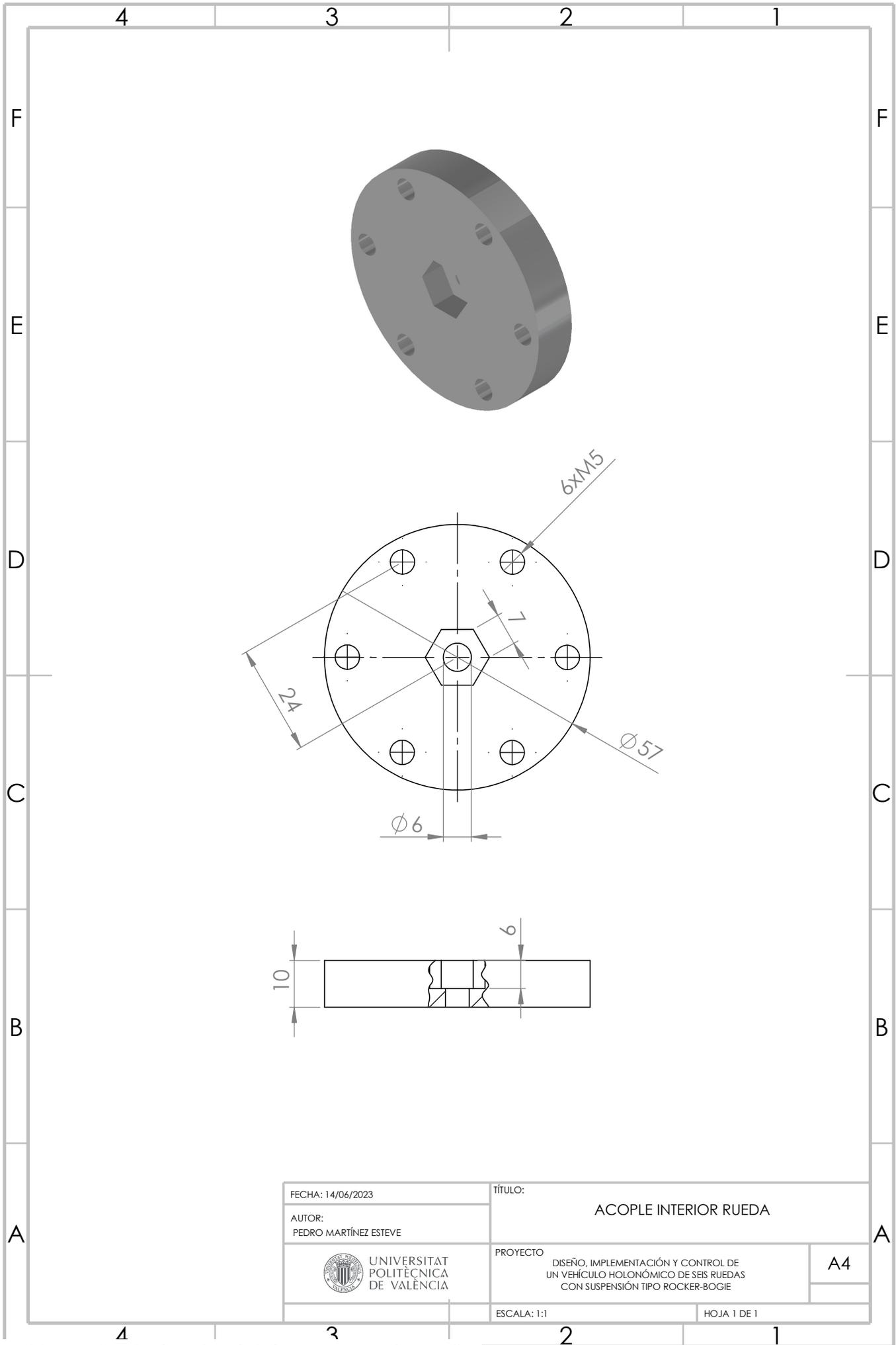
FECHA: 14/06/2023		TÍTULO: ACOPLAMIENTO RODAMIENTO SERVO	
AUTOR: PEDRO MARTÍNEZ ESTEVE		PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA		ESCALA: 2:1	A4
		HOJA 1 DE 1	



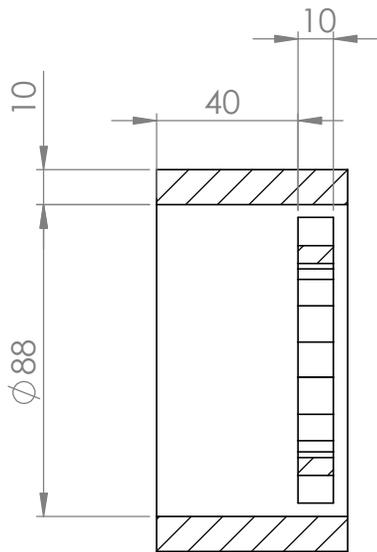
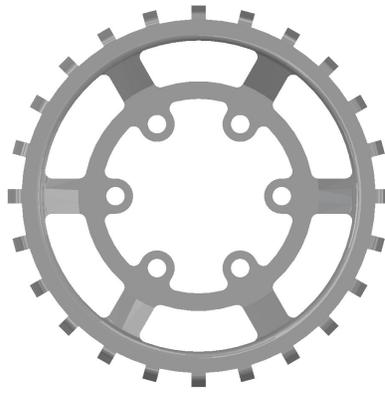
FECHA: 14/06/2023	TÍTULO: ARCO ESTRUCTURAL	
AUTOR: PEDRO MARTÍNEZ ESTEVE		
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
ESCALA: 1:1	HOJA 1 DE 1	



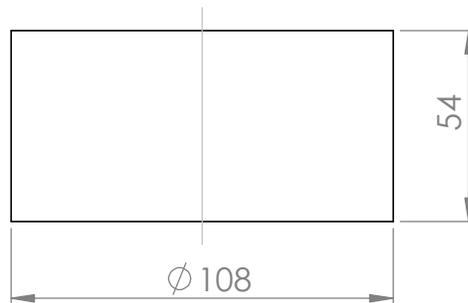
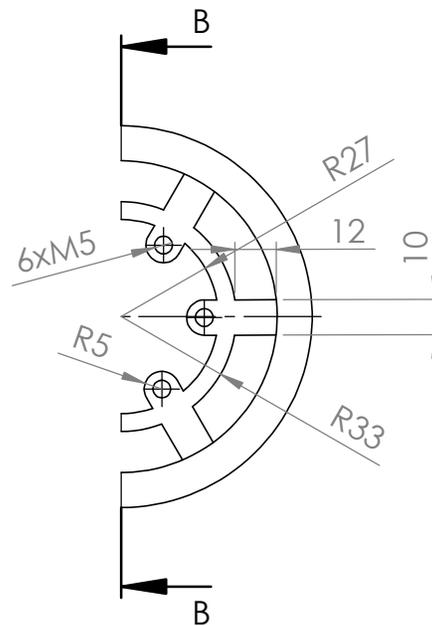
FECHA: 14/06/2023	TÍTULO: ALOJAMIENTO MOTOR	
AUTOR: PEDRO MARTÍNEZ ESTEVE		
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
ESCALA: 1:1	HOJA 1 DE 1	



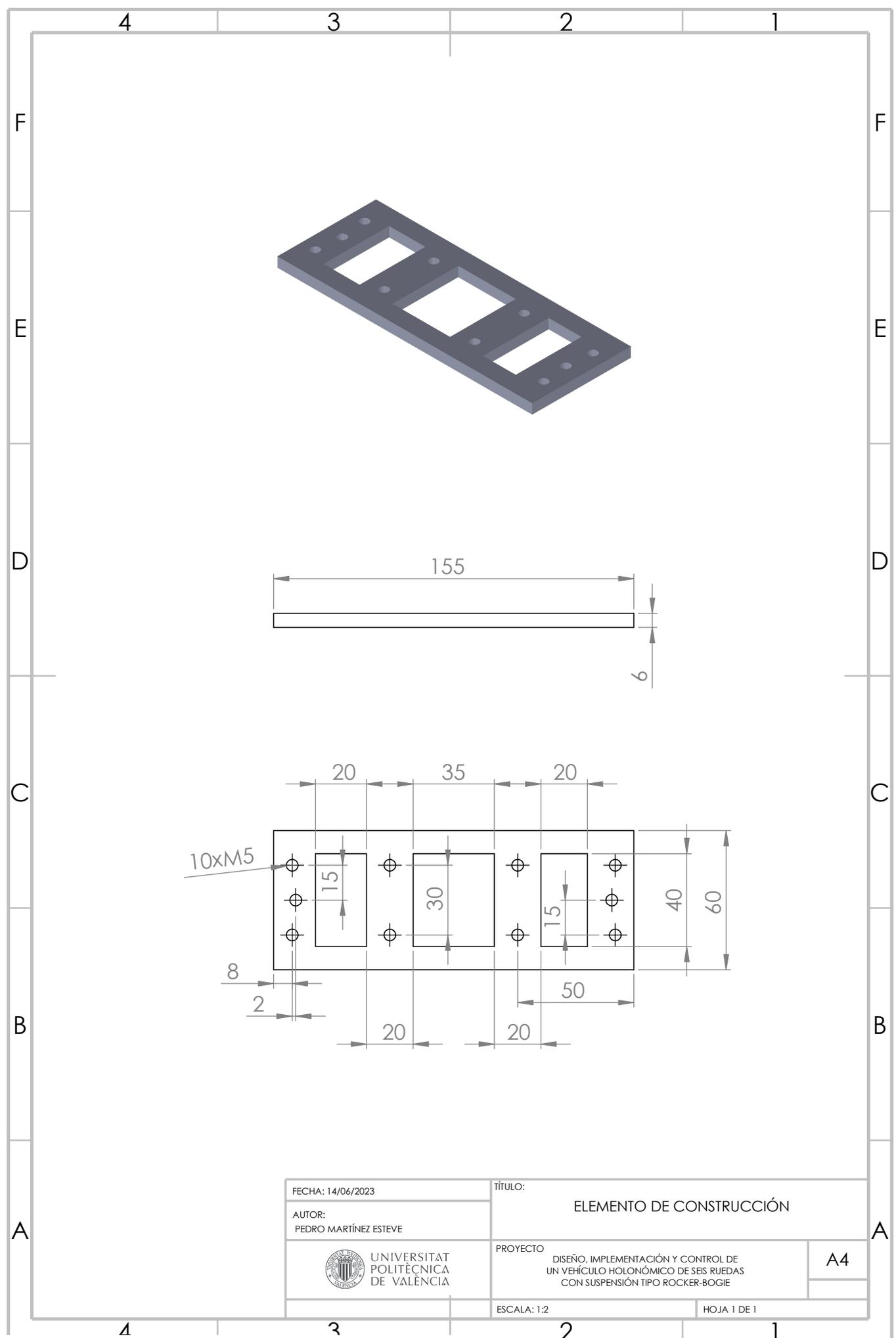
FECHA: 14/06/2023	TÍTULO: ACOPLE INTERIOR RUEDA	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:1	A4 HOJA 1 DE 1



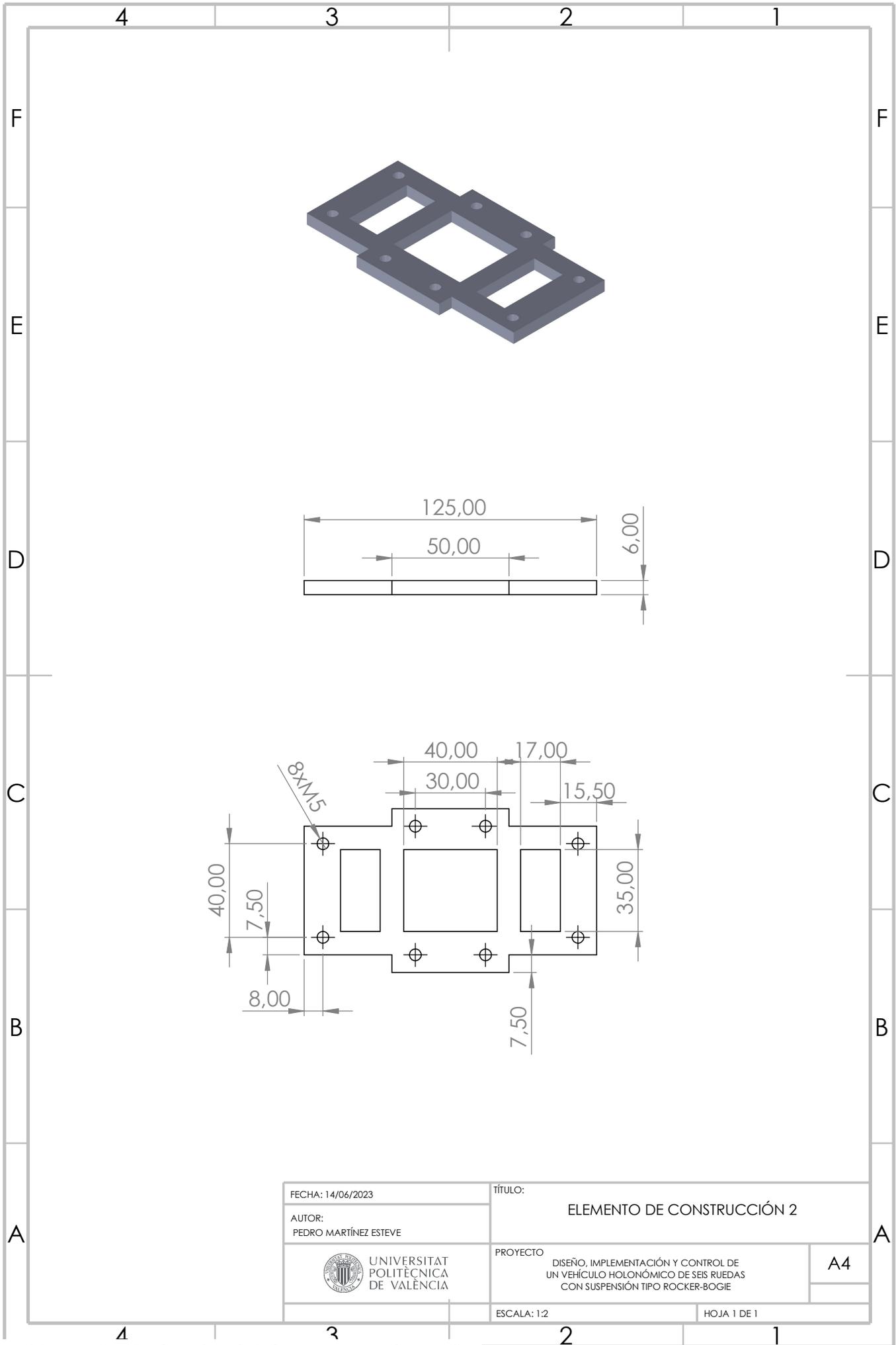
SECCIÓN B-B



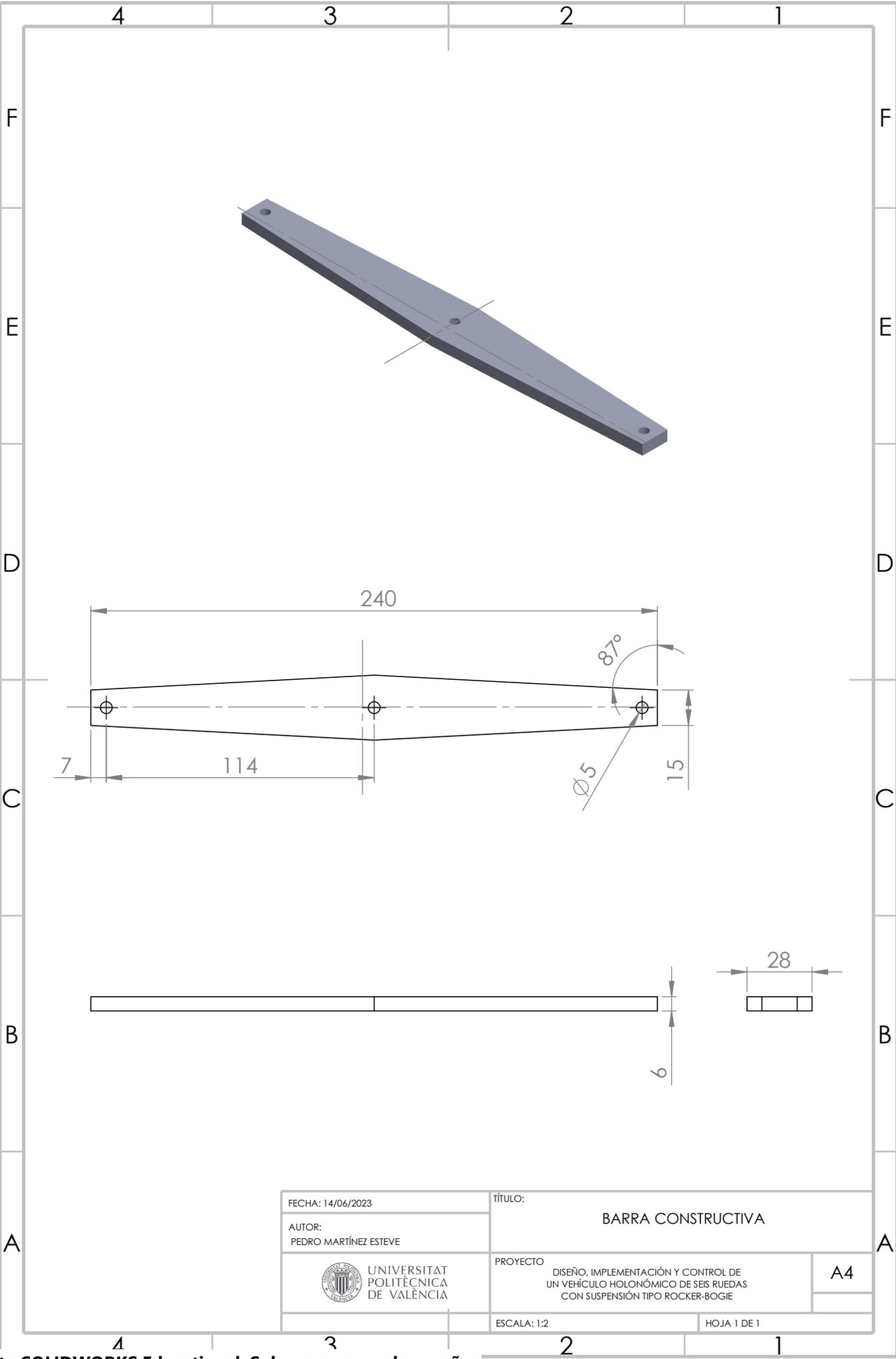
FECHA: 14/06/2023	TÍTULO: RUEDA	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:2	A4
	HOJA 1 DE 1	



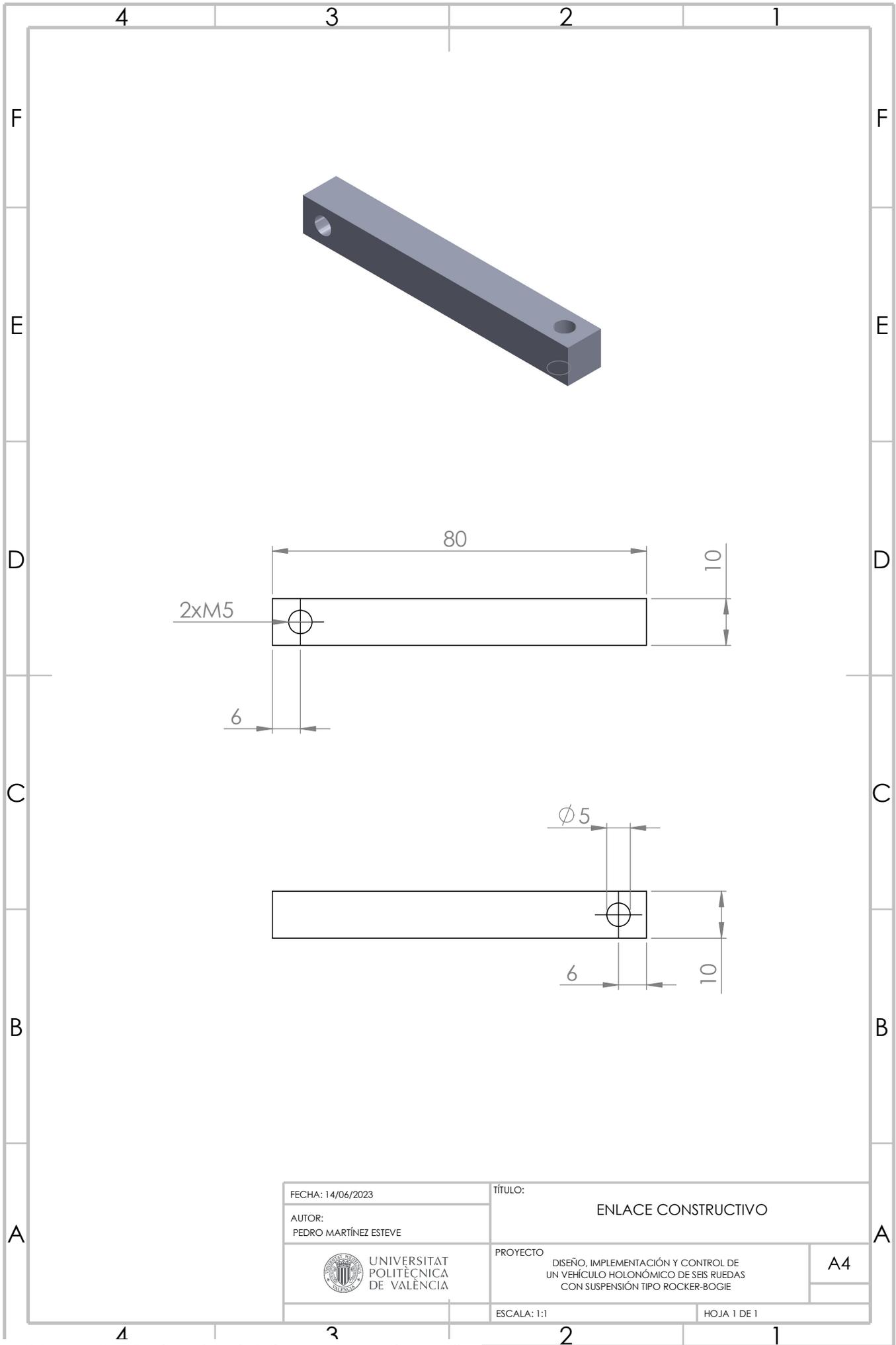
FECHA: 14/06/2023	TÍTULO: ELEMENTO DE CONSTRUCCIÓN	
AUTOR: PEDRO MARTÍNEZ ESTEVE		
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
ESCALA: 1:2	HOJA 1 DE 1	



FECHA: 14/06/2023	TÍTULO: ELEMENTO DE CONSTRUCCIÓN 2	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:2	A4
	HOJA 1 DE 1	



FECHA: 14/06/2023	TÍTULO: BARRA CONSTRUCTIVA	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ESCALA: 1:2	A4 HOJA 1 DE 1



FECHA: 14/06/2023
 AUTOR:
 PEDRO MARTÍNEZ ESTEVE



UNIVERSITAT
 POLITÈCNICA
 DE VALÈNCIA

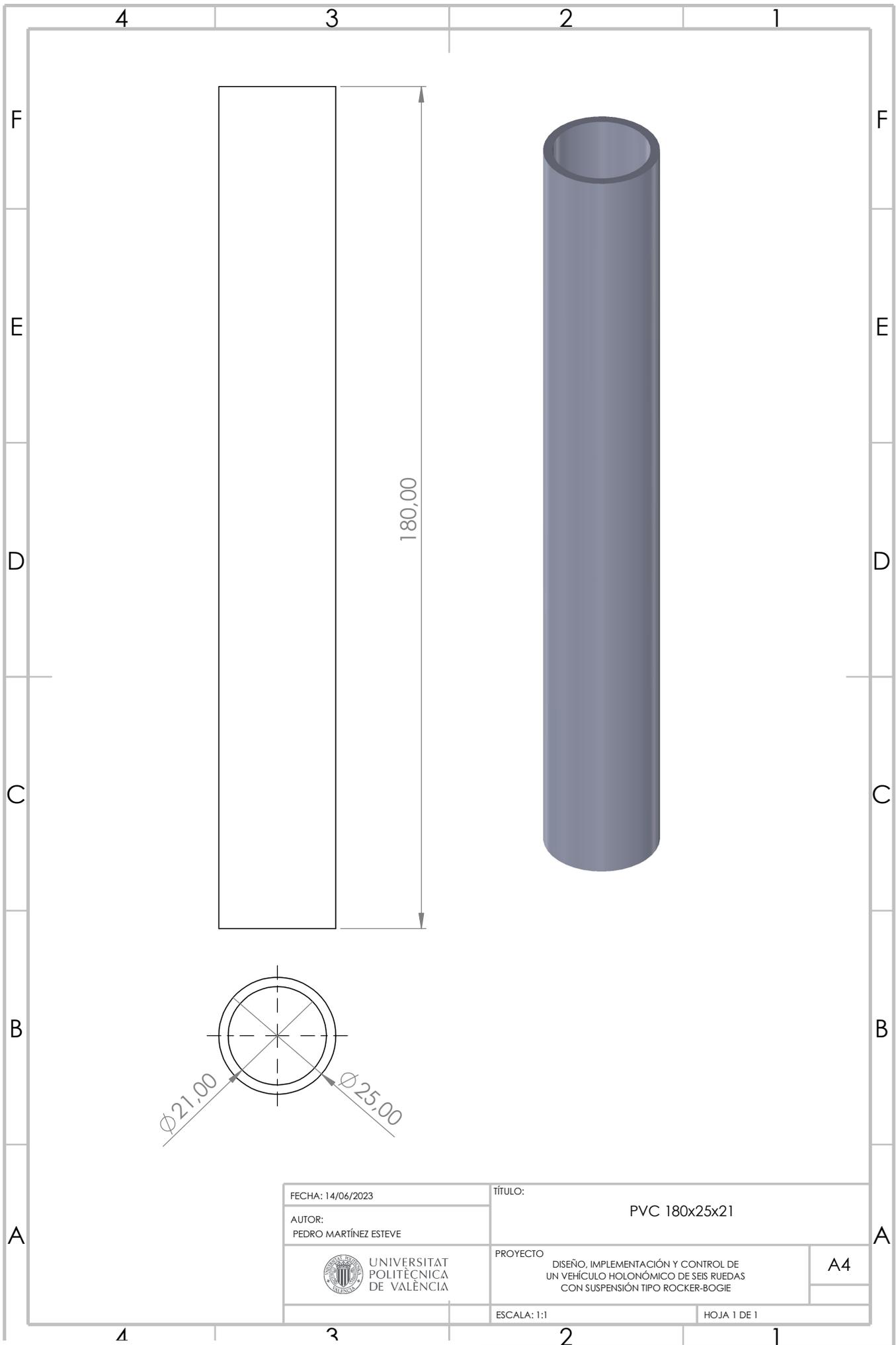
TÍTULO:
 ENLACE CONSTRUCTIVO

PROYECTO
 DISEÑO, IMPLEMENTACIÓN Y CONTROL DE
 UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS
 CON SUSPENSIÓN TIPO ROCKER-BOGIE

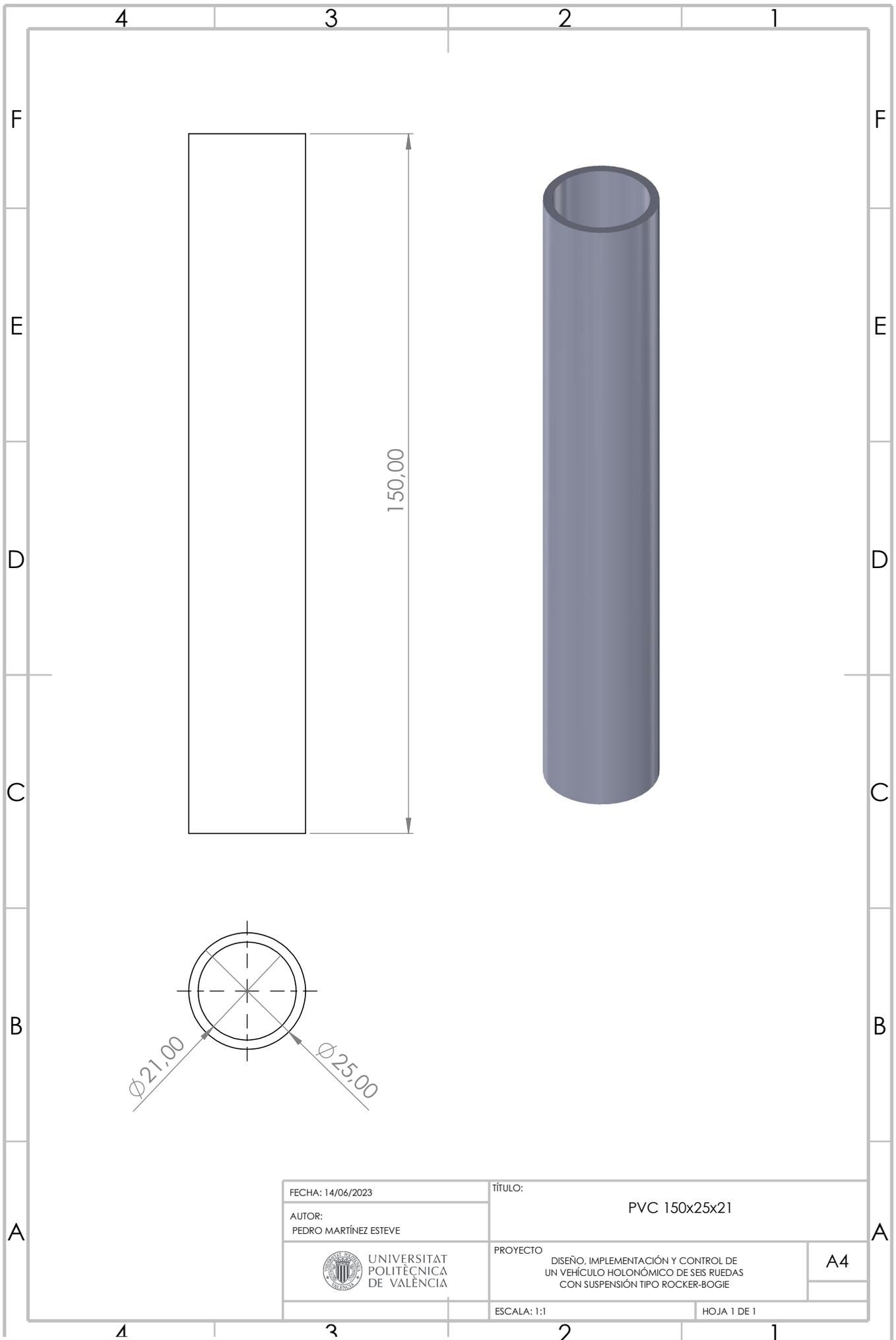
A4

ESCALA: 1:1

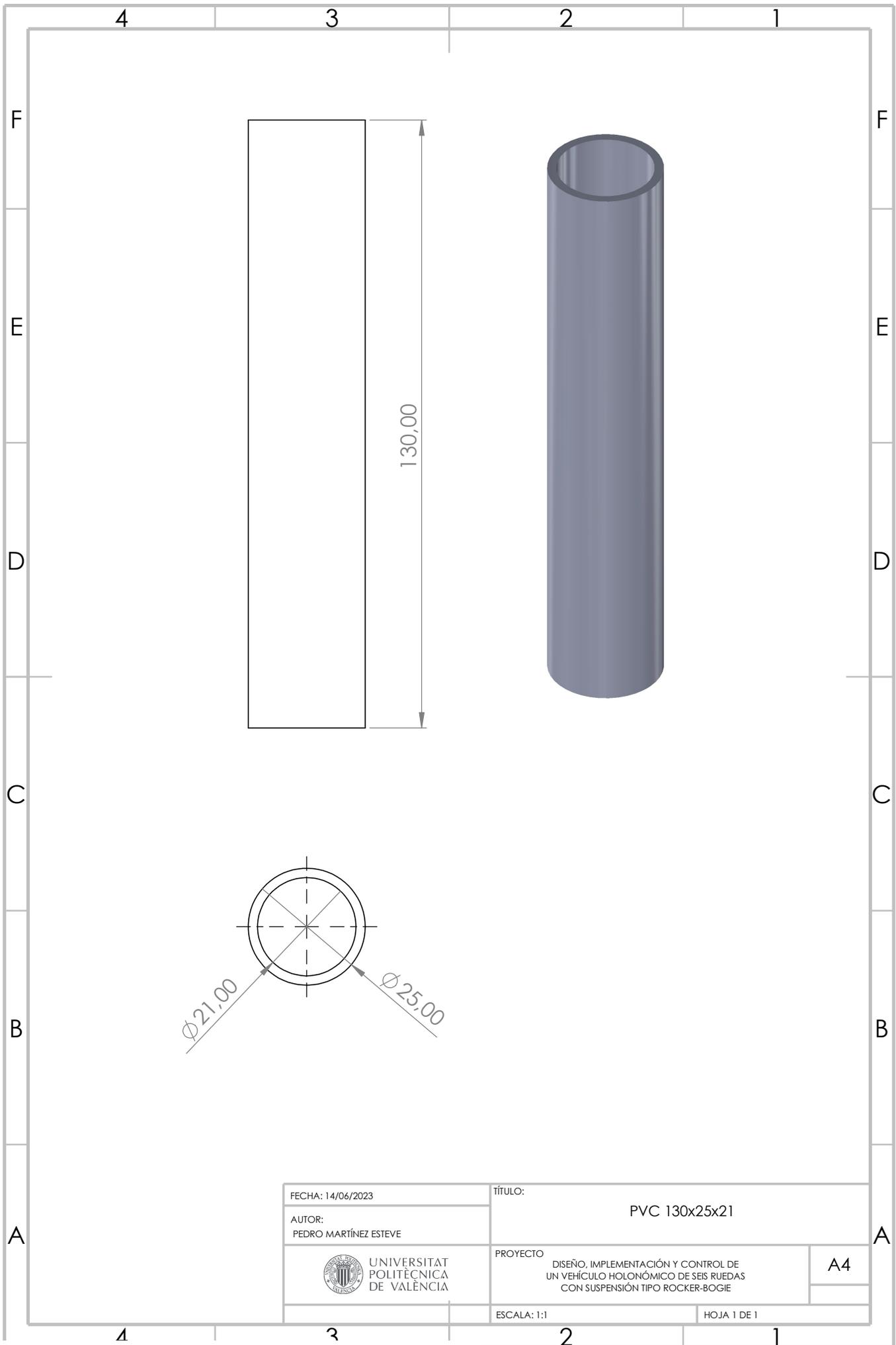
HOJA 1 DE 1



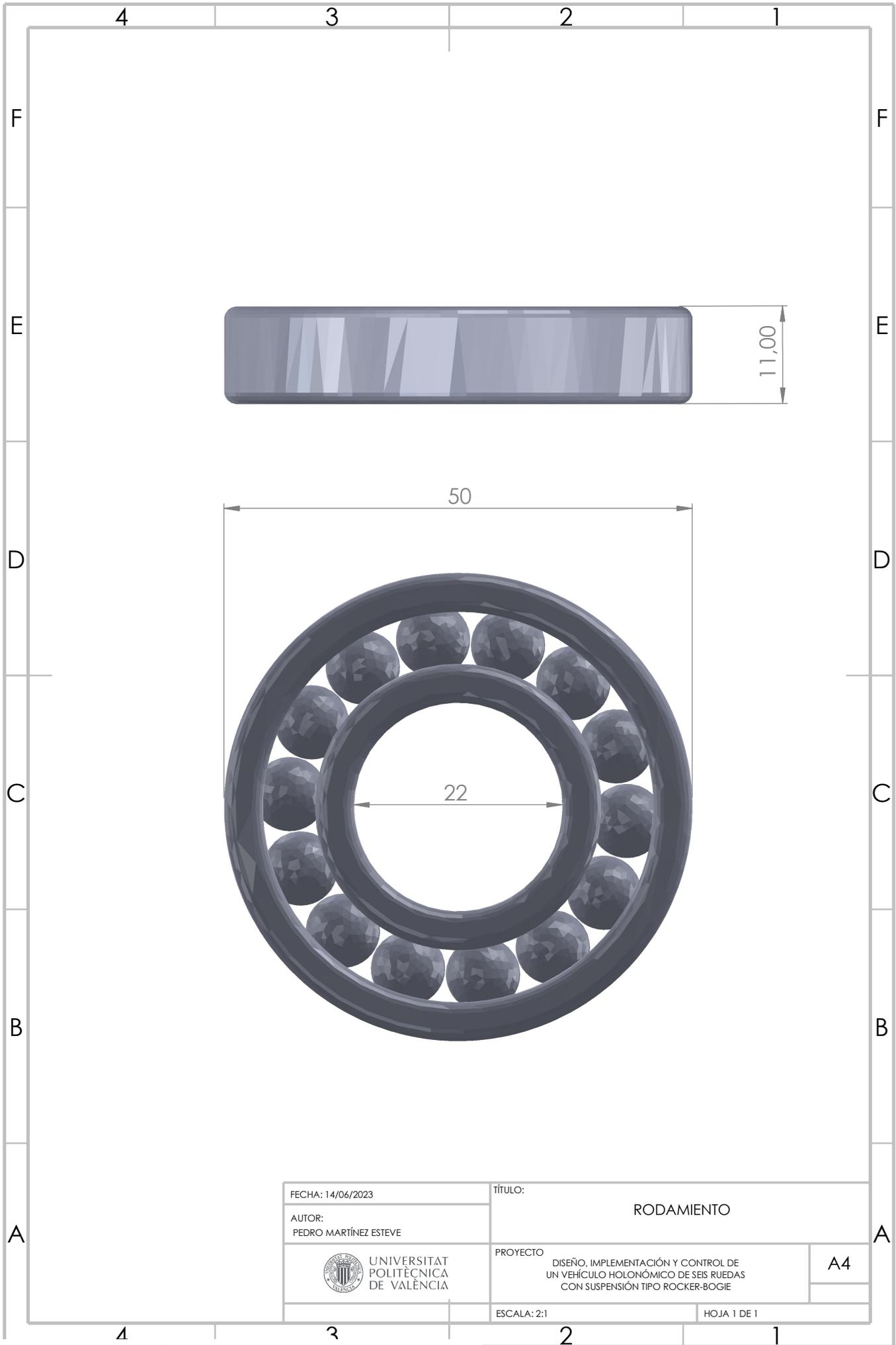
FECHA: 14/06/2023	TÍTULO:	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PVC 180x25x21	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
ESCALA: 1:1	HOJA 1 DE 1	



FECHA: 14/06/2023	TÍTULO:	
AUTOR: PEDRO MARTÍNEZ ESTEVE	PVC 150x25x21	
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
	ESCALA: 1:1	HOJA 1 DE 1



FECHA: 14/06/2023	TÍTULO: PVC 130x25x21	
AUTOR: PEDRO MARTÍNEZ ESTEVE		
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO	A4
	DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLONÓMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	
ESCALA: 1:1	HOJA 1 DE 1	



FECHA: 14/06/2023	TÍTULO: RODAMIENTO	
AUTOR: PEDRO MARTÍNEZ ESTEVE		
 UNIVERSITAT POLITÈCNICA DE VALÈNCIA	PROYECTO: DISEÑO, IMPLEMENTACIÓN Y CONTROL DE UN VEHÍCULO HOLÓNOMICO DE SEIS RUEDAS CON SUSPENSIÓN TIPO ROCKER-BOGIE	A4
	ESCALA: 2:1	HOJA 1 DE 1

ANEXO 2: CÓDIGO

```
// TFM PEDRO MARTINEZ ESTEVE

//Incluimos librerias

#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
#include <SoftwareSerial.h>

Adafruit_PWMServoDriver servoController =
    Adafruit_PWMServoDriver(0x40);

// Definicion de los pines de recepcion y transmision

#define rxPin 14
#define txPin 15
SoftwareSerial Serial1 = SoftwareSerial(rxPin, txPin);

// Definiciones de pines de los motores de corriente continua

const int pinENADD = 10; // Pin PWM del motor DD
const int pinMotorIN1DD = 26; // Pin IN1 del driver L298N para el
    motor DD
const int pinMotorIN2DD = 27; // Pin IN2 del driver L298N para el
    motor DD

const int pinENBDI = 13; // Pin PWM del motor DI
const int pinMotorIN3DI = 12; // Pin IN3 del driver L298N para el
    motor DI
const int pinMotorIN4DI = 11; // Pin IN4 del driver L298N para el
    motor DI

const int pinENACD = 9; // Pin PWM del motor CD
const int pinMotorIN1CD = 25; // Pin IN1 del driver L298N para el
    motor CD
const int pinMotorIN2CD = 24; // Pin IN2 del driver L298N para el
    motor CD

const int pinENBCI = 6; // Pin PWM del motor CI
```

```
const int pinMotorIN3CI = 7; // Pin IN3 del driver L298N para el
    motor CI
const int pinMotorIN4CI = 8; // Pin IN4 del driver L298N para el
    motor CI

const int pinENATD = 5; // Pin PWM del motor TD
const int pinMotorIN1TD = 23; // Pin IN1 del driver L298N para el
    motor TD
const int pinMotorIN2TD = 22; // Pin IN2 del driver L298N para el
    motor TD

const int pinENBTI = 2; // Pin PWM del motor TI
const int pinMotorIN3TI = 3; // Pin IN3 del driver L298N para el
    motor TI
const int pinMotorIN4TI = 4; // Pin IN4 del driver L298N para el
    motor TI

const int speed = 50; // Velocidad del 20% (50/255)

const uint8_t frequency = 50; // Frecuencia PWM de 50Hz o T=20ms
const uint16_t ServoMinTicks = 102; // ancho de pulso en ticks para
    pocicion 0°
const uint16_t ServoMaxTicks = 512; // ancho de pulso en ticks para
    la pocicion 360°

// Definiciones de canales de los servomotores en la PCA9685
const int servo1Channel = 15; // Canal de la PCA9685 para el servo 1
const int servo2Channel = 11; // Canal de la PCA9685 para el servo 2
const int servo3Channel = 8; // Canal de la PCA9685 para el servo 3
const int servo4Channel = 7; // Canal de la PCA9685 para el servo 4
const int servo5Channel = 4; // Canal de la PCA9685 para el servo 5
const int servo6Channel = 1; // Canal de la PCA9685 para el servo 6

char inByte = 0; // Comando Bluetooth Electronics

void setup() {

    // Configuracion de los pines TX y RX
```

```
pinMode(rxPin, INPUT);
pinMode(txPin, OUTPUT);

Serial.begin(9600);
Wire.begin();
Serial1.begin(9600);

// Inicialización de la comunicación con la PCA9685
servoController.begin();
servoController.setPWMPFreq(frequency);

// Configuración de pines de los motores de corriente continua
como salidas
pinMode(pinENADD, OUTPUT);
pinMode(pinMotorIN1DD, OUTPUT);
pinMode(pinMotorIN2DD, OUTPUT);

pinMode(pinENBDI, OUTPUT);
pinMode(pinMotorIN3DI, OUTPUT);
pinMode(pinMotorIN4DI, OUTPUT);

pinMode(pinENACD, OUTPUT);
pinMode(pinMotorIN1CD, OUTPUT);
pinMode(pinMotorIN2CD, OUTPUT);

pinMode(pinENBCI, OUTPUT);
pinMode(pinMotorIN3CI, OUTPUT);
pinMode(pinMotorIN4CI, OUTPUT);

pinMode(pinENATD, OUTPUT);
pinMode(pinMotorIN1TD, OUTPUT);
pinMode(pinMotorIN2TD, OUTPUT);

pinMode(pinENBTI, OUTPUT);
pinMode(pinMotorIN3TI, OUTPUT);
pinMode(pinMotorIN4TI, OUTPUT);

}
```

```
void loop() {
  if (Serial1.available()) {
    inByte = Serial1.read();
    switch (inByte) {
      case '1':
        moveForwardStraight();
        break;
      case '3':
        moveBackwardStraight();
        break;
      case '2':
        moveRight();
        break;
      case '4':
        moveLeft();
        break;
      case 'H':
        rotateClockwise();
        break;
      case 'A':
        rotateCounterClockwise();
        break;
      case '5':
        turnRightForward();
        break;
      case '8':
        turnLeftForward();
        break;
      case '6':
        turnRightBackward();
        break;
      case '7':
        turnLeftBackward();
        break;
      case 'S':
        stopMotion();
        break;
    }
  }
}
```

```
        default:
            stopMotion();
            break;
    }
}

// Función movimiento servos

void moveServo(int servoChannel, int angulo) {
    int pulso = map(angulo, 0, 360, ServoMinTicks, ServoMaxTicks); //
    Mapea el ángulo a un rango de pulsos adecuado
    servoController.setPWM(servoChannel, 0, pulso); // Establece el
    ancho de pulso para el canal del servo
}

// Funciones de movimiento del vehículo

void moveForwardStraight() {

    Serial.println("Moviendo hacia delante");

    moveServo(servo1Channel, 0);
    moveServo(servo2Channel, 0);
    moveServo(servo3Channel, 0);
    moveServo(servo4Channel, 0);
    moveServo(servo5Channel, 0);
    moveServo(servo6Channel, 0);
    delay(1000);

    digitalWrite(pinMotorIN1DD, HIGH);
    digitalWrite(pinMotorIN2DD, LOW);
    analogWrite(pinENADD, speed);

    digitalWrite(pinMotorIN3DI, HIGH);
    digitalWrite(pinMotorIN4DI, LOW);
    analogWrite(pinENBDI, speed);
}
```

```
digitalWrite (pinMotorIN1CD, HIGH);
digitalWrite (pinMotorIN2CD, LOW);
analogWrite (pinENACD, speed);

digitalWrite (pinMotorIN3CI, HIGH);
digitalWrite (pinMotorIN4CI, LOW);
analogWrite (pinENBCI, speed);

digitalWrite (pinMotorIN1TD, HIGH);
digitalWrite (pinMotorIN2TD, LOW);
analogWrite (pinENATD, speed);

digitalWrite (pinMotorIN3TI, HIGH);
digitalWrite (pinMotorIN4TI, LOW);
analogWrite (pinENBTI, speed);

delay (1000);
}

void moveBackwardStraight () {

    Serial.println ("Moviendo hacia atrás");

    moveServo (servo1Channel, 0);
    moveServo (servo2Channel, 0);
    moveServo (servo3Channel, 0);
    moveServo (servo4Channel, 0);
    moveServo (servo5Channel, 0);
    moveServo (servo6Channel, 0);
    delay (1000);

    digitalWrite (pinMotorIN1DD, LOW);
    digitalWrite (pinMotorIN2DD, HIGH);
    analogWrite (pinENADD, speed);

    digitalWrite (pinMotorIN3DI, LOW);
    digitalWrite (pinMotorIN4DI, HIGH);
    analogWrite (pinENBDI, speed);
```

```
digitalWrite(pinMotorIN1CD, LOW);
digitalWrite(pinMotorIN2CD, HIGH);
analogWrite(pinENACD, speed);

digitalWrite(pinMotorIN3CI, LOW);
digitalWrite(pinMotorIN4CI, HIGH);
analogWrite(pinENBCI, speed);

digitalWrite(pinMotorIN1TD, LOW);
digitalWrite(pinMotorIN2TD, HIGH);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, LOW);
digitalWrite(pinMotorIN4TI, HIGH);
analogWrite(pinENBTI, speed);

delay(1000);
}

void moveRight() {

  Serial.println("Moviendo hacia la derecha");

  moveServo(servo1Channel, 90);
  moveServo(servo2Channel, -90);
  moveServo(servo3Channel, 90);
  moveServo(servo4Channel, -90);
  moveServo(servo5Channel, 90);
  moveServo(servo6Channel, -90);
  delay(1000);

  digitalWrite(pinMotorIN1DD, HIGH);
  digitalWrite(pinMotorIN2DD, LOW);
  analogWrite(pinENADD, speed);

  digitalWrite(pinMotorIN3DI, HIGH);
  digitalWrite(pinMotorIN4DI, LOW);
```

```
    analogWrite (pinENBDI, speed);

    digitalWrite (pinMotorIN1CD, HIGH);
    digitalWrite (pinMotorIN2CD, LOW);
    analogWrite (pinENACD, speed);

    digitalWrite (pinMotorIN3CI, HIGH);
    digitalWrite (pinMotorIN4CI, LOW);
    analogWrite (pinENBCI, speed);

    digitalWrite (pinMotorIN1TD, HIGH);
    digitalWrite (pinMotorIN2TD, LOW);
    analogWrite (pinENATD, speed);

    digitalWrite (pinMotorIN3TI, HIGH);
    digitalWrite (pinMotorIN4TI, LOW);
    analogWrite (pinENBTI, speed);

    delay (1000);
}

void moveLeft() {

    Serial.println("Moviendo hacia la izquierda");

    moveServo (servo1Channel, 90);
    moveServo (servo2Channel, -90);
    moveServo (servo3Channel, 90);
    moveServo (servo4Channel, -90);
    moveServo (servo5Channel, 90);
    moveServo (servo6Channel, -90);
    delay (1000);

    digitalWrite (pinMotorIN1DD, LOW);
    digitalWrite (pinMotorIN2DD, HIGH);
    analogWrite (pinENADD, speed);

    digitalWrite (pinMotorIN3DI, LOW);
```

```
digitalWrite(pinMotorIN4DI, HIGH);
analogWrite(pinENBDI, speed);

digitalWrite(pinMotorIN1CD, LOW);
digitalWrite(pinMotorIN2CD, HIGH);
analogWrite(pinENACD, speed);

digitalWrite(pinMotorIN3CI, LOW);
digitalWrite(pinMotorIN4CI, HIGH);
analogWrite(pinENBCI, speed);

digitalWrite(pinMotorIN1TD, LOW);
digitalWrite(pinMotorIN2TD, HIGH);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, LOW);
digitalWrite(pinMotorIN4TI, HIGH);
analogWrite(pinENBTI, speed);

delay(1000);
}

void rotateClockwise() {

    Serial.println("Girando en sentido horario");

    moveServo(servo1Channel, -45);
    moveServo(servo2Channel, 45);
    moveServo(servo3Channel, 0);
    moveServo(servo4Channel, 0);
    moveServo(servo5Channel, 45);
    moveServo(servo6Channel, -45);
    delay(1000);

    digitalWrite(pinMotorIN1DD, LOW);
    digitalWrite(pinMotorIN2DD, HIGH);
    analogWrite(pinENADD, speed);
```

```
digitalWrite(pinMotorIN3DI, HIGH);
digitalWrite(pinMotorIN4DI, LOW);
analogWrite(pinENBDI, speed);

digitalWrite(pinMotorIN1CD, LOW);
digitalWrite(pinMotorIN2CD, LOW);
analogWrite(pinENACD, 0);

digitalWrite(pinMotorIN3CI, LOW);
digitalWrite(pinMotorIN4CI, LOW);
analogWrite(pinENBCI, 0);

digitalWrite(pinMotorIN1TD, LOW);
digitalWrite(pinMotorIN2TD, HIGH);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, HIGH);
digitalWrite(pinMotorIN4TI, LOW);
analogWrite(pinENBTI, speed);

delay(1000);
}

void rotateCounterClockwise() {

Serial.println("Girando en sentido antihorario");

moveServo(servo1Channel, -45);
moveServo(servo2Channel, 45);
moveServo(servo3Channel, 0);
moveServo(servo4Channel, 0);
moveServo(servo5Channel, 45);
moveServo(servo6Channel, -45);
delay(1000);

digitalWrite(pinMotorIN1DD, HIGH);
digitalWrite(pinMotorIN2DD, LOW);
analogWrite(pinENADD, speed);
```

```
digitalWrite(pinMotorIN3DI, LOW);
digitalWrite(pinMotorIN4DI, HIGH);
analogWrite(pinENBDI, speed);

digitalWrite(pinMotorIN1CD, LOW);
digitalWrite(pinMotorIN2CD, LOW);
analogWrite(pinENACD, 0);

digitalWrite(pinMotorIN3CI, LOW);
digitalWrite(pinMotorIN4CI, LOW);
analogWrite(pinENBCI, 0);

digitalWrite(pinMotorIN1TD, HIGH);
digitalWrite(pinMotorIN2TD, LOW);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, LOW);
digitalWrite(pinMotorIN4TI, HIGH);
analogWrite(pinENBTI, speed);

delay(1000);
}

void turnRightForward() {

  Serial.println("Girando hacia la derecha");

  moveServo(servo1Channel, 22);
  moveServo(servo2Channel, 22);
  moveServo(servo3Channel, 0);
  moveServo(servo4Channel, 0);
  moveServo(servo5Channel, -22);
  moveServo(servo6Channel, -22);
  delay(1000);

  digitalWrite(pinMotorIN1DD, HIGH);
  digitalWrite(pinMotorIN2DD, LOW);
```

```
    analogWrite(pinENADD, speed);

    digitalWrite(pinMotorIN3DI, HIGH);
    digitalWrite(pinMotorIN4DI, LOW);
    analogWrite(pinENBDI, speed);

    digitalWrite(pinMotorIN1CD, HIGH);
    digitalWrite(pinMotorIN2CD, LOW);
    analogWrite(pinENACD, speed);

    digitalWrite(pinMotorIN3CI, HIGH);
    digitalWrite(pinMotorIN4CI, LOW);
    analogWrite(pinENBCI, speed);

    digitalWrite(pinMotorIN1TD, HIGH);
    digitalWrite(pinMotorIN2TD, LOW);
    analogWrite(pinENATD, speed);

    digitalWrite(pinMotorIN3TI, HIGH);
    digitalWrite(pinMotorIN4TI, LOW);
    analogWrite(pinENBTI, speed);

    delay(1000);
}

void turnLeftForward() {

    Serial.println("Girando hacia la izquierda");

    moveServo(servo1Channel, -22);
    moveServo(servo2Channel, -22);
    moveServo(servo3Channel, 0);
    moveServo(servo4Channel, 0);
    moveServo(servo5Channel, 22);
    moveServo(servo6Channel, 22);
    delay(1000);

    digitalWrite(pinMotorIN1DD, HIGH);
```

```
digitalWrite(pinMotorIN2DD, LOW);
analogWrite(pinENADD, speed);

digitalWrite(pinMotorIN3DI, HIGH);
digitalWrite(pinMotorIN4DI, LOW);
analogWrite(pinENBDI, speed);

digitalWrite(pinMotorIN1CD, HIGH);
digitalWrite(pinMotorIN2CD, LOW);
analogWrite(pinENACD, speed);

digitalWrite(pinMotorIN3CI, HIGH);
digitalWrite(pinMotorIN4CI, LOW);
analogWrite(pinENBCI, speed);

digitalWrite(pinMotorIN1TD, HIGH);
digitalWrite(pinMotorIN2TD, LOW);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, HIGH);
digitalWrite(pinMotorIN4TI, LOW);
analogWrite(pinENBTI, speed);

delay(1000);
}

void turnRightBackward() {

  Serial.println("Retrocediendo hacia la derecha");

  moveServo(servo1Channel, -22);
  moveServo(servo2Channel, -22);
  moveServo(servo3Channel, 0);
  moveServo(servo4Channel, 0);
  moveServo(servo5Channel, 22);
  moveServo(servo6Channel, 22);
  delay(1000);
}
```

```
digitalWrite(pinMotorIN1DD, LOW);
digitalWrite(pinMotorIN2DD, HIGH);
analogWrite(pinENADD, speed);

digitalWrite(pinMotorIN3DI, LOW);
digitalWrite(pinMotorIN4DI, HIGH);
analogWrite(pinENBDI, speed);

digitalWrite(pinMotorIN1CD, LOW);
digitalWrite(pinMotorIN2CD, HIGH);
analogWrite(pinENACD, speed);

digitalWrite(pinMotorIN3CI, LOW);
digitalWrite(pinMotorIN4CI, HIGH);
analogWrite(pinENBCI, speed);

digitalWrite(pinMotorIN1TD, LOW);
digitalWrite(pinMotorIN2TD, HIGH);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, LOW);
digitalWrite(pinMotorIN4TI, HIGH);
analogWrite(pinENBTI, speed);

delay(1000);
}

void turnLeftBackward() {

  Serial.println("Retrocediendo hacia la izquierda");

  moveServo(servo1Channel, 22);
  moveServo(servo2Channel, 22);
  moveServo(servo3Channel, 0);
  moveServo(servo4Channel, 0);
  moveServo(servo5Channel, -22);
  moveServo(servo6Channel, -22);
  delay(1000);
```

```
digitalWrite(pinMotorIN1DD, LOW);
digitalWrite(pinMotorIN2DD, HIGH);
analogWrite(pinENADD, speed);

digitalWrite(pinMotorIN3DI, LOW);
digitalWrite(pinMotorIN4DI, HIGH);
analogWrite(pinENBDI, speed);

digitalWrite(pinMotorIN1CD, LOW);
digitalWrite(pinMotorIN2CD, HIGH);
analogWrite(pinENACD, speed);

digitalWrite(pinMotorIN3CI, LOW);
digitalWrite(pinMotorIN4CI, HIGH);
analogWrite(pinENBCI, speed);

digitalWrite(pinMotorIN1TD, LOW);
digitalWrite(pinMotorIN2TD, HIGH);
analogWrite(pinENATD, speed);

digitalWrite(pinMotorIN3TI, LOW);
digitalWrite(pinMotorIN4TI, HIGH);
analogWrite(pinENBTI, speed);

delay(1000);
}

void stopMotion() {

    Serial.println("Vehiculo en parada");

    moveServo(servo1Channel, 0);
    moveServo(servo2Channel, 0);
    moveServo(servo3Channel, 0);
    moveServo(servo4Channel, 0);
    moveServo(servo5Channel, 0);
    moveServo(servo6Channel, 0);
```

```
    delay(1000);

    digitalWrite(pinMotorIN1DD, LOW);
    digitalWrite(pinMotorIN2DD, LOW);
    analogWrite(pinENADD, 0);

    digitalWrite(pinMotorIN3DI, LOW);
    digitalWrite(pinMotorIN4DI, LOW);
    analogWrite(pinENBDI, 0);

    digitalWrite(pinMotorIN1CD, LOW);
    digitalWrite(pinMotorIN2CD, LOW);
    analogWrite(pinENACD, 0);

    digitalWrite(pinMotorIN3CI, LOW);
    digitalWrite(pinMotorIN4CI, LOW);
    analogWrite(pinENBCI, 0);

    digitalWrite(pinMotorIN1TD, LOW);
    digitalWrite(pinMotorIN2TD, LOW);
    analogWrite(pinENATD, 0);

    digitalWrite(pinMotorIN3TI, LOW);
    digitalWrite(pinMotorIN4TI, LOW);
    analogWrite(pinENBTI, 0);

    delay(1000);
}
```