

**DISEÑO Y CONTRUCCION DE UN HELIBOT 3 DOF (HELICOPTERO DE 3GDL)  
COMO HERRAMIENTA DE INVESTIGACION Y DESARROLLO PARA EL  
LABORATORIO DE SISTEMAS DINAMICOS E INGENIERIA DEL CONTROL.**

**ANDRES ALCIDES III BERMUDEZ ROA  
MANUEL ANTONIO DIAZ DURAN**

**ESCUELA DE INGENIERÍA MECÁNICA  
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS  
UNIVERSIDAD INDUSTRIAL DE SANTANDER  
BUCARAMANGA**

**2010**

**DISEÑO Y CONTRUCCION DE UN HELIBOT 3 DOF (HELICOPTERO DE 3GDL)  
COMO HERRAMIENTA DE INVESTIGACION Y DESARROLLO PARA EL  
LABORATORIO DE SISTEMAS DINAMICOS E INGENIERIA DEL CONTROL**

**ANDRES ALCIDES III BERMUDEZ ROA  
MANUEL ANTONIO DIAZ DURAN**

**Trabajo de grado para optar al título de  
Ingeniero mecánico**

**Director  
CARLOS BORRAS PINILLA  
Doctor, Ingeniero Mecánico**

**ESCUELA DE INGENIERÍA MECÁNICA  
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS  
UNIVERSIDAD INDUSTRIAL DE SANTANDER  
BUCARAMANGA**

**2010**

A mi padre, un gran ingeniero quien me enseñó el valor del trabajo.

A mamá, la mejor de las mujeres y mi principal pilar.

A mis hermanos, para quienes espero ser un buen ejemplo.

Andrés Bermúdez

Primeramente a Dios.  
Al apoyo incondicional de mis Padres  
y a mis hermanos por creer en mí.  
Manuel Díaz

## **AGRADECIMIENTOS**

A la Universidad Industrial de Santander.

Al profesor Carlos Borrás por confiar en nosotros el desarrollo de este proyecto e incentivarnos a la investigación y al desarrollo de temas novedosos en la UIS.

A los grupos de investigación ERA (Electrónica y robótica avanzada) y DICBOT de la Universidad Industrial de Santander, por todo el apoyo brindado desde el principio del proyecto en todos los campos.

A Raúl Hernando Cadena, presidente del grupo ERA por su respaldo y colaboración oportuna.

## CONTENIDO

<b>INTRODUCCIÓN .....</b>	<b>21</b>
<b>ANTECEDENTES.....</b>	<b>23</b>
<b>1 FORMULACIÓN DEL PROBLEMA.....</b>	<b>24</b>
1.1 IDENTIFICACIÓN DEL PROBLEMA .....	24
1.2 JUSTIFICACIÓN PARA LA SOLUCION DEL PROBLEMA .....	25
1.3 OBJETIVOS DEL TRABAJO DE GRADO .....	27
1.3.1 <i>Objetivo General</i> .....	27
1.3.2 <i>Objetivos Específicos</i> .....	27
<b>2 MODELO ESTRUCTURAL DEL HELIBOT COMO SISTEMA DE CONTROL AUTOMÁTICO</b>	<b>29</b>
2.1 PARTE OPERATIVA .....	30
2.1.1 <i>La planta dinámica</i> .....	31
2.1.2 <i>Sensores</i> .....	32
2.1.3 <i>Actuador</i> .....	34
2.2 PARTE DE CONTROL.....	37
2.2.1 <i>Preactuador</i> .....	37
2.2.2 <i>Controlador</i> .....	38
<b>3 MODELADO MATEMATICO DEL SISTEMA.....</b>	<b>40</b>
3.1 ECUACIONES DINAMICAS DE MOVIMIENTO .....	41
3.1.1 <i>Angulo de elevación o pitch</i> .....	41
3.1.2 <i>Angulo de alabeo</i> .....	46
<b>4 DISEÑO DEL CONTROLADOR .....</b>	<b>49</b>
4.1 INTRODUCCION AL CONTROL PID .....	49
4.2 ESTRUCTURA DEL CONTROL PID .....	50
4.3 DISEÑO DEL CONTROLADOR DEL ANGULO DE ELEVACIÓN.....	51
4.3.1 <i>Requerimientos de diseño</i> .....	51
4.3.2 <i>Análisis de la respuesta transitoria</i> .....	52

4.3.3	<i>Función de Transferencia del ángulo de elevación</i> .....	52
4.3.4	<i>Análisis de los diferentes métodos de control PID</i> .....	54
4.4	DISEÑO DEL CONTROLADOR DEL ANGULO DE ALABEO .....	68
4.4.1	<i>Requerimientos de diseño</i> .....	68
4.4.2	<i>Función de transferencia del ángulo de alabeo</i> .....	68
4.4.3	<i>Análisis de los diferentes métodos de control PID</i> .....	71
<b>5</b>	<b>DISEÑO ESTRUCTURAL DEL HELIBOT .....</b>	<b>80</b>
5.1	ESTRUCTURA BASE DEL HELIBOT .....	82
5.2	BLOQUE CENTRAL DEL HELIBOT .....	82
5.3	BLOQUE VIGA PRINCIPAL.....	84
5.4	BLOQUE HELICE - MOTOR.....	85
5.5	ESTRUCTURA DE APOYO.....	86
5.6	SISTEMA DE CONTROL.....	86
<b>6</b>	<b>DISEÑO DE LA ESTRUCTURA BASE DEL HELIBOT .....</b>	<b>88</b>
6.1	DISEÑO DEL RESORTE .....	89
6.2	SELECCIÓN DEL RODAMIENTO LINEAL.....	93
6.3	DISEÑO DEL EJE VERTICAL .....	94
<b>7</b>	<b>DISEÑO DEL BLOQUE CENTRAL DEL HELIBOT.....</b>	<b>97</b>
7.1	SOPORTE BLOQUE CENTRAL.....	98
7.2	SELECCIÓN DEL RODAMIENTO DE BOLAS .....	100
7.3	ESTRUCTURA SOPORTE DEL ENCODER DE ELEVACION.....	103
7.4	SELECCIÓN DEL SENSOR .....	104
7.5	DISEÑO EJE DE ELEVACION .....	105
<b>8</b>	<b>DISEÑO DEL BLOQUE VIGA PRINCIPAL.....</b>	<b>108</b>
8.1	DISEÑO BRAZO PRINCIPAL .....	109
8.2	DISEÑO DEL BLOQUE SENSOR DE ALABEO .....	111
	<i>Diseño del eje ángulo de alabeo</i> .....	112
8.3	DISEÑO DEL CONTRAPESO .....	114
<b>9</b>	<b>DISEÑO DEL BLOQUE HELICE – MOTOR .....</b>	<b>116</b>
9.1	DISEÑO DE LAS CARCASAS DE PROTECCION .....	117
9.2	DISEÑO DEL SOPORTE CONJUNTO HELICE-MOTOR .....	118

9.3	DISEÑO DEL EJE BLOQUE HELICE-MOTOR.....	119
<b>10</b>	<b>SISTEMA DE CONTROL.....</b>	<b>122</b>
10.1	SISTEMA DE CONTROL EMBEBIDO .....	122
10.1.1	<i>Características</i> .....	122
10.1.2	<i>Arquitectura de los Sistemas Embebidos</i> .....	123
10.1.3	<i>Metodología de Diseño</i> .....	125
10.2	MICROCONTROLADOR .....	128
	<i>Microcontrolador MCF51QE128</i> .....	129
10.3	SOFTWARE.....	130
10.4	COMPONENTES DEL SISTEMA EMBEBIDO .....	133
10.4.1	<i>Bloque de regulación y potencia</i> .....	134
10.4.2	<i>Bloque de comunicación con el pc</i> .....	134
10.4.3	<i>Puerto de conexión de teclado</i> .....	134
10.4.4	<i>Puerto de conexión de los sensores</i> .....	134
10.4.5	<i>Puerto de conexión botón de emergencia</i> .....	135
10.4.6	<i>Puerto de conexión LCD</i> .....	135
10.4.7	<i>Puertos de conexión de los motores</i> .....	135
10.4.8	<i>Puertos de control de mando inalámbrico</i> .....	136
10.4.9	<i>Microcontrolador</i> .....	136
10.4.10	<i>Selector de funcionamiento</i> .....	136
10.5	CONEXIONES A LA TARJETA.....	137
10.6	DIALOGO SISTEMA – OPERADOR .....	137
10.6.1	<i>Panel de control</i> .....	138
10.6.2	<i>Interface en MATLAB</i> .....	139
<b>11</b>	<b>INTERFACE EN MATLAB.....</b>	<b>140</b>
11.1	GENERALIDADES .....	140
	<i>PRESTACIONES DE MATLAB</i> .....	140
11.2	CONEXIÓN DE LA INTERFAZ EN EL ENTORNO MATLAB .....	141
	<i>Definición de las variables del proceso</i> .....	143
<b>12</b>	<b>PRUEBAS HELIBOT 3DOF .....</b>	<b>146</b>
12.1	CONTROL PROPORCIONAL .....	147
12.1.1	<i>Ángulo de elevación con controlador P</i> .....	147

12.1.2	<i>Ángulo de alabeo con controlador P</i> .....	149
12.2	CONTROL PROPORCIONAL – INTEGRAL.....	151
12.2.1	<i>Ángulo de elevación con controlador PI</i> .....	151
12.2.2	<i>Ángulo de alabeo con controlador PI</i> .....	154
12.3	CONTROL PROPORCIONAL-DERIVATIVO.....	156
12.3.1	<i>Ángulo de elevación con controlador PD</i> .....	157
12.3.2	<i>Ángulo de alabeo con controlador PD</i> .....	159
12.4	CONTROL PROPORCIONAL – INTEGRAL - DERIVATIVO.....	162
12.4.1	<i>Ángulo de elevación con control PID</i> .....	163
12.4.2	<i>Ángulo de alabeo con control PID</i> .....	165
<b>13</b>	<b>CONCLUSIONES</b> .....	<b>169</b>
<b>14</b>	<b>RECOMENDACIONES</b> .....	<b>172</b>
	<b>BIBLIOGRAFÍA</b> .....	<b>175</b>

## LISTA DE FIGURAS

FIGURA 1. DIAGRAMA DE BLOQUES DEL SISTEMA AUTOMÁTICO .....	29
FIGURA 2. DISEÑO ESQUEMÁTICO DEL BANCO	
FUENTE: LOS AUTORES.....	30
FIGURA 3. PLANTA DINÁMICA DE PRUEBAS .....	31
FIGURA 4. DIAGRAMA DE UN ENCODER INCREMENTAL	
FUENTE: MAZZONE VIRGINIA, MALLO SEBASTIÁN;DISEÑO Y CONSTRUCCIÓN DE UN PÉNDULO	
INVERTIDO ROTATIVO .....	33
FIGURA 5. CANALES DE SINCRONIZACIÓN DEL ENCODER INCREMENTAL	
<i>FUENTE: WWW.AUTOMATIONDIRECT.COM/ENCODERS</i> .....	33
FIGURA 6. ENCODER INCREMENTAL TRD-S2500-VD	
FUENTE: <i>WWW.AUTOMATIONDIRECT.COM/EN</i> .....	34
FIGURA 7. CONJUNTO BRUSHLESS – SPEED CONTROL	
FUENTE: <i>WWW.HOBBYKING.COM</i> .....	35
FIGURA 8. HÉLICES GWS HD8040	
FUENTE: <i>WWW.HOBBYKING.COM</i> .....	36
FIGURA 9. <i>TOWERPRO W18A BRUSHLESS SPEED CONTROLLER</i> .....	38
FIGURA 10. <i>MICROCONTROLADOR MCF51QE128 COLDFIRE V1</i> .....	38
FIGURA 11. GEOMETRÍA DEL ÁNGULO DE ELEVACIÓN .....	41
FIGURA 12. MODELO DINÁMICO DEL HELIBOT 3 DOF .....	43
FIGURA 13. GEOMETRÍA DEL ÁNGULO DE ALABEO .....	46
FIGURA 14. DIAGRAMA DE BLOQUES DEL SISTEMA DE CONTROL DEL PROCESO. ....	51
FIGURA 15. RESPUESTA AL ESCALÓN DEL ÁNGULO DE ELEVACIÓN SIN REGULACIÓN.....	53
FIGURA 16. SIMULACIÓN DEL SISTEMA DE CONTROL EN LAZO CERRADO CON CONTROL PROPORCIONAL	54
FIGURA 17. ANÁLISIS DE LA ACCIÓN DERIVATIVA EN UN CONTROLADOR PD .....	56
FIGURA 18. ANÁLISIS DE LA ACCIÓN INTEGRAL EN UN CONTROLADOR PID .....	57
FIGURA 19. RESPUESTA AL ESCALÓN PARA EL ÁNGULO DE ELEVACIÓN COMPENSADO.....	58
FIGURA 20. LUGAR DE LAS RAÍCES Y DIAGRAMA DE BODE, DEL COMPENSADOR $C_e(s)$ SINTONIZADO A TRAVÉS DEL ALGORITMO DE ZIEGLER-NICHOLS, DETERMINADOS MEDIANTE LA HERRAMIENTA SISOTOOL DE MATLAB. ....	60
FIGURA 21. RESPUESTA AL ESCALÓN E IMPULSO DEL SISTEMA DE LAZO CERRADO CON LOS PARÁMETROS DEL COMPENSADOR $C_e(s)$ , SINTONIZADO A TRAVÉS DEL ALGORITMO DE ZIEGLER –NICHOLS. ....	61

FIGURA 22. LUGAR DE LAS RAÍCES Y DIAGRAMA DE BODE, DEL CONTROLADOR PID CON FILTRO DERIVATIVO, DETERMINADOS MEDIANTE LA HERRAMIENTA SISOTOOL DE MATLAB.....	63
FIGURA 23. RESPUESTA AL ESCALÓN Y RESPUESTA AL IMPULSO DEL CONTROLADOR PID CON FILTRO DERIVATIVO .....	64
FIGURA 24. ANÁLISIS DE DIFERENTES MÉTODOS DE CONTROL CON BASE EN EL CONTROLADOR PID ; A. RESPUESTA AL ESCALÓN; B. RESPUESTA AL IMPULSO.....	66
FIGURA 25. TREN DE SEÑALES, PARA EL MONITOREO DEL ÁNGULO DE ELEVACIÓN.....	67
FIGURA 26. RESPUESTA AL ESCALÓN DEL ÁNGULO DE ALABEO SIN REGULACIÓN.....	70
FIGURA 27. RESPUESTA AL ESCALÓN PARA EL ÁNGULO DE ALABEO COMPENSADO.....	71
FIGURA 28. LUGAR DE LAS RAÍCES Y DIAGRAMA DE BODE, DEL COMPENSADOR Cc(S) SINTONIZADO A TRAVÉS DEL ALGORITMO DE ZIEGLER-NICHOLS, DETERMINADOS MEDIANTE LA HERRAMIENTA SISOTOOL DE MATLAB. ....	73
FIGURA 29. RESPUESTA AL ESCALÓN DEL SISTEMA DE LAZO CERRADO CON LOS PARÁMETROS DEL COMPENSADOR Cc(S), SINTONIZADO A TRAVÉS DEL ALGORITMO DE ZIEGLER –NICHOLS.....	74
FIGURA 30. LUGAR DE LAS RAÍCES Y DIAGRAMA DE BODE, DEL CONTROLADOR PID CON FILTRO DERIVATIVO, DETERMINADOS MEDIANTE LA HERRAMIENTA SISOTOOL DE MATLAB.....	75
FIGURA 31. RESPUESTA AL ESCALÓN Y RESPUESTA AL IMPULSO DEL CONTROLADOR PID CON FILTRO DERIVATIVO .....	76
FIGURA 32. ANÁLISIS DE DIFERENTES MÉTODOS DE CONTROL CON BASE EN EL CONTROLADOR PID A. RESPUESTA AL ESCALÓN; B. RESPUESTA AL IMPULSO .....	78
FIGURA 33. TREN DE SEÑALES, PARA EL MONITOREO DEL ÁNGULO DE CABECEO .....	79
FIGURA 34. PLANTA DINÁMICA HELIBOT 3 DOF .....	80
FIGURA 35. PARTES DEL BANCO.....	81
FIGURA 36. ESTRUCTURA BASE DEL HELIBOT.....	82
FIGURA 37. BLOQUE CENTRAL DEL HELIBOT 3 DOF .....	83
FIGURA 38. BLOQUE VIGA PRINCIPAL.....	84
FIGURA 39. BLOQUE HÉLICE-MOTOR DEL HELIBOT 3 DOF.....	85
FIGURA 40. ESTRUCTURA DE APOYO DEL HELIBOT 3 DOF .....	86
FIGURA 41. DISEÑO FINAL DE LA TARJETA DE CONTROL.....	87
FIGURA 42. DISEÑO FINAL DE LA ESTRUCTURA BASE DEL HELIBOT 3 DOF .....	88
FIGURA 43. ESTRUCTURA BASE DEL HELIBOT 3 DOF .....	89
FIGURA 44. DISEÑO FINAL DEL RESORTE .....	90
FIGURA 45. RODAMIENTO LINEAL NTN .....	93
FIGURA 46. DISEÑO FINAL DEL EJE .....	95

FIGURA 47. DISEÑO FINAL DEL BLOQUE CENTRAL DEL HELIBOT .....	97
FIGURA 48. DESPIECE BLOQUE CENTRAL DEL HELIBOT .....	98
FIGURA 49. SOPORTE BLOQUE CENTRAL .....	99
FIGURA 50. ESCALA PARA EL CÁLCULO DE LA VIDA NOMINAL BÁSICA DE LOS RODAMIENTOS.....	101
FIGURA 51. SOPORTE SENSOR DE ELEVACIÓN .....	103
FIGURA 52. ENCODER INCREMENTAL TRD-S2500-VD	
FUENTE: <a href="http://WWW.AUTOMATIONDIRECT.COM/ENCODERS">WWW.AUTOMATIONDIRECT.COM/ENCODERS</a> .....	104
FIGURA 53. ACOPLE DE ENCODER DE 6 MM (FIBRA DE VIDRIO).....	105
FIGURA 54. SOPORTE EJE DE ELEVACIÓN .....	106
FIGURA 55. SIMULACIÓN DE ESFUERZOS EN EL EJE DE ELEVACIÓN.....	106
FIGURA 56. ANÁLISIS DE DEFLEXIÓN EN EL EJE DE ELEVACIÓN. ....	107
FIGURA 57. DISEÑO FINAL BLOQUE VIGA PRINCIPAL.....	108
FIGURA 58. ESTRUCTURA BLOQUE VIGA PRINCIPAL.....	109
FIGURA 59. SIMULACIÓN DE ESFUERZOS EN LA VIGA PRINCIPAL .....	109
FIGURA 60. ANÁLISIS DE DEFORMACIÓN EN LA VIGA PRINCIPAL. ....	110
FIGURA 61. BLOQUE SENSOR DE ALABEO .....	111
FIGURA 62. PARTES DEL BLOQUE SENSOR DE ALABEO .....	111
FIGURA 63. EJE ÁNGULO DE ALABEO .....	113
FIGURA 64. SIMULACIÓN DE ESFUERZOS EN EL EJE DE ALABEO.....	113
FIGURA 65. ANÁLISIS DE DEFORMACIÓN EN EL EJE DE ALABEO.....	114
FIGURA 66. DISEÑO FINAL CONTRAPESO .....	115
FIGURA 67. SOPORTE DEL CONTRAPESO.....	115
FIGURA 68. DISEÑO FINAL DEL BLOQUE HÉLICE-MOTOR.....	116
FIGURA 69. DESPIECE DEL BLOQUE HÉLICE-MOTOR .....	117
FIGURA 70. CARCASA DE PROTECCIÓN.....	118
FIGURA 71. SOPORTE CONJUNTO HÉLICE - MOTOR.....	119
FIGURA 72. DISEÑO FINAL DEL EJE BLOQUE HÉLICE - MOTOR .....	120
FIGURA 73. SIMULACIÓN DE ESFUERZOS EN EL EJE DEL BLOQUE HÉLICE-MOTOR .....	120
FIGURA 74. ANÁLISIS DE DEFORMACIÓN EN EL EJE BLOQUE HÉLICE-MOTOR.....	121
FIGURA 75. ARQUITECTURA DE UN SISTEMA EMBEBIDO .....	123
FIGURA 76. FLUJO DE DISEÑO DE UN SISTEMA EMBEBIDO .....	127
FIGURA 77. ESQUEMÁTICO Y ASIGNACIÓN DE PINES DEL MICROCONTROLADOR MCF51QE128.....	131
FIGURA 78. DIAGRAMA DE FLUJO DE LA PROGRAMACIÓN DEL MICROCONTROLADOR.....	132
FIGURA 79. BLOQUES PRINCIPALES DE LA TARJETA.....	133

FIGURA 80. CONEXIONES DEL HARDWARE Y SOFTWARE A LA TARJETA .....	137
FIGURA 81. CAJA DE CONTROL .....	139
FIGURA 82. ESTABLECER LA CARPETA DE LA INTERFAZ .....	141
FIGURA 83. CONFIGURACIÓN DE LA CONEXIÓN A LA INTERFAZ .....	142
FIGURA 84. PANTALLAZO INICIAL DE LA INTERFAZ.....	143
FIGURA 85. PANTALLAZO DE RESULTADOS EN LA INTERFAZ .....	145
FIGURA 86. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL DE PRUEBA 1. .....	147
FIGURA 87. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL DE PRUEBA 2. .....	148
FIGURA 88. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL DE PRUEBA 3. .....	149
FIGURA 89. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL DE PRUEBA 4.	150
FIGURA 90. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL-INTEGRAL DE PRUEBA 5. ....	151
FIGURA 91. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL-INTEGRAL DE PRUEBA 6. ....	152
FIGURA 92. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL-INTEGRAL DE PRUEBA 7. ....	153
FIGURA 93. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL-INTEGRAL DE PRUEBA 8. ....	154
FIGURA 94. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL-INTEGRAL DE PRUEBA 9. ....	155
FIGURA 95. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL-INTEGRAL DE PRUEBA 10. ....	156
FIGURA 96. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL-DERIVATIVO DE PRUEBA 11. ....	157
FIGURA 97. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL-DERIVATIVO DE PRUEBA 12. ....	158
FIGURA 98. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROL PROPORCIONAL-DERIVATIVO DE PRUEBA 13. ....	159
FIGURA 99. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL-DERIVATIVO DE PRUEBA 14. ....	160

FIGURA 100. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL-DERIVATIVO DE PRUEBA 15. ....	161
FIGURA 101. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROL PROPORCIONAL-DERIVATIVO DE PRUEBA 16. ....	162
FIGURA 102. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROLADOR PID Y SET POINT EN 10.8°. ....	163
FIGURA 103. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROLADOR PID Y SET POINT EN 18°. ....	164
FIGURA 104. RESPUESTA REAL DEL ÁNGULO DE ELEVACIÓN CON CONTROLADOR PID Y SET POINT EN 25.2°. ....	164
FIGURA 105. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROLADOR PID Y SET POINT EN 15°. ....	165
FIGURA 106. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROLADOR PID Y SET POINT EN -15°. ....	166
FIGURA 107. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROLADOR PID Y SET POINT EN 45°. ....	167
FIGURA 108. RESPUESTA REAL DEL ÁNGULO DE ALABEO CON CONTROLADOR PID Y SET POINT EN -45°. ....	167

## LISTA DE TABLAS

TABLA 1. VALORES DE LOS PARÁMETROS.....	44
TABLA 2. INTERVALOS PARAMÉTRICOS DEL MODELO DEL ÁNGULO DE ELEVACIÓN .....	45
TABLA 3. INTERVALOS PARAMÉTRICOS DEL MODELO DEL ÁNGULO DE ALABEO .....	48
TABLA 4. SÍNTESIS DE LAS CARACTERÍSTICAS DEL SISTEMA COMPENSADO.....	65
TABLA 5. SÍNTESIS DE LAS CARACTERÍSTICAS DEL SISTEMA COMPENSADO. ALABEO.....	77
TABLA 6. SELECCIÓN DEL RODAMIENTO LINEAL KH 1228.....	94
TABLA 7. CARGA RADIAL DINÁMICA EQUIVALENTE .....	102
TABLA 8. PARÁMETROS PARA PRUEBA DE LA INTERFAZ.....	144
TABLA 9. SÍNTESIS DE LOS REQUERIMIENTOS DE DISEÑO DE LOS 2 LAZOS DE CONTROL.....	146

## RESUMEN

**TITULO:** DISEÑO Y CONSTRUCCIÓN DE UN HELIBOT 3DOF (HELICOPTERO 3GDL) COMO HERRAMIENTA DE INVESTIGACION Y DESARROLLO PARA EL LABORATORIO DE SISTEMAS DINÁMICOS E INGENIEÍA DEL CONTROL<sup>1</sup>.

**AUTORES:** ANDRÉS ALCIDES III BERMÚDEZ ROA\*\*  
MANUEL ANTONIO DIAZ DURAN\*\*

**PALABRAS CLAVE:**

Controladores PID, Helicóptero 3GDL, Control automático, Sistemas Dinámicos.

Este proyecto resulta como una respuesta ante el imponente dinamismo industrial que demanda ingenieros multidisciplinarios capaces de enfrentar los retos del control moderno, y la evidente necesidad de mostrar una aplicación de la teoría del control con un sentido pedagógico que facilite la asimilación del conocimiento por parte del estudiante así como la enseñanza por parte del educador; además se ofrezca como una plataforma de investigación para futuros desarrollos en el campo de la Ingeniería del control y Sistemas Dinámicos en la Universidad Industrial de Santander. El proyecto se resume en el diseño y construcción del Helibot 3DOF (helicóptero de 3 grados de libertad) como planta dinámica, además de la construcción de una tarjeta de adquisición de datos basada en el procesador Motorola MCF51QE128 @ 50MHZ y 32-Bit que permite el dialogo constante entre la planta dinámica y la interfaz grafica; esta tarjeta fue programada utilizando como algoritmo de control la familia de controladores PID el cual sobresale entre los diferentes algoritmos de control por la celeridad en la respuesta, por su precisión en la regulación automática, y por la flexibilidad en su estructura pues permite obtener variaciones en su esquema consistentes en la no introducción de alguno o algunos de sus términos dando como resultado cinco aprovechables combinaciones: control P, I, PI, PD y control PID, que podrán ser aplicadas en el dominio del Helibot 3DOF por medio de la implementación de una interfaz grafica que utiliza Matlab como lenguaje de programación, y que permite el monitoreo en tiempo real de las variables de estado así como la variación de las características dinámicas del controlador.

---

<sup>1</sup> Tesis de Grado

\*\* Facultad de ingenierías físicomecánicas. Escuela de ingeniería mecánica. Director: Carlos borras pinilla.

## ABSTRACT

**TITLE:** DESIGN AND CONSTRUCTION OF A HELIBOT 3DOF (HELICOPTER 3DOF) AS RESEARCH AND DEVELOPMENT TOOL FOR THE DYNAMIC SYSTEMS LABORATORY AND CONTROL ENGINEERING<sup>2</sup>

**AUTHORS:** ANDRÉS ALCIDES III BERMÚDEZ ROA\*\*  
MANUEL ANTONIO DIAZ DURAN\*\*

**KEYWORDS:**

PID controllers, 3DOF Helicopter, automatic control, dynamic systems.

This project is a response to imposing industrial dynamism that demand multidisciplinary engineers able to face the challenges of the modern control, and the obvious need to show an application of control theory with a pedagogical sense that it facilitates the assimilation of knowledge by the student and the teaching by the teacher. In addition is offered as a platform of research for future developments in the field of Engineering of control and Dynamic Systems in the Industrial University of Santander. The project is summarized in the design and construction of Helibot 3DOF (helicopter 3 degrees of freedom) as dynamic plant, besides the construction of data acquisition card based on the processor Motorola MCF51QE128 @ 50MHZ and 32-bit that allows the constant dialog between the plant dynamics and graphic interface; this card was programmed using as control algorithm the PID controller family which stands between the different control algorithms for the speed in the response, by its precision in the automatic regulation and by the flexibility in its structure because it allows to obtain variations in his scheme consisting of the introduction or not of some or some of its terms resulting in five usable combinations: Control P, I, PI, PD and control PID, which may be applied in the domain of Helibot 3DOF through the implementation of a graphic interface that uses Matlab like programming language and that allows to the monitoring in real time of the state variables as well as the variation of the controller's dynamic characteristics.

---

<sup>2</sup> Undergraduate thesis

\*\* Facultad de ingenierías físicomecánicas. Escuela de ingeniería mecánica. Director: Carlos borras pinilla.

## INTRODUCCIÓN

El gran auge que tiene en la actualidad el estudio de diferentes técnicas de control modernas y su aplicación en la industria, ha venido planteando retos importantes tanto a los educadores como a los estudiantes, razón por la cual se hace necesario comprometernos con su cuidadoso estudio así como con el fortalecimiento en la investigación de nuevas técnicas de control en la Escuela de Ingeniería Mecánica en su continuo afán de proporcionar a la industria moderna profesionales multidisciplinarios y competentes. Motivo que fundamentó el diseño y construcción de una plataforma de investigación y desarrollo en base a un Helibot 3DOF (helicóptero de 3 grados de libertad) que permita analizar y validar a futuro las diferentes leyes de control moderno (dominio del tiempo y la frecuencia), aunque inicialmente regulado por los diferentes miembros de la familia de controladores PID a través de una interfaz grafica en Matlab que permite la interacción con el usuario por medio de un entorno amable y didáctico, tratando de imprimir en todo su desarrollo un enfoque estrictamente pedagógico.

El Helibot 3DOF es un sistema dinámico flexible, versátil, que además de ser muy grafico, representa directamente muchos de los retos de la ingeniería mecánica, ingeniería del control y aeroespacial, y favorece la evaluación de las diferentes estrategias de control como la Teoría de Retroalimentación Cuantitativa (QFT), Regulador Cuadrático Lineal (LQR), las redes neuronales, lógica fuzzy y su combinación neurofuzzy entre otras.

Su principio de control se basa en estabilizar un largo brazo en torno a un pivote vertical a través de variaciones en el voltaje de dos motores DC (actuadores), instalados en un extremo del brazo que impulsan las hélices y crean una fuerza de empuje que junto con el contrapeso, ubicado en el otro extremo permiten su

estabilidad. Utiliza 2 sensores, uno para registrar las variaciones en el ángulo de la barra que sostiene los motores (alabeo), y otro para registrar la variación en el ángulo de elevación del sistema.

Utilizamos el algoritmo PID como alternativa de control dinámico de la planta, ya que consideramos, es el primer paso para introducirse en el universo de los métodos de control, teniendo en cuenta que es el más ampliamente utilizado en la industria moderna, controlando más del 95% de los procesos industriales en lazo cerrado, además destaca por la simplicidad de su estructura, celeridad en la respuesta y por ser uno de los métodos más precisos en la regulación automática. La flexibilidad en su estructura es una de las ventajas que favorece el matiz pedagógico que concedimos al proyecto, pues permite obtener variaciones en su esquema consistentes en la no introducción de alguno o algunos de sus términos dando como resultando cinco aprovechables combinaciones: control P, I, PI, PD y control PID, que podrán ser aplicadas en el dominio del helicóptero de 3 grados de libertad.

Se recurrió a Matlab como lenguaje de programación por ser un software matemático con entorno de desarrollo integrado (IDE) que se adapta a las diferentes demandas ingenieriles con un enfoque muy académico y cuenta con múltiples aplicaciones graficas que nos permitieron analizar y contrastar a través de una interfaz amable y de forma independiente las respuestas obtenidas por los diferentes miembros de las familia de controladores PID que intervienen en el dominio del Helibot 3DOF, así como la supervisión del proceso, cambios en los modelos de la planta, cambios en los algoritmos de control y cambios en los parámetros del controlador entre otros.

## ANTECEDENTES

Análisis y síntesis de los modelos de helicóptero han llamado mucho la atención de los investigadores (ver, ej., [19,20] y las referencias dentro de estos trabajos). En particular para la plataforma experimental de helicóptero de 3 grados de libertad (GDL) fabricado por Quanser se han realizado pocos trabajos relevantes. En [3] se resolvió el problema de control por retroalimentación de estados por medio de control predictivo lineal. En [9] se presentan resultados referentes a la implementación de controladores predictivos para el modelo de helicóptero linealizado alrededor del punto de equilibrio. En [17] se presentan resultados referentes al análisis de las técnicas de movimientos primitivos básicos extendidos a la teoría de contracción no lineal con la aplicación al aprendizaje de maniobras agresivas dentro del control de vuelo del robot. En [10] se presentan resultados referentes a la solución del problema de control de movimiento del prototipo de helicóptero basados en pasividad con técnicas de control adaptivo. En [11] se presentan resultados relevantes relativos a la solución del problema de seguimiento de trayectorias para helicóptero por medio de control de retroalimentación hacia adelante y retroalimentación. En [21] se presentan resultados respecto a la solución del problema de control vía internet para la supervisión y seguridad de su uso en un laboratorio. En [14] se presentan resultados relevantes relativos a la solución del problema de controlar el ángulo de cabeceo del helicóptero utilizando controlador PID. Debido a que los modelos de los helicópteros tienen cierto grado de incertidumbre y están sujetos a perturbaciones externas se hace interesante para la aplicación del popular controlador PID desde un punto de vista práctico.

# 1 FORMULACIÓN DEL PROBLEMA

## 1.1 IDENTIFICACIÓN DEL PROBLEMA

El gran auge que tiene en la actualidad el estudio de diferentes técnicas de control modernas y su aplicación en la industria, ha venido planteando retos importantes tanto a los educadores como a los estudiantes, razón por la cual su estudio y aplicación se ha venido incluyendo con más interés en los programas académicos de carreras afines con esta rama de la ingeniería.

La enseñanza del control desde el punto de vista pedagógico, está llamada a realizar enormes cambios en su estructura académica. Por una parte, el estudiante que inicia tiene una gran expectativa acerca de la aplicabilidad de la teoría de control en la ingeniería, pero por otra, rápidamente se encuentra inmerso en planteamientos matemáticos y teorías con un elevado nivel de abstracción; es por esto que muchas universidades en el mundo poseen en sus laboratorios de Sistemas Dinámicos, Ingeniería del Control y Robótica varios bancos de trabajo, los cuales muestran a escala una aplicación de la teoría del control de manera sencilla y didáctica de diferentes sistemas y procesos industriales del mundo actual.

La Escuela de Ingeniería Mecánica de la Universidad industrial de Santander, comprometida en la misión de formar Ingenieros Mecánicos que respondan a los retos tecnológicos planteados por el dinamismo industrial, no cuenta en la actualidad con herramientas didácticas de Investigación y Desarrollo tecnológico que permitan aplicar los conocimientos adquiridos en cátedras como Control Automático y Sistemas Dinámicos.

Para los docentes, no es fácil explicar y dar a entender un tema que necesita mucha aplicación para ser asimilado por parte del estudiante; en la escuela se

han hecho diferentes intentos para dar explicación de los temas a los estudiantes de una manera didáctica, pero estos equipos tienen problemas de funcionamiento y no reflejan con precisión el contenido de la temática, es por esta razón, que nosotros nos hemos propuesto como meta en este trabajo de grado, diseñar y construir un modelo para el control de un helicóptero 3GDL (3 grados de libertad), como forma de contribuir a la innovación en cuanto a la infraestructura de laboratorios se refiere, ayudando a la formación de los ingenieros emergentes y de alguna forma devolver a la universidad todo lo que hemos recibimos durante estos años.

## **1.2 JUSTIFICACIÓN PARA LA SOLUCION DEL PROBLEMA**

El control automático es una de las disciplinas que ha tenido un importante y dinámico desarrollo, constituyéndose hoy por hoy en la piedra angular de lo que algunos autores han denominado la segunda revolución industrial, su imponente avance es consecuencia del uso generalizado de las técnicas de medición y control asociadas a la industria que en su incesante carrera por reducir costos, mejorar la calidad de los productos, acelerar los procesos productivos permitiendo estandarización y exactitud, reduciendo la mano de obra pasiva y remplazándola por trabajo especializado, ha encontrado en el control automático una solución necesaria y oportuna, base esencial y soporte teórico para otras áreas de interés tecnológico y científico, como es la Robótica, Ingeniería Aeronáutica, Ingeniería Aeroespacial, la Automatización, la Instrumentación, Ingeniería del Control, los Microsistemas y sus sistemas del Control.

Con todo esto la importancia de la construcción de equipos orientados a la enseñanza del control automático de sistemas dinámicos en la Escuela de Ingeniería Mecánica, que favorezcan un aprendizaje aplicativo, interactivo, didáctico e investigativo y que permitan explorar las universalmente reconocidas

ventajas del control automático se hace evidente. El Helicóptero 3 GDL es un experimento muy útil e interesante que puede ser utilizado para explicar y demostrar en forma práctica diferentes métodos de modelado, análisis y técnicas de control modernas y clásicas.

El diseño y construcción de este equipo se prefiere sobre la compra de el mismo en el mercado por 3 principales razones: sus altos costos de adquisición, la dependencia de proveedores foráneos para su mantenimiento, y el limitado acomodo que presentan los equipos del mercado con los requerimientos que exige la formación en control del estudiante de Ingeniería Mecánica, inconvenientes que serán solucionados con la ejecución de este proyecto lo que justifica su realización.

La implementación de estos equipos didácticos surgirá como apoyo a las asignaturas de Ingeniería de Control y Sistemas Dinámicos como una manera pedagógica y directa de fortalecer los conocimientos adquiridos en su contenido, ya que el estudiante tendrá la oportunidad de practicar y aplicar lo que en un futuro podrá experimentar como profesional de la Ingeniería Mecánica.

## 1.3 OBJETIVOS DEL TRABAJO DE GRADO

### 1.3.1 Objetivo General

- Contribuir a la misión educativa impartida por la Universidad Industrial de Santander, mediante el desarrollo de una ayuda pedagógica, que ofrezca a los estudiantes de la escuela de Ingeniería Mecánica una herramienta que permita ilustrar muchas de las dificultades asociadas a los problemas de control en la industria moderna, y favorezca la posterior investigación en esta rama de la ingeniería.

### 1.3.2 Objetivos Específicos

1. Diseño, selección y construcción de la plataforma mecánica del HeliBot 3DOF. El proceso de diseño y construcción comprenderá:

- Seleccionar y adquirir los diferentes elementos que constituyen el hardware del Helibot 3 DOF, como son:
  - Encoder Ópticos (entre 2000 y 3000 pulsos/revolución, diámetro < 40mm).
  - Motores DC (12V y 20A).
  - Hélices (3 aspas, 13cm < diámetro < 23cm)
- Diseño y construcción de la estructura dinámica del HeliBot 3DOF. Para este diseño se utilizaran herramientas de apoyo CAD y CAE, tales como Solidworkss y SimulationXpress study. Este diseño comprenderá:
  - ✓ Diseño y construcción de la mesa en la cual va soportado la planta dinámica con dimensiones:  $3000\text{cm}^2 < \text{Área de trabajo} < 7000\text{cm}^2$ ;  $100\text{cm} < \text{Altura} < 130\text{cm}$ .

- ✓ Diseño y construcción de la estructura de apoyo de la planta dinámica del HeliBot 3DOF con dimensiones:  
100cm < Largo < 130cm;  
40cm < Ancho < 60cm;  
40cm < Alto < 60cm.

1. Diseñar y construir una tarjeta de control embebida, basada en el procesador Motorola MCF51 QE128 @ 50 MHz y 32-Bit para el control de la planta del Helibot 3 DOF.
2. Diseño y programación del algoritmo de control para realizar pruebas, basado en el popular controlador PID.
3. Realizar una fase de pruebas y ajustes del controlador.
4. Diseño e implementación de una interfaz grafica en Matlab para el monitoreo en tiempo real de las variables de estado que permita cambiar las características dinámicas de los lazos de control y que además permita almacenar datos con facilidad para el análisis posterior de los resultados (off-line).
5. Realizar y documentar pruebas del sistema dinámico con el fin de:
  - Analizar el tiempo de respuesta y el comportamiento del Helicóptero de 3 grados de libertad al someterlo a diferentes magnitudes de contrapeso.
  - Obtener y contrastar los resultados gráficos de la relación existente entre la variable controlada y las variables a controlar en el helicóptero de 3 grados de libertad (ángulo de elevación y ángulo de alabeo).

## 2 MODELO ESTRUCTURAL DEL HELIBOT COMO SISTEMA DE CONTROL AUTOMÁTICO

Helibot 3DOF es un proyecto enfocado al desarrollo de una planta dinámica para posteriores estudios de diversos métodos de control, sin pretender profundizar en todos los aspectos que esto implica. Se busca diseñar e implementar un control simple para controlar únicamente el ángulo de elevación y alabeo con respecto a un punto o set point, restringiendo el rango de altura y alabeo del sistema.

Para términos de control la variable manipulada serán los anchos de pulsos (PWM) enviados a los motores y la variable controlada serán los ángulos de elevación y alabeo, se utilizó como sistema de control la familia de controladores PID, por su sencillez y flexibilidad. Los sensores, actuadores y en general la planta física se definieron más adelante a lo largo de esta capítulo. Así mismo, se busca hacer las pruebas del control en la planta dinámica analizando el comportamiento del error ante distintos parámetros del controlador PID.

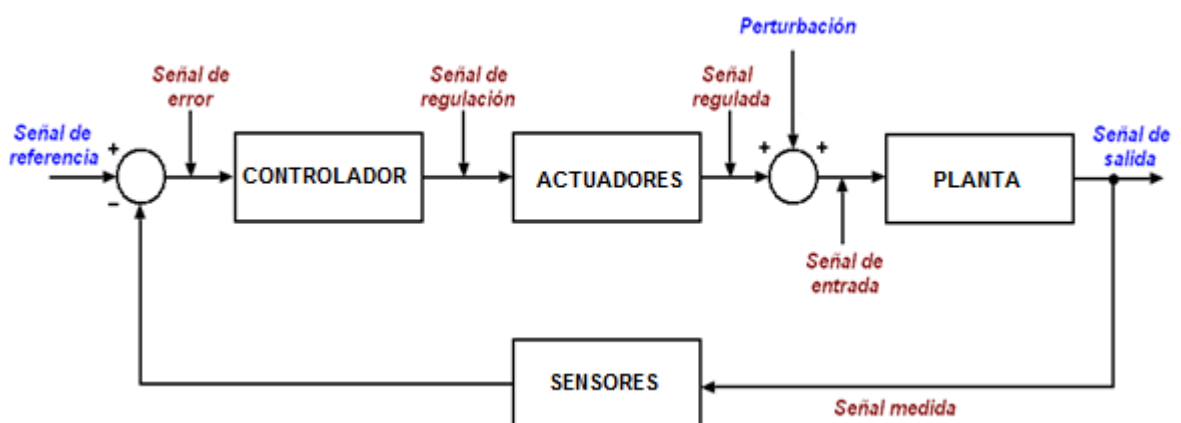
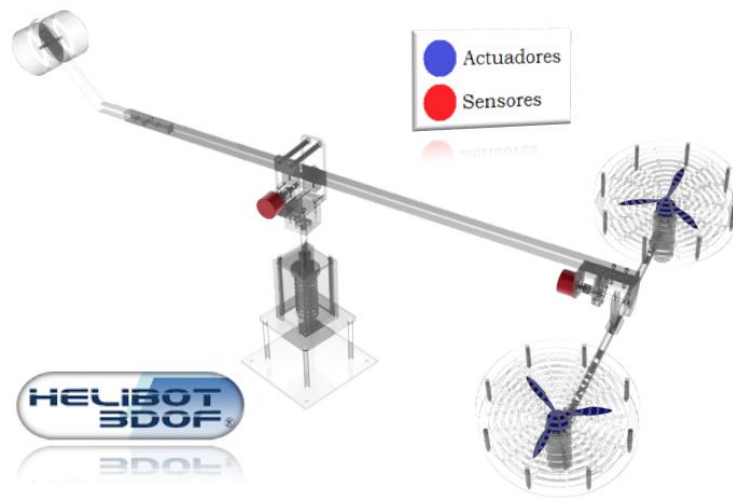


FIGURA 1. Diagrama de bloques del sistema automático

Se definieron los componentes del sistema automático compuestos por el diagrama de bloques citado anteriormente, en capítulos posteriores se realizó una completa descripción de estos elementos.

## 2.1 PARTE OPERATIVA

Como se mostro en la figura 1, la parte operativa de un sistema automatizado está compuesta por el conjunto (Maquina, sensores y actuadores), los cuales han sido dispuestos en una estructura fabricada en perfiles de Aluminio Estructural, los cuales soportan estos elementos y comprenden la estructura central del Helibot 3 DOF, en la figura 2 se desarrollo un diseño esquemático de la parte operativa del banco.



**FIGURA 2. Diseño Esquemático del banco**  
Fuente: Los Autores

Este diseño consta de 2 encoders incrementales, los cuales registran el ángulo de alabeo y el otro censa el ángulo de elevación, el ángulo de avance no se censa este se toma como una salida producto de los 2 movimientos combinados. La fuerza motriz está representada por los actuadores, los cuales son dos motores

de corriente continua, controlados con un control de velocidad, estos componentes se seleccionaran más adelante.

### 2.1.1 La planta dinámica

La maquina en este caso está representada por el HeliBot 3 DOF. Como se pudo ver en la selección de alternativas en el Anexo A, este sistema automático nos proporciona las características pedagógicas necesarias para enseñar de una manera didáctica los alcances de la familia de controladores PID, al ser un sistema inestable-No lineal y presenta alta complejidad en el control, se nos presento como un reto, pero encajaba perfectamente en la premisa inicial del proyecto, el cual era la construcción de una plataforma dinámica para posterior investigación y desarrollo en la ingeniería de control. En la figura 3 se muestra el banco de pruebas del Helibot 3 DOF.

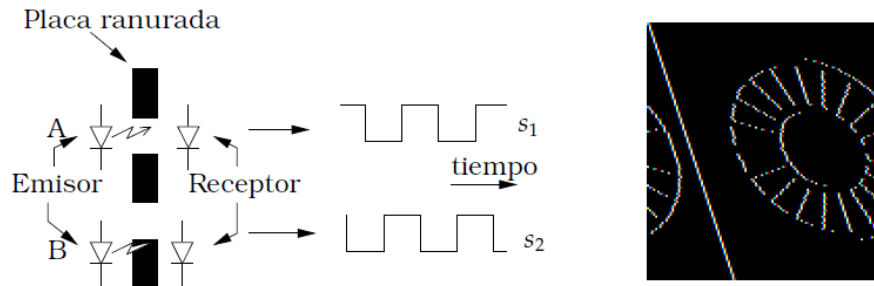


FIGURA 3. Planta dinámica de pruebas

### 2.1.2 Sensores

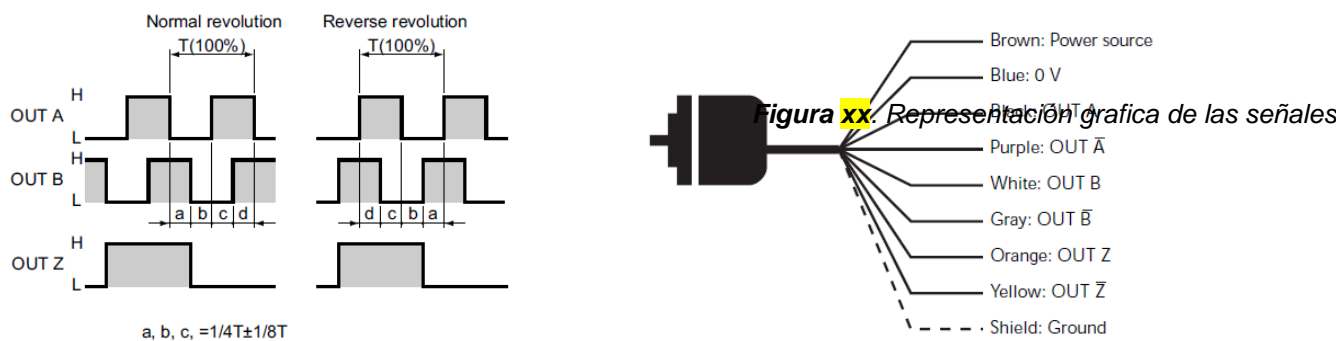
Para el control de la planta dinámica del HeliBot 3 DOF, se utilizaron 2 sensores, el primero se encarga de medir el ángulo de alabeo de la barra que soporta el bloque motor-hélice, y el segundo censa el ángulo de elevación de la planta. El método de medición utilizado es el mismo para los dos sensores. La descripción detallada de los sensores la encontramos en el anexo B. Utilizamos encoders incrementales pues tiene un rango de medición ilimitado, aunque debido a que los pulsos entregados por el encoder son de alta frecuencia es posible que al contarlos se pierdan pulsos. La lógica encargada de contar los pulsos fue implementada en la PC, utilizando cuatro entradas digitales de la placa adquisidora (dos por sensor) para leer los pulsos de los encoders.

El encoder es un transductor rotativo que transforma un movimiento angular en una serie de impulsos digitales, en nuestro caso estos impulsos generados son utilizados para controlar el desplazamiento angular. La detección del movimiento angular se ejecuta en base al principio de exploración fotoeléctrica. El sistema de lectura se basa en la rotación de un disco graduado con un reticulado radial formado por líneas opacas, alternados con espacios transparentes. Este conjunto está iluminado de modo perpendicular por una fuente de rayos infrarrojos. El disco proyecta de este modo su imagen sobre la superficie de varios receptores oportunamente enmascarados por otro reticulado que tiene el mismo paso que el anterior llamado colimador. Los receptores tienen la tarea de detectar las variaciones de luz que se producen con el desplazamiento del disco convirtiéndolas en las correspondientes variaciones eléctricas. En la figura 4 se muestra el funcionamiento de un encoder incremental.



**FIGURA 4.** Diagrama de un encoder incremental  
Fuente: Mazzone Virginia, Mallo Sebastián; Diseño y construcción de un péndulo invertido rotativo

El encoder incremental proporciona normalmente dos formas de ondas cuadradas y desfasadas entre sí en  $90^\circ$  eléctricos, los cuales por lo general son “canal A” y “canal B”. Con la lectura de un solo canal se dispone de la información correspondiente a la velocidad de rotación, mientras que si se capta también la señal “B” es posible discriminar el sentido de rotación en base a la secuencia de datos que producen ambas señales. Está disponible además otra señal llamado canal Z, que proporciona la posición absoluta de cero del eje del encoder. Esta señal se presenta bajo la forma de impulso cuadrado con fase y amplitud centrada en el canal A. En la figura 5 se muestran los canales de sincronización del sensor.



**FIGURA 5.** Canales de sincronización del encoder incremental  
Fuente: [www.automationdirect.com/encoders](http://www.automationdirect.com/encoders)

El sensor utilizado tiene las siguientes características técnicas:

Marca: Automation Direct

Modelo: TRD-S2500-VD

Voltaje de Operación: +4.75 -5.25 VDC

Resolución: 2500 ppr

Consumo de corriente: 50mA máx.

Frecuencia de respuesta: 200kHz

### **TRD-S2500-VD**



**FIGURA 6.** Encoder incremental TRD-S2500-VD  
Fuente: [www.automationdirect.com/en](http://www.automationdirect.com/en)

### **2.1.3 Actuator**

En nuestro sistema, la variable a controlar será la tensión consumida por los motores los cuales dependiendo de las revoluciones que generen determina la fuerza de empuje ejercida por las hélices que me permitirán estabilizar el sistema en torno a un punto de equilibrio.

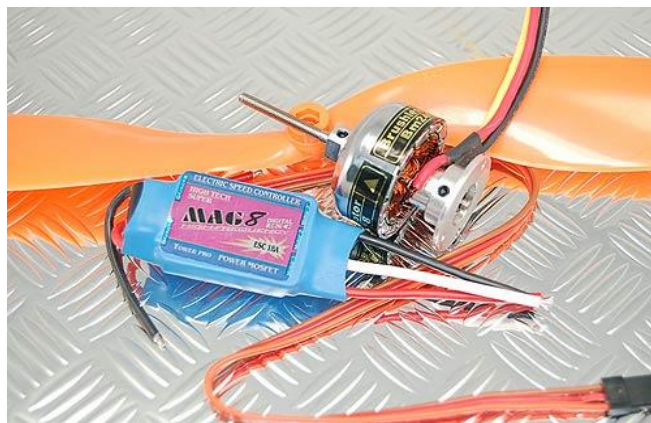
### **Motor Brushless**

Entre las alternativas posibles a utilizar como motor para este sistema se encuentran los motores brushed (con escobillas), y los motores brushless (sin escobillas), este último presenta mayores ventajas sobre el primero las cuales mencionare a continuación:

- Mayor eficiencia (menos pérdida por calor)
- Mayor rendimiento (mayor duración de las baterías para la misma potencia)
- Conmutación electrónica basada en sensores de posición de efecto Hall

- Requieren menos mantenimiento al no tener escobillas
- Relación velocidad/par motor es casi una constante
- Mayor potencia para el mismo tamaño
- Mejor disipación de calor
- Rango de velocidad elevado al no tener limitación mecánica.
- Menor ruido electrónico (menos interferencias en otros circuitos).

Los motores brushless tienen la característica de que no emplean escobillas en la conmutación para la transferencia de energía; en este caso, la conmutación se realiza electrónicamente. Esta propiedad elimina el gran problema que poseen los motores eléctricos convencionales con escobillas, los cuales producen rozamiento, disminuyen el rendimiento, desprenden calor, son ruidosos y requieren una sustitución periódica y, por tanto, un mayor mantenimiento.



**FIGURA 7. Conjunto Brushless – Speed Control**  
Fuente: [www.hobbyking.com](http://www.hobbyking.com)

Estos motores se validaron en el laboratorio y los resultados fueron 11000 Rpm @ 16 A y 11.5 V.

## Hélice

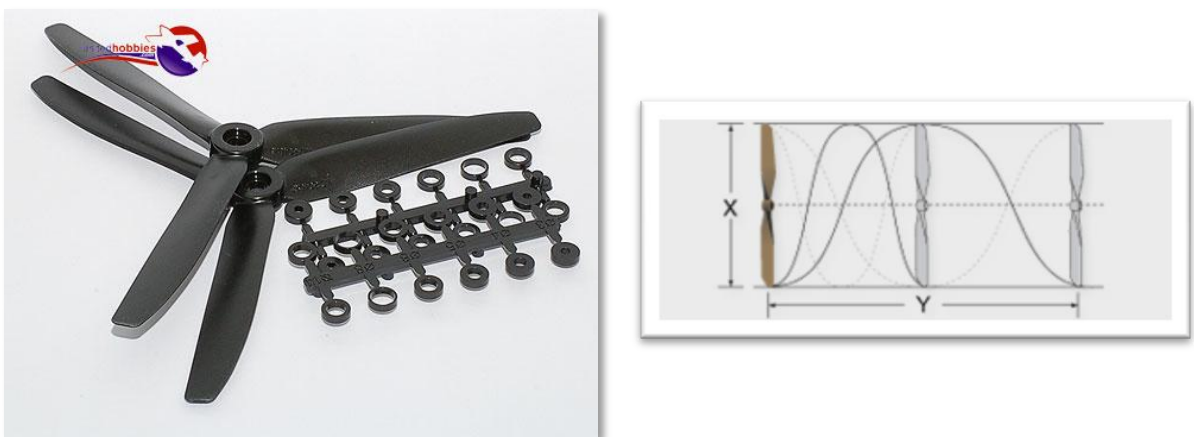
Seleccionamos hélices de aerodelismo cuya geometría favoreciera la generación de una fuerza de empuje entre 800g y 1200g cuando el motor gire a su máxima revolución de aproximadamente 11000 rpm.

Se validaron las siguientes hélices para calcular el empuje producido por estas a las siguientes condiciones: 12.15pA, 143pW y 10350 Rpm y los resultados obtenidos fueron

GWS HD8040 2 Blade Prop      Fuerza de Empuje promedio: 610 g

GWS HD8040 3 Blade Prop      Fuerza de Empuje promedio: 810 g

En el Anexo C, se presentan unas recomendaciones de helices para los motores towerpro brushless, se decidio seleccionar las helices de referencia GWS HD8040, ya que estas nos da el empuje necesario cuando el motor esta funcionando a 10350 Rpm.



**FIGURA 8.** Hélices GWS HD8040  
Fuente: [www.hobbyking.com](http://www.hobbyking.com)

Especificaciones técnicas:

Length (Inch [X]) = 8

Paso (Inch [Y]) = 4

Empuje: 810 g @ 10350 Rpm.

Peso: 25 g

## **2.2 PARTE DE CONTROL**

La parte de control en la planta dinámica del HeliBot 3DOF está constituida por el preactuador y el controlador junto con el dialogo y las comunicaciones con otros sistemas.

### **2.2.1 Preactuador**

Se hizo necesario para el control de la velocidad de los motores brushless la implementación de un speed control (Regulador de Velocidad), el cual puede ser seteado a una cierta cantidad de vueltas, supongamos 7000 Rpm, para la cual debería entregar cierta corriente programada. Si la carga aumenta, como podría ser el caso de un rotor de helicóptero, entonces el regulador, debería entregar más corriente para que se mantengan esas 7000 Rpm. El regulador controla la forma del pulso contra electromotriz (EMF) y así ajusta la corriente de salida.

Es el dispositivo encargado de convertir la alimentación continua en trifásica. Además controla de velocidad del motor mediante una señal PWM que proviene del control ubicado en tierra. Para su selección, es necesario que el control de velocidad al menos proporcione la corriente máxima de alimentación que el motor requiera.

El control de velocidad es marca TURNIGY modelo TR W18A. Estos dispositivos soportan hasta 18 [A] y cuentan con un circuito eliminador de batería BEC.



**Amp rating:** 18A (max 20A burst)  
**FETs:** 12  
**BEC Current:** 1.5A  
**Timing:** Auto  
**Frecuencia:** 8khz  
**Lipo:** 3 cell lipo  
**NiMH:** 6~16  
**Peso:** 19.5g (22 with wires)

FIGURA 9. *TowerPro w18A Brushless Speed Controller*

Fuente: [www.hobbyking.com](http://www.hobbyking.com)

## 2.2.2 Controlador

Se utilizó un microcontrolador *ColdFire core V1* de la serie *Flexis QE* del fabricante *Freescale* (Antes Motorola), con un microcontrolador MCF51QE128, el cual se adaptó a la planta dinámica del HeliBot 3 DOF mediante el algoritmo de programación en lenguaje C++ utilizando el software *CodeWarrior*.

### MCF51QE128 Coldfire V1 de freescale



FIGURA 10. *Microcontrolador MCF51QE128 Coldfire V1*

Fuente: [www.freescale.com](http://www.freescale.com)

## Características

- 50.33 MHz ColdFire core V1 /F bus 25 MHz
- Memoria 128 KB (flash)
- Memoria 16 KB (RAM)
- Rango de operación de voltaje 2.7 a 5.5V
- Reloj interno altamente exacto (ICS)
- 16 pines (A/D)
- Rango de temperatura -40°C a +105°C
- Encapsulados entre 64 y 80 pines QFP
- Compatibilidad pin a pin entre versión 8 bits y 32 bits.
- 24-canales, 12-bit ADC
- 2 Comparadores análogos
- 2 Interface serial de comunicaciones (SCI/UART)
- 2 Periféricos seriales(SPI)

### 3 MODELADO MATEMATICO DEL SISTEMA

La planta dinámica a controlar se representa en la figura 11, consiste de un brazo montado sobre una base con un conjunto hélice-motor atado a cada final de una barra pivotada en su centro y un contrapeso en su otra punta. El contrapeso se calcula de tal forma que el bloque hélice motor quede suspendido en la mesa y se modifica la distancia del contrapeso a la base hasta que el sistema quede en equilibrio. Las hélices se encuentran fijas sobre una barra la cual puede rotar alrededor del brazo principal, las cuales crean una variación en el ángulo de alabeo  $\alpha$ , si hay una diferencia en el voltaje aplicado a los motores. El resultado de el ángulo  $\alpha$  es medido desde un plano horizontal, asumiendo  $\alpha \approx 0$ , un voltaje igual enviado a los motores causa una rotación en la viga principal y por tanto una variación en el ángulo de elevación  $e$ . En adicción, la viga principal puede girar alrededor de la base generando un cambio en el ángulo de vuelo  $v$ , para generar una variación en este eje, primero es necesario crear un ángulo  $\alpha = 0$ , y luego aplicar un voltaje a ambos motores para generar una fuerza en la dirección deseada.

Por otra parte, los actuadores se reducen a dos motores eléctricos de tensión continua acoplados a cada lado del eje longitudinal que junto con sendas hélices producen una fuerza proporcional al ancho de pulso PWM (*Pulse Width Modulation*) del motor, y forman parte del cuerpo del sistema. Para presentar el control de estos ejes, en esta propuesta se presenta el modelo matemático del HeliBot 3 DOF.

Una de las razones para ejercer el análisis dinámico del sistema es para conseguir una ecuación que nos relacione la variable de entrada con la variable de salida, esta ecuación es popularmente conocida como función de transferencia la cual nos ayudara a diseñar nuestro controlador en el siguiente capítulo.

### 3.1 ECUACIONES DINAMICAS DE MOVIMIENTO

#### 3.1.1 Angulo de elevación o pitch

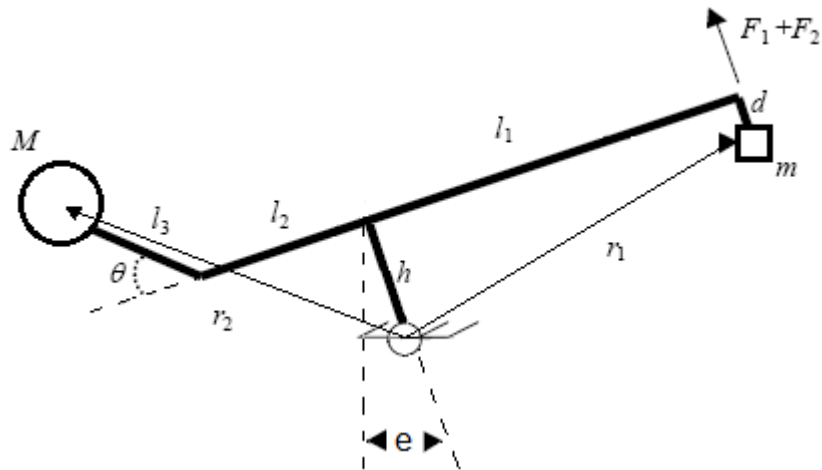


FIGURA 11. Geometría del ángulo de elevación

Teniendo el diagrama para analizar la dinámica del HeliBot 3 DOF, procedemos a descomponer las fuerzas que nos generan torque con respecto al punto fijo mostrado en la figura 11.

Tal y como se puede deducir a partir de las ecuaciones de LaGrange, el modelo matemático es el siguiente:

$$\sum T = I * \ddot{e} \quad (3.1)$$

$$(F_1 + F_2) * l_1 - mg[(d - h) \sin e + l_1 \cos e] + Mg \cos e (l_2 + l_3 \cos \theta) + Mg \sin e (h + l_3 \sin \theta) - b_e = I_e * \ddot{e} \quad (3.2)$$

Siendo  $h, d, l_1, l_2, l_3$ , longitudes del sistema,  $m$ , la suma de la masa de los dos motores y  $M$  la masa del contrapeso;  $b_e$ , el rozamiento dinámico de la elevación;  $g$ , la aceleración de la gravedad.

### **Función de transferencia**

Tal y como se observa, se trata de una ecuación diferencial que es necesario linealizar en torno al ángulo de elevación  $e = 0$ .

$$(F_1 + F_2)l_1 - mg(d - h)e + Mg(h + l_3 \sin \theta)e - b_e \dot{e} = I_e \ddot{e} \quad (3.3)$$

$$(F_1 + F_2)l_1 - e[mg(d - h) - Mg(h + l_3 \sin \theta)] - b_e \dot{e} = I_e \ddot{e} \quad (3.4)$$

$$(F_1 + F_2) \frac{l_1}{I_e} = \ddot{e} + \frac{b_e}{I_e} * \dot{e} + e \frac{[mg(d-h) - Mg(h + l_3 \sin \theta)]}{I_e} \quad (3.5)$$

$$F_{suma}(s) * \frac{l_1}{I_e} = s^2 e(s) + \frac{b_e}{I_e} * s * e(s) + e(s) * \frac{[mg(d-h) - Mg(h + l_3 \sin \theta)]}{I_e} \quad (3.6)$$

Factorizando  $e(s)$  y reorganizando la ecuación, tenemos la función de transferencia, donde la entrada al sistema es la suma de las fuerzas provocadas por las hélices accionadas por los dos motores eléctricos, y la salida es el ya mencionado ángulo de elevación,  $e$ .

$$\frac{e(s)}{F_{suma}(s)} = \frac{\frac{l_1}{I_e}}{s^2 + \frac{b_e}{I_e} * s + \frac{[mg(d-h) - Mg(h + l_3 \sin \theta)]}{I_e}} \quad (3.7)$$

Calculamos el momento de inercia en el eje de elevación, teniendo en cuenta las figuras 11 y 12.

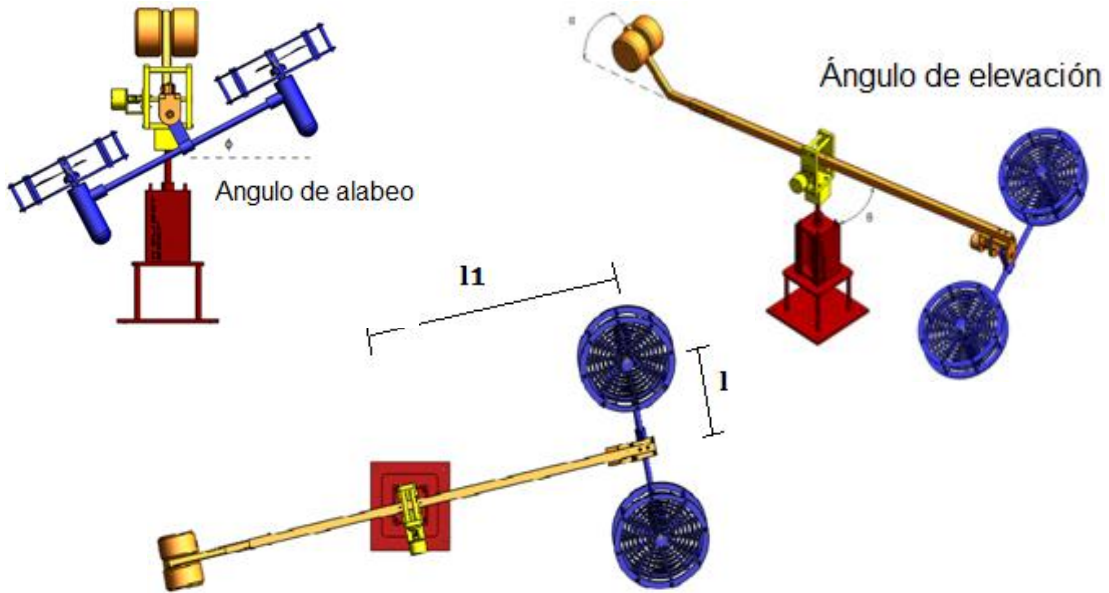


FIGURA 12. Modelo Dinámico del HeliBot 3 DOF

$$I_e = m(l_1^2 + l^2) + m_b(f^2) + M[(l_3 \sin \theta)^2 + (l_2 + l_3 \cos \theta)^2] \quad (3.8)$$

Reemplazando los valores dados en la tabla 1 en la ecuación 3.9, el momento de inercia en el eje de elevación es:

$$I_e = 1.0213 \quad kg * m^2 \quad (3.9)$$

Parámetro	Valor	Unidad	Descripción
$m$	1.296	Kg	masa del conjunto rotor
$M$	1.319	Kg	masa del contrapeso
$Mb$	0.177	Kg	masa de la barra principal
$L_1$	0.67	m	longitud desde el punto del pivote al cuerpo del helibot
$L_2$	0.33	m	longitud desde el final de la barra a el cuerpo del helibot
$L_3$	0.235	m	longitud del contrapeso al final de la barra
$l$	0.22	m	longitud desde el punto del pivote al rotor
$d$	0.09	m	longitud del pivote a la barra que sostiene el conjunto rotor
$h$	0.02	m	longitud desde el eje del ángulo de elevación hasta la barra principal
$c$	0.04	m	longitud desde la barra que sostiene el conjunto rotor y la hélice
$f$	0.17	m	longitud desde la mitad de la barra principal y el momento de inercia de esta
$I_e$	10.213	Nm	momento de inercia en el eje de elevación
$I_a$	0.06596	Nm	momento de inercia en el eje de alabeo

TABLA 1. Valores de los parámetros

## Modelo experimental

A partir de la ecuación 3.7 y conociendo que la suma de las fuerzas ejercidas por las dos hélices se supone proporcional, mediante la constante  $km$ , a la suma de tensiones eléctricas aplicadas a cada motor, se puede deducir la siguiente ecuación:

$$\frac{e(s)}{V(s)} = \frac{k_m l_1}{J_e s^2 + b_e s + [mg(d-h) - Mg(h+l_3 \sin \theta)]} \quad (3.10)$$

Esta función de transferencia relaciona el ángulo de elevación con la tensión eléctrica, y advirtiéndose que los motores presentan una saturación en su comportamiento en +/- 10 voltios que representaría el 100% de la cantidad de energía que se envía, es decir el 100% de PWM (ancho de pulso), se obtuvo una ecuación de segundo orden que se puede generalizar con la expresión.

$$\frac{e(s)}{\%PWM(s)} = \frac{k w_n^2}{s^2 + 2\xi w_n s + w_n^2} \quad (3.11)$$

En donde  $w_n$  es la frecuencia natural  $\xi$  es el coeficiente de amortiguamiento. A partir de los parámetros físicos tales como longitudes, masas y rozamientos del Helibot 3DOF estimados en la tabla 1, se puede estimar los parámetros del modelo lineal (3.11). La tabla 2 muestra los resultados, considerando la posible incertidumbre paramétrica asociada al modelo.

Parámetro	Mínimo	Máximo
$k$	0.3	1.2
$w_n$	0.55	0.60
$\xi$	0.1	0.16

TABLA 2. Intervalos paramétricos del modelo del ángulo de **elevación**

Para nuestro sistema seleccionamos  $k = 1.13$ ,  $w_n = 0.58$  y  $\xi = 0.11$ , reemplazando estos valores en la ecuación 3.11 se obtuvo la siguiente función de transferencia:

$$\frac{\alpha(s)}{V(s)} = \frac{0.3801}{s^2 + 0.128s + 0.336} \quad (3.12)$$

### 3.1.2 Angulo de alabeo

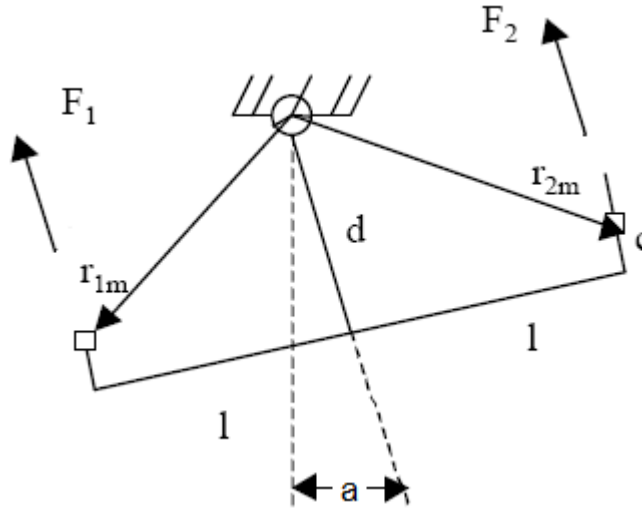


FIGURA 13. Geometría del ángulo de alabeo

$$\sum T = I * \ddot{a} \quad (3.13)$$

$$(F_2 - F_1) * l - m_1 g \sin a (d - c) + m_1 g l \cos a - m_2 g l \cos a - m_2 g \sin a (d - c) - b_p \dot{a} = I_p \ddot{a} \quad (3.14)$$

$$\text{Como } m_1 = m_2 = m \quad (3.15)$$

$$(F_2 - F_1) * l - mg \sin a (d - c) + mgl \cos a - mgl \cos a - mg \sin a (d - c) - b_a \dot{a} = I_p \ddot{a} \quad (3.16)$$

$$(F_2 - F_1) * l - 2mg \sin a (d - c) - b_a \dot{a} = I_p \ddot{a} \quad (3.17)$$

## Función de transferencia

Linealizando la ecuación de Euler-Lagrange con respecto a  $a = 0$

$$(F_2 - F_1) * l - 2mg * a * (d - c) - b_p \dot{a} = I_p \ddot{a} \quad (3.18)$$

$$F_{resta}(s) * \frac{l}{I_a} = s^2 a(s) + \frac{b_a}{I_a} * s * a(s) + \frac{2mg(d-c)}{I_a} * a(s) \quad (3.19)$$

Factorizando  $a(s)$  y reorganizando la ecuación, tenemos la función de transferencia, donde la entrada al sistema está constituida por las fuerzas  $F_1$  y  $F_2$  provocadas por las acciones de las hélices impulsadas por los dos motores eléctricos, y la salida es el ángulo de alabeo,  $a$ .

$$\frac{A(s)}{F_{resta}(s)} = \frac{\frac{l}{I_a}}{s^2 + \frac{b_a}{I_a} s + \frac{2mg(d-c)}{I_a}} \quad (3.20)$$

Ahora calculamos el momento de inercia en el eje de alabeo, teniendo en cuenta las figuras 12 y 13.

$$I_a = m[l^2 + (d - c)^2] \quad (3.21)$$

Reemplazando los valores dados en la tabla 1, el momento de inercia en el eje de alabeo es:

$$I_a = 0.06596 \text{ kgm}^2 \quad (3.22)$$

### 2.1.1.1. Función de transferencia del ángulo de cabeceo

La ecuación 3.20 Se puede generalizar (al igual que la del ángulo de elevación) en una ecuación de transferencia de segundo orden (ecuación 3.11), en la cual los parámetros que la definen se estimaron a través de las condiciones técnicas y dimensionales de la planta dinámica. La tabla 3 muestra los resultados, considerando la posible incertidumbre paramétrica asociada al modelo.

Parámetro	Mínimo	Máximo
$k$	0.9	2.7
$w_n$	0.039	0.60
$\xi$	1.90	2.28

TABLA 3. Intervalos paramétricos del modelo del ángulo de alabeo

Para nuestro sistema seleccionamos  $k = 2$ ,  $w_n = 2.1$  y  $\xi = 0.068$ , reemplazando estos valores en la ecuación 3.11 se obtuvo la siguiente función de transferencia.

$$\frac{\alpha(s)}{V(s)} = \frac{8.820}{s^2 + 0.2856s + 4.41} \quad (3.23)$$

## 4 DISEÑO DEL CONTROLADOR

### 4.1 INTRODUCCION AL CONTROL PID

En el universo del control automático existen infinidad de alternativas que responden de forma eficaz a las necesidades de nuestra planta, nosotros en el desarrollo de este proyecto hemos escogido la familia de controladores PID ya que son de estructura simple (aunque su simpleza puede ser también su debilidad), son los más usados en la industria, además su estudio sugiere el primer paso para incursionar con sólidos fundamentos en el ámbito de la ingeniería de control. En este capítulo mostraremos los enfoques tradicionales en el diseño de controladores PID aplicados a la planta del Helibot 3DOF, estudiaremos diferentes opciones de control, y además con el apoyo de MatLab y SISOtool diseñaremos un controlador que cumpla con todos las exigencias que el Helibot requiere.

El controlador PID (Proporcional, Integral y Derivativo) es un controlador realimentado cuyo propósito es hacer que el error en estado estacionario, entre la señal de referencia y la señal de salida de la planta, sea cero de manera asintótica en el tiempo, lo que se logra mediante el uso de la acción integral. Además el controlador tiene la capacidad de anticipar el futuro a través de la acción derivativa que tiene un efecto predictivo sobre la salida del proceso<sup>3</sup>.

Los controladores PID son suficientes para resolver nuestro problema de control ya que la dinámica del proceso lo permite (el Helibot 3DOF es un proceso que puede ser descrito con una dinámica de segundo orden) y los requerimientos de diseño se limitan a especificaciones del comportamiento del error en estado

---

<sup>3</sup> Apuntes de control PID. Ing. Mauricio Améstegui Moreno, Universidad Mayor de San Andrés; La Paz – Bolivia. Enero de 2001.

estacionario y a una rápida respuesta a cambios en la señal de referencia. Su uso es suficiente para resolver el problema de control de muchas aplicaciones en la industria a punto tal que el 95% de los lazos de control que existen en estas aplicaciones son del tipo PID, de los cuales la mayoría son controladores PI, lo que muestra la preferencia del usuario en el uso de leyes de control muy simples. Los controladores PID son usados generalmente en aplicaciones con modestos requerimientos y a un nivel de control inferior en comparación con algunos de los dispositivos de mediano nivel como supervisores, PLCs y sistemas de monitoreo. Sin embargo, su importancia es tal que se convierte en el “pan de cada día” del ingeniero de control. El controlador PID del Helibot está basado en un microprocesador, que proporciona una mayor flexibilidad debido a su programabilidad. El microprocesador ha tenido una importante influencia en el desarrollo del controlador PID ya que favorece la implementación de funciones adicionales como el ajuste automático de parámetros y los cambios de modos de control.

Uno de los objetivos de este proyecto es evitar el desconocimiento acerca de los detalles involucrados en la construcción del algoritmo de control PID ya que el ajuste de estos controladores es un trabajo tedioso que requiere cierta intuición basada en los principios de funcionamiento tanto de los procesos físicos controlados como de la misma teoría de control, por esta razón en este capítulo se hará un análisis, diseño y posterior ajuste de nuestro controlador en base a los requerimiento de funcionamiento del sistema, para su posterior confirmación en planta del Helibot.

## **4.2 Estructura del Control PID**

Consideremos un lazo de control de una entrada y una salida (SISO) de un grado de libertad, el Helibot 3DOF aunque es un sistema de múltiples entrada y múltiples

salidas (MIMO) puede ser descrito en 2 sistemas SISO que corresponden al ángulo de elevación y al ángulo de cabeceo ya que el ángulo de avance está en función del anterior.

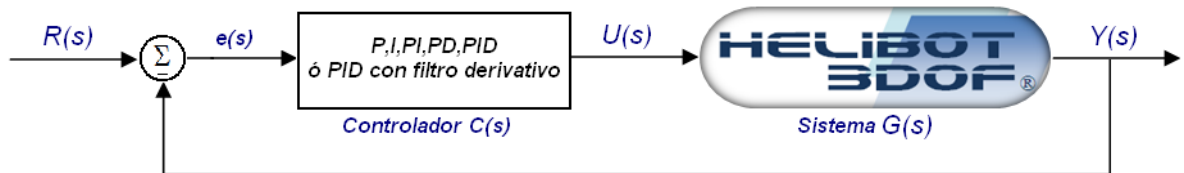


FIGURA 14. Diagrama de bloques del sistema de control del proceso.

En este diagrama el sistema y el controlador están representados por cajas y las flechas denotan las entradas y salidas a cada bloque. Note que existe un símbolo especial que denota una suma de señales. El diagrama de bloques muestra que el sistema y el controlador están conectados en un lazo realimentado.

Los miembros de la familia de controladores PID, incluyen tres acciones: proporcional (P), integral (I) y derivativa (D). Estos controladores son los denominados P, I, PI, PD y PID ó PID con filtro derivativo.

### 4.3 DISEÑO DEL CONTROLADOR DEL ANGULO DE ELEVACIÓN

#### 4.3.1 Requerimientos de diseño

Queremos diseñar un controlador realimentado de modo que la salida tenga un sobrepico menor que 25%, tiempo de subida menor que 3 segundos, tiempo de establecimiento menor a 15 segundos, y error de estado estacionario menor a 2%. Por ejemplo, si la entrada es 1 rad (escalón unitario), el ángulo de inclinación vertical no excederá los 1.2 rad, llegando a 1 rad antes de los 3 segundos, y

permaneciendo dentro del 2% del estado estacionario a los 15 segundos, o sea permanece entre 0.98 a 1.02 rad a partir del estado estacionario.

- Sobrepico: Menor que el 20%
- Tiempo de Subida: Menor que 3 segundos
- Tiempo de establecimiento: Menor que 15 segundos
- Error de estado estacionario: Menor que el 2%

#### **4.3.2 Análisis de la respuesta transitoria**

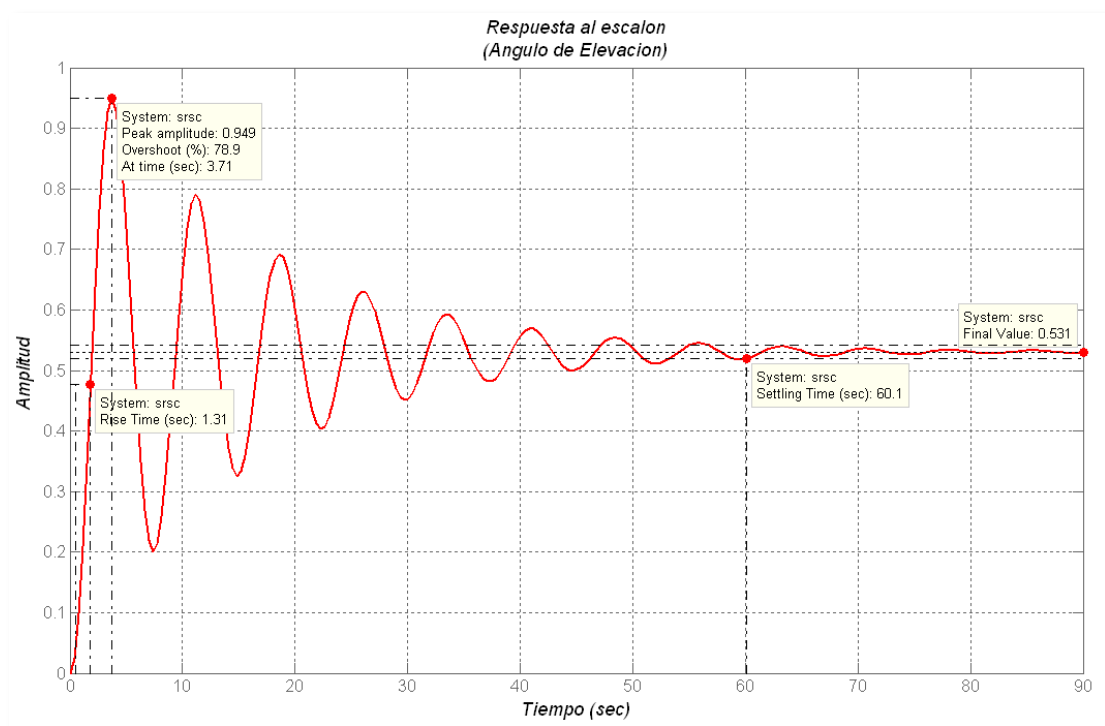
En nuestro proceso no conocemos la señal de entrada para el sistema de control, solo sabemos que es de naturaleza aleatoria, por esta razón utilizamos señales de prueba ofrecidas en Matlab ya que existe gran correlación entre las características de respuesta de un sistema para una señal de entrada de prueba común y la capacidad del sistema de manejar las señales de entrada reales. Para investigar las características en el dominio del tiempo de los sistemas de control utilizamos la señal escalón unitario (step), ya que el sistema está sujeto a perturbaciones repentinas; bajo algunas condiciones el sistema puede estar sujeto a entradas de choque para lo que utilizaremos una señal impulso (impulse) con las que validaremos del diseño de nuestros controladores, tanto para el ángulo de elevación como para el de alabeo.

#### **4.3.3 Función de Transferencia del ángulo de elevación**

En el capítulo anterior encontramos una ecuación de transferencia que relaciona el ángulo de elevación con la tensión eléctrica a partir de algunos parámetros técnicos y dimensionales del sistema, esta ecuación es de segundo orden y se define a continuación:

$$\frac{\alpha(s)}{V(s)} = \frac{0.038}{s^2 + 0.128s + 0.336}$$

A partir de esta función de transferencia y utilizando como señal de prueba la función escalón (step), podemos analizar el comportamiento del ángulo de elevación retroalimentado sin compensar, que se describe figura 15.



**FIGURA 15.** Respuesta al escalón del ángulo de elevación sin regulación

El sistema se estabiliza en un valor final de 0.531 (rad) necesitamos que es valor final este entre 0.98 (rad) y 1.02 (rad), además presenta un sobrepico de 78.9%, y un tiempo de estabilización de 60.1 seg muy superior a nuestros requerimientos de diseño, a continuación aplicaremos las acciones de control necesarias para satisfacer las exigencias del sistema.

#### 4.3.4 Análisis de los diferentes métodos de control PID

##### Control Proporcional

Para un controlador con acción de control proporcional, la relación entre la salida del controlador  $y(t)$  y la señal de error  $e(t)$  en cantidades transformadas por el método de Laplace es:

$$\frac{Y(s)}{E(s)} = K_p$$

En donde  $K_p$  es la ganancia proporcional. Es decir, la acción de control es simplemente proporcional al error de control. En la figura 16 vemos como se comporta el sistema al ajustar el  $K_p$ .

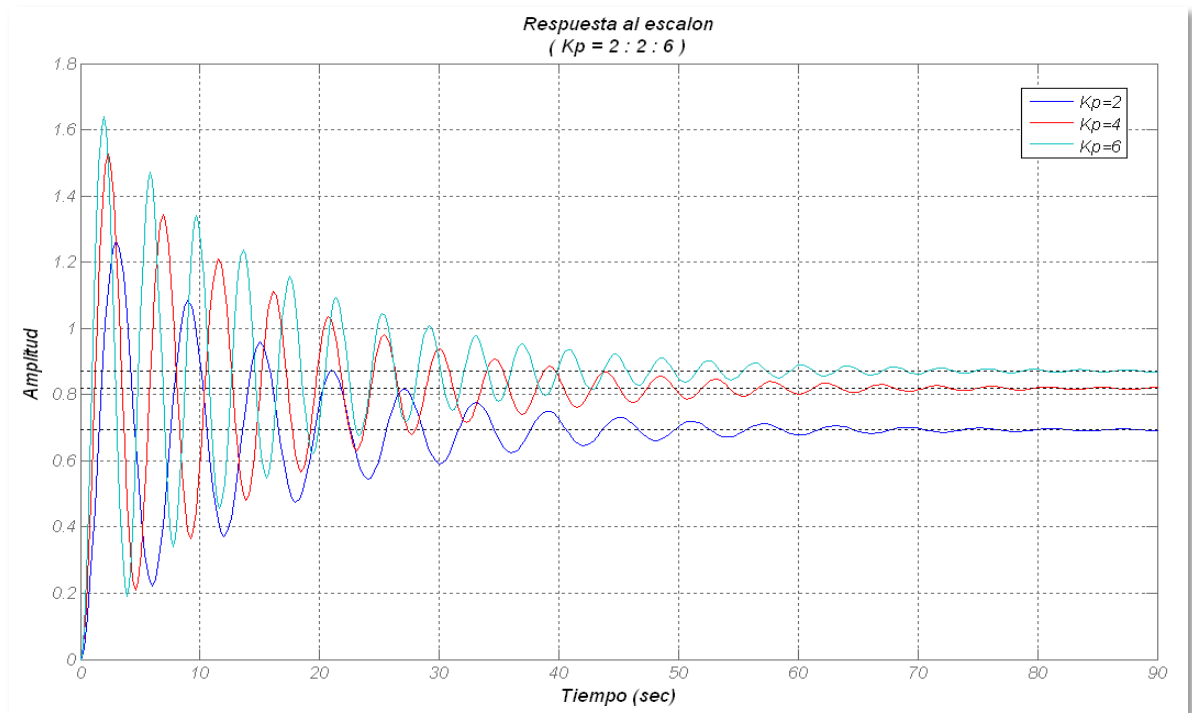


FIGURA 16. Simulación del sistema de control en lazo cerrado con control proporcional

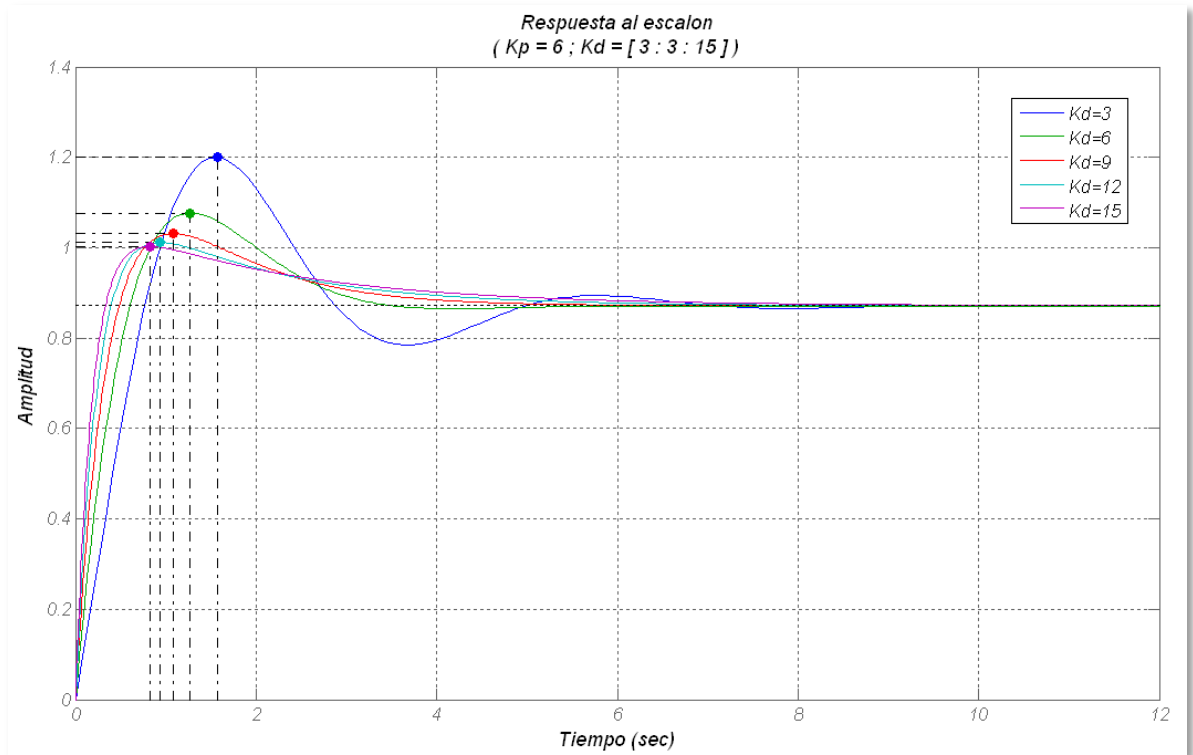
Al aplicar el control proporcional vemos que el error en estado estacionario disminuye (sin eliminarlo) conforme la ganancia aumenta, pero la respuesta del sistema se vuelve más oscilatoria y aumenta el sobrepico. El tiempo de estabilización permanece cercano e invariante en 60 seg. En nuestro sistema será necesario que la acción proporcional este acompañado de una acción derivativa que mejore la estabilidad del lazo cerrado

### **Control PD (Proporcional–Derivativo)**

La acción de control de un controlador proporcional-derivativa (PD) se define mediante la siguiente función de transferencia

$$\frac{Y(s)}{E(s)} = Kp(1 + T_d s) \quad ; \quad T_d = \frac{Kd}{Kp}$$

Donde  $T_d$  es una constantes ajustable denominada tiempo derivativo, y  $Kd$  es la ganancia derivativa. Las propiedades de la acción derivativa en nuestro controlador PD se muestran en la figura 17.



**FIGURA 17.** Análisis de la acción derivativa en un controlador PD

Como muestra la figura la acción derivativa mejora la estabilidad en lazo cerrado de nuestro sistema, inicialmente el amortiguamiento se incrementa con el incremento del tiempo derivativo es decir al aumentar la ganancia derivativa se reduce el sobrepico y el tiempo de estabilización que se está acercando a los 10 segundos exigidos por el sistema, aunque el error en estado estacionario disminuyó no se ha eliminado por completo aunque es un poco menor del 20%.

La acción de control derivativa tiene un carácter de previsión. Pero tiene la desventaja de que amplifica las señales de ruido y además puede provocar un efecto de saturación en el actuador.

Para eliminar el error en estado estacionario es necesario la intervención de una acción integral que me lleve el valor final de estabilización del sistema a 1.

## Control PID (Proporcional-Integral-Derivativo)

Como nuestro controlador PD seguía presentando error en estado estacionario introdujimos una acción integral para obtener un controlador PID la ecuación de este controlador se obtiene mediante la siguiente función de transferencia.

$$\frac{Y(s)}{E(s)} = Kp \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad ; \quad T_i = \frac{Kp}{Ki}$$

Donde  $T_i$  es una constante ajustable denominada tiempo integral, y  $Ki$  es la ganancia integral. Las propiedades de la acción integral en nuestro controlador PID se aclaran en la figura 18.

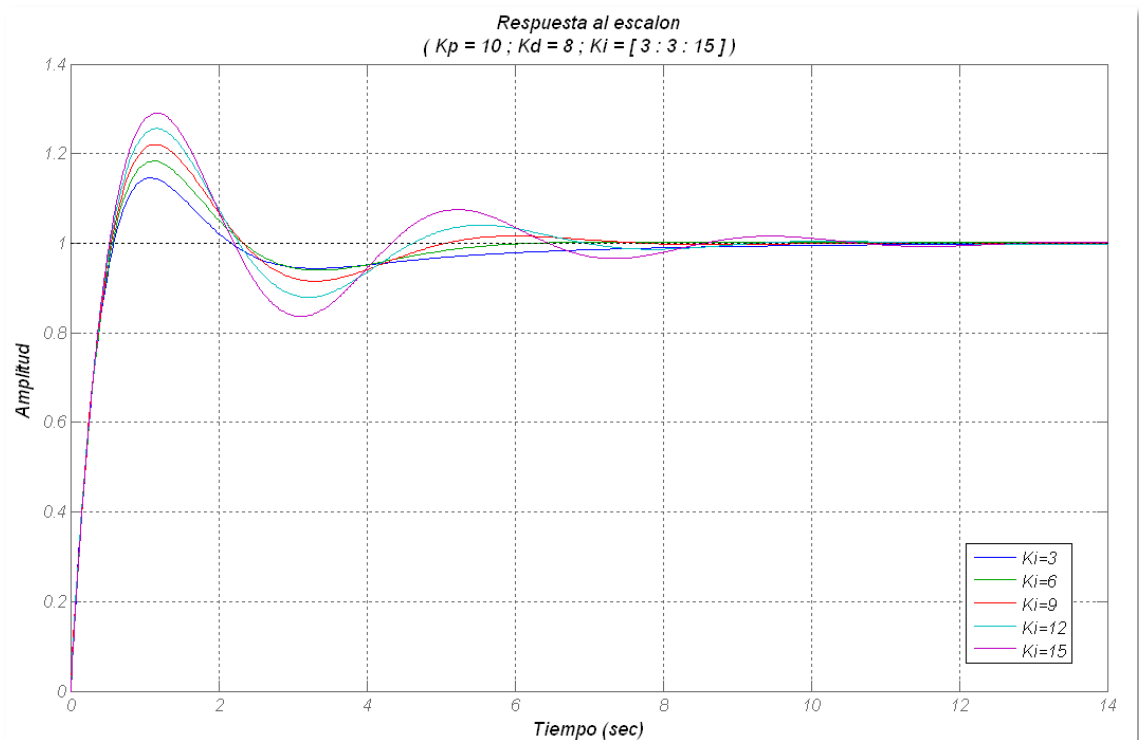


FIGURA 18. Análisis de la acción integral en un controlador PID

En la grafica vemos que con la intervención la acción integral se elimina el error en estado estacionario ya que si se presenta un pequeño error positivo siempre producirá un incremento en la señal de control, y un error negativo siempre dará una señal decreciente sin importar cuán pequeño sea el error. Para valores pequeños de  $K_i$  la respuesta se desliza lentamente hacia la referencia, cuando estos valores incrementan el sistema se hace mas oscilatorio.

El controlador PID combina las ventajas de cada una de las tres acciones de control individuales, con el ajuste adecuado de las variables de control cumplimos con los requerimientos del sistema y diseñamos un controlador PID en el que describe el comportamiento mostrado en la figura 19, es importante aclarar que este método es experimental y se baso en la respuesta al escalón que representa el sistema al combinar las diferentes acciones de control.

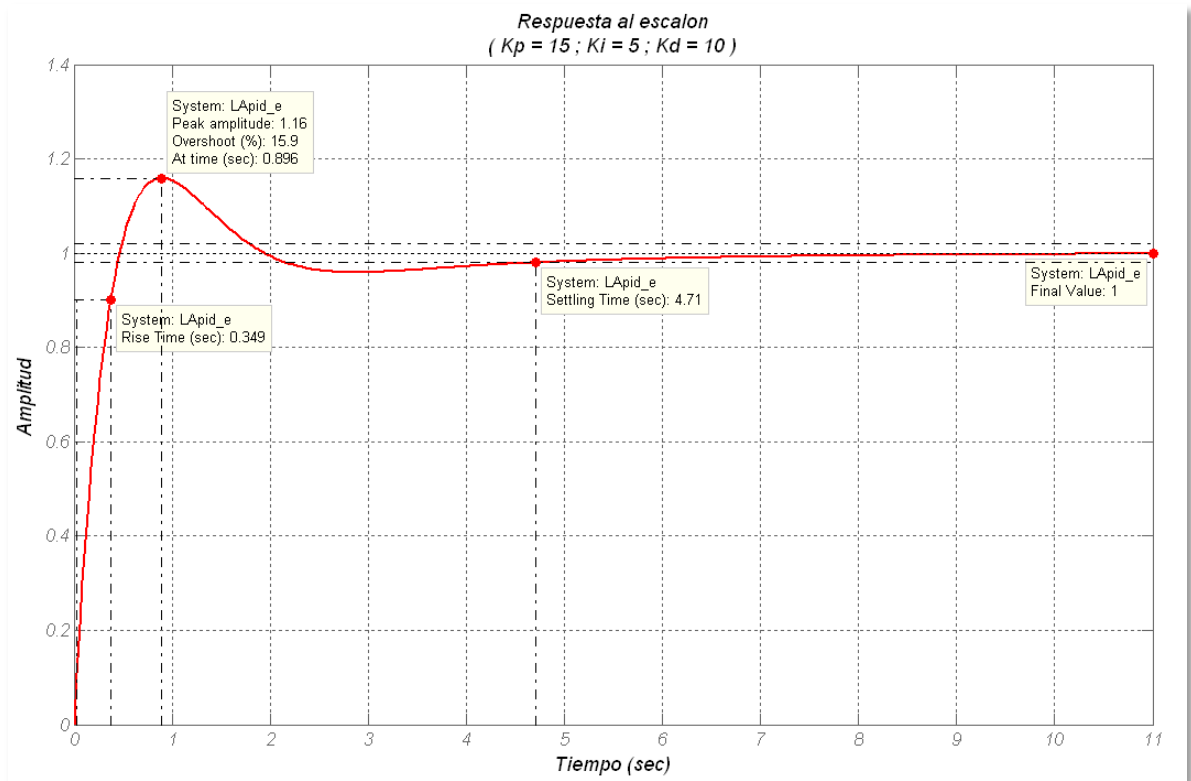


FIGURA 19. Respuesta al escalón para el ángulo de elevación compensado

Con la aplicación de este controlador el sistema evidencia un movimiento amortiguado, sin oscilaciones, con un tiempo de elevación de 0.349 (seg), un sobrepico del 15.9% un tiempo de estabilización de 4.71 (seg) y no presenta error en estado estacionario.

La función de transferencia que describe el controlador  $C(s)$  es entonces:

$$C_e(s) = \frac{Y_e(s)}{E_e(s)} = \frac{10s^2 + 15s + 5}{s}$$

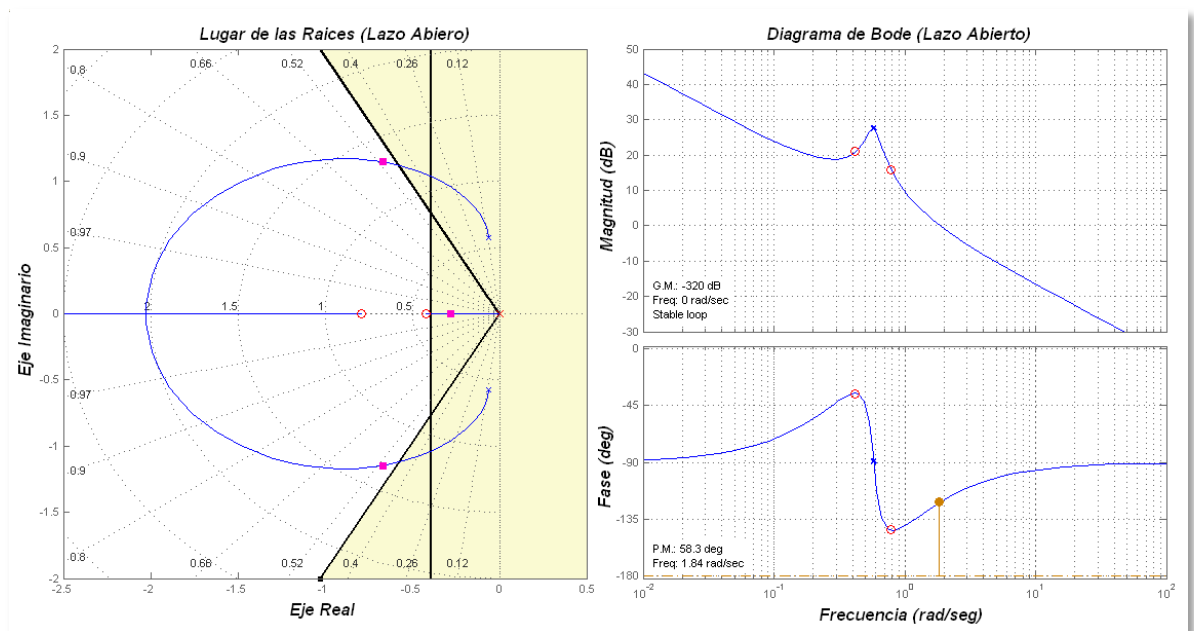
Esta función tiene dos ceros (-1 y -0.5), un polo integrador en cero (0) y una ganancia de 100, este es un control clásico PID que como hemos dicho no se resulto del uso de un método analítico sino que por el contrario fue deducido en base a la observación de la respuesta dinámica del sistema.

### **Ajuste del controlador PID implementando la SISOTOOL de MATLAB**

Sisotool es una herramienta gráfica ofrecida por MatLab que permite el análisis de sistemas lineales, y a través de la cual pudimos diseñar nuestro controlador PID utilizando el algoritmo de sintonización de Ziegler-Nichols y vigilando el cumplimiento de las exigencias de diseño.

Esta herramienta nos permite obtener y analizar el lugar de las raíces del sistema. Después de introducir la función de transferencia que representa la planta dinámica, y de seleccionar el controlador PID como método de diseño y el método de Ziegler y Nichols como algoritmo de sintonización, obtuvimos el diagrama mostrado en la figura 20 que muestra la grafica del lugar de las raíces en lazo abierto y el diagrama de bode para un posterior análisis frecuencial.

El objetivo de este diseño es lograr que la línea de acción de los polos dominantes este muy cerca al lugar donde se cruzan el sobrepico y el tiempo de establecimiento para garantizar que el compensador cumpla con los requerimientos del sistema, es necesario que los polos dominantes se encuentren lo más cercano posible a la región de requerimientos (región amarilla clara, figura 20) sin llegar a cruzarla.



**FIGURA 20.** Lugar de las raíces y Diagrama de Bode, del compensador  $C_e(s)$  sintonizado a través del algoritmo de Ziegler-Nichols, determinados mediante la herramienta SISOTOOL de MATLAB.

La grafica muestra un compensador  $C_e(s)$  con una ganancia de 3.923, con 2 ceros ubicados en -0.79 y -0.42, y un polo integrador ubicado en el origen, además se puede observar en la ventana del lugar de las raíces el sobrepico ajustado al 20% y el tiempo de establecimiento en 10seg.

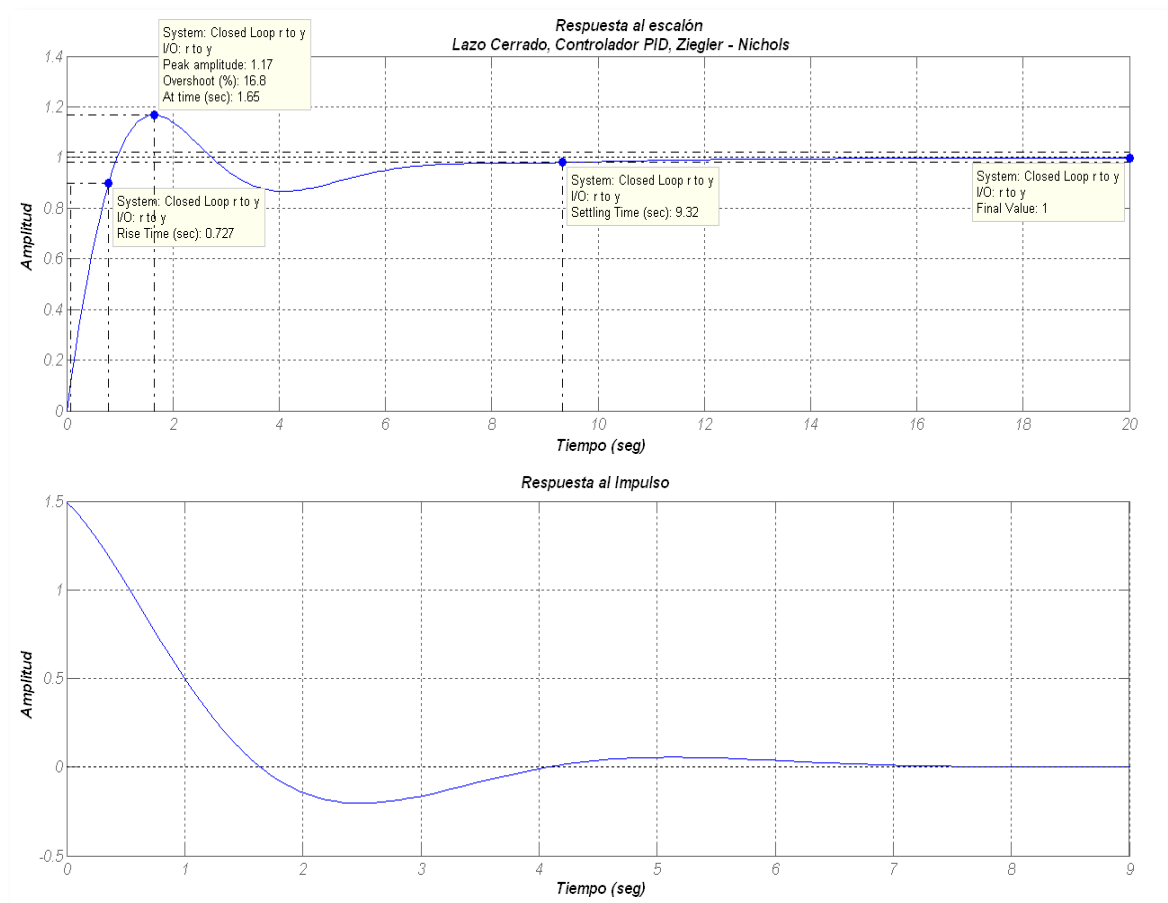
El controlador  $C_e(s)$  obtenido en términos de polos ceros y ganancia es:

$$C_e(s) = \frac{Y_e(s)}{E_e(s)} = \frac{3.9225(s + 0.7899)(s + 0.4202)}{s}$$

El controlador en términos de potencias

$$C_e(s) = \frac{Y_e(s)}{E_e(s)} = \frac{3.923s^2 + 4.747s + 1.302}{s}; K_p = 4.747 ; K_i = 1.302 ; K_d = 3.923$$

En la figura 21 vemos la respuesta escalón, y la repuesta al impulso del sistema en lazo cerrado.



**FIGURA 21.** Respuesta al escalón e impulso del sistema de lazo cerrado con los parámetros del compensador  $C_e(s)$ , sintonizado a través del algoritmo de Ziegler –Nichols.

Con la aplicación de este controlador el sistema presenta un movimiento amortiguado, con un sobrepico de 16.8%, un tiempo de estabilización de 9.32 (seg), un tiempo de subida de 0.727 (seg) y no presenta error en estado estacionario.

### **Ajuste del controlador PID con filtro en la acción derivativa**

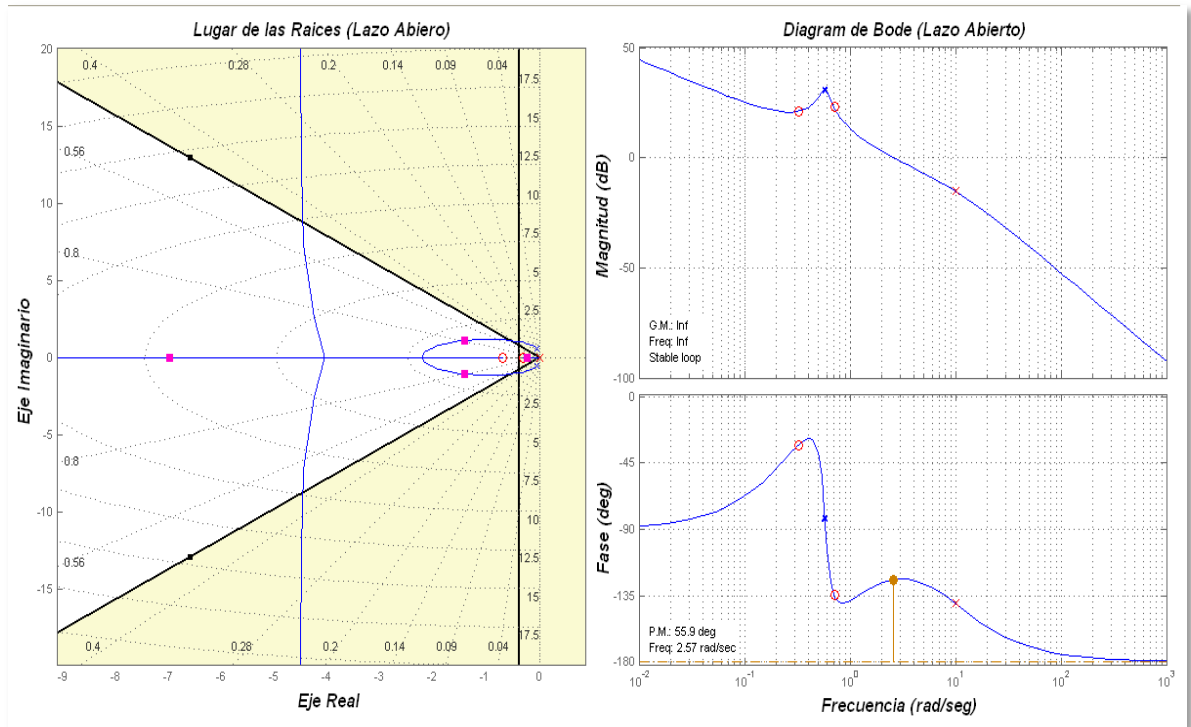
La función de transferencia que describe un controlador con filtro en la acción derivativa esta dada por:

$$C(s) = Kp \left( 1 + \frac{1}{T_i s} + \frac{T_d s}{T_f s + 1} \right) \quad ; \quad T_f = \frac{T_d}{N}$$

Con este tipo de controlador incluimos un polo con el fin de utilizar acciones de control grandes en respuesta a errores de control de alta frecuencia, tales como errores inducidos por cambios de setpoint (referencia) o mediciones de ruido.

En esta modificación consideramos que la derivada ideal es filtrada por un sistema de primer orden con una constante de tiempo igual a  $T_d/N$ . esta aproximación es muy buena para las bajas frecuencias, En la práctica, los valores de N están comprendidos entre 8 y 40. En el diseño de este lazo de control utilizamos un valor de  $N=10$ .

Con la ayuda de Sisotool en Matlab diseñamos un controlador con filtro derivativo que presenta 2 polos, uno ubicado en el origen y el otro en -10; 2 ceros ubicados en -0.711 y en -0.326 y tiene una ganancia de 63.5304 como se muestra en la figura 22.



**FIGURA 22.** Lugar de las raíces y Diagrama de Bode, del controlador PID con filtro derivativo, determinados mediante la herramienta SISOTOOL de MATLAB.

En este diagrama la línea de acción de los polos dominantes se cruza con la intersección del sobrepico de 20% y la línea de tiempo de estabilización en 10 segundos.

El controlador  $C(s)$  obtenido en términos de polos ceros y ganancia es:

$$C_e(s) = \frac{Y_e(s)}{E_e(s)} = \frac{63.5304(s + 0.711)(s + 0.326)}{s(s + 10)}$$

El controlador en términos de potencias se define como:

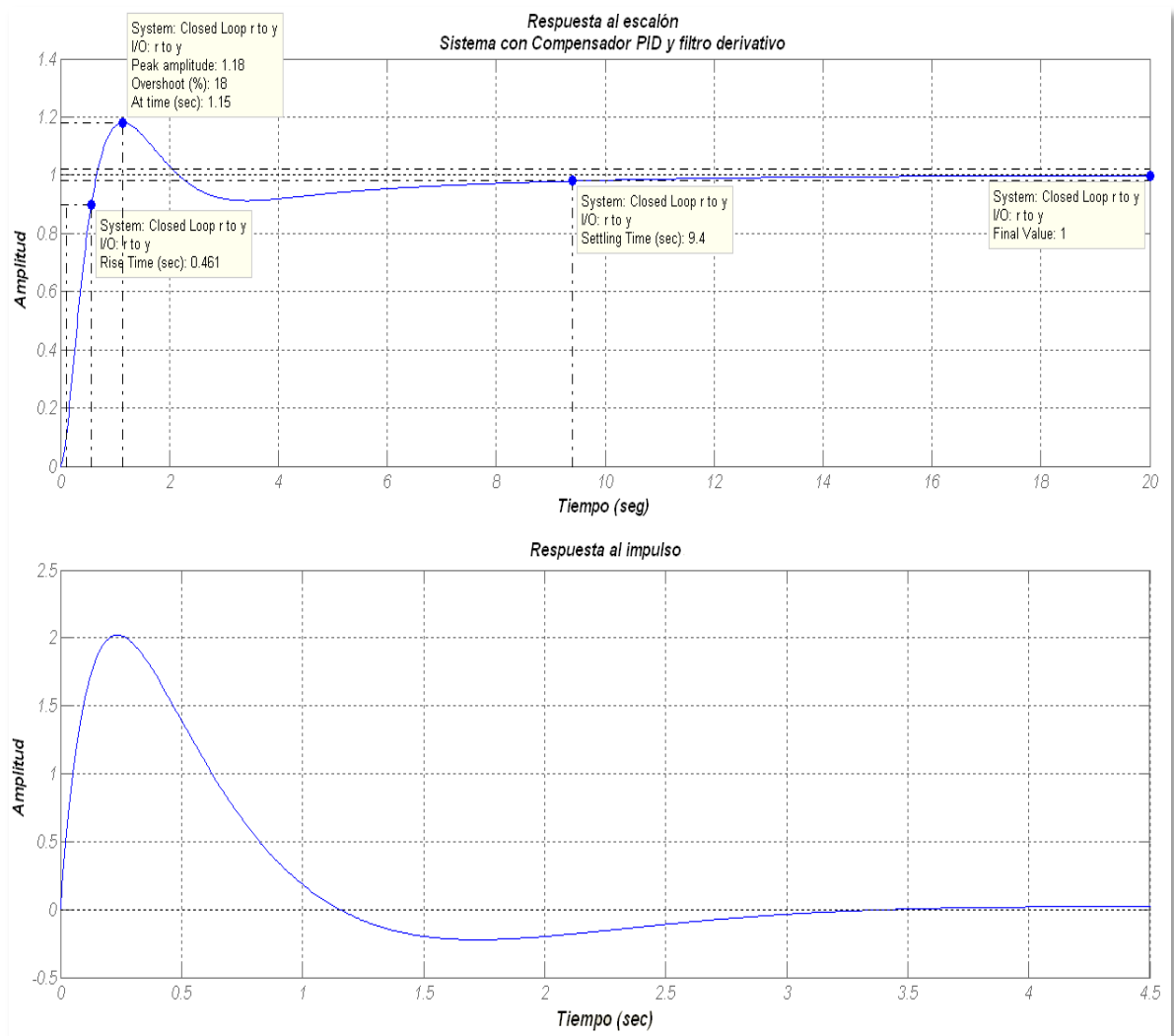
$$C_e(s) = \frac{Y_e(s)}{E_e(s)} = \frac{63.53s^2 + 65.86s + 14.71}{s^2 + 10s}$$

Parámetros del controlador:

$$K_p = 10.312 ; K_i = 2.656 ; K_d = 9.194$$

$$T_i = 3.882 ; T_d = 0.8916 ; T_f = 0.0892$$

Este controlador se ajusta estrechamente a los requerimientos de diseño y describe el comportamiento mostrado en la figura 23.



**FIGURA 23.** Respuesta al escalón y respuesta al impulso del controlador PID con filtro derivativo

## Elección del Controlador

En el desarrollo de este capítulo hemos propuesto tres diferentes métodos de control en base al controlador PID, para la compensación del ángulo de elevación del Helibot 3DOF. Luego de hacer un análisis de la respuesta transitoria del sistema compensado para una entrada escalón unitario, hemos condensado en la tabla 2 las características de estas respuestas.

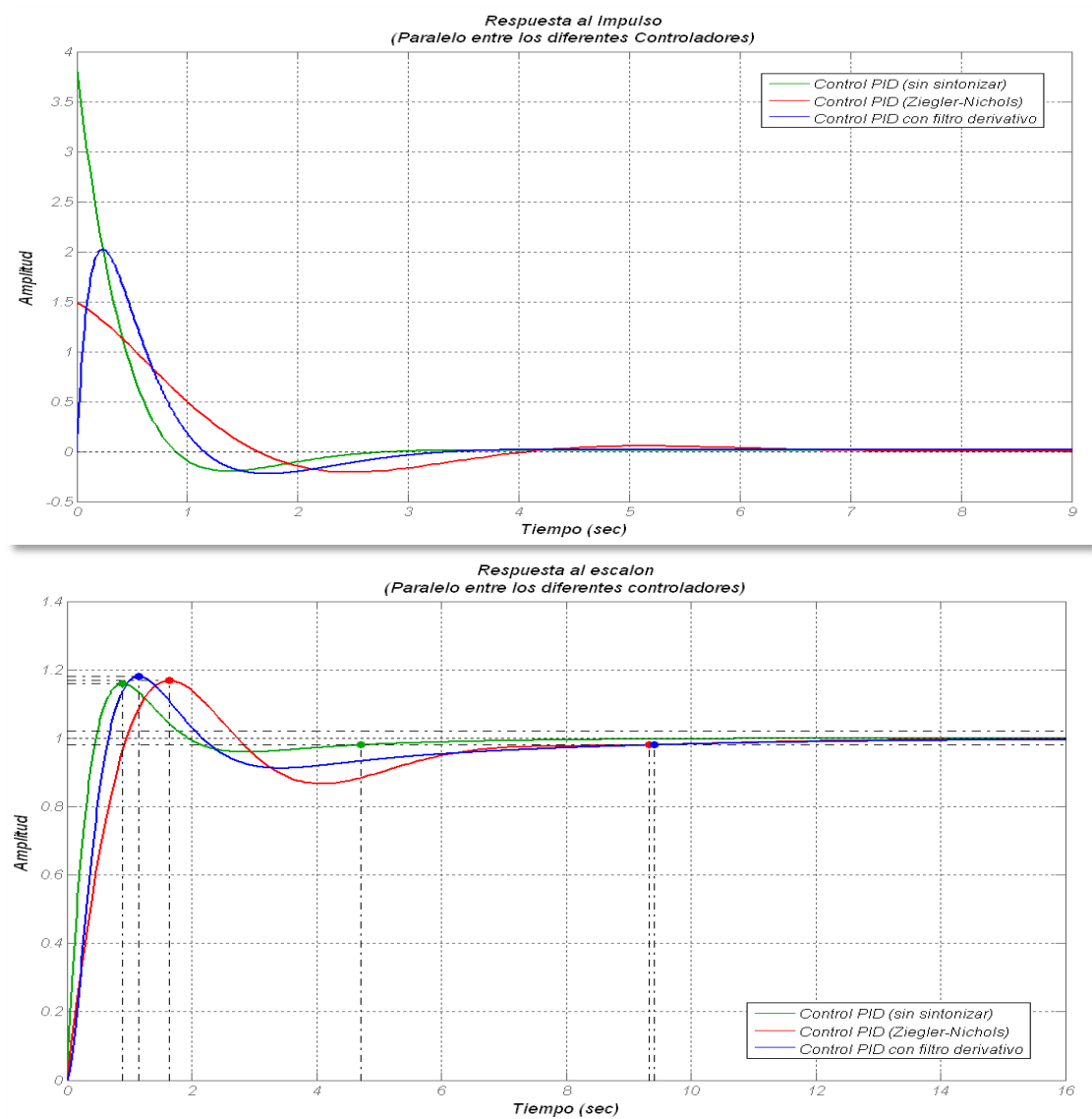
<b>Característica</b>	<b>Propuestas de Control</b>		
	<b>Control PID (sin sintonizar)</b>	<b>Control PID (Ziegler - Nichols)</b>	<b>Control PID con Filtro derivativo</b>
<b>Tiempo de Subida</b>	0.349(seg)	0.727(seg)	0.461(seg)
<b>Tiempo de Pico</b>	0.896(seg)	1.65(seg)	1.15(seg)
<b>Amplitud de pico</b>	1.16	1.17	1.18
<b>Sobrepico</b>	15.9%	16.8%	18%
<b>Tiempo de establecimiento</b>	4.71(seg)	9.32(seg)	9.4(seg)
<b>Error en estado estacionario</b>	ninguno	ninguno	ninguno

**TABLA 4.** Síntesis de las características del sistema compensado

Con la aplicación de los diferentes métodos de control el sistema se ajusta a las exigencias antes señaladas y describen la respuesta al escalón unitario mostrado en la grafica 24.

Conceptualmente, las ideas fundamentales para nuestro diseño se pueden resumir en cuatro puntos objetivo:

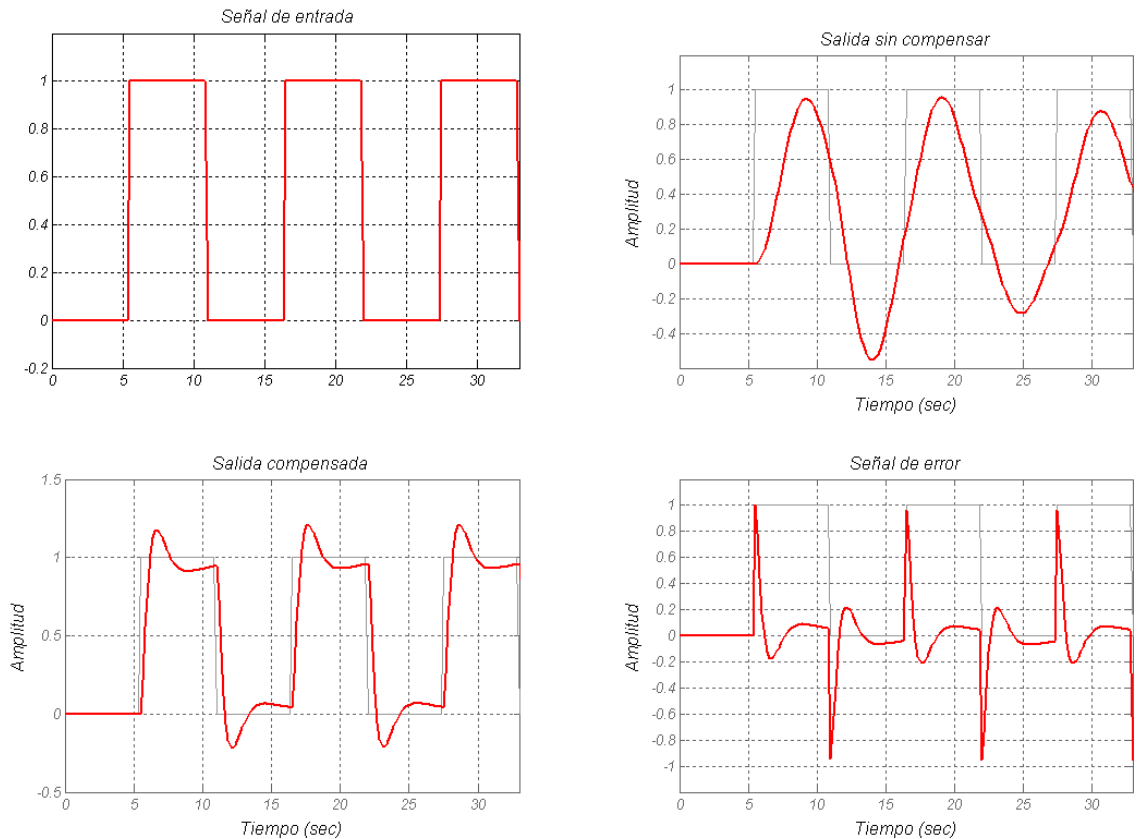
- Reducción o eliminación el error de seguimiento en estado estacionario (polos).
- Aumento del amortiguamiento y la estabilidad del sistema, y reducción del sobreimpulso (ceros).
- Rechazo del ruido a alta frecuencia en la medida de los sensores (filtros).
- Consideración de la saturación de los actuadores reales.



**FIGURA 24.** Análisis de diferentes métodos de control con base en el controlador PID ; a. Respuesta al escalón; b. Respuesta al impulso

Seleccionamos el controlador PID con filtro derivativo porque es el que mejor responde a los requerimientos de diseño, este método de control favorece el movimiento amortiguado, con pocas oscilaciones con un subimpulso inferior al 10%, además este tipo de control presenta mayores ventajas pedagógicas pues resume la mayor parte de la teoría de controladores PID.

Para simular las diferentes respuestas del sistema al utilizar el controlador PID con filtro derivativo como método de compensación, creamos un tren de pulsos que va a ser la entrada del sistema. Generamos una onda cuadrada, que va a ser la señal de entrada, con un periodo de 11 y el resultado se muestra en la figura 24.



**FIGURA 25. Tren de señales, para el monitoreo del ángulo de Elevación**

Como vemos en las graficas anteriores, la salida compensada es muy similar a la señal de entrada, por lo que el compensador calculado cumple satisfactoriamente los parámetros asignados. El error que se presenta es inducido para garantizar un retardo en el tiempo de estabilización ( $T_s < 10$  (seg)) necesario para evitar el sobre esfuerzo de los motores y provocar un movimiento amortiguado.

#### **4.4 DISEÑO DEL CONTROLADOR DEL ANGULO DE ALABEO**

##### **4.4.1 Requerimientos de diseño**

Queremos diseñar un controlador realimentado de modo que la salida tenga un sobreimpulso menor que 10%, un subimpulso menor al 7%, tiempo de subida menor que 2 segundos, tiempo de establecimiento menor que 5 segundos, error de estado estacionario menor que 2%, además necesitamos que el sistema describa un movimiento amortiguado sin que presente oscilaciones en torno al punto de equilibrio.

- Sobreimpulso (Overshoot): Menor que el 10%
- Subimpulso (Undershoot): Menor que el 7%
- Tiempo de Subida: Menor que 2 segundos
- Tiempo de establecimiento: Menor que 5 segundos
- Error de estado estacionario: Menor que el 2%

##### **4.4.2 Función de transferencia del ángulo de alabeo**

En el capítulo anterior encontramos una ecuación de transferencia que relaciona el ángulo de alabeo con la tensión eléctrica a partir de algunos parámetros técnicos

y dimensionales del sistema, esta ecuación es de segundo orden y se describe a continuación:

$$\frac{\alpha(s)}{V(s)} = \frac{8.820}{s^2 + 0.2856s + 4.41}$$

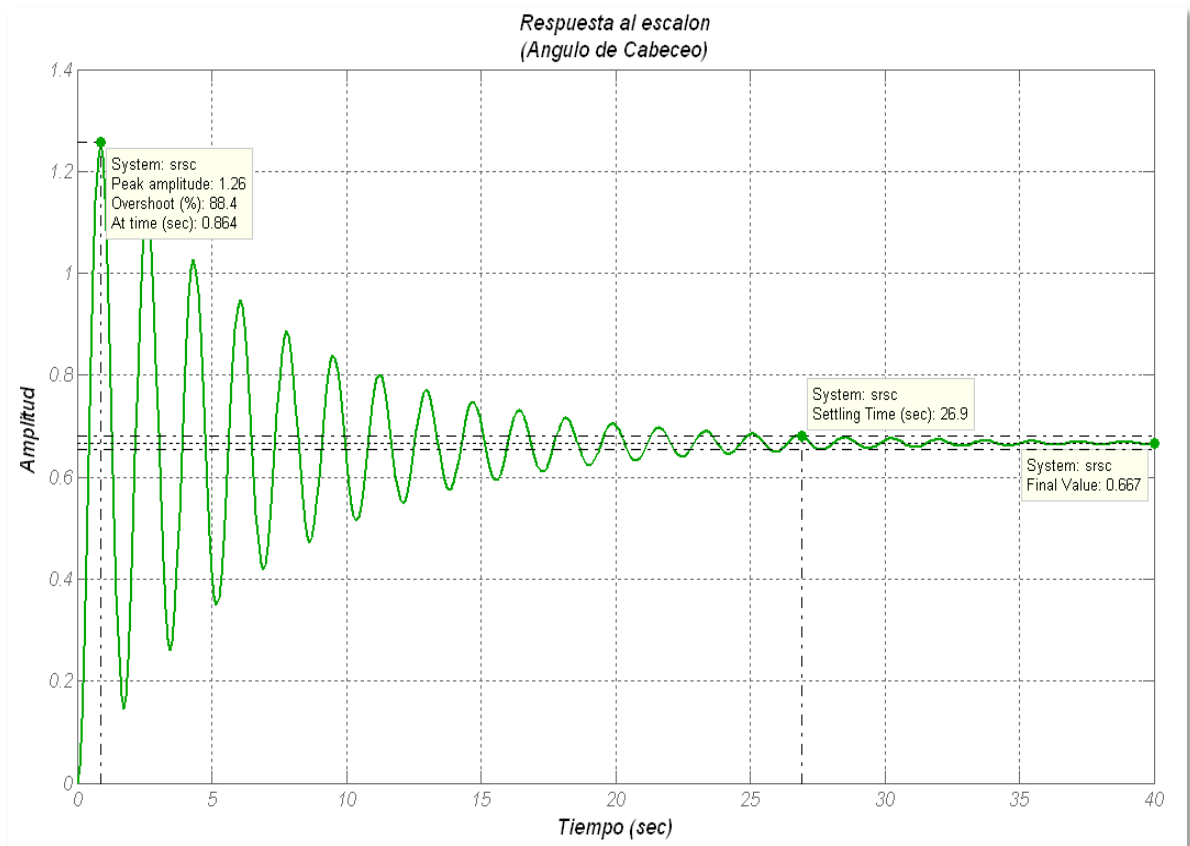
Determinada a partir de la ecuación generalizada:

$$\frac{\alpha(s)}{V(s)} = \frac{k\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

$$k = 0.2 \quad ; \quad \xi = 0.068 \quad ; \quad \omega_n = 2.1$$

Esta ecuación tiene un factor de amortiguamiento relativo ( $\xi$ ) menor al del ángulo de elevación ( $\xi=0.11$ ), ambos sistemas son subamortiguados ( $0 < \xi < 1$ ), pero la respuesta del ángulo de cabeceo será más oscilatoria, sin embargo se requerirá la aplicación de un controlador que se ajuste a exigencias más estrechas de amortiguamiento.

En la figura 26 se traza el comportamiento del sistema retroalimentado sin compensar ante una señal escalón unitario.



**FIGURA 26.** Respuesta al escalón del ángulo de alabeo sin regulación

El sistema se estabiliza en un valor final de 0.667(% ancho de pulso) necesitamos que es valor final este entre 0.98 (rad) y 1.02 (rad), además presenta un sobreimpulso de 88.4%, y un tiempo de estabilización de 26.9 seg muy superior a nuestros requerimientos de diseño, a continuación aplicaremos las acciones de control necesarias para satisfacer las exigencias del sistema.

Para el ángulo de alabeo no se mostraran las influencias de cada acción de control individual, pues ya fueron explicadas en el desarrollo del diseño del controlador en el ángulo de elevación, estas influencia son proporcionales ya que la función de transferencia para cada ángulo responden a la misma ecuación generalizada pero con diferentes parámetros.

### 4.4.3 Análisis de los diferentes métodos de control PID

#### Control PID (sin sintonizar)

Después de probar con varias combinaciones y valores de parámetros PID, diseñamos un controlador que responde satisfactoriamente a los requerimientos de diseño. Exhibe un sobreimpulso de 8.79% y un subimpulso de 11.7%, el tiempo de estabilización es de 4.48 segundos y no se presenta erro de estado estacionario. La función de Transferencia que describe este controlador es:

$$Cc(s) = \frac{Yc(s)}{Ec(s)} = \frac{0.9s^2 + 2.8s + 1.5}{s}$$

Este Controlador traza la respuesta al escalón que se muestra en la figura 26.

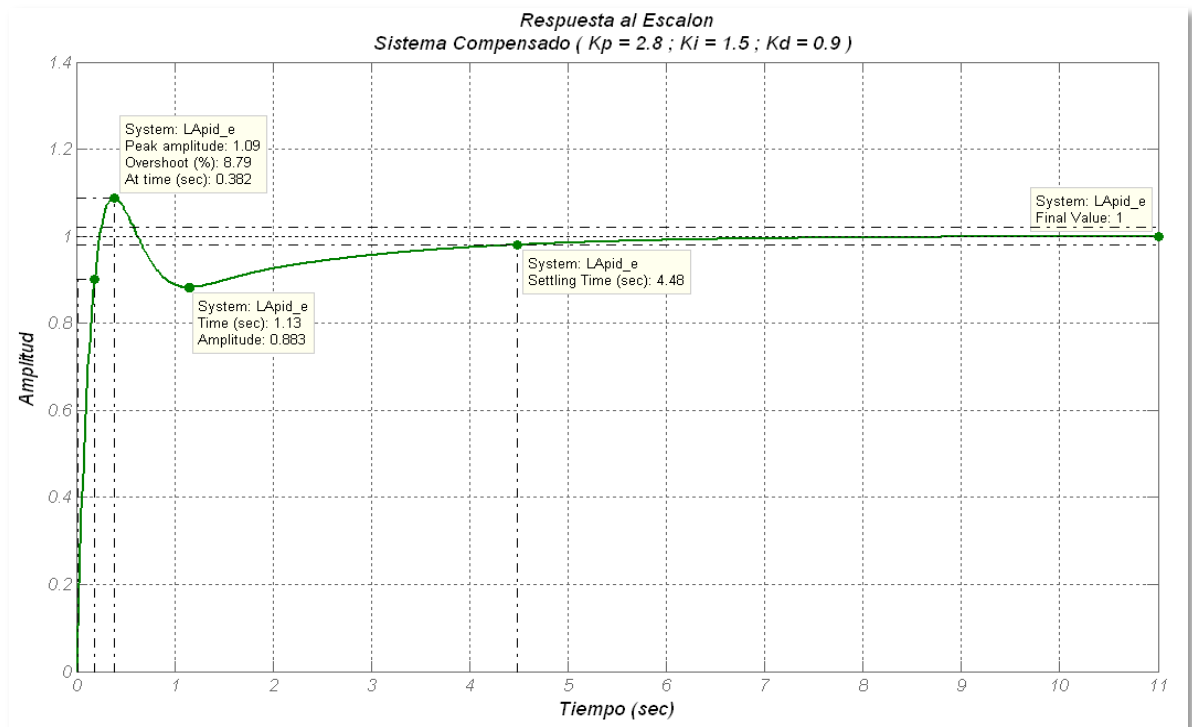


FIGURA 27. Respuesta al escalón para el ángulo de alabeo compensado

Esta función tiene dos ceros (-2.4234 y -0.6878), un polo integrador ubicado en el origen, y una ganancia de 0.9. Este es un control clásico PID que como hemos dicho no resultado del uso de un método analítico, sino que por el contrario fue deducido en base a la observación de la respuesta dinámica del sistema.

### **Ajuste del controlador PID implementando la SISOTOOL de MATLAB**

Después de introducir la función de transferencia de la planta dinámica, y de seleccionar el controlador PID como método de diseño y el método de Ziegler y Nichols como algoritmo de sintonización, obtuvimos el diagrama mostrado en la figura 28 que muestra la grafica del lugar de las raíces en lazo abierto y el diagrama de bode para un posterior análisis frecuencial.

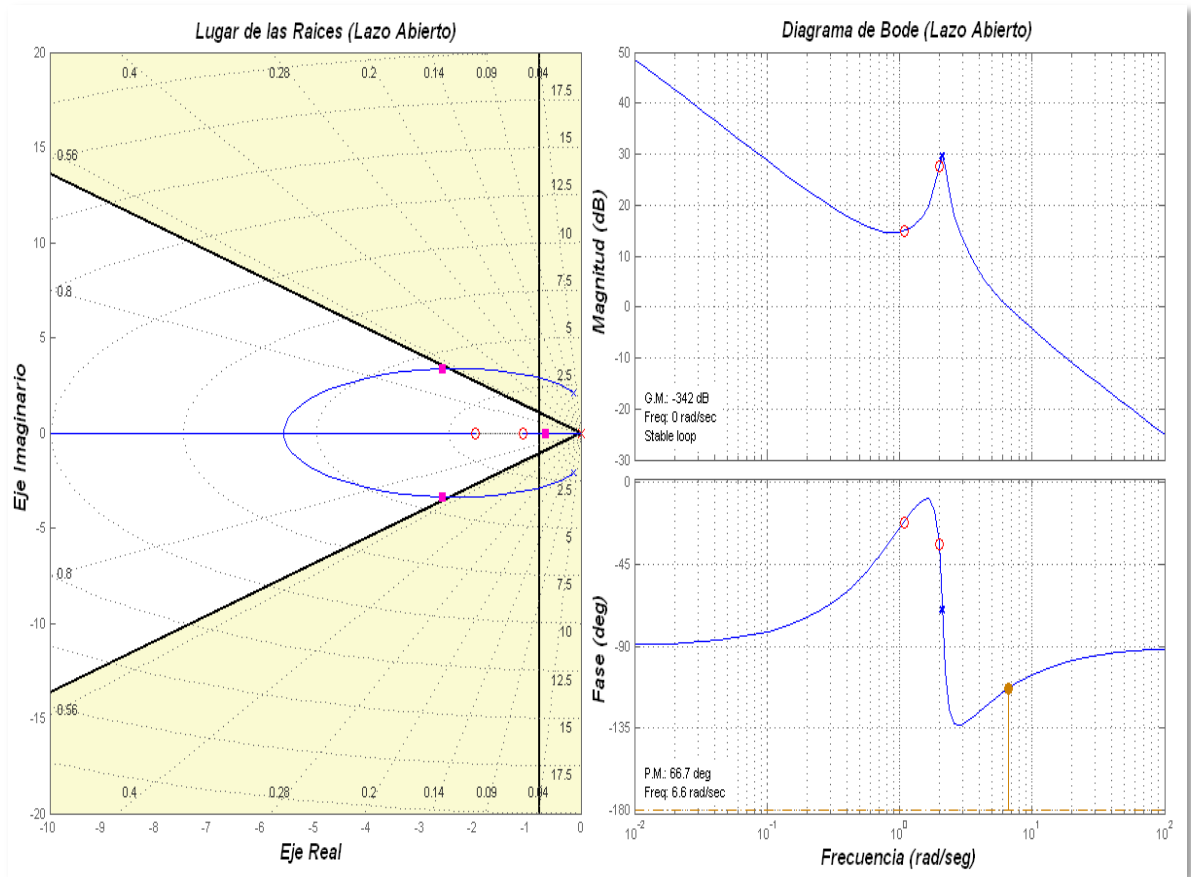
A partir de este análisis resulto el controlador con la siguiente función de transferencia:

$$C_c(s) = \frac{Y_c(s)}{E_c(s)} = \frac{0.636s^2 + 1.952s + 1.367}{s}$$

$$K_p = 1.952 \quad ; \quad K_i = 1.367 \quad ; \quad K_d = 0.636$$

El controlador  $C(s)$  obtenido en términos de polos ceros y ganancia es:

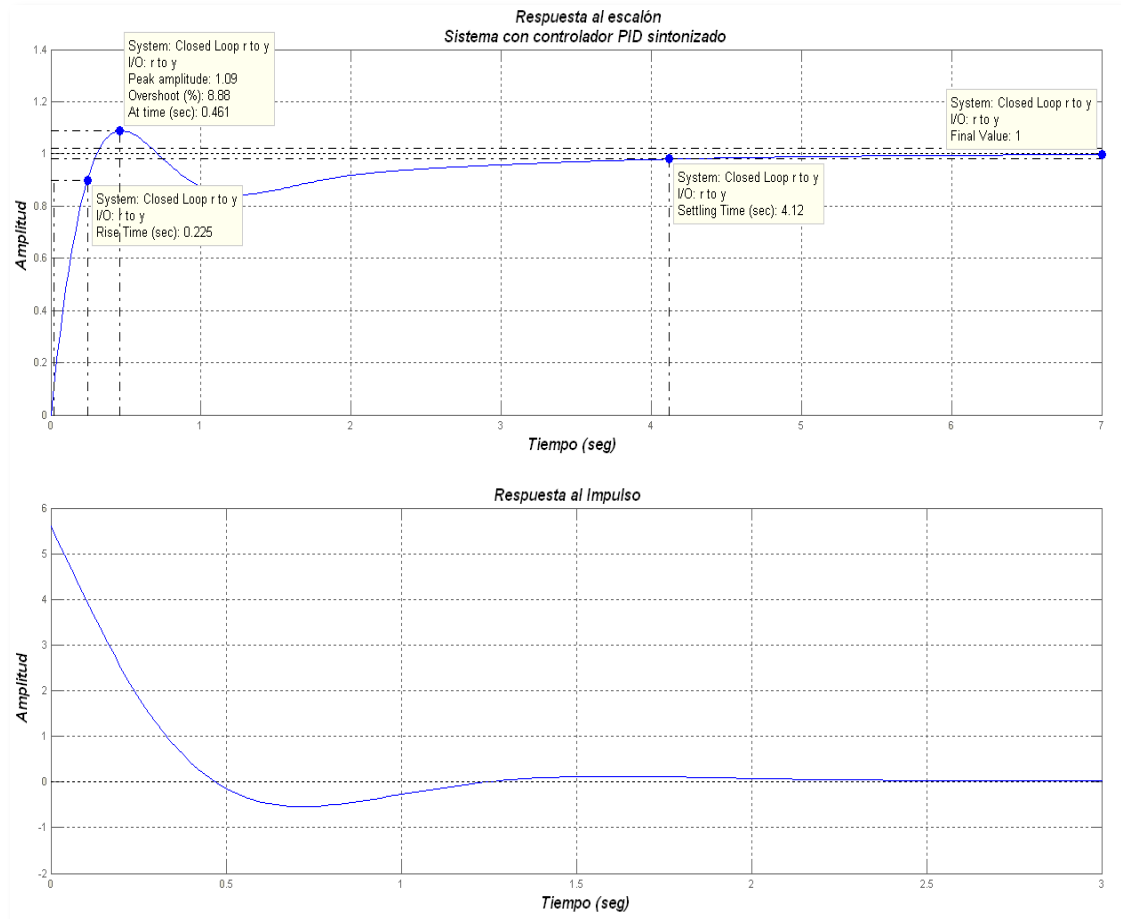
$$C_c(s) = \frac{Y_c(s)}{E_c(s)} = \frac{0.636(s + 1.985)(s + 1.082)}{s}$$



**FIGURA 28.** Lugar de las raíces y Diagrama de Bode, del compensador  $C_c(s)$  sintonizado a través del algoritmo de Ziegler-Nichols, determinados mediante la herramienta SISOTOOL de MATLAB.

La grafica muestra un compensador  $C_c(s)$  con una ganancia de 0.6363, con 2 ceros ubicados en -1.9853 y -1.0819 un polo integrador ubicado en el origen, además se puede observar en la ventana del lugar de las raíces, el sobreimpulso ajustado al 10% y el tiempo de establecimiento en 5seg.

En la figura 29 se puede ver la respuesta al escalón unitario e impulso que traza el sistema compensado, con el controlador propuesto.



**FIGURA 29.** Respuesta al escalón del sistema de lazo cerrado con los parámetros del compensador  $C_c(s)$ , sintonizado a través del algoritmo de Ziegler –Nichols.

Con la aplicación de este controlador el sistema presenta un movimiento amortiguado, con un sobreimpulso de 8.88%, un subimpulso de 15.7% un tiempo de estabilización de 4.12 (seg), y no presenta error en estado estacionario.

### Ajuste del controlador PID con filtro en la acción derivativa

Con la ayuda de Sisotool en Matlab diseñamos un controlador con filtro derivativo que presenta 2 polos, uno ubicado en el origen y el otro en  $-35.5117$ ; 2 ceros

ubicados en -1.4565 y en -0.8010 y tiene una ganancia de 41.7746 como se muestra en la figura 29. En este controlador se utilizó una N (filtro derivativo) = 35.5. El controlador  $C(s)$  obtenido en términos de polos, ceros y ganancia es:

$$C_c(s) = \frac{Y_e(s)}{E_e(s)} = \frac{41.7746(s + 1.456)(s + 0.801)}{s(s + 35.51)}$$

El controlador en términos de potencias se define como:

$$C_c(s) = \frac{Y_e(s)}{E_e(s)} = \frac{41.77s^2 + 94.31s + 48.74}{s^2 + 35.51s}$$

Parámetros del controlador:

$$K_p = 2.617 ; K_i = 1.3724 ; K_d = 1.1027$$

$$T_i = 1.9068 ; T_d = 0.4214 ; T_f = 0.0421$$

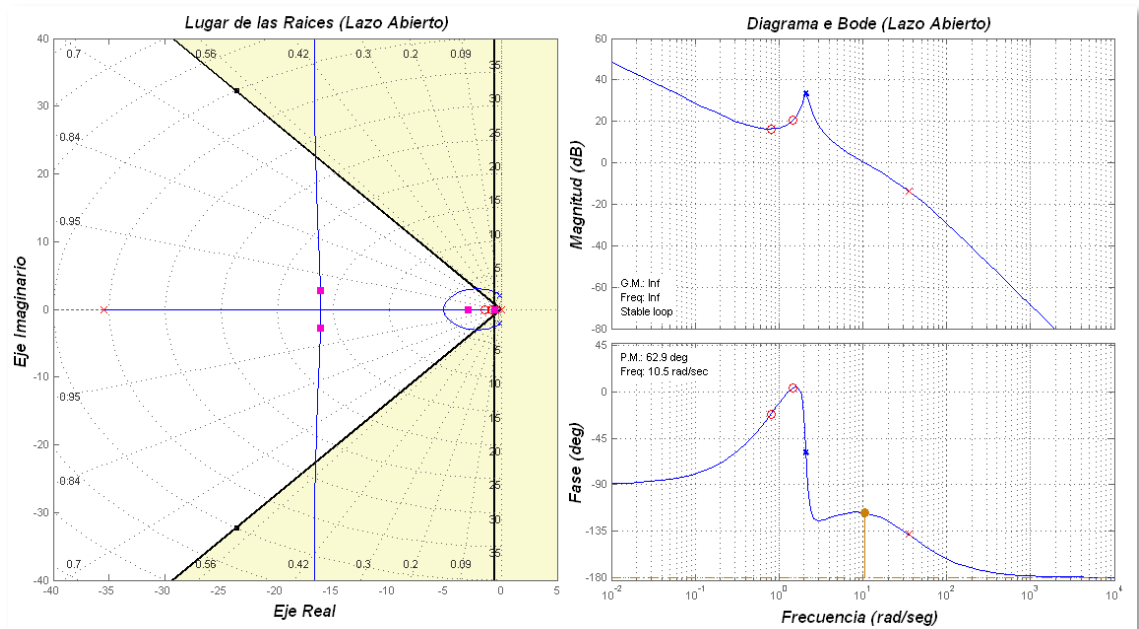


FIGURA 30. Lugar de las raíces y Diagrama de Bode, del controlador PID con filtro derivativo, determinados mediante la herramienta SISOTOOL de MATLAB.

Este controlador disminuye a los rangos deseados el valor del subimpulso y se ajusta cómodamente a los requerimientos de diseño como se muestra en la figura 31.

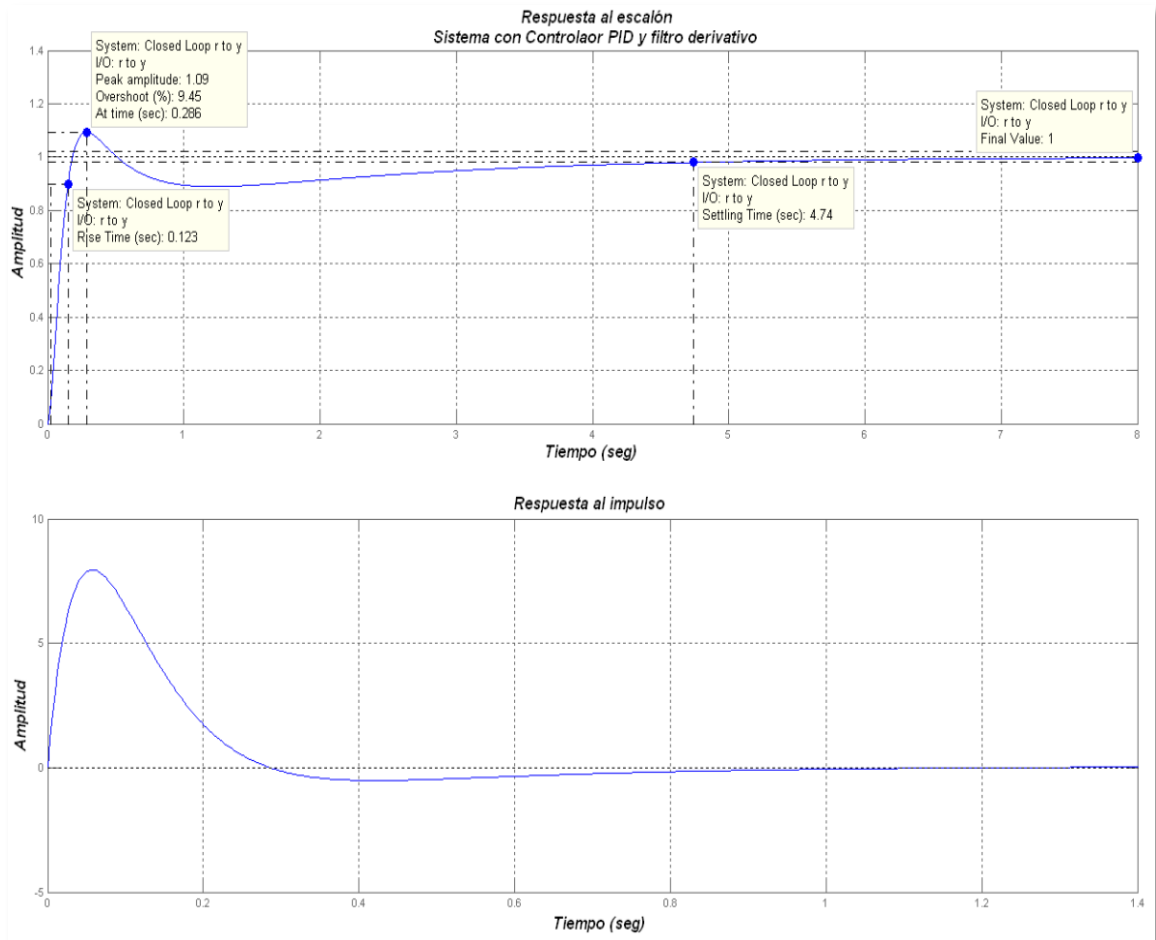


FIGURA 31. Respuesta al escalón y respuesta al impulso del controlador PID con filtro derivativo

## Elección del Controlador

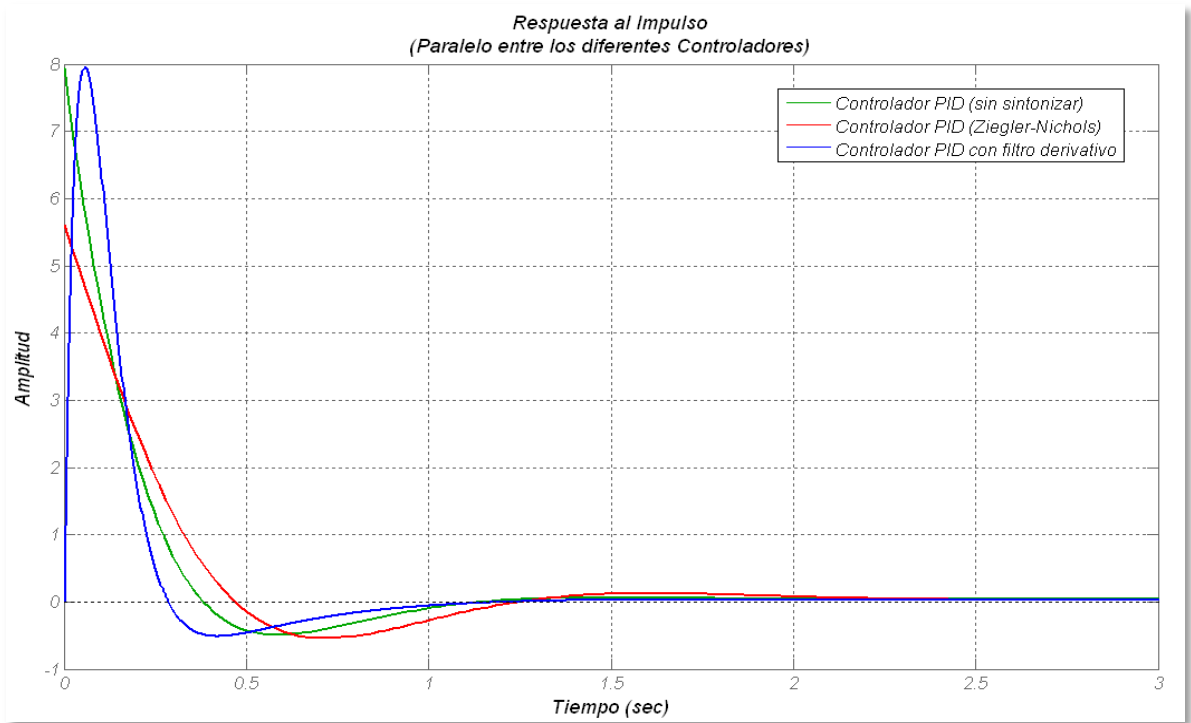
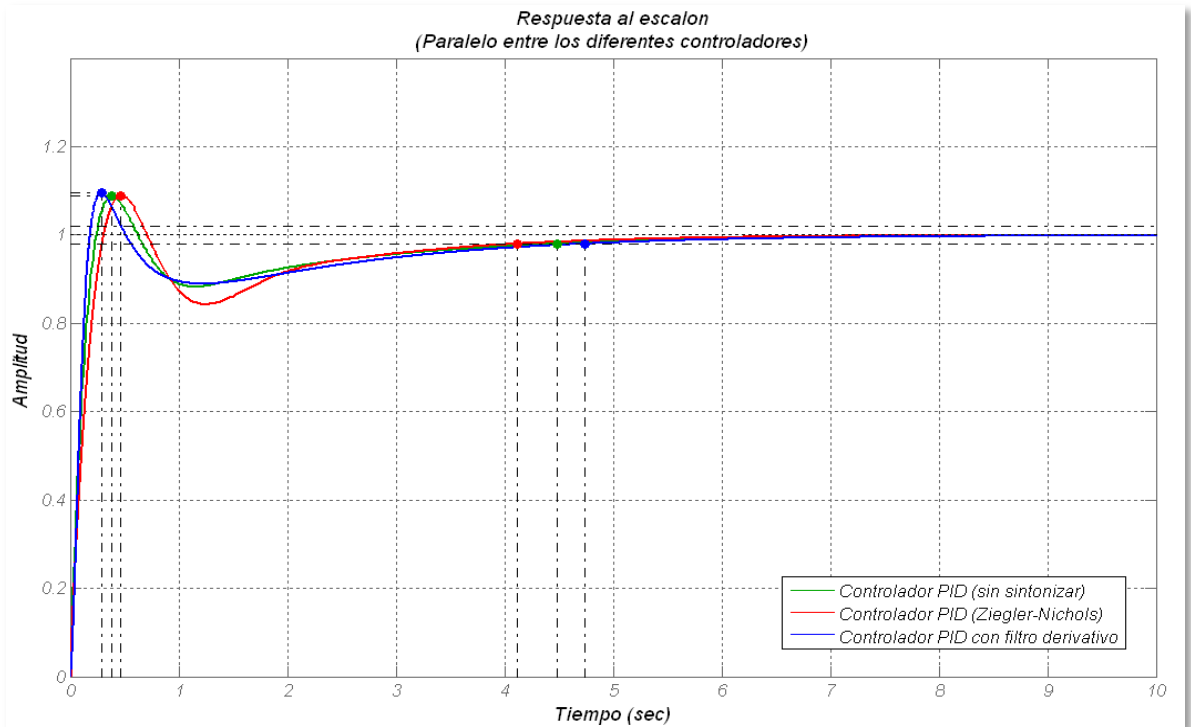
Al igual que en el ángulo de elevación, para la compensación del ángulo de alabeo se propusieron tres diferentes métodos de control en base al controlador PID. Luego de hacer un análisis de la respuesta transitoria del sistema compensado

para una entrada escalón unitario e impulso, resumimos en la tabla 3 las características de estas respuestas.

<b>Característica</b>	<b>Propuestas de Control</b>		
	<b>Angulo de Alabeo -HeliBot 3DOF-</b>		
	<b>Control PID (sin sintonizar)</b>	<b>Control PID (Ziegler - Nichols)</b>	<b>Control PID con Filtro derivativo</b>
<b>Tiempo de Subida</b>	0.173(seg)	0.225(seg)	0.123(seg)
<b>Tiempo de Pico</b>	0.382(seg)	0.461(seg)	0.286(seg)
<b>Amplitud de pico</b>	1.09	1.09	1.09
<b>Sobreimpulso</b>	8.79%	8.88%	9.45%
<b>Subimpulso</b>	11.7%	15.7%	10%
<b>Tiempo de establecimiento</b>	4.48(seg)	4.12(seg)	4.74(seg)
<b>Error en estado estacionario</b>	ninguno	ninguno	ninguno

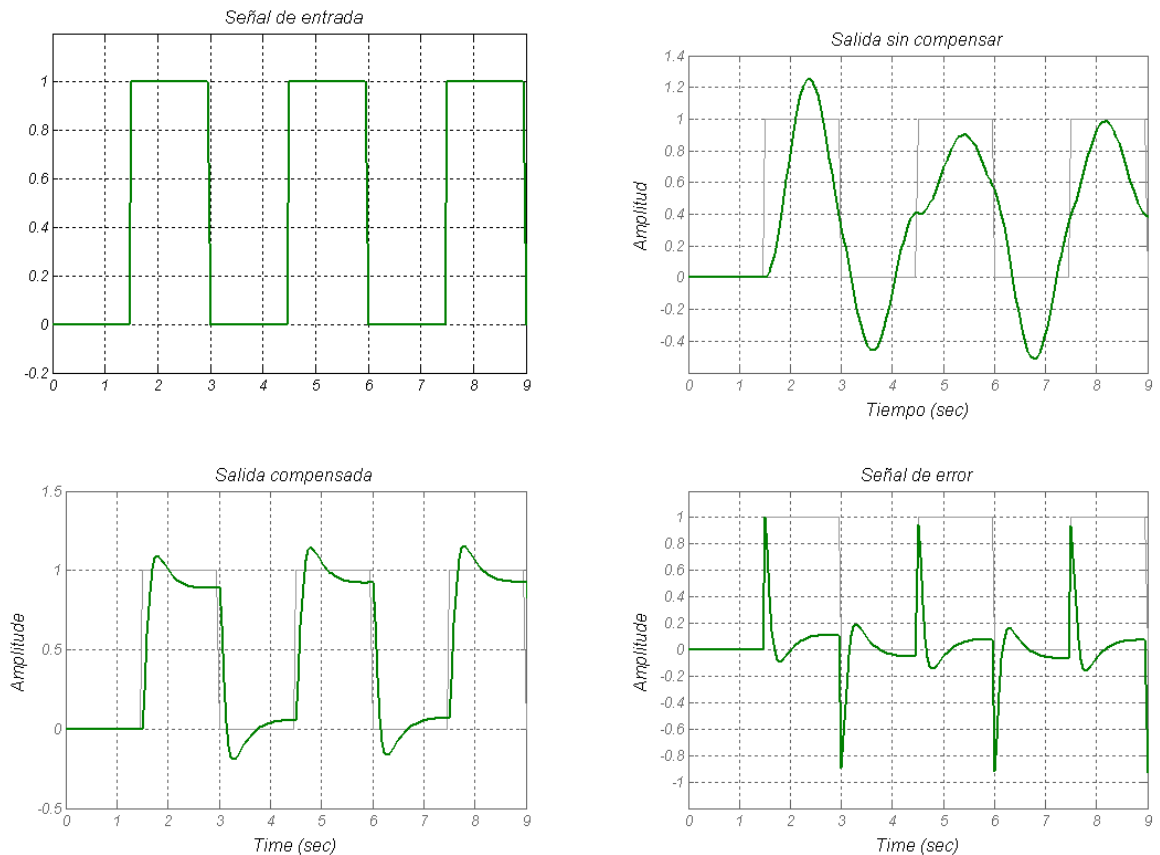
**TABLA 5.** Síntesis de las características del sistema compensado. Alabeo

Al igual que para el ángulo de elevación, en la compensación de este lazo de control seleccionamos el controlador PID con filtro derivativo porque es el que mejor responde a los requerimientos de diseño, además es el único en el cual el valor del subimpulso está ajustado a las necesidades del sistema aunque sea el que más tiempo tarda en su estabilización. Este método de control favorece el movimiento amortiguado, sin oscilaciones al mismo tiempo presenta mayores ventajas pedagógicas pues resume la mayor parte de la teoría de controladores PID.



**FIGURA 32.** Análisis de diferentes métodos de control con base en el controlador PID  
a. Respuesta al escalón; b. Respuesta al impulso

Para simular las diferentes respuestas del sistema al utilizar el controlador PID con filtro derivativo como método de compensación, creamos un tren de pulsos que va a ser la entrada del sistema. Generamos una onda cuadrada, que va a ser la señal de entrada, con un periodo de 4 y el resultado se muestra en la figura 33.



**FIGURA 33. Tren de señales, para el monitoreo del ángulo de Cabeceo**

Como vemos en las graficas anteriores, la salida compensada es muy similar a la señal de entrada, por lo que el compensador calculado cumple satisfactoriamente los parámetros asignados. El error que se presenta es inducido para garantizar un retardo en el tiempo de estabilización ( $T_s < 10$  (seg)) necesario para evitar el sobre esfuerzo de los motores y provocar un movimiento amortiguado.

## 5 DISEÑO ESTRUCTURAL DEL HELIBOT

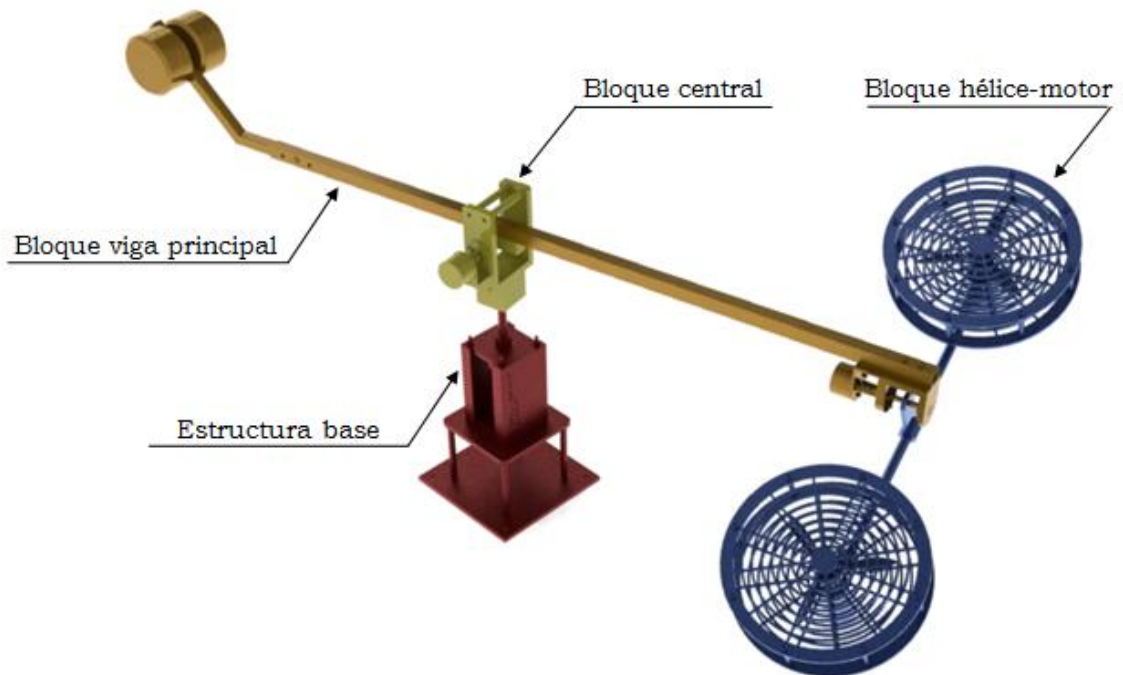
El diseño final de la planta dinámica de control resulta finalmente en un banco acorde con las exigencias de un laboratorio, como es: fácil manejo, modular, de fácil transporte y pedagógico. El aspecto general del banco se muestra en la figura 34.



**FIGURA 34.** Planta Dinámica HeliBot 3 DOF

La planta dinámica está compuesta de una serie de subconjuntos o bloques los cuales se muestran en la figura 35 y se resumen a continuación:

- Estructura base del HeliBot.
- Bloque central del HeliBot.
- Bloque viga principal
- Bloque hélice-motor.



**FIGURA 35.** Partes del banco

Es importante aclarar que aunque la estructura de apoyo (mesa) no hace parte integral de la planta dinámica, es un elemento importante en la concepción del Helibot 3DOF como banco de pruebas.

Las diferentes partes del banco fueron primero diseñadas minuciosamente en CAD con ayuda de Solidworks 2010, para su posterior construcción, los materiales principalmente utilizados fueron: Lamina HR en diferentes calibres, aluminio, acrílico, teflón y madera.

## 5.1 ESTRUCTURA BASE DEL HELIBOT

La estructura base del heliBot da soporte a la planta dinámica, proporcionando la rigidez necesaria para que se puedan generar los movimientos de rotación y de amortiguamiento, posee un muñón en una de sus placas para apoyar un resorte y en la placa superior posee un buje el cual aloja un rodamiento axial, como se muestra en la figura 36.

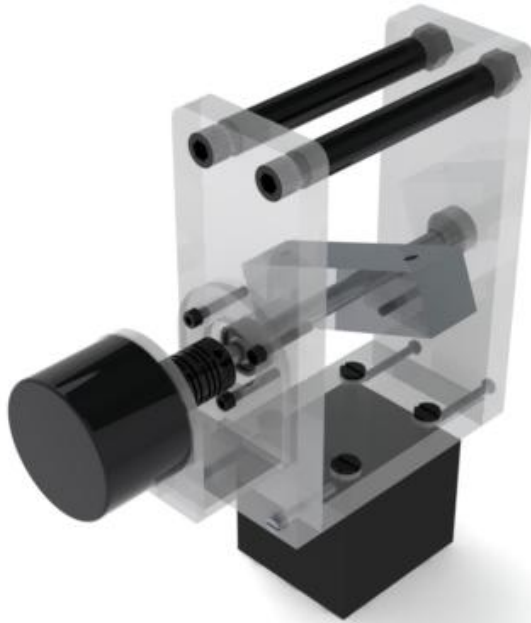


FIGURA 36. Estructura base del HeliBot

## 5.2 BLOQUE CENTRAL DEL HELIBOT

Este Bloque soporta la viga principal y el sensor que mide el ángulo de elevación y permite la rotación libre con respecto al eje vertical del sistema, en su parte inferior

se encuentra incrustado un rodamiento radial de bolas el cual se encuentra solidario con la estructura de apoyo por medio de un eje, este bloque se construyo en acrílico y aluminio aleado 1060.

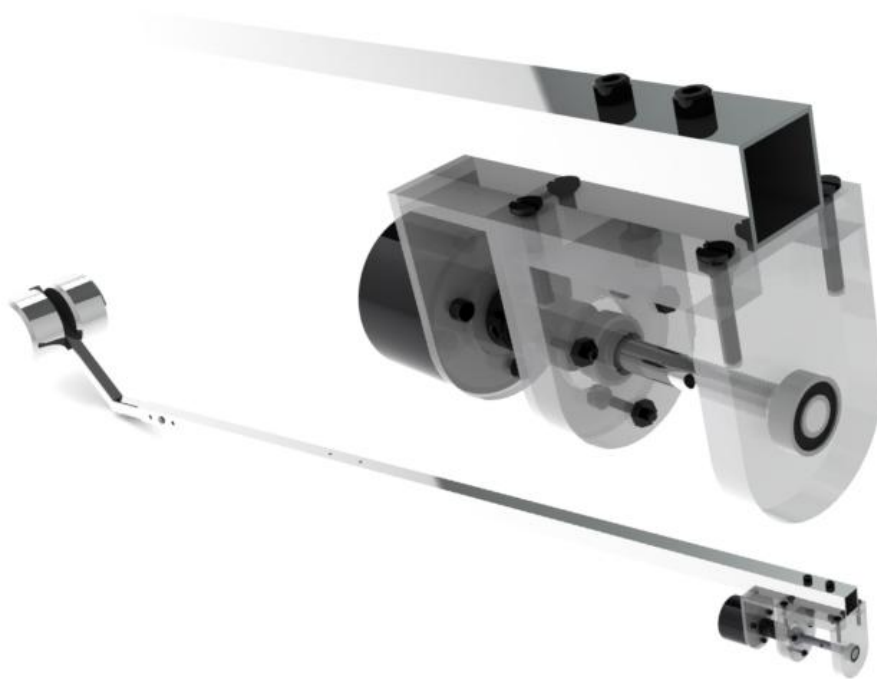


**FIGURA 37.** Bloque central del HeliBot 3 DOF

Este bloque posee además un eje el cual atraviesa el soporte de la viga principal y conecta en cada uno de sus extremos con rodamientos radiales de bolas, más adelante se hará la selección del rodamiento, el análisis de esfuerzos y deflexión de dicho eje a través del programa SimulationXpress Study de Solidworks 2010, por medio del cual validaremos el diseño.

### 5.3 BLOQUE VIGA PRINCIPAL

Este brazo posee en uno de sus extremos un contrapeso el cual fue diseñado para que cumpliera con la función de ayudar a estabilizar la planta dinámica, y en otro extremo viene conectado con la unión del Bloque Hélice-Motor.



**FIGURA 38.** Bloque viga principal

La viga principal es de sección tubular de aluminio de 1.9 X 1.9 mm, la razón de hacerlo tubular fue en primera instancia para hacer el sistema más liviano, además para que nos sirviera para pasar los cables que provienen del sensor y de los motores hacia la tarjeta.

El contrapeso y la unión viga principal-contrapeso, se construyo de Aluminio macizo, la distancia del contrapeso con respecto a la base del helibot se puede

aumentar por medio de unas guías que se pueden graduar con un prisionero de cabeza bristol para variar los diferentes pesos del contrapeso.

#### 5.4 BLOQUE HELICE - MOTOR

El conjunto paquete motriz está constituido principalmente por el bloque helice-motor. En este paquete se incluye un speed control *TowerPro w18A* el cual controla la velocidad de los motores brushless cuando se generan variaciones en el voltaje de entrada.



**FIGURA 39.** Bloque hélice-motor del HeliBot 3 DOF

Las canastas de protección fueron diseñadas en Solidworks seguidamente exportadas a CorelDraw X4, para ser cortadas en acrílico calibre 3mm en una maquina laser, se eligió acrílico ya que era un material que nos proporcionaba la resistencia requerida y además es económico comparado con el aluminio.

## 5.5 ESTRUCTURA DE APOYO

Finalmente, el banco cuenta con una estructura construida en tubería circular en HR, con un recubrimiento superficial electrostático y una tabla en madera con recubrimiento en fórmico para una mayor duración y fácil limpieza. Esta mesa permite el manejo y traslado de la planta dinámica como un equipo independiente. Esta estructura genera el equilibrio necesario para evitar funcionamientos erróneos del HeliBot 3 DOF, debido a las vibraciones.



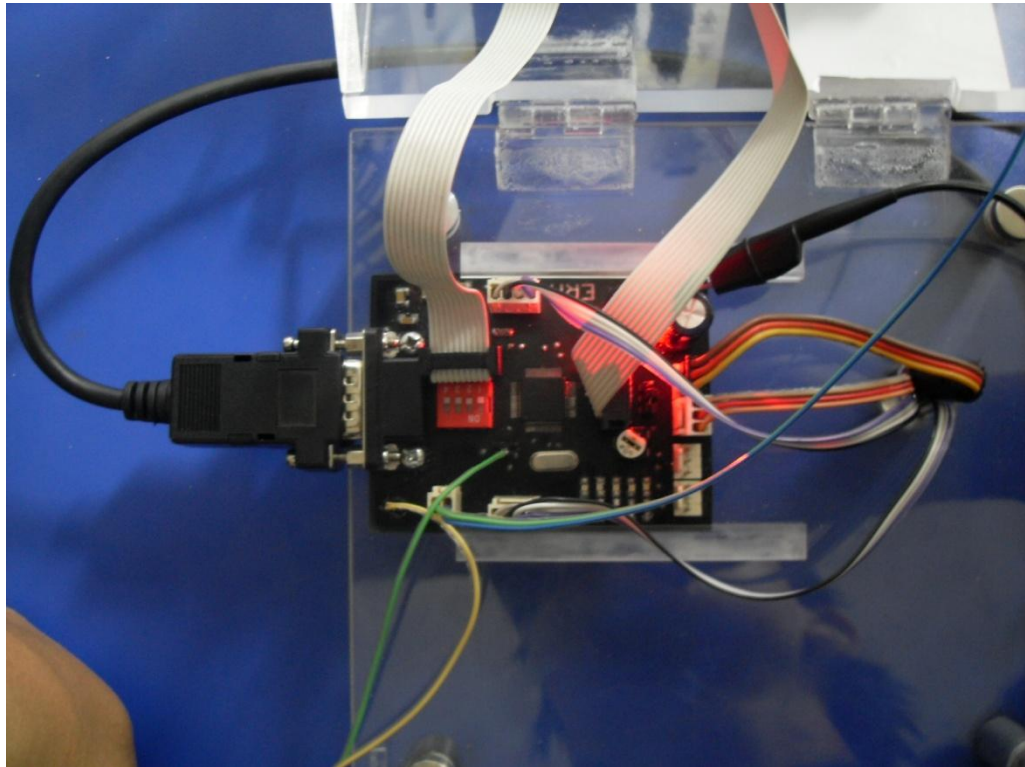
FIGURA 40. Estructura de apoyo del HeliBot 3 DOF

## 5.6 SISTEMA DE CONTROL

Para el control del sistema se decidió diseñar, seleccionar y construir una tarjeta de control mediante un sistema embebido, programada bajo el microcontrolador

*MCF51QE128*, esta tarjeta proporciona los medios para generar las comunicaciones entre el hardware y el software.

En el capítulo 10 se presenta la arquitectura y la metodología de diseño del sistema embebido, además se explica los bloques que componen la tarjeta y las funciones que estos ejercen sobre el sistema. El diseño final de la tarjeta de control se muestra en la figura 41.



**FIGURA 41.** Diseño final de la tarjeta de control

## 6 DISEÑO DE LA ESTRUCTURA BASE DEL HELIBOT

El diseño final de la estructura base del HeliBot puede observarse en la figura 42.



**FIGURA 42.** Diseño final de la estructura base del HeliBot 3 DOF

Esta estructura representa la base del HeliBot, fue construida con lamina HR (hot rolling), con recubrimiento en pintura electrostática negra, el rodamiento de deslizamiento axial utilizado para permitir el movimiento de amortiguación fue seleccionado del catalogo NTN y el data sheet de este rodamiento se encuentra en el anexo D. El resorte se construyo con respecto a las dimensiones que proporcionaban la correcta funcionalidad del sistema, este resorte se cromo para brindar una mayor duración y evitar la corrosión. Utilizando estos elementos, se

desarrollo la estructura completa de la base del HeliBot, la cual está constituida por las partes mostradas en la figura 43.

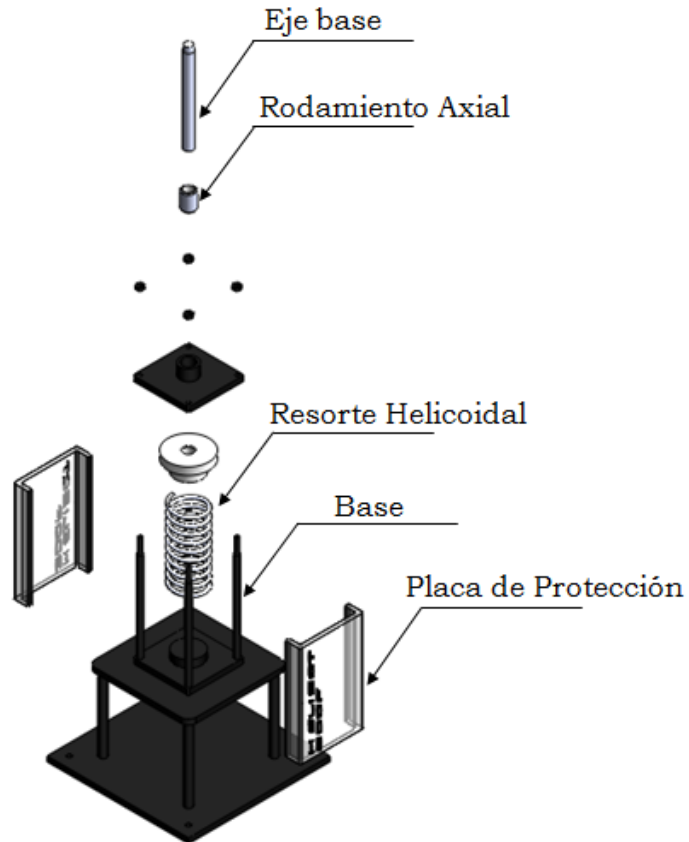


FIGURA 43. Estructura base del HeliBot 3 DOF

## 6.1 DISEÑO DEL RESORTE

Para evitar funcionamientos erróneos del sistema producto de las vibraciones y de los torques inducidos al eje que sostiene el bloque central del HeliBot, se diseñó un resorte, para que cumpliera con la función de suavizar los cambios de dirección

producto de las variaciones en los voltajes en el bloque hélice-motor y la inercia producto de la masa del Bloque superior de la planta dinámica.

Para la correcta selección del resorte primero debemos conocer los parámetros de operación del resorte, los cuales vienen definidos dependiendo la aplicación, para nuestro caso necesitábamos un resorte con una constante de elasticidad  $K$  no muy baja, ya que esto podría ocasionar problemas para estabilizar la planta dinámica, por tanto, sistema oscilaría mucho antes de alcanzar el equilibrio.

Para la selección del resorte contábamos con los siguientes datos:

Diámetro interno= 36 mm

Altura=100 mm

Tipo de extremo= Plano

Paso= Constante

Material= Acero 1003

Diámetro Hilo= 4 mm



**FIGURA 44.** Diseño final del resorte

Cuando la longitud del resorte es grande surge el peligro de que este pierda la estabilidad longitudinal, o sea, su pandeo lateral.

Para los resortes con elementos centradores bien orientados y aseguren el paralelismo entre los topes de los resortes durante la carrera de trabajo, la relación límite entre la altura  $L$  del resorte en estado libre y el diámetro  $D$  de este es igual  $\approx 5$  con lo que el resorte todavía conserva la estabilidad; para los resortes que tienen elementos centradores articulados  $L/D = 2.5^4$

Para nuestro resorte esta relación es igual a:

$$L/D = \frac{100}{40} = 2.5$$

Por tanto la estabilidad de nuestro resorte no se va a ver comprometida.

Ahora para calcular la constante de elasticidad del resorte aplicamos la siguiente ecuación:

$$K = \frac{G * D^4}{64 * N_a * R^3}$$

*Donde*

*G=modulo de Rigidez*

*D= Diámetro del hilo del resorte*

*Na= Numero de vueltas efectivas*

*R=Radio medio del resorte*

---

<sup>4</sup> Vásquez, Leonidas. Diseño de Maquinas. Colombia. Universidad Industrial de Santander. 2001. p.104.

Para resortes con extremos escuadrados y rectificadas se tienen dos vueltas inactivas por lo que tenemos:

$$N_a = N_t - 2$$

Calculamos la constante de elasticidad del resorte para los siguientes valores

$$G = 81.000 \text{ MPa}$$

$$D = 4 \text{ mm}$$

$$N_a = N_t - 2 = 12 - 2 = 10$$

$$R = 20 \text{ mm}$$

$$K = 4050 \text{ N/m}$$

Estos datos habían sido determinados de acuerdo a las posibilidades que nos brindaba las dimensiones del bloque Base del Helibot, ya que contábamos con un volumen exacto para instalar el resorte, después de construido el resorte se le cubrió con una capa de cromo para darle un acabado superficial, que de alguna forma lo hiciera ver bien y que principalmente lo protegiera de la corrosión y el oxido.

Durante la fase de pruebas estructurales y de estabilidad realizadas a la planta en lazo abierto y estable, se validó este diseño ya que cumplía con las condiciones inicialmente establecidas.

## 6.2 SELECCIÓN DEL RODAMIENTO LINEAL

Es apropiado emplear rodamientos de bolas por la naturaleza de las cargas, economía y además porque son estandarizados internacionalmente, intercambiables y fáciles de obtener. Estos rodamientos pueden soportar tanto cargas radiales, como axiales al mismo tiempo.

Para este cálculo se empleara el catalogo New Technology Network 'NTN' de rodamientos de bolas. Se selecciona el rodamiento NTN KH1228 de la sección B243 por diseño en cuanto al elemento que lo va a soportar.

El diseño exigía que el eje, el cual transmitía la fuerza vertical al resorte la cual era aproximadamente igual al producto de la masa del bloque superior por la gravedad, oscilara libremente con la menor fricción posible, por tanto, se instalo en la tapa superior del bloque base del sistema un buje el cual iba a alojar el rodamiento axial para que permitiera el libre movimiento vertical del sistema.

El diámetro interno del rodamiento debe ser aproximadamente  $\varnothing = 1/2$  ", ya que fue el diámetro externo elegido para nuestro eje vertical.

Para la selección del rodamiento axial de bolas contamos con los siguientes datos y la tabla 4.

Diámetro interno= 12 mm

Altura= 30 mm

Rodadura= Bolas

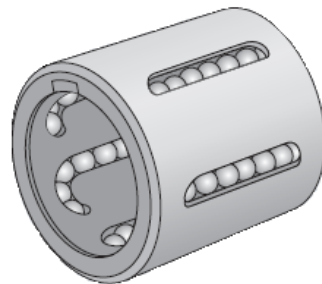


FIGURA 45. Rodamiento lineal NTN

Boundary dimensions				Bearing numbers	Basic load ratings				Number of ball rows	Mass kg (approx.)
mm					dynamic	static	dynamic	static		
$F_w$	$D$	$C$	$a^{(1)}$		N	kgf				
					$C_r$	$C_{or}$	$C_r$	$C_{or}$		
6	12	22	4	KH0622 <sup>(2)</sup>	380	225	39	23	4	0.007
8	15	24	5	KH0824 <sup>(2)</sup>	420	255	43	26	4	0.012
10	17	26	5	KH1026 <sup>(2)</sup>	480	325	49	33	4	0.015
12	19	28	6	KH1228	605	495	62	51	5	0.018
	19	28	6	KH1228LL	605	495	62	51	5	0.018

**TABLA 6.** Selección del rodamiento lineal KH 1228

El rodamiento axial de bolas con estas características fue seleccionado del catalogo de rodamientos NTN, con las siguientes dimensiones 12x19x28, como se puede ver en el anexo D.

### 6.3 DISEÑO DEL EJE VERTICAL

El eje, en nuestro diseño se comporta como una columna que en su parte superior tiene un hombro para instalar un rodamiento de bolas radial y en su otro extremo tiene un roscado para hacerlo solidario con la placa que conecta con el resorte.

Las dimensiones del eje son:

- Diámetro= 12 mm
- Altura= 150 mm
- Material= Acero AISI 1020
- Tratamiento térmico= Endurecimiento
- Acabado superficial= Cromoduro



**FIGURA 46.** Diseño final del eje

Calculamos el eje como si fuera una columna empotrada en sus extremos, y el módulo de elasticidad del acero es  $E = 30 * 10^6$  psi, y la resistencia de fluencia es de  $S_y = 51000$  psi

Hallamos primero el cambio crítico  $C_c$  y lo comparamos con la relación de esbeltez  $Re$  para determinar si la columna es larga o intermedia y aplicar las ecuaciones respectivamente.

$$C_c = \sqrt{\frac{2 \cdot \pi^2 \cdot E}{S_y}} = \sqrt{\frac{2 \cdot \pi^2 \cdot 30 \cdot 10^6}{51000}} = 108$$

En general  $r_g = \sqrt{\frac{I}{A}}$  para una sección circular maciza  $I = \frac{\pi d^4}{4}$ , por tanto  $r_g = \frac{d}{4}$ , entonces  $r_g = \frac{d}{4} = 0.3cm$ .

Calculamos ahora la relación de esbeltez

$$Re = \frac{Leq}{r_g}$$

$$Leq = 0.4L = 0.4 * 15cm = 6 cm$$

$$Re = \frac{6}{0.3} = 20 \ll C_c = 108 \quad \text{Columna Intermedia (J.B. JOHNSON)}$$

$$\frac{P_{cr}}{A} = S_y - b(Re)^2$$

$$b = \frac{S_y}{2C_c^2}$$

$$b = \frac{51000}{2 * (108)^2} = 2.19 \text{ psi}$$

$$A = \frac{\pi d^2}{4} = \frac{\pi(0.4724 \text{ in})^2}{4} = 0.1753 \text{ in}^2$$

$$\frac{P_{cr}}{0.1753} = 51000 - 2.19(20)^2 = 8786.74 \text{ Lb} - f$$

Con esta carga, la columna comenzaría apenas a pandearse. Una carga segura tendría un valor menor, que se calcula al aplicar el factor de diseño la carga crítica.

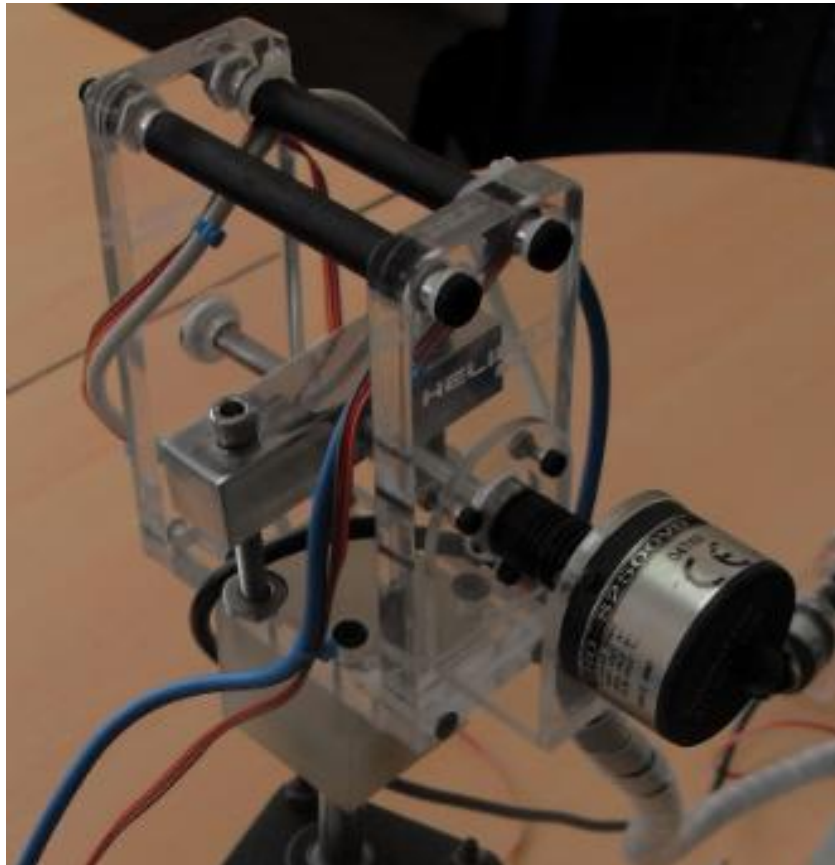
Se empleara  $N=3$  para calcular la carga admisible  $P_{adm} = \frac{P_{cr}}{N}$ :

$$P_{adm} = \frac{P_{cr}}{N} = \frac{8786.74}{3} = 2929 \text{ lb} - f$$

Aproximadamente la fuerza real que soporta el eje son  $11.24 \text{ lb} - f$ , por tanto el eje vertical jamás va a fallar con la carga aplicada a este.

## 7 DISEÑO DEL BLOQUE CENTRAL DEL HELIBOT

El diseño final del bloque central del HeliBot se muestra en la figura 47.



**FIGURA 47.** Diseño final del bloque central del HeliBot

La estructura del bloque central tiene como función la de soportar la viga principal del sistema y proporcionar los medios para generar los movimientos de rotación y el movimiento de elevación de la viga que va a ser medido por el encoder incremental TRD-S2500 VD, el cual viene unido al eje por medio de un acople de referencia GJ-6D en fibra de vidrio seleccionados de automationdirect<sup>5</sup> y sus

---

<sup>5</sup> [www.automationdirect.com](http://www.automationdirect.com)

características se encuentran en el anexo E, el bloque central del HeliBot está constituido por las partes mostradas en la figura 48.

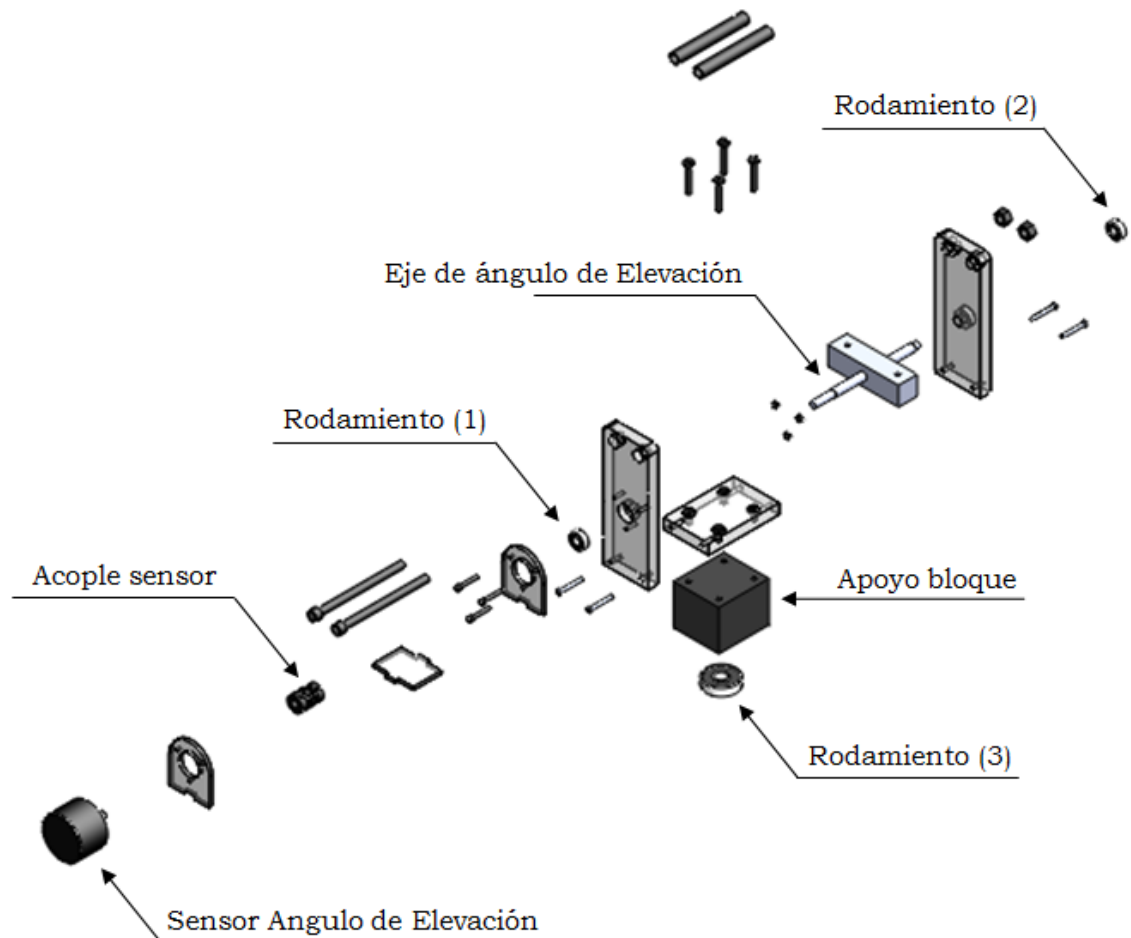


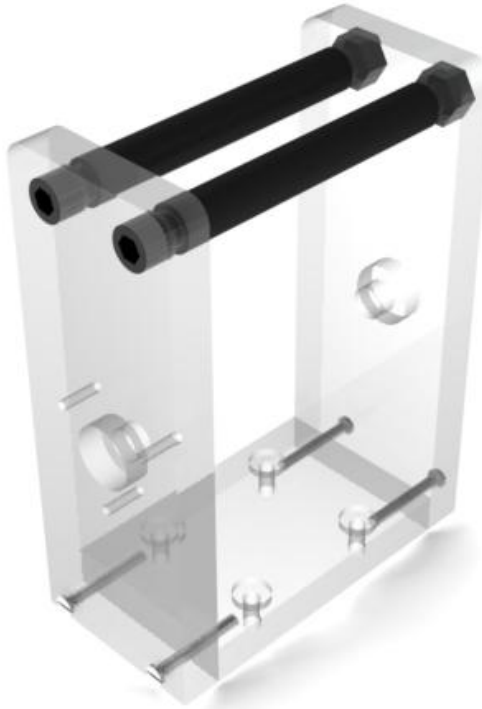
FIGURA 48. Despiece bloque central del HeliBot

## 7.1 SOPORTE BLOQUE CENTRAL

El diseño del soporte de acrílico fue muy importante ya que este iba a albergar los rodamientos que soportarían el eje, esta pieza tenía que ser completamente exacta ya que algún desalineamiento en sus partes representarían una incorrecta medición del ángulo de elevación, por tanto, se tenía que garantizar que los rodamientos entraran a presión en las cavidades hechas en los laterales del

soporte, para que no se permita juegos o tolerancias que comprometieran la correcta medición y control del sistema.

En la figura 49 se muestra el soporte bloque central.



**FIGURA 49.** Soporte bloque central

Siguiendo la premisa inicial que se planteo al principio del proyecto, que era la de construir una planta modular, el soporte está compuesto de tres piezas desarmables en acrílico transparente de calibre 10 mm, la cual tiene en sus laterales los orificios donde irán montados los rodamientos junto con el eje, y la base tiene cuatro orificios donde irán los tornillos golosos de 30 mm de largo los cuales unirán el soporte de acrílico con una base en teflón donde se alberga otro rodamiento radial, el cual se especificara más adelante .

Los tornillos junto con los bujes instalados en la parte superior del soporte de acrílico mantiene una separación constante entre las piezas laterales y a la vez, sirve como restricción, ya que impide que el bloque viga principal choque contra la mesa.

## 7.2 SELECCIÓN DEL RODAMIENTO DE BOLAS

Se optó por la selección de un rodamiento radial de bolas, ya que este nos soportaba alta carga radial y de empuje, aunque poco desalineamiento. Una de las ventajas de utilizar este tipo de rodamiento es por su economía, flexibilidad y de fácil consecución en el mercado.

Una vez establecido el diámetro del eje donde se va a soportar el rodamiento procedemos a seleccionar el rodamiento del catalogo general de la NTN teniendo ciertas características tanto técnicas como geométricas del rodamiento.

$d$  = Diametro de barreno= 10 mm

$L_{10h}$  =Vida nominal esperada= 20.000 horas de operación

$n$  = 10 rpm= Velocidad de rotación

Con los valores de la velocidad angular y la vida nominal esperada, hallamos los factores  $f_n$  y  $f_h$ , los cuales son igual a:

$$f_n = 1.49$$

$$f_h = 3.42$$

Rodamientos de bolas      Rodamientos de rodillos

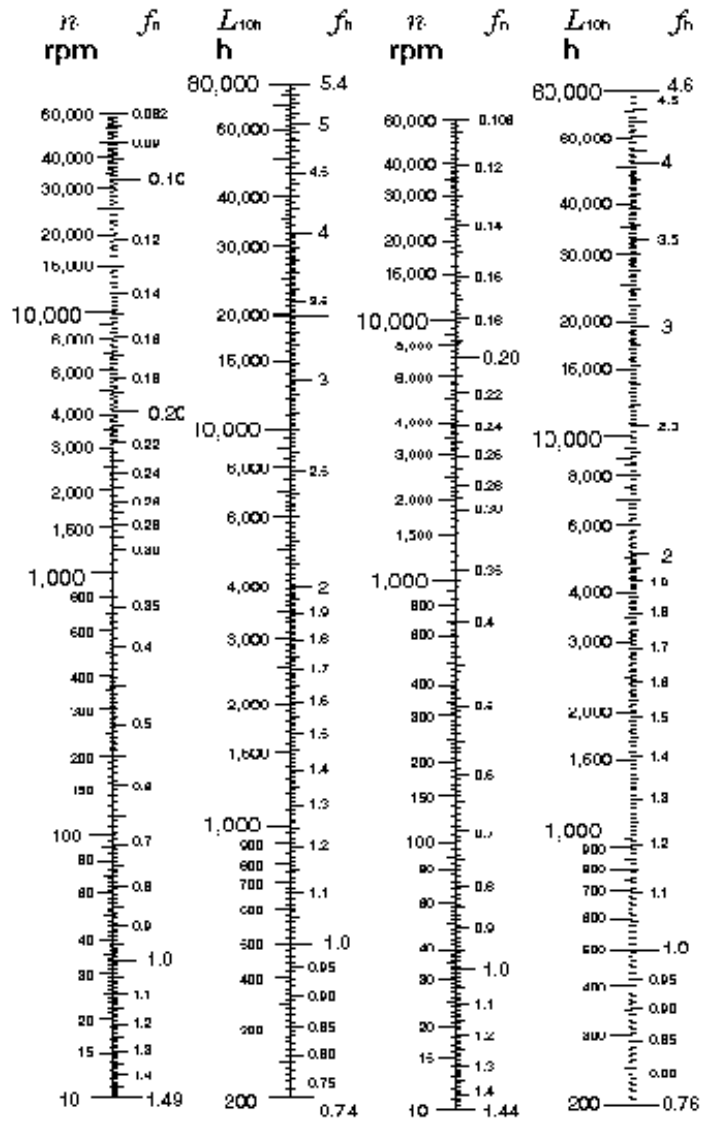


FIGURA 50. Escala para el cálculo de la vida nominal básica de los rodamientos

Como el rodamiento está sometido a carga radial y de empuje aplicamos la siguiente ecuación:

$$P_r = XF_r + YF_a$$

Donde:

$$F_r = 10 \text{ N}$$

$$F_a = 50 \text{ N}$$

$X$  = Factor de carga radial

$Y$  = Factor de carga axial

Ahora determinamos los factores  $X$  y  $Y$  haciendo uso de la siguiente tabla.

$\frac{f_0 \cdot F_a}{C_{or}}$	$e$	$\frac{F_a}{F_r} \leq e$		$\frac{F_a}{F_r} > e$	
		$X$	$Y$	$X$	$Y$
0.172	0.19				2.30
0.345	0.22				1.99
0.689	0.26				1.71
1.03	0.28				1.55
1.38	0.30	1	0	0.56	1.45
2.07	0.34				1.31
3.45	0.38				1.15
5.17	0.42				1.04
6.89	0.44				1.00

**TABLA 7.** Carga radial dinámica equivalente

$$\frac{f_0 F_a}{C_{or}} = \frac{13.2 * 0.05 \text{ KN}}{2.39} = 0.2761$$

Interpolando en la tabla 5 para hallar los valores de  $e$  y de  $Y$ .

$e = 0.21$ ; Por tanto  $X=0.56$  y  $Y=2.11$

$$P_r = X F_r + Y F_a = 0.56 * 10 + 2.11 * 50 = 0.11 \text{ KN}$$

Ahora hallamos la capacidad del rodamiento  $C_r$

$$C_r = \frac{f_h}{f_n} * Pr$$

$$C_r = \frac{f_h}{f_n} * Pr = \frac{3.42}{1.49} * 0.11 = 0.25 \text{ KN}$$

En la página B-8 del catalogo de la NTN, el menor rodamiento que cumple con todos los requerimientos de esta aplicación es el **6200** ( $C_r=5.10 \text{ kN}$ ) {520 kgf}.

### 7.3 ESTRUCTURA SOPORTE DEL ENCODER DE ELEVACION

La estructura del montaje del encoder que censa el ángulo de elevación se fabrico en acrílico calibre 3 mm para evitar que el sensor quede suspendido y se produzca desalineamientos entre su eje y el acople. Esta estructura se monta a un lado del soporte bloque central del HeliBot utilizando tornillos brístol de 15 mm de largo. Debido al reducido peso del encoder, y a la ausencia casi total de cargas importantes en este, se obviaron los cálculos de esfuerzo de la estructura soporte. La figura 51 muestra el resultado final de La estructura de montaje del encoder.



**FIGURA 51.** Soporte sensor de elevación

El soporte del sensor se diseño en Solidworks 2010 y se exporto a coreldraw X4 para ser cortado en acrílico transparente de 3 mm.

#### 7.4 SELECCIÓN DEL SENSOR

Para el control del ángulo de elevación de la planta dinámica se utilizo un encoder incremental TRD- S2500 VD, de *automationdirect* ya que el director de este proyecto de grado los tenia de anteriores trabajos y estos se ajustaban correctamente a las necesidades de medición. La figura 52 muestra el enconder incremental utilizado.

En el anexo B se muestran las características técnicas de este sensor.

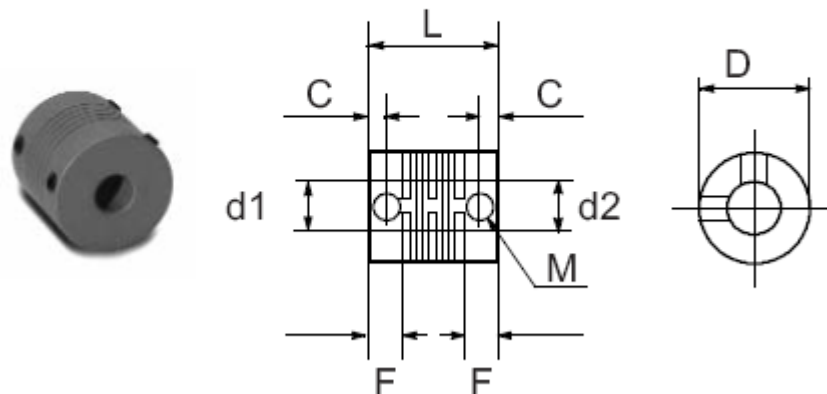
#### TRD-S2500-VD



**FIGURA 52.** Encoder incremental TRD-S2500-VD  
Fuente: [www.automationdirect.com/encoders](http://www.automationdirect.com/encoders)

Para conectar el encoder con el eje se utilizo el acople en fibra de vidrio GJ-6D de *Automationdirect*, el cual aparte de cumplir con la función de conectar estas piezas tolera cierto desalineamiento ya que el acople presenta un cuerpo flexible, en la figura 53, se presenta el esquema de este acople.

## GJ-6D, GJ-8D, and GJ-10D fiber-glass couplings



**FIGURA 53.** Acople de encoder de 6 mm (Fibra de vidrio)

Fuente: [www.automationdirect.com/encoders/encodersaccessories](http://www.automationdirect.com/encoders/encodersaccessories)

### 7.5 DISEÑO EJE DE ELEVACION

El soporte del ángulo de elevación se encuentra compuesto por un eje perpendicular al plano de movimiento del HeliBot, el cual se convierte en el pivote de la viga principal y viene conectado al sensor por medio de un acople en fibra de vidrio.

Debido a que se pretende tener un movimiento de rotación limitado, este eje permite el movimiento de elevación de la viga principal hasta un valor máximo de  $40^\circ$  y un valor mínimo de  $-40^\circ$  con respecto a un plano horizontal.

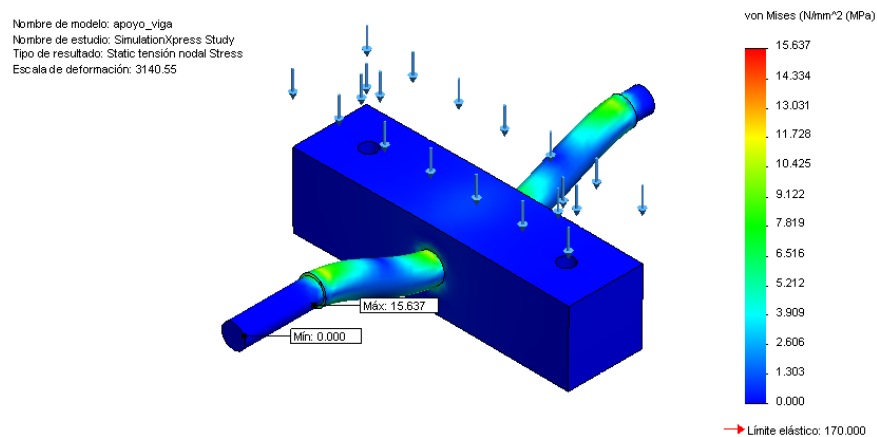
El eje se fija a una platina de aluminio mediante un tornillo prisionero para transmitir de esta forma el movimiento rotacional el cual será medido por el encoder incremental.

Como se puede observar en la figura 54, el eje se encuentra montado en sus extremos en unos rodamientos de bolas, esto se debe a consideraciones de montaje, ya que al instalar estos rodamientos, el eje va a girar mas suavemente, disminuyendo así la fricción de rotación y la inercia y de esta forma aumentar el tiempo de respuesta de control.



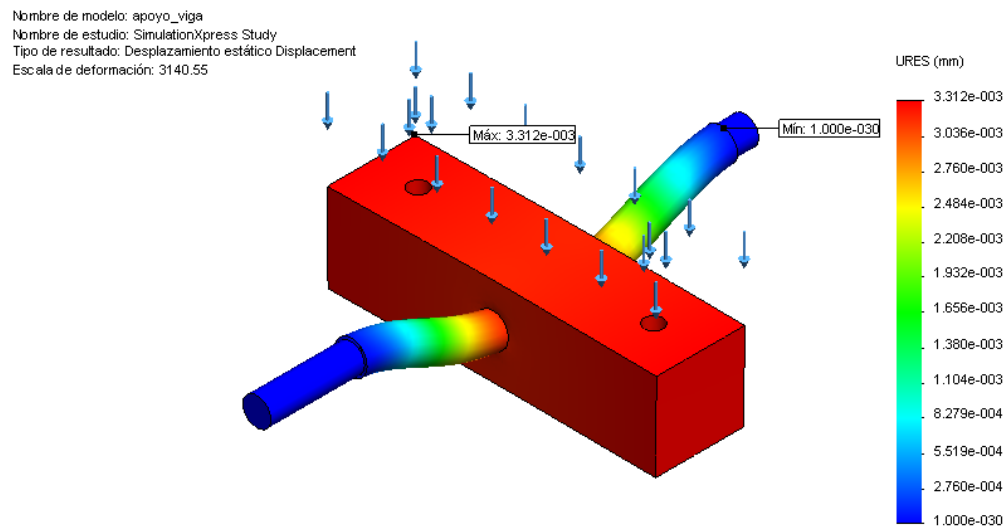
**FIGURA 54.** Soporte eje de elevación

Los cálculos estructurales de esfuerzo y deformación de las piezas relacionadas al soporte de elevación del HeliBot, se hicieron mediante el programa SimulationXpress Study de Solidworks 2010. El resultado del análisis se muestra en la figuras 55 y 56.



**FIGURA 55.** Simulación de esfuerzos en el eje de elevación

El eje del ángulo de elevación se fabrica en Acero inoxidable AISI 316L. En el gráfico de simulación de esfuerzos, se observa que el valor máximo de esfuerzo sobre este eje alcanza los 15.64 MPa. El esfuerzo máximo admisible para un material de este tipo es 485 MPa, lo que nos da un factor de seguridad para este elemento de 31.01, la deflexión máxima de este eje es de 0.00331166 mm, tal y como se muestra en la figura 56.



**FIGURA 56.** Análisis de deflexión en el eje de elevación.

Para la simulación de estos datos se estimo un peso total del Bloque Viga principal junto con el paquete motriz de 5.0968 Kg.

## 8 DISEÑO DEL BLOQUE VIGA PRINCIPAL

El diseño final del bloque viga principal se muestra en la figura 57.



**FIGURA 57.** Diseño final bloque viga principal

El bloque viga principal proporciona los medios para generar el movimiento de elevación y da el soporte necesario para el movimiento de alabeo.

El brazo de la viga principal se fabrica en aluminio anodizado de sección cuadrada tubular de 1.9 mm; el brazo se diseñó tubular principalmente para hacer la estructura más liviana y de esta forma disminuir la inercia que genera dificultad en la estabilización y en la respuesta de control generada por el sistema y para facilitar el cableado hacia el paquete motriz, evitando el enredo de los cables cuando el sistema empieza a girar en su eje vertical. El bloque viga principal está compuesto por los siguientes sub-bloques mostrados en la figura 58.

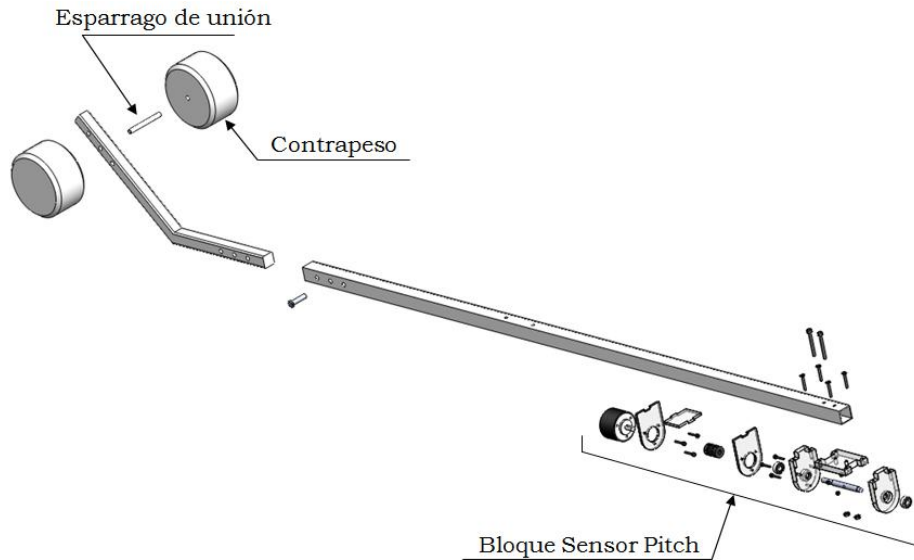


FIGURA 58. Estructura Bloque viga principal

## 8.1 DISEÑO BRAZO PRINCIPAL

Los cálculos estructurales de esfuerzo y deformación de la viga principal fueron hechos utilizando el programa SimulationXpress Study el cual genero las siguientes graficas.

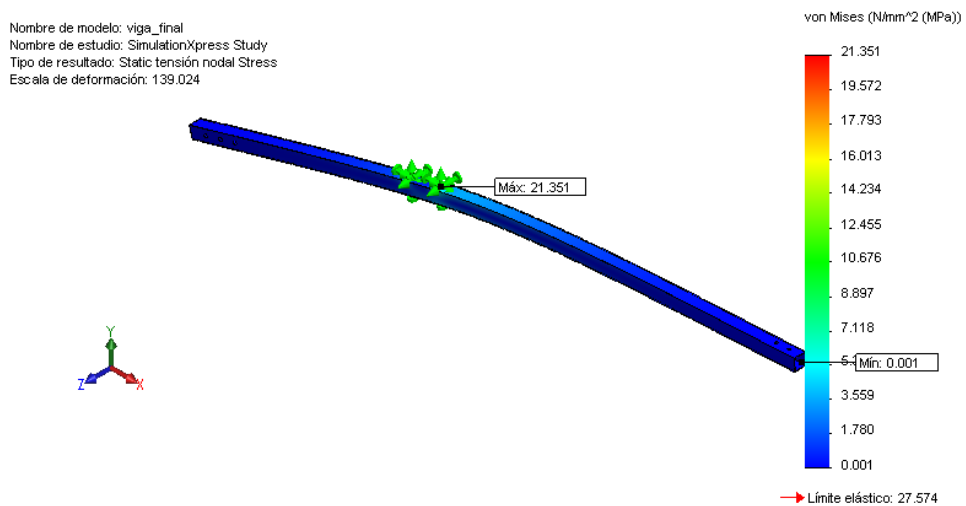
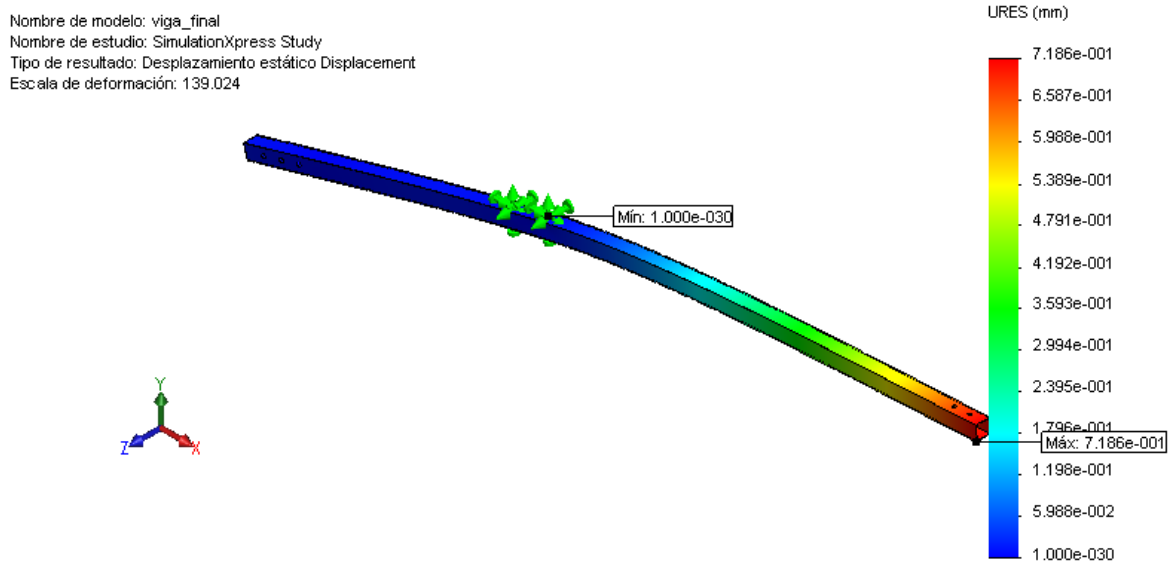


FIGURA 59. Simulación de esfuerzos en la viga principal

Aproximadamente las fuerzas que actúan en los extremos de la viga producto de las masas de 1 Kg son 10 N, para este análisis se supuso una distribución de fuerzas constantes de 10 N sobre toda la superficie de la viga, con todo esto el esfuerzo máximo fue de 21.351 MPa, comparado con el esfuerzo admisible para el aluminio 1060 el cual es 27.57 MPa el factor de seguridad es 1.29.



**FIGURA 60.** Análisis de deformación en la viga principal.

La desplazamiento máximo para esta viga es 0.718575 mm y tiene lugar en el extremo superior del brazo.

La longitud total de la viga es de 1m, vemos por tanto que el análisis estructural de esfuerzos por en método de Von Mises, da un resultado positivo y nuestra viga no va a fallar con la carga de trabajo aplicada.

## 8.2 DISEÑO DEL BLOQUE SENSOR DE ALABEO

Básicamente es el encargado de censar el ángulo de alabeo el esquema se muestra en la figura 61.

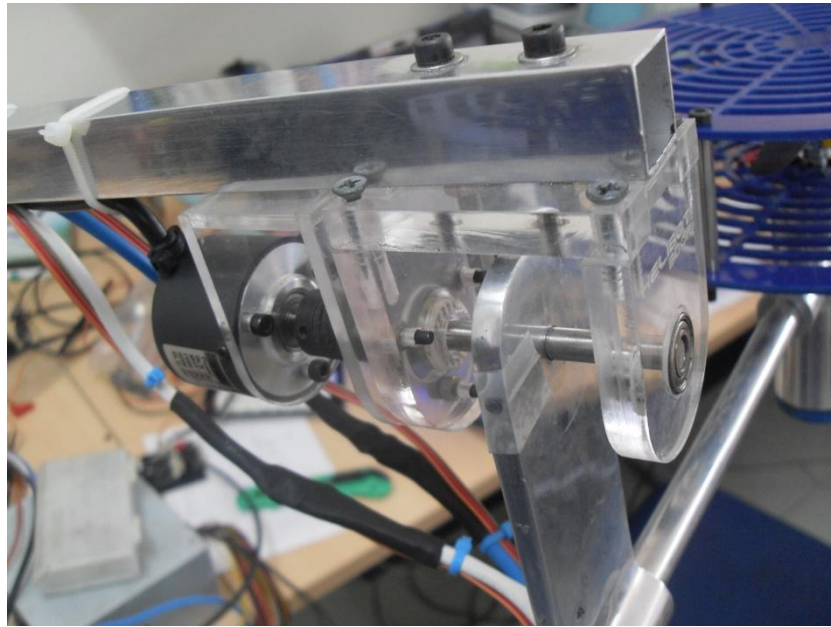


FIGURA 61. Bloque sensor de alabeo

En la figura 62 se muestra las partes correspondientes del sub-bloque sensor de alabeo.

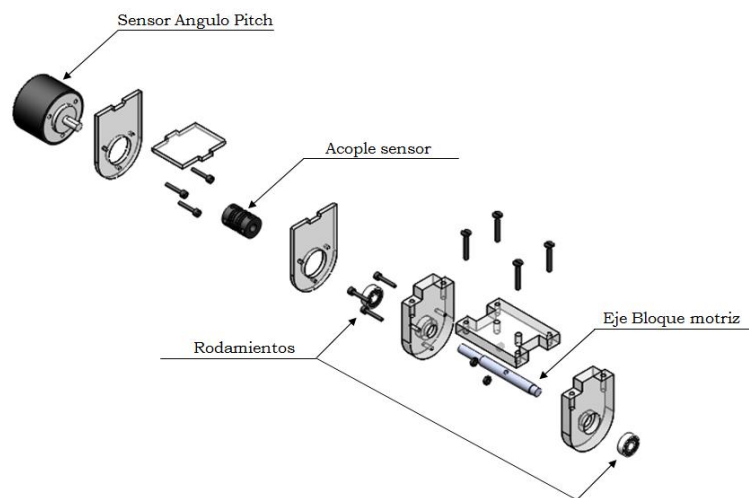


FIGURA 62. Partes del bloque sensor de alabeo

El sensor utilizado para censar el ángulo de alabeo, es un encoder incremental similar al utilizado para medir el ángulo de elevación, las características técnicas de este encoder fueron dadas anteriormente y el datasheet se puede ver en el Anexo B.

Al igual que el encoder, el soporte del sensor de alabeo presenta las mismas características tanto de diseño como de materiales que el soporte del encoder de elevación mostrado anteriormente, por tanto también se obvia de nuevo este proceso de diseño.

### **Diseño del eje ángulo de alabeo**

El eje de elevación se encuentra compuesto por un eje perpendicular al plano de movimiento longitudinal del HeliBot, el cual se convierte en el pivote de la barra del bloque hélice - motor y viene conectado al sensor por medio de un acople en fibra de vidrio.

Debido a que se pretende tener un movimiento de rotación limitado, este eje permite el movimiento de cabeceo del paquete motriz hasta un valor máximo de  $60^\circ$  y un valor mínimo de  $-60^\circ$  con respecto a un plano horizontal.

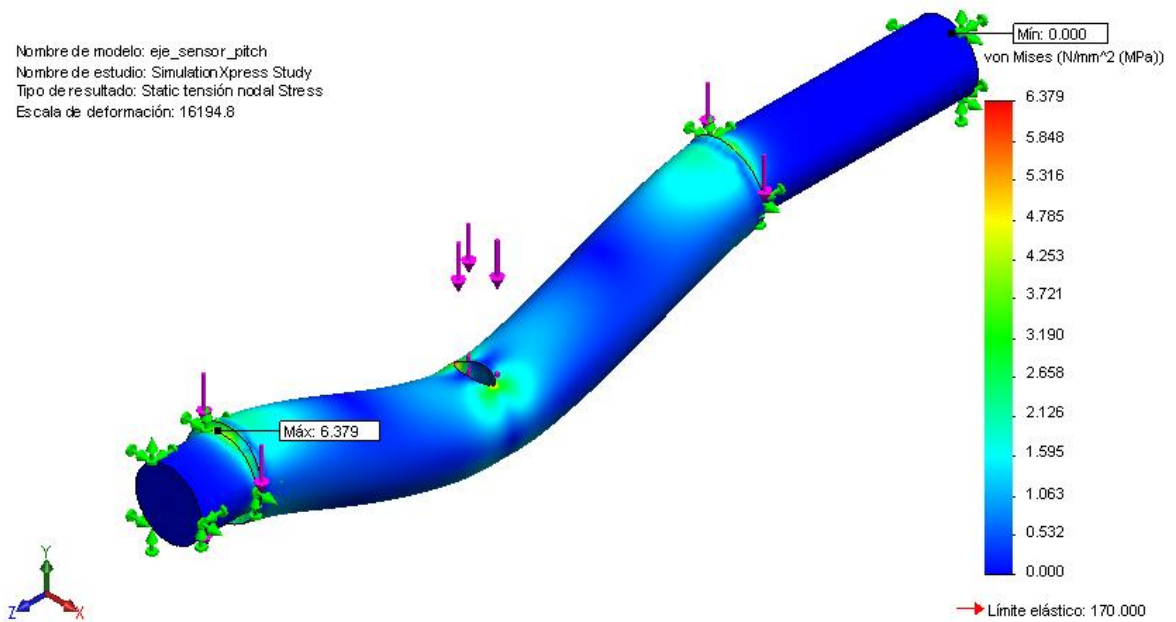
El eje se fija a cada lado de sus extremos al soporte de acrílico por medio de dos rodamientos de bolas, los cuales le permiten el libre movimiento de rotación y por consiguiente la medición de la variación del ángulo de alabeo, a través de un encoder incremental acoplado a este.

En la figura 63 se muestra el eje del ángulo de elevación.



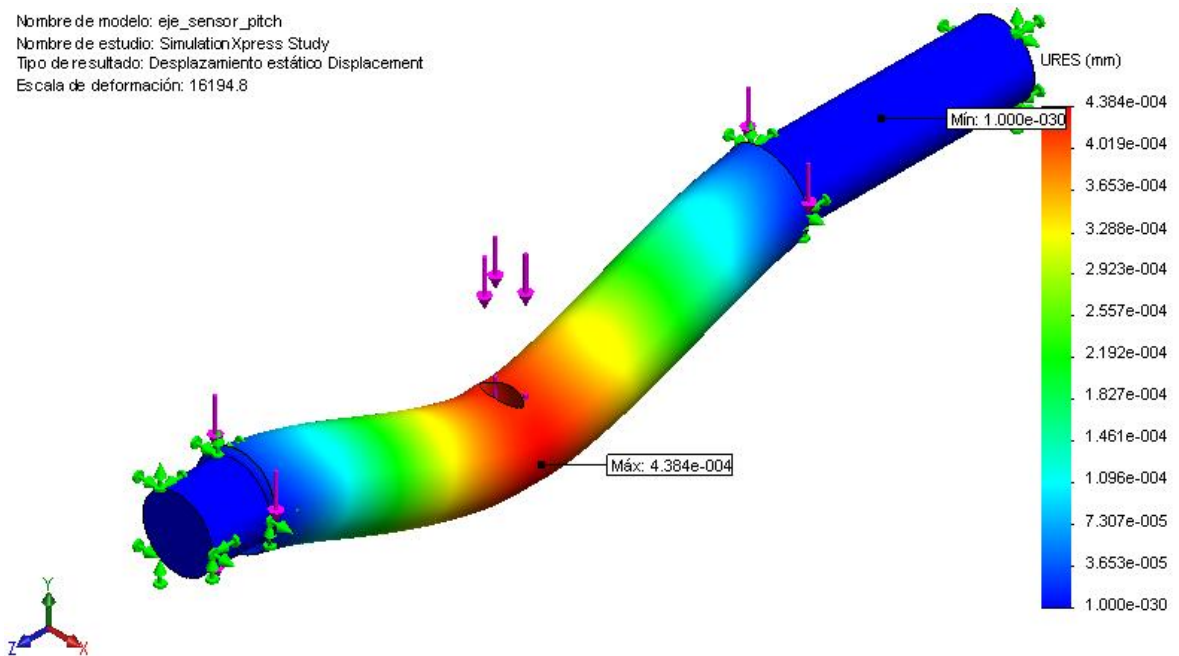
**FIGURA 63.** Eje ángulo de alabeo

Los cálculos estructurales de esfuerzo y deformación del eje del ángulo de alabeo del HeliBot, se hicieron mediante el programa SimulationXpress Study de Solidworks 2010. El resultado del análisis se muestra en la figuras 64 y 65.



**FIGURA 64.** Simulación de esfuerzos en el eje de alabeo

El eje del ángulo de elevación se fabrica en Acero inoxidable AISI 316L. En el gráfico de simulación de esfuerzos, se observa que el valor máximo de esfuerzo sobre este eje alcanza los 6.37945 MPa. El esfuerzo máximo admisible para un material de este tipo es 485 MPa, lo que nos da un factor de seguridad para este elemento de 76.025, la deflexión máxima de este eje es de 0.000438412 mm, tal y como se muestra en la figura 65.



**FIGURA 65.** Análisis de deformación en el eje de alabeo.

Para la simulación de estos datos se estimó un peso total del bloque hélice-motor de 2.0387 Kg.

### 8.3 DISEÑO DEL CONTRAPESO

El contrapeso es el encargado de ayudar a estabilizar la planta dinámica, está ubicado en el extremo opuesto del brazo de la viga principal a una distancia  $L/3$  del

bloque base del Helibot, exactamente a 33.33 cm. El contrapeso está constituido por dos pesas de aluminio macizo cada una de 1 Libra de masa tal y como se muestra en la figura 66, conectadas entre sí por medio de un esparrago el cual atraviesa el soporte del contrapeso que se muestra en la figura 66.



**FIGURA 66.** Diseño final contrapeso

Esta pieza fue fabricada en aluminio estructural macizo de sección cuadrada de 1.8 cm, esta va incrustada dentro de la sección cuadrada tubular de la viga principal la cual se puede graduar la distancia por medio de unas guías hechas ajustadas por medio de un tornillo prisionero, en la parte superior de este soporte se fabricaron también unos huecos pasantes de 3/8 de diámetro separados cada uno 3 cm de longitud, estos también nos ayudan a variar el momento al cambiar la distancia del contrapeso.



**FIGURA 67.** Soporte del contrapeso

La razón de hacer este soporte a un ángulo de  $45^\circ$  es que aumenta el torque efectivo del contrapeso con respecto a la base, ya que aparecen dos componentes del peso adicionales.

## 9 DISEÑO DEL BLOQUE HELICE – MOTOR

El diseño final del bloque hélice-motor se muestra en la figura 68.



**FIGURA 68.** Diseño final del bloque hélice-motor

El bloque hélice-motor oscila alrededor del eje de alabeo ubicado en el bloque viga principal, este bloque contiene el paquete motriz, compuesto por el conjunto hélice- motor, donde un aumento en el voltaje en alguno de los dos motores provoca un aumento en el ángulo de alabeo, además de un movimiento de rotación con respecto al eje vertical del sistema. En la figura 69 se muestra el despiece del bloque hélice-motor donde se pueden ver las partes que constituyen esta estructura.

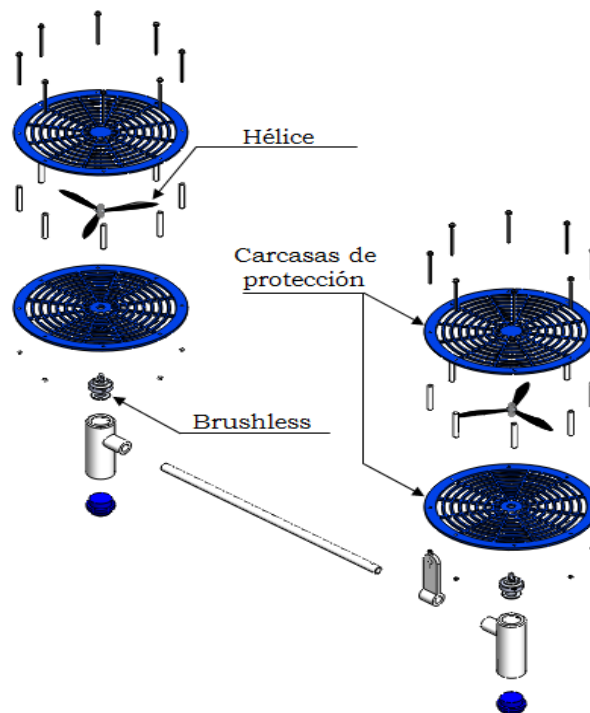


FIGURA 69. Despiece del bloque hélice-motor

## 9.1 DISEÑO DE LAS CARCASAS DE PROTECCION

Para evitar algun accidente con las aspas de las helices, se hizo necesario, la fabricacion de unas tapas arriba y abajo de cada helice con unas rejillas para permitir el paso del aire ademas de servir de soporte para los motores brushless. Estas carcasas fueron fabricadas en acrilico calibre 3 mm, ya que estas cumplan con los requerimientos de rigidez y peso.

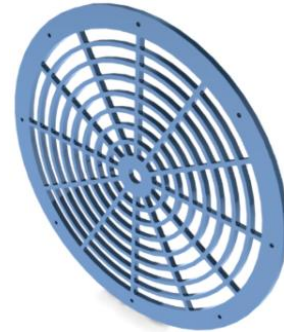
Estas carcasas cuentan con 8 huecos de 3mm de diametro en la superficie externa de la circunferencia , en donde van unos tornillos de 6 cm de largo y 3 mm de diametro con cabeza bristol y unos bujes en acero inoxidable los cuales nos aseguran una distancia de 4.5 cm entre las tapas superior e inferior de acrilico. En la figura 70 se muestra la carcaza de las helices.

*Diametro Externo = 23 cm*

*Diametro interno = 3 cm*

*Espesor = 3 mm*

*Material = Acrílico*



**FIGURA 70.** Carcasa de protección

## **9.2 DISEÑO DEL SOPORTE CONJUNTO HELICE-MOTOR**

Este soporte es el encargado de sostener el conjunto hélice motor, se fabrica en aluminio estructural A 36, su diseño en forma de pipa se debe a las siguientes consideraciones técnicas:

- Permitir la conexión con otro eje hacia el siguiente soporte
- Albergar dentro de su interior hueco el speed control
- Sostener en su tapa superior el soporte del motor y la carcasa
- Bajo peso
- Resistencia a golpes al caer

El tapón inferior se fabrica en teflón, para disminuir un poco el impacto del bloque hélice – motor en las pruebas de estabilidad y es desmontable, para facilitar el intercambio de piezas y el mantenimiento de los elementos internos.

En la figura 71 se muestra el diseño final del soporte conjunto hélice-motor.



**FIGURA 71.** Soporte conjunto hélice - motor

En el anexo J, se pueden encontrar los planos de esta pieza, donde se especifican claramente las dimensiones y los materiales utilizados.

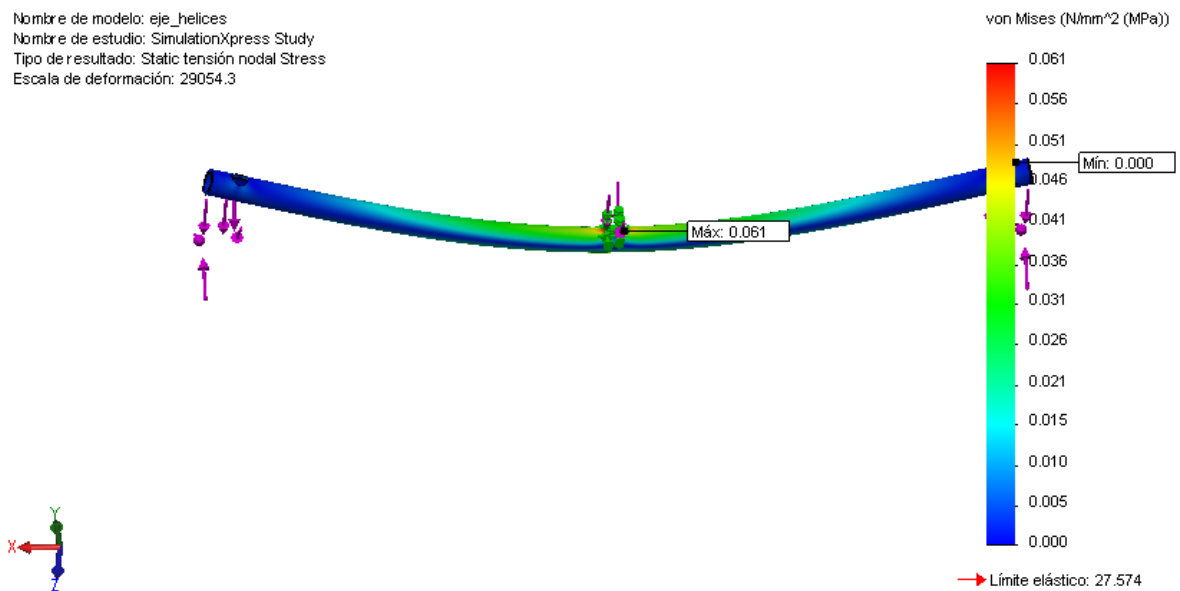
### **9.3 DISEÑO DEL EJE BLOQUE HELICE-MOTOR**

Este eje sostiene en cada uno de sus extremos un conjunto hélice-motor y esta pivotado en su centro, está fabricado de aluminio 1060 de sección circular hueca, al igual que las consideraciones anteriores esto ayuda a disminuir la inercia del elemento y sirve para pasar los cables de los motores hacia la tarjeta. En sus extremos tiene unos huecos de 3/8 in en los cuales va un tornillo prisionero que ajusta el eje a los soportes del conjunto hélice-motor, en su centro tiene otro hueco por el cual ingresan los cables hacia los motores brushless. En la figura 72 se muestra el diseño final del eje.



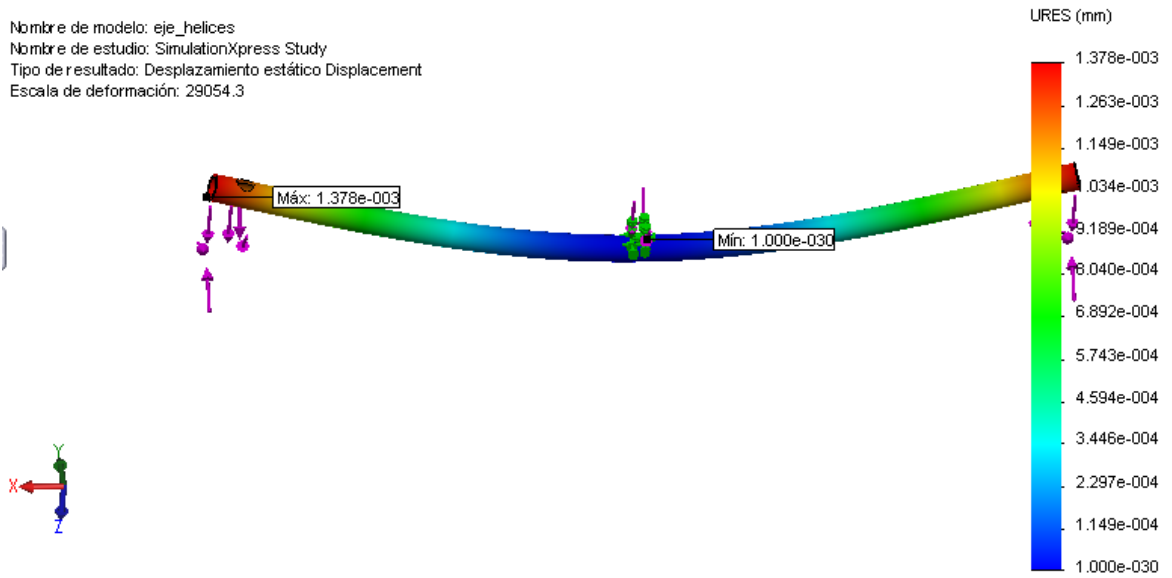
**FIGURA 72.** Diseño final del eje bloque hélice - motor

Los cálculos estructurales de esfuerzo y deformación del eje bloque hélice- motor del HeliBot, se hicieron mediante el programa SimulationXpress Study de Solidworks 2010. El resultado del análisis se muestra en la figuras 73 y 74.



**FIGURA 73.** Simulación de esfuerzos en el eje del bloque hélice-motor

El eje se fabrica en Aluminio 1060. En el gráfico de simulación de esfuerzos, se observa que el valor máximo de esfuerzo sobre este eje alcanza los 0.061 MPa. El esfuerzo máximo admisible para un material de este tipo es 27.547 MPa, lo que nos da un factor de seguridad para este elemento de 451.59, la deflexión máxima de este eje es de 0.00137831 mm, tal y como se muestra en la figura 74.



**FIGURA 74.** Análisis de deformación en el eje bloque hélice-motor.

Para la simulación de estos datos se estimo un peso total de cada conjunto bloque hélice-motor 1 Kg.

## 10 SISTEMA DE CONTROL

### 10.1 SISTEMA DE CONTROL EMBEBIDO

Se conoce como sistema embebido a un circuito electrónico computarizado que está diseñado para cumplir una función específica en un producto.

La inteligencia artificial, secuencias y algoritmos de un sistema embebido, están residentes en la memoria de una pequeña computadora denominada microcontrolador<sup>6</sup>.

#### 10.1.1 Características

- Los sistemas embebidos son diseñados para una aplicación específica, es decir, estos sistemas realizan un grupo de funciones previamente definidas y una vez el sistema es diseñado, no se puede cambiar su funcionalidad. Por ejemplo, el control de un ascensor siempre realizará las mismas acciones durante su vida útil.
- Debido a su interacción con el entorno los ES deben cumplir estrictamente restricciones temporales. El término Sistemas de Tiempo Real es utilizado para enfatizar este aspecto.
- Los Sistemas Embebidos son heterogéneos, es decir, están compuestos por componentes Hardware y Software. Los componentes Hardware, como ASICs y Dispositivos Lógicos Programables (PLD) proporcionan la

---

<sup>6</sup> GALEANO, Gustavo. Programación de sistemas embebidos en C. Colombia. 2009, pág. 3.

velocidad de ejecución y el consumo de potencia necesarios en algunas aplicaciones.

- Los Sistemas Embebidos tienen grandes requerimientos en términos de confiabilidad. Errores en aplicaciones como la aviación y el automovilismo, pueden tener consecuencias desastrosas.

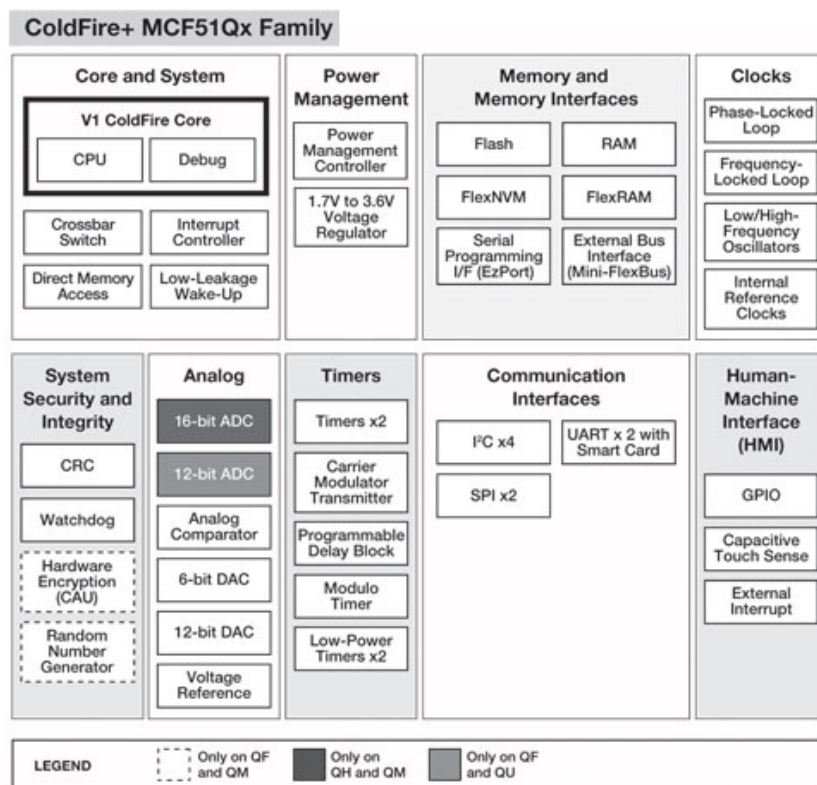


FIGURA 75. Arquitectura de un sistema embebido

Fuente: [www.freescale.com](http://www.freescale.com)

### 10.1.2 Arquitectura de los Sistemas Embebidos

En la Figura 75 se muestra la arquitectura típica de un Sistema Embebido. La cual integra un componente hardware, conocido con el nombre de periféricos y un

componente software (procesador o DSP) capaz de ejecutar software, la parte del procesador está dividida en la CPU (En algunos casos posee una caché) y las unidades de Memoria, en algunas aplicaciones se suele reemplazar el DSP o microcontrolador por un arreglo lógico programable o FPGA.

Al momento de diseñar un Sistema Embebido se pueden dividir los recursos de hardware y software de la siguiente manera:

HW y SW Integrados en un dispositivo semiconductor (SoC): En la actualidad existen muchas compañías que fabrican procesadores de 32 bits integrados a una gran variedad de periféricos, lo cual simplifica el diseño y reduce costos.

Este tipo de implementación es muy popular en los dispositivos de consumo masivo (Reproductores de MP3, consolas de juego, etc.), debido a los grandes niveles de producción (del orden de millones de unidades) resulta más económico contar con un dispositivo que integre el mayor número de funcionalidades, esto disminuye el costo de componentes y reduce el área de circuito impreso.

Componente SW en un SoC y componente HW en una FPGA: Cuando no existen en el mercado un SoC con la cantidad de periféricos requerida para una determinada aplicación, o con una funcionalidad específica, es necesario recurrir a la utilización de dispositivos comerciales que implementen dicha operación, en algunas ocasiones el periférico puede realizar funciones poco comunes y no se proporciona comercialmente, la solución es entonces, implementar estas funcionalidades en una FPGA; también se recomienda la utilización de FPGAs en sistemas que requieren la utilización de la misma funcionalidad un gran número de veces (Puertos seriales, Pines de Entrada/Salida). Esta decisión está atada al

nivel de producción, ya que al incluir una FPGA aumenta el costo global del proyecto.

Componente SW y HW en una FPGA: Esta es tal vez la opción más flexible, pero la de menor desempeño, ya que al utilizar los recursos lógicos de la FPGA para la implementación del procesador (softcore) la longitud de los caminos de interconexión entre los bloques lógicos aumentan el retardo de las señales, lo cual disminuye la máxima velocidad de funcionamiento. Los procesadores softcore más populares en la actualidad son:

- Microblaze y Picoblaze de Xilinx<sup>7</sup>
- Leon de Gaisler Research<sup>8</sup>
- LatticeMico32 de Lattice Semiconductors<sup>9</sup>
- OpenRisc<sup>10</sup>

### 10.1.3 Metodología de Diseño

El proceso de diseño de un Sistema Embebido comienza con la especificación del sistema, (ver Figura 76), en este punto se describe la funcionalidad y se definen las restricciones físicas, eléctricas y económicas. Esta especificación debe ser muy general y no deben existir dependencias (tecnológicas, metodológicas) de ningún tipo.

La especificación puede ser verificada a través de una serie de pasos de análisis cuyo objetivo es determinar los componentes necesarios mínimos para el funcionamiento del sistema, en esta etapa se evalúan los costos, y las

---

<sup>7</sup> <http://www.xilinx.com>

<sup>8</sup> <http://www.gaisler.com>

<sup>9</sup> <http://www.latticesemi.com>

<sup>10</sup> <http://www.opencores.com>

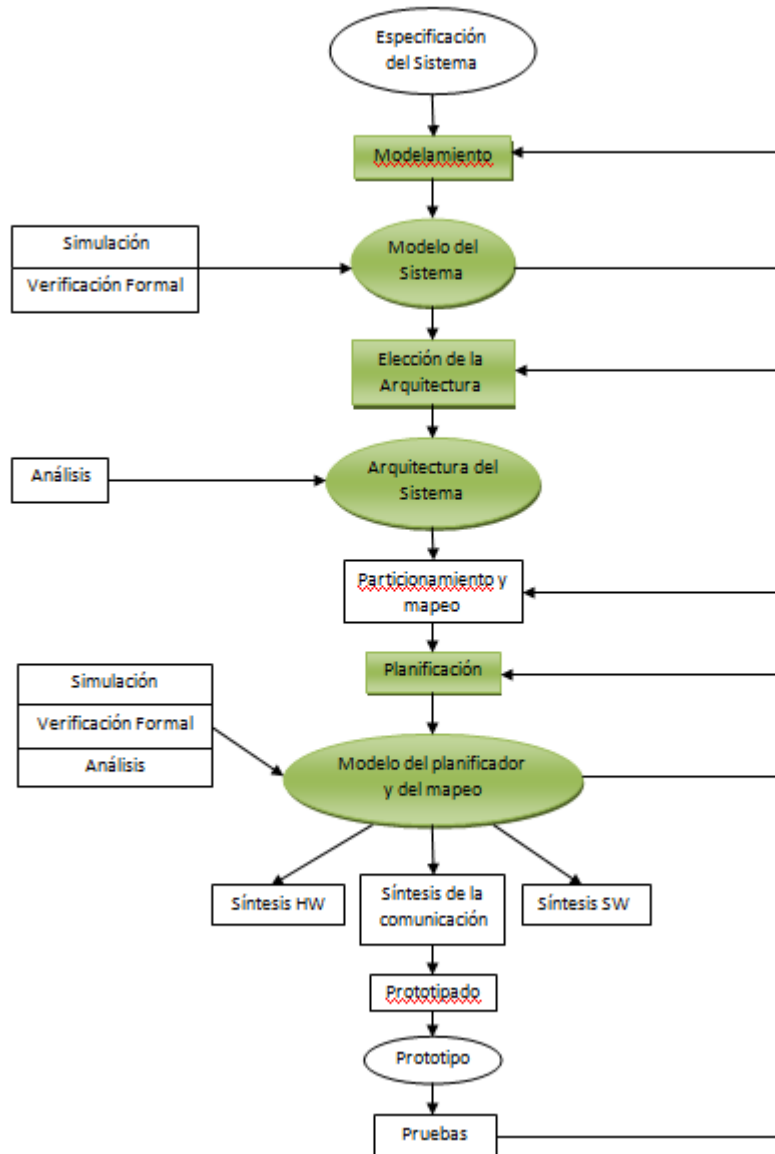
capacidades de cada dispositivo elegido a fin de seleccionar la mejor relación costo/beneficio.

Una vez se ha obtenido el modelo del sistema se procede a determinar su arquitectura, esto es, el número y tipo de componentes y su inter-conexión. Este paso no es más que una exploración del espacio de diseño en búsqueda de soluciones que permitan la implementación de una funcionalidad dada, y puede realizarse con varios criterios en mente: Costos, confiabilidad, viabilidad comercial.

Una vez definidos los componentes para el sistema se realiza una distribución física y de tareas dentro del sistema embebido, a fin de ganar flexibilidad y desempeño.

Una vez definidas las especificaciones del sistema se debe realizar un modelamiento que permita extraer de estas la funcionalidad. El modelamiento es crucial en el diseño ya que de él depende el paso exitoso de la especificación a la implementación. Es importante definir qué modelo matemático debe soportar el entorno de diseño. Los modelos más utilizados son: Máquinas de estados finitos, diagramas de flujos de datos, Sistemas de Eventos Discretos y Redes de Petri. Cada modelo posee propiedades matemáticas que pueden explotarse de forma eficiente para responder preguntas sobre la funcionalidad del sistema sin llevar a cabo dispendiosas tareas de verificación. Todo modelo obtenido debe ser verificado para comprobar que cumple con las restricciones del sistema.

Utilizando como base la arquitectura obtenida en el paso anterior las tareas del modelo de los sistemas son mapeadas dentro de los componentes. Esto es, asignación de funciones a los componentes de la arquitectura.



**FIGURA 76.** Flujo de diseño de un sistema embebido

Para cumplir las especificaciones del sistema algunas tareas deben ser implementadas en Hardware, esto con el fin de no ocupar al procesador en tareas cíclicas, un ejemplo típico de estas tareas es la generación de bases de tiempos. La decisión de que tareas se implementan en SW y que tareas se implementan en

HW recibe el nombre de particionamiento, esta selección es fuertemente dependiente de restricciones económicas y temporales.

Las tareas Software deben compartir los recursos que existan en el sistema (procesador y memoria), por lo tanto se deben hacer decisiones sobre el orden de ejecución y la prioridad de estas. Este proceso recibe el nombre de planificación. En este punto del diseño el modelo debe incluir información sobre el mapeo, el particionamiento y la planificación del sistema.

Las siguientes fases corresponden a la implementación del modelo, para esto las tareas hardware deben ser llevadas al dispositivo elegido (ASIC o FPGA) y se debe obtener el ejecutable de las tareas software, este proceso recibe el nombre de síntesis HW y SW respectivamente, así mismo se deben sintetizar los mecanismos de comunicación.

El proceso de prototipado consiste en la realización física del sistema, finalmente el sistema físico debe someterse a pruebas para verificar que se cumplen con las especificaciones iniciales.

Como puede verse en el flujo de diseño existen realimentaciones, estas realimentaciones permiten depurar el resultado de pasos anteriores en el caso de no cumplirse con las especificaciones iniciales.

## **10.2 MICROCONTROLADOR**

Un microcontrolador es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: unidad central de procesamiento, memoria y unidades de E/S (entrada/salida), este tipo de dispositivos son catalogados como SoC.

Son diseñados para reducir el costo económico y el consumo de energía de un sistema en particular. Por eso el tamaño de la unidad central de procesamiento, la cantidad de memoria y los periféricos incluidos dependerán de la aplicación.

Un microcontrolador típico tendrá un generador de reloj integrado y una pequeña cantidad de memoria RAM y ROM/EPROM/EEPROM/FLASH, significando que para hacerlo funcionar, todo lo que se necesita son unos pocos programas de control. Los microcontroladores disponen generalmente de una gran variedad de dispositivos de entrada/salida, como convertidores de analógico a digital, temporizadores, UARTs (*Universal Asynchronous Receiver-Transmitter*) y buses de interfaz serie especializados, como I<sup>2</sup>C (*Circuitos Inter-Integrados*) y CAN (*Controller Area Network*). Frecuentemente, estos dispositivos integrados pueden ser controlados por instrucciones de procesadores especializados. Los modernos microcontroladores frecuentemente disponen de una interfaz de programación integrada IDE (*Integrated Drive Electronics*), como el BASIC, C, C++ o java que se utilizan bastante con este propósito<sup>11</sup>.

### **Microcontrolador MCF51QE128**

Como se definió en un principio en los objetivos del proyecto de grado, la utilización del microcontrolador Colfire V1 MCF51QE128 de la familia Flexis de Freescale, los cuales tienen las siguientes características técnicas de operación:

50.33 MHz ColdFire core V1 @ 25 MHz bus

Memoria 128 KB (flash)

Memoria 16 KB (RAM)

24 pines (A/D)

---

<sup>11</sup> <http://es.wikipedia.org/wiki/Microcontrolador>

Rango de temperatura de funcionamiento de -40°C a +105°C

Encapsulados entre 64 y 80 pines LQFP

Compatibilidad pin a pin entre versión 8 bits (núcleo HCs08) y 32 bits (coldfire V1).

En el Anexo F, se presenta el data sheet de el microcontrolador utilizado, donde podemos ver la asignación de pines para los encapsulados de 64 y 80 pines, y la diferencia entre estas, así como las características térmicas y eléctricas y planos del microcontrolador MCF51QE128.

### **10.3 SOFTWARE**

El programa es desarrollado de manera secuencial para llevar a cabo todo el proceso de control del sistema desde la adquisición de los datos hasta la acción de control. Este programa está escrito en lenguaje C en el programa Codewarrior suministrado por Freescale para la programación de sus microcontroladores (IDE).

El programa hace uso de librerías creadas por los autores que contiene las diferentes funciones que ejecuta el algoritmo. Aunque se mencionó que el programa se desarrolla de manera secuencial, ciertos procesos son realizados en paralelo gracias a que microcontrolador maneja sus módulos de manera independiente y permite el uso de interrupciones lo que mejora el tiempo de ejecución del programa.

En el anexo H se presenta el programa desarrollado para funcionar en el Microcontrolador de 32 bits, MCF51QE128 de Freescale. Las librerías a las que se hace un llamado para funcionar en el programa principal también son aquí presentados.

En la figura 77, se muestra el diagrama esquemático del microcontrolador y la asignación de las funciones a cada pin, se puede ver las conexiones entre los pines del micro y sus periféricos como lo son los sensores de elevación y cabeceo y los motores del HeliBot, el teclado, LCD, y comunicación serial.

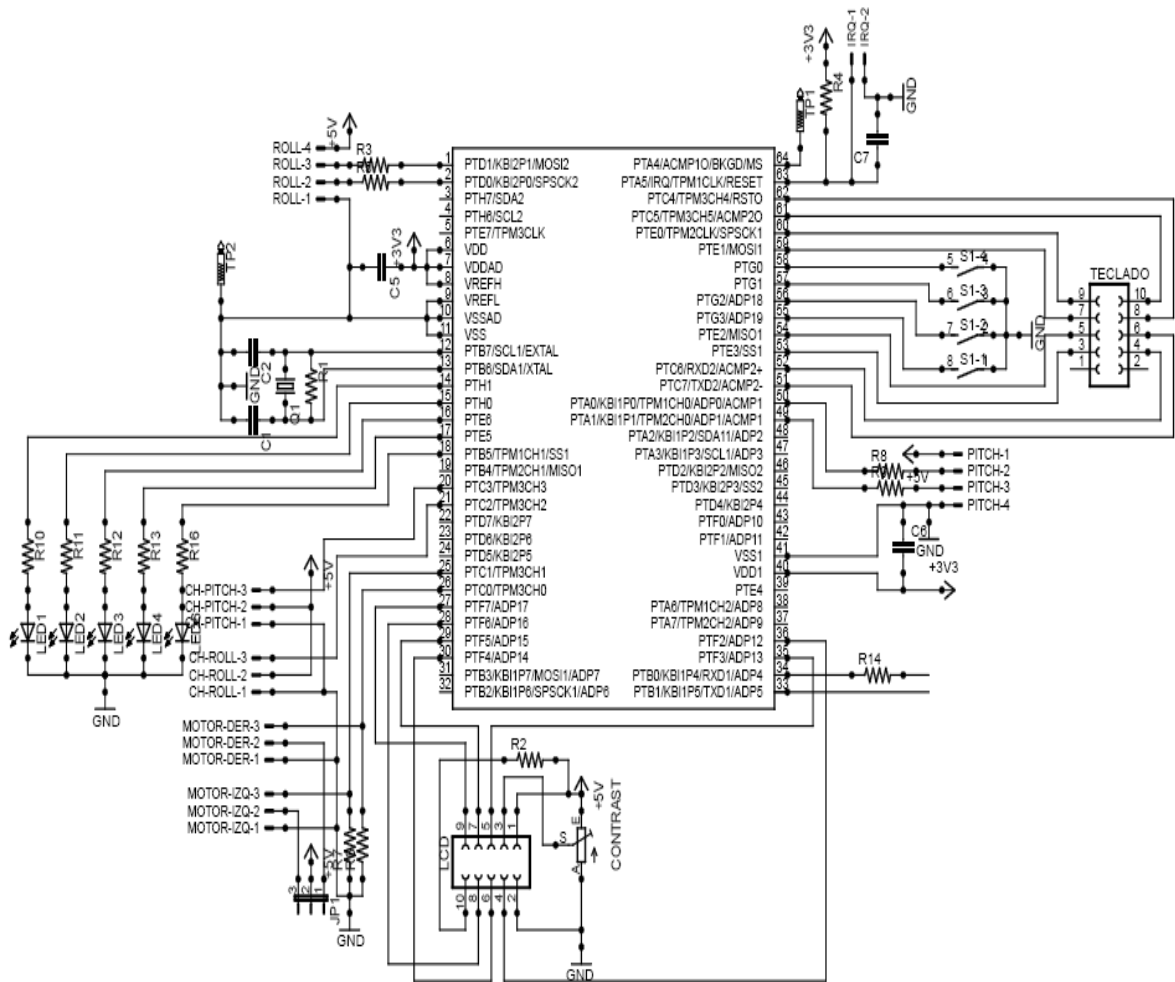


FIGURA 77. Esquemático y asignación de pines del microcontrolador MCF51QE128

En la figura 78 se muestra el diagrama de flujo de la programación del microcontrolador, en el anexo H se muestra el código completo con sus

respectivos comentarios para facilitar la comprensión del programa y facilite los trabajos posteriores en base al nuestro.

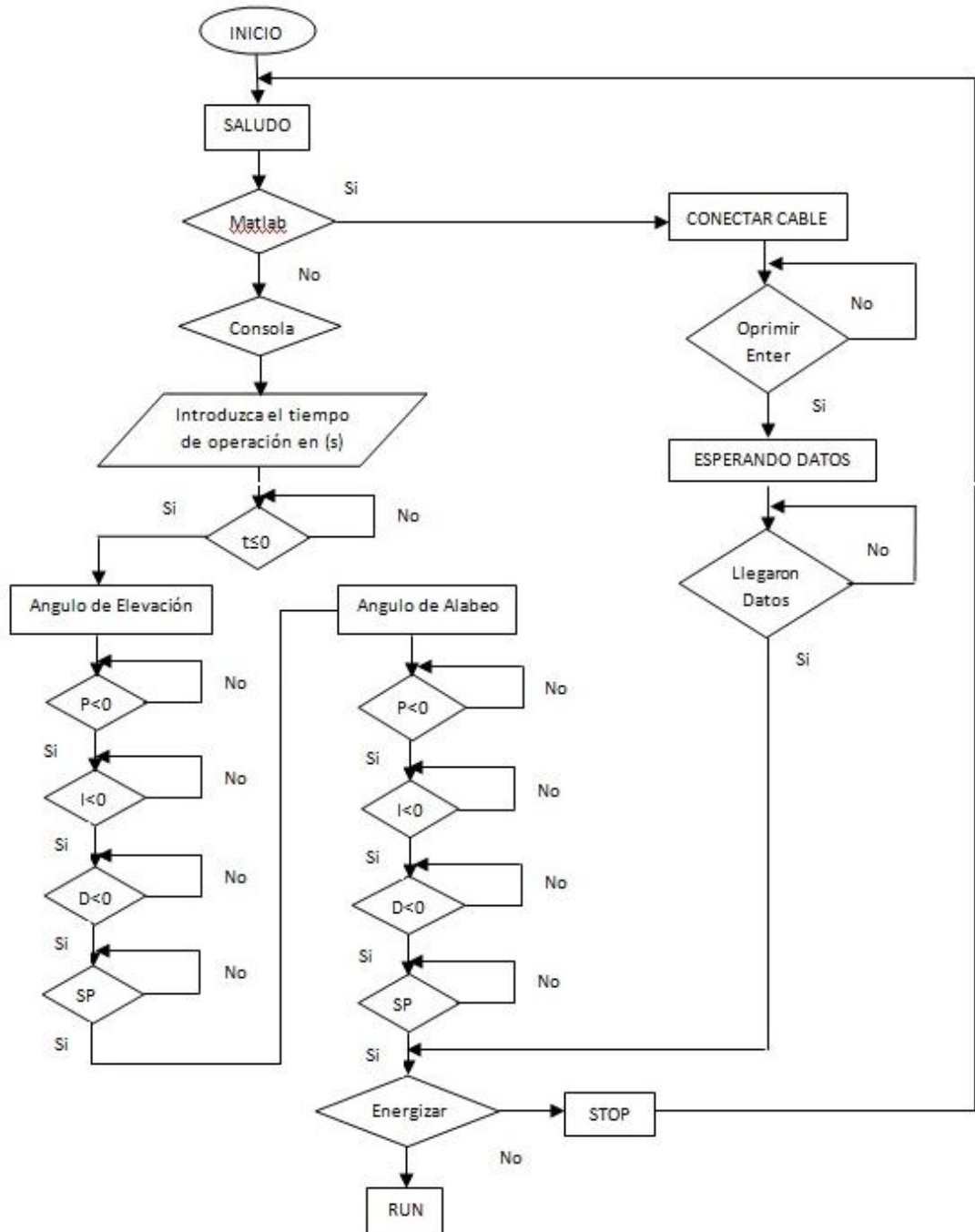


FIGURA 78. Diagrama de flujo de la programación del microcontrolador

## 10.4 COMPONENTES DEL SISTEMA EMBEBIDO

La tarjeta de control contiene los componentes electrónicos necesarios para su adecuado funcionamiento, para facilitar la descripción de esta, hicimos una agrupación del sistema por bloques funcionales como aparece a continuación en la figura 79.

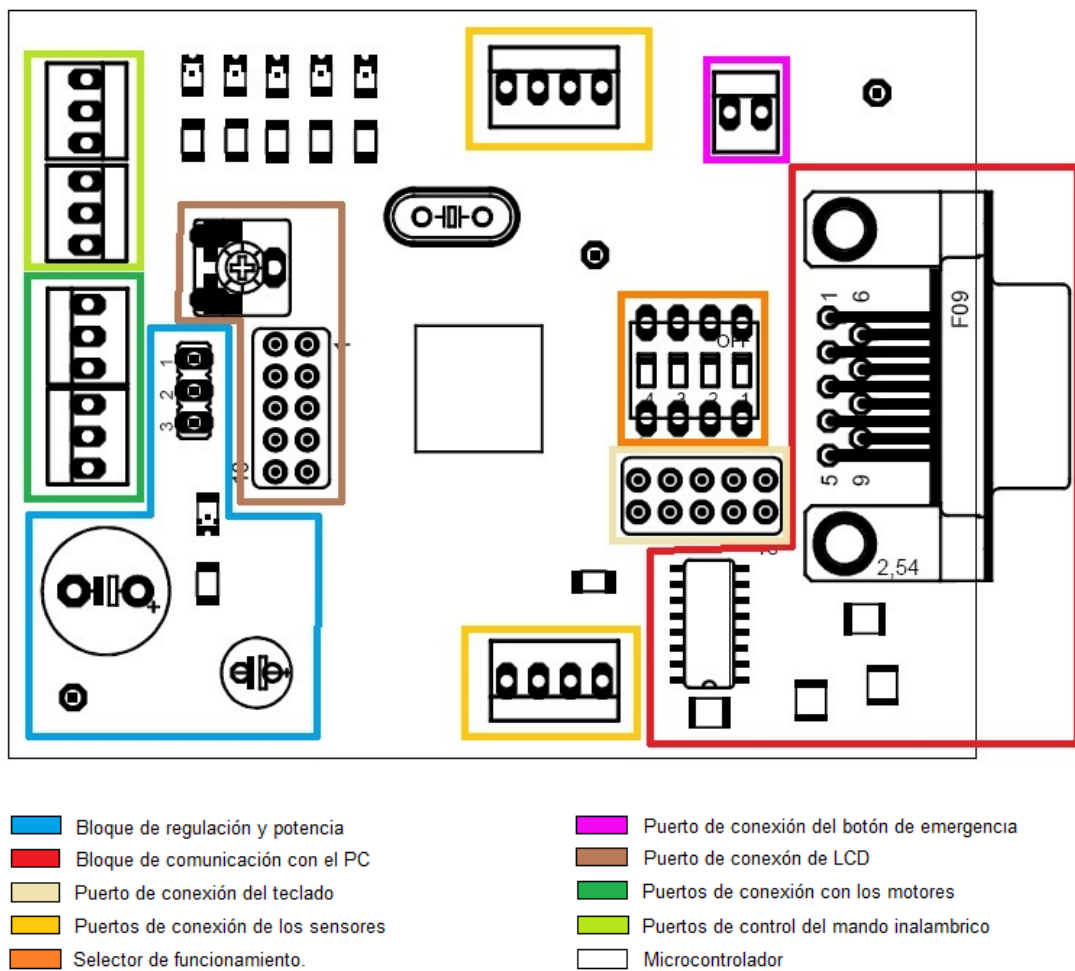


FIGURA 79. Bloques principales de la tarjeta

Al momento de diseñar un Sistema Embebido se pueden dividir los recursos de hardware y software de la siguiente manera:

#### **10.4.1 Bloque de regulación y potencia**

Su función principal radica en adecuar los voltajes requeridos para el funcionamiento del microcontrolador y sus periféricos, también permite seleccionar entre dos fuentes de alimentación, cada una de ellas dada por el sistema BEC (Battery Eliminator Circuit) proveniente de cada controlador de velocidad de los motores.

#### **10.4.2 Bloque de comunicación con el pc**

Esta parte del circuito adecúa los voltajes de la señal de entrada/salida del microcontrolador (3.3 voltios) a los voltajes que son requeridos por el computador (+/- 15 voltios) para el correcto envío de datos.

#### **10.4.3 Puerto de conexión de teclado**

En este puerto se conecta el teclado numérico para la interfaz on-board del controlador PID.

#### **10.4.4 Puerto de conexión de los sensores**

Estos puertos permiten proporcionar la alimentación de los sensores y adquirir adecuadamente las señales provenientes de estos, las señales de los pines del conector corresponden a 5 voltios que los entrega el bloque de potencia, y a las señales I-Q que entrega cada encoder.

#### **10.4.5 Puerto de conexión botón de emergencia**

La función principal de este botón es detener instantáneamente el proceso de control, para que dado el caso que el sistema se comporte de manera inestable se pueda detener para evitar accidentes del(os) operario(s) o del mismo dispositivo electromecánico. Como función secundaria se utiliza para activar el sistema cuando se opera en modo on-board.

#### **10.4.6 Puerto de conexión LCD**

Este conector da las señales requeridas por la pantalla de cristal liquido (LCD) para desplegar los mensajes de emergencia y comandos cuando está en modo on-board, junto a este conector se encuentra un potenciómetro que permite ajustar el contraste de la pantalla para visualizar de manera mas comoda los datos que aparecen en esta, cabe decir que este ajuste se realiza una sola vez y no requiere ajustes posteriores.

#### **10.4.7 Puertos de conexión de los motores**

Estos puertos son de doble función, la primera de ellas consiste en recibir los voltajes requeridos para la alimentación de toda la tarjeta, puesto que los controladores de velocidad son los que permiten tomar parte de la energía principal correspondiente a 12 voltios proporcionados por la fuente principal y reducirla a 5 voltios, esto es gracias a que los controladores poseen el sistema BEC (Battery Eliminator Circuit), de esta manera se reduce el tamaño la etapa de polarización y se minimizan las perdidas por disipación de calor en las etapas reguladoras intermedias. Como segunda función permiten enviar la señal de control a cada motor para controlar su velocidad por medio de modulación de ancho de pulso o PWM.

#### **10.4.8 Puertos de control de mando inalámbrico**

Este puerto permite la conexión de un radiocontrol de aeromodelismo de dos canales para controlar en tiempo real los set points de cada eje de control (pitch y roll), estos comandos pueden o no pasar a través del control PID, de manera que se pueda utilizar tanto en lazo abierto como en lazo cerrado, si bien este puerto está físicamente cableado, no se encuentra activo por software, puesto que se puede tomar como un puerto de expansión adicional para futuras investigaciones en el campo de controles MIMO.

#### **10.4.9 Microcontrolador**

Es el sistema de procesamiento principal, ya que este dispositivo es el encargado de todas las funciones de control de Helibot, desde el manejo de mensajes de la LCD, hasta el trazado de gráficas en el computador por medio de la interfaz serial.

#### **10.4.10 Selector de funcionamiento**

Este conjunto de interruptores permiten acceder a diferentes partes del programa para facilitar su depuración al momento de programarlo, también es utilizado para hacer las calibraciones previas de los motores cuando estos no son usados en largos períodos de tiempo (mayores de dos años), cuando se hace un cambio de speed control o conjunto controlador-motor. Estos selectores no deben ser manipulados por el usuario, puesto que su uso indebido puede alterar los parámetros pre programados en helibot, así como también puede provocar accidentes al manipularlos de manera inadecuada, por defecto deben estar en la posición OFF antes de energizar todo el sistema, por estas mismas restricciones estos selectores están dentro de la consola principal lejos de posibles manipulaciones.

## 10.5 CONEXIONES A LA TARJETA

Una vez hecho el diseño y construcción del sistema embebido y luego de seleccionar los componentes principales del Helibot como: sensores, motores, teclado, pantalla lcd, procedemos a conectar estos elementos del hardware a la tarjeta de control, tal y como se muestra en la figura 80.

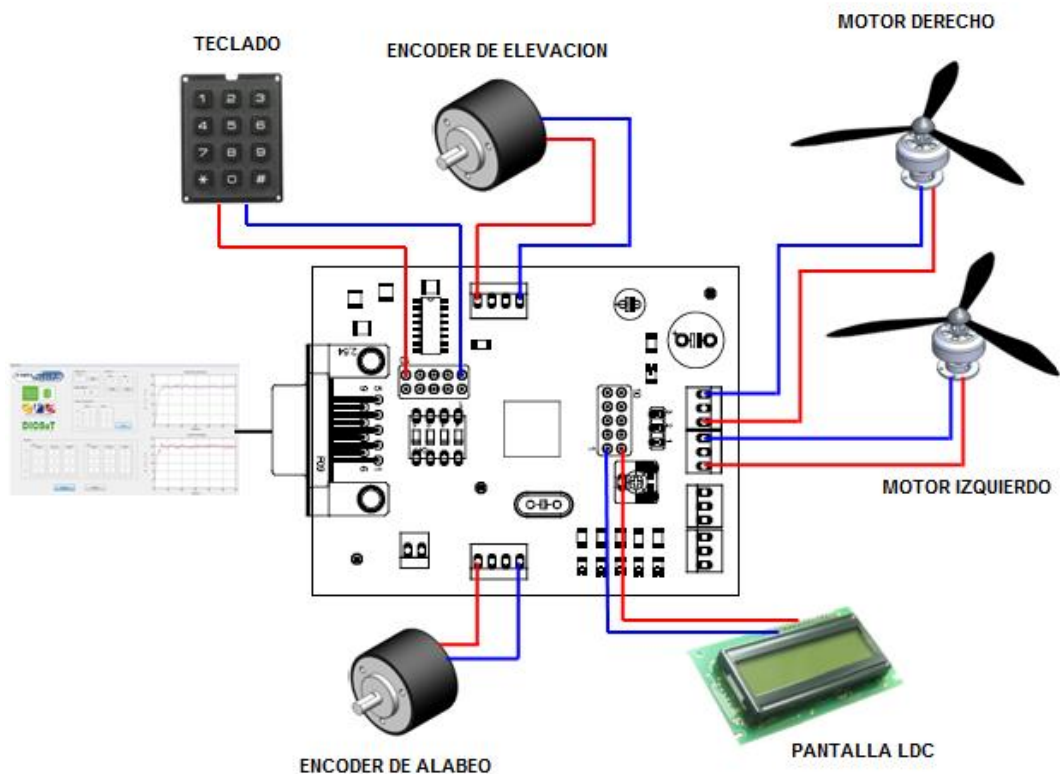


FIGURA 80. Conexiones del Hardware y Software a la tarjeta

## 10.6 DIALOGO SISTEMA – OPERADOR

Los elementos de dialogo entre maquina - operador con que cuenta el banco pueden dividirse en elementos de hardware y software. Los elementos de

hardware incluyen el panel de control, el cual no estaba establecido en los objetivos, pero en el transcurso de la elaboración del trabajo de grado nos pareció que ayudaría a la manipulación manual del banco sin hacer uso de una computadora, en este panel podemos por medio de un teclado numérico ingresar las constantes del controlador PID y los puntos de equilibrio del sistema y observar directamente los cambios en la plataforma del Helibot 3DOF, además se incluyó un botón de paro de emergencia del banco para que dado el caso que el sistema se comporte de manera inestable se pueda detener para evitar accidentes del(os) operario(s) o del mismo dispositivo electromecánico. Los elementos de software utilizados para la visualización es una interface en matlab, diseñada por los autores del proyecto exclusivamente para manipular el sistema y obtener diferentes graficas para su posterior estudio.

#### **10.6.1 Panel de control**

La caja de control permite el montaje de los elementos eléctricos y electrónicos de la etapa de control del banco, así como del panel de control como tal. Desarrollada en acrílico transparente calibre 5 mm, contiene en su interior unas guías también en acrílico para montar la tarjeta de control. La cara superior de la caja tiene un ángulo de inclinación de aproximadamente  $25^\circ$ , sobre la cual se coloca el panel de control. Esta inclinación del banco permite la fácil manipulación del banco por una persona de pie o sentada. Tal y como se muestra en la figura 81.



FIGURA 81. Caja de control

### 10.6.2 Interface en MATLAB

Se desarrolló una interface en MATLAB R2007b, para que el usuario/estudiante pudiera interactuar con la plataforma dinámica del helibot, además para presentar algunas graficas que nos muestran el comportamiento del sistema frente a diferentes valores del controlador PID. En el siguiente capítulo se presenta las utilidades y prestaciones de esta interfaz gráfica.

En el anexo H, se muestra el código fuente de esta interfaz en MATLAB 2007b.

## **11 INTERFACE EN MATLAB**

### **11.1 GENERALIDADES**

MATLAB es un entorno de computación y desarrollo de aplicaciones totalmente integrado orientado para llevar a cabo proyectos en donde se encuentren implicados elevados cálculos matemáticos y la visualización gráfica de los mismos. MATLAB integra análisis numérico, cálculo matricial, proceso de señal y visualización gráfica.

MATLAB dispone también en la actualidad de un amplio abanico de programas de apoyo especializados, denominados Toolbox, destacando entre ellos el 'toolbox' de proceso de imágenes, señal, control robusto, estadística, análisis financiero, matemáticas simbólicas, redes neurales, lógica difusa, identificación de sistemas, simulación de sistemas dinámicos, etc.

### **PRESTACIONES DE MATLAB**

Entre sus prestaciones básicas de MATLAB se hallan:

- La manipulación de matrices.
- La representación de datos y funciones, la implementación de algoritmos.
- La creación de interfaces de usuario (GUI).
- Comunicación con programas en otros lenguajes y con otros dispositivos hardware.

## 11.2 CONEXIÓN DE LA INTERFAZ EN EL ENTORNO MATLAB

Mostraremos los pasos para abrir la interfaz desde MATLAB, además describimos el procedimiento para utilizar correctamente los comandos principales para poder enlazarla con el sistema para habilitar los puertos para recibir y enviar datos hacia la tarjeta de control.

Primero desde Matlab abrimos la carpeta donde se encuentra la interfaz grafica del helibot 3DOF de la siguiente forma:

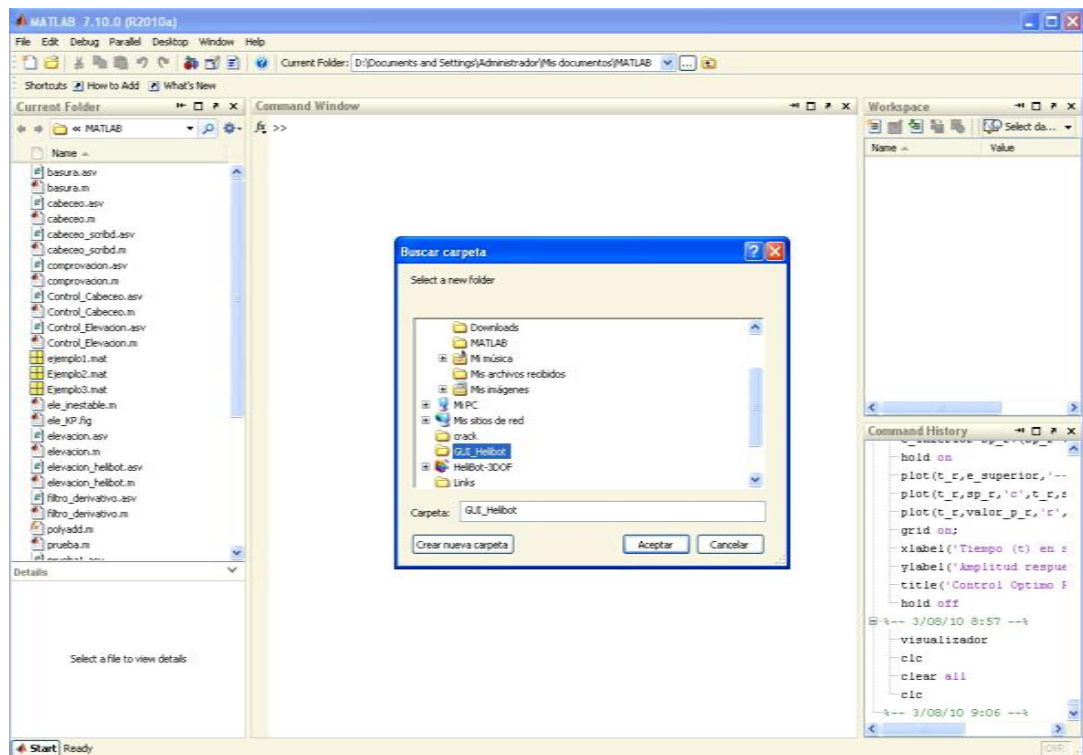
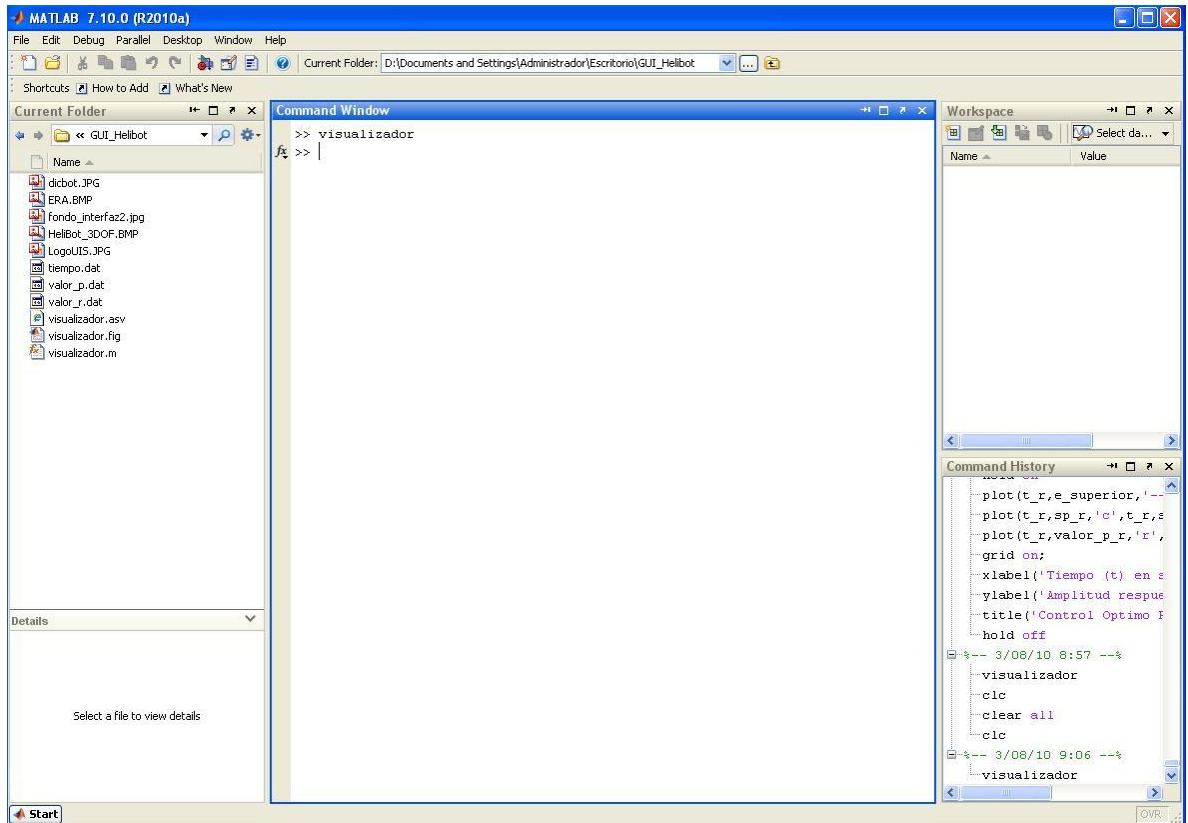


FIGURA 82. Establecer la carpeta de la interfaz

Una vez que abramos la carpeta GUI\_Helibot, escribimos en el PROM de MATLAB la palabra visualizador y damos enter.



**FIGURA 83.** Configuración de la conexión a la interfaz

Una vez ejecutamos `//visualizador//` en el PROM de MATLAB, el programa genera automáticamente un pantallazo inicial de la interfaz grafica diseñada en el GUIDE (editor de interfaces de usuario - GUI), donde podemos ver las variables del proceso así como un campo exclusivo para las graficas de la respuesta de salida de la planta en los ejes de elevación y alabeo, tal y como se presenta en la figura 83.

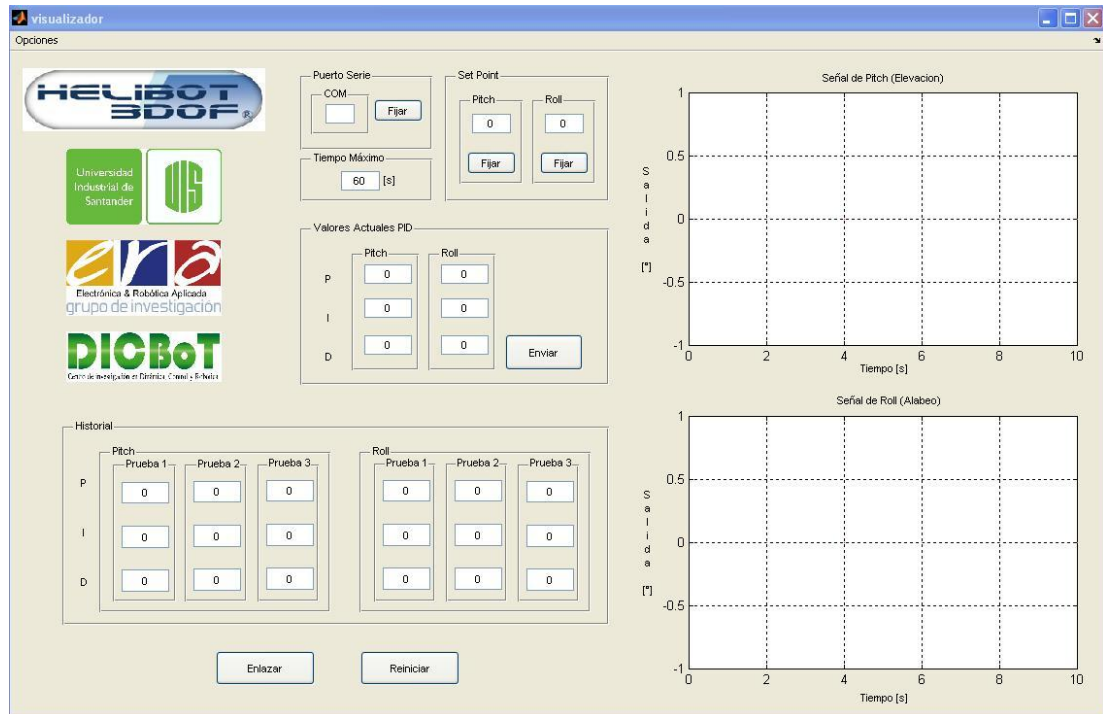


FIGURA 84. Pantallazo inicial de la interfaz

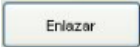

## Definición de las variables del proceso


Las variables utilizadas en la interfaz, representan o bien valores reales, tales como la posición inicial o set point en el ángulo de elevación y alabeo, el tiempo de funcionamiento de la planta y los valores dados al controlador P, I, D; además de los valores de conexión del puerto serial el cual se define inicialmente de acuerdo al puerto serial que se está utilizando.

La interfaz está dividida principalmente en los siguientes bloques:

- Puerto serial
- Set point
- Valores del controlador PID

- Historial de pruebas
- Gráficas de respuesta de la salida del ángulo de elevación y alabeo.

Para  MATLAB con la tarjeta de control, tenemos que definir primero el puerto serial de comunicación que se está utilizando, una vez enlazado el sistema, definimos los valores de set point, o punto de estabilización de la planta en los ejes de elevación y alabeo, posteriormente se define los valores de P, I, D que va a tomar el controlador seguidamente enviamos los datos, pulsando el botón .

Una vez terminada la prueba la interfaz guarda hasta 3 historiales de los valores PID, tanto para el ángulo pitch como para ángulo roll, para iniciar otra prueba pulsamos el botón .

Ahora probamos el funcionamiento de la interfaz frente a las siguientes condiciones, antes de esto debemos fijar el puerto serial COM y después enlazar, para esta práctica el tiempo de funcionamiento del sistema será de 60 segundos.

En la figura 85 vemos un pantallazo de la interfaz, donde se puede apreciar la respuesta de salida de la planta frente a las siguientes condiciones.

	Elevación Set point=19°	Alabeo Set point= 20°
Proporcional	9	1
Integral	30	8
Derivativo	2.5	0.35

**TABLA 8.** Parámetros para prueba de la interfaz

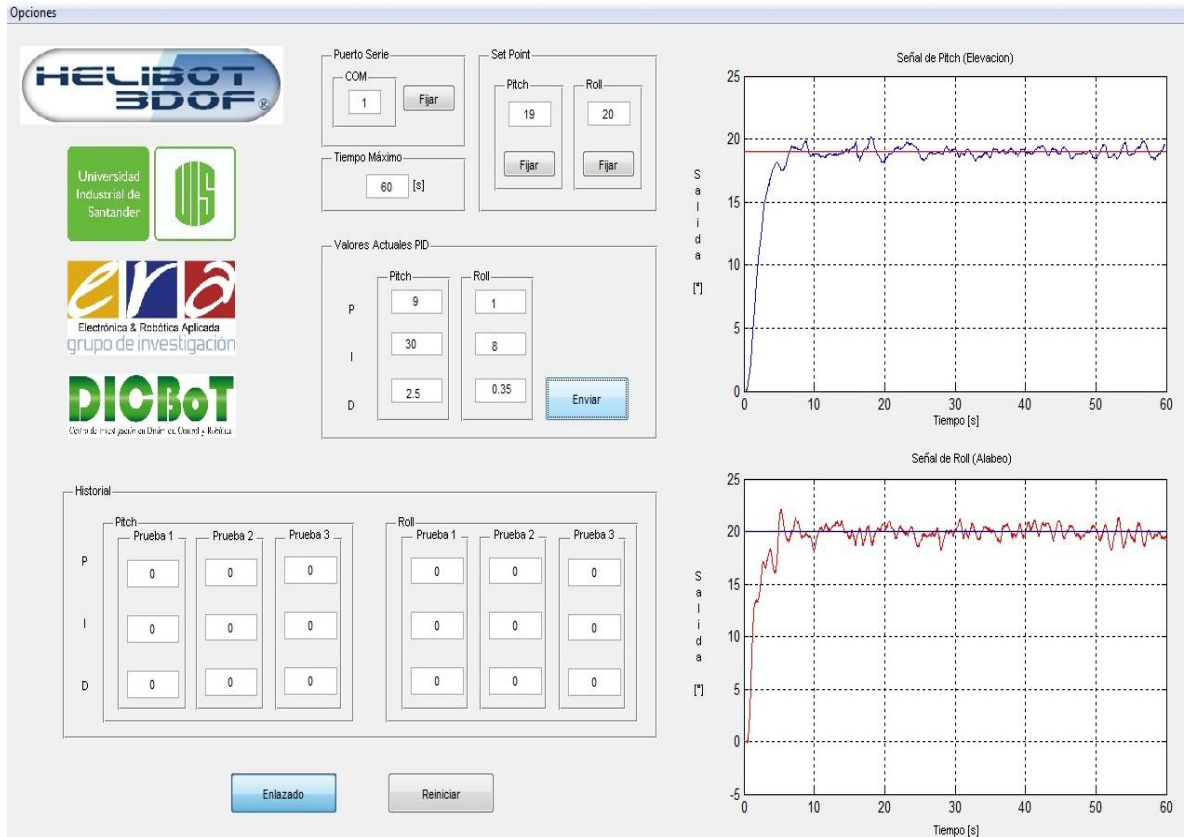


FIGURA 85. Pantallazo de resultados en la interfaz

## 12 PRUEBAS HELIBOT 3DOF

Para la realización de las pruebas se convino utilizar los 4 miembros clásicos de la familia de controladores PID, Control P, PI, PD, y PID. Se analizó la respuesta del sistema bajo la intervención de diferentes combinaciones y magnitudes de las acciones de control.

Como se dijo anteriormente el sistema está compuesto por dos subsistemas SISO (Single Into Single Out), es decir tenemos dos lazos de control independiente, uno para el ángulo de elevación y uno para el ángulo de alabeo, el ángulo de viaje no tiene controlador independiente y su respuesta se puede explicar por la combinación en la respuesta de los dos anteriores.

El fin último de estas pruebas es obtener un controlador que cumpla con los requerimientos de diseño descritos en la tabla 7, mientras en el proceso se confirma el efecto de la aplicación de las acciones de control P, I, y D estudiadas en el capítulo de diseño del controlador.

<b>Requerimientos de Diseño</b>		
<b>Característica</b>	<b>Angulo de Elevación</b>	<b>Angulo de Alabeo</b>
Sobreimpulso	[0-20]%	[0-10]%
Subimpulso	[0-12]%	[0-7]%
Tiempo de subida	[0-4] seg.	[0-5] seg.
Tiempo de establecimiento	[4-15] seg.	[5-8] seg.
Error estado estacionario	[0-2]%	[0-2]%

TABLA 9. Síntesis de los requerimientos de diseño de los 2 lazos de control

## 12.1 CONTROL PROPORCIONAL

En esta etapa se estudió el efecto de la acción proporcional en el control del Helibot 3DOF tanto para el ángulo de alabeo como para el ángulo de elevación, es importante resaltar que todas las pruebas siguientes se hicieron con un tiempo de muestreo de  $10e-3$  seg, lo que sugiere 100 muestras por segundo, además con el sistema en condición inicial de inestabilidad.

### 12.1.1 Ángulo de elevación con controlador P

PRUEBA 1	Características de la respuesta
Proporcional ( $K_p$ ) = 5	El sistema es inestable en el tiempo, presenta máximos cada vez más altos y mínimos cada vez más bajos, en esta prueba solo necesitamos de 10 segundos para darnos cuenta de su comportamiento inestable.
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 10 seg	

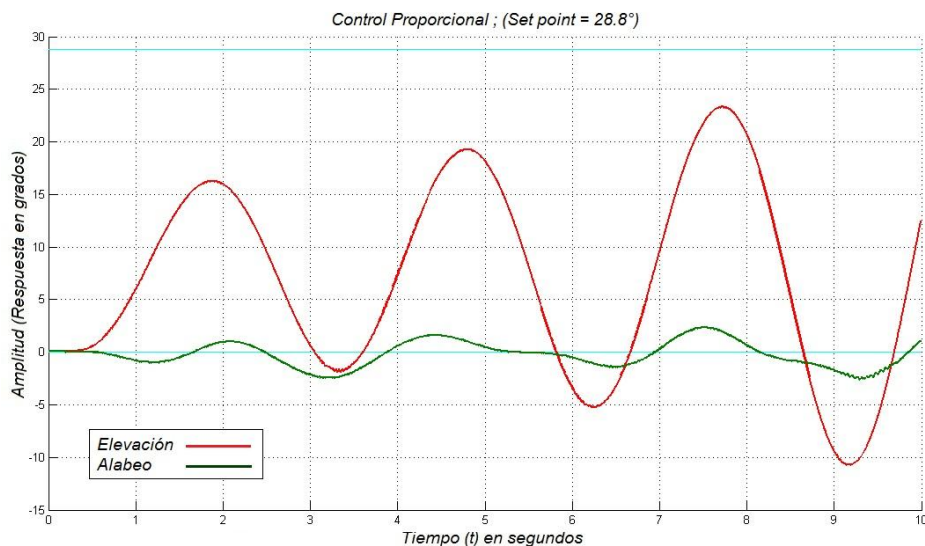


FIGURA 86. Respuesta real del ángulo de Elevación con control proporcional de prueba 1.

PRUEBA 2	Características de la respuesta
Proporcional ( $K_p$ ) = 11	El sistema es inestable en el tiempo, presenta máximos cada vez más altos y mínimos cada vez más bajos, en esta prueba solo necesitamos de 8 segundos para darnos cuenta de su comportamiento inestable.
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 8 seg	

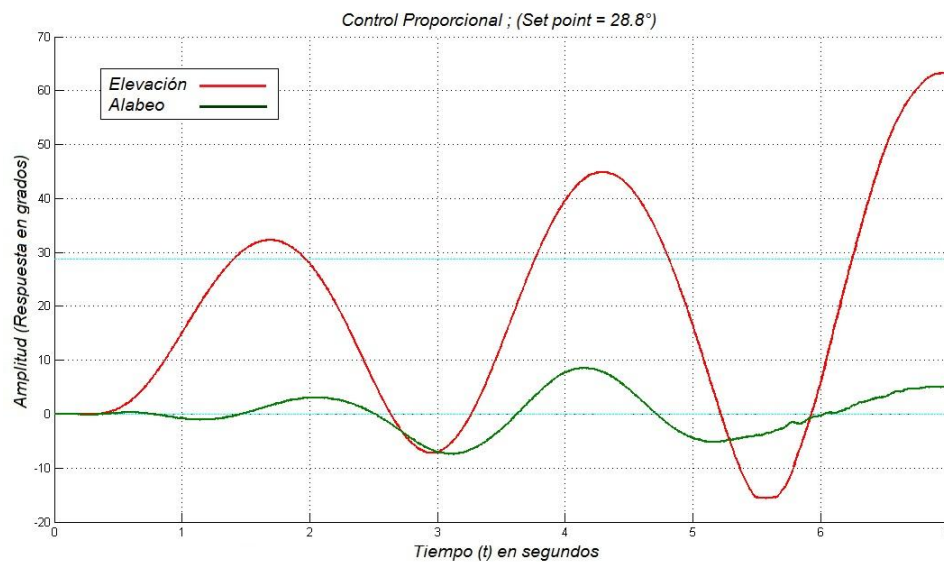


FIGURA 87. Respuesta real del ángulo de Elevación con control proporcional de prueba 2.

Con las pruebas anteriormente mostradas, y luego de realizar otras muchas no publicadas, se concluyó que el ángulo de elevación no se logra estabilizar con la aplicación de un control proporcional únicamente, ya que a mayores magnitudes de  $K_p$  el sistema presenta máximos muy altos que podrían comprometer la estructura de la planta pues demanda un voltaje mayor al de saturación lo que ocasiona la interrupción del proceso; y con menores magnitudes de  $K_p$  no se podrá reducir el error de estado estacionario y mucho menos lograr el 2% requerido. La respuesta del ángulo de alabeo es simplemente el resultado del comportamiento del ángulo de elevación ya que se encuentra en lazo abierto.

### 12.1.2 Angulo de alabeo con controlador P

Para sintonizar el controlador del ángulo de alabeo se probaron valores de  $K_p$  más bajos, ya que esta lazo de control aunque no está compensado por el contrapeso (como en el ángulo de elevación), sino son 2 motores que tienen que corregir el delta de PWM (ancho de pulso) lo que requiere un menor esfuerzo en control y por lo tanto una menor ganancia del controlador.

PRUEBA 3	Características de la respuesta
Proporcional ( $K_p$ ) = 1.5	El sistema es inestable en el tiempo, la acción de control proporcional es insuficiente para lograr estabilización, y el ángulo de alabeo llega a un peligroso máximo de $90^\circ$ .
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $45^\circ$	
Tiempo de Prueba = 60 seg	

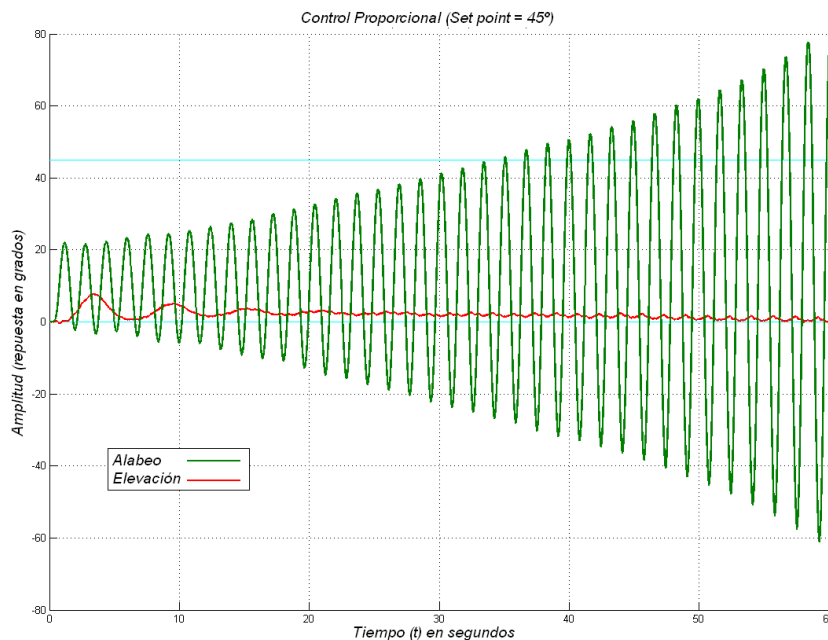


FIGURA 88. Respuesta real del ángulo de Alabeo con control proporcional de PRUEBA 3.

PRUEBA 4	Características de la respuesta
Proporcional ( $K_p$ ) = 1	El sistema se estabiliza en una amplitud de $7^\circ$ muy inferior al punto de equilibrio, en un tiempo de 16 seg. antes del cual presenta un comportamiento sobreamortiguado con un sobreimpulso de 50% y un subimpulso de 52%
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $45^\circ$	
Tiempo de Prueba = 60 seg	

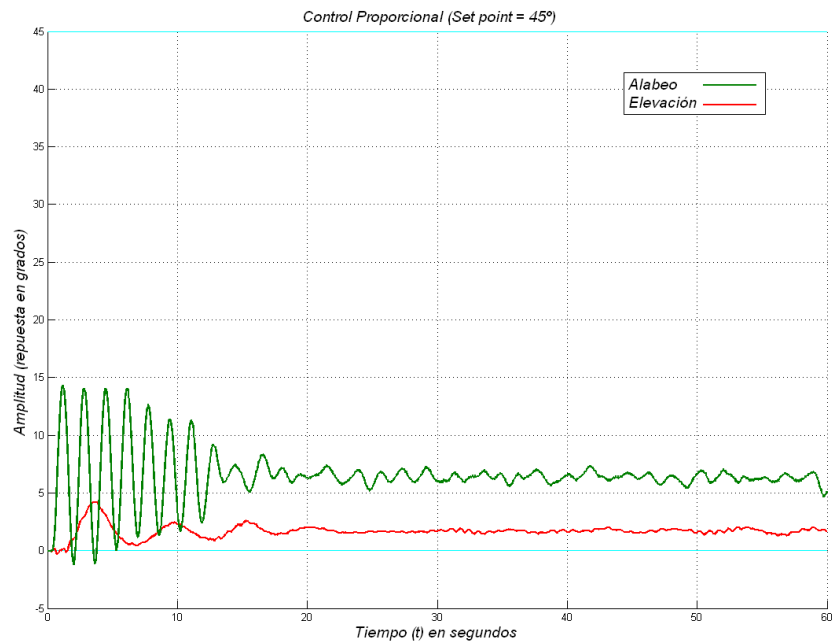


FIGURA 89. Respuesta real del ángulo de Alabeo con control proporcional de prueba 4.

La acción proporcional por sí sola no representa en este caso un método de control efectivo igual que en el control del ángulo de elevación a magnitudes grandes de  $K_p$  el sistema se hace inestable ( $> 1.3$ ), y con pequeñas magnitudes ( $< 1.3$ ) el sistema no logra corregir el error de estado estacionario.

La respuesta del ángulo de elevación es solo el resultado del comportamiento de la respuesta del ángulo de alabeo ya que se encuentra en lazo abierto.

## 12.2 CONTROL PROPORCIONAL – INTEGRAL

El modo de control Integral tiene como propósito disminuir y eliminar el error en estado estacionario, provocado por el modo proporcional. El control integral actúa cuando hay una desviación entre la variable y el punto de consigna, integrando esta desviación en el tiempo y sumándola a la acción proporcional.

### 12.2.1 Ángulo de elevación con controlador PI

PRUEBA 5	Características de la respuesta
Proporcional ( $K_p$ ) = 1	El sistema no logra estabilizarse en los 90 segundos de prueba pero de continuar su tendencia se podría afirmar que se lograra estabilizar, lo que sugiere intentar un controlador con una acción integral mayor.
Integral ( $K_i$ ) = 1	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 90 seg	

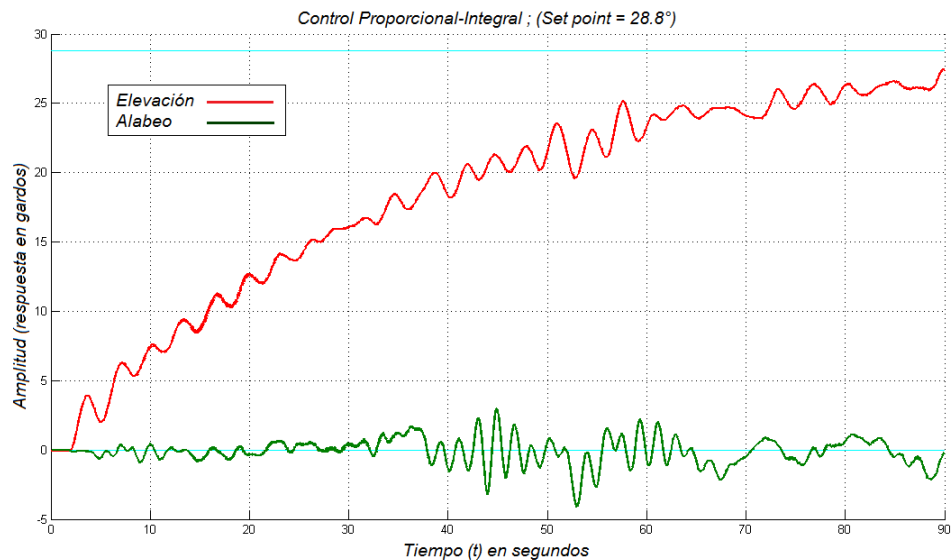


FIGURA 90. Respuesta real del ángulo de elevación con control proporcional-integral de prueba 5.

PRUEBA 6	Características de la respuesta
Proporcional ( $K_p$ ) = 1	Con este controlador el sistema presenta un comportamiento sobreamortiguado, aunque alcanza el punto de equilibrio no logra estabilizarse, y se vuelve inestable con el tiempo.
Integral ( $K_i$ ) = 5	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 90 seg	

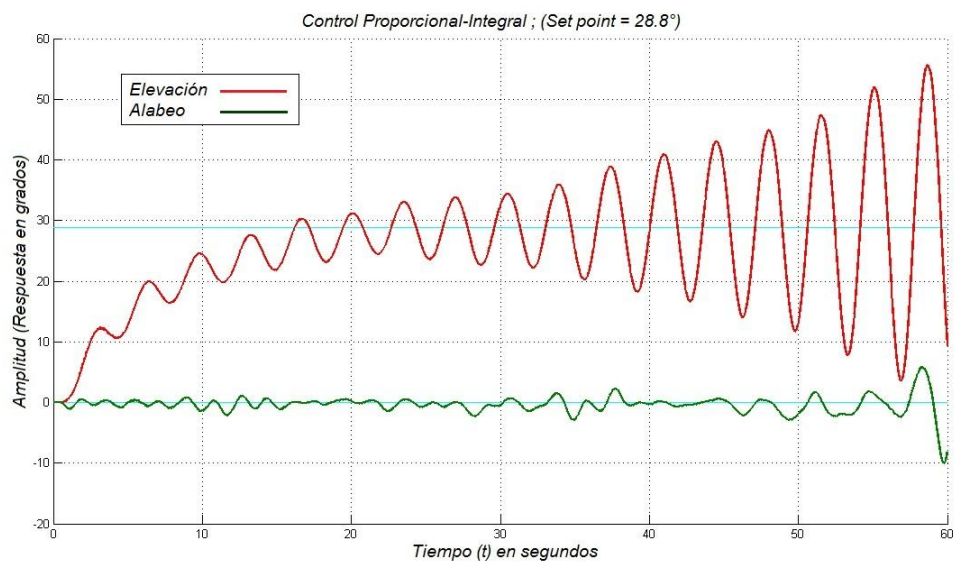


FIGURA 91. Respuesta real del ángulo de elevación con control proporcional-integral de prueba 6.

Se observó que con la intervención de la acción integral se logra eliminar el error en estado estacionario, el sistema alcanza el punto de equilibrio y lo hace en menor tiempo a medida que aumenta la magnitud de  $K_i$ , mientras que si aumentamos la magnitud de  $K_p$  la respuesta se hace mas oscilatoria y sobreamortiguada.

El control integral se utiliza para obviar el inconveniente del offset (desviación permanente de la variable con respecto al punto de consigna) de la banda proporcional, pero se obtiene una mayor variación del set point.

PRUEBA 7	Características de la respuesta
Proporcional ( $K_p$ ) = 0.5	Este controlador permite la estabilización del sistema en un tiempo de 60 segundos, con un error en estado estacionario de 4.5%, la planta no presenta sobreimpulso ni subimpulsos importantes y describe un movimiento amortiguado
Integral ( $K_i$ ) = 2	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 90 seg	

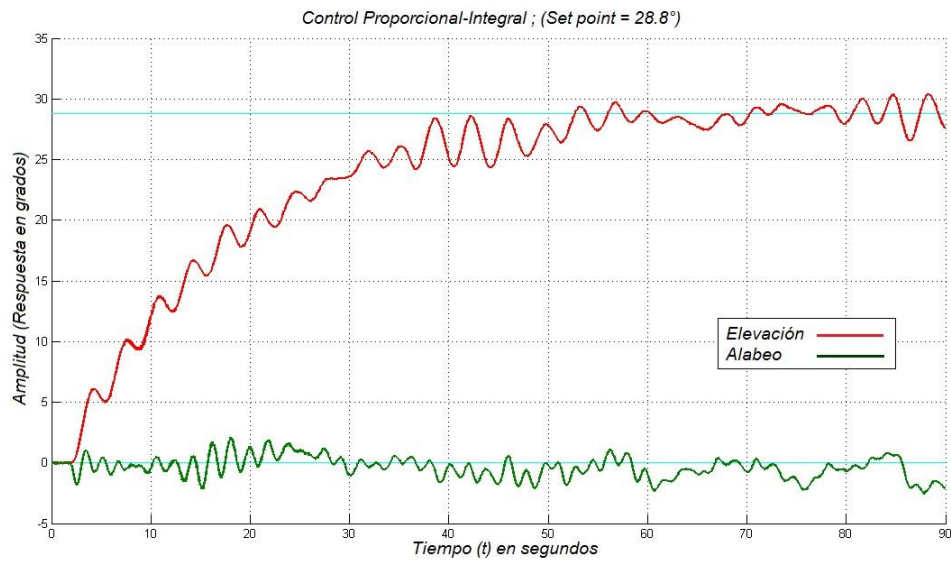


FIGURA 92. Respuesta real del ángulo de elevación con control proporcional-integral de prueba 7.

Con el ajuste adecuado de la ganancia proporcional ( $K_p$ ) e integral ( $K_i$ ) se obtuvo una respuesta que en la cual está corregida la desviación con respecto al set point y además se evitó el sobreamortiguamiento, el inconveniente de este controlador es que no responde bien ante las perturbaciones y además retarda el tiempo de respuesta fuera del rango de requerimientos.

## 12.2.2 Ángulo de alabeo con controlador PI

PRUEBA 8	Características de la respuesta
Proporcional ( $K_p$ ) = 0.5	Con la aplicación de la acción de control integral, en un control proporcional, logramos disminuir o eliminar el error en estado estacionario sin embargo el sistema demanda de un gran tiempo de estabilización (> 90 seg)
Integral ( $K_i$ ) = 0.5	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $30^\circ$	
Tiempo de Prueba = 90 seg	

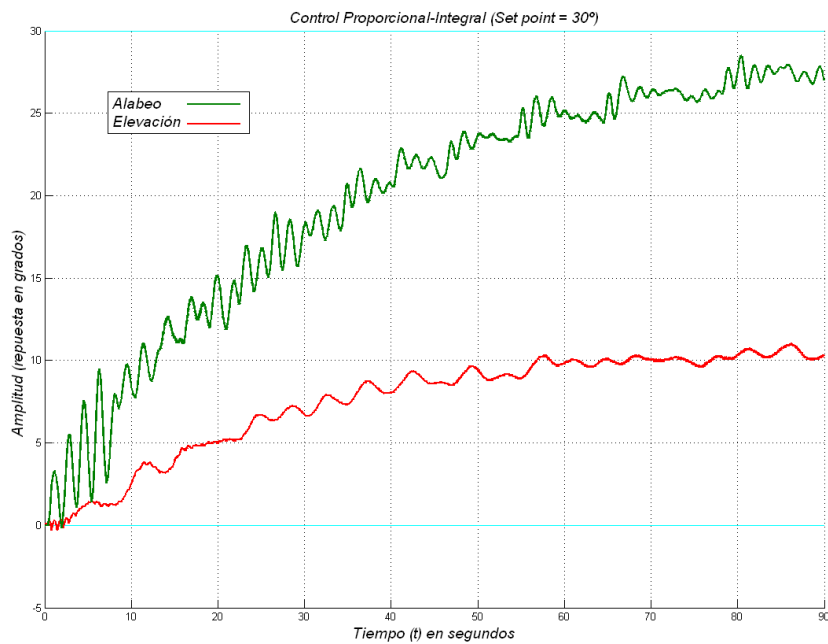


FIGURA 93. Respuesta real del ángulo de alabeo con control proporcional-integral de prueba 8.

PRUEBA 9	Características de la respuesta
Proporcional ( $K_p$ ) = 0.25	Con la aplicación de este controlador el sistema logra estabilizarse con un criterio del 13%, el cual no es muy conveniente en una aplicación industrial (criterio máximo de 5%)
Integral ( $K_i$ ) = 0.8	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = $30^\circ$	
Tiempo de Prueba = 90 seg	

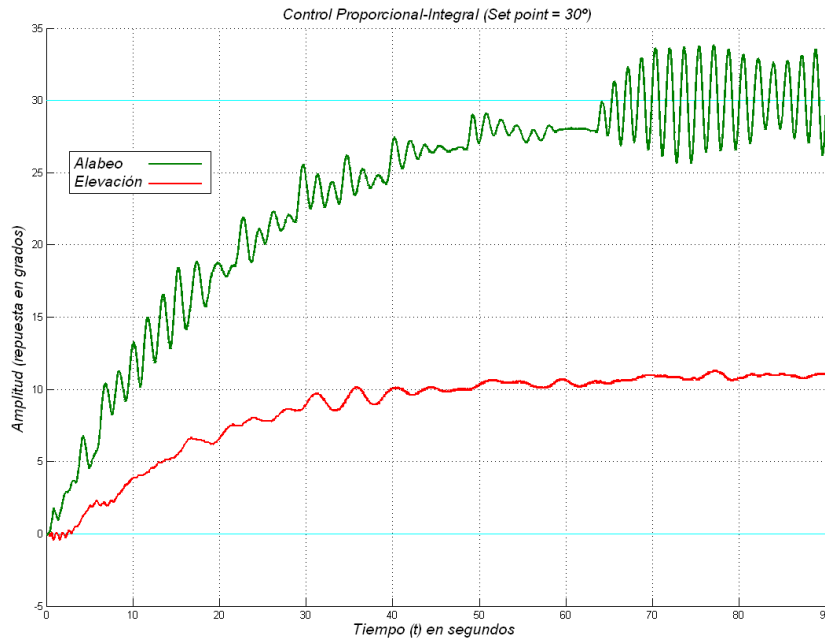


FIGURA 94. Respuesta real del ángulo de alabeo con control proporcional-integral de prueba 9.

De la prueba 9, vemos que al aumentar la constante integral ( $K_i$ ) el sistema reduce el error en estado estacionario, pero como también aumentamos la ganancia proporcional ( $K_p$ ) el sistema aumenta la inestabilidad, y no compensa con eficiencia una perturbación repentina, esta deficiencia es rectificada con la intervención de una acción derivativa ( $K_d$ ). En la prueba 10 veremos como el sistema tiene un comportamiento altamente inestable cuando se incrementa el  $K_p$  y disminuye el  $K_i$

PRUEBA 10	Características de la respuesta
Proporcional ( $K_p$ ) = 1.5	Este sistema es inestable en el tiempo
Integral ( $K_i$ ) = 0.25	
Derivativo ( $K_d$ ) = 0	
Punto de equilibrio = 30°	
Tiempo de Prueba = 90 seg	

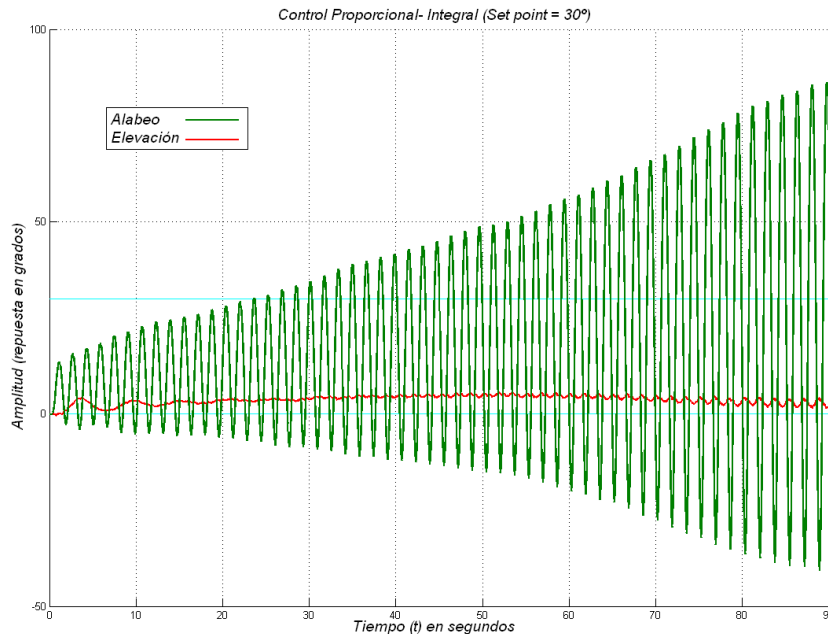


FIGURA 95. Respuesta real del ángulo de alabeo con control proporcional-integral de prueba 10.

### 12.3 CONTROL PROPORCIONAL-DERIVATIVO

La acción derivativa se manifiesta cuando hay un cambio en el valor absoluto del error; (si el error es constante, solamente actúan los modos proporcional e integral). La función de la acción derivativa ( $K_d$ ) es mantener el error al mínimo corrigiéndolo proporcionalmente con la misma velocidad con que se produce; de esta manera evita que el error se incremente, es decir esta acción corrige el error ante perturbaciones repentinas o inducidas y facilita el amortiguamiento en la respuesta del sistema manteniendo las variaciones del error casi constante (dependiendo de la magnitud del  $K_d$ ) luego de la estabilización.

La magnitud óptima la acción derivativa será entonces la que retorne la variable controlada (ángulo de elevación y alabeo) al punto de equilibrio con las mínimas oscilaciones.

### 12.3.1 Angulo de elevación con controlador PD

PRUEBA 11	Características de la respuesta
Proporcional ( $K_p$ ) = 5	El sistema describe un comportamiento sobreamortiguado con una respuesta brusca a cualquier pequeña perturbación, la línea media de estabilización se encuentra en un ángulo de $7.5^\circ$ un error de estado estacionario del 74%.
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 6	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 60 seg	

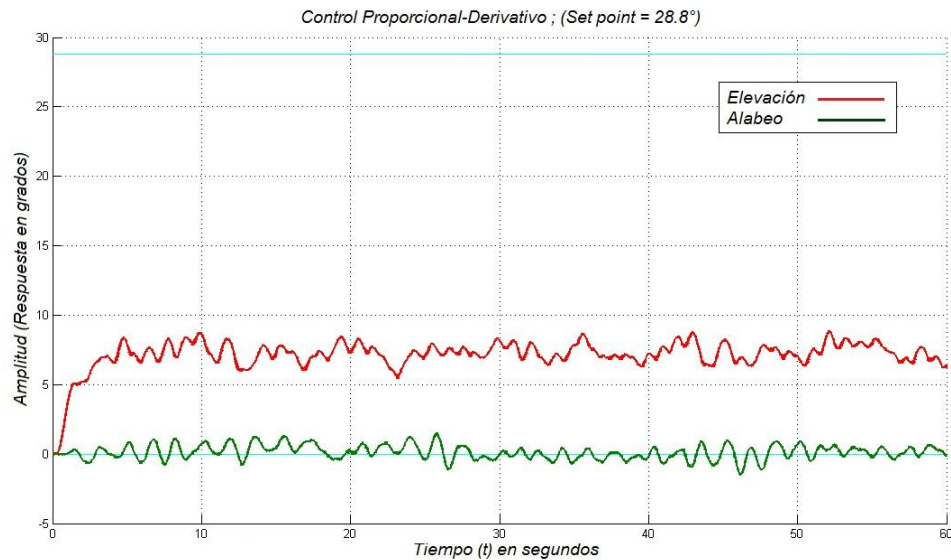


FIGURA 96. Respuesta real del ángulo de elevación con control proporcional-derivativo de prueba 11.

La acción derivativa elimina las oscilaciones que presentaba este controlador sin la intervención de la acción derivativa (PRUEBA 1). El sistema tiene una respuesta brusca en la corrección del error, es decir la velocidad de corrección es demasiada producto de la magnitud de  $K_d$ , en la siguiente prueba se experimentara el mismo controlador pero con un valor de  $K_d$  menor.

<b>PRUEBA 12</b>	<b>Características de la respuesta</b>
Proporcional ( $K_p$ ) = 5	El sistema describe un comportamiento amortiguado, la línea media de estabilización se encuentra en un ángulo de $7.5^\circ$ un error de estado estacionario del 74%. Un sobreimpulso del 33%.
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 2	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 60 seg	

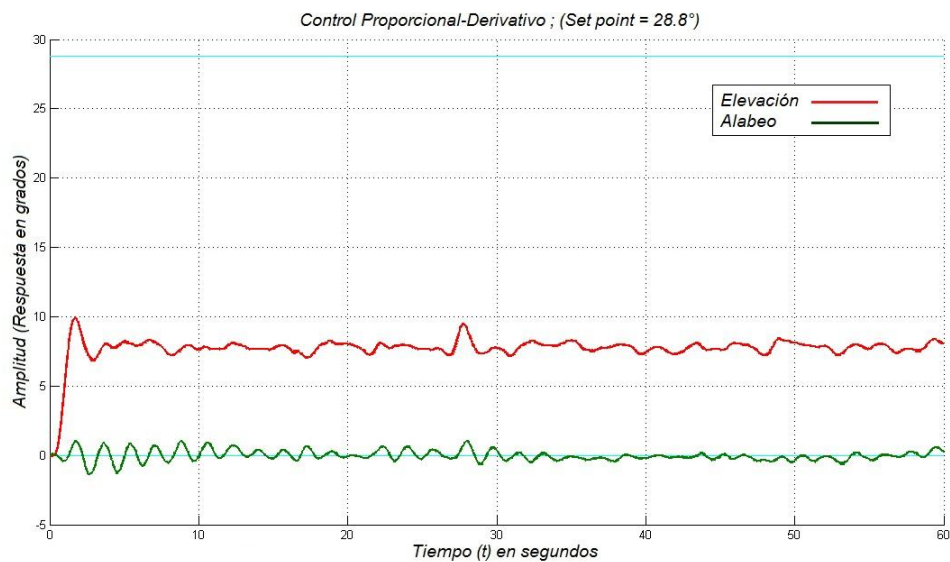


FIGURA 97. Respuesta real del ángulo de elevación con control proporcional-derivativo de prueba 12.

Se observó que al disminuir la magnitud de  $K_d$  las oscilaciones disminuyen sin embargo el error de estado estacionario se mantuvo constante en 74%,

<b>PRUEBA 13</b>	<b>Características de la respuesta</b>
Proporcional ( $K_p$ ) = 8	El sistema describe un comportamiento amortiguado, aunque responde de forma brusca a las perturbaciones, la línea media de estabilización se encuentra en un ángulo de $10^\circ$ , un error de estado estacionario del 66%.
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 4	
Punto de equilibrio = $28.8^\circ$	
Tiempo de Prueba = 60 seg	

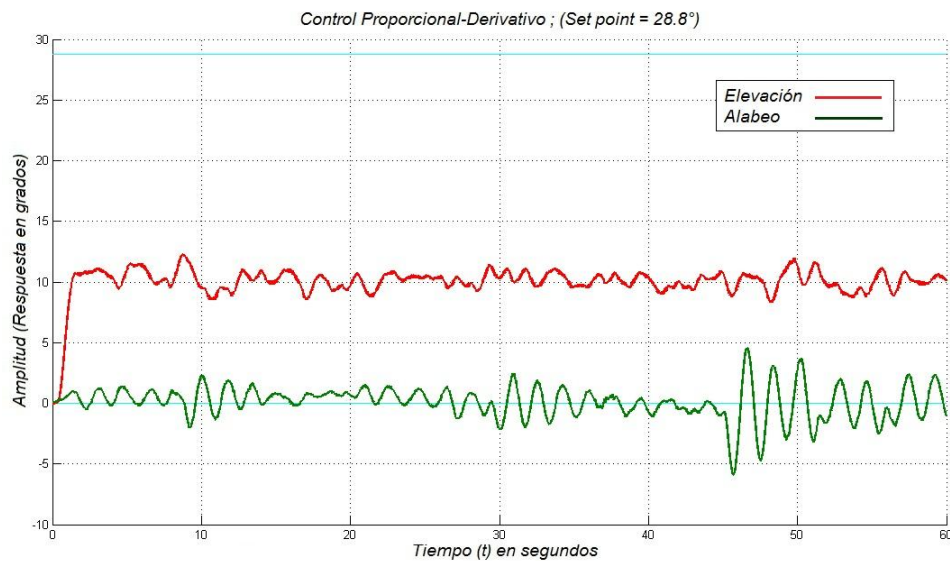


FIGURA 98. Respuesta real del ángulo de elevación con control proporcional-derivativo de prueba 13.

Con el aumento de la ganancia proporcional disminuyó el error de estado estacionario de 74% a 63% sin embargo por más que se aumente el  $K_p$  el controlador proporcional-derivativo nunca podrá eliminar el *offset*.

### 12.3.2 Ángulo de alabeo con controlador PD

PRUEBA 14	Características de la respuesta
Proporcional ( $K_p$ ) = 2.5	El sistema tiene una estabilización en 6 segundos en un ángulo de 12.5°, lo que indica un error en estado estacionario del 73% un tiempo de subida de 1 segundo, un sobreimpulso del 40 % y un subimpulso del 12%
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 0.3	
Punto de equilibrio = 45°	
Tiempo de Prueba = 90 seg	

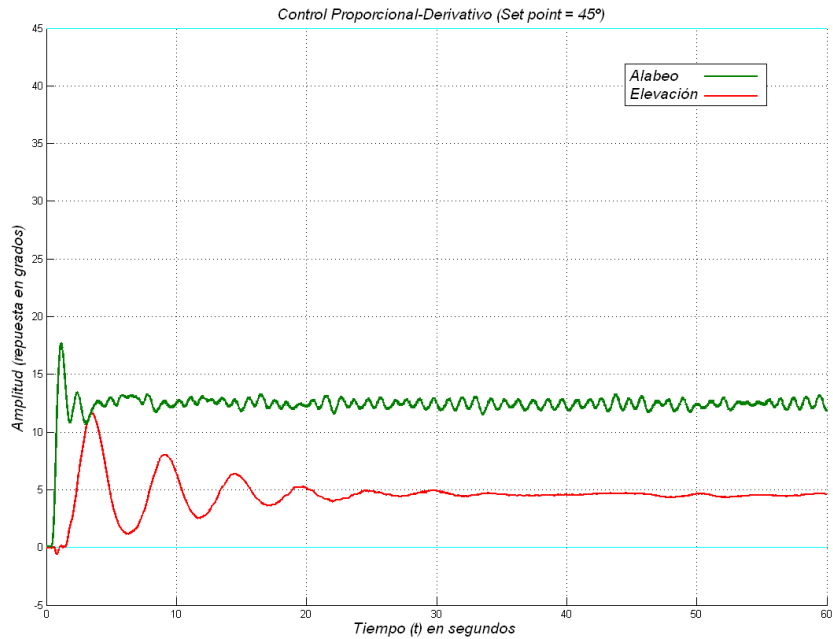


FIGURA 99. Respuesta real del ángulo de alabeo con control proporcional-derivativo de prueba 14.

Para empezar a ajustar se eligió un valor pequeño de  $K_d$  y se ajustó el  $K_p$  a un valor en el cual ya se conocía la respuesta altamente inestable del sistema y se observó que la acción derivativa tiene una gran influencia en la estabilización, logrando la disminución de las oscilaciones sin eliminar el *offset* (error de estado estacionario).

PRUEBA 15	Características de la respuesta
Proporcional ( $K_p$ ) = 4	El sistema tiene una estabilización en 6 segundos en un ángulo de 20°, lo que indica un error en estado estacionario del 55.5%, un tiempo de subida de 1 segundo, un sobreimpulso del 60 % y un subimpulso del 22.5%
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 0.3	
Punto de equilibrio = 45°	
Tiempo de Prueba = 90 seg	

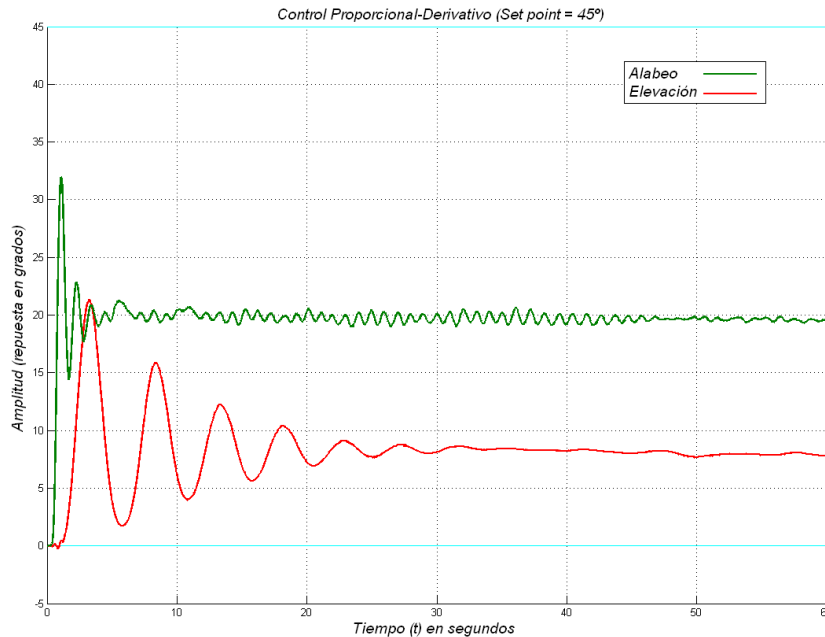


FIGURA 100. Respuesta real del ángulo de alabeo con control proporcional-derivativo de prueba 15.

Al aumentar la ganancia proporcional manteniendo constante la ganancia derivativa la repuesta experimenta un sobreimpulso bastante grande manteniendo el error en un rango estrecho después de la estabilización en un punto muy por debajo del punto de consigna.

En la siguiente prueba se corregirá el sobreimpulso con un aumento de la ganancia derivativa ( $K_d$ ) sin que esto rectifique el offset manteniéndolo constante en un 60%.

PRUEBA 16	Características de la respuesta
Proporcional ( $K_p$ ) = 4	El sistema tiene una estabilización en 9 segundos en un ángulo de 20°, lo que indica un error en estado estacionario del 60%, un tiempo de subida de 4 segundos y describe un movimiento amortiguado
Integral ( $K_i$ ) = 0	
Derivativo ( $K_d$ ) = 3	
Punto de equilibrio = 45°	
Tiempo de Prueba = 90 seg	

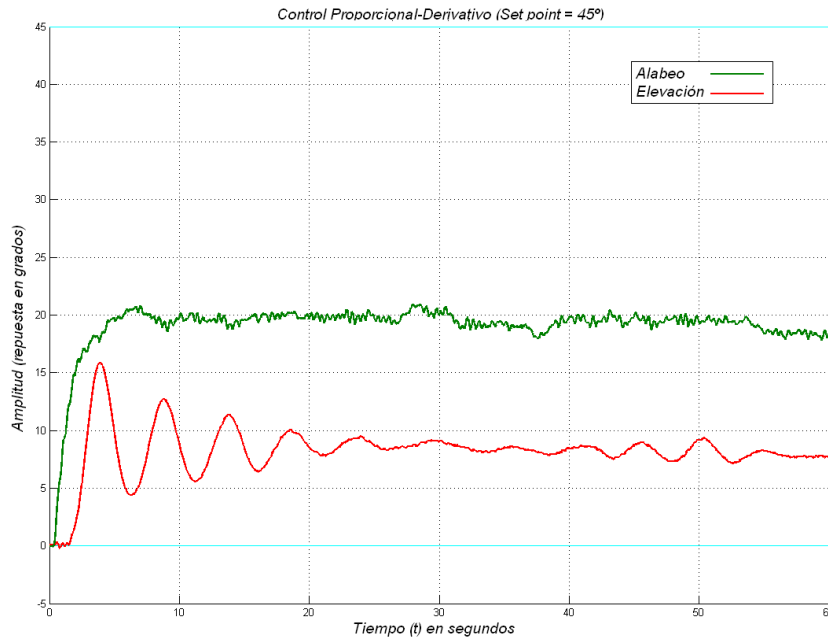


FIGURA 101. Respuesta real del ángulo de alabeo con control proporcional-derivativo de prueba 16.

## 12.4 CONTROL PROPORCIONAL – INTEGRAL - DERIVATIVO

Como anteriormente se señaló esta acción combinada reúne las ventajas de las tres acciones de control individuales

Luego de probar diferentes métodos de control PID, y diferentes magnitudes y combinaciones en su estructura, se sintonizó un controlador que cumple cómodamente con los requerimientos del sistema. Se probó este controlador con diferentes condiciones dinámicas iniciales y diferentes puntos de equilibrio, y se obtuvieron las respuestas que se publican a continuación.

### 12.4.1 Ángulo de elevación con control PID

PRUEBA 17	Características de la respuesta
Proporcional ( $K_p$ ) = 4	El sistema describe un comportamiento amortiguado, con un error en estado estacionario del 6.5%, un tiempo de estabilización de 5 segundos, y un tiempo de subida de 3 segundos
Integral ( $K_i$ ) = 30	
Derivativo ( $K_d$ ) = 2.2	
Punto de equilibrio = $10.8^\circ$	
Tiempo de Prueba = 90 seg	

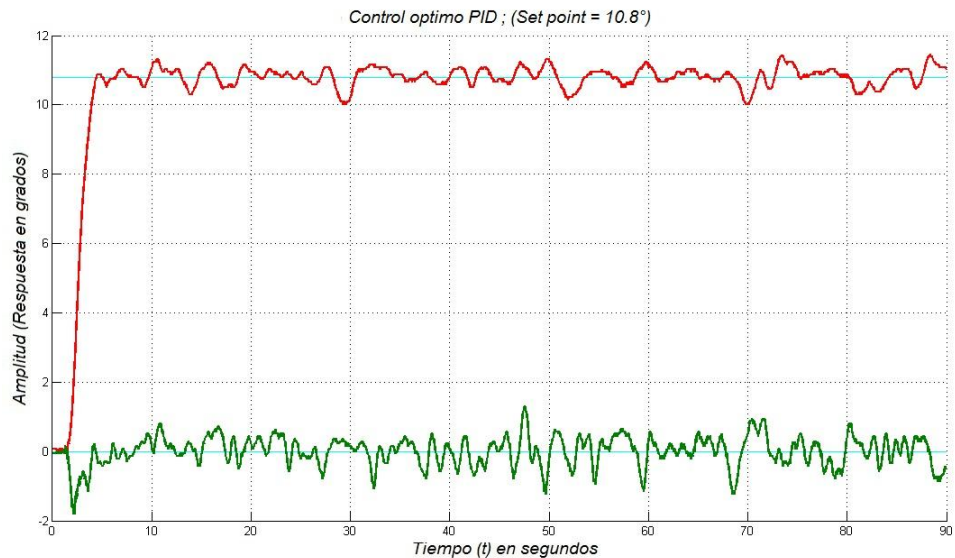


FIGURA 102. Respuesta real del ángulo de elevación con controlador PID y set point en  $10.8^\circ$ .

PRUEBA 18	Características de la respuesta
Proporcional ( $K_p$ ) = 4	El sistema describe un comportamiento amortiguado, con un error en estado estacionario del 2%, un tiempo de estabilización de 8 segundos, un tiempo de subida de 4 segundos, un sobreimpulso del 16%.
Integral ( $K_i$ ) = 30	
Derivativo ( $K_d$ ) = 2.2	
Punto de equilibrio = $18^\circ$	
Tiempo de Prueba = 90 seg	

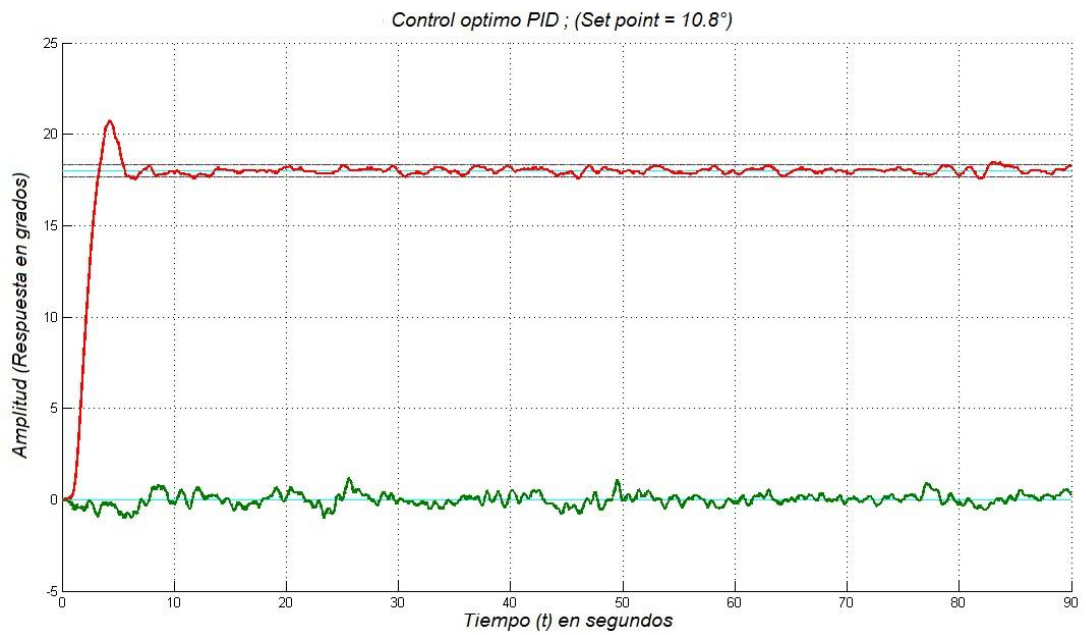


FIGURA 103. Respuesta real del ángulo de elevación con controlador PID y set point en 18°.

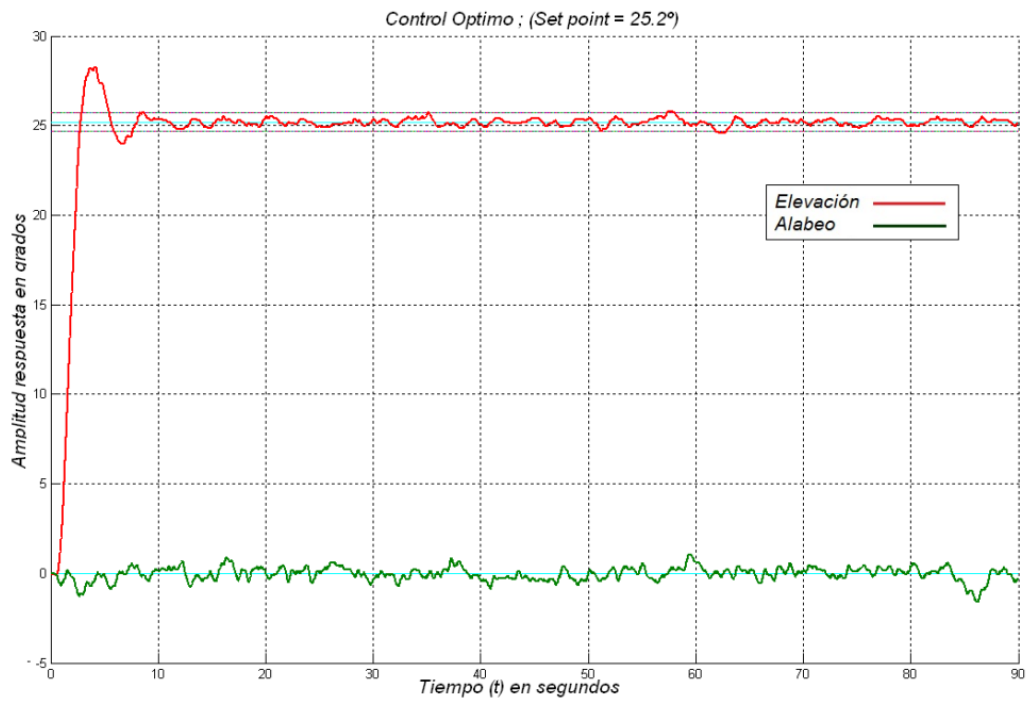


FIGURA 104. Respuesta real del ángulo de elevación con controlador PID y set point en 25.2°.

PRUEBA 19	Características de la respuesta
Proporcional ( $K_p$ ) = 4	El sistema describe un comportamiento amortiguado, con un error en estado estacionario del 2%, un tiempo de estabilización de 8 segundos, un tiempo de subida de 4 segundos, un sobreimpulso del 11%, y un subimpulso del 3%.
Integral ( $K_i$ ) = 30	
Derivativo ( $K_d$ ) = 2.2	
Punto de equilibrio = $25.2^\circ$	
Tiempo de Prueba = 90 seg	

### 12.4.2 Ángulo de alabeo con control PID

PRUEBA 20	Características de la respuesta
Proporcional ( $K_p$ ) = 1	El sistema tiene una estabilización en 9 segundos, y un tiempo de subida de 4 segundos, con un error en estado estacionario del 4%, no presenta sobrepicos ni subpicos importantes
Integral ( $K_i$ ) = 8	
Derivativo ( $K_d$ ) = 0.35	
Punto de equilibrio = $+15^\circ$	
Tiempo de Prueba = 90 seg	

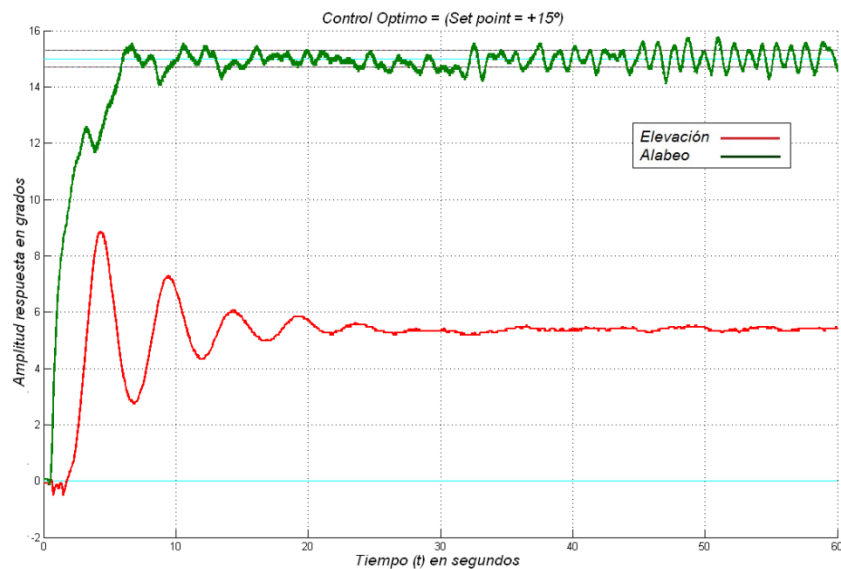


FIGURA 105. Respuesta real del ángulo de alabeo con controlador PID y set point en  $15^\circ$ .

<b>PRUEBA 21</b>	<b>Características de la respuesta</b>
Proporcional ( $K_p$ ) = 1	El sistema tiene una estabilización en 9 segundos, y un tiempo de subida de 4 segundos, con un error en estado estacionario del 4%, no presenta sobrepicos ni subpicos importantes
Integral ( $K_i$ ) = 8	
Derivativo ( $K_d$ ) = 0.35	
Punto de equilibrio = $-15^\circ$	
Tiempo de Prueba = 90 seg	

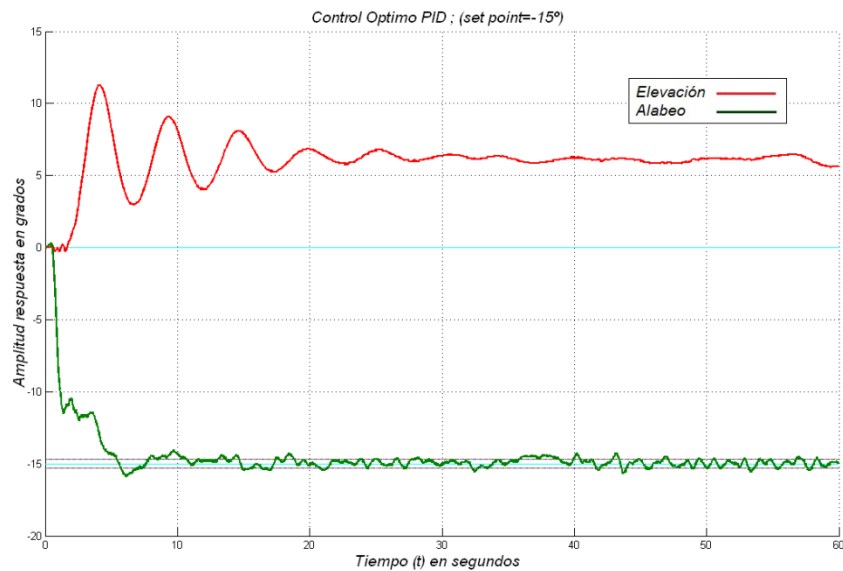


FIGURA 106. Respuesta real del ángulo de alabeo con controlador PID y set point en  $-15^\circ$ .

<b>PRUEBA 22</b>	<b>Características de la respuesta</b>
Proporcional ( $K_p$ ) = 1	El sistema tiene una estabilización en 12 segundos, y un tiempo de subida de 5 segundos, con un error en estado estacionario del 2%, presenta un sobrepico del 4.5% y un par de oscilaciones antes de llegar a la estabilizacion
Integral ( $K_i$ ) = 8	
Derivativo ( $K_d$ ) = 0.35	
Punto de equilibrio = $+45^\circ$	
Tiempo de Prueba = 90 seg	

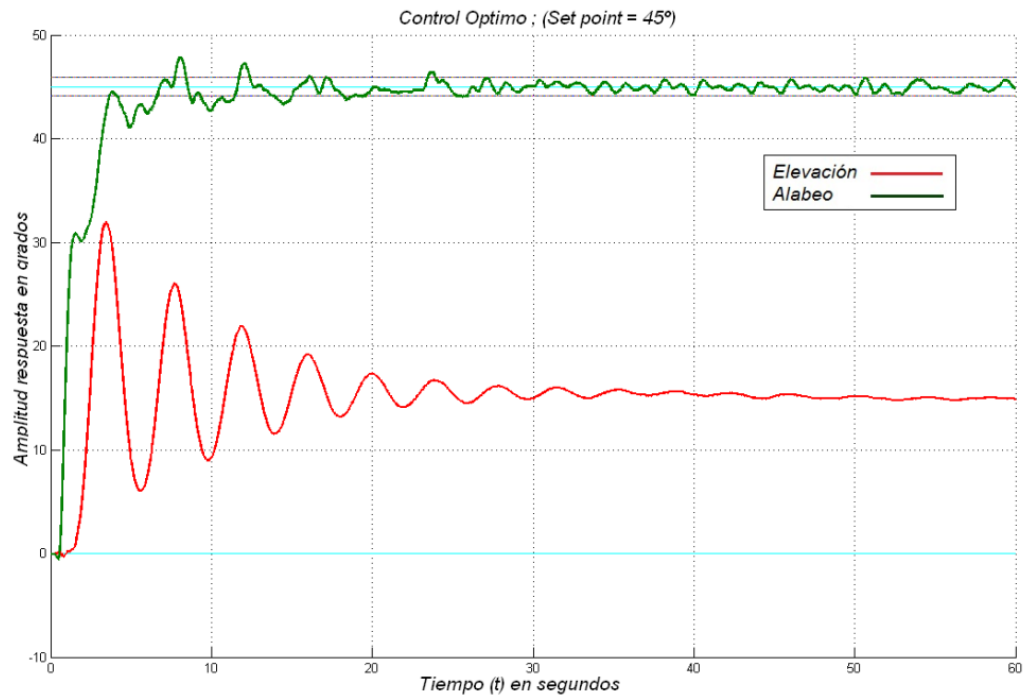


FIGURA 107. Respuesta real del ángulo de alabeo con controlador PID y set point en 45°.

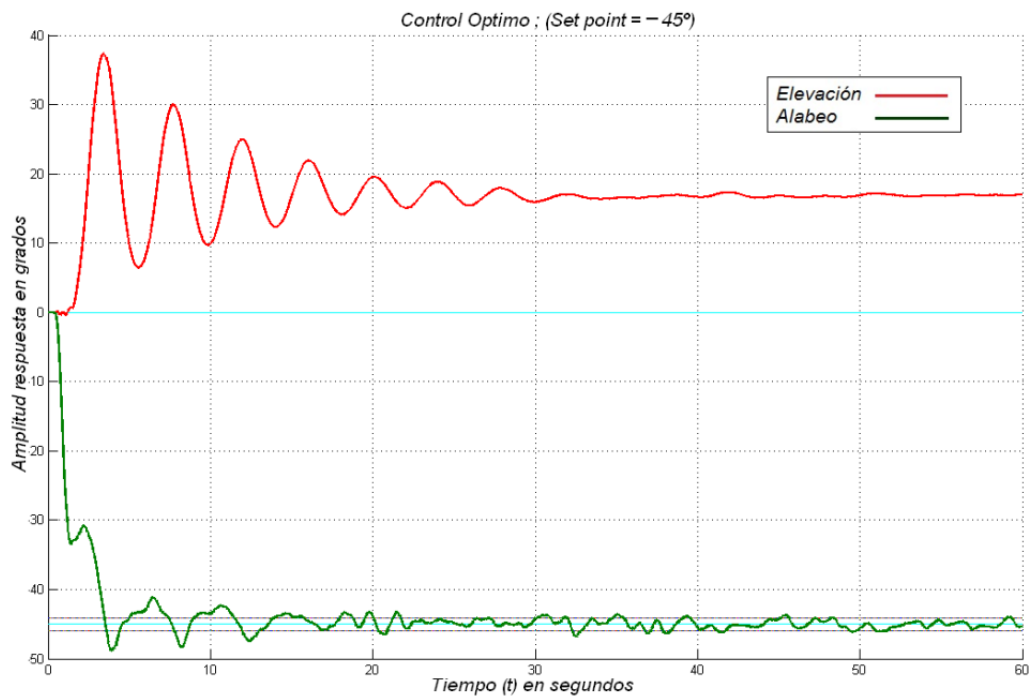


FIGURA 108. Respuesta real del ángulo de alabeo con controlador PID y set point en -45°.

PRUEBA 23	Características de la respuesta
Proporcional ( $K_p$ ) = 1	El sistema tiene una estabilización en 12 segundos, y un tiempo de subida de 5 segundos, con un error en estado estacionario del 2%, presenta un sobrepico del 4.5% y un par de oscilaciones antes de llegar a la estabilización
Integral ( $K_i$ ) = 8	
Derivativo ( $K_d$ ) = 0.35	
Punto de equilibrio = $-45^\circ$	
Tiempo de Prueba = 90 seg	

La combinación de acciones anteriormente elegidos no sugiere que sea la única combinación posible que cumpla con los requerimientos de diseño, será labor de futuras investigaciones encontrar un controlador que responda de manera más eficiente ya sea a partir de la afinación del controlador PID propuesto o simplemente con la aplicación de otro método de control.

## 13 CONCLUSIONES

Se diseñó y construyó la plataforma mecánica del Helibot 3DOF que cumple con condiciones técnicas y dimensionales las cuales se validaron con la implementación del controlador PID en el etapa de sintonización, su diseño modular es flexible, permite cambios posteriores en su estructura y se ofrece como herramienta de apoyo en otros ambientes de investigación.

En el presente trabajo de grado se resolvió el problema de la estabilización para el Helibot 3DOF gobernado por dos entradas de control. A partir del modelamiento matemático se advirtió que el sistema describe un comportamiento de segundo orden por lo cual, los parámetros para el diseño del controlador se obtuvieron a partir de los parámetros dinámicos  $\xi$  y  $W_n$  de un sistema de este tipo que son definidos por las características técnicas y dimensionales de la planta. Luego se procedió al diseño del controlador PID en base a los objetivos de control utilizando MATLAB como herramienta de simulación en esta etapa llegamos a las siguientes conclusiones:

- De los diferentes métodos de control propuestos en la etapa de simulación nos dimos cuenta que aunque el controlador PID con filtro derivativo es el que mejor ajusta la respuesta a los requerimientos de diseño, el método del PID clásico, de estructura más simple y fácil implementación, responde eficientemente como estrategia de compensación, motivo por el cual se definió como método de control en la estabilización del Helibot 3DOF.
- Aunque se propusieron dos diferentes métodos de sintonización (método de las raíces y método de Ziegler-Nichols) en la simulación, es imprescindible

hacer una sintonización experimental sobre la planta real para ajustar el modelo y definir así el valor adecuado de las ganancias.

A través de la aplicación del controlador diseñado en la etapa de simulación sobre la planta real nos dimos cuenta que el tratamiento matemático es insuficiente y no regula de forma apropiada el comportamiento del Helibot 3DOF, ya que este no tiene en cuenta condiciones como la fricción, ruido, vibraciones, etc.

En cuanto a estos resultados del diseño se encontró que el sistema controlado satisface las condiciones de estabilidad, presenta un buen seguimiento de la trayectoria deseada en diferentes configuraciones y con diferentes condiciones iniciales, y se lograron atenuar las perturbaciones. Estos resultados por lo tanto prueban que los objetivos de control considerados para el Helibot 3DOF se cumplieron satisfactoriamente.

La combinación y magnitudes de las ganancias propuestas en el capítulo 12, no sugiere que sea el único controlador posible que cumpla con los requerimientos de diseño, será labor de futuras investigaciones encontrar un controlador que responda de manera más eficiente ya sea a partir de la afinación del controlador PID propuesto o simplemente con la aplicación de otro método de control.

El uso de sistemas embebidos facilitan la variación de parámetros de los sistemas de control actuales basados en PID, de manera económica y confiable, debido a esto, los costos de implementación se reducen considerablemente con su contraparte analógica. Los sistemas de control basados en sistemas digitales tienen mejoras en su control como lo son inmunidad al ruido, y la reducción en el consumo de potencia.

El sistema desarrollado permitirá aumentar la curva de aprendizaje en las áreas de control y robótica al ser esto un modelo didáctico en donde el usuario, en este caso el estudiante, puede interactuar con el sistema y verificar físicamente los resultados de cálculos e hipótesis relacionadas con el control óptimo del sistema.

## 14 RECOMENDACIONES

Normalmente los reguladores suelen dar la acción de control oportuna para que la salida del sistema presente las especificaciones requeridas, pero a menudo dicha acción de control es bastante superior a la soportada por el sistema. Aunque en el sistema con el que estamos trabajando dispone de los mecanismos de protección necesarios para limitar la tensión de salida a los 10 voltios soportados por el sistema, recomendamos comprobar en la etapa de simulación cómo afecta dicha limitación a la respuesta del sistema.

Se recomienda la implementación de un control adicional en el plano Z o guiñada, que permitirá un control más robusto de Helibot 3DOF.

La tarjeta que fue diseñada, permite el ensamble de otros métodos de control alternativo como modelos predictivos, logica fuzzy, y controladores adaptativos, entre muchos otros, cada uno de estos controles requiere de un estudio metódico para su correcto funcionamiento, por lo cual se hace indispensable la integración de personal capacitado en el área de electrónica, lo que a su vez permitirá el complemento de dos áreas de la ingeniería.

El sistema embebido diseñado permite la variación dinámica del setpoint a través de control remoto, ya que la estructura del hardware favorece esta disposición, sin embargo recomendamos que en trabajos futuros se haga la implementación del software y de esta manera explorar este método de control utilizando como plataforma de pruebas el Helibot 3DOF.

La interfaz desarrollada en MATLAB cumple con su función de comunicar el sistema con el usuario y permite cambiar parámetros del controlador así como analizar offline la respuesta del sistema controlado, esta interfaz tiene una estructura básica sin mayores prestaciones, sin embargo será tema de futuros trabajos elaborar una interfaz que permita la utilización de otros métodos de control con herramientas especializadas como LABVIEW y VISUALBASIC.

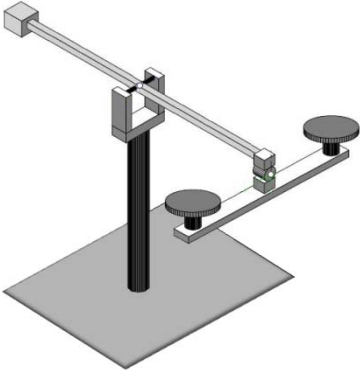
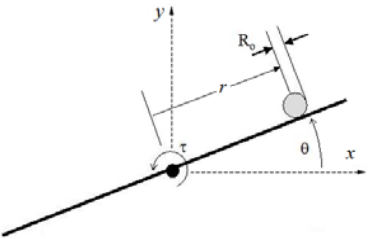
## BIBLIOGRAFÍA

- [1] MOTT, Robert L. Diseño de elementos de máquinas. Pearson Education, Mexico, 4 edition, 2006.
- [2] NORTON, Robert L. Diseño de Máquinas. México: Prentice Hall, 1999
- [3] OGATA, Katsuhiko. Ingeniería de control moderna. Prentice hall, México, 4 edition, 2002.
- [4] ERONINI, Umez. Dinámica de Sistemas y Control. Thomson Learning International. 2001.
- [5] DORF, Richard C; BISHOP, Robert H. Modern Control Systems. Prentice Hall Inc. 2001.
- [6] SPOONER, Jeffrey; MAGGIORE, Manfredi; ORDÓÑES, Raúl; PASSINO, Kevin. Control estable adaptativo para sistemas no lineales, Prentice Hall Inc. 1998.
- [7] ASTRÖM, Karl, HÄGGLUND, Tore, PID Controllers: Theory, Design, and Tuning. 2<sup>nd</sup> Edition. Editorial ISA, 2004.
- [8] SHAW, John A. The PID Control Algorithm. 2<sup>nd</sup> Edition. 2003.
- [9] LOPES R.V; GALVAO, R. K. H; MILHAN, A. P; BECERRAY V. M and YONEYAMA, T. 'Modelling and Constrained Predictive Control of a 3DOF Helicopter. XVI Congreso Brasileiro de Automatica. 2006
- [10] ANDRIEVSKY B; PEAUCELLE D and FRANKOV A.L. Adaptive Control of 3DOF Motion for LAAS Helicopter Benchmark: Design and Experiments, Proceedings of the American Control Conference. 2007.
- [11] KIEFER, T; KUGI, A; GRAICHEN, K; ZEITZ, M. Feedforward and Feedback Tracking Control of a 3DOF Helicopter Experiment under Input and Output Constraints," Proceedings of the 45<sup>th</sup> IEEE Conference on Decision and control. 2006.

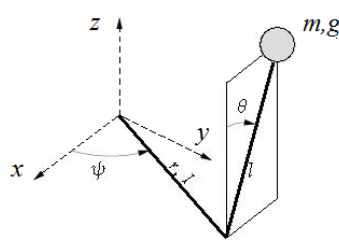
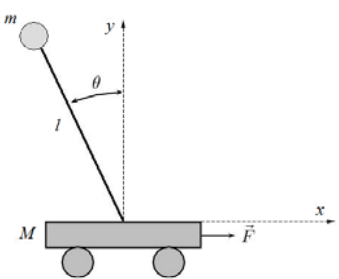
- [12] IRIBAS, M. Controlador de vuelo de la maqueta de un helicóptero, mediante técnicas de control robusto QFT, Proyecto de Fin de Carrera presentado en la Universidad Pública de Navarra. España.
- [13] GARCÍA-SANZ, Mario; ELSO, Jorge; EGAÑA, Igor. Control del Ángulo de Cabeceo de un Helicóptero como Benchmark de diseño de controladores. Revista Iberoamericana de Automatica e Informatica Industrial. vol. 3, no. 2. 2006.
- [14] GARCÍA-SANZ, Mario; EGAÑA, Igor; CASTILLEJO, Javier. Torneo de Loop-Shaping en el Control de un Helicoptero de Laboratorio. Departamento de Automática y Computación, Campus Arrosadía. España. 2006
- [15] EGAÑA, Igor, IRIBAS, M.; CASTILLEJO, Javier; GARCÍA-SANZ, Mario. Control Robusto QFT de un Helicóptero de 3 GDL". *XXII Jornadas de Automática*, Barcelona, España. 2001.
- [16] STARKOV, Kontantin, AGUILAR, Luis Tupack, ORLOV, Yuri. Sliding mode control synthesis of a 3-DOF helicopter prototype using position feedback. in 10<sup>th</sup> International Workshop on Variable Structure Systems, Antalya, Turkey. 2008.
- [17] PERK, E. B, JEAN-JACQUES E., "Motion Primitives for Robotic Flight Control. *CoRR abs/cs/0609140*. 2006.
- [18] TONDEL, P. and JOHANSEN T.A. "Complexity Reduction in Explicit Linear Model Predictive Control. IFAC 15<sup>th</sup> Triennial World Congress. 2002.
- [19] AVILA-VILCHIS, J.C., BROGLIATO, B., DZUL, A. and LOZANO R. Nonlinear modelling and Control of Helicopter," *Automática*, vol. 39. 2003.
- [20] ISIDORI, L. MARCONI and SERRANI, A. Robust Nonlinear motion Control of a Helicopter. *IEEE Trans. On Automatic Control*, vol. 48. 2003.
- [21] ISHUTKINA M. A., CASINI, M. and VICINO, A. An Internet Based Laboratory for Control of a Safety Critical System. *Cybernetics*. 2004.

# **ANEXOS**

## ANEXO A.ANALISIS DE LAS ALTERNATIVAS

EQUIPOS CONSIDERADOS I				
	Esquema	Objetivo de control	Ventajas	Desventajas
<b>3DOF HELICOPTER</b>		<p>Estabilizar la estructura entorno a un pivote a través de variaciones en el voltaje de los motores DC que impulsan las hélices y crean una fuerza de empuje que junto con el contrapeso permite su estabilidad. Utiliza 2 sensores, uno para registrar las variaciones en el ángulo de la barra que sostiene los motores, y otro para registrar la variación en el ángulo de elevación del sistema.</p>	<ul style="list-style-type: none"> <li>✓ Didáctico, permite la interacción con el usuario.</li> <li>✓ Versátil.</li> <li>✓ Movimiento en 3 ejes.</li> <li>✓ Representa muchos de los retos de la ingeniería Mecánica y aeroespacial.</li> <li>✓ Permite su control en lazo cerrado.</li> <li>✓ Flexibilidad.</li> </ul>	<ul style="list-style-type: none"> <li>✗ Difícil control.</li> <li>✗ Necesita de un espinoso tratamiento matemático.</li> <li>✗ Inestable.</li> <li>✗ No lineal.</li> <li>✗ Alto Costo de Adquisición (ver anexos).</li> <li>✗ Difícil Construcción.</li> </ul>
<b>BALL &amp; BEAM</b>		<p>Regular automáticamente la posición de la bola en el balancín cambiando el ángulo de la barra a través de variaciones en el voltaje del motor DC. Utiliza 2 sensores, uno que determina la posición de la bola en la viga, y el otro para sensar la posición angular de la viga.</p>	<ul style="list-style-type: none"> <li>✓ Fácil construcción</li> <li>✓ Control sencillo</li> <li>✓ Simple</li> <li>✓ Útil e importante para enseñar la teoría de control en sistemas inestables.</li> </ul>	<ul style="list-style-type: none"> <li>✗ Rápida estabilidad</li> <li>✗ El movimiento se limita a un plano.</li> <li>✗ No Lineal</li> <li>✗ Alto Costo.</li> <li>✗ Inestable en lazo abierto</li> <li>✗ Poco Didáctico</li> </ul>

## EQUIPOS CONSIDERADOS II

	Esquema	Objetivo de control	Ventajas	Desventajas
<b>ROTARY INVERTED PENDULUM</b>		<p>Mantener la barra (péndulo) en una posición vertical (set point) a través de la aplicación de un movimiento rotacional en el brazo causado por la variación en el voltaje de un motor DC.</p> <p>Utiliza 2 sensores: uno para registrar la variación del ángulo del péndulo, y el otro para sensar el cambio en el ángulo del brazo.</p>	<ul style="list-style-type: none"> <li>✓ Representa nociones básicas de la robótica.</li> <li>✓ Didáctico, permite la interacción con el usuario.</li> <li>✓ Fácil construcción.</li> <li>✓ Permite la aplicación de diferentes tipos de control.</li> </ul>	<ul style="list-style-type: none"> <li>✗ Alto Costo de Adquisición.</li> <li>✗ No Lineal.</li> <li>✗ Inestable.</li> <li>✗ Su funcionamiento está más direccionado a la robótica y no a la Ingeniería Mecánica.</li> </ul>
<b>INVERTED LINEAR PENDULUM</b>		<p>Mantener la barra (péndulo) en una posición vertical (set point) a través de la aplicación de un movimiento translacional en el carro causada por la variación en el voltaje de un motor DC. Utiliza 2 sensores; uno que registra la variación angular del péndulo, y el otro registra la posición del carro en el riel.</p>	<ul style="list-style-type: none"> <li>✓ Fácil control.</li> <li>✓ Flexible</li> <li>✓ Didáctico.</li> <li>✓ Permite la aplicación de diferentes tipos de control.</li> <li>✓ Ya existe uno en nuestra escuela donde el control se hace por lógica fuzzy.</li> </ul>	<ul style="list-style-type: none"> <li>✗ El movimiento se limita a un plano.</li> <li>✗ Alto Costo (ver anexos).</li> <li>✗ No lineal.</li> <li>✗ Requiere gran disponibilidad de espacio lo que dificulta su transporte y almacenaje.</li> </ul>

Cuadro 1. Alternativas para la solución

En la Tabla 1 se evalúan las alternativas anteriores dándole una ponderación entre 1 y 5 según las expectativas y requerimientos del diseño.

			3dof Helicopter		Ball & Beam		Rotary Inverted Pendulum		Inverted Linear Pendulum	
Criterio de valoración	Parámetro	Ponderación Individual	Nota	Ponderación	Nota	Ponderación	Nota	Ponderación	Nota	Ponderación
Usuario	0,3	Fácil de operar	5	0,225	5	0,225	4	0,18	4	0,18
		Interactivo	5	0,225	3	0,135	3	0,135	4	0,18
		Flexible	4	0,12	3	0,09	5	0,15	5	0,15
Costo	0,25	Económico	3	0,1125	4	0,15	4	0,15	3	0,1125
		Repuestos	4	0,1	4	0,1	4	0,1	4	0,1
Diseño	0,25	Bajas Vibraciones	5	0,09375	5	0,09375	3	0,05625	3	0,05625
		Estético	5	0,125	4	0,1	5	0,125	5	0,125
Dimensiones	0,2	Almacenamiento	5	0,075	5	0,075	3	0,045	5	0,075
		Movilidad	5	0,1	5	0,1	3	0,06	5	0,1
		<b>TOTAL</b>	<b>1,17625</b>		<b>1,06875</b>		<b>1,00125</b>		<b>1,07875</b>	

Tabla 1. Selección de la Solución.

De lo anterior elegimos el helicóptero 3GDL ya que obtuvo la ponderación total más alta, es el equipo con mayores cualidades pedagógicas, y presenta además un acople óptimo con las especificaciones de diseño.

# ANEXO B. DATASHEET ENCODER INCREMENTAL

## Light Duty Incremental Encoders

### Features

A light-duty encoder is a cost-effective encoder for small applications and has the following features:

- Small body with 38 mm diameter and 30 mm depth
- Dust proof (IP40 rating)
- 6 mm standard shaft or 8 mm hollow shaft
- Resolution available from 100 pulses per revolution to 2500 pulses per revolution
- Open collector or line driver output
- Up to 200 kHz response frequency
- Two-meter cable, tinned ends



Standard shaft (TRD-S) model



Hollow shaft (TRD-SH) model

Note: Yellow shaded part numbers are non-stock. Availability may range from four to six weeks.

Light Duty Standard Shaft Incremental Encoders (NPN Open Collector and Line Driver models)						Light Duty Hollow Shaft Incremental Encoders (NPN Open Collector and Line Driver models)					
Part Number	Price	Pulses per Revolution	Input Voltage	Output	Body Diameter	Part Number	Price	Pulses per Revolution	Input Voltage	Output	Body Diameter
TRD-S100-BD	<-->	100	12-24 VDC	NPN open collector	38mm	TRD-SH100-BD	<-->	100	12-24 VDC	NPN open collector	38mm
TRD-S200BD	<-->	200				TRD-SH200BD	<-->	200			
TRD-S250BD	<-->	250				TRD-SH250BD	<-->	250			
TRD-S300BD	<-->	300				TRD-SH300BD	<-->	300			
TRD-S360-BD	<-->	360				TRD-SH360-BD	<-->	360			
TRD-S400BD	<-->	400				TRD-SH400BD	<-->	400			
TRD-S500-BD	<-->	500				TRD-SH500-BD	<-->	500			
TRD-S600BD	<-->	600				TRD-SH600BD	<-->	600			
TRD-S800BD	<-->	800				TRD-SH800BD	<-->	800			
TRD-S1000-BD	<-->	1000				TRD-SH1000-BD	<-->	1000			
TRD-S1024-BD	<-->	1024				TRD-SH1024BD	<-->	1024			
TRD-S1200BD	<-->	1200				TRD-SH1200BD	<-->	1200			
TRD-S2000BD	<-->	2000				TRD-SH2000BD	<-->	2000			
TRD-S2500-BD	<-->	2500				TRD-SH2500-BD	<-->	2500			
TRD-S100-VD	<-->	100				5VDC	Line driver (differential)	38mm			
TRD-S200VD	<-->	200	TRD-SH200VD	<-->	200						
TRD-S250VD	<-->	250	TRD-SH250VD	<-->	250						
TRD-S300VD	<-->	300	TRD-SH300VD	<-->	300						
TRD-S360-VD	<-->	360	TRD-SH360-VD	<-->	360						
TRD-S400VD	<-->	400	TRD-SH400VD	<-->	400						
TRD-S500-VD	<-->	500	TRD-SH500-VD	<-->	500						
TRD-S600VD	<-->	600	TRD-SH600VD	<-->	600						
TRD-S800VD	<-->	800	TRD-SH800VD	<-->	800						
TRD-S1000-VD	<-->	1000	TRD-SH1000-VD	<-->	1000						
TRD-S1024-VD	<-->	1024	TRD-SH1024VD	<-->	1024						
TRD-S1200VD	<-->	1200	TRD-SH1200VD	<-->	1200						
TRD-S2000VD	<-->	2000	TRD-SH2000VD	<-->	2000						
TRD-S2500-VD	<-->	2500	TRD-SH2500-VD	<-->	2500						

<http://www.automationdirect.com/pn/TRD-SH1024VD>

# Light Duty Incremental Encoders

## Specifications

Electrical Specifications				
<b>Model</b>		<i>TRD-Sxxxx-BD</i> <i>TRD-SHxxxxBD</i> (open collector)	<i>TRD-Sxxxx-VD</i> <i>TRD-SHxxxxVD</i> (line driver)	
<b>Power Supply</b>	<b>Operating Voltage</b>	10.8 - 26.4VDC*	+4.75 - 5.25VDC*	
	<b>Allowable Ripple</b>	3% max.	-	
	<b>Current Consumption</b>	50 mA max.		
<b>Signal Waveform</b>		Two-phase + home position		
<b>Max. Response Frequency</b>		200kHz		
<b>Duty Ratio</b>		50 ± 25%		
<b>Phase Difference Width</b>		25 ± 12.5%		
<b>Signal Width at Home Position</b>		100 ± 50%		
<b>Output</b>	<b>Rise/Fall Time</b>		1µs max. (when cable length is 1m)	-
	<b>Output Type</b>		NPN open collector output, sinking	Line driver output (26C31 or equivalent)
	<b>Output Logic</b>		Negative logic (active low)	Negative logic (active high)
	<b>Output Current</b>	<b>H</b>	-	2.5 V min.
	<b>Output Voltage</b>	<b>L</b>	0.4 V max.	0.5 V max.
	<b>Influx Current</b>		30mA max.	-
	<b>Load Power Voltage</b>		35 VDC max.	-
	<b>Short-Circuit Protection</b>		Between output and power supply	
* To be supplied by Class II source				
Mechanical Specifications				
<b>Starting Torque</b>	Max. 0.001 Nm (.00074 ft./lbs)			
<b>Max. Allowable Shaft Load</b>	Radial: 20N (4.5 lbs) Axial: 10N (2.25 lbs)			
<b>Max. Allowable Speed</b>	6000 rpm (highest speed that can support the mechanical integrity of encoder)			
<b>Wire Size</b>	AWG26			
<b>Weight</b>	Approx. 150g (5.3 oz) with 2m cable			
Environmental Specifications				
<b>Ambient Temperature</b>	10 to 70°C; 14 to 158°F			
<b>Storage Temperature</b>	-25 to 85°C; -13 to 185°F			
<b>Operating Humidity</b>	35-85% RH			
<b>Voltage Withstand</b>	500VAC (50/60Hz) for one minute			
<b>Insulation Resistance</b>	50MΩ min.			
<b>Vibration Resistance</b>	Durable for one hour along three axes at 10 to 55 Hz with 0.75 amplitude			
<b>Shock Resistance</b>	11 ms with 490 m/s <sup>2</sup> applied three times along three axes			
<b>Protection</b>	IP40: dust proof			



## ANEXO C. MOTOR Y SPEED CONTROL

El motor seleccionado para la aeronave es un motor brushless outrunner. Estos motores carecen de colector y escobillas. Funcionan en AC debido a que se alimentan con una señal trifásica (Por tal motivo se requiere de los controladores de velocidad), sin embargo se clasifica como un motor de DC porque al igual que los motores comunes tienen imanes permanentes.

El motor es marca TURNIGY modelo TR2830-1050 y presenta las siguientes características proporcionadas por el distribuidor.

Parámetro	Valor
Kv	1050 [rpm/V]
Máxima carga	14 [A]
Máxima eficiencia	12 [A]
Voltaje	6 - 10 [V]
Empuje	400 - 900 [g]
Peso	68 [g]
Hélices (Recomendada)	8 x 4.0 ''
Corriente sin carga	0.9 [A]

TABLA 2. Motor Brushless outrunner TR2830-1050

### CONTROL DE VELOCIDAD

Es el dispositivo encargado de convertir la alimentación continua en trifásica. Además controla de velocidad del motor mediante una señal PWM que proviene del control ubicado en tierra. Para su selección, es necesario que el control de velocidad al menos proporcione la corriente máxima de alimentación que el motor requiera. El control de velocidad es marca TURNIGY modelo TR P18A. Estos dispositivos soportan hasta 18[A] y cuentan con un circuito eliminador de batería BEC.

## ANEXO D. RODAMIENTO LINEAL

### Linear Ball Bearings, Solid and Drawn Cup Types

NTN

### Linear Ball Bearings, Solid and Drawn Cup Types

Four to nine rows of balls are configured equally in the outer ring (outer cylinder). The ball rows circulate in axial direction while being guided by the cage. Thus, these bearing types move infinitely on a shaft in axial direction. However, these bearing types can't rotate.

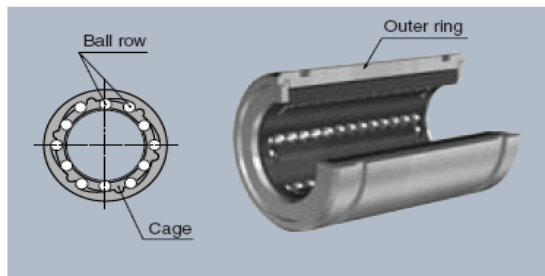


Fig. 1

#### Dimensional accuracy

The respective accuracy of the boundary dimensions, inscribed circle diameter ( $F_w$ ), outer diameter ( $D$ ), and with higher dimensional accuracy is offerable on special request. Feel free to contact NTN when such a bearing is needed.

**For Type KH** (drawn cup type), its outer ring is so thin-walled that it deforms inevitably to some extent in the manufacturing processes, particularly heat-treating process, but it is so designed as to be restored normally from such deformation and fulfill its specific function with the required accuracy by being press-fitted in a housing with the required dimensional accuracy.

Feel free to contact NTN for the method of measuring the dimensional accuracy.

#### Bearing fit

The use of a shaft or a housing with the dimensional tolerance shown in **Table 1** would ensure proper radial clearance. Where further small radial clearance is wanted or preload is applied, the radial clearance is adjusted using a split housing or otherwise bearing fit required for smaller clearance or preload is selected.

**Table 1 Bearing fit**

Type	Shaft	Housing
series KLM Solid type	g6 (g5)	H7 (H6)
series HK Drawn-cup type	h6 (j5)	H7 (H6) - steel series - K7 (K6) - light metal alloy series -

Note) The parenthesized data is applied to shaft/housing subjected to higher accuracy or of vertical construction.

#### Shaft and housing requirements

Any shaft /housing on/in which these bearing types are fitted must meet the requirements specified in **Table 2**.

**Table 2 Shaft and housing requirements**

Characteristics	Shaft	Housing
Roundness (max)	IT3	IT4
Cylindricity (max)	IT2	IT4
Surface roughness (max.)	0.4a	1.6a
Surface hardness	HRC58~64	—
Hardened layer depth (min)	0.4mm	—

#### How to mount

The housing for **Type KLM** (solid type) can't be fixed perfectly with interference only and, therefore, must be fixed in axial direction using a snap ring.

On the other hand, **Type KH** (drawn cup type) needs no axial fixing by a snap ring because it is press-fitted in a housing with interference. For press-fitting, press the outer ring at its stamped mark side (hardened side) using

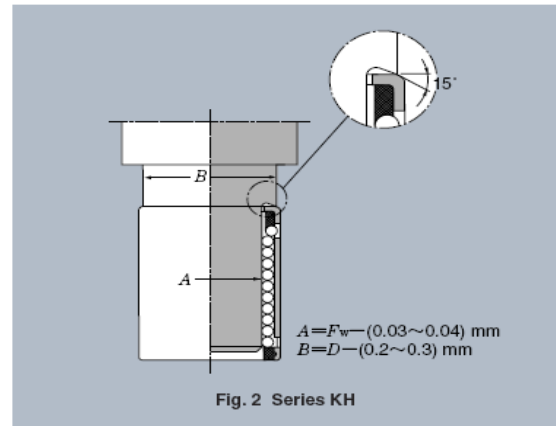


Fig. 2 Series KH

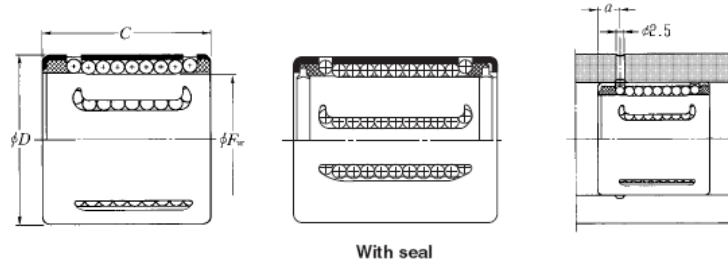
#### Accessories

Shafts, shaft support stands and housings exclusive for NTN linear ball bearings are also offerable. Feel free to contact NTN for the detailed information.

# Machined cup linear ball bearings

NTN

Type KH  
Type KH · LL



$F_w$  6~50mm

$F_w$	Boundary dimensions mm			Bearing numbers	Basic load ratings				Number of ball rows	Mass kg (approx.)
	$D$	$C$	$a^{1)}$		dynamic N	static	dynamic kgf	static		
					$C_r$	$C_{or}$	$C_r$	$C_{or}$		
6	12	22	4	KH0622 <sup>2)</sup>	380	225	39	23	4	0.007
8	15	24	5	KH0824 <sup>2)</sup>	420	255	43	26	4	0.012
10	17	26	5	KH1026 <sup>2)</sup>	480	325	49	33	4	0.015
12	19	28	6	KH1228	605	495	62	51	5	0.018
	19	28	6	KH1228LL	605	495	62	51	5	0.018
14	21	28	6	KH1428	600	505	61	51	5	0.021
16	24	30	7	KH1630LL	775	600	79	61	5	0.027
20	28	30	7	KH2030	1 050	880	107	90	6	0.033
	28	30	7	KH2030LL	1 050	880	107	90	6	0.033
25	35	40	8	KH2540	1 930	1 560	197	159	6	0.066
	35	40	8	KH2540LL	1 930	1 560	197	159	6	0.066
30	40	50	8	KH3050	2 700	2 450	275	250	7	0.095
	40	50	8	KH3050LL	2 700	2 450	275	250	7	0.095
40	52	60	9	KH4060	4 250	4 000	435	410	8	0.18
50	62	70	9	KH5070	5 300	5 700	540	580	9	0.24

Note 1) Showing a-value from the side face with stamped mark thereon.  
2) Imported product from INA, Germany.

## ANEXO E. DATASHEET ACOPLE GJ – 6D

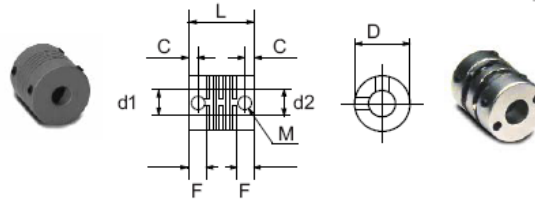
# Encoder Couplings

Couplings provide a connection between encoders and solid shafts. We offer fiberglass and aluminum metric-to-metric couplings and aluminum metric-to-S.A.E. couplings. Use the selection guide below to help select your coupling.

Couplings Selection Guide and Dimensions														
Type	Part No.	Price	Applicable Encoders	Material	d1	d2	D	L	F	C	M	a	E	S
					mm*									
Fiberglass (metric)	<a href="#">GJ-6D</a>	<-->	TRD-S	Glass-fiber reinforced polyacetal	6	6	15	21.6	5.2	2.8	M3 set screw	6° max.	0.5mm max.	0.12mm max.
	<a href="#">GJ-8D</a>	<-->	TRD-N,NA		8	8	19	24	6.8	3.5	M4 set screw	5° max.	0.5mm max.	0.12mm max.
	<a href="#">GJ-10D</a>	<-->	TRD-GK		10	10	22	26.2	7.1	3.6	M4 set screw	5° max.	0.5mm max.	0.12mm max.
Aluminum (metric)	<a href="#">RU-075D</a>	<-->	TRD-S	Aluminum alloy	6	6	19.1	19.1	4.8	9.5	M3 set screw	5° max.	0.25mm max.	0.12mm max.
	<a href="#">JU-100D</a>	<-->	TRD-N,NA		8	8	25.4	25.4	6.9	3.8	M5 set screw	5° max.	0.25mm max.	0.12mm max.
	<a href="#">RU-100D</a>	<-->	TRD-GK		10	10	25.4	25.4	6.9	3.8	M5 set screw	5° max.	0.25mm max.	0.12mm max.
Aluminum (metric to SAE)	<a href="#">MCGL16-6-635</a>	<-->	TRD-S	Aluminum alloy (Bent plate: Polyimide)	6	6.35 (0.25")	16	23.2	7	3	M3 set screw	3.5° max.	0.3mm max.	0.3mm max.
	<a href="#">MCGL20-8-635</a>	<-->	TRD-N,NA		8	6.35 (0.25")	20	26	7.5	3.7	M3 set screw	3.5° max.	0.3mm max.	0.4mm max.
	<a href="#">MCGL20-8-952</a>	<-->	TRD-N,NA		8	9.52 (0.375")	20	26	7.5	3.7	M3 set screw	3.5° max.	0.3mm max.	0.4mm max.
	<a href="#">MCGL25-10-635</a>	<-->	TRD-GK		10	6.35 (0.25")	25	30.2	9	4	M4 set screw	3.5° max.	0.3mm max.	0.5mm max.
	<a href="#">MCGL25-10-952</a>	<-->	TRD-GK		10	9.52 (0.375")	25	30.2	9	4	M4 set screw	3.5° max.	0.3mm max.	0.5mm max.

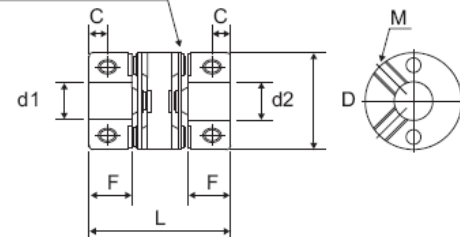
\* mm÷25.4 = inches

### GJ-6D, GJ-8D, and GJ-10D fiberglass couplings

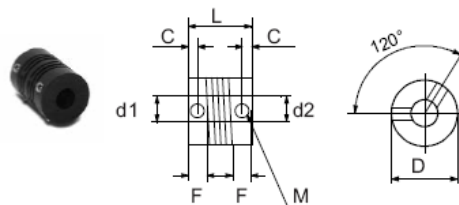


### MCGLxx aluminum couplings

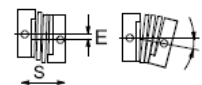
Bent plate: polyimide



### RS-075D, RS-100D, and JU-100D aluminum couplings



### Misalignment compensation



# ANEXO F. BROCHURE DEL MICRONTROLADOR

Flexis™ Microcontroller Series



## MCF51QE128 32-bit Fact Sheet



### Target Applications

- HVAC building and control systems
- Health care monitoring and instrumentation
- Fire/security control and monitoring systems
- Factory and automation systems
- Measurement equipment
- Hand-held medical/industrial applications
- Low-power industrial applications

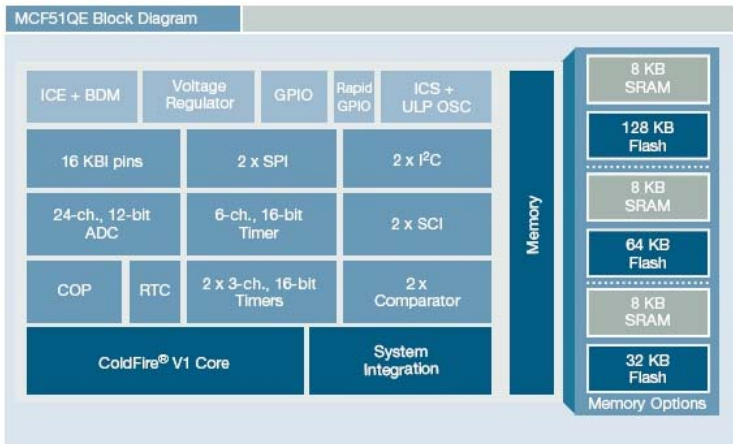
### Overview

The Flexis™ series of controllers is the connection point on the Freescale Controller Continuum, where 8- and 32-bit compatibility becomes reality. The Flexis series includes complementary families of 8-bit S08 and 32-bit ColdFire® V1 microcontrollers that share a common set of peripherals and development tools to deliver the ultimate in migration flexibility.

The QE family, comprised of a pin-compatible 8-bit and 32-bit device duo, is the first family in the Flexis series.

The 32-bit MCF51QE128 device extends the low end of the ColdFire embedded controller family with up to 128 KB flash memory and a 12-bit analog to digital converter (ADC) with up to 24 channels. The MCF51QE128 includes up to 3.6V supply voltage, a 50 MHz CPU core and three timers for improved motor control—perfect for medical devices such as health care monitoring instrumentation and industrial equipment including networked smoke detectors and security cameras.

The 32-bit MCF51QE128 is pin-, peripheral- and tool-compatible with the 8-bit S08QE128 device, providing unprecedented design freedom across the performance spectrum.



Features	Benefits
<b>32-Bit ColdFire V1 Central Processing Unit (CPU)</b> <ul style="list-style-type: none"> <li>• Up to 50 MHz ColdFire V1 core from 2.1V to 3.6V, and 20 MHz CPU at 1.8V to 2.1V across temperature range of -40°C to +85°C</li> </ul>	<ul style="list-style-type: none"> <li>• Offers high performance, even at low voltage levels for battery operated applications</li> <li>• Provides bus speed operation of 25.117 MHz from 2.1V to 3.6V and 10 MHz from 1.8 to 2.1V</li> </ul>
<ul style="list-style-type: none"> <li>• ColdFire Instruction Set Revision C (ISA_C)</li> </ul>	<ul style="list-style-type: none"> <li>• Provides additional instructions for easy handling of 8-bit and 16-bit data</li> </ul>
<ul style="list-style-type: none"> <li>• Support for up to 256 interrupt/reset sources</li> </ul>	<ul style="list-style-type: none"> <li>• Allows for software flexibility and optimization for real-time applications</li> </ul>
<b>On-Chip Memory</b> <ul style="list-style-type: none"> <li>• Up to 128 KB flash read/program/erase over full operating voltage and temperature</li> <li>• Up to 8 KB random-access memory (RAM)</li> </ul>	<ul style="list-style-type: none"> <li>• Security circuitry prevents unauthorized access to RAM and flash contents to reduce system power consumption</li> </ul>
<b>Power-Saving Modes</b> <ul style="list-style-type: none"> <li>• Two ultra-low-power (ULP) stop modes, one of which allows limited use of peripherals</li> <li>• New ULP power wait mode</li> <li>• 6 μs typical wake up time from stop3 mode</li> </ul>	<ul style="list-style-type: none"> <li>• Allows continued application sampling in a reduced power state which extends the battery life</li> </ul>
<ul style="list-style-type: none"> <li>• Internal clock Source (ICS) Module containing a frequency locked-loop (FLL) controlled by internal or external reference</li> <li>• Oscillator (OSC) Loop-control Pierce oscillator; crystal or ceramic resonator range of 31.25 kHz to 38.4 kHz or 1 MHz to 16 MHz</li> </ul>	<ul style="list-style-type: none"> <li>• Eliminates use of an external clock source. This ultimately reduces system costs associated with development</li> <li>• Includes ultra-low-power OSC for accurate timebase in low-power modes</li> </ul>

Features		Benefits
<b>Peripherals</b>		
<ul style="list-style-type: none"> <li>Two analog comparators with option to compare to an internal reference—output can be optionally routed to timer/pulse width modulator (PWM) as input capture trigger</li> </ul>	<ul style="list-style-type: none"> <li>Requires only single pin for input signal, freeing up additional pins for other use</li> <li>Allows other components in system to see result of comparator with minimal delay</li> <li>Can be used for single slope ADC and RC time constant measurements</li> </ul>	
<ul style="list-style-type: none"> <li>Analog Digital Converter (ADC) up to 24-channel, 12-bit resolution; 2.5 <math>\mu</math>s conversion time; automatic compare function; 1.7 mV/°C temperature sensor; internal bandgap reference channel; operation in stop3</li> </ul>	<ul style="list-style-type: none"> <li>Having 24 channels allows up to 24 analog devices to be sampled at extremely high speeds. Provides functionality across operational voltage of the MCU</li> </ul>	
<ul style="list-style-type: none"> <li>2x Serial Communications Interface (SCI)—Two modules offering asynchronous communications, 13-bit break option, flexible baud rate generator, double buffered transmit and receive and optional H/W parity checking and generation</li> </ul>	<ul style="list-style-type: none"> <li>Provides standard UART communications peripheral</li> <li>Allows full-duplex, asynchronous, NRZ serial communication between MCU and remote devices</li> <li>Edge interrupt can wake up MCU from low-power mode</li> </ul>	
<ul style="list-style-type: none"> <li>2x SCI (Serial Peripheral Interfaces)—Two modules with full-duplex or single-wire bidirectional; double-buffered transmit and receive; master or slave mode; MSB-first or LSB-first shifting</li> </ul>	<ul style="list-style-type: none"> <li>Having two SPI provides dedication to two separate devices. An example would be to have one SPI dedicated to a ZigBee®-ready transceiver, and the other for MCUs or peripherals</li> </ul>	
<ul style="list-style-type: none"> <li>Time pulse-width modulation (TPM) one 6-channel (TMP3) and two 3-channel (TPM1 and TPM2); selectable input capture, output compare, or buffered edge- or center-aligned PWM on each channel</li> </ul>	<ul style="list-style-type: none"> <li>Three TPMs allow for three different time bases, with a total of twelve timer channels</li> </ul>	
<ul style="list-style-type: none"> <li>Two IFCs with; Up to 100 kbps with maximum bus loading; multi-master operation; programmable slave address; interrupt-driven byte-by-byte data transfer; supports broadcast mode and 10-bit addressing</li> </ul>	<ul style="list-style-type: none"> <li>Two IFC ports enable increased system memory by using an additional IFC EEPROM. This also creates an opportunity to add an additional IFC device</li> </ul>	
<b>Input/Output</b>		
<ul style="list-style-type: none"> <li>16 bits of Rapid General Purpose Input/Output (RGPIO) connected to the CPU's high-speed local bus with set, clear and toggle functionality</li> <li>70 GPIO (General Purpose Input/Output), one input-only and one output-only pin</li> </ul>	<ul style="list-style-type: none"> <li>Results in large number of flexible I/O pins that allow developers to easily interface device into their own designs</li> </ul>	
<ul style="list-style-type: none"> <li>16 Keyboard Interrupts (KBI) pins with selectable polarity</li> </ul>	<ul style="list-style-type: none"> <li>Can be used for reading input from a keypad or used as general pin interrupts</li> </ul>	
<b>System Protection</b>		
<ul style="list-style-type: none"> <li>Watchdog computer operating properly (COP) reset with option to run from dedicated 1 kHz internal clock source or bus clock</li> </ul>	<ul style="list-style-type: none"> <li>Allows device to recognize runaway code (infinite loops) and resets processor to avoid lock-up states</li> </ul>	
<ul style="list-style-type: none"> <li>Low-voltage detection with reset or interrupt; selectable trip points</li> </ul>	<ul style="list-style-type: none"> <li>Alarms the developer of voltage drops outside of the typical operating range</li> </ul>	
<ul style="list-style-type: none"> <li>Illegal op code detection with reset</li> </ul>	<ul style="list-style-type: none"> <li>Allows the device to recognize erroneous code and resets the processor to avoid lock-up states</li> </ul>	
<ul style="list-style-type: none"> <li>Flash block protection</li> </ul>	<ul style="list-style-type: none"> <li>Prevents unintentional programming of protected flash memory, which greatly reduces the chance of losing vital system code for vendor applications</li> </ul>	
<b>Development Support</b>		
<ul style="list-style-type: none"> <li>Classic ColdFire Debug B+ functionality mapped into a single-pin BDM interface</li> </ul>	<ul style="list-style-type: none"> <li>Allows developers to use the same hardware cables between S08 and ColdFire V1 platforms</li> </ul>	
<ul style="list-style-type: none"> <li>Real-time debug support</li> </ul>	<ul style="list-style-type: none"> <li>Six hardware breakpoints which can be configured into a 1- or 2-level trigger with a programmable response (CPU halt or interrupt)</li> </ul>	
<ul style="list-style-type: none"> <li>Program trace support</li> </ul>	<ul style="list-style-type: none"> <li>Capture of processor status and debug data into on-chip trace buffer provides program trace capabilities and programmable start/stop recording conditions</li> </ul>	

### Package Options

Part Number	Temp. Range	Package
MCF51QE128CLK	-40°C to +85°C	80 LQFP
MCF51QE128CLH	-40°C to +85°C	64 LQFP
MCF51QE64CLH	-40°C to +85°C	64 LQFP
MCF51QE32CLH	-40°C to +85°C	64 LQFP
MCF51QE32LH	0°C to +70°C	64 LQFP

### Cost-Effective Development Tools

#### DEMOQE128

\$99\*

Cost-effective demonstration kit, including the S08 and ColdFire® V1 daughter cards, as well as a serial port and built-in USB-BDM cable for debugging and programming.

#### EVBQE128

\$325\*

Full-featured evaluation system for the QE128 device family. This evaluation system enables full evaluation of both the MC9S08QE128 and MCF51QE128 devices.

#### CodeWarrior® Development Studio for Microcontrollers 6.0

Complimentary\*\* Special Edition

CodeWarrior Development Studio for Microcontrollers is a single tool suite that supports software development for Freescale's 8-bit and 32-bit ColdFire V1 microcontrollers. Designers can further accelerate application development with the help of Processor Expert, an award-winning rapid application development tool integrated into the CodeWarrior tool suite.

\* Prices indicated are MSRP

\*\* Subject to license agreement

**Learn More:** For more information about the Flexis QE family, please visit [www.freescale.com/flexis](http://www.freescale.com/flexis).

Freescale and the Freescale logo are trademarks or registered trademarks of Freescale Semiconductor, Inc. in the U.S. and other countries. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2009.

Document Number: MCF51QE128FS  
REV 1



## ANEXO G. CODIGO FUENTE DISEÑO DEL CONTROLADOR EN MATLAB

```
%%% Scribd: ANGULO DE ELEVACION %%%

% parámetros seleccionados
K=1.13; %ganacia de estado estable
E=0.11; %coeficiente amortiguamiento
Wn=0.58; %frecuencia natural

%funcion de transferencia del angulo de elvacion
num=[K*(Wn^2)];
den=[1 2*E*Wn Wn^2];
Elevacion=tf(num,den);

%Sistema realimentado sin compensar
srsc=feedback(Elevacion,1);
step(srsc,'r');grid
[A,B,C,D] = tf2ss(num,den);

s=tf('s');

%Analisis del control Proporcional
for Kp=[2:2:6], LA1=feedback(Kp*Elevacion,1);
    figure(1); step(LA1); hold on
    figure(11); impulse(LA1); hold on
end

%Analisis de la accion integral en un controlador PI
for Ki=[5:5:25]; Kp=6;
    PI=(Kp+(Ki/s)); LA5=feedback(Elevacion*PI,1);
    figure(6); step (LA5); hold on
end

%Analisis de la accion derivativa en un controlador PD
for Kd=[3:3:15]; Kp=6;
    PD=((Kd*s)+Kp); LA2=feedback(Elevacion*PD,1);
    figure(2); step(LA2), hold on
end

%Analisis de la accion integral en un control PID
for Ki=[3:3:15]; Kd=8; Kp=10;
    PID=(Kp+(Kd*s)+(Ki/s)); LA3=feedback(Elevacion*PID,1);
    figure(3); step(LA3); hold on
end

%Analisis del controlador PID sin utilizar algoritmo de sintonización
```

```

for Ki=5; Kp=15; Kd=10;
    PID=(Kp+(Kd*s)+(Ki/s));
    LApid_e=feedback(Elevacion*PID,1);
    PD=((Kd*s)+Kp); LA4=feedback(Elevacion*PD,1);
    figure (4); step(LApid_e)
    figure (5); step(LA4)
end

%compensador PID con filtro derivativo (Ziegler-nichols)
Cfd=tf([63.53 65.86 14.71],[1 10 0]);
    LAfd=feedback(Elevacion*Cfd,1); %lazo abierto filtro derivativo

%Compensador PID (Ziegler-Nichols)
Cpid=tf([3.923 4.747 1.302],[1 0]);
    LApid_zn=feedback(Elevacion*Cpid,1); %Lazo abierto pid

%Análisis del controlador amortiguado
for Kp=6; Ti=(2*E/Wn);
    Td=(1/(2*E*Wn)); B=(E/(5*((1-(2*(E^2)))^(1/2)))));
    N=1/B; Tf=Td/N;
    H=Kp*(1+(1/(Ti*s))+((Td*s)/((Tf*s)+1)));
    Lamortiguado=feedback(Elevacion*H,1);
    %figure (6); step(Lamortiguado);grid
end

t=0:0.01:14;
figure;step(LApid_e,'g',LApid_zn,'r',LAfd,'b'); grid
figure;impulse(LApid_e,'g',LApid_zn,'r',LAfd,'b'); grid

%Generamos una onda cuadrada, que va a ser la señal de
%entrada, con un periodo de 11, y n tiempo de muestreo de
%0.01 segundos
[u,t] = gensig('square',11); %,5e-3,(1e-3)/100);

%Ecuación del sistema retroalimentado compensado
%src=Compensador*Planta/(1+Compensador*Planta);
src=feedback(Elevacion*Cfd,1);

%Ecuación del sistema retroalimentado no compensado
%srnc=(Planta/(1+Planta));
srnc=feedback(Elevacion,1);

%Señal de error
s_error=1-src;

%Graficamos todas las señales

```

```

subplot(2,2,1),plot(t,u);
axis([0 33 -0.2 1.2]);
    title('Señal de entrada'),grid;
subplot(2,2,2),lsim(srnc,u,t);
axis([0 33 -0.6 1.2]);
    title('Salida sin compensar'),grid;
subplot(2,2,3),lsim(src,u,t);
axis([0 33 -0.5 1.5]);
    title('Salida compensada'),grid;
subplot(2,2,4),lsim(s_error,u,t);
axis([0 33 -1.2 1.2]);
    title('Señal de error'),grid;

```

```

%%% Scribd: ANGULO DE ALABEO %%%

```

```

K=0.2; %ganancia de estado estable
E=0.068; %coeficiente amortiguamiento
Wn=2.1; %frecuencia natural

```

```

% parámetros seleccionados
num=[10*K*(Wn^2)];
den=[1 2*E*Wn Wn^2];
Cabeceo=tf(num,den);

```

```

%Sistema realimentado sin compensar
srsc=feedback(Cabeceo,1);
step(srsc,'g');grid
[A,B,C,D] = tf2ss(num,den);

```

```

s=tf('s');

```

```

%Análisis del control Proporcional
for Kp=[10:10:40], LA1=feedback(Kp*Cabeceo,1);
    figure(2); step(LA1); hold on
    figure(22); impulse(LA1); hold on
end

```

```

%Análisis de la acción derivativa en un controlador PD
for Kd=[4:2:12]; Kp=10;
    PD=((Kd*s)+Kp); LA2=feedback(Cabeceo*PD,1);
    figure(2); step(LA2), hold on
    figure(21); impulse(LA2), hold on
end

```

```

%Análisis de la acción integral en un control PID
for Ki=[8:2:16]; Kd=3; Kp=6;

```

```

    PID=(Kp+(Kd*s)+(Ki/s)); LA3=feedback(Cabeceo*PID,1);
    figure(3); step(LA3); hold on
    figure(33); impulse(LA3); hold on
end

%Análisis del controlador PID sin utilizar algoritmo de sintonización
for Ki=1.5; Kp=2.8; Kd=0.9;
    PID=(Kp+(Kd*s)+(Ki/s));
    LApid_e=feedback(Cabeceo*PID,1);
    PD=((Kd*s)+Kp); LA4=feedback(Cabeceo*PD,1);
    figure (4); step(LApid_e)
    figure (5); step(LA4)
end

%compensador PID con filtro derivativo (Ziegler-nichols)
Cfd=tf([41.77 94.31 48.74],[1 35.51 0]);
    LApid_zn=feedback(Cabeceo*Cpid,1);

%Compensador PID (Ziegler-Nichols)
Cpid=tf([0.636 1.952 1.367],[1 0]);
    LAfd=feedback(Cabeceo*Cfd,1); %lazo abierto filtro derivativo

t=0:0.01:14;
figure;step(LApid_e,'g',LApid_zn,'r',LAfd,'b'); grid
%figure;impulse(LApid_e,'g',LApid_zn,'r',LAfd,'b'); grid

%Generamos una onda cuadrada, que va a ser la señal de
%entrada, con un periodo de, y con tiempo de muestreo de
%0.01 segundos.
[u,t] = gensig('square',3);

%Ecuación del sistema retroalimentado compensado
%src=Compensador*Planta/(1+Compensador*Planta);
src=feedback(Cabeceo*Cfd,1);

%Ecuación del sistema retroalimentado no compensado
%srnc=(Planta/(1+Planta));
srnc=feedback(Cabeceo,1);

%Señal de error
s_error=1-src;

%Graficamos todas las señales
subplot(2,2,1),plot(t,u);
axis([0 9 -0.2 1.2]);
    title('Señal de entrada'),grid;

```

```
subplot(2,2,2),lsim(srnc,u,t);
axis([0 9 -0.6 1.4]);
    title('Salida sin compensar'),grid;
subplot(2,2,3),lsim(src,u,t);
axis([0 9 -0.5 1.5]);
    title('Salida compensada'),grid;
subplot(2,2,4),lsim(s_error,u,t);
axis([0 9 -1.2 1.2]);
    title('Señal de error'),grid;
```

## ANEXO H. CODIGO FUENTE DE LA INTERFAZ EN MATLAB

```
%-----Parámetros de la interfaz-----
function varargout = visualizador(varargin)

%inicialización de código, no editar
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @visualizador_OpeningFcn, ...
                  'gui_OutputFcn',  @visualizador_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% ---se ejecuta despues de que la interfaz se muestra
function visualizador_OpeningFcn(hObject, eventdata, handles, varargin)

% Choose default command line output for visualizador
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = visualizador_OutputFcn(hObject, eventdata, handles)
% Get default command line output from handles structure
varargout{1} = handles.output;

%-----Funciones de Llamdo-----

function inicia_com_Callback(hObject, eventdata, handles)

if(get(hObject, 'value') == 1)
    set(hObject, 'String', 'Enlazado');
    set(handles.com, 'Enable', 'inactive');
else
    set(hObject, 'String', 'Enlazar');
end
```

```

function para_com_Callback(hObject, eventdata, handles)

function valor_i_pitch_Callback(hObject, eventdata, handles)

function valor_d_pitch_Callback(hObject, eventdata, handles)

function valor_p_pitch_Callback(hObject, eventdata, handles)

function p_3_p_Callback(hObject, eventdata, handles)

function d_3_p_Callback(hObject, eventdata, handles)

function Salir_menu_Callback(hObject, eventdata, handles)

function envia_datos_Callback(hObject, eventdata, handles)
    if (get(handles.inicia_com, 'value') == 0)
        msgbox('No se establecido la comunicación.');
```

else
 contador=str2double(get(handles.contador, 'string'));
 contador=contador+1;
 switch(contador);
 case 1

```

set(handles.p_1_p, 'string', get(handles.valor_p_pitch, 'string'));
set(handles.i_1_p, 'string', get(handles.valor_i_pitch, 'string'));
set(handles.d_1_p, 'string', get(handles.valor_d_pitch, 'string'));

set(handles.p_1_r, 'string', get(handles.valor_p_roll, 'string'));
set(handles.i_1_r, 'string', get(handles.valor_i_roll, 'string'));
set(handles.d_1_r, 'string', get(handles.valor_d_roll, 'string'));
            case 2

```

```

set(handles.p_2_p, 'string', get(handles.valor_p_pitch, 'string'));
set(handles.i_2_p, 'string', get(handles.valor_i_pitch, 'string'));
set(handles.d_2_p, 'string', get(handles.valor_d_pitch, 'string'));

set(handles.p_2_r, 'string', get(handles.valor_p_roll, 'string'));
set(handles.i_2_r, 'string', get(handles.valor_i_roll, 'string'));
set(handles.d_2_r, 'string', get(handles.valor_d_roll, 'string'));

```

```

        case 3

set(handles.p_3_p, 'string', get(handles.valor_p_pitch, 'string'));
set(handles.i_3_p, 'string', get(handles.valor_i_pitch, 'string'));
set(handles.d_3_p, 'string', get(handles.valor_d_pitch, 'string'));

set(handles.p_3_r, 'string', get(handles.valor_p_roll, 'string'));
set(handles.i_3_r, 'string', get(handles.valor_i_roll, 'string'));
set(handles.d_3_r, 'string', get(handles.valor_d_roll, 'string'));

        otherwise
            msgbox('No es posible enviar mas opciones.');
```

end

```

set(handles.contador, 'string', num2str(contador));
if(contador < 4)
    %-----CONFIGURACIÓN PUERTO-----
    ncom=get(handles.com, 'String');
    s=serial(['COM', ncom]);
    set(s, 'BaudRate', 115200);
    set(s, 'StopBits', 1);
    set(s, 'DataBits', 8);
    set(s, 'Parity', 'none');
    set(s, 'Terminator', ',');
    set(s, 'requesttosend', 'off');
    set(s, 'InputBufferSize', 10);
    set(s, 'timeout', 5);
    %-----
    set(handles.exit, 'String', '0');
    %-----CAPTURA DE DATOS-----
    sp_p=get(handles.setpoint_p, 'String');
    sp_r=get(handles.setpoint_r, 'String');

    v_p_p=get(handles.valor_p_pitch, 'String');
    v_i_p=get(handles.valor_i_pitch, 'String');
    v_d_p=get(handles.valor_d_pitch, 'String');

    v_p_r=get(handles.valor_p_roll, 'String');
    v_i_r=get(handles.valor_i_roll, 'String');
    v_d_r=get(handles.valor_d_roll, 'String');
    t_max=get(handles.tmax, 'String');
    %-----INICIALIZACION DE VARIABLES PARA RECEPCÓN-----
    -----
    t1=[];
    p1=[];
    r1=[];
    t=[];

```

```

p=[];
r=[];
%-----ENVÍO-----

trama=['s', '~', num2str(str2num(t_max)+1), '~', sp_p, '~', sp_r, '~', v_p_p, '~',
v_i_p, '~', v_d_p, '~', v_p_r, '~', v_i_r, '~', v_d_r, '~', 'f'];
tam=whos('trama');
set(s, 'outputBufferSize', tam.bytes);
fopen(s);
fprintf(s, '%s', trama);           %envío de configuracion.
%fclose(s);
%-----CÁLCULO NUMERICO DE VALORES PARA GRAFICAS-----
-----

sp_p=str2num(sp_p);
sp_r=str2num(sp_r);
muestras=str2num(t_max)/10e-3;
limite=muestras/100;
%-----RECEPCION DE DATOS-----
%Formato de datos de entrada=[it,p,r]
%fopen(s);
for j=1:limite
    for i=1:100
        if(get(handles.exit, 'String')== '0')
            while(fscanf(s, '%c', 1) ~= 'i')
                end
                t1=[t1 fscanf(s, '%c')];
                p1=[p1 fscanf(s, '%c')];
                r1=[r1 fscanf(s, '%c')];
            else
                break;
            end
        end
        if(get(handles.exit, 'String')== '1')
            break;
        end
        t=[t 0.01*str2num(t1)];
        p=[p 0.072*str2num(p1)];
        r=[r 0.072*str2num(r1)];

        t1=[];
        p1=[];
        r1=[];

        axes(handles.axes7);
        plot(t,p, 'b');
        axes(handles.axes6);
        plot(t,r, 'r');
    end

save tiempo.dat t -ASCII;
save valor_p.dat p -ASCII;

```

```

save valor_r.dat r -ASCII;

fclose(s);
delete(s);
clear s;
if(get(handles.exit,'String')== '0')
    axes(handles.axes7);
    hold on;
    plot(t,sp_p,'r');
    set(handles.axes7,'XGrid','on');
    set(handles.axes7,'YGrid','on');
    hold off
    axes(handles.axes6);
    hold on;
    plot(t,sp_r,'b');
    set(handles.axes6,'XGrid','on');
    set(handles.axes6,'YGrid','on');
    hold off
end
set(handles.exit,'String','0');
end
end

function setpoint_r_Callback(hObject, eventdata, handles)
    if (get(handles.inicia_com,'value') == 0)
        set(hObject,'String','0');
        msgbox('No se establecido la comunicación.');
```

```

    else
        if(str2num(get(hObject,'String')) < -45 |
str2num(get(hObject,'String'))> 45)
            msgbox('No se puede establecer setpoint de pitch fuera del
rango [-45,45]');
```

```

            set(hObject,'String','0');
        else
            cla(handles.axes6,'reset');
            plot(handles.axes6,
0:0.5:500,str2double(get(hObject,'string')),'-');
            set(handles.axes6,'Xgrid','on');
            set(handles.axes6,'Ygrid','on');
```

```

        end
    end
end

function setpoint_p_Callback(hObject, eventdata, handles)
    if (get(handles.inicia_com,'value') == 0)
        set(hObject,'String','0');
        msgbox('No se establecido la comunicación.');
```

```

    else
        if(str2num(get(hObject,'String')) < 0 |
str2num(get(hObject,'String'))> 45)

```

```

        msgbox('No se puede establecer setpoint de pitch fuera del
rango [0,45]');
        set(hObject,'String','0');
    else
        cla(handles.axes7,'reset');
        plot(handles.axes7,
0:0.5:500,str2double(get(hObject,'string')),'-');
        set(handles.axes7,'Xgrid','on');
        set(handles.axes7,'Ygrid','on');
    end
end

function fijar_sp_r_Callback(hObject, eventdata, handles)
    if (get(handles.inicia_com,'value') == 0)
        msgbox('No se establecido la comunicación.');
```

```

    else
        if(str2num(get(handles.setpoint_r,'String')) < -45 |
str2num(get(handles.setpoint_r,'String'))> 45)
            msgbox('No se puede establecer setpoint de roll fuera del
rango [-45,45]');
```

```

            set(handles.setpoint_r,'String','0');
        else
            sp=get(handles.setpoint_r,'string');
            cla(handles.axes6,'reset');
            plot(handles.axes6,0:0.5:500,str2double(sp),'-');
            set(handles.axes6,'Xgrid','on');
            set(handles.axes6,'Ygrid','on');
        end
    end
end

function fijar_sp_p_Callback(hObject, eventdata, handles)

    if (get(handles.inicia_com,'value') == 0)
        msgbox('No se establecido la comunicación.');
```

```

    else
        if(str2num(get(handles.setpoint_p,'String')) < 0 |
str2num(get(handles.setpoint_p,'String'))> 45)
            msgbox('No se puede establecer setpoint de pitch fuera del
rango [0,45]');
```

```

            set(handles.setpoint_r,'String','0');
        else
            sp=get(handles.setpoint_p,'string');
            cla(handles.axes7,'reset');
            plot(handles.axes7, 0:0.5:500,str2double(sp),'-');
            set(handles.axes7,'Xgrid','on');
            set(handles.axes7,'Ygrid','on');
        end
    end
end

function reinicia_com_Callback(hObject, eventdata, handles)

```

```
set(handles.exit, 'String', '1');
set(handles.contador, 'String', '0');

set(handles.tmax, 'String', '60');

set(handles.valor_p_pitch, 'String', '0');
set(handles.valor_i_pitch, 'String', '0');
set(handles.valor_d_pitch, 'String', '0');

set(handles.valor_p_roll, 'String', '0');
set(handles.valor_i_roll, 'String', '0');
set(handles.valor_d_roll, 'String', '0');

set(handles.setpoint_p, 'String', '0');
axes(handles.axes7);
cla(handles.axes7, 'reset');
plot(0:2:10,0, '-');
set(handles.axes7, 'XGrid', 'on');
set(handles.axes7, 'YGrid', 'on');

set(handles.setpoint_r, 'String', '0');
axes(handles.axes6);
cla(handles.axes6, 'reset');
plot(0:2:10,0, '-');
set(handles.axes6, 'XGrid', 'on');
set(handles.axes6, 'YGrid', 'on');

set(handles.inicia_com, 'value', 0);
set(handles.inicia_com, 'String', 'Enlazar');

set(handles.p_1_p, 'String', '0');
set(handles.i_1_p, 'String', '0');
set(handles.d_1_p, 'String', '0');

set(handles.p_1_r, 'String', '0');
set(handles.i_1_r, 'String', '0');
set(handles.d_1_r, 'String', '0');

set(handles.p_2_p, 'String', '0');
set(handles.i_2_p, 'String', '0');
set(handles.d_2_p, 'String', '0');

set(handles.p_2_r, 'String', '0');
set(handles.i_2_r, 'String', '0');
set(handles.d_2_r, 'String', '0');

set(handles.p_3_p, 'String', '0');
set(handles.i_3_p, 'String', '0');
set(handles.d_3_p, 'String', '0');
```

```
    set(handles.p_3_r,'String','0');
    set(handles.i_3_r,'String','0');
    set(handles.d_3_r,'String','0');

function valor_p_roll_Callback(hObject, eventdata, handles)

function valor_i_roll_Callback(hObject, eventdata, handles)

function valor_d_roll_Callback(hObject, eventdata, handles)

function i_2_p_Callback(hObject, eventdata, handles)

function p_2_p_Callback(hObject, eventdata, handles)

function d_2_p_Callback(hObject, eventdata, handles)

function i_1_p_Callback(hObject, eventdata, handles)

function p_3_r_Callback(hObject, eventdata, handles)

function i_3_r_Callback(hObject, eventdata, handles)

function d_3_r_Callback(hObject, eventdata, handles)

function p_2_r_Callback(hObject, eventdata, handles)

function i_2_r_Callback(hObject, eventdata, handles)

function d_2_r_Callback(hObject, eventdata, handles)

function p_1_r_Callback(hObject, eventdata, handles)

function d_1_r_Callback(hObject, eventdata, handles)

function i_3_p_Callback(hObject, eventdata, handles)

function i_1_r_Callback(hObject, eventdata, handles)

function p_1_p_Callback(hObject, eventdata, handles)

function d_1_p_Callback(hObject, eventdata, handles)

function com_Callback(hObject, eventdata, handles)

    if(get(handles.inicia_com,'value') == 1)
```

```

        msgbox('No puede modificar estos parametros cuando esta
enlazado.');
```

end

```

function fijar_puerto_Callback(hObject, eventdata, handles)

function tmax_Callback(hObject, eventdata, handles)
if (get(handles.inicia_com,'value') == 0)
    set(hObject,'String','60');
    msgbox('No se establecido la comunicación.');
```

else

```

    if(str2num(get(hObject,'String')) <= 0)
        msgbox('No se puede establecer tiempo menor o igual a cero');
        set(hObject,'String','60');
```

else

```

        set(hObject,'String',round(str2num(get(hObject,'String')));
    end
end

end

%-----Funciones del menu-----
function salir_menu_Callback(hObject, eventdata, handles)
    delete(gcf)

function nuevo_Callback(hObject, eventdata, handles)
    set(handles.contador,'String','0');

    set(handles.valor_p_pitch,'String','0');
    set(handles.valor_i_pitch,'String','0');
    set(handles.valor_d_pitch,'String','0');

    set(handles.valor_p_roll,'String','0');
    set(handles.valor_i_roll,'String','0');
    set(handles.valor_d_roll,'String','0');

    set(handles.setpoint_p,'String','0');
    cla(handles.axes7,'reset');
    plot(handles.axes7,0:2:10,0,'-');
    set(handles.axes7,'XGrid','on');
    set(handles.axes7,'YGrid','on');

    set(handles.setpoint_r,'String','0');
    cla(handles.axes6,'reset');
    plot(handles.axes6,0:2:10,0,'-');
    set(handles.axes6,'XGrid','on');
    set(handles.axes6,'YGrid','on');

    set(handles.com,'String','');
    set(handles.com,'Enable','on');
```

```

set(handles.tmax, 'String', '60');

set(handles.inicia_com, 'value', 0);
set(handles.inicia_com, 'String', 'Enlazar');

set(handles.p_1_p, 'String', '0');
set(handles.i_1_p, 'String', '0');
set(handles.d_1_p, 'String', '0');

set(handles.p_1_r, 'String', '0');
set(handles.i_1_r, 'String', '0');
set(handles.d_1_r, 'String', '0');

set(handles.p_2_p, 'String', '0');
set(handles.i_2_p, 'String', '0');
set(handles.d_2_p, 'String', '0');

set(handles.p_2_r, 'String', '0');
set(handles.i_2_r, 'String', '0');
set(handles.d_2_r, 'String', '0');

set(handles.p_3_p, 'String', '0');
set(handles.i_3_p, 'String', '0');
set(handles.d_3_p, 'String', '0');

set(handles.p_3_r, 'String', '0');
set(handles.i_3_r, 'String', '0');
set(handles.d_3_r, 'String', '0');

function Untitled_1_Callback(hObject, eventdata, handles)
%-----Funciones de creaciòn de objetos de la interfaz-----
function figure1_CreateFcn(hObject, eventdata, handles)

function fondo_CreateFcn(hObject, eventdata, handles)
    img=imread('HeliBot_3DOF.BMP'); %fijaciòn de la imagen.
    image(img)
    axis off

function Valor_P_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
        set(hObject, 'BackgroundColor', 'white');
    end

function valor_i_pitch_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
        set(hObject, 'BackgroundColor', 'white');
    end

```

```

function valor_d_pitch_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

function valor_p_pitch_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

%-----
%function graficas_pitch_CreateFcn(hObject, eventdata, handles)
%    plot(0:0.5:500,1,'-');
function axes7_CreateFcn(hObject, eventdata, handles)
    plot(0:2:10,0,'-');

function axes6_CreateFcn(hObject, eventdata, handles)
    plot(0:2:10,0,'-');

function axes8_CreateFcn(hObject, eventdata, handles)
    img1=imread('LogoUIS.JPG');
    image(img1)
    axis off

function axes9_CreateFcn(hObject, eventdata, handles)
    img2=imread('ERA.BMP');
    image(img2)
    axis off

function axes10_CreateFcn(hObject, eventdata, handles)
    img3=imread('dicbot.JPG');
    image(img3)
    axis off

%-----
function tmax_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function i_1_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

function d_3_p_CreateFcn(hObject, eventdata, handles)

```

```

        if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
            set(hObject,'BackgroundColor','white');
        end

function setpoint_p_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

function p_3_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function valor_p_roll_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function valor_i_roll_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function valor_d_roll_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function setpoint_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function p_2_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function i_2_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

```

```

        set(hObject, 'BackgroundColor', 'white');
end

function d_2_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function p_3_p_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function i_1_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function p_3_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function i_3_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function d_3_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function p_2_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function i_2_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUicontrolBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

```

```
function d_2_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function p_1_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function d_1_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function d_3_p_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function i_3_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function i_1_r_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function p_1_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function d_1_p_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function com_CreateFcn(hObject, eventdata, handles)
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function fijar_sp_r_CreateFcn(hObject, eventdata, handles)

function fijar_sp_p_CreateFcn(hObject, eventdata, handles)

%-----Funciones para presión de tecla derecha del mouse-----
-

function valor_p_pitch_KeyPressFcn(hObject, eventdata, handles)

function valor_i_pitch_KeyPressFcn(hObject, eventdata, handles)

function valor_d_pitch_KeyPressFcn(hObject, eventdata, handles)

function valor_p_roll_KeyPressFcn(hObject, eventdata, handles)

function valor_i_roll_KeyPressFcn(hObject, eventdata, handles)

function valor_d_roll_KeyPressFcn(hObject, eventdata, handles)

function setpoint_p_KeyPressFcn(hObject, eventdata, handles)

function setpoint_r_KeyPressFcn(hObject, eventdata, handles)

function p_1_p_KeyPressFcn(hObject, eventdata, handles)

function i_1_p_KeyPressFcn(hObject, eventdata, handles)

function d_1_p_KeyPressFcn(hObject, eventdata, handles)

function p_2_p_KeyPressFcn(hObject, eventdata, handles)

function i_2_p_KeyPressFcn(hObject, eventdata, handles)

function d_2_p_KeyPressFcn(hObject, eventdata, handles)

function p_3_p_KeyPressFcn(hObject, eventdata, handles)

function i_3_p_KeyPressFcn(hObject, eventdata, handles)

function d_3_p_KeyPressFcn(hObject, eventdata, handles)

function p_1_r_KeyPressFcn(hObject, eventdata, handles)

```

```
function i_1_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function d_1_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function p_2_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function i_2_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function d_2_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function p_3_r_KeyPressFcn(hObject, eventdata, handles)
```

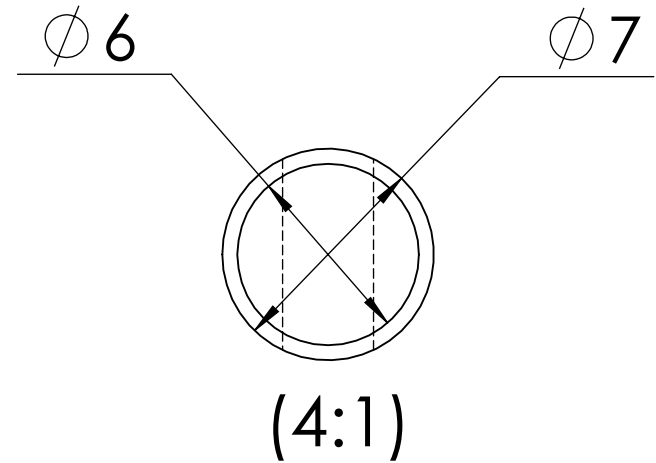
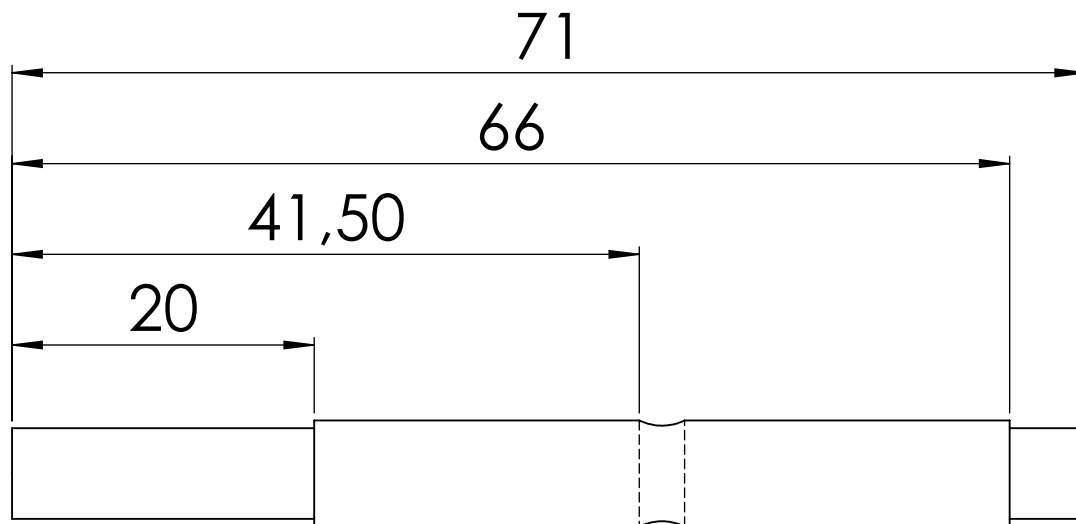
```
function i_3_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function d_3_r_KeyPressFcn(hObject, eventdata, handles)
```

```
function com_KeyPressFcn(hObject, eventdata, handles)
```

```
%-----  
function envia_datos_DeleteFcn(hObject, eventdata, handles)
```

```
function axes9_DeleteFcn(hObject, eventdata, handles)
```

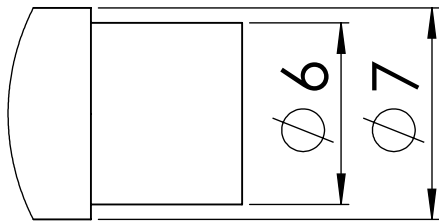
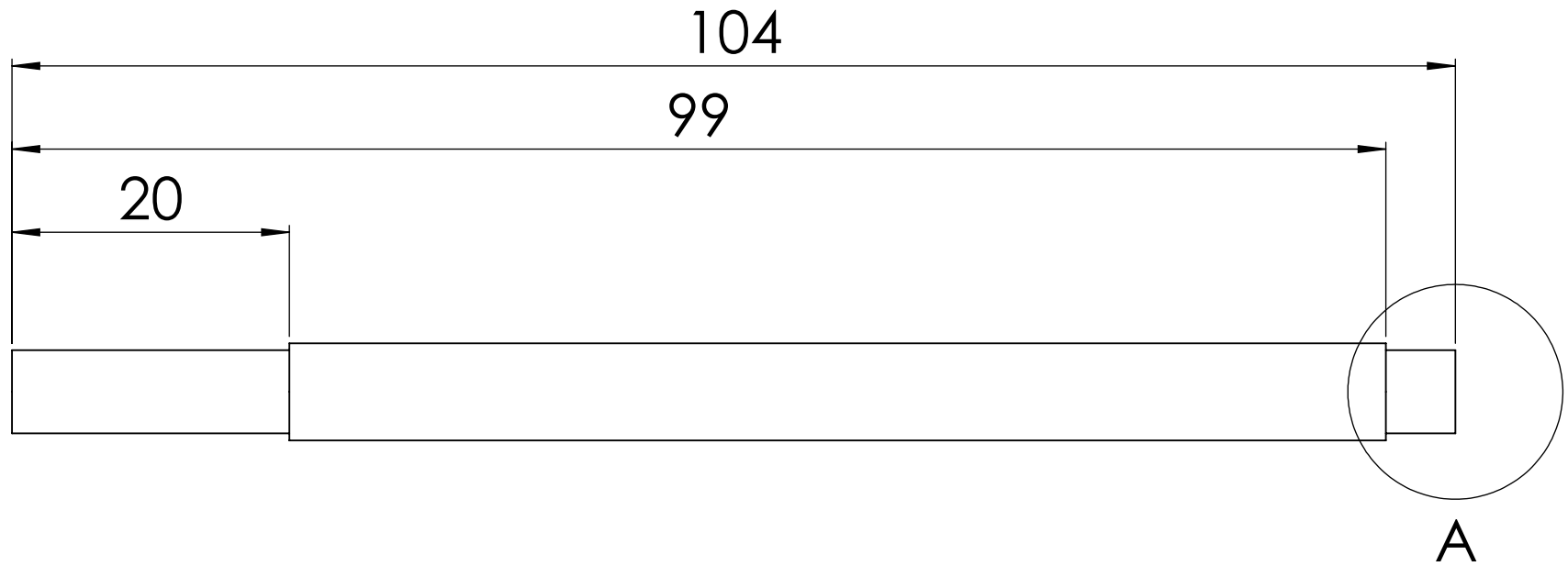


$\varnothing 3$  POR TODO



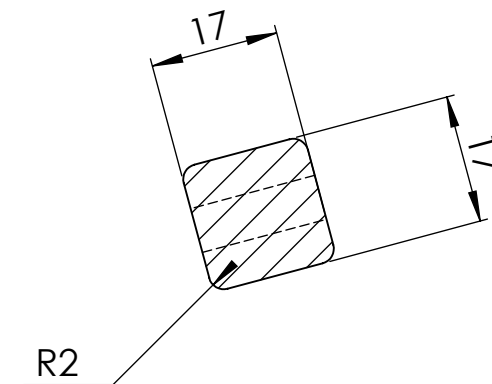
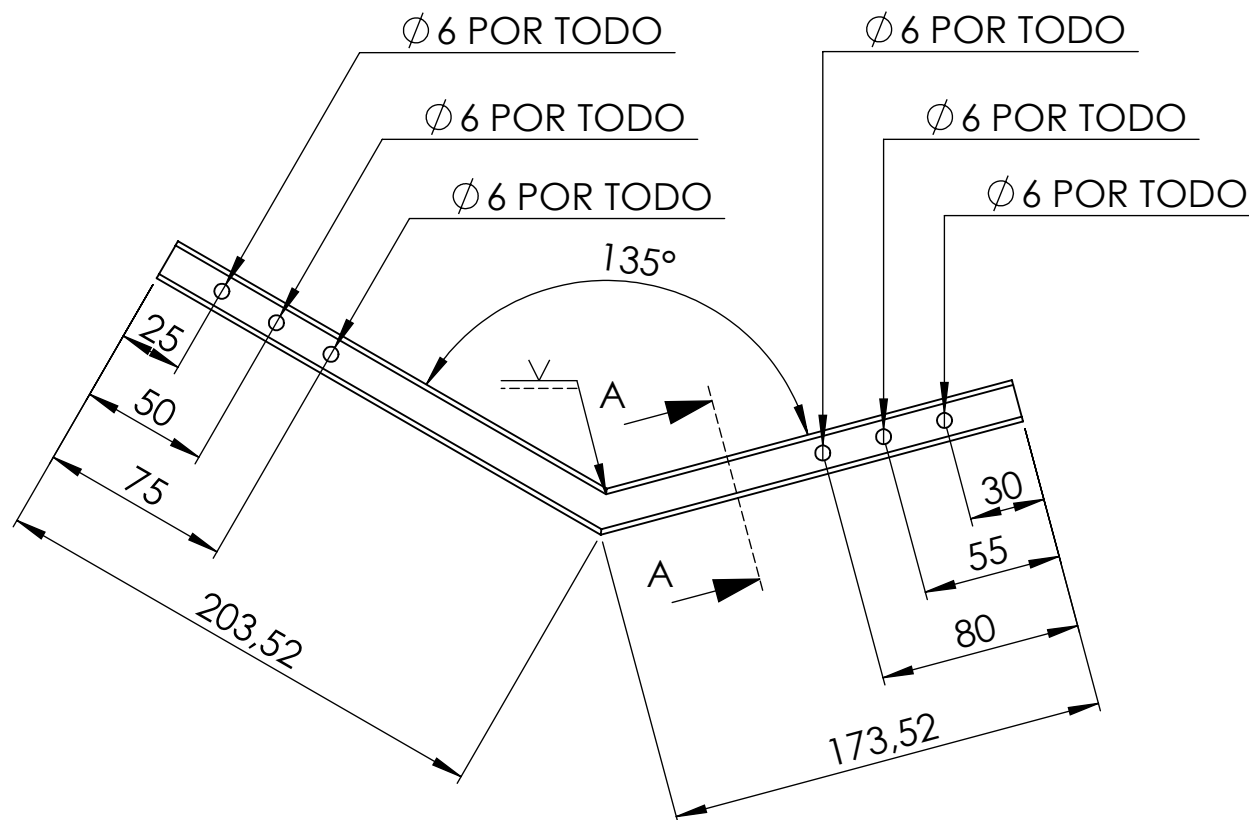
Bloque viga principal

DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque viga principal	
COMENTARIOS				PIEZA	Eje de cabeceo	No. PLANO
				MATERIAL	AISI 316L Acero inoxidable	
				ESCALA	2:1	REV.



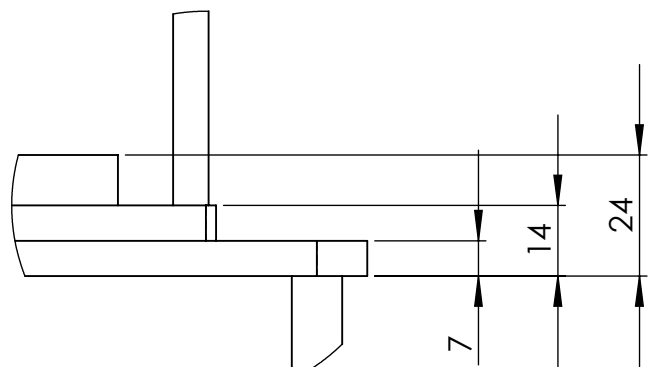
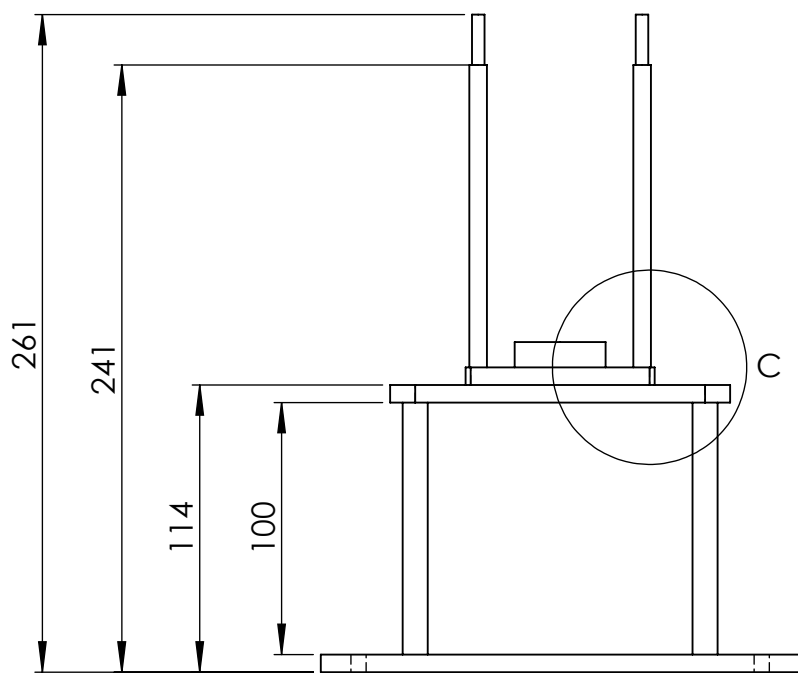
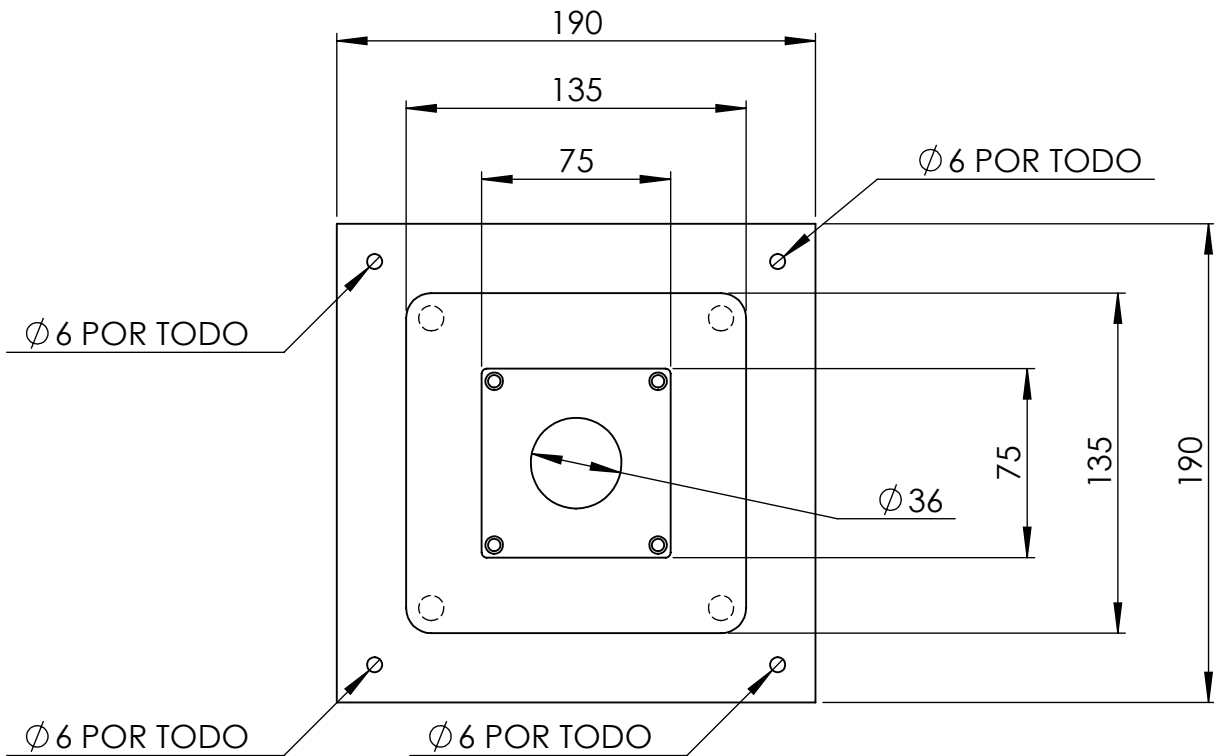
DETALLE A  
ESCALA 4 : 1

DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque central	
COMENTARIOS				PIEZA	Eje de elevación	No. PLANO
				MATERIAL	AISI 316L Acero inoxidable	
				ESCALA	2:1	REV.

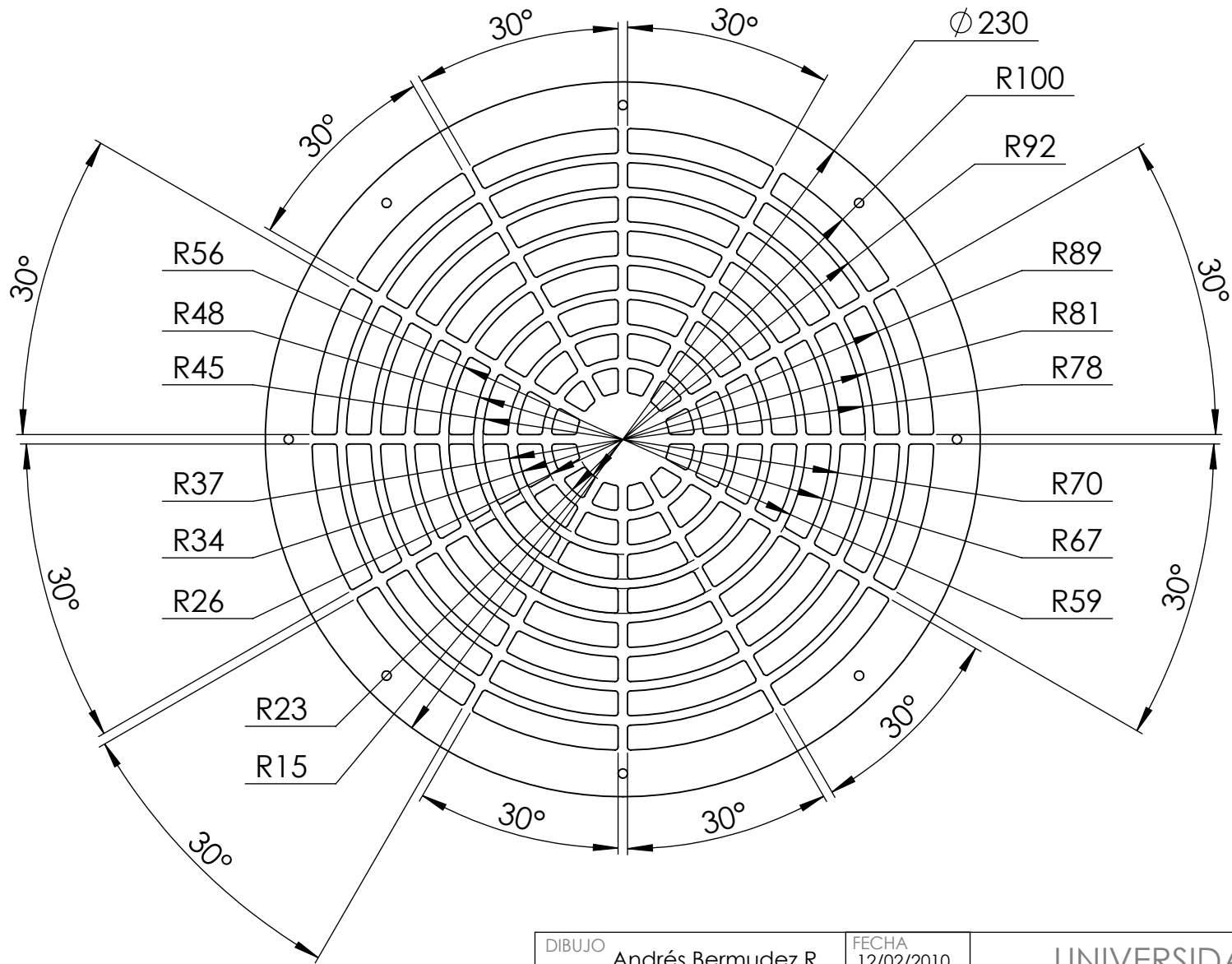


SECCIÓN A-A  
ESCALA 1 : 1

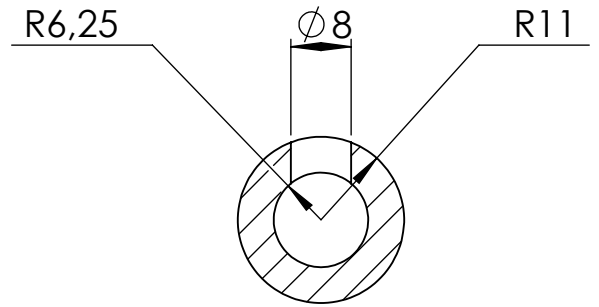
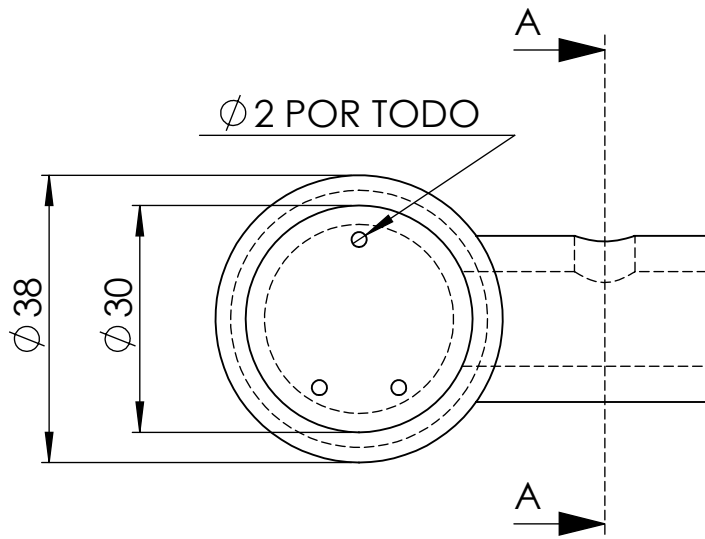
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque viga principal	
COMENTARIOS				PIEZA	Soporte contrapeso	No. PLANO
				MATERIAL	Aleación de aluminio 1060	
				ESCALA	1:3	REV.



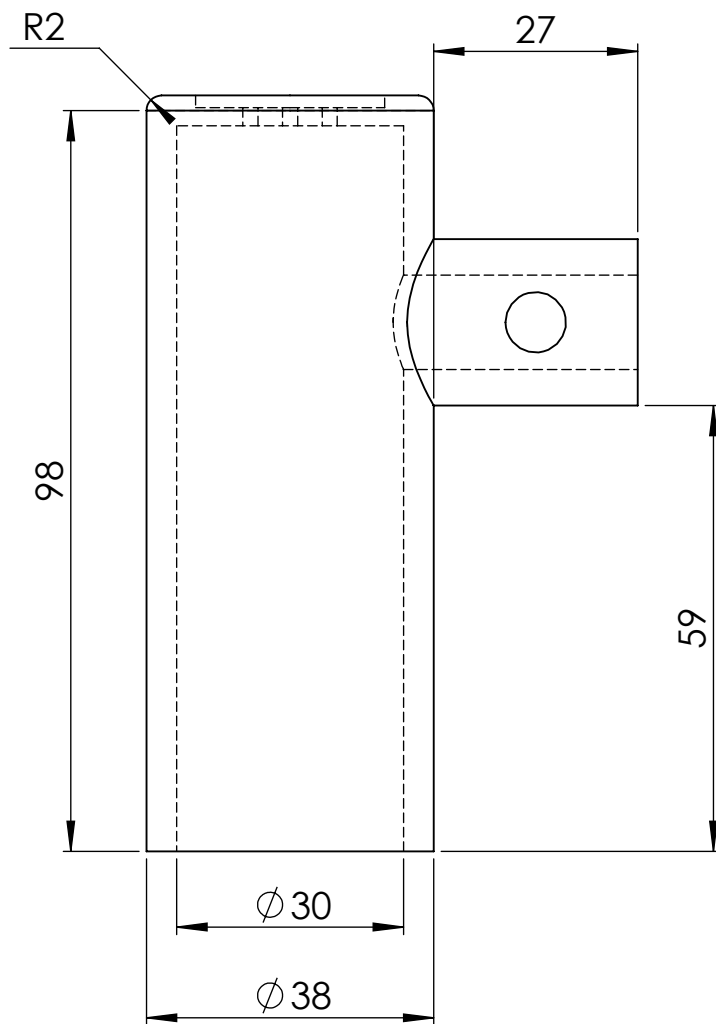
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Estructura base	
COMENTARIOS				PIEZA	Base	No. PLANO
				MATERIAL	A36 Acero	
				ESCALA	1:3	REV.



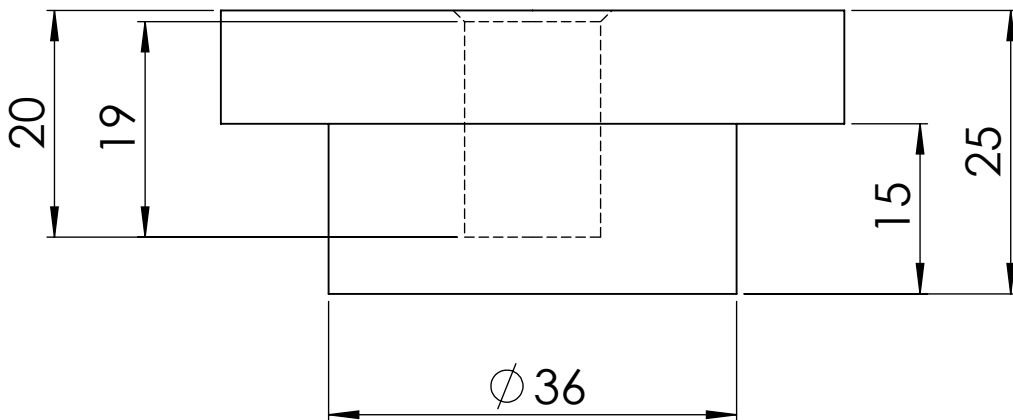
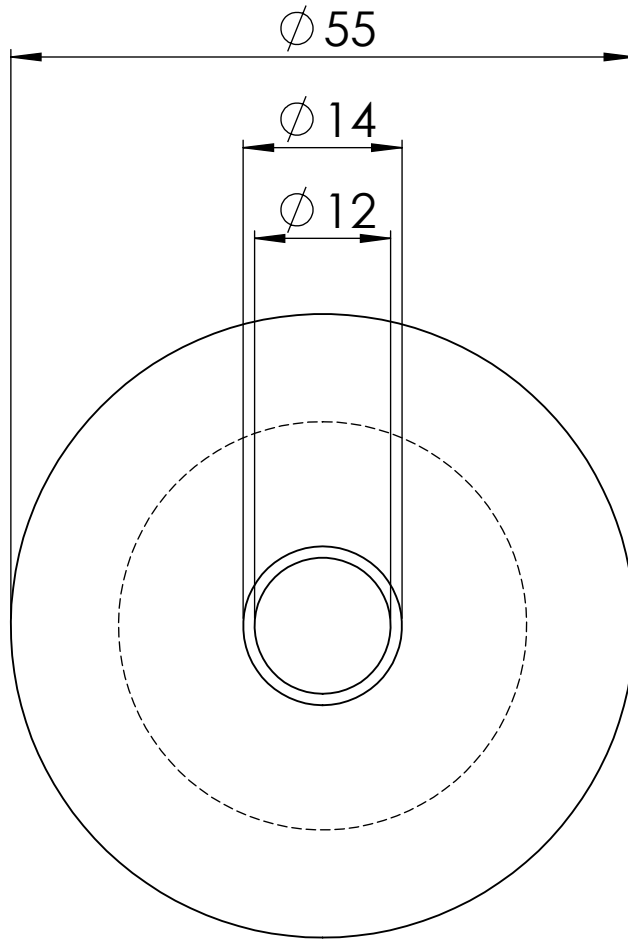
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque helice-motor	
COMENTARIOS				PIEZA	Carcaza protectora	No. PLANO
				MATERIAL	Acrílico	
				ESCALA	1:2	REV.



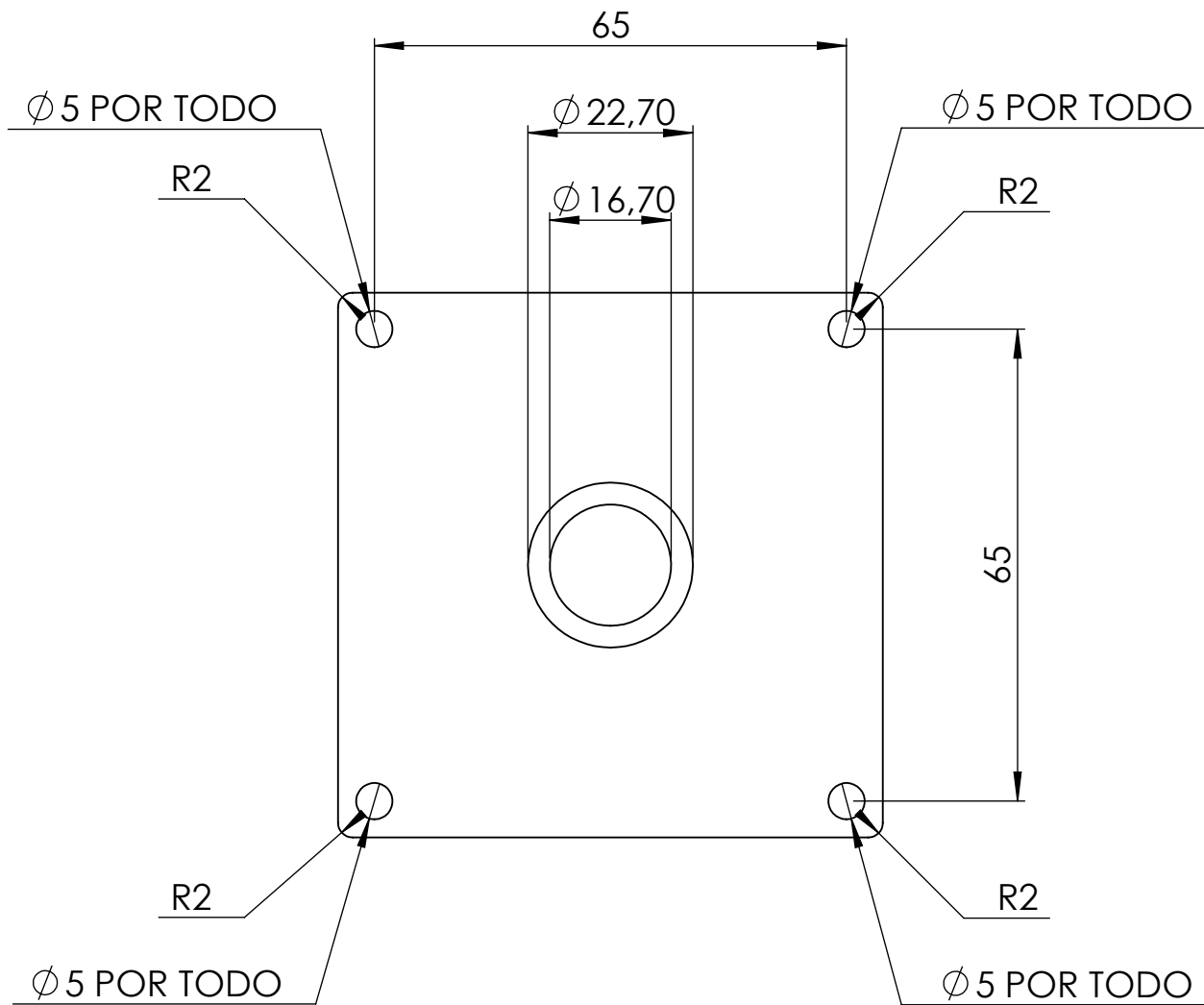
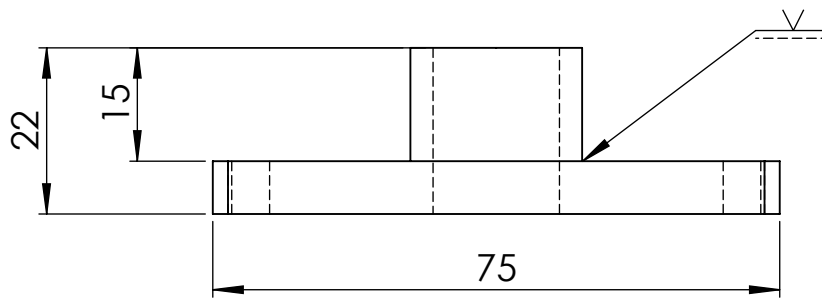
SECCIÓN A-A  
ESCALA 1 : 1



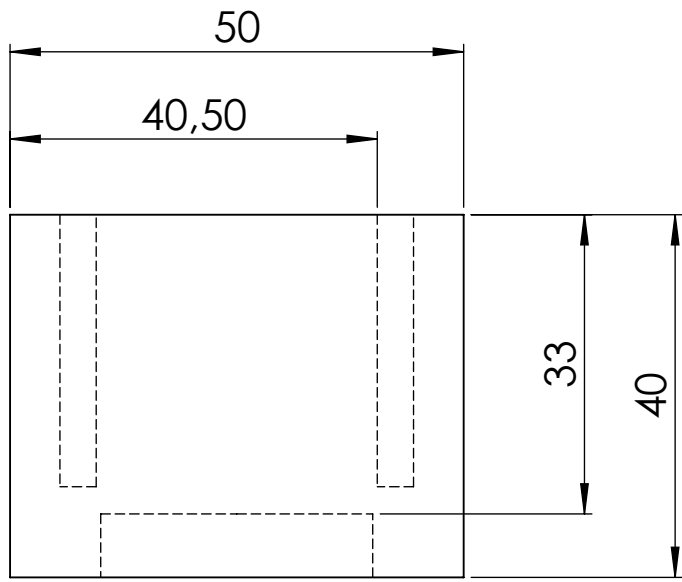
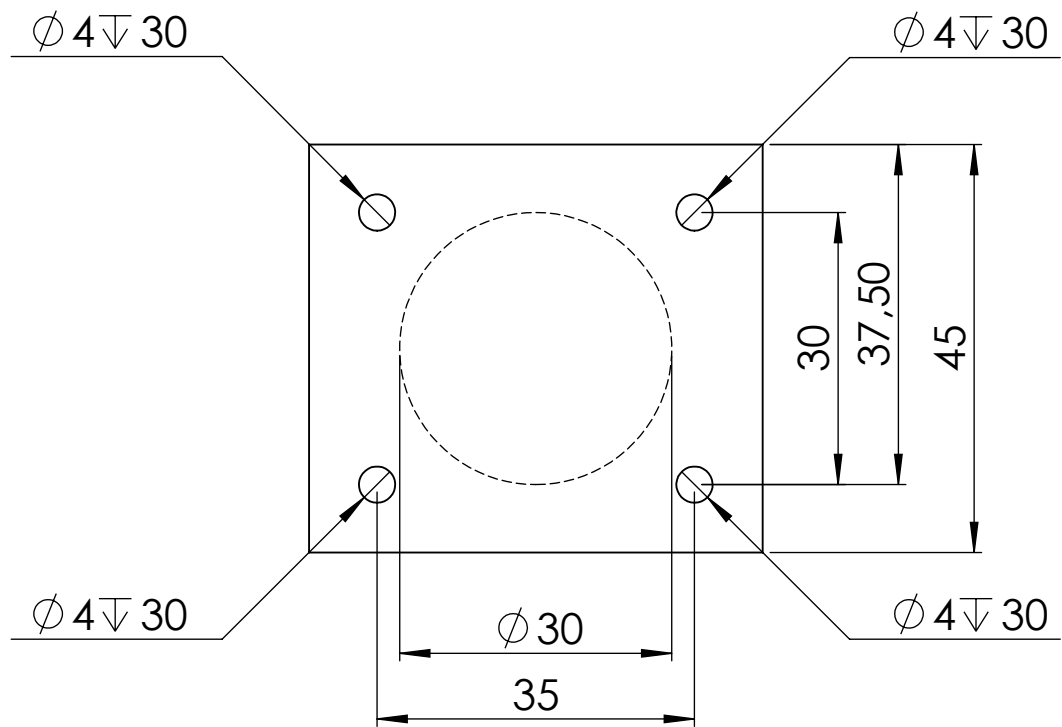
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque helice-motor	
COMENTARIOS				PIEZA	Soporte paquete motriz	No. PLANO
				MATERIAL	Aleación de aluminio 1060	
				ESCALA	1:1	REV.



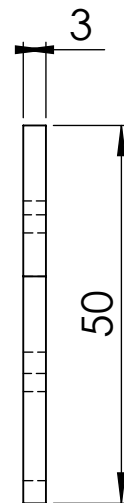
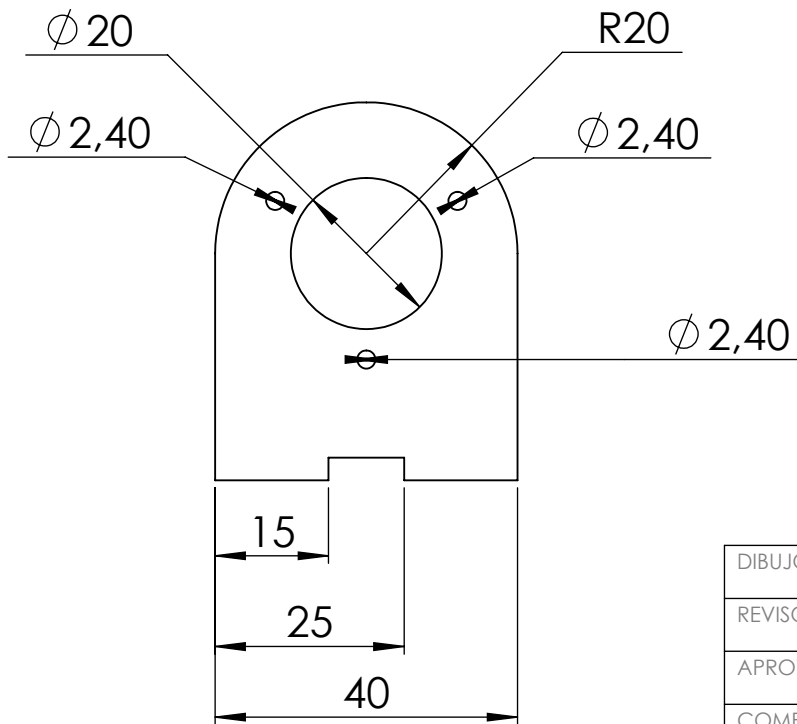
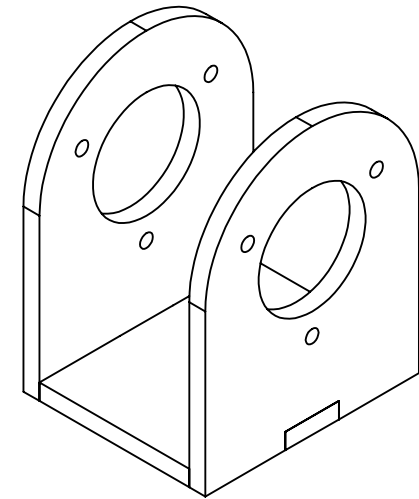
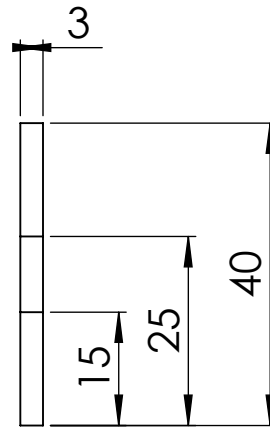
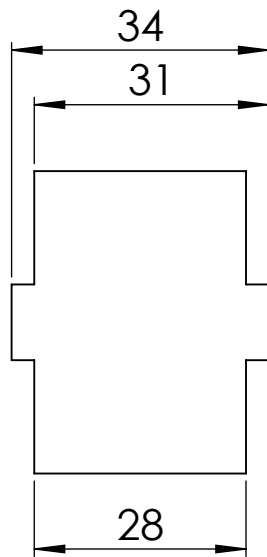
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Estructura base	
COMENTARIOS				PIEZA	Soporte resorte	No. PLANO
				MATERIAL	A36 Acero	
				ESCALA	1.5:1	REV.



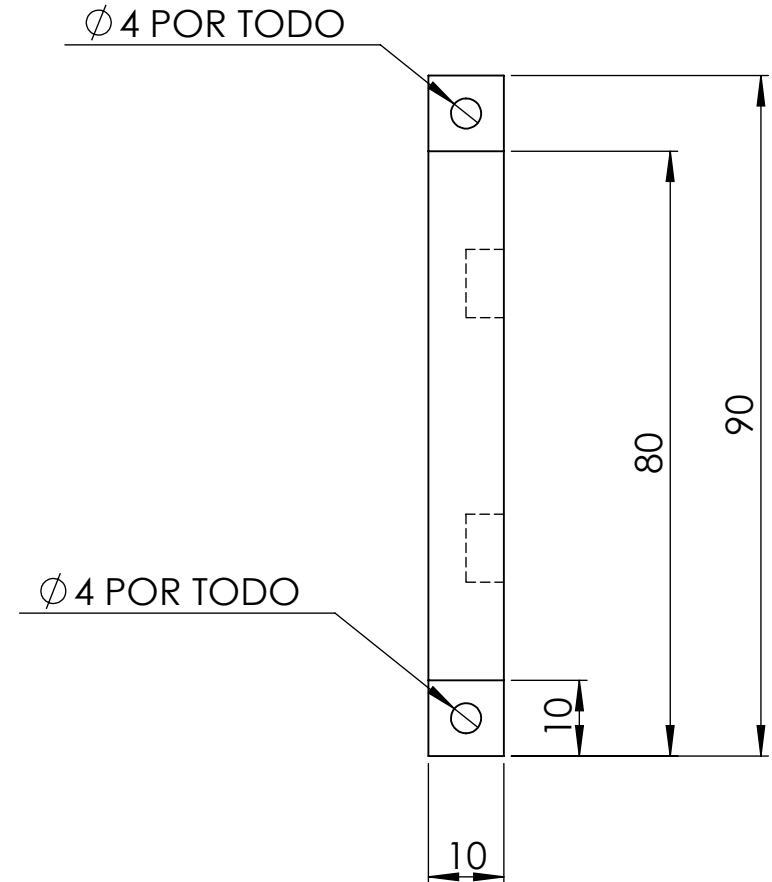
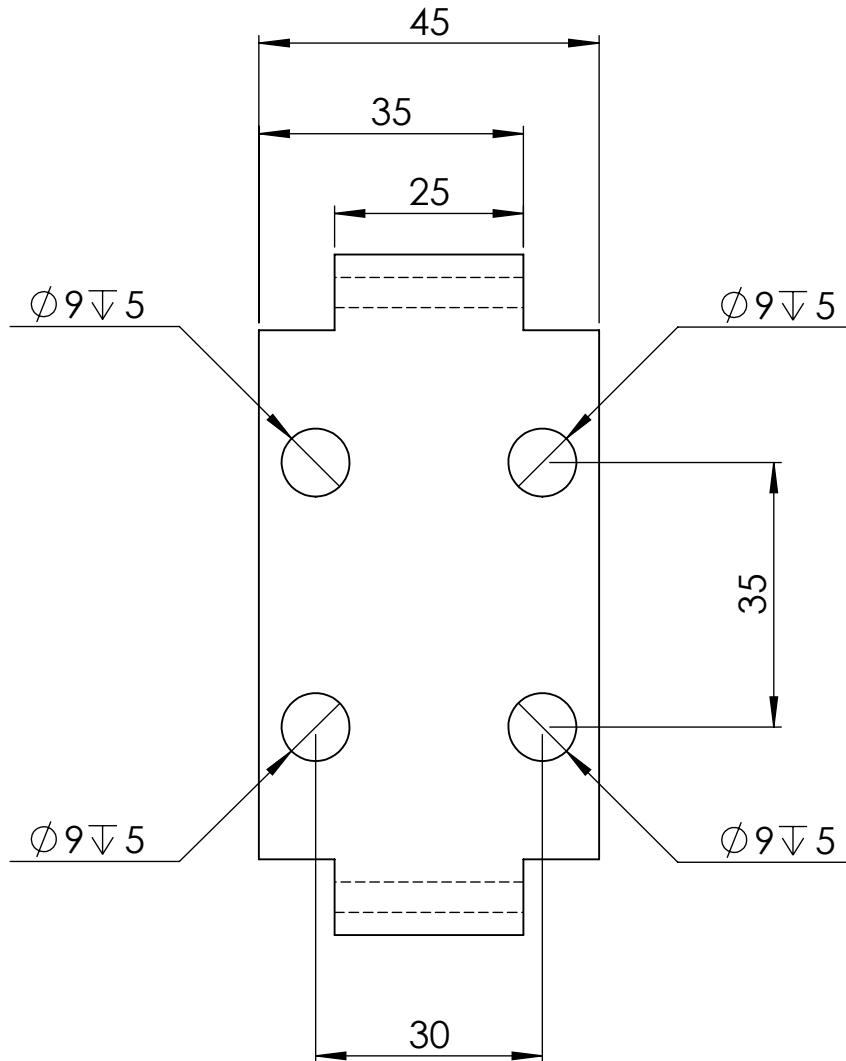
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Estructura base	
COMENTARIOS				PIEZA	Tapa base	No. PLANO
				MATERIAL	A36 Acero	
				ESCALA	1:1	REV.



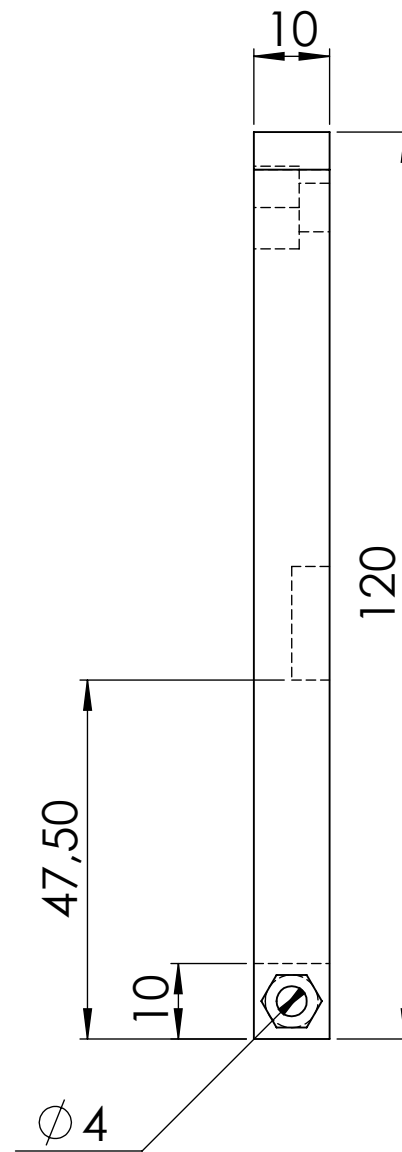
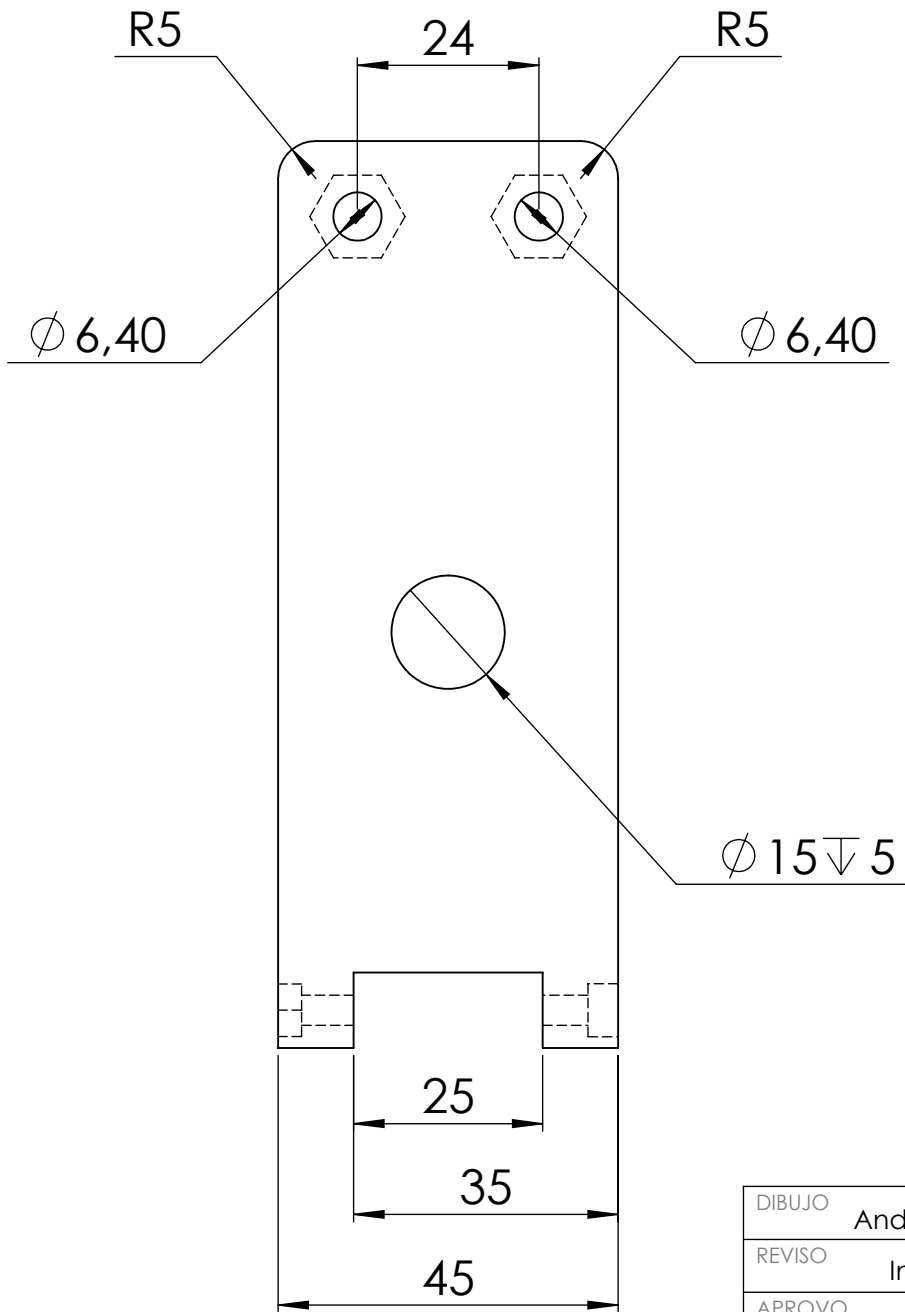
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque central	
COMENTARIOS				PIEZA	Apoyo bloque central	No. PLANO
				MATERIAL	Nilon 101	
				ESCALA	1.2:1	REV.



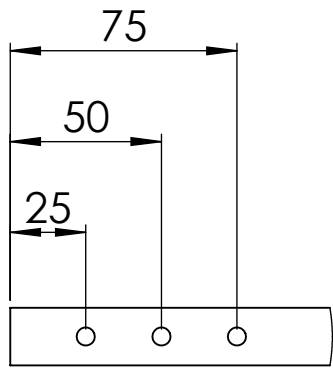
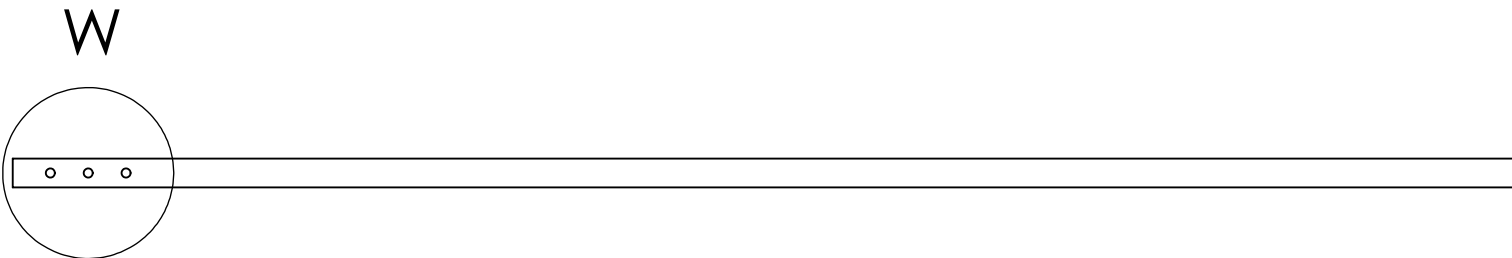
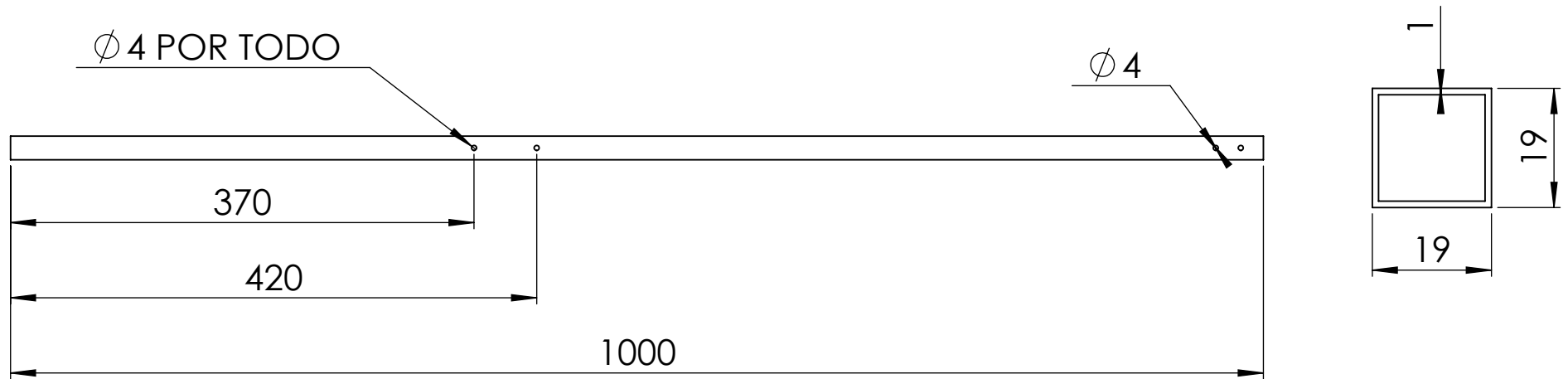
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque central	
COMENTARIOS				PIEZA	Soporte sensor elevación	No. PLANO
				MATERIAL	Acrilico	
				ESCALA	1:1	REV.



DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque central	
COMENTARIOS				PIEZA	Base bloque central	No. PLANO
				MATERIAL	Acrilico	
				ESCALA	1:1	REV.

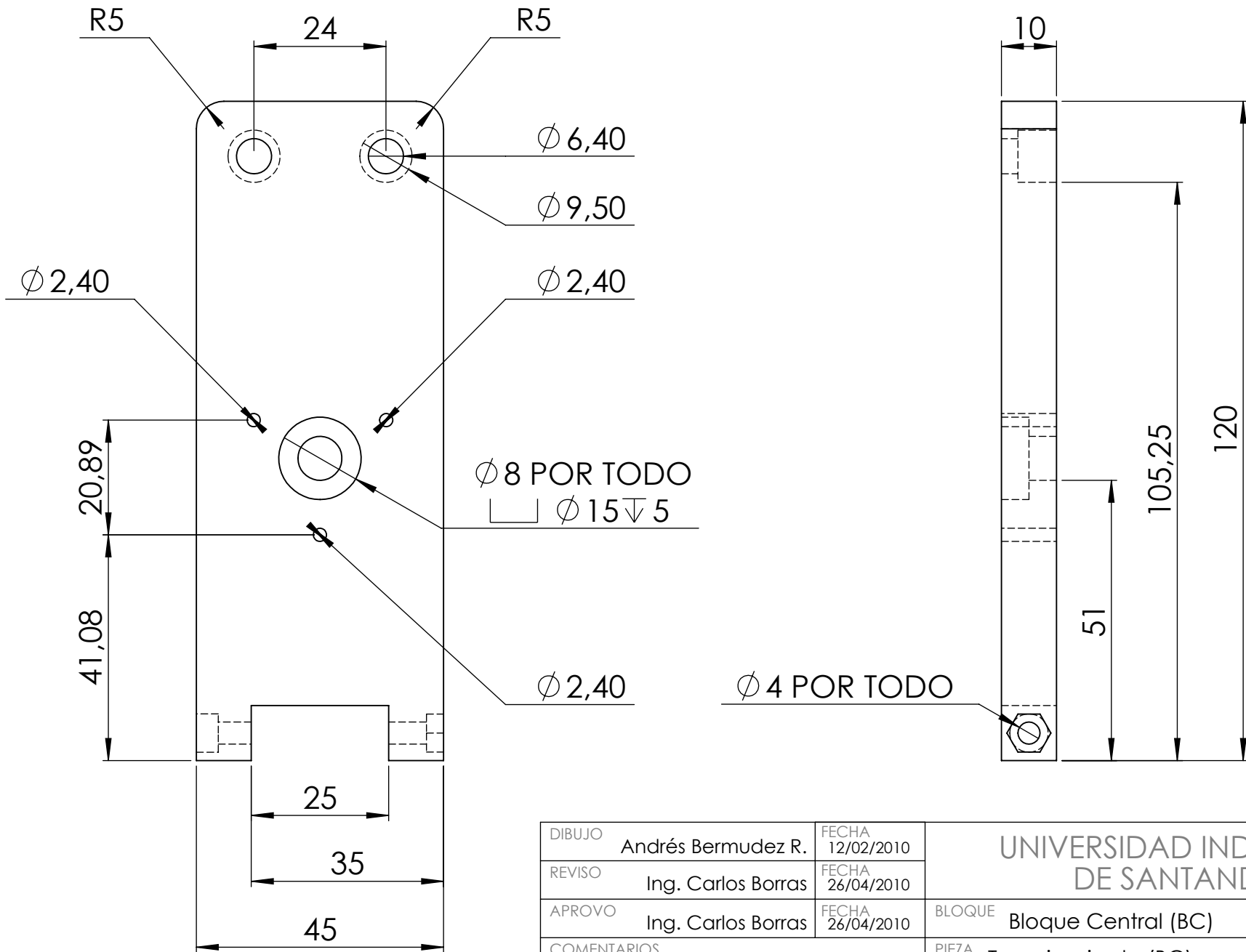


DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque Central (BC)	
COMENTARIOS				PIEZA	Tapa derecha BC	No. PLANO
				MATERIAL	Acrilico	
				ESCALA	1:1	REV.

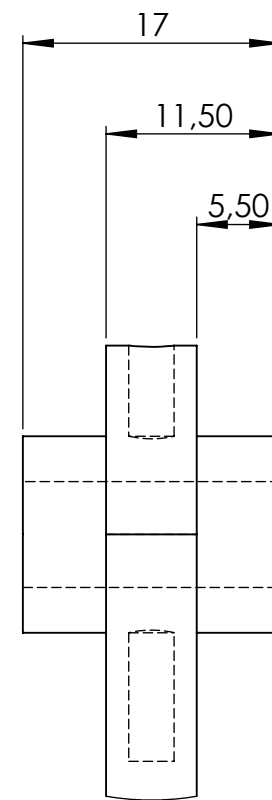
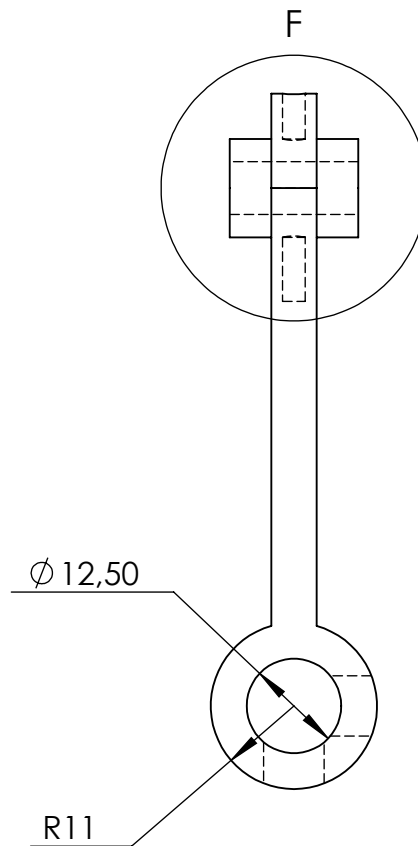
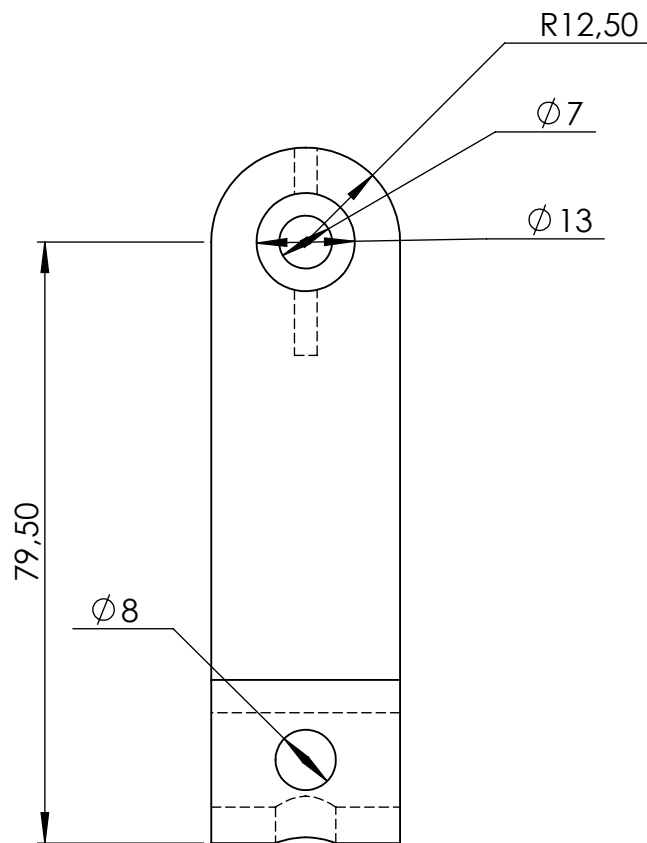


DETALLE W  
ESCALA 2 : 5

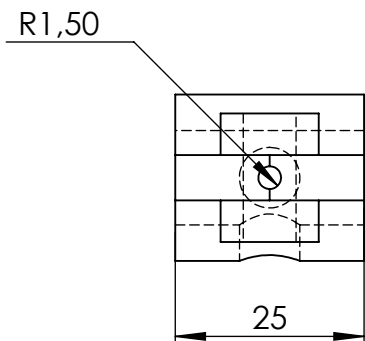
DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque viga principal	
COMENTARIOS				PIEZA	Viga Principal	No. PLANO
				MATERIAL	Aleación de aluminio 1060	
				ESCALA	1:5	REV.



DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque Central (BC)	
COMENTARIOS				PIEZA	Tapa izquierda (BC)	No. PLANO
				MATERIAL	Acrilico	
				ESCALA	1:1	REV.



DETALLE F  
ESCALA 2 : 1



DIBUJO	Andrés Bermudez R.	FECHA	12/02/2010	UNIVERSIDAD INDUSTRIAL DE SANTANDER		
REVISO	Ing. Carlos Borrás	FECHA	26/04/2010			
APROVO	Ing. Carlos Borrás	FECHA	26/04/2010	BLOQUE	Bloque helice-motor	
COMENTARIOS				PIEZA	Tapa izquierda (BC)	No. PLANO
				MATERIAL	Acrilico	
				ESCALA	1:1	REV.