



**UNIVERSIDAD POLITÉCNICA SALESIANA  
SEDE GUAYAQUIL  
CARRERA DE MECATRÓNICA**

**TEMA: IMPLEMENTACIÓN DE UN MÓDULO CLASIFICADOR  
LONGITUDINAL DE 2 GRADOS DE LIBERTAD CON UN  
CONTROLADOR EN VARIABLES DE ESTADO.**

Trabajo de titulación previo a la obtención del  
Título de Ingeniero en Mecatrónica

**AUTORES:** Carlos Andrés Granda Romero  
Christian Andrés Mora Yagual  
**TUTOR:** Franklin Illich Kuonquí Gaínza

Guayaquil - Ecuador

2022

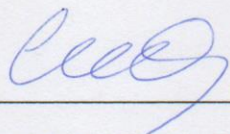
## CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, **Carlos Andrés Granda Romero** con documento de identificación N° **0706389178** y **Christian Andrés Mora Yagual** con documento de identificación N° **0921013868**; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo.

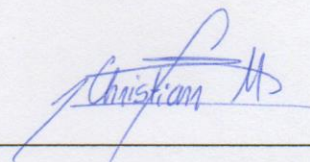
Guayaquil, 17 de marzo del año 2022

Atentamente,



---

Carlos Andrés Granda Romero  
0706389178



---

Christian Andrés Mora Yagual  
0921013868

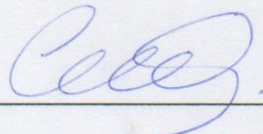
**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA  
UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, **Carlos Andrés Granda Romero** con documento de identificación N° **0706389178** y **Christian Andrés Mora Yagual** con documento de identificación N° **0921013868**, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del **Dispositivo Tecnológico: IMPLEMENTACIÓN DE UN MÓDULO CLASIFICADOR LONGITUDINAL DE 2 GRADOS DE LIBERTAD CON UN CONTROLADOR EN VARIABLES DE ESTADO**, el cual ha sido desarrollado para optar por el título de: Ingeniero en Mecatrónica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo a final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana

Guayaquil, 17 de marzo del año 2022

Atentamente,



---

Carlos Andrés Granda Romero  
0706389178



---

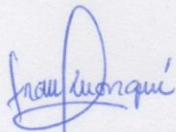
Christian Andrés Mora Yagual  
0921013868

## CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, **Franklin Illich Kuonquí Gaínza**, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **IMPLEMENTACIÓN DE UN MÓDULO CLASIFICADOR LONGITUDINAL DE 2 GRADOS DE LIBERTAD CON UN CONTROLADOR EN VARIABLES DE ESTADO**, realizado por **Carlos Andrés Granda Romero** con documento de identificación N° **0706389178** y por **Christian Andrés Mora Yagual** con documento de identificación N° **0921013868**, obteniendo como resultado final el trabajo de titulación bajo la opción **Dispositivo Tecnológico** que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 17 de marzo del año 2022

Atentamente,



---

Ing. Franklin Illich Kuonquí Gaínza, Mg.  
0909627432

## DEDICATORIA

Este trabajo de titulación está dedicado a mi padre José Luis Granda Bravo, por las enseñanzas sobre disciplina, responsabilidad y ética; a mi madre Fabiola Rocío Romero Pesantes, por su amor inmenso, su cuidado y apoyo sin igual para asegurar mi bienestar en el transcurso de todos estos años de estudios universitarios.

Finalmente agradezco a mi abuela materna Rosa Lilia Pesantes y mi hermana Allison María Sol Granda por su confianza absoluta que me ha impulsado a seguir adelante y quienes siempre han cuidado de mí.

**Carlos Andrés Granda Romero**

Este trabajo de titulación está dedicado a mi madre Elsa del Pilar Yagual, quien es el motor de mi vida y gracias a su esfuerzo y gran sacrificio lograré terminar mis estudios universitarios. A mi padre Lenín Mora quien me inspiró a estudiar esta carrera, me enseñó a trabajar arduamente para conseguir y lograr todos mis objetivos, quien en vida me enseñó lo importante que es la perseverancia y lo duro que puede ser la vida, pero a pesar de la dificultad mientras exista un rayo de luz nada malo pasará. Ambos me guiaron con su amor y perseverancia a lo largo de mi vida. A mis hermanos Joel y Alexandra Mora quienes además de mis padres son las personas que más amo en mi vida y quienes al igual que mis padres han estado en todo momento en cada paso que he dado como también en cada caída.

Dedico esta investigación a toda mi familia, que me han enseñado que el mejor regalo que se puede dar en la vida es la educación, el cual es un puente que nos permite aprender e innovar a nuevos mundos en donde los sueños son posibles.

**Christian Andrés Mora Yagual**

## AGRADECIMIENTO

Agradezco principalmente a Dios, a mi padre y madre por su amor incondicional y por inculcarme el valor de la perseverancia para afrontar los retos y dificultades de la vida.

Agradezco a la Universidad Politécnica Salesiana, a los docentes y a mis compañeros por formar importante del desarrollo y cumplimiento del presente trabajo de investigación.

Agradezco además de manera especial al Ing. Franklín Kuonquí por su guía ante la realización de este proyecto con sugerencias y por la confianza brindada para culminar el mismo.

**Carlos Andrés Granda Romero**

Agradezco a los docentes de la carrera de ingeniería en mecánica que han formado parte de mi formación profesional, quienes con su conocimiento y su apoyo brindaron las herramientas para un futuro exitoso como un ingeniero mecánico.

Agradezco a todas esas amistades que me dio directamente como indirectamente la universidad, porque me han demostrado ser grandes seres humanos que siempre me han brindado de su apoyo aun cuando fue difícil avanzar, por acompañarme y hacer de este camino más agradable y enriquecedor.

**Christian Andrés Mora Yagual**

## RESUMEN

El presente proyecto tiene como objetivo la implementación de un módulo clasificador para la realización de prácticas de ingeniería en control. Este sistema se basa en un modelo de viga y bola, que es uno de los modelos clásicos más utilizados para el aprendizaje y diseño de controladores de sistemas dinámicos inestables que son muy complejos de estabilizar.

Para la implementación de este prototipo, se cuenta con un sistema de visión artificial que permite la clasificación de las esferas considerando su color y definiendo la ubicación del recipiente donde debe ser almacenada. A continuación, el sensor infrarrojo mide la distancia a la que se encuentra la esfera del punto de referencia y el controlador en LabVIEW decide la inclinación requerida de las vigas para que dicha esfera se acerque a la ubicación del recipiente seleccionado. Finalmente cuando la esfera está en una sección determinada en las cercanías de la referencia se da la orden para que abran las vigas y caiga en el recipiente correspondiente.

Por último, se diseñaron y afinaron dos controladores: un controlador proporcional y otro por reubicación de polos mediante realimentación de estados. Para el diseño de estos controladores, se obtuvo el modelo linealizado de la planta y se realizaron los cálculos y simulaciones en MATLAB/Simulink. Sin embargo, la afinación se la realizó en LabVIEW con el prototipo construido donde una placa Arduino sirvió como tarjeta de adquisición de datos.

**Palabras claves:** sistemas de control, viga y bola, visión artificial, PID, realimentación de estados, reubicación de polos, MATLAB, LabVIEW, Arduino Uno.

## ABSTRACT

The objective of this project is the implementation of a classifier module for carrying out control engineering practices. This system is based on a ball and beam model, which is one of the most widely used classical models for learning and designing controllers for unstable dynamic systems that are very complex to stabilize.

For the implementation of this prototype, there is an artificial vision system that allows the classification of the spheres considering their color and defining the location of the container where it should be stored. The infrared sensor then measures how far the sphere is from the reference point, and the controller in LabVIEW decides the required tilt of the beams to bring the sphere closer to the selected container location. Finally, when the sphere is in a certain section in the vicinity of the reference, the order is given to open the beams and fall into the corresponding container.

Finally, two controllers were designed and tuned: a proportional controller and a pole relocation controller using state feedback. For the design of these controllers, the linearized model of the plant was obtained and the calculations and simulations were carried out in MATLAB/Simulink. However, the tuning was done in LabVIEW with the prototype built where an Arduino board served as data acquisition card.

**Keywords:** control systems, ball and beam, artificial vision, PID, state feedback, root locus, MATLAB, LabVIEW, Arduino Uno.



## ÍNDICE

<b>I.</b>	<b>INTRODUCCIÓN</b>	13
<b>II.</b>	<b>PROBLEMA</b>	14
<b>III.</b>	<b>OBJETIVOS</b>	14
III-A.	Objetivo general . . . . .	14
III-B.	Objetivos específicos . . . . .	14
<b>IV.</b>	<b>FUNDAMENTOS TEÓRICOS</b>	14
IV-A.	Robots . . . . .	14
IV-B.	Configuraciones de sistemas viga bola . . . . .	15
IV-C.	MATLAB . . . . .	16
IV-D.	LabVIEW . . . . .	16
IV-D1.	Panel frontal . . . . .	17
IV-D2.	Diagrama de bloques . . . . .	17
IV-E.	NI Vision and Motion . . . . .	18
IV-E1.	NI IMAQ . . . . .	18
IV-E2.	IMAQ VISION . . . . .	18
IV-E3.	Vision Assistant . . . . .	18
IV-F.	LINUX . . . . .	19
IV-G.	Sensor de distancia infrarrojo SHARP GP2Y0A21 . . . . .	19
IV-H.	Arduino UNO . . . . .	20
IV-I.	Servomotor . . . . .	20
IV-J.	Estabilidad de sistemas . . . . .	22
IV-K.	Linealidad de sistemas . . . . .	26
IV-L.	Elementos de un sistema de control . . . . .	27
IV-L1.	Planta . . . . .	27
IV-L2.	Referencia . . . . .	27
IV-L3.	Perturbación . . . . .	27
IV-L4.	Controlador . . . . .	27
IV-L5.	Retroalimentación . . . . .	27
IV-M.	Controlador PID . . . . .	27
IV-N.	Control por realimentación de estado . . . . .	28
IV-N1.	Controlabilidad . . . . .	29
IV-N2.	Forma canónica controlable . . . . .	29
IV-N3.	Asignación de polos por realimentación de estados . . . . .	29
<b>V.</b>	<b>MARCO METODOLÓGICO</b>	30
V-A.	Circuito electrónico del módulo didáctico . . . . .	30
V-B.	Diseño mecánico de la planta . . . . .	30
V-C.	Construcción de planta . . . . .	31
V-C1.	Piezas metálicas . . . . .	31
V-C2.	Piezas en acrílico . . . . .	32
V-C3.	Piezas en PLA . . . . .	33
V-C4.	Sistema de apertura . . . . .	34
V-C5.	Ensamble . . . . .	35
V-D.	Modelo matemático . . . . .	39
V-D1.	Modelo de la función de transferencia . . . . .	42
V-D2.	Modelo matemático en espacios de estados . . . . .	44

V-E.	Control teórico del sistema . . . . .	45
V-F.	Diseño de controladores para el modelo matemático . . . . .	45
	V-F1. Sintonización de controlador PID para el modelo matemático . . . . .	45
	V-F2. Controlador por realimentación de estados por asignación de polos . . . . .	46
V-G.	Comunicación entre Arduino y LabVIEW . . . . .	50
V-H.	Conexión de cámara USB de smartphone con LabVIEW . . . . .	52
V-I.	Diseño e implementación de los controladores en LabVIEW . . . . .	52
V-J.	Clasificación por color mediante NI Vision Express . . . . .	53
<b>VI.</b>	<b>RESULTADOS</b>	<b>56</b>
VI-A.	Simulación de la respuesta del sistema con el controlador PID . . . . .	56
VI-B.	Simulación de la respuesta del sistema con el controlador por realimentación de estados . . . . .	56
VI-C.	Comparación de técnicas de control utilizadas . . . . .	57
VI-D.	Desempeño del sistema implementado . . . . .	58
<b>VII.</b>	<b>CRONOGRAMA</b>	<b>60</b>
<b>VIII.</b>	<b>PRESUPUESTO</b>	<b>62</b>
<b>IX.</b>	<b>CONCLUSIONES</b>	<b>62</b>
<b>X.</b>	<b>RECOMENDACIONES</b>	<b>62</b>
	<b>Apéndice A: planos</b>	<b>65</b>
	<b>Apéndice B: Equipos de Manufactura</b>	<b>78</b>
	<b>Apéndice C: Herramientas LabVIEW</b>	<b>81</b>
	<b>Apéndice D: Programación en LabVIEW</b>	<b>85</b>

## ÍNDICE DE FIGURAS

1.	Robots planares de 2 GdL [21]. . . . .	15
2.	Configuración viga-biela [18]. . . . .	15
3.	Configuración viga con motor central [18]. . . . .	15
4.	Configuración viga central con correa [18]. . . . .	16
5.	Componentes de un VI en LabVIEW: Panel frontal y diagrama de bloque [15]. . . . .	17
6.	Funciones de visión artificial en LabVIEW [15]. . . . .	18
7.	Bloques Vision Express [15]. . . . .	18
8.	Funciones para abrir y cerrar comunicación serial; periféricos, sensores del Toolkit de LINX - LabVIEW Makerhub [14]. . . . .	19
9.	Sensor de distancia infrarrojo SHARP GP2Y0A21 [16]. . . . .	19
10.	Bloques de E/S analógicas en LINX [14]. . . . .	20
11.	Tarjeta Arduino UNO R3 [1]. . . . .	20
12.	Servomotores marca TowerPro [12]. . . . .	21
13.	Modulación de ancho de pulso vs. posición angular del eje en un servomotor [12]. . . . .	21
14.	Bloque de servo en LINX [14]. . . . .	21
15.	Bloque de PWM en LINX [14]. . . . .	22
16.	Respuesta escalón de un sistema con $\zeta = 0$ . . . . .	23
17.	Respuesta escalón de un sistema con $\zeta = 0,2$ y $\omega_n = 2$ . . . . .	24
18.	Respuesta escalón de un sistema con $\zeta = 1$ y $\omega_n = 2$ . . . . .	24
19.	Respuesta escalón de un sistema con $\zeta = 2$ y $\omega_n = 2$ . . . . .	25
20.	Parámetros de la respuesta escalón de un sistema de segundo orden subamortiguado [7]. . . . .	26
21.	Propiedad de proporcionalidad [7]. . . . .	26
22.	Propiedad de aditividad [7]. . . . .	27
23.	Principio de proporcionalidad [7]. . . . .	27
24.	Representación típica de un sistema en el espacio de estados [7] . . . . .	28
25.	Diagrama de bloques de un sistema realimentado en el espacio de estados. . . . .	29
26.	Conexiones del Arduino UNO con la PC, sensor infrarrojo y servomotores. . . . .	30
27.	Diseño de planta, por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	31
28.	Piezas metálicas, por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	32
29.	Plano de las piezas metálicas a cortar por láser usando oxígeno, por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	32
30.	Piezas de acrílico, por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	33
31.	Plano de las piezas de acrílico a cortar por láser usando plasma, por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	33
32.	Piezas para impresión en 3D , por C. Mora y C. Granda, Inventor Autodesk, 2020. . . . .	34
33.	Piezas en formato STL para ser impresas por medio de impresora de 3D, por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	34
34.	Sistema de apertura de las vigas, por C. Mora y C. Granda, Inventor Autodesk, 2020. . . . .	35
35.	Sistema de engranajes, por C. Mora y C. Granda, Inventor Autodesk, 2020. . . . .	35
36.	Módulo clasificador longitudinal de 2 grados de libertad, por C. Mora y C. Granda. . . . .	36
37.	Plano explosionado del módulo por C. Granda y C. Mora, Inventor Autodesk, 2020. . . . .	37
38.	Representación gráfica de la trayectoria de la bola con sus componentes $q_1$ y $q_2$ [19]. . . . .	40
39.	Representación de las constantes y variables que influyen en el sistema para modelar la función de transferencia de la planta [19]. . . . .	42
40.	Respuesta transiente del sistema en lazo abierto a una entrada escalón. . . . .	45
41.	Sistema realimentado con un controlador de tipo PID. . . . .	46
42.	Parámetros obtenidos del PID Tuner y respuesta del sistema bajo los criterios de diseño . . . . .	46
43.	Creación del modelo en lazo abierto en espacio de estados de la planta . . . . .	47
44.	Modelo de espacio de estado en lazo abierto en MATLAB . . . . .	47

45.	Comprobación de controlabilidad en MATLAB . . . . .	48
46.	Matriz de controlabilidad en MATLAB . . . . .	48
47.	Obtención de matriz de ganancias K . . . . .	49
48.	Matriz de ganancias K en MATLAB . . . . .	49
49.	Obtención de sistema en lazo cerrado e implementación de una ganancia $k_0$ . . . . .	49
50.	Controlador realimentado en espacio de estados con precompensador estático $k_0$ . . . . .	50
51.	Elección de placa a comunicar mediante LINX Firmware Wizard. . . . .	51
52.	Elección de puerto serial de comunicación entre Arduino y LabVIEW. . . . .	51
53.	Mensaje de comunicación serial exitosa entre Arduino y LabVIEW. . . . .	52
54.	Interfaz de usuario de los controladores didáctico en LabVIEW. . . . .	53
55.	Esquema de bloques utilizados para la clasificación por color y determinación de setpoint . . . . .	53
56.	Adquisición de la imagen a procesar mediante la herramienta NI Vision Acquisition Express . . . . .	54
57.	Funciones de procesamiento de imágenes de la herramienta NI Vision Assistant . . . . .	54
58.	Delimitación de la región de interés mediante la función Image Mask . . . . .	55
59.	Identificación del color de la pelota amarilla mediante la función Color Pattern Matching . . . . .	55
60.	Identificación del color de la pelota celeste mediante la función Color Pattern Matching . . . . .	56
61.	Respuesta del sistema con el controlador PID en Simulink . . . . .	56
62.	Respuesta del sistema con el controlador por realimentación de estados con precompensador $k_0$ . . . . .	57
63.	Modelo para comparación de técnicas de control . . . . .	57
64.	Respuestas multiplexadas de las 2 técnicas de control utilizadas . . . . .	58
65.	Gráficas de la respuesta del controlador PID implementado en LabVIEW . . . . .	59
66.	Gráficas de la respuesta del controlador por asignación de polos por realimentación de estados implementado en LabVIEW . . . . .	60
67.	Cronograma . . . . .	61
68.	Lámina acotada del soporte de viga, por C.Granda y C.Mora, Inventor Autodesk . . . . .	65
69.	Lámina acotada del brazo porta viga, por C.Granda y C.Mora, Inventor Autodesk . . . . .	66
70.	Lámina acotada de la viga, por C.Granda y C.Mora, Inventor Autodesk . . . . .	67
71.	Lámina acotada del soporte del sistema de abertura, por C.Granda y C.Mora, Inventor Autodesk . . . . .	68
72.	Lámina acotada del soporte del servomotor, por C. Granda y C. Mora, Inventor Autodesk . . . . .	69
73.	Lámina acotada de la caja porta estructura, por C.Granda y C.Mora, Inventor Autodesk . . . . .	70
74.	Lámina acotada de los sub brazos, por C.Granda y C.Mora, Inventor Autodesk . . . . .	71
75.	Lámina acotada del brazo, por C.Granda y C.Mora, Inventor Autodesk . . . . .	72
76.	Lámina acotada de las esferas del sistema, por C.Granda y C.Mora, Inventor Autodesk . . . . .	73
77.	Lámina acotada del soporte de los sub brazos y servo, por C.Granda y C.Mora, Inventor Autodesk . . . . .	74
78.	Lámina acotada del soporte del sistema de abertura, por C.Granda y C.Mora, Inventor Autodesk . . . . .	75
79.	Lámina acotada del soporte del sensor infrarrojo, por C.Granda y C.Mora, Inventor Autodesk . . . . .	76
80.	Lámina del contorno del prototipo en acrílico, por C.Granda y C.Mora, Inventor Autodesk . . . . .	77
81.	Corte láser del prototipo en acrílico, por C.Granda y C.Mora . . . . .	78
82.	Impresión en 3D de las piezas. . . . .	79
83.	Corte láser de la estructura metálica . . . . .	80
84.	Menú de funciones, estructuras de control e indicadores del panel frontal en LabVIEW [15]. . . . .	81
85.	Paleta de funciones del diagrama de bloques del VI[15]. . . . .	82
86.	Paleta de funciones numéricas [15]. . . . .	83
87.	Paleta de funciones lógicas [15]. . . . .	84
88.	Paleta de funciones de hileras(arrays) [15]. . . . .	84
89.	Paleta de funciones de comparación [15]. . . . .	85
90.	Diagrama de bloques completo del controlador PID en LabVIEW . . . . .	86

## I. INTRODUCCIÓN

El presente estudio tiene como fin la creación de un módulo de viga y bola adaptado a dos grados de libertad, para que además de tener la posibilidad de posicionar longitudinalmente la bola, permita clasificar las mismas en recipientes mediante el reconocimiento del color de las bolas al hacer contacto con la planta, para lo cual se implementará el uso de visión artificial para el reconocimiento del color y mediante un sistema de apertura en las vigas pueda caer por gravedad la bola en su correspondiente recipiente.

Para el funcionamiento de la planta, se implementarán dos servomotores que generan un mecanismo de tipo rotacional. El primero representará la variación del desplazamiento angular por medio de la entrada del sistema, la cual es la posición de la bola y el segundo formara parte del sistema de apertura de las vigas para la clasificación en los recipientes.

Además de la implementación de una cámara para la visión artificial y el uso de un sensor de proximidad de tipo infrarrojo para la recolección de datos del sistema que a su vez darán el control a la bola para su posicionamiento longitudinal, mediante un controlador en variables de estado que será diseñado y probado utilizando las herramientas de MATLAB, Simulink y LabVIEW; los mismos que se desarrollaran dentro de este estudio.

## II. PROBLEMA

En la actualidad las teorías de control PID usadas normalmente en la automatización de procesos son: teoría de control clásico que hace referencia a los procesos donde el sistema cuenta con una salida y una entrada (SISO) y, en las plantas de producción modernas donde los procesos se caracterizan por ser más intrincados, se utilizan sistemas de múltiples entradas y salidas (MIMO) [2].

A nivel mundial, en industrias grandes y medianas el controlador PID posibilita acelerar la producción y paralelamente mejorar la calidad del producto, dando como resultado mayor eficiencia debido al ahorro de energía, mejor rendimiento de la maquinaria e incremento de ingresos y estabilidad económica de una compañía a mediano plazo.

Un estudio manifiesta que el 95 % de la automatización industrial implementan controladores PID en plataformas como los microcontroladores y PLC [13]. Sin embargo, existen circunstancias donde los parámetros y funciones preestablecidas en estas plataformas no son suficientes para controlar adecuadamente procesos industriales más complejos. Cuando un proceso llega a complicarse ya sea por su no linealidad, inestabilidad natural, complejidad, etc. y dichas técnicas de control (PID) no logran satisfacer completamente la automatización del proceso se opta por implementar técnicas de control novedosas y eficientes, pero poco implementadas en la industria como el uso de la lógica difusa, controladores en variables de estado, control no lineal, etc.

En Latinoamérica “el mecanismo viga y bola es un caso de estudio típico en la ingeniería de control, puesto que introduce el tema de sistemas de segundo orden inestables. Esto se debe a que en lazo abierto la salida del sistema, es decir, la posición de la bola en la viga cambia indefinidamente hasta encontrar un final de la viga, cuando el ángulo de la viga permanece constante en cualquier valor distinto de cero, lo que puede asociarse con una señal de entrada al sistema tipo escalón.” [10].

En los laboratorios de la carrera de Ingeniería Mecatrónica de la sede no existe un módulo, bola y viga donde el estudiante pueda adquirir de una forma práctica los conocimientos de Ingeniería de Control y que además tenga una aplicación industrial de clasificación con visión artificial.

## III. OBJETIVOS

### III-A. *Objetivo general*

Desarrollar un módulo clasificador de dos grados de libertad utilizando elementos de bajo costos localizables en su mayoría en el mercado local, que permita la aplicación práctica de los conceptos de las materias de Ingeniería de Control e Ingeniería de Control Moderno.

### III-B. *Objetivos específicos*

- Construir el módulo de clasificación, considerando componentes mecánicos y electrónicos; y tomando en consideración el espacio del área de trabajo de los estudiantes en el laboratorio.
- Analizar el comportamiento en lazo abierto de la planta para la caracterización las condiciones físicas, cinemáticas y dinámicas para el planteamiento de las ecuaciones que definen la dinámica del sistema.
- Diseñar un controlador PID utilizando la función de transferencia identificada para el reconocimiento desempeño mínimo deseado en los controladores avanzados que se diseñarán posteriormente.
- Diseñar un controlador en variables de estado por reubicación de polos que tenga un desempeño superior al del PID.
- Integrar los sistemas de control diseñados en el módulo de clasificación para la verificación del desempeño de dichos controladores en el prototipo.

## IV. FUNDAMENTOS TEÓRICOS

### IV-A. *Robots*

Un robot es una máquina compuesta de sensores que reciben datos de entradas controlados por un computador, esta es programada para realizar un movimiento, manipular un objeto y realizar acciones a la vez en contacto a su área de trabajo. Su objetivo principal es la sustitución del ser humano en tareas peligrosas, difíciles, repetitivas o desagradables de manera más precisa, rápidas y seguras [4].

Un robot está hecho por un conjunto de series de eslabones conectados por medio de articulaciones que dan como resultado el movimiento relativo entre eslabones. Las articulaciones del robot podrán tener un grado prismático, dos grados planares o tres grados rotulares de libertad. El grado de libertad depende del número de movimientos que puede dar un robot en el espacio [11].

En los sistemas planares su movimiento se reduce como se observa en la figura 1. Por ende, pueden tener 2 o 3 grados de libertad, correspondientes al movimiento de desplazamiento en el plano y a un movimiento de rotación sobre un eje perpendicular al plano de movimiento [21].

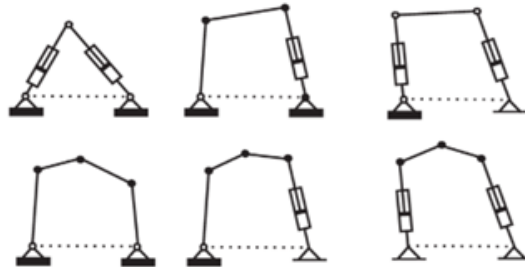


Figura 1. Robots planares de 2 GdL [21].

#### IV-B. Configuraciones de sistemas viga bola

Básicamente, el sistema de viga bola funciona de manera que la bola rueda sobre la viga y pueda posicionar la bola por medio de un control, cambiando su ángulo por medio de un motor. Como se muestra en la figura 2 esta configuración tiene un mecanismo de viga-biela en la que la viga se queda fija en unos de sus extremos mientras que el otro extremo está conectada a una biela y motor eléctrico, la inclinación dependerá de la longitud de la biela [18].

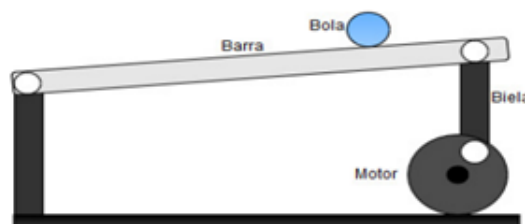


Figura 2. Configuración viga-biela [18].

En la siguiente configuración que se observa en la figura 3 el eje del motor queda fijado en el centro de la viga, el movimiento angular del motor es proporcional al de la viga [18].

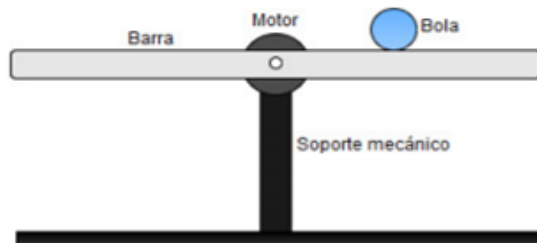


Figura 3. Configuración viga con motor central [18].

Como se puede observar en la figura 4 se implementa el uso de una correa para esta configuración, mientras a su vez el motor queda sobre la base, el ángulo del desplazamiento angular dependerá si viga ya no puede girar más [18].

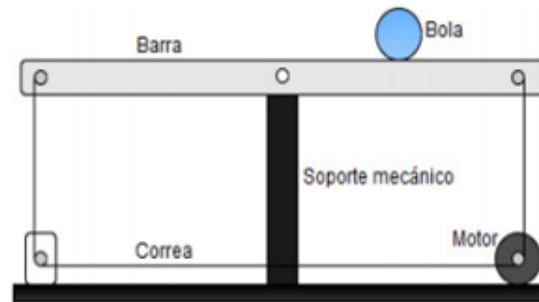


Figura 4. Configuración viga central con correa [18].

#### IV-C. MATLAB

Es un programa el cual su nombre viene de la abreviación de Matrix Laboratory cuyo fin es la realización de cálculos numéricos técnicos y científicos de manera rápida, puede hacer operaciones con matrices y vectores. MATLAB posibilita realizar operaciones con números escalares y números complejos. Una de las características que nos da esta herramienta es la gran cantidad de gráficos en dos y tres dimensiones. MATLAB dispone de un código básico y sin fin de librerías especializadas, puede ser arrancada como cualquiera otra aplicación de Windows [20].

#### IV-D. LabVIEW

Se trata de la abreviatura de Laboratory Virtual Instrument Engineering Workbench. El cual es un entorno de programación gráfico en el que se pueden hacer aplicaciones de forma sencilla y rápida. Fue creado por National Instruments en abril de 1986 [9].

Originalmente, el programa estaba dirigido al control de aplicaciones de instrumentos electrónicos para desarrollar sistemas de instrumentos, los llamados instrumentos virtuales. Por tal motivo estos programas hechos en LabVIEW se guardan en archivos nombrados VIs y tienen la misma extensión, instrumentos virtuales. Dado que los instrumentos reales tienen un panel frontal con botones y pantallas de usuario, y el circuito interno entre sus componentes, LabVIEW tiene dos ventanas como se muestra en la figura 5: el panel frontal es lo que el usuario podrá visualizar, generalmente con un fondo gris; y el diagrama de bloques, que corresponde al lugar donde se lleva a cabo la programación, generalmente con un fondo blanco [9].



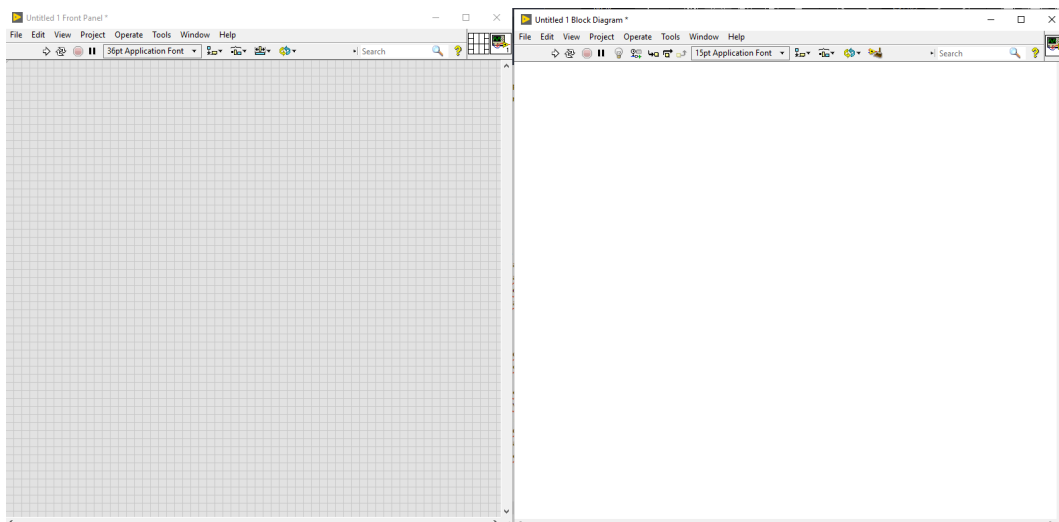


Figura 5. Componentes de un VI en LabVIEW: Panel frontal y diagrama de bloque [15].

Para cada programa de control o indicador en el panel frontal del usuario tiene una terminal que corresponde al diagrama de bloques. Al ejecutar el VI, los valores del controlador que fluyen a través de los bloques del diagrama, se aplican a funciones en el gráfico y los resultados se pasan a otras funciones o indicadores. Además, los diagramas de bloques contienen bibliotecas de LabVIEW como estructuras y funciones utilizadas para crear programas. Casi todos los nodos están conectados al diagrama de bloques, incluidos los terminales, funciones y estructuras de control e indicadores [15].

*IV-D1. Panel frontal:* Las herramientas del panel frontal que proporciona LabVIEW, mostradas en los anexos figura 84, son menús, barras de herramientas y paletas flotantes, entre otras [9] y se detallan las utilizadas a continuación:

- Numeric Controls: Para la introducción de cantidades numéricas.
- Numeric Indicators: Para la visualización de cantidades numéricas.
- Alphanumeric indicators: Para la visualización de expresiones alfanuméricas como el puerto serial del Arduino
- Botones: Para la entrada de valores lógicos.
- Graph: Para representar gráficamente los datos.

*IV-D2. Diagrama de bloques:* Las estructuras, los VI y las funciones de la paleta Funciones mostrada en el anexo figura 85 (denominados colectivamente como "nodos") brindan funcionalidad a los VI. Cuando se agregan nodos a los bloques, es posible conectarse entre sí o a los terminales creados por controles e indicadores en el panel de control, usando la opción de cableado en el panel de herramientas. Finalmente, un diagrama de bloques completo es similar a un diagrama de flujo [9].

Los nodos que se emplearon en el diagrama de bloque son:

- Funciones Numéricas: Se observan las funciones numéricas en el anexo figura 86, que se pueden usar para crear y realizar diversas operaciones matemáticas.
- Funciones lógicas: Ejecutan operaciones lógicas como: y, o, o exclusivo, negación, etc. Como se muestra en anexo figura 87.
- Funciones de hileras: Son usadas para concatenar dos o más cadenas, extraer una secuencia de caracteres, buscar y reemplazar uno o más caracteres, convertir datos numéricos en cadenas, formatear cadenas para su uso en procesadores de texto u hojas de cálculo [9], a continuación en el anexo figura 88 se muestra las funciones de hileras de LabVIEW.
- Funciones de comparación: Comparan valores lógicos, de hileras de caracteres, numéricos, y de clúster. En el anexo figura 89 se muestran las funciones de comparación que encontramos en LabVIEW.

#### IV-E. NI Vision and Motion

*IV-E1. NI IMAQ:* Es un conjunto de elementos que controlan el software de adquisición de imágenes y también tienen formas para realizar tareas que van desde la inicialización de la cámara, acciones simples y secuenciales hasta adquisición avanzada con velocidades altas [15].

*IV-E2. IMAQ VISION:* Es la biblioteca VI, se utiliza para el diseño de aplicaciones de visión artificial porque tiene elementos específicos para implementar sistemas de visión [15]. Como se observa en la figura 6 se muestran las siguientes funciones:

- Utilidades de visión: permite la creación de imágenes, calibrarlas, cambiar su tamaño, mostrarlas en pantalla, manipular píxeles y guardar los archivos en otro tipo de formato.
- Procesamiento de imágenes: Permite que las aplicaciones procesen imágenes mediante procesos de análisis como histogramas, análisis de partículas, umbralización y filtros.
- Visión por máquina: Este elemento permite realizar tareas como la identificación del número de objetos, identificación de bordes, la coincidencia de patrones y la medición de ángulos y distancias.

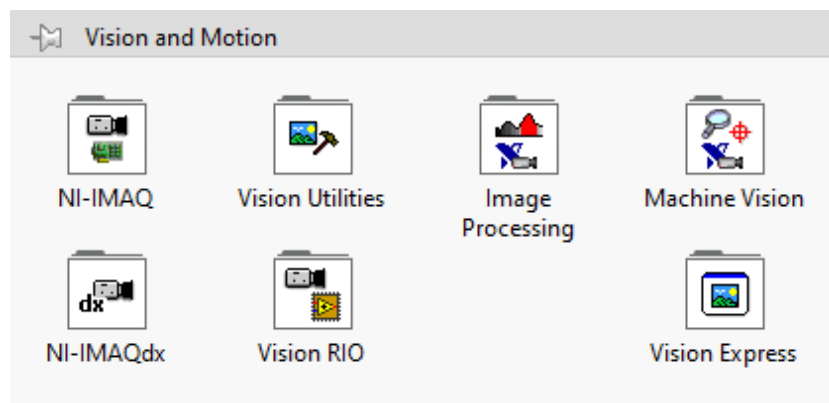


Figura 6. Funciones de visión artificial en LabVIEW [15].

*IV-E3. Vision Assistant:* Es un ayudante como su nombre lo indica, el cual permite ejecutar estrategias en programas sin ninguna programación, dado que genera automáticamente el código, en LabVIEW se representa su icono como se muestra en la figura 7. Sin embargo, luego de poner en marcha al ayudante, se deben incluir algunos elementos adicionales al código para aumentar el rendimiento del programa creado. El ayudante proporciona el ingreso a todas las bibliotecas, ya sea de LABVIEW, IMAQ Vision o NI IMAQ, lo que reduce las etapas iniciales del desarrollo de aplicaciones. Además, el este ayudante posibilita en gran manera el proceso de calibración de la imagen [9].

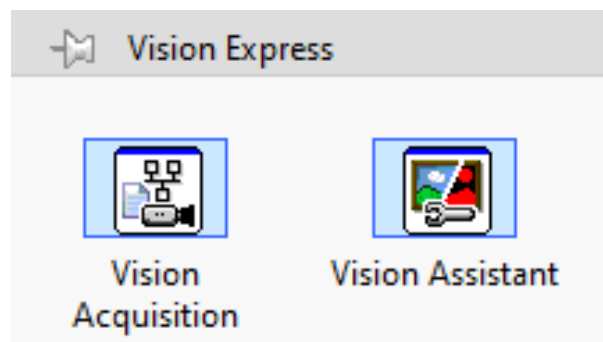


Figura 7. Bloques Vision Express [15].

#### IV-F. LINX

LINX es un programa de código abierto de Digilent creado para simplificar el crecimiento en aplicaciones integradas usando LabVIEW. LINX como se aprecia en la figura 8, introduce Vis para más de 30 de los sensores integrados, así como API de hardware de diagnóstico para acceder a componentes como E/S digital como MakerHub, E/S analógica, SPI, PWM, UART y I2C. El uso de LINX permite el uso de tarjetas Arduino, Raspberry Pi 2/3, Sparkfun, chipKIT o BeagleBone Black como tarjetas de adquisición de datos a través de conexiones Wi-Fi, Ethernet o USB/Serial. La descarga se realiza a través de LabVIEW VI Package Manager [14].

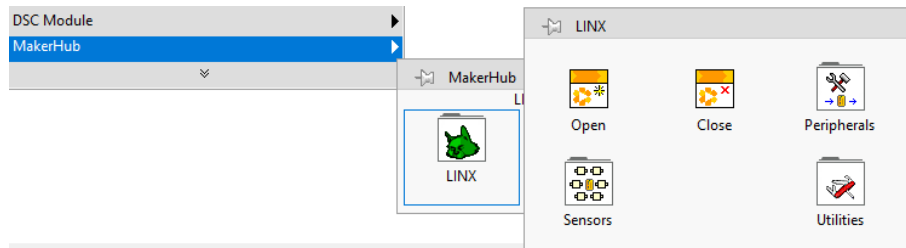


Figura 8. Funciones para abrir y cerrar comunicación serial; periféricos, sensores del Toolkit de LINX - LabVIEW Makerhub [14].

#### IV-G. Sensor de distancia infrarrojo SHARP GP2Y0A21

El sensor Sharp GP2Y0A21, como se observa en la figura 9, facilita la obtención de la medida entre el sensor y el objeto en un rango de 10 a 80 cm. el cual integra dispositivos como: detector sensible a la posición (PSD), diodo emisor de luz infrarroja (IRED) y circuito de procesamiento de señal [16].



Figura 9. Sensor de distancia infrarrojo SHARP GP2Y0A21 [16].

El sensor proporciona una salida de tensión con respecto a la distancia de detección. Es posible implementarlo como sensor de aproximación mediante un comparador o por medio del software. Debido al uso de métodos de detección basados en triangulación, las diferencias en la reflectividad del material, así como la temperatura de funcionamiento, no afectan significativamente el funcionamiento del sensor [16].

Para usar sensores analógicos en LabVIEW, la biblioteca LINX como se observa en la figura 10, proporciona una función que define un pin AI para leer señales de voltaje analógicas. Este módulo muestra la salida llamada "Voltage" como un tipo de dato double de 64 bits.



Figura 10. Bloques de E/S analógicas en LINX [14].

#### IV-H. Arduino UNO

Como se muestra en la figura 11 es una placa basada en el microcontrolador ATmega328P que fue diseñada para ser una plataforma de aprendizaje fácil de usar para la codificación, la computación física y la creación de prototipos de proyectos. La placa tiene 20 pines de E/S que pueden ser utilizados como entradas o salidas digitales para el microcontrolador (ATmega328). Esto incluye los pines marcados como analógicos, que se configuran y utilizan de la misma manera que los pines digitales. Estos son los que se conectan a los potenciómetros, sensores, etc [1].

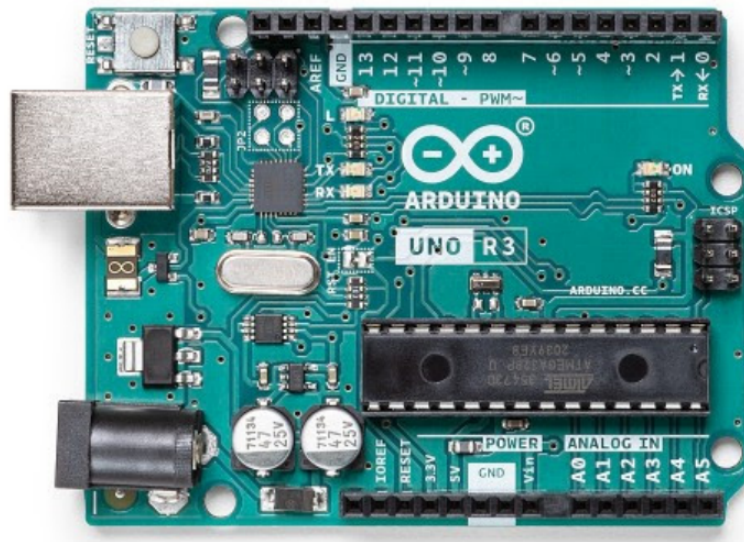


Figura 11. Tarjeta Arduino UNO R3 [1].

Para la comunicación serial por USB entre la placa y LabVIEW se necesita tener instalada la librería de LINX - LabVIEW MakerHub [14].

#### IV-I. Servomotor

Un servomotor como se observa en la figura 12, es un elemento electrónico implementado para controlar el desplazamiento angular desde cualquier rango de operación, el cual está formado por un motor, engranes, una tarjeta de control y tres cables, dos de alimentación que se conecta al Vcc y al Gnd y el tercero se conecta a la señal que aplicará un tren de pulsos para su control [12].



Figura 12. Servomotores marca TowerPro [12].

El servomotor funciona debido a la regulación del ancho de pulso (PWM) que permite que un circuito de control diferencial interno coloque el servomotor en el lugar deseado, como se representa en la figura 13. La frecuencia utilizada para enviar el tren de pulsos al servomotor es de 50 Hz, lo que representa que cada ciclo dura 20 milisegundos. El tiempo de cada pulso se interpreta como comandos de posicionamiento del motor y se desprecia el espacio entre cada pulso [12].

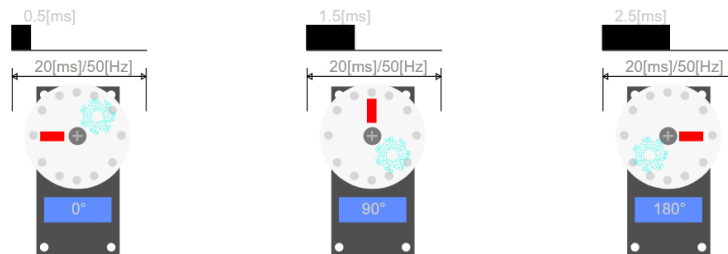


Figura 13. Modulación de ancho de pulso vs. posición angular del eje en un servomotor [12].

En términos generales, un pulso entre 0,5 y 1,0 ms corresponde a 0°, un pulso de 1,5 ms corresponde a 90° y un pulso entre 2,0 y 2,5 ms corresponde a 180° [12].

Para poder operar servomotores en el entorno de LabVIEW, se requiere la librería LINX, que proporciona los siguientes bloques mostrados en la figura 14; el primero permite configurar uno de los pines de la tarjeta de adquisición de datos para que envíe las señales adecuadas para operar los servomotores. El segundo recibe la longitud del ancho de pulso de la señal de control del servomotor como entrada y la aplica a los pines configurados en el módulo uno. El tercero permite liberar un pin configurado como salida de servomotor, por lo que puede ser utilizado como entrada/salida digital [14].

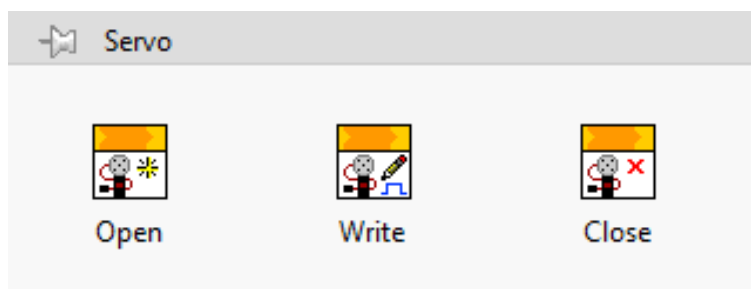


Figura 14. Bloque de servo en LINX [14].

Otra alternativa es mostrada en la figura 15 donde permite configurar el % de ciclo de trabajo de la señal PWM y enviarlo al pin de la tarjeta de adquisición de datos conectado al servomotor [14].

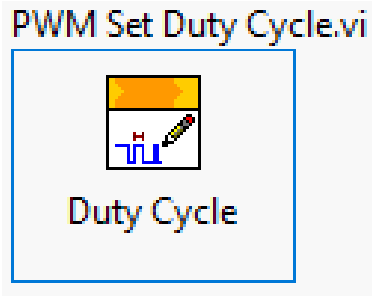


Figura 15. Bloque de PWM en LINX [14].

#### IV-J. Estabilidad de sistemas

Un sistema es estable cuando ante entradas acotadas sus salidas también se mantienen acotadas; es decir, su salida no tienden al infinito cuando los cambios en sus entradas son finitos [17]. Este concepto corresponde a la estabilidad absoluta, dividiendo a los sistemas en estables o inestables. A continuación, se aplicará el criterio de estabilidad a sistemas de segundo orden como el mostrado en la ecuación 1.

$$\frac{dy^2(t)}{dt^2} + 2\zeta\omega_n \frac{dy(t)}{dx} + \omega_n^2 y(t) = k\omega_n^2 u(t) \quad (1)$$

donde  $y(t)$  representan la salida y  $u(t)$  representa la entrada para el sistema. La función de transferencia del sistema asociado con la ecuación 1 se muestra en la ecuación 2.

$$G(s) = \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (2)$$

Los polinomios del denominador de la función de transferencia son definidos como polinomios característicos  $J(s)$ , de los cuales sus raíces (respuestas a la ecuación característica  $J(s) = 0$ ) son los polos de las funciones de transferencia a las que corresponden. En este caso particular, en los sistemas de segundo orden las raíces pueden ser reales o complejas [7]. Los parámetros de la función de transferencia son:

- $k$  es la ganancia estática, que se define como el cociente entre la magnitud de la salida del sistema en estado estable y la magnitud de la entrada escalón aplicada al mismo. También se puede calcular aplicando el límite cuando  $s \rightarrow 0$  en la función de transferencia, obteniéndose  $G(0) = k$  [17].
- $\zeta$  es el factor de amortiguamiento relativo del sistema, que establecerá la forma a la respuesta transiente. De acuerdo a su valor, los sistemas se pueden clasificar de la siguiente forma [17]:
  - $\zeta < 0 \rightarrow$  sistema inestable.
  - $\zeta = 0 \rightarrow$  sistema críticamente estable o no amortiguado.
  - $0 < \zeta < 1 \rightarrow$  sistema subamortiguado.
  - $\zeta = 1 \rightarrow$  sistema críticamente amortiguado.
  - $\zeta > 1 \rightarrow$  sistema sobreamortiguado.
- $\omega_n$  representa la frecuencia resonante no amortiguada, la cual está dada por la frecuencia en la que oscilaría el sistema sin amortiguamiento ( $\zeta = 0$ , respuesta sinusoidal) [17].

El resultado en el tiempo ante una entrada escalonada de magnitud  $U_0$  ( $U(s) = \frac{U_0}{s}$ ) puede obtenerse mediante la ecuación 3:

$$Y(s) = G(s) * U(s) = \left( \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) \left( \frac{U_0}{s} \right) \quad (3)$$

Se detalla los diferentes sistemas dependiendo el valor de  $\zeta$ :

- Sistema críticamente estable: Cuando se presenta un coeficiente de amortiguamiento relativo,  $\zeta = 0$  como se aprecia en la figura 16, es fácil verificar que los polos del polinomio característico resultante son conjugados

imaginarios puros [17]. Lo cual indica que se encuentra en el límite de su estabilidad y el resultado siempre se representa como una oscilación de amplitud fija:

$$y(t) = kU_0[1 - \cos \omega_n t], t \geq 0 \quad (4)$$

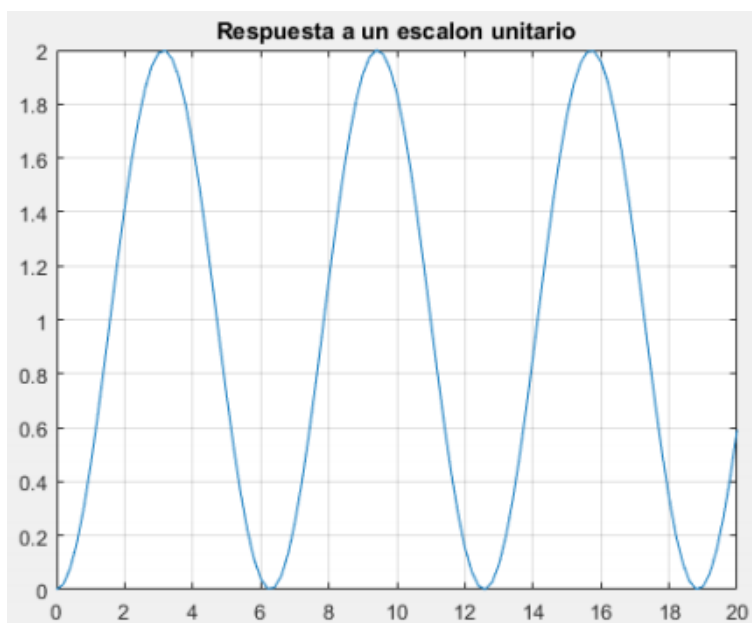


Figura 16. Respuesta escalón de un sistema con  $\zeta = 0$  [7].

- Sistema subamortiguado: En este caso de  $0 < \zeta < 1$ , los polos (raíces de las ecuaciones características) del sistema obtenido son conjugados complejos,  $s_1 = -\zeta\omega_n - j\omega_n\sqrt{1 - \zeta^2}$  y  $s_1^* = -\zeta\omega_n + j\omega_n\sqrt{1 - \zeta^2}$  puesto que la respuesta temporal ante entrada escalón la ecuación 5:

$$y(t) = kU_0 \left[ 1 - e^{-\zeta\omega_n t} \left( \cos \omega_d t + \frac{\zeta}{\sqrt{1 - \zeta^2}} \sin \omega_d t \right) \right], t \geq 0; \quad (5)$$

en el que  $\omega_d = \omega_n\sqrt{1 - \zeta^2}$  es la frecuencia resonante amortiguada. Como se observa en la figura 17, las componentes complejas en los polos producen un resultado temporal con presencia de funciones como cosenos y senos que producen oscilaciones que decaen con una tendencia exponencial.

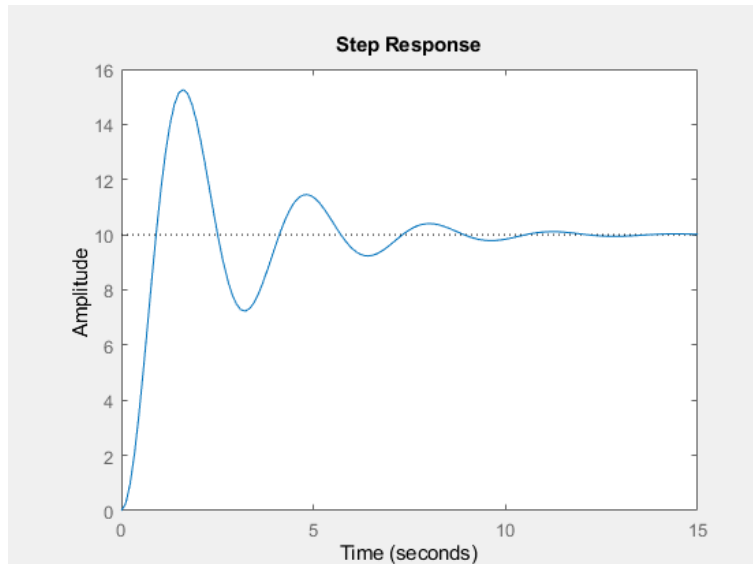


Figura 17. Respuesta escalón de un sistema con  $\zeta = 0,2$  y  $\omega_n = 2$ .

- Sistema críticamente amortiguado: Cuando  $\zeta = 1$  los polos del sistema son reales e iguales. La respuesta escalonada se observa en la figura 18.

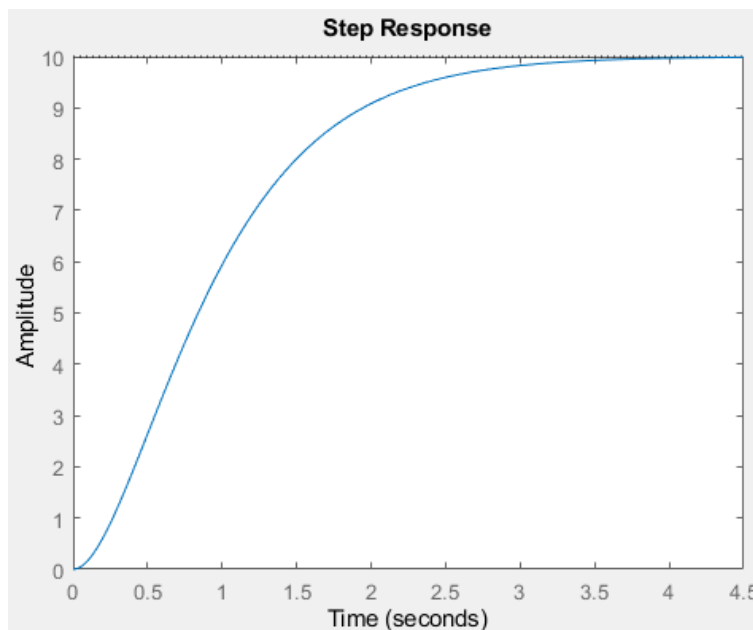


Figura 18. Respuesta escalón de un sistema con  $\zeta = 1$  y  $\omega_n = 2$

- Sistema sobreamortiguado: Es cuando el factor de amortiguamiento relativo  $\zeta > 1$ , en este caso los polos del sistema son reales,  $s_1 = -\frac{1}{\tau_1} = -\zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1}$  y  $s_2 = -\frac{1}{\tau_2} = -\zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1}$ . La función de transferencia se expresa en la siguiente ecuación 6:

$$G(s) = \frac{k}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad (6)$$

En el cual  $\tau_2$  representa la constante de tiempo asociada al polo con mayor cercanía al eje imaginario, lo que



resulta en la exponencial más lenta  $\tau_2 > \tau_1$ , se puede observar en la figura 19, como la respuesta no oscila, sin embargo, presenta un comportamiento similar a un sistema de primer orden.

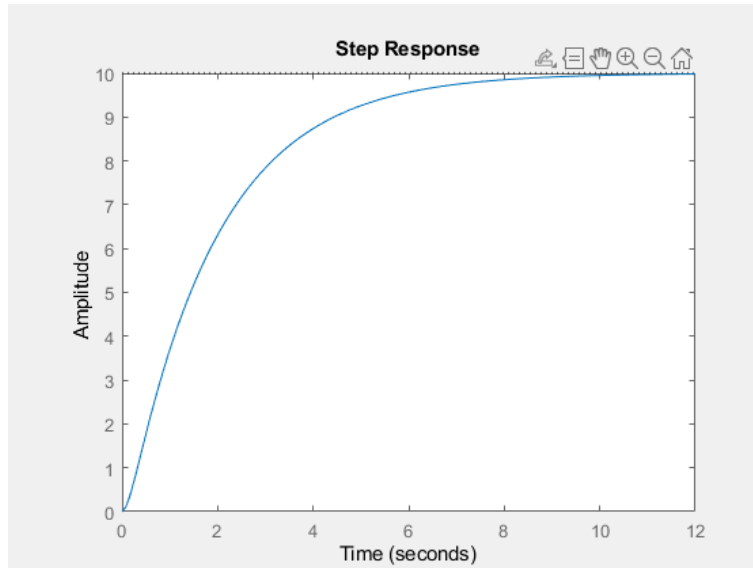


Figura 19. Respuesta escalón de un sistema con  $\zeta = 2$  y  $\omega_n = 2$

Un estudio interesante consiste en analizar las posiciones de los polos del sistema de segundo orden en función de los parámetros característicos de la función de transferencia en los casos sub amortiguado y sobreamortiguado. A partir de estos parámetros, se pueden obtener las relaciones mediante algunas características específicas a la respuesta temporal del sistema a una entrada de paso, tales como:

- Sobreoscilación (SO(%)): Expresa el sobrenivel o overshoot del sistema en %, que es la diferencia entre el valor pico máximo de la respuesta y el valor de estado estable, relativo a dicho valor de estado estable (en %)[17], es decir:

$$SO(\%) = \frac{y_{max} - y_{ss}}{y_{ss}} * 100\%$$

Para el caso sub amortiguado el valor de sobreimpulso, derivando la ecuación 5 e igualando a 0, se obtiene la siguiente ecuación 7:

$$SO(\%) = (100\%)e^{\left(\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}\right)} \quad (7)$$

- Tiempo de pico ( $t_p[s]$ ): El tiempo que demora la respuesta del sistema en obtener el valor máximo desde el momento en que se introduce el paso [17].

Este parámetro para el caso sub amortiguado, viene dado por:

$$t_p = \frac{\pi}{\omega_d} = \frac{\pi}{\omega_n \sqrt{1-\zeta^2}} \quad (8)$$

- Tiempo de subida ( $t_s[s]$ ): Para el caso subamortiguado, es el tiempo transcurrido desde que la salida inicia a evolucionar hasta que logra por primera vez su valor de estado estable [5]:

$$t_s = \frac{\pi - \phi}{\omega_d} \quad (9)$$

con  $\zeta = \cos \phi$ , siendo  $\phi$  el ángulo formado por el polo complejo conjugado y el semieje real negativo. En un sistema sobreamortiguado, se determina por el tiempo que tarda la respuesta en cambiar entre el 10% y el 90% de su valor de estado estable [17].

- Tiempo de establecimiento ( $t_e[s]$ ): El tiempo desde que la salida del sistema comienza a variar hasta que la respuesta se estabiliza en alrededor del 2% de su valor de estado estable [17]:

$$t_{ss} = \frac{4}{\zeta\omega_n} \quad (10)$$

Se puede analizar en la figura 20 los efectos que se obtienen con la variación de la localización de los polos en un sistema de segundo orden subamortiguado con una expresión en función de la parte real e imaginaria de las raíces [5], donde  $|\sigma| = \zeta\omega_d$  con  $\omega_d = \omega_n\sqrt{1 - \zeta^2}$ , como se muestra en la ecuación 11:

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{\omega_d^2 + \sigma^2}{s^2 + 2\sigma s + (\omega_d^2 + \sigma^2)} \quad (11)$$

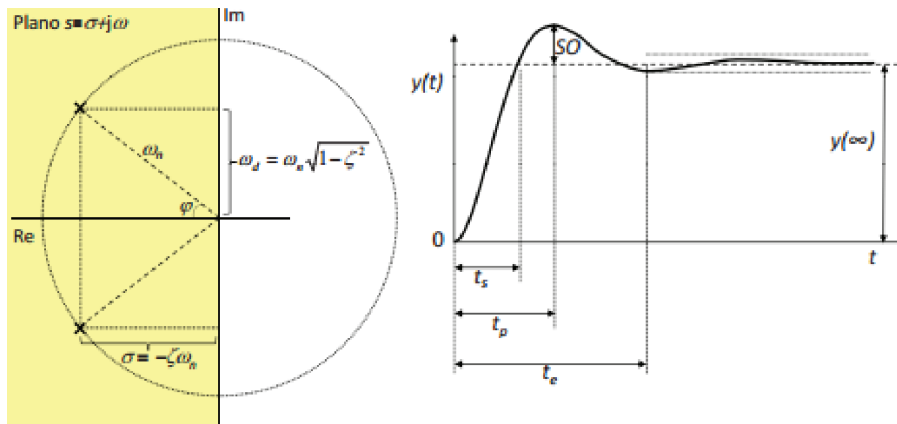


Figura 20. Parámetros de la respuesta escalón de un sistema de segundo orden subamortiguado [7].

#### IV-K. Linealidad de sistemas

Se dice que un sistema es lineal si satisface las propiedades de proporcionalidad y aditividad, que se conocen colectivamente como el principio de superposición. Proporcional significa que cuando la entrada del sistema se multiplica por un factor, la salida del sistema también se ve afectada en el mismo grado como se muestra en la figura 21. Por otro lado, el sistema es aditivo, si la entrada es el resultado de la suma de dos entradas como se muestra en la figura 22, la salida será el resultado de la suma de las salidas producidas por cada una de esas entradas individualmente [5].

- Propiedad de proporcionalidad

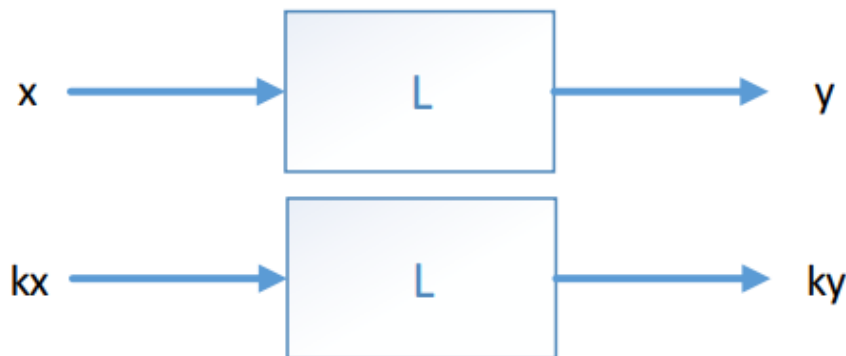


Figura 21. Propiedad de proporcionalidad [7].

- Propiedad de Aditividad

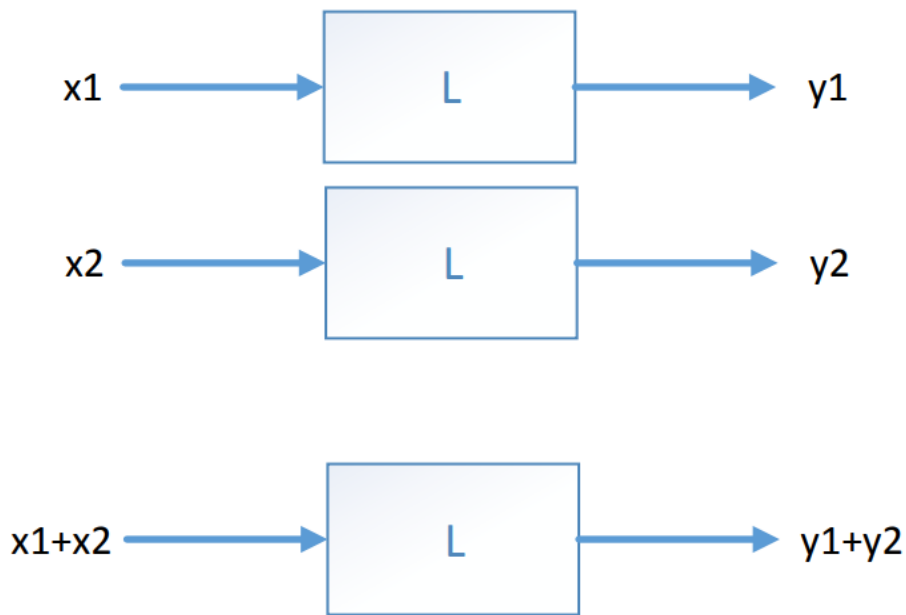


Figura 22. Propiedad de aditividad [7].

Al satisfacer la unión entre las propiedades se logra el principio de proporcionalidad como se muestra en la figura 23:

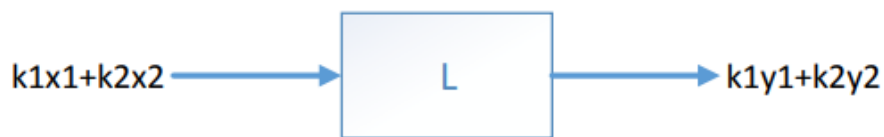


Figura 23. Principio de proporcionalidad [7].

#### IV-L. Elementos de un sistema de control

*IV-L1. Planta:* Conjunto de componentes que interactúan entre sí para lograr un objetivo específico [5].

*IV-L2. Referencia:* Entrada que indica como se desea que se comporte la planta y es manipulable [5].

*IV-L3. Perturbación:* Es una señal que tiende a afectar negativamente la salida de la planta y que no son manipulables [5].

*IV-L4. Controlador:* Se encarga de generar las señales de mando para que el comportamiento de la salida de la planta se asemeje al comportamiento de la referencia [5].

*IV-L5. Retroalimentación:* Operación que permite comparar la diferencia entre la salida de un sistema y el valor de entrada de referencia, que es la información requerida por el controlador [5].

#### IV-M. Controlador PID

La lógica de control PID se usa ampliamente en la industria de control de procesos. Los controladores PID han sido elegidos tradicionalmente por los ingenieros de sistemas de control debido a su flexibilidad y confiabilidad. Un controlador PID tiene términos proporcionales, integrales y derivativos, que se pueden representar en forma de una función de transferencia [17], como se muestra en la ecuación 12.

$$G_C(s) = K_p + \frac{K_i}{s} + K_d s \quad (12)$$

Donde  $K_p$  se define como la ganancia proporcional,  $K_i$  se define como la ganancia integral y  $K_d$  como la ganancia diferencial. Al ajustar dichas ganancias del controlador PID se puede generar acciones necesarias para que la salida del sistema cumpla con las especificaciones deseadas del proceso [17].

El término proporcional produce un cambio de salida proporcional al error actual. Este término proporcional se refiere al estado actual de la variable de proceso. El término integral es proporcional tanto a la magnitud del error como a la duración del error. La acción derivativa consiste en anticipar el comportamiento de la señal de error que recibe el PID [5].

#### IV-N. Control por realimentación de estado

Las variables de estado son los subconjuntos más pequeños de variables del sistema que puede representar todo su estado dinámico en un momento dado. Dichas variables de estados serán linealmente independientes; la variable de estado no debe ser una combinación lineal de otras variables de estado. El mínimo número de variables de estado  $n$  necesitadas para representar un sistema dado, suele ser igual al orden de las ecuaciones diferenciales que definirán el sistema [7].

Se entiende por un espacio de estados cuyo eje son las  $n$  variables de estado seleccionadas. Por ende, cualquier estado podrá ser representado por un vector de  $n$  coordenadas en este espacio [5]. La manera común de mostrarlo será por medio de un diagrama de bloques, como se puede ver en la figura 24.

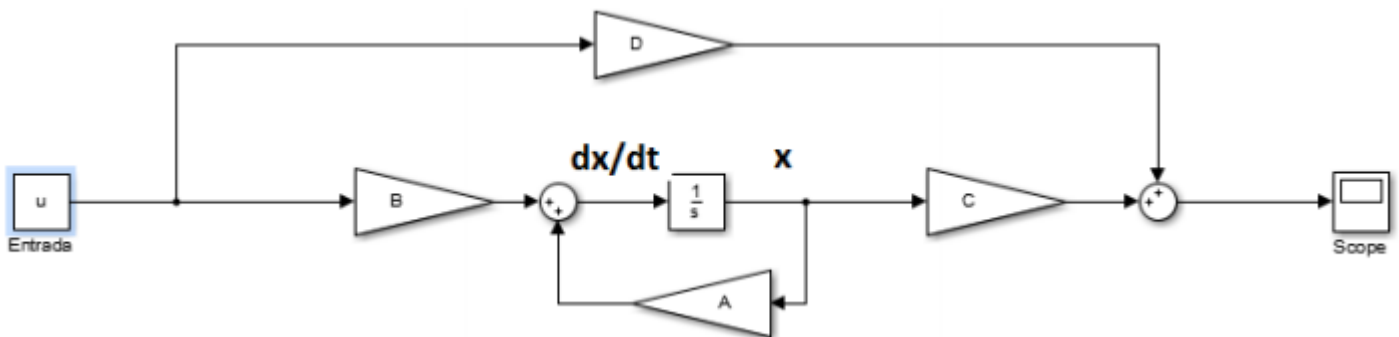


Figura 24. Representación típica de un sistema en el espacio de estados [7]

Acorde a cada una de las letras que aparecen en el diagrama de bloques [17], consisten en:

- **Vector de estados (x):** Es un conjunto de variables de estado definidas en el sistema.
- **Vector de entradas (u):** Conjunto de variables de entrada.
- **Vector de salidas (y):** Conjunto de variables de salida.
- **Matriz de estados (A):** Contiene información que asocia el estado con sus cambios.
- **Matriz de entrada (B):** Consiste en la relación entre los cambios de estado y las entradas.
- **Matriz de salida (C):** Contiene la relación entre el estado y la salida.
- **Matriz de transmisión directa (D):** Relación directa entre entrada y salida. La mayoría de los sistemas que se modelan carecen de esta relación, y para el caso de bola y viga  $D=0$ .

Del diagrama de bloques de la figura 24 se puede deducir la expresión matricial de un sistema en espacio de estados representado por las ecuaciones 13 y 14.

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (13)$$

$$y(t) = C(t)x(t) + D(t)u(t) \quad (14)$$

La estabilidad del sistema puede ser analizada por los autovalores de la matriz  $A$ . Para calcular estos valores propios, se resuelve la siguiente ecuación:

$$\lambda(s) = |sI - A| = 0 \quad (15)$$

donde  $\lambda(s)$  es el polinomio característico y sus raíces son los polos del sistema [17].

IV-N1. *Controlabilidad*: Un sistema se define que es controlable cuando, partiendo del estado en tiempo inicial, a través de un conjunto de entradas aceptables, se logra obtener un estado final dentro de un rango de tiempo finito [7].

Un sistema continuo invariante en el tiempo es controlable si se cumple la siguiente ecuación 16 [5]:

$$\text{rg}(R) = \text{rg} [B \ AB \ \dots \ A^{n-1}B] = n \quad (16)$$

donde:

$R$ : matriz de controlabilidad.

$\text{rg}(R)$ : número de columnas independientes de  $R$ .

$n$ : cantidad de filas de  $A$ .

IV-N2. *Forma canónica controlable*: A partir de una función de transferencia estrictamente propia que se muestra en la ecuación 17, se puede escribir la representación en espacio de estados del mismo sistema usando las ecuaciones :

$$Y(s) = \frac{b_1 s^{n-1} + \dots + b_{n-1} s + b_n}{s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} U(s) = \frac{B(s)}{A(s)} U(s) = P(s) U(s). \quad (17)$$

Los coeficientes ahora se pueden insertar directamente en el modelo de espacio de estado usando [8]:

$$\dot{x}(t) = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{n-1} & -a_n \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t) \quad (18)$$

$$y = [b_1 \ b_2 \ \dots \ b_{n-1} \ b_n] x(t) \quad (19)$$

Se puede observar que los coeficientes de la ecuación 17 y de las ecuaciones 18 y 19 son los mismos. Esta implementación del espacio de estado se denomina Forma Canónica Controlable porque garantiza que el modelo resultante sea controlable [8].

IV-N3. *Asignación de polos por realimentación de estados*: Una de las formas de proyectar la acción de control para obtener la dinámica de lazo cerrado deseada es ubicar los valores propios de la matriz  $A$  en la región deseada, siempre que el sistema sea controlable [3].

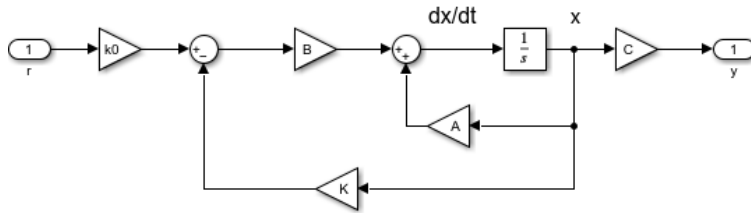


Figura 25. Diagrama de bloques de un sistema realimentado en el espacio de estados.

Se observa que la primera fila de la matriz  $A$  de la ecuación 18 pertenece al polinomio característico del sistema (el denominador de la función de transferencia), recordando que el polinomio característico se encarga de definir los autovalores. Si un sistema lineal de  $n$  dimensiones es controlable, entonces hay una retroalimentación de estado que ubica arbitrariamente los valores propios de la matriz  $(A - BK)$  [3].

El lazo cerrado viene dado por [8]:

$$\dot{x}(t) = \begin{bmatrix} -(a_1 + k_1) & -(a_2 + k_2) & \dots & -(a_{n-1} + k_{n-1}) & -(a_n + k_n) \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} k_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} r(t) \quad (20)$$

Su ecuación característica por:

$$\Delta_f = s^n + (a_1 + k_1)s^{n-1} + \dots + (a_{n-1} + k_{n-1})s + (a_n + k_n) = 0 \quad (21)$$

## V. MARCO METODOLÓGICO

### V-A. Circuito electrónico del módulo didáctico

En la figura 26 se observa el esquema de conexiones electrónicas necesarias para la comunicación serial entre Arduino y Labview. El sensor infrarrojo conectado al pin analógico A0, así como los servomotores de inclinación y apertura/cierre conectados a los pines PWM 3 y 5, respectivamente.

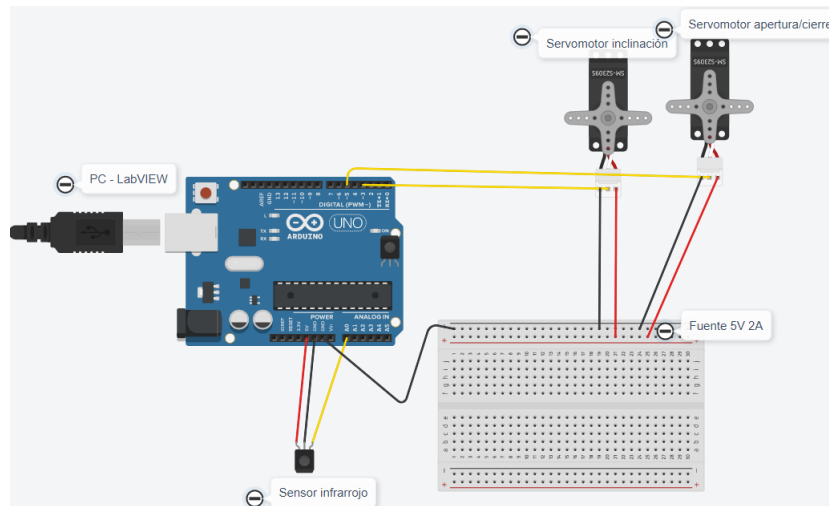


Figura 26. Conexiones del Arduino UNO con la PC, sensor infrarrojo y servomotores.

### V-B. Diseño mecánico de la planta

Como se observa en la figura 27, se presenta el diseño de la planta ensamblada, la cual representa un sistema de tipo de viga bola con la variante de poseer un sistema de apertura en las vigas, la cual permite tener dos grados de libertad, cumpliendo con el funcionamiento de realizar la clasificación en el sistema. Cuyos planos se encontraran en los anexos.

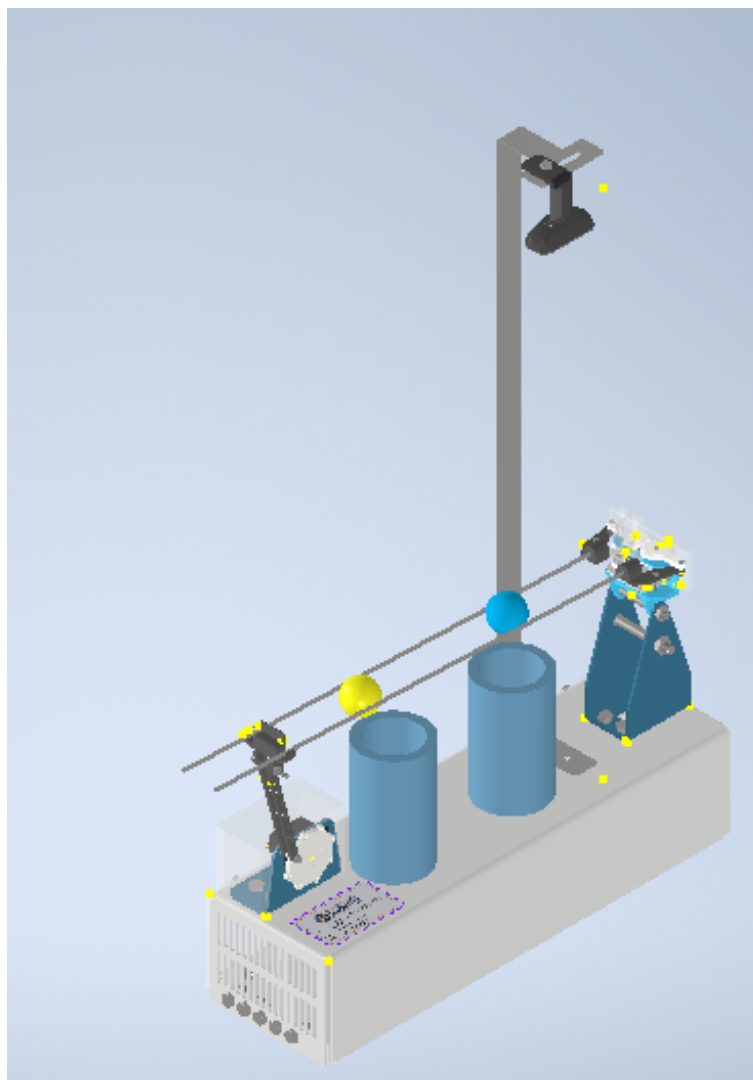


Figura 27. Diseño de planta, por C. Granda y C. Mora, Inventor Autodesk, 2020.

#### V-C. Construcción de planta

La planta se ha construido mediante el uso de tres materiales, los cuales son acero inoxidable, acrílico y PLA. Para la construcción de cada pieza se utilizaron diferentes tipos de procedimiento de manufactura, los cuales fueron corte láser por oxígeno, corte por plasma y por impresión en 3D.

*V-C1. Piezas metálicas:* Se construyeron los soportes y base principalmente de acero inoxidable de 1,2 mm para las bases. Sin embargo, el soporte de la cámara se construyó en acero inoxidable de 2 mm. esta decisión se tomó porque estos elementos dado que son las que van a soportar la mayor carga del sistema y es necesario que sean rígidas y con alta vida útil, como se puede apreciar en la figura 28, se presenta las piezas construidas en acero, el diseño de estas piezas se realizaron en Inventor Autodesk 2020, los planos se encontraran dentro de los anexos.

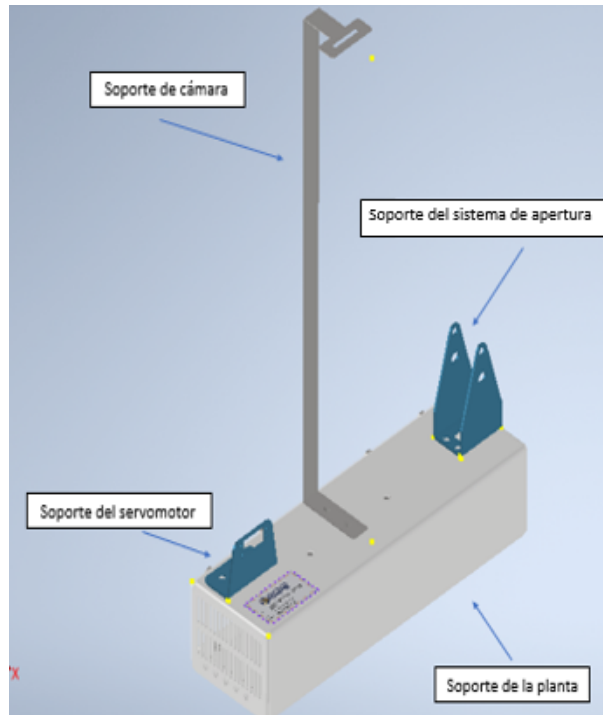


Figura 28. Piezas metálicas, por C. Granda y C. Mora, Inventor Autodesk, 2020.

las piezas fueron diseñadas con la herramienta de chapa metálica de Inventor Autodesk, una vez elaborado los diseños de las piezas, se procede a poner los contorno en 2d, como se muestra en la figura 29

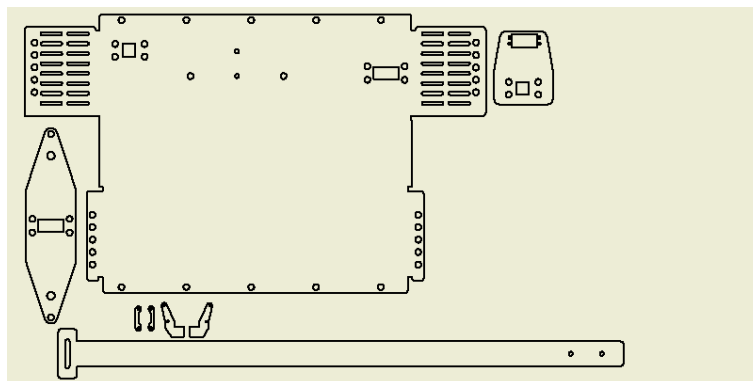


Figura 29. Plano de las piezas metálicas a cortar por láser usando oxígeno, por C. Granda y C. Mora, Inventor Autodesk, 2020.

Para un buen acabado y definición se realizó mediante el uso de una cortadora láser, lo cual garantiza la exactitud en las medidas, terminado este proceso se procede a doblar las aristas correspondientes de cada pieza para dar la forma correspondiente.

*V-C2. Piezas en acrílico:* Para la construcción de las piezas en este material se utilizó un acrílico de 5 mm de espesor, que entre otras características es resistente, maleable y económico. En la figura 30, se observan las piezas que se diseñaron para ser construidas en acrílico, dichos diseños se encontrarán dentro de los anexos.



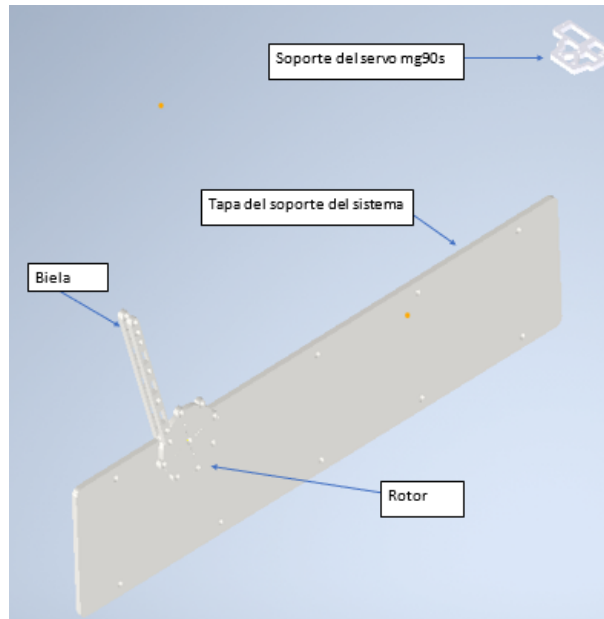


Figura 30. Piezas de acrílico, por C. Granda y C. Mora, Inventor Autodesk, 2020.

Una vez terminado los diseños se procede a crear un archivo de tipo dwg en inventor con los contornos de las piezas en un plano en escala 1:1 como se representa en la figura 31.

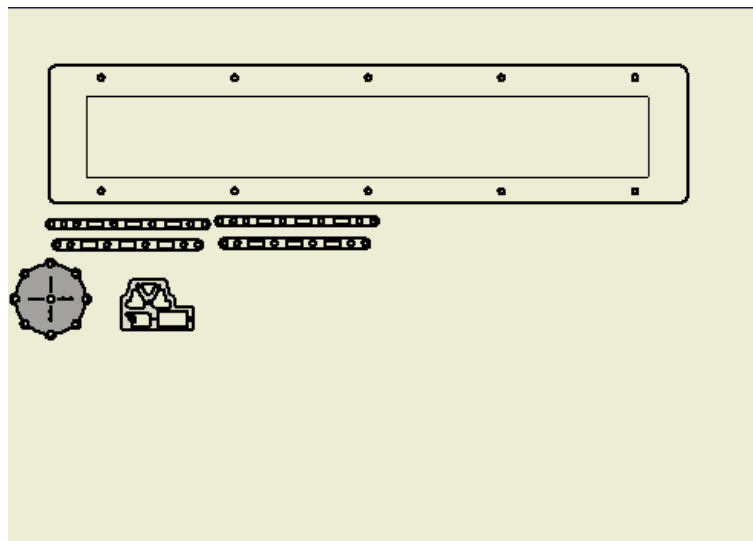


Figura 31. Plano de las piezas de acrílico a cortar por láser usando plasma, por C. Granda y C. Mora, Inventor Autodesk, 2020.

Para una buena definición se realizó mediante el uso de una cortadora láser, lo cual además de garantizar la exactitud en las medidas, resulta a bajo costo en comparación al corte láser del acero.

*V-C3. Piezas en PLA:* Para la construcción de las piezas con más detalle se implementó el uso de una impresora en 3D y el material usado fue el filamento PLA, las mismas fueron diseñadas por medio de Inventor Autodesk, como se muestra en la figura 32, los planos de las piezas se encontraran dentro de los anexos.

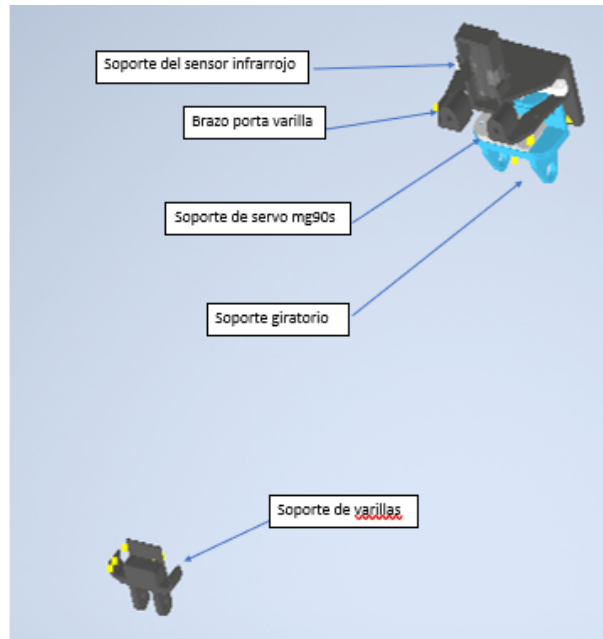


Figura 32. Piezas para impresión en 3D , por C. Mora y C. Granda, Inventor Autodesk, 2020.

Las cual fueron exportadas a formato STL como se puede observar en la figura 33.

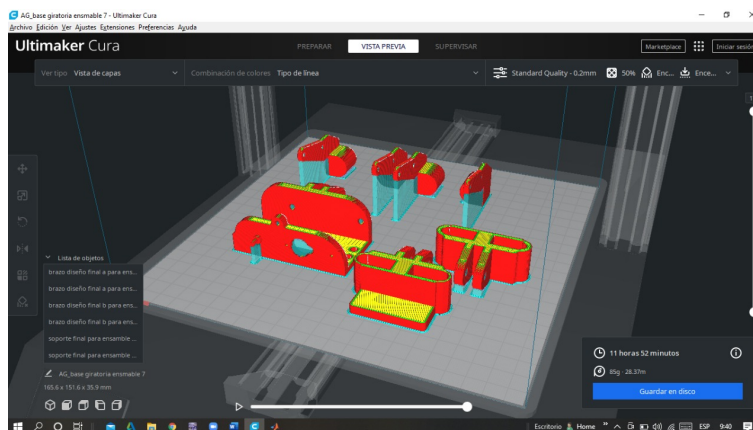


Figura 33. Piezas en formato STL para ser impresas por medio de impresora de 3D, por C. Granda y C. Mora, Inventor Autodesk, 2020.

*V-C4. Sistema de apertura:* El sistema de apertura, como se muestra en la figura 34, cumplirá con dos funciones, la primera es hacer de soporte para que la bola pueda desplazarse sobre la horizontal, dependiendo únicamente de la variable de entrada del sistema, la cual dictará el sentido del desplazamiento de la bola. Además, se implementó varillas huecas de aluminio de 8mm de espesor por 53 cm de largo, el principal motivo para usar esta longitud es dada por el sensor infrarrojo, dadas sus características, comienza a presentar errores en la medición al aumentar la distancia al pasar de los 38 cm, por esta razón se establece que el margen de trabajo para la realización del proceso de control del sistema.

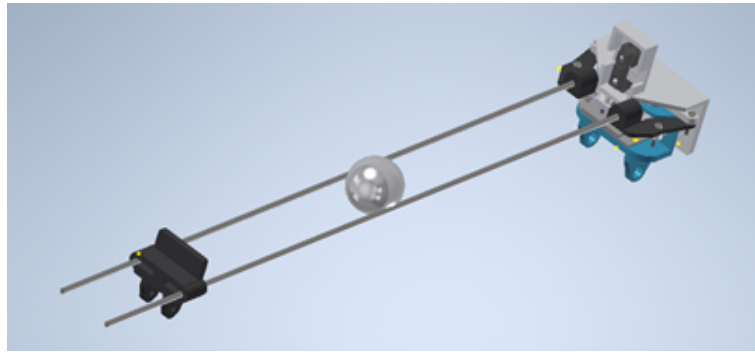


Figura 34. Sistema de apertura de las vigas, por C. Mora y C. Granda, Inventor Autodesk, 2020.

La segunda función del sistema antes descrito cumplirá con la acción de juntar y separar las varillas de aluminio, esto provocará que por gravedad la bola caiga sobre la posición elegida una vez realizada la identificación del color de la esfera, lo cual es realizado por medio de un servomotor conectado al sistema de engranajes, como se muestra en la figura 35.

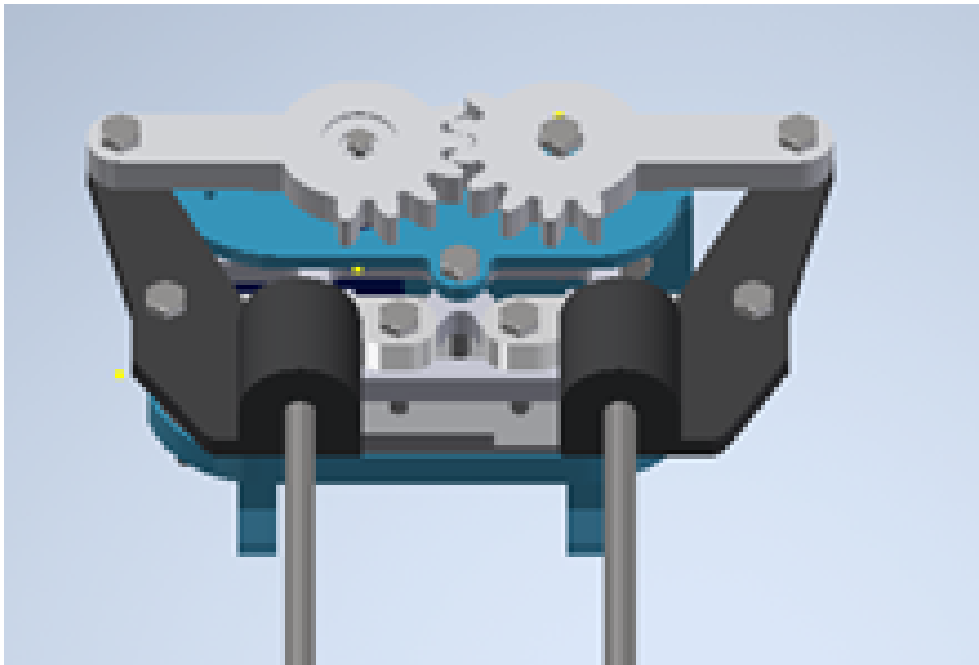


Figura 35. Sistema de engranajes, por C. Mora y C. Granda, Inventor Autodesk, 2020.

V-C5. *Ensamble*: Una vez finalizado los anteriores procesos tanto de corte como de impresión, se procede a ensamblar físicamente el sistema, una vez terminado los pliegues de los cortes en acero inoxidable y uniendo todos los componentes, se obtiene el siguiente módulo mostrado en la figura 36 el diseño del módulo clasificador de dos grados de libertad.



Figura 36. Módulo clasificador longitudinal de 2 grados de libertad, por C. Mora y C. Granda.

A continuación se muestra en la figura 37, los componentes de la planta mediante una vista explosionada.

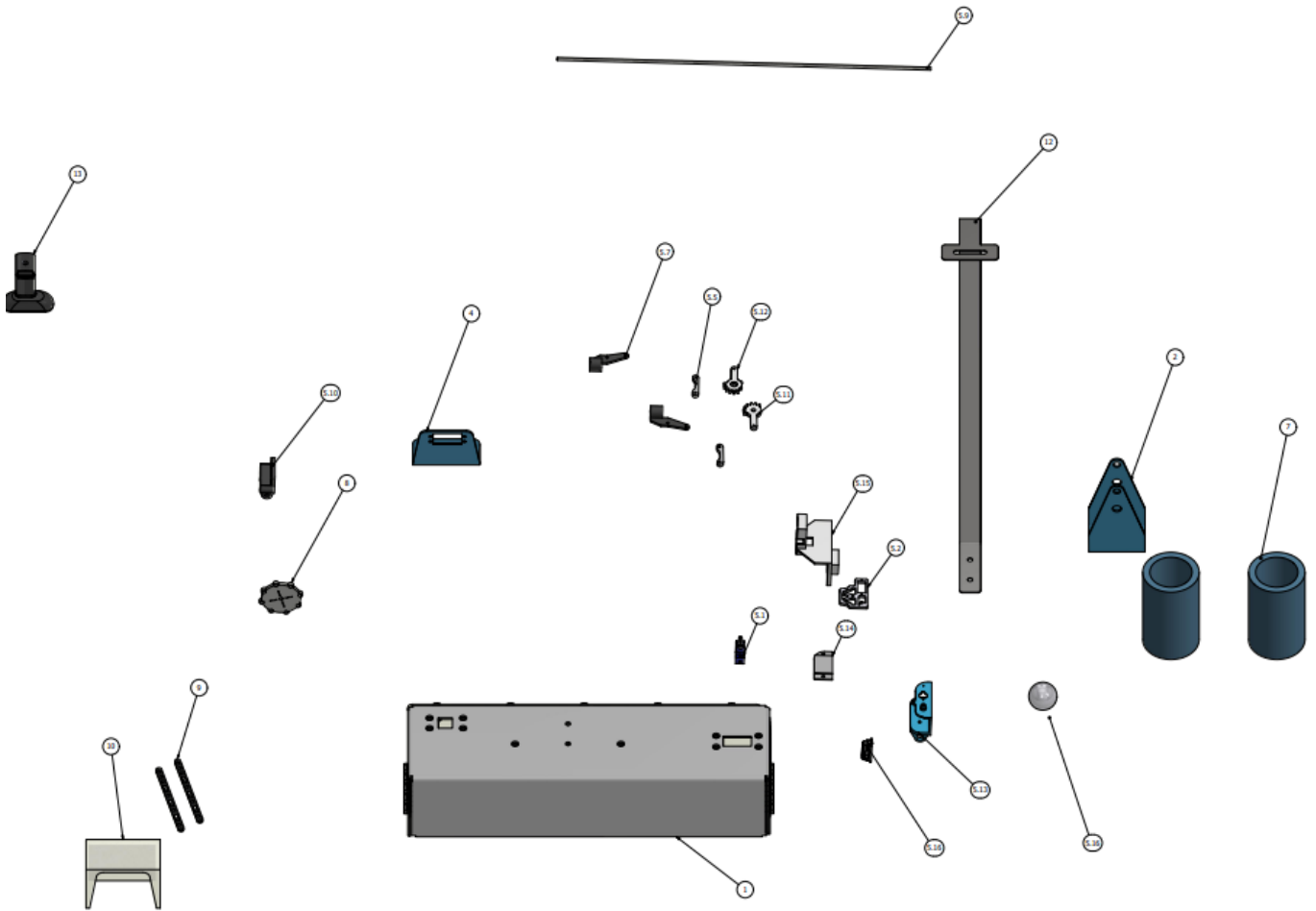


Figura 37. Plano explosionado del módulo por C. Granda y C. Mora, Inventor Autodesk, 2020.

Tabla I

TABLA DE ELEMENTOS DEL PLANO EXPLOSIONADO, POR C. GRANDA Y C. MORA, INVENTOR AUTODESK, 2020.

LISTA DE PIEZAS				
ELEMENTO	CTDAD	Nº DE PIEZA	DESCRIPCIÓN	PROCESO DE MANUFACTURA
1	1	Caja porta estructura	Acero inox. de 1.2 mm	Corte láser con oxígeno
2	1	Base porta sistema de garra chapa metálica	Acero inox. de 1.2 mm	Corte láser con oxígeno
3	1	Tapa de estructura	Acrílico de 5mm	Corte láser con plasma
4	1	Soporte porta servo mg90s	Acero inox. de 1.2 mm	Corte láser con oxígeno
5	1	Sistema de apertura	Generico	
5.1	1	Servo MG90s	Generico	
5.2	1	Estructura porta brazos y servo	Generico	
5.3	1	Componente del servomotor mg90s	Generico	
5.4	1	Componente del servomotor mg90s	Generico	
5.5	1	Sub brazo a	Acero inox. de 1.2 mm	Corte láser con oxígeno
5.6	1	Sub brazo b	Acero inox. de 1.2 mm	Corte láser con oxígeno
5.7	1	Brazo a porta varilla	Filamento PLA	Impresión 3D
5.8	1	Brazo b porta varilla	Filamento PLA	Impresión 3D
5.9	2	VARILLA	Aluminio de 8mm	
5.10	1	soporte porta varilla	Filamento PLA	Impresión 3D
5.11	1	Brazo b en forma de engranaje	Filamento PLA	Impresión 3D
5.12	1	Brazo a en forma de engranaje	Filamento PLA	Impresión 3D
5.13	1	Base giratoria	Filamento PLA	Impresión 3D
5.14	1	Soporte porta servo	Filamento PLA	Impresión 3D
5.15	1	Soporte porta sensor sharp	Generico	
5.16	1	Long Range SHARP IR Sensors - #GP2Y0A21YK0F	Generico	
5.17	1	ESFERA	Generico	
5.18	13	AS 1110 - M2,5 x 16	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
5.19	13	AS 1112 - M2,5(2)	Tuercas hexagonales ISO métricas, incluidas	
5.20	1	AS 1110 - M2,5 x 25	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
5.21	1	AS 1110 - M1,6 x 6	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
6	1	MG995	Generico	
7	2	Envase de pelota	Generico	
8	1	Rotor	Acrílico de 5mm	
9	2	Blela	Acrílico de 5mm	

10	1	CAJA DE ACRILICO PARA SERVO MG995	Acrilico de 5mm	
11	1	AS 1111 - M8 x 65	Tornillos y pernos ISO comerciales hexagonales métricos	
12	1	Soporte de carama	Acero inox. de 2mm	corte láser con oxígeno
13	1	Cámara	Generico	
14	1	AS 1111 - M10 x 70	Tornillos y pernos ISO comerciales hexagonales métricos	
15	31	AS 1112 - M8(2)	Tuercas hexagonales ISO métricas, incluidas	
16	1	AS 1112 - M10(2)	Tuercas hexagonales ISO métricas, incluidas	
17	30	AS 1110 - M8 x 20	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
18	2	AS 1110 - M6 x 20	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
19	1	AS 1110 - M6 x 12	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
20	1	AS 1112 - M5(2)	Tuercas hexagonales ISO métricas, incluidas	
21	1	AS 1110 - M4 x 40	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
22	1	AS 1110 - M4 x 20	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	
23	1	AS 1112 - M4(2)	Pernos y tornillos métricos ISO de precisión de cabeza hexagonal	

#### V-D. Modelo matemático

Si es desconocido el modelo matemático de la planta, es imposible usar el sistema y analizarlo. El modelo no tiene que ser único para cada sistema, ya que puede tener múltiples soluciones a modelos con el mismo mecanismo, dependiendo de los requisitos emergentes en temas de control.

Se expresará el modelo del sistema de dos maneras diferentes, tanto la función de transferencia y la representación del espacio de estados, posteriormente se utilizará MATLAB y Simulink para analizar los resultados y sacar conclusiones. Al analizar y construir el modelo matemático, se identifica las variables físicas a controlar; para nuestro prototipo son dos medidas, la posición de la pelota y el ángulo de giro, son las variables del sistema de ecuaciones que definirán al sistema y las coordenadas generalizadas,  $q_1$  y  $q_2$ .

Al hacer las ecuaciones matemáticas para el modelo, se comienza por simplificar el sistema, ya que es el eje de rotación del motor y la viga que mueve la bola, se encuentra perpendiculares entre sí, se cruzan. entonces con esto evita términos cuadráticos en algunas ecuaciones.

Además, como se explicara más adelante, se supondrá que la posición del servomotor está cerca del centro de masa de las vigas en relación con los otros componentes del sistema, por lo tanto, la energía potencial del sistema es independiente del ángulo de giro del motor.

Implementaciones de la función lagrangiana para el modelo de función de transferencia y el modelo de espacios de estados se basa en la formulación de lagrange, no en la ecuación movimiento newtoniano. La lagrangiana de un sistema es la diferencia entre La energía cinética y la energía potencial ( $L=T-V$ ) son, por lo tanto, un escalar.

Así, mientras que el modelo de Newton se centra en factores externos (fuerzas) que actúan sobre el cuerpo, lagrange maneja magnitudes asociadas al cuerpo (energía cinética y potencial) es importante aclarar porque hace que la lagrangiana del sistema no se ve afectado por los cambios de coordenadas. Esto hace posible saltar del espacio ordinario (donde las ecuaciones de movimiento pueden ser muy complejas) a un espacio de configuración seleccionado de forma simplificada.

Además, no es necesario determinar con precisión las interacciones que actúan sobre un cuerpo, esto a veces ni siquiera es posible.

Se comienza por la ecuación de lagrange, expresada por:

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_L} \right) - \frac{\partial L}{\partial q_L} + \frac{\partial F}{\partial \dot{q}_L} = Q_i \quad (22)$$

Siendo:

$L=T-V$ ; La ecuación lagrangiana, donde T es energía cinética y V es energía potencial.

$Q_i$ =fuerza generalizada

$q_i$ =cada coordenada generalizada.

F = fuerza disipativa de Rayleigh Las coordenadas generalizadas son:

$q_1$  es la posición de la bola a lo largo de las vigas.

$q_2$  es el ángulo de giro de las vigas con respecto al plano horizontal.

Se inicia calculando la energía cinética total del sistema, que será las vigas más la que aporta la esfera, las cuales son dos partes en movimiento.

$$T = (T_{VIGA} + T_{BOLA}) \quad (23)$$

$$T_{VIGA} = T_{rot} = \frac{1}{2} I_v \omega_v^2 = \frac{1}{2} I_v \dot{q}_2^2 T_{BOLA} \quad (24)$$

$$T_{BOLA} = T_{rot} + T_{tras} = \frac{1}{2} m_b v_b^2 + \frac{1}{2} I_b \omega_b^2 \quad (25)$$

Se ilustra a continuación en la figura 38, las variables  $q_1$  y  $q_2$ , en el sistema de forma gráfica para comprender de mejor manera el significado y el alcance de cada elemento, según la trayectoria de la bola sobre las vigas.

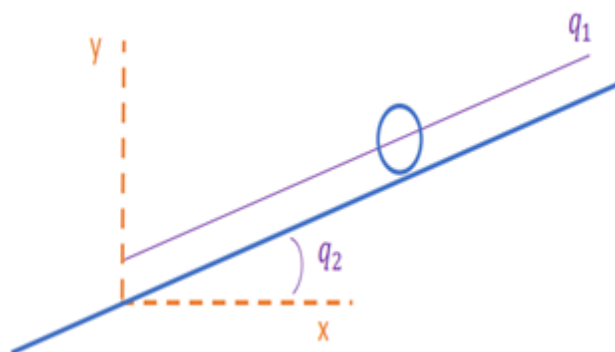


Figura 38. Representación gráfica de la trayectoria de la bola con sus componentes  $q_1$  y  $q_2$  [19].



Como se puede ver en la figura, la bola se mueve a lo largo de los ejes vertical y horizontal. Debido a su propio movimiento y la inclinación de las vigas. Este movimiento se define dentro de un triángulo rectángulo, entonces:

$$v_b^2 = \dot{x}^2 + \dot{y}^2 \quad (26)$$

Donde:

$$x = q_1 \cos q_2 \quad (27)$$

$$\dot{x} = \dot{q}_1 \cos q_2 - q_1 \dot{q}_2 \sin q_2 \quad (28)$$

$$y = q_1 \sin q_2 \quad (29)$$

$$\dot{y} = \dot{q}_1 \sin q_2 + q_1 \dot{q}_2 \cos q_2 \quad (30)$$

Reemplazando:

$$v_b^2 = \dot{q}_1^2 \cos^2 q_2 + q_1^2 \dot{q}_2^2 \sin^2 q_2 - 2\dot{q}_1 q_1 \dot{q}_2 \sin q_2 \cos q_2 + \dot{q}_1^2 \sin^2 q_2 + q_1^2 \dot{q}_2^2 \cos^2 q_2 + 2\dot{q}_1 q_1 \dot{q}_2 \sin q_2 \cos q_2 \quad (31)$$

$$v_b^2 = \dot{q}_1^2 + q_1^2 \dot{q}_2^2$$

La velocidad angular de la bola se calcula a partir de la condición de rodadura, sin desplazarse, dando como resultado:

$$w_b = \frac{\dot{q}_1}{R_b} \quad (32)$$

Reemplazando estas expresiones de velocidad en la ecuación de energía cinética del sistema resulta:

$$T = \frac{1}{2} I_v \dot{q}_2^2 + \frac{1}{2} m_b (\dot{q}_1^2 + q_1^2 \dot{q}_2^2) + \frac{1}{2} I_b \left( \frac{\dot{q}_1}{R_b} \right)^2 \quad (33)$$

Continuando con el procedimiento, se calcula la energía potencial de la viga, tomando la posición horizontal como punto de referencia.

$$V = V_{VIGA} + V_{BOLA} \quad (34)$$

$$V_{VIGA} = -g m_v r_v \cos q_2 \quad (35)$$

$$V_{BOLA} = -g m_b q_1 \sin q_2 \quad (36)$$

Donde,  $r_v$  representa la distancia entre el centro de masa la viga y el eje de coordenadas, la  $g$  es la aceleración de la gravedad, considerada negativa  $\left(-9,8 \frac{m}{s^2}\right)$  y  $r$  es la distancia de la línea que va desde el centro de la bola hasta el eje del motor, este se simplificará y se considerará muy pequeño en comparación con otras medidas, por lo tanto, la energía potencial de la viga puede despreciarse.

$$V = -g m_b q_1 \sin q_2 \quad (37)$$

Finalmente, para la aplicación de la función lagrangiana, también es necesario calcular la fuerza de disipación de Rayleigh, que se considera cero en este caso porque no contiene fuerza disipativa.

Como hay dos coordenadas generalizadas, resultan dos ecuaciones que controlan el comportamiento del sistema, en la cual la fuerza generalizada al sistema de la posición de la bola es cero, y la fuerza generalizada al sistema de la viga es el torque del motor,  $\tau$ .

$$\left( m_b + \frac{I_b}{R_b^2} \right) \ddot{q}_1 + g m_b \sin q_2 - m_b q_1 \dot{q}_2^2 = 0 \quad (38)$$

$$(I_v + m_b q_1^2) \ddot{q}_2 + 2m_b q_1 \dot{q}_1 \dot{q}_2 - m_b g q_1 \cos q_2 = \tau \quad (39)$$

Por lo tanto, el sistema estará definido dinámicamente, donde se encuentran tres constantes, la masa de la bola ( $m_b$ ) y los momentos de inercia de la bola ( $I_b$ ) y la viga ( $I_v$ ). Sin embargo, la segunda ecuación se ha ignorado porque no se controlará el lazo de control para ángulo de la viga sino para el ángulo del servo, por lo tanto no influye en el modelo de función de transferencia [19].

V-D1. *Modelo de la función de transferencia* : Para la obtención del modelo de la función de transferencia, se empieza por la ecuación 38 resultante de la posición de la bola , luego se aplicará la transformada de Laplace. A continuación, en la Figura 39, se representa el modelo de la planta y todas las variables reales que afectan al sistema.

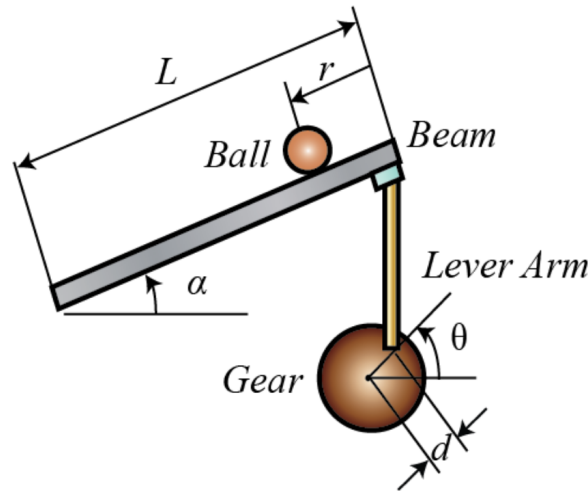


Figura 39. Representación de las constantes y variables que influyen en el sistema para modelar la función de transferencia de la planta [19].

Donde la nomenclatura se muestra en la tabla II: Para obtener la función de transferencia, se empieza de la

Tabla II  
SÍMBOLOS Y MAGNITUDES DE LAS CONSTANTES Y VARIABLES QUE INFLUYEN EN EL SISTEMA

Símbolo	Descripción
m	Masa de bola
R	Radio de bola
d	Desplazamiento de la biela (offset)
g	Gravedad $(9,8 \frac{m}{s^2})$
L	Longitud de la viga
$\alpha$	Ángulo de la viga con respecto a la horizontal
J	Momento de inercia de la bola
r	coordenada de la posición de la bola
$\theta$	Ángulo del servomotor

siguiente ecuación, donde las coordenadas generalizadas han sido reemplazadas por su verdadero valor.

$$\left( \frac{I_b}{R_b^2} + m_b \right) \ddot{r} + gm_b \sin \alpha - m_b r \dot{\alpha}^2 = 0 \quad (40)$$

En la cual r es la posición de la bola medida desde el final de la viga y  $\alpha$  es el ángulo giro como se muestra en

Tabla III  
RELACIÓN DE LAS COORDENADAS GENERALIZADAS A SU VALOR DENTRO DEL SISTEMA

Coordenada	Representación
$q_1$	L-r
$q_2$	$\alpha$

la Figura 39. Para hallar la función de transferencia usando ecuaciones diferenciales, esta debe ser de tipo lineal,

dato que en este caso no lo es, debe ser linealizada a un punto de equilibrio, en cuyo caso solo es posible cuando  $\alpha = 0$ , que es la posición de la viga cuando se encuentra totalmente horizontal. Debe recordar que al linealizar una ecuación no lineal, el modelo obtenido solo es adecuado para el valor cerca del punto de equilibrio. La ecuación linealizada toma la siguiente forma, asumiendo que cuando  $\alpha \approx 0$ , el  $\sin \alpha$  la cual se puede aproximar por el ángulo  $\alpha$  ( $\sin \alpha \approx \alpha$ ).

$$\left( \frac{I_b}{R_b^2} + m_b \right) \ddot{r} = -gm_b \alpha \quad (41)$$

Se debe relacionar el ángulo de la viga con el ángulo del servomotor, la cual se puede aproximar a una ecuación lineal mediante la siguiente igualdad.

$$\alpha = \frac{d}{L} \sin(\theta + \theta_0) \quad (42)$$

Procedemos a la linealización de la ecuación 42:

$$\sin(\theta + \theta_0) = \sin \theta \cos \theta_0 + \cos \theta \sin \theta_0 \quad (43)$$

$$\theta \approx 0 \quad (44)$$

$$\sin(\theta + \theta_0) \approx \theta \cos \theta_0 + \sin \theta_0 \quad (45)$$

Reemplazando las ecuaciones 38 y 45 en la ecuación 42:

$$\left( \frac{I_b}{R_b^2} \right) \ddot{r} = -gm_b \frac{d}{L} \theta \cos \theta_0 \quad (46)$$

Como la ecuación se linealizó usando la transformada de Laplace para pasar de las ecuaciones diferenciales a un espacio polinomial definido por "s", donde los conceptos de la función de transferencia existe.

$$\left( \frac{I_b}{R_b^2} \right) R(s)s^2 = -gm_b \cos \theta_0 \frac{d}{L} \theta(s) \quad (47)$$

Ya definida la función de transferencia como el coeficiente de salidas  $R(s)$  y la entrada del sistema  $\theta(s)$ , la cual viene expresada por:

$$G(s) = \frac{R(s)}{\theta(s)} = -\frac{gm_b d \cos \theta_0}{L \left( \frac{I_b}{R_b^2} + m_b \right)} \frac{1}{s^2} \left[ \frac{m}{rad} \right] \quad (48)$$

Falta considerar la dinámica no lineal del voltaje aplicado al servomotor y el ángulo que gira su eje  $\theta$ , esta relación no se calculó debido a la falta de herramientas para realizar la medición de los parámetros de este sistema. Se puede ver que la función de transferencia hay en el origen dos polos iguales, por lo que ya se puede esperar que su comportamiento sea inestable en el lazo abierto, es decir, el lazo de control debe estar cerrado para lograr estabilizar el sistema. Para implementar el modelo en MATLAB, se necesita parámetros físicos reales de sistema para estudiar la plantas, dichos parámetros se muestra en la tabla IV:

Tabla IV  
PARÁMETROS FÍSICOS Y VALORES REALES DE LA PLANTA

Parámetro	Magnitud
m	0.003 kg
R	0.02 m
d	0.056 m
$\theta_0$	$\frac{\pi}{6}$ rad
g	$-9,8 \frac{m}{s^2}$
L	0.43 m
J	$\frac{2}{5}m_b R_b^2 = 4,8 * 10^{-7}$

Se importa a MATLAB estos valores en el espacio de trabajo con el siguiente comando:

```
8 s=tf('s');
9 G=-(m*g*d*cos(theta_0))/(L*((J/R^2)+m)*s^2);
```

Lo cual genera la función de transferencia como se describe a continuación.

$$G(s) = \frac{0,001426}{0,001806s^2} = \frac{0,7895}{s^2} \quad (49)$$

Una vez que se tiene la función de transferencia, podemos estudiarla y analizar los comportamiento de una entrada y diseñar un controlador para estabilizar el sistema, lo cual veremos más adelante.

*V-D2. Modelo matemático en espacios de estados:* Para la obtención del modelo representado en espacio de estados, se empieza por la ecuación 34 de las dos ecuaciones lagrangianas obtenidas en la sección de modelado general. Según la teoría explicada anteriormente es necesario elegir las variables de estado, por lo tanto, la posición de la bola ( $x$ ) y su velocidad ( $\dot{x}$ ) serán las variables del sistema, dado que se conoce su información en el estado inicial, aparte es importante establecer la variable de entrada, la cual será el ángulo de giro del motor, dado que es la variable que provoca cambios en el sistema y se conoce su valor en cualquier instante de tiempo. Además, se toma para este caso el sistema linealizado, debido a que es una representación habitual en el modelo en espacio de estado.

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (50)$$

$$y(t) = Cx(t) + Du(t) \quad (51)$$

Se deberá identificar términos de la ecuación diferencial linealizada:

$$\left(\frac{I_b}{R_b^2}\right) \ddot{r} = -gm_b \cos \theta_0 \frac{d}{L} \theta \quad (52)$$

Para finalmente obtener el modelo en variable de estado del sistema de la planta, donde la matriz se elige como [10] y la matriz D es nula:

$$\begin{bmatrix} \dot{r} \\ \ddot{r} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ \dot{r} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{-gm_b d \cos \theta_0}{L \left(\frac{I_b}{R_b^2} + m_b\right)} \end{bmatrix} \theta \quad (53)$$

### V-E. Control teórico del sistema

Como se mencionó anteriormente, se han previsto sistemas de inestabilidad de lazo abierto por la posición de los polos en el plano. De todos modos, se comprobará al colocar el modelo de función de transferencia en la entrada y analiza sus respuestas.

Recordamos que la función de transferencia resultante es la siguiente:

$$G(s) = \frac{0,7895}{s^2} \quad (54)$$

Una forma de analizar la estabilidad de un modelo es ingresar una entrada tipo salto y mirar la salida para ver si alcanza un valor estable en estado estacionario, es decir, que cuando la entrada esté acotada la salida también estará acotada.

Para analizar lo mencionado anterior se ingresa el comando:

```
>> step(G(s))
```

Se muestra la gráfica continuación en la figura 40.

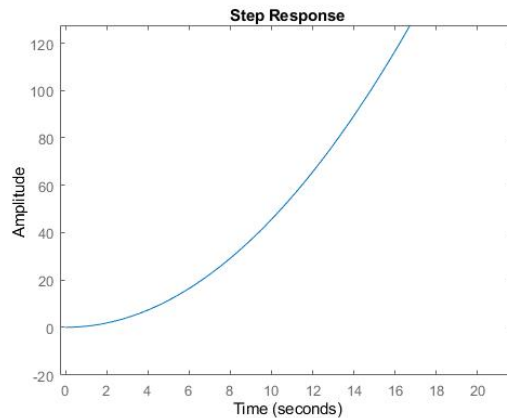


Figura 40. Respuesta transiente del sistema en lazo abierto a una entrada escalón.

Se puede ver que la salida oscila cuando  $t$  tiende al infinito, lo que prueba que el sistema es inestable en lazo cerrado. Esta respuesta es predicha por la observación de la ubicación de los polos en el plano, si se considera que el sistema físico es de esta forma, la entrada es la tensión del motor en el cual se introduce un valor fijo, por ende la viga se inclinará hacia un lado y la esfera rodará sin control hacia un extremo y es imposible llegar a un punto de equilibrio. Por lo tanto, como se mencionó anteriormente, es necesario agregar un controlador que estabilice el sistema.

En las próximas secciones, se desarrollará 2 métodos para controlar este sistema con ayuda del software de MATLAB, estos métodos son: PID mediante la herramienta PID Tuner y por realimentación de estados por asignación de polos.

### V-F. Diseño de controladores para el modelo matemático

V-F1. *Sintonización de controlador PID para el modelo matemático:* Para el controlador PID se obtuvo los parámetros mediante la herramienta PID Tuner de MATLAB, sintonizando el controlador con la forma ideal mostrada en la ecuación 55.

$$P \left( 1 + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}} \right) \quad (55)$$

A continuación, en la figura 41, se representa en Simulink el sistema general para ver la estructura que tiene el sistema de lazo cerrado con un controlador PID.

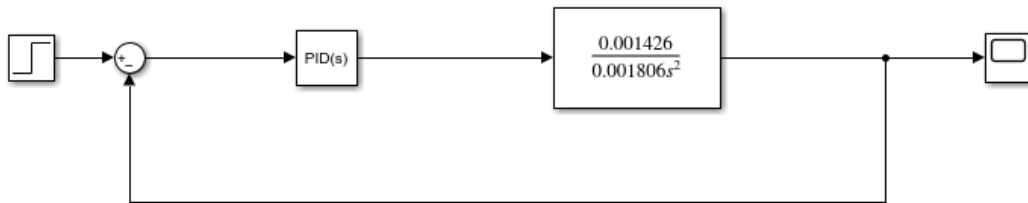


Figura 41. Sistema realimentado con un controlador de tipo PID.

Siguiendo los mismos criterios de diseño, un sobrepico menor al 10 % y un tiempo de estabilización menor a 4 segundos se obtuvieron los parámetros K, I, D y N (filtro de la componente derivativa) mostrados en la figura 42.

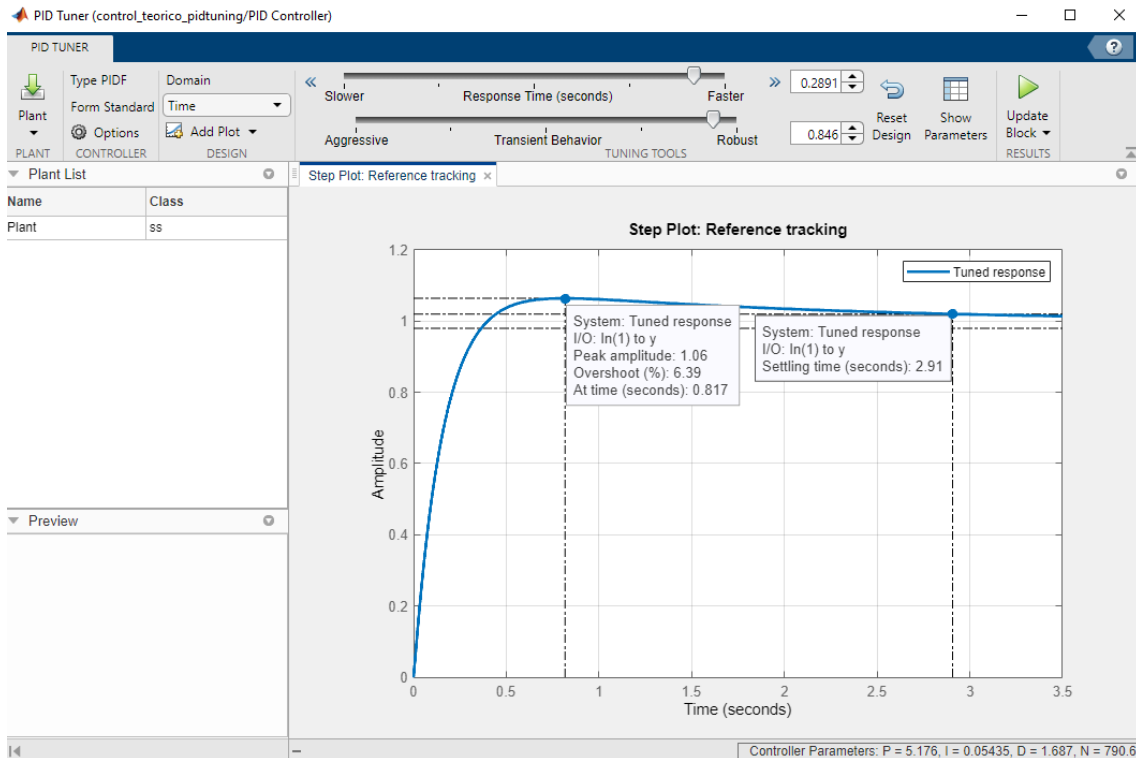


Figura 42. Parámetros obtenidos del PID Tuner y respuesta del sistema bajo los criterios de diseño

V-F2. *Controlador por realimentación de estados por asignación de polos:* Lo que debe hacer es calcular el valor de K, que será una matriz de 1x2 que satisfaga la especificación dada sobre la respuesta del sistema de retroalimentación. Para hacer esto, debe calcular la ubicación del poste deseado. Con esto, obtienes el valor de K. En este caso, tome un sobrepico máximo del 10 % y un tiempo de establecimiento de régimen permanente de 4 segundos. Para ello se utiliza el software MATLAB para ejecutar los siguientes aspectos importantes de la programación:

1. Primero se genera el sistema representado en espacio de estados con el comando “ss”.

```

%% Parámetros
m = 0.003; %masa de la bola (kg)
R = 0.02; %radio de la bola (m)
g = -9.8; %aceleracion gravitacional - eje vertical positivo
L = 0.43; %Longitud de la viga
d = 0.056; %desplazamiento del brazo (offset)
theta_0 = pi/6; %Angulo inicial para viga horizontal
J = 4.8e-7; %momento de inercia de la bola [ Jesfera=(2/5)
s=tf('s');
%% Representacion en espacio de estados
% Variables estado: posición de la bola(x), velocidad (x-dot)
%de entrada angulo de giro del motor (theta)
A=[0 1; 0 0];
B=[0; -(m*g*d*cos(theta_0))/(L*((J/R^2)+m))];
C=[1 0];
D=[0];
ballandbeamOL=ss(A,B,C,D)

```

Figura 43. Creación del modelo en lazo abierto en espacio de estados de la planta

Al ejecutar esta sección de código se obtiene el sistema visto en el “Command Window”:

```

ballandbeamOL =

    A =

        x1    x2
    x1     0     1
    x2     0     0

    B =

        u1
    x1         0
    x2    0.7895

    C =

        x1    x2
    y1     1     0

    D =

        u1
    y1     0

Continuous-time state-space model.

```

Figura 44. Modelo de espacio de estado en lazo abierto en MATLAB

- Después, se comprueba si el sistema es controlable obteniendo la matriz de controlabilidad con el comando “**ctrb**” y verificar si cumple con la ecuación 16.

```

%% Comprobacion de controlabilidad
Co = ctrb(A,B)
[m, n] = size(Co);
if m==n
    rank(Co) %si Co es cuadrada
    if rank(Co)==m %si el rango es igual al num de var
        disp('El sistema es controlable');
        es_controlable=true;
    else
        disp('El sistema no es controlable');
        es_controlable=false;
    end
end

```

Figura 45. Comprobación de controlabilidad en MATLAB

La ejecución resulta en la conclusión de que el sistema es controlable.

```

Co =

         0    0.7895
    0.7895         0

ans =

     2

El sistema es controlable

```

Figura 46. Matriz de controlabilidad en MATLAB

3. A continuación se obtiene la matriz de ganancias K mediante el comando “**place**” con el polinomio característico en la ecuación 21 que, en este sistema, es un polinomio de orden 2; y la ecuación deseada bajo los parámetros  $\zeta$  y  $\omega_n$  del diseño.



```

%% Diseño de controlador regulador por reubicación de
%zeta y Wn para SP y Tss objetivos
SP=10; %sobrenivel porcentual menor a 10%
Tss=4; %Tiempo de estabilización 4s
zeta=-log((SP-6)/100)/sqrt(pi^2+(log((SP-6)/100)^2));
zeta=round(zeta,1);
Wn=4/(zeta*(Tss-2));
Wn=round(Wn,1);

%% Ecuación característica del Sistema
E_Ca=poly(eig(ballandbeamOL));

%% Ecuación deseada
Ps= [1 2*zeta*Wn Wn^2];
Ed=roots(Ps);

%% Ganancia de Realimentación de Estados
K = place(A,B,Ed)

```

Figura 47. Obtención de matriz de ganancias K

Se obtiene así la matriz de ganancias en el “Command Window”:

```

K =

    10.6524    5.1425

```

Figura 48. Matriz de ganancias K en MATLAB

- Finalmente, se construye el sistema en lazo cerrado y se añade una ganancia  $k_0$  como precompensador estático para minimizar el error de estado estacionario a la salida del proceso.

```

% Sistema en lazo cerrado
Alc=A-B*K;
ballandbeamCL=ss(Alc,B,C,D)
% Pre-compensador estático
Gdc =dcgain(ballandbeamCL);
k0=1/Gdc

```

Figura 49. Obtención de sistema en lazo cerrado e implementación de una ganancia  $k_0$

Se obtiene lo siguiente:

```

ballandbeamCL =

A =
      x1      x2
x1      0      1
x2 -8.41 -4.06

B =
      u1
x1      0
x2 0.7895

C =
      x1 x2
y1      1  0

D =
      u1
y1      0

Continuous-time state-space model.

k0 =

10.6524

```

Figura 50. Controlador realimentado en espacio de estados con precompensador estático  $k_0$

A diferencia del PID, el controlador de reubicación de polos multiplica las variables de estado por un vector de ganancias, cuyos resultado será restado de la señal de referencia escalada por un factor  $k_0$ .

#### V-G. Comunicación entre Arduino y LabVIEW

Para realizar la conexión entre Arduino y LabVIEW es necesario previamente tener instalada la librería de LINX/LabVIEW MakerHub, una vez cumplido este requisito se describe los pasos a seguir:

1. Conectar la placa mediante cable USB a la PC y abrir el VI a compilar.
2. En el menú superior del programa dar clic en Tools > MakerHub > LINX > Linx Firmware Wizard.
3. Aparecerá una nueva ventana donde se procede a escoger el Arduino UNO así como se observa en la figura 51 y luego dar clic en “Next”.

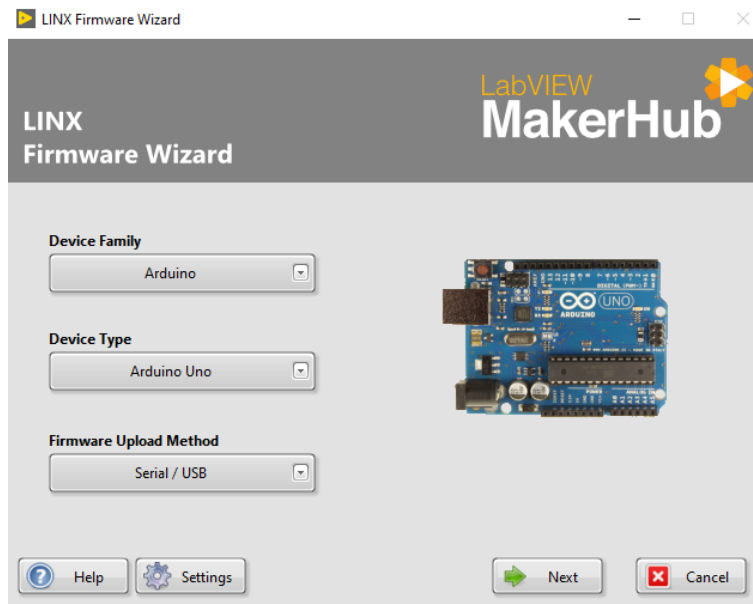


Figura 51. Elección de placa a comunicar mediante LINX Firmware Wizard.

4. En el siguiente menú desplegable escoger el puerto serial de la placa, en este caso particular el puerto serial es COM4 como se observa en la figura 52 y luego dar clic en “Next”.

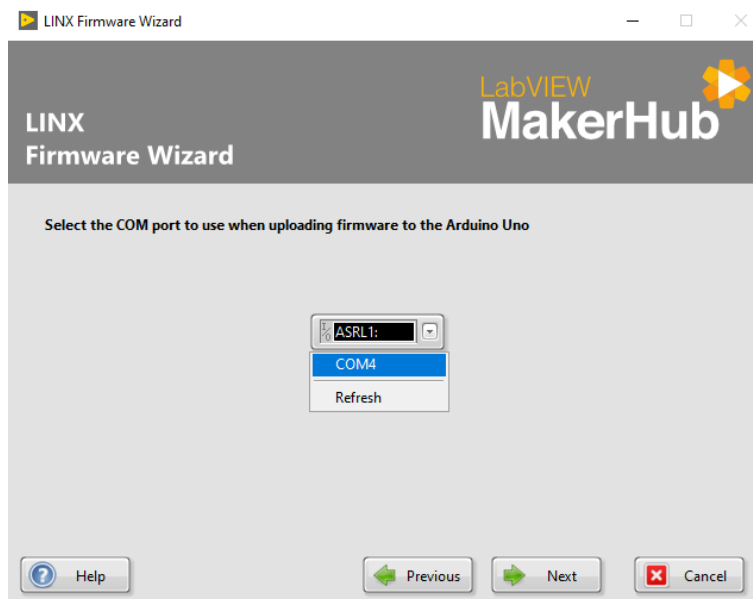


Figura 52. Elección de puerto serial de comunicación entre Arduino y LabVIEW.

5. Verificar que debajo de Firmware Version y Upload Type estén seleccionadas las opciones LINX - Serial/USB y Pre-Built Firmware, respectivamente; luego dar clic en Next y en la siguiente pantalla esperar que complete la carga.
6. Si la carga fue exitosa aparecerá el mensaje que se observa en la figura 53, sino es así se sugiere desconectar la tarjeta, cerrar LabVIEW y posteriormente repetir nuevamente los pasos del 1 al 5; dar clic en “Finish”.

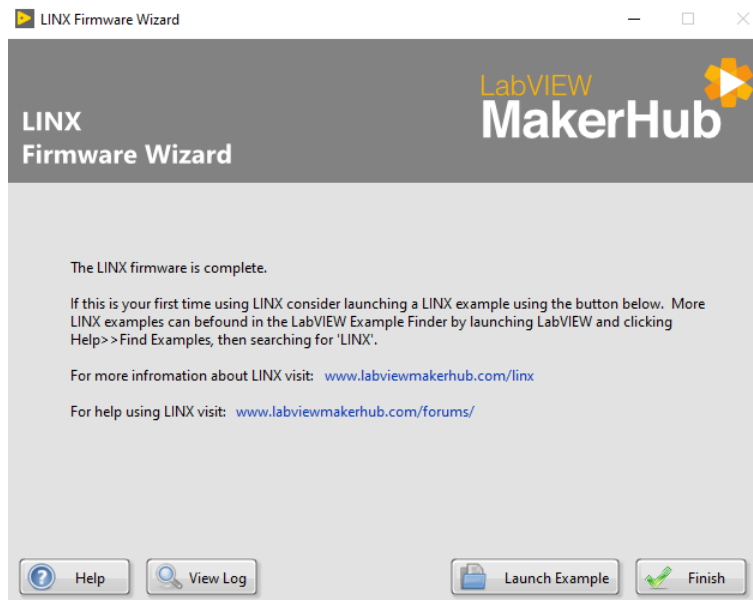


Figura 53. Mensaje de comunicación serial exitosa entre Arduino y LabVIEW.

#### *V-H. Conexión de cámara USB de smartphone con LabVIEW*

Debido al requerimiento del módulo de un rendimiento alto para la clasificación de color se optó por el uso de la cámara de un smartphone para un rápido y preciso procesamiento de la imagen. Para esto se debe instalar la aplicación DroidCam de la tienda de aplicaciones de android Google Play Store así como su equivalente de Windows, DroidCamApp [22].

Para dispositivos iphone la alternativa es la aplicación EpocCam que se encuentra en la App Store donde es necesario instalar los drivers dependiendo el sistema operativo macOS y Windows [6].

#### *V-I. Diseño e implementación de los controladores en LabVIEW*

Para la implementación didáctica en el módulo se realizó la siguiente interfaz de usuario en el panel frontal de LabVIEW:

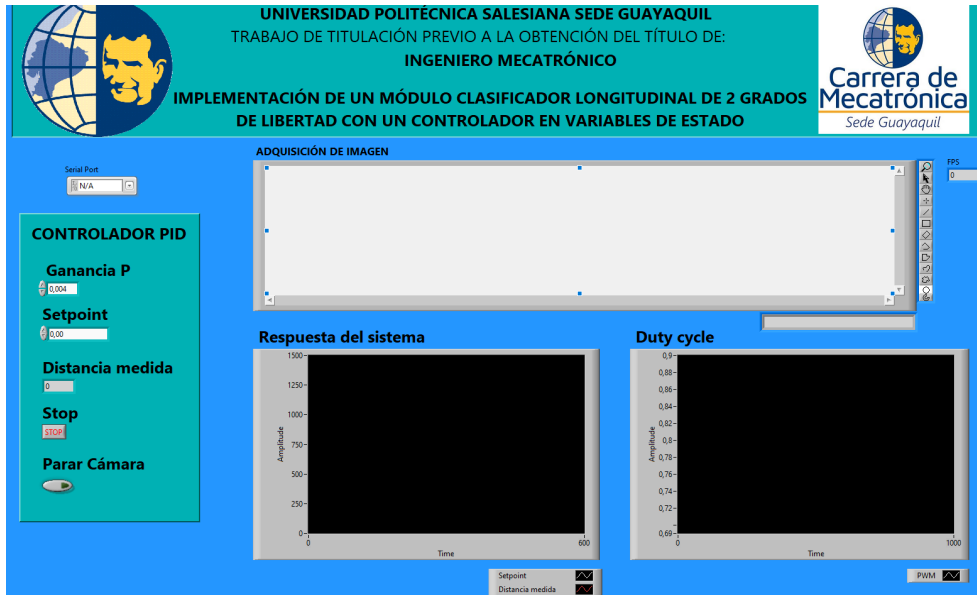


Figura 54. Interfaz de usuario de los controladores didáctico en LabVIEW.

Como se muestra en la figura 54 la interfaz cuenta con el título y los autores del proyecto; el puerto serial para la conexión con la placa Arduino, la ganancia P, ganancias K1 y K2, el setpoint, la distancia medida por el sensor, el botón de paro de la cámara, el botón Stop, la imagen obtenida de la cámara y las gráficas de la respuesta del sistema y duty cycle del servomotor de inclinación. Del diagrama de bloques completo de la figura 90 se detallan las siguientes partes:

#### V-J. Clasificación por color mediante NI Vision Express

En la figura 55 se muestra el esquema de conexiones de los bloques correspondientes a la sección IV-E destinados a la adquisición y procesamiento de imagen; y la determinación de la posición de la bola dependiendo su color.

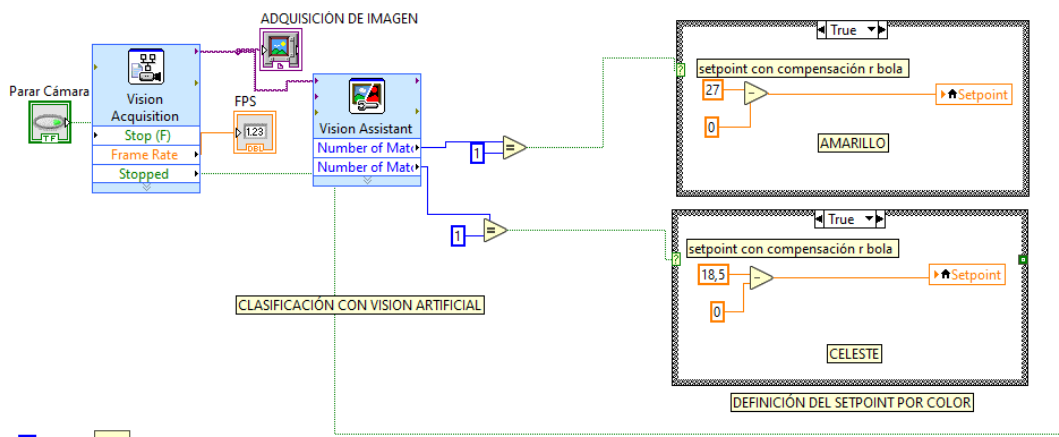


Figura 55. Esquema de bloques utilizados para la clasificación por color y determinación de setpoint

Después de completar la instalación de la librería Vision and Motion descrita en la sección IV-E y realizar la configuración del hardware de la sección V-H y se procede a la adquisición de la imagen desde el bloque Vision Acquisition Express.

Al analizar la figura 56 se concluye que la imagen adquirida tiene una resolución de 640x480 píxeles de tipo RGB de 32 bits y la tasa de adquisición es de 30 cuadros por segundos o fps por sus siglas en inglés (frames per second).

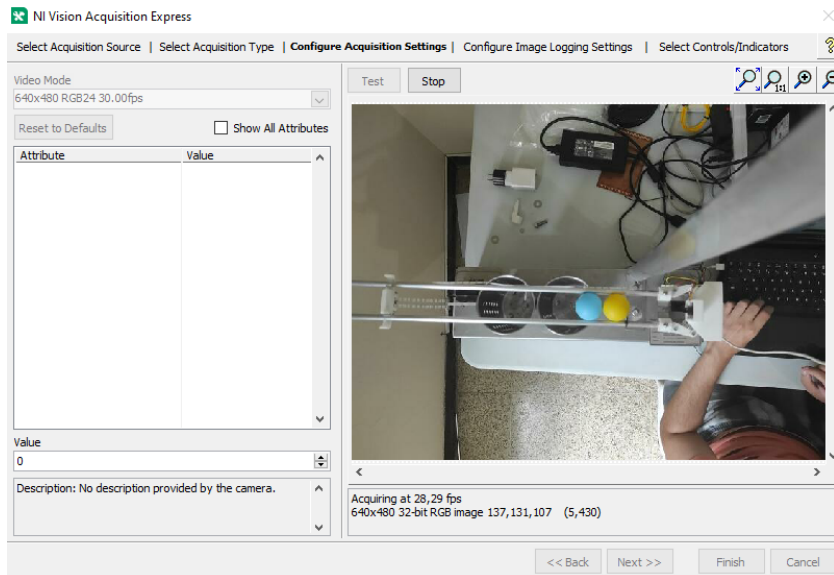


Figura 56. Adquisición de la imagen a procesar mediante la herramienta NI Vision Adquisition Express

Posteriormente se conecta la salida de la imagen adquirida al bloque NI Vision Assistant para su respectivo procesamiento.

El menú de la figura 57 contiene una amplia gama de funciones para delimitar y definir los parámetros a trabajar a partir de un conjunto de imágenes. Entre ellas están opciones de obtención de región de interés, brillo, tamaño, detección de patrones por color, forma, simetría, escala de grises, etc.

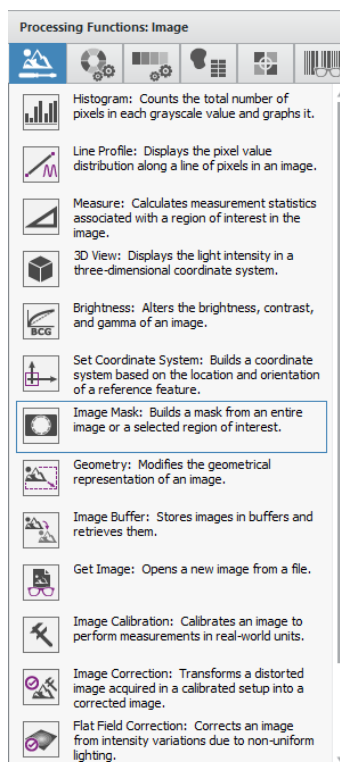


Figura 57. Funciones de procesamiento de imágenes de la herramienta NI Vision Assistant

Se describen las funciones utilizadas y el orden de ejecución a continuación:

1. Se escoge una región de interés con la función Image Mask del panel “Processing Functions: Image”, para la correcta clasificación por color de las pelotas sobre la viga se delimitó la región mostrada en la figura 58.

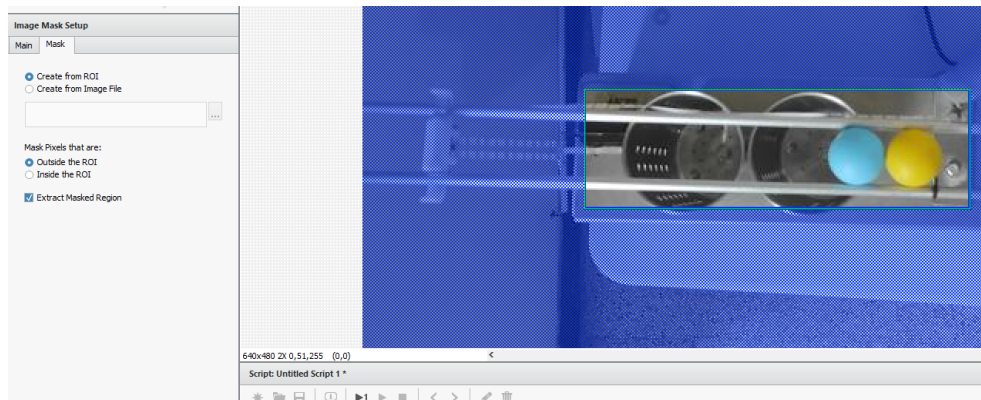


Figura 58. Delimitación de la región de interés mediante la función Image Mask

2. Se obtiene los colores desde la región de interés mediante la función Color Pattern Matching del panel “Processing Functions: Color”; el proceso es mostrado en las figuras 59 y 60 para la pelota de color amarillo y celeste, respectivamente.

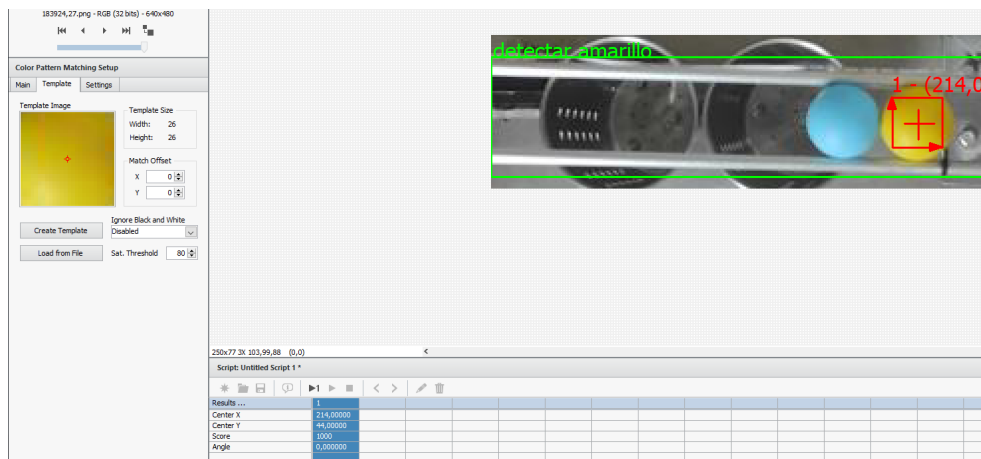


Figura 59. Identificación del color de la pelota amarilla mediante la función Color Pattern Matching

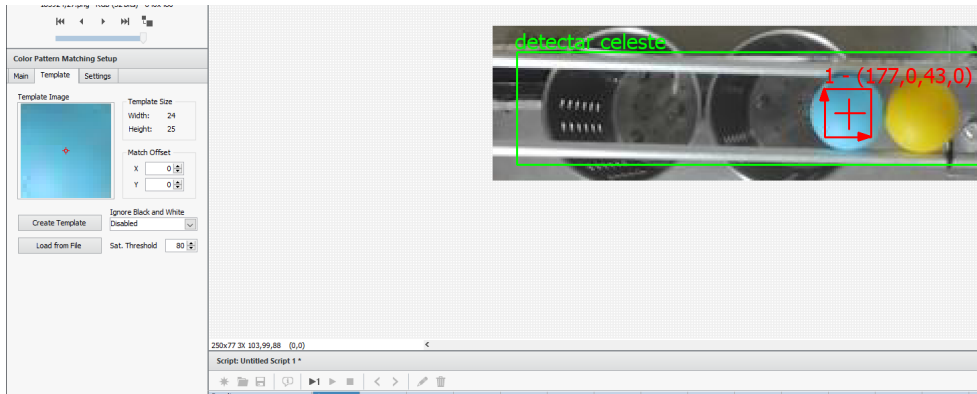


Figura 60. Identificación del color de la pelota celeste mediante la función Color Pattern Matching

3. El algoritmo final proporciona la información necesaria para el proceso de clasificación denominada “Number of Matches”; éste parámetro posteriormente es comparado con un valor de 1 para ambos colores para luego definir el setpoint en 27 cm para la pelota amarilla; y 18.5 cm para la celeste.

## VI. RESULTADOS

### VI-A. Simulación de la respuesta del sistema con el controlador PID

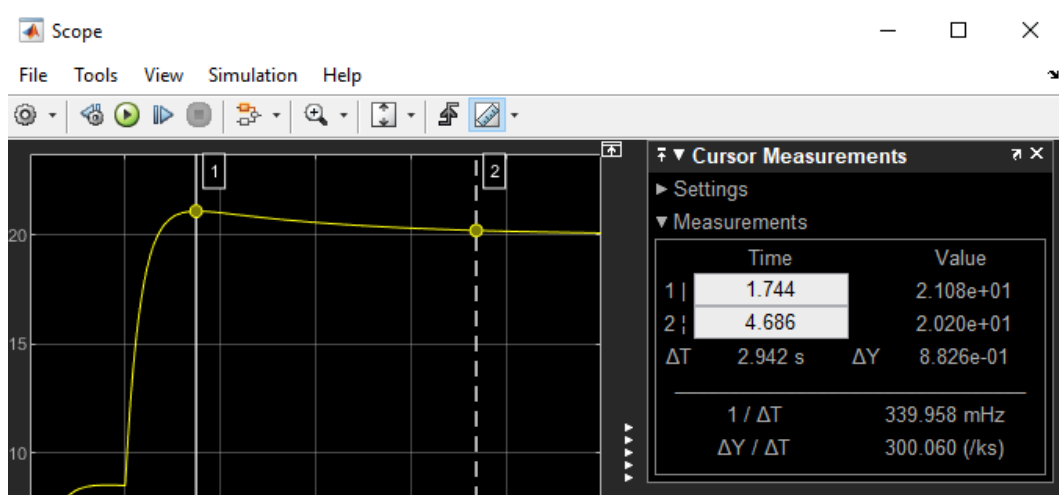


Figura 61. Respuesta del sistema con el controlador PID en Simulink

El controlador PID de la figura 61 está simulado con una entrada generada por un bloque Step configurado a  $u(0) = 8cm$  y a  $u(t) = 20cm$  como ejemplo del funcionamiento del prototipo implementado, así la respuesta del sistema tiene un sobrenivel de aproximadamente 10.7% y un tiempo de establecimiento de 3.686 segundos.

### VI-B. Simulación de la respuesta del sistema con el controlador por realimentación de estados



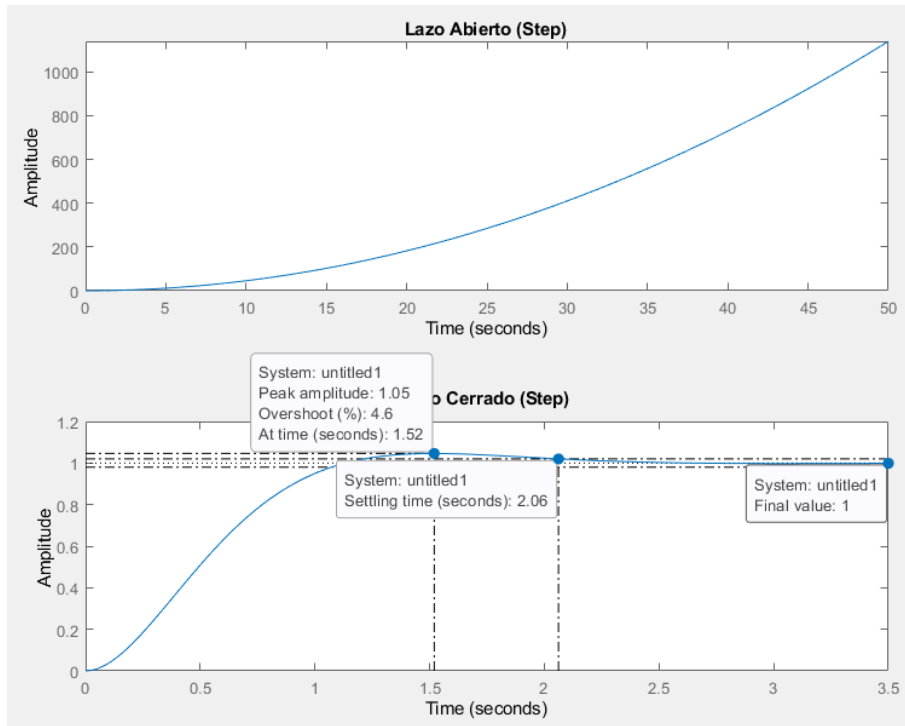


Figura 62. Respuesta del sistema con el controlador por realimentación de estados con precompensador  $k_0$

Como se puede observar en la figura 62 obtenida del comando “step”, las características del sistema son: sobrenivel de aproximadamente 4.6% y un tiempo de establecimiento de 2.06 segundos.

#### VI-C. Comparación de técnicas de control utilizadas

Para la comparación se realizó un diagrama de bloques mostrado en la figura 63 en el add-on Simulink de MATLAB que consta de individualmente los diagramas de cada técnica estudiada y mediante un bloque multiplexor “Mux” se unen las 2 señales de salida y la señal escalón que son visualizadas en el bloque Scope.

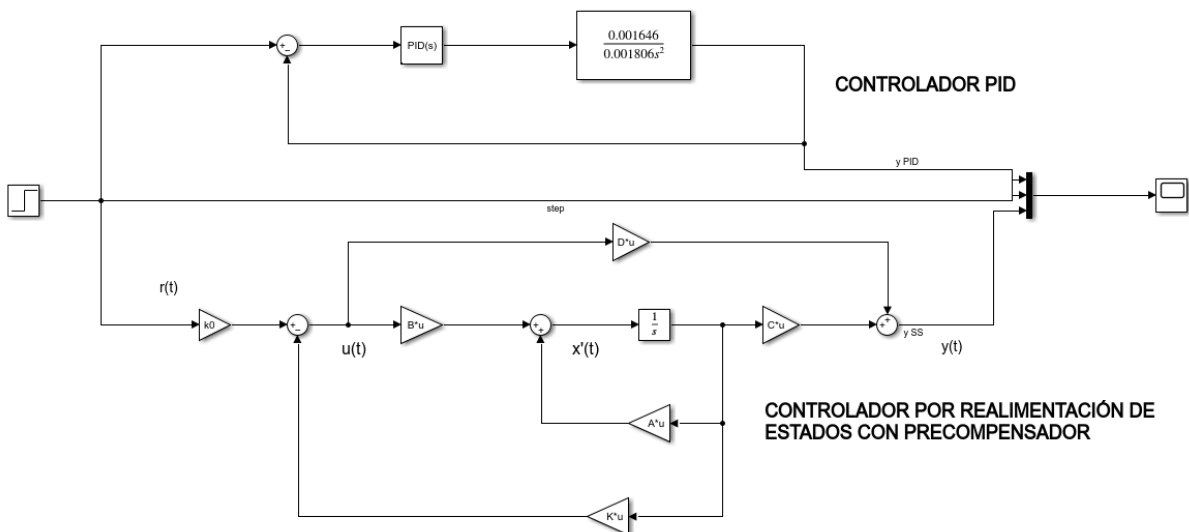


Figura 63. Modelo para comparación de técnicas de control

Las señales vistas desde el Scope para una señal de referencia igual a la utilizada en la figura 61, se ilustran en la figura 64.

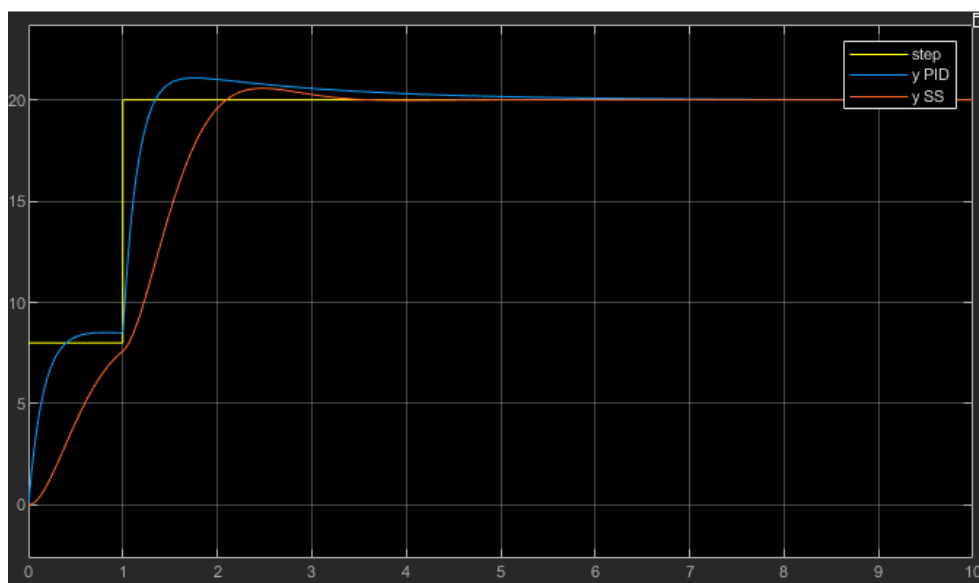


Figura 64. Respuestas multiplexadas de las 2 técnicas de control utilizadas

En la tabla V se comparan las características obtenidas en las simulaciones en el software MatLAB y su addon Simulink.

Tabla V  
CARACTERÍSTICAS DE RESPUESTA DEL SISTEMA CON CADA TÉCNICA DE CONTROL

Técnica	SO(%)	Tss(s)	Ess(%)
PID Tuner	10.7	3.686	0
Realimentación de estados con precompensador	4.6	2.06	0

#### VI-D. Desempeño del sistema implementado

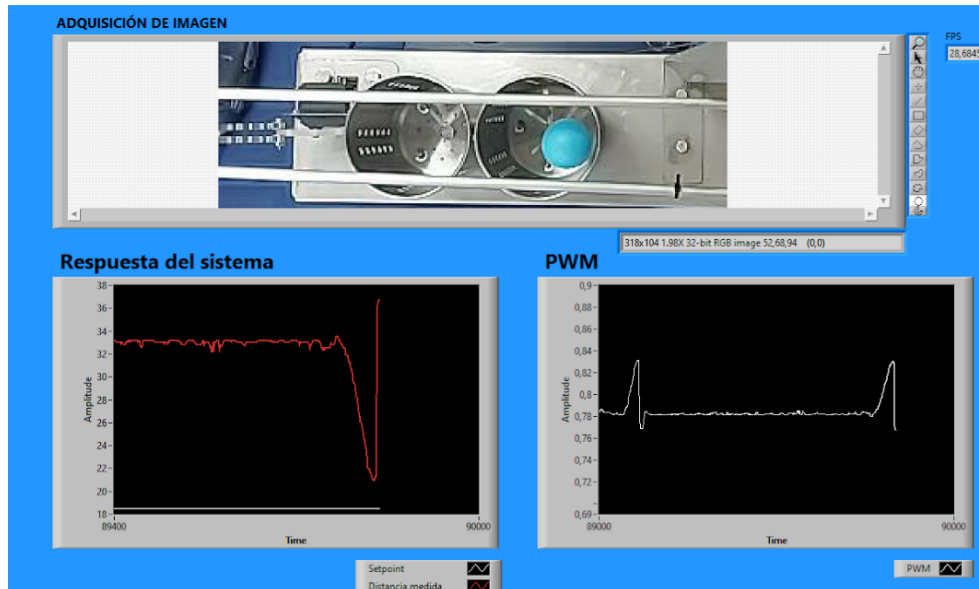


Figura 65. Gráficas de la respuesta del controlador PID implementado en LabVIEW

En la figura 65 se muestra el desempeño del controlador PID donde se detallan las siguientes partes:

- Adquisición de imagen: Pantalla donde se visualiza la imagen en tiempo real del proceso de clasificación de la esfera en el recipiente.
- Respuesta del sistema: Gráfico cm vs t que muestra la posición deseada (línea blanca) y la distancia medida (línea roja).
- PWM: Gráfica del ciclo de trabajo de la señal PWM del servomotor de inclinación con respecto al tiempo.

En el gráfico “Respuesta del sistema” se observa la posición de la esfera hasta que cae de las vigas porque está en el rango seleccionado para el punto de referencia correspondiente para su color. Entonces, el valor mostrado de la posición se incrementa drásticamente hasta aproximadamente 37 cm, distancia máxima que mide el sensor debido a que la esfera no está presente. En el gráfico “PWM” muestra el porcentaje del ciclo de trabajo para acercar la esfera y reducir su velocidad para luego caer en el recipiente este valor permanece constante para dejar la viga en posición horizontal.

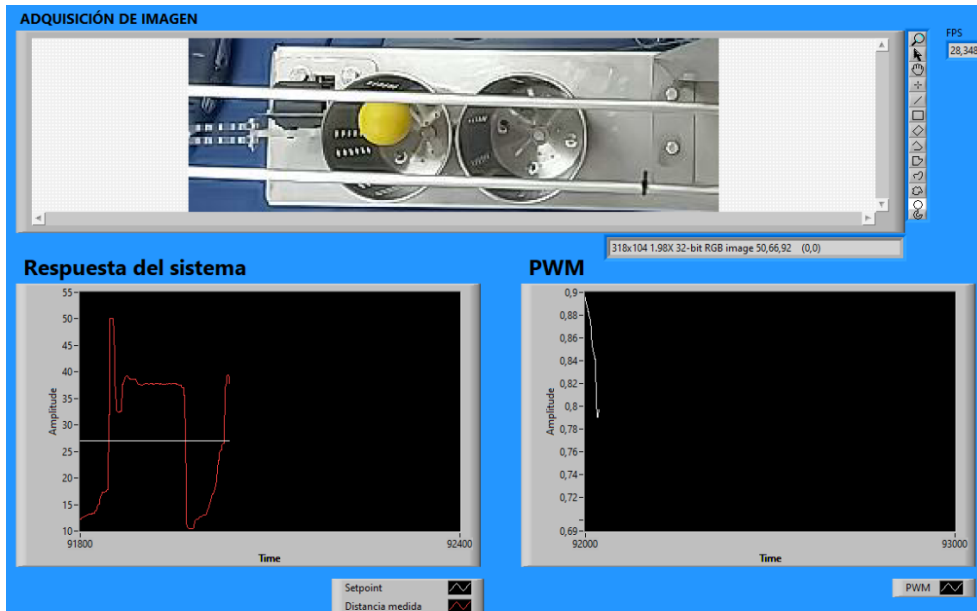


Figura 66. Gráficas de la respuesta del controlador por asignación de polos por realimentación de estados implementado en LabVIEW

En la figura 66 se observa la gráfica “Respuesta del sistema” donde la esfera se acerca para caer cerca del valor de referencia definido por su color; la velocidad es realimentada por la ganancia  $K_2$  del controlador, el servomotor abre las vigas y la distancia aumenta drásticamente, un comportamiento similar al del controlador PID de la figura 65.

## VII. CRONOGRAMA

A continuación se muestra el cronograma de trabajo en la figura 67.

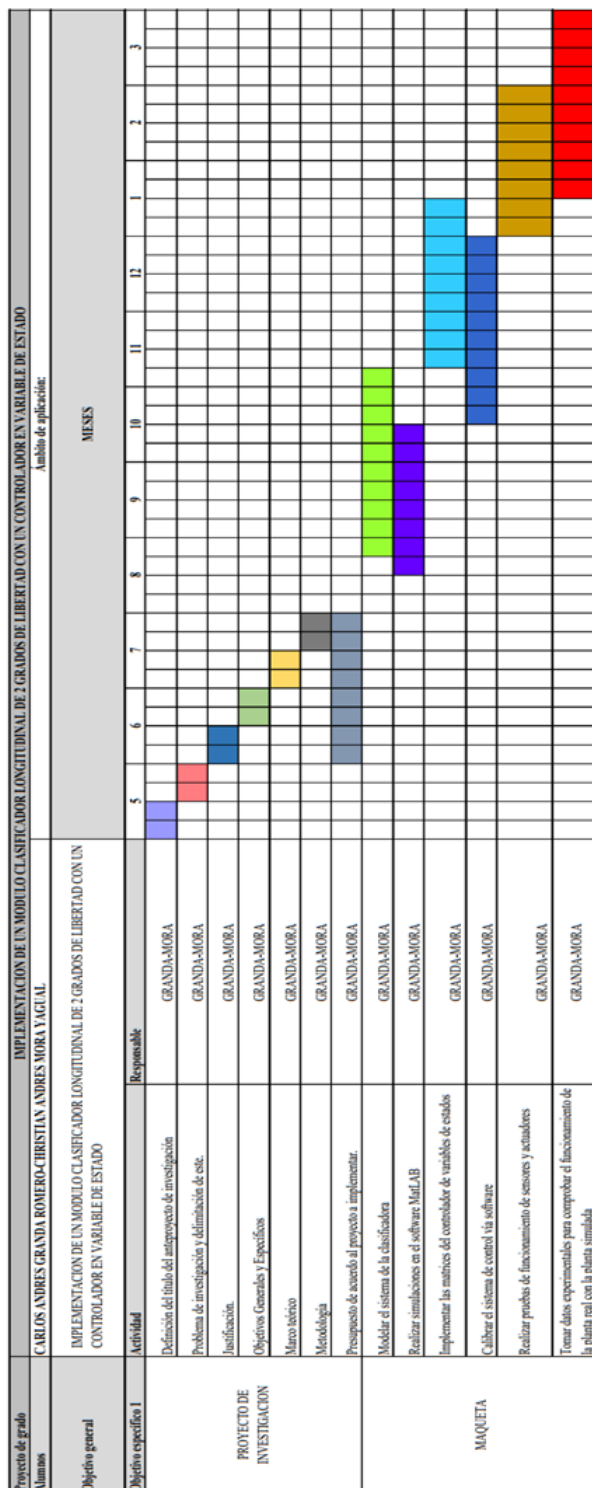


Figura 67. Cronograma

## VIII. PRESUPUESTO

Nombre del elemento	Descripción	Cantidad	Valor total
Varilla	En aluminio de 8mm de diámetro	1	5\$
Sensor Sharp GP2Y0A21	Sensor de distancia	1	20 \$
Servomotor	MG995	3	45 \$
Servomotor	MG90s	3	30 \$
bolas	plásticas	6	4\$
Plancha de acrílico	5mm	1	130 \$
Plancha de acero inoxidable	cuarto de plancha de 1.2mm	1	120 \$
Perno	10mm	45	18 \$
Anillo plano	10mm	45	4 \$
Anillo de presión	10mm	45	4 \$
Tuerca	10mm	45	4 \$
Perno	7mm	10	5 \$
Anillo plano	7mm	10	2 \$
Anillo de presión	7mm	10	2 \$
Tuerca	7mm	10	2 \$
Perno	3mm	15	3 \$
Anillo plano	3mm	15	1.5 \$
Tuerca	3mm	15	2 \$
Fuente de alimentación	5V 2A	1	20\$
Arduino uno	tarjeta electrónica	1	30\$
Redboard Qwiic	tarjeta electrónica	1	30\$
<b>SUBTOTAL MAQUETA</b>			<b>481.5\$</b>
Mano de obra	corte láser de acero inoxidable	1	130\$
Mano de obra	corte láser de acrílico de 5mm	1	25\$
transporte	transporte público y gasolina		500\$
Alimentación			250\$
Horas de trabajo		350 horas	3500\$
<b>SUBTOTAL COSTOS LOGÍSTICOS</b>			<b>4405\$</b>
<b>TOTAL(MAQUETA + COSTOS LOGÍSTICOS)</b>			<b>4886.5\$</b>

## IX. CONCLUSIONES

Se logró estabilizar la bola sobre la viga en los puntos deseados, por medio de la implementación del controlador en espacio de estados, haciendo uso del módulo diseñado, en el cual se implementó la visión artificial para su clasificación longitudinal dependiendo el color.

Se usó la visión artificial para facilitar la determinación de los colores en tiempo real, en la cual se implementó las herramientas de LabVIEW para el procesamiento de imágenes, ya que resulta muy eficaz la adquisición de los colores en las bolas teniendo en cuenta las condiciones de luz de esto dependerá la funcionalidad del sistema.

Además, se realizó el análisis de la planta para obtener el modelo matemático, donde se tomó los datos reales de la estructura, tanto la parte mecánica y los grados de libertad que permiten el desplazamiento y clasificación de la bola, de esta manera se logra linealizar la planta para lograr la estabilidad del sistema.

Se diseñó un controlador, el cual logra estabilidad del sistema, al comprobar que disminuyen los márgenes de errores a perturbaciones, esto se puede verificar mediante el funcionamiento de la planta.

## X. RECOMENDACIONES

Es de mucha importancia calibrar la cámara antes de trabajar con visión artificial para evitar errores en el procesamiento de imagen, dado que la luz varía mucho con respecto al entorno en el cual se esté utilizando, es recomendable para obtener un mejor funcionamiento que el entorno se encuentre con una buena iluminación y para evitar problemas con la detección del color se recomienda pintar las bolas con pintura mate, esto ayudara a mejorar el procesado de la imagen, el tiempo y los resultados dependen también de la complejidad del sistema.

Es importante que la cámara se encuentre fija, dado que si se presentan movimientos provocara que la cámara no procese los datos correctamente y falle el sistema haciendo que el sistema presente inestabilidad.

Antes, de poner en funcionamiento la planta se recomienda primero calibrar tanto el sensor infrarrojo como los servomotores para evitar daños en los componentes de la planta, para el sensor infrarrojo se verificó el voltaje de salida cuando se cambiaba la posición de la bola y convirtiendo el voltaje resultante a centímetros se verificó su funcionalidad esto gracias a LabVIEW y para los servomotores es necesario establecer el correcto ángulo de operación.

Una de las acciones a tomar para mejorar el tiempo de adquisición de la imagen, es utilizar una cámara de mayor prestaciones, la cual ayudara con la respuesta del procesado en LabVIEW.

## REFERENCIAS

- [1] Arduino, *Uno R3 — Arduino Documentation*, Accedido: 28-12-2021, 2019. dirección: <https://docs.arduino.cc/hardware/uno-rev3>.
- [2] Avilés, Domínguez, Muños y Sánchez, *Ingeniería de control moderna*. Sahagún, España: Ingenio Y Conciencia Boletín Científico De La Escuela Superior Ciudad Sahagún, 2018.
- [3] S. Castaño, *Control por Realimentación de Estados*, Accedido: 01-01-2022, 2020. dirección: <https://controlautomaticoec.com/sistemas-dinamicos-lineales/control-por-realimentacion-de-estados/>.
- [4] M. Chong, *Robótica e inteligencia artificial*. Santiago, Chile: El Cid Editor, 2009.
- [5] R. Dorf y R. Bishop, *Sistemas de control moderno*. Madrid: Pearson Educacion S.A., 2005.
- [6] elgato, *EpocCam*, Accedido: 05-03-2022, 2022. dirección: <https://www.elgato.com/es/epoccam>.
- [7] I. Gallego, *Comparativa de diferentes técnicas de control con finalidad didáctica sobre una maqueta bola y viga de bajo coste*. Madrid, España: Escuela Técnica Superior de Ingenieros Industriales, 2016.
- [8] L. Kevicky, R. Bars, J. Hetthéssy y C. Bányász, *Control Engineering - Advanced Textbooks in Control and Signal Processing*. Singapur: Springer, 2019.
- [9] J. R. Lajara y J. P. Sebastián, *LabVIEW Entorno gráfico de programación*. Barcelona, España: Marcombo, 2011.
- [10] F. López, P. Monroy y J. A. D. Rairan, «Control de posición de un sistema bola y viga con actuadores magnéticos,» 2011.
- [11] A. Martínez, M. Gómez, S. Sánchez y J. Martínez, *Robótica e inteligencia artificial*. Santiago, Chile: El Cid Editor, 2009.
- [12] MecatrónicaLATAM, *R: Servomotor*, Mecatrónica LATAM queremos ser tu punto de referencia para el mundo tecnológico, 2021.
- [13] A. Mejía, D. Jabba, C. Caballero y J. Caicedo, «Influencia de la Ingeniería de Software en los Procesos de Automatización Industrial,» 2019.
- [14] National Instruments, *LINX de Diligent/LabVIEW Hobbyist Toolkit*, Accedido: 9-01-2022, 2021. dirección: <https://www.ni.com/en-us/support/downloads/tools-network/download.labview-hobbyist-toolkit.html#376574>.
- [15] National Instruments Support, *Aprenda LabVIEW: Introducción a programación gráfica en NI LabVIEW*, Accedido: 10-10-2021, 2021. dirección: <https://www.ni.com/getting-started/labview-basics/esa/>.
- [16] Naylamp Mechatronics, *Sensor de distancia infrarrojo SHARP GP2Y0A21*, Accedido: 15-02-2022, 2022. dirección: <https://naylampmechatronics.com/sensores-proximidad/203-sensor-de-distancia-infrarrojo-sharp-gp2y0a21.html>.
- [17] K. Ogata, *Ingeniería de control moderna*. Madrid, España: Pearson Educacion S.A., 2010.
- [18] L. R. Pérez, «Diseño e integración de un sistema de realimentación para una maqueta bola y barra e implementación de estrategias de control,» 2018.
- [19] H. M. M. Rodríguez, *Control de un sistema Bola Balancín*. Valladolid, España: Escuela de Ingenierías Industriales, 2019.
- [20] J. Rodríguez y J. García, *MATLAB: guía de aprendizaje*. Córdoba, Argentina: Jorge Sarmiento Editor - Universitas, 2020.
- [21] R. Saltaren, L. Puglisi y J. Sabater, *Robótica aplicada: análisis y diseño de robots paralelos y seriales con MATLAB*. Madrid, España: Dextra Editorial, 2017.
- [22] J. Soto, *Droidcam: Cómo usar la cámara de tu móvil como Webcam del PC*, Accedido: 05-03-2022, 2020. dirección: <https://www.geeknetic.es/Guia/1763/Droidcam-Como-usar-la-camara-de-tu-movil-como-Webcam-del-PC.html>.



APÉNDICE A  
PLANOS

Planos de las piezas acotadas en inventor 2020:

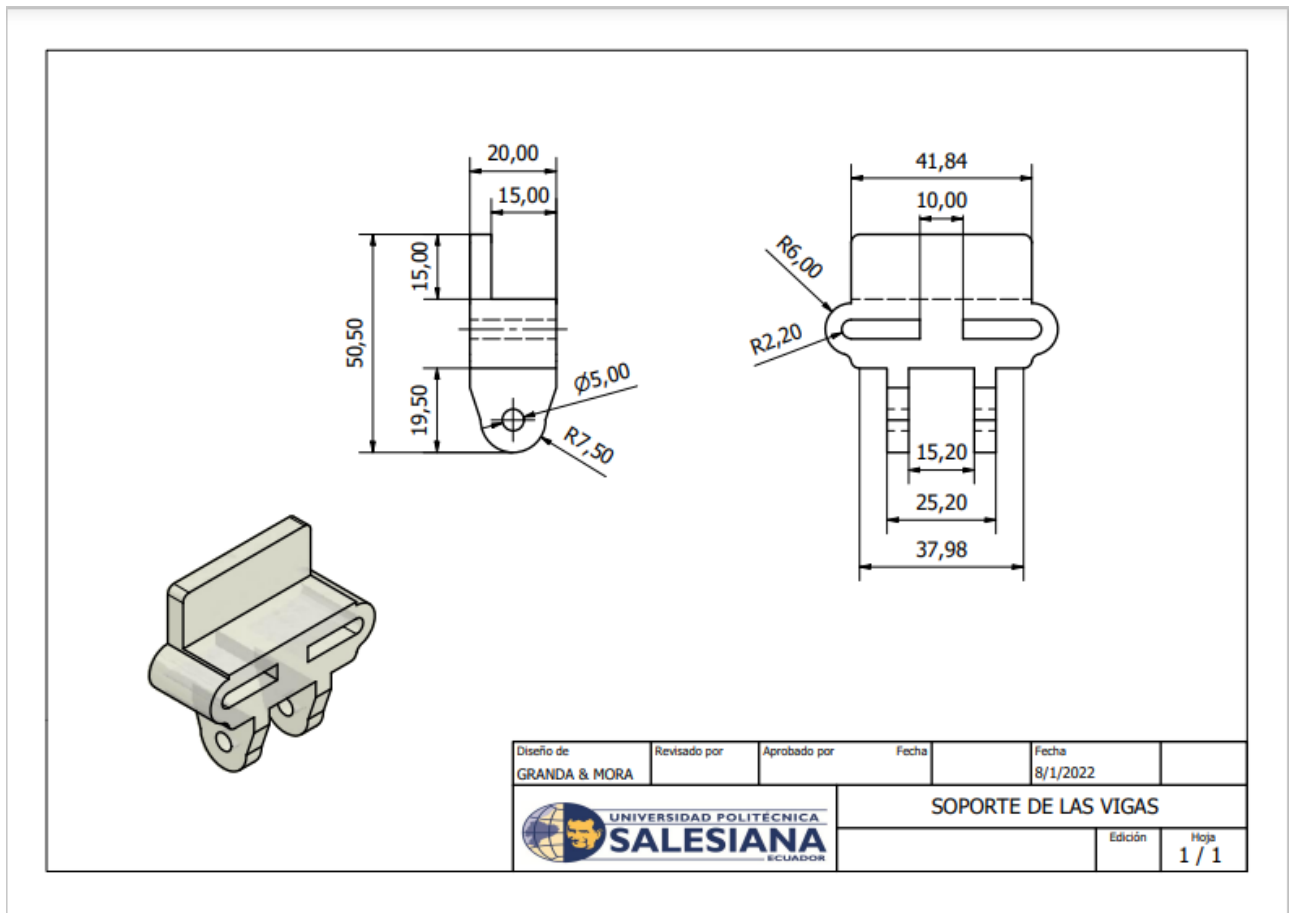


Figura 68. Lámina acotada del soporte de viga, por C.Granda y C.Mora, Inventor Autodesk

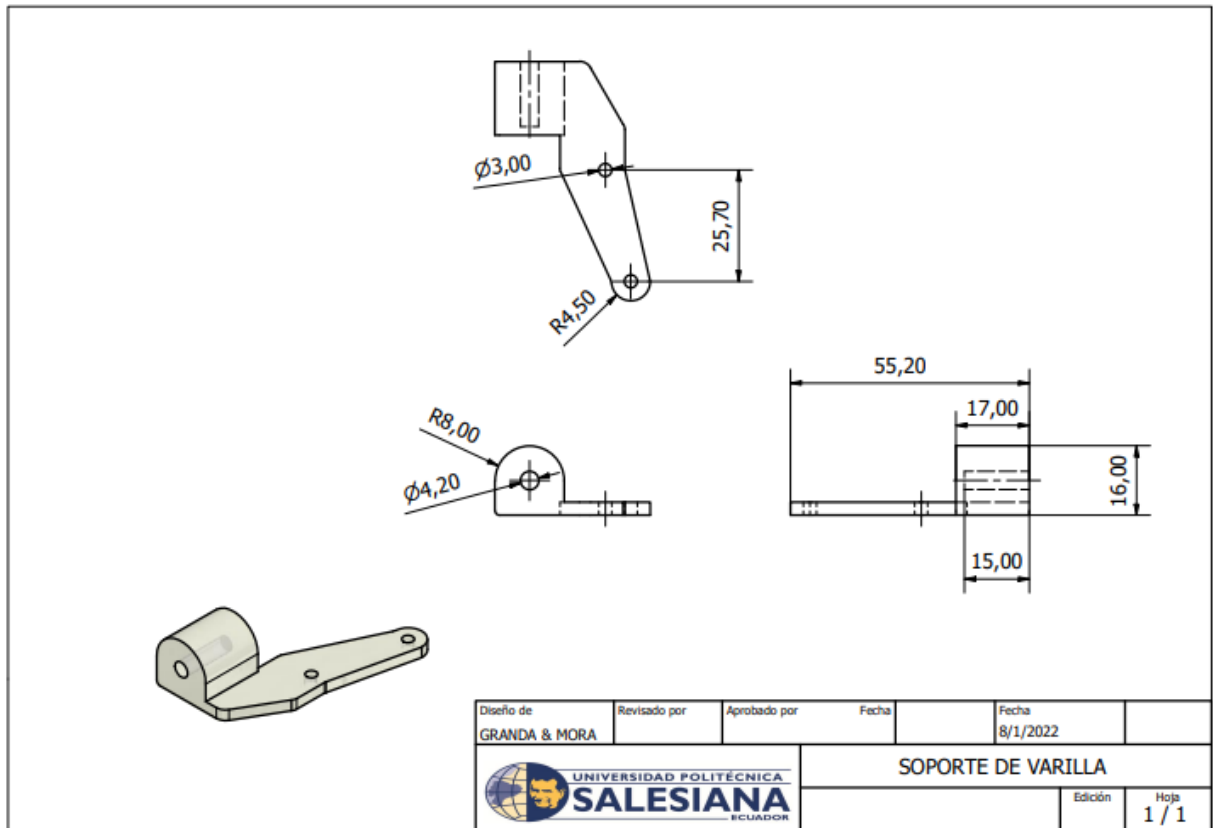


Figura 69. Lámina acotada del brazo porta viga, por C.Granda y C.Mora, Inventor Autodesk

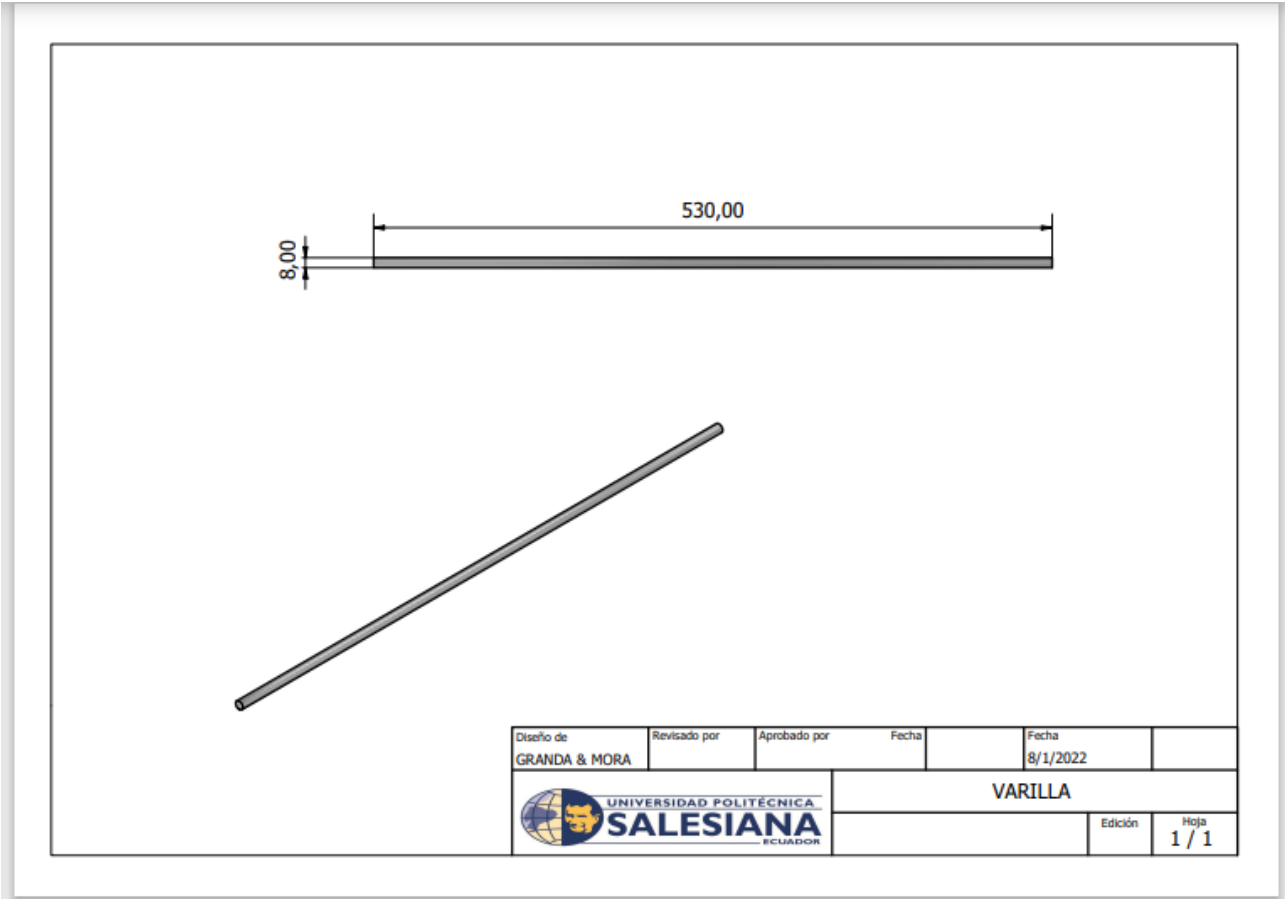


Figura 70. Lámina acotada de la viga, por C.Granda y C.Mora, Inventor Autodesk

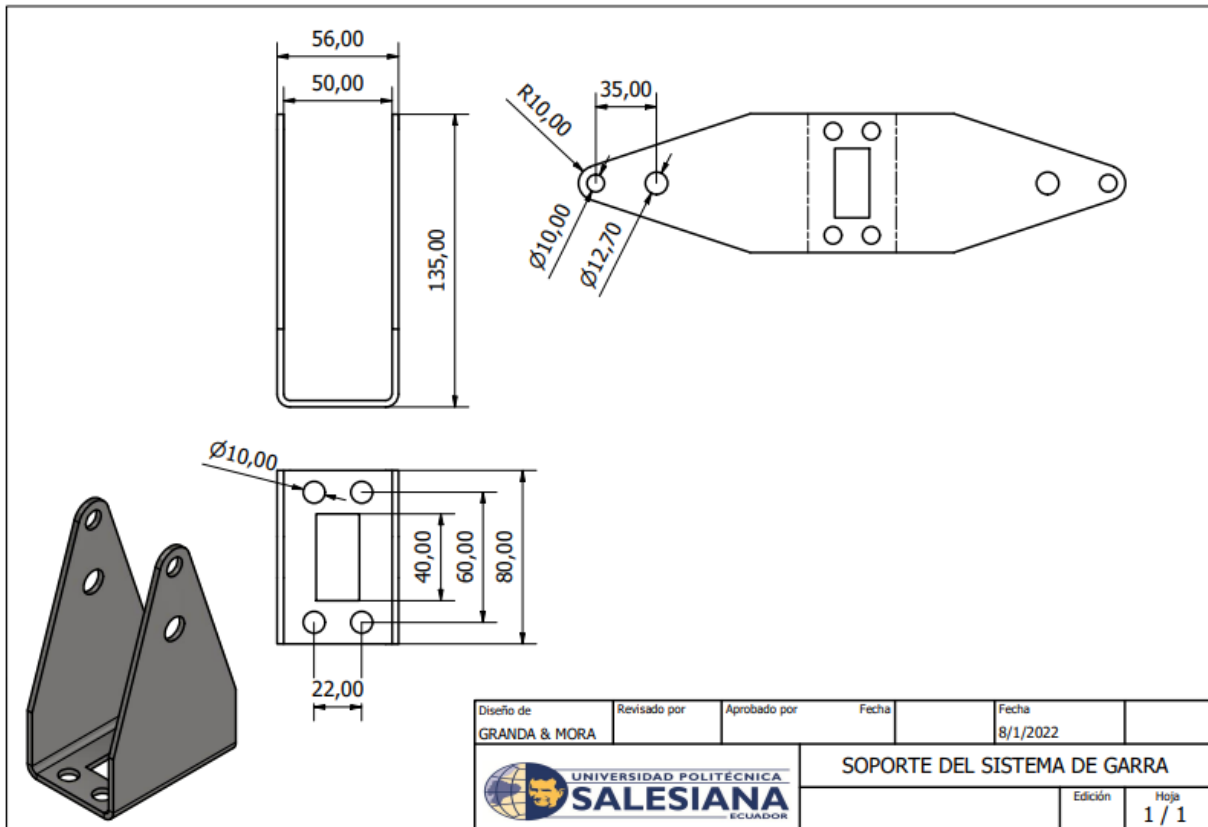


Figura 71. Lámina acotada del soporte del sistema de abertura, por C.Granda y C.Mora, Inventor Autodesk

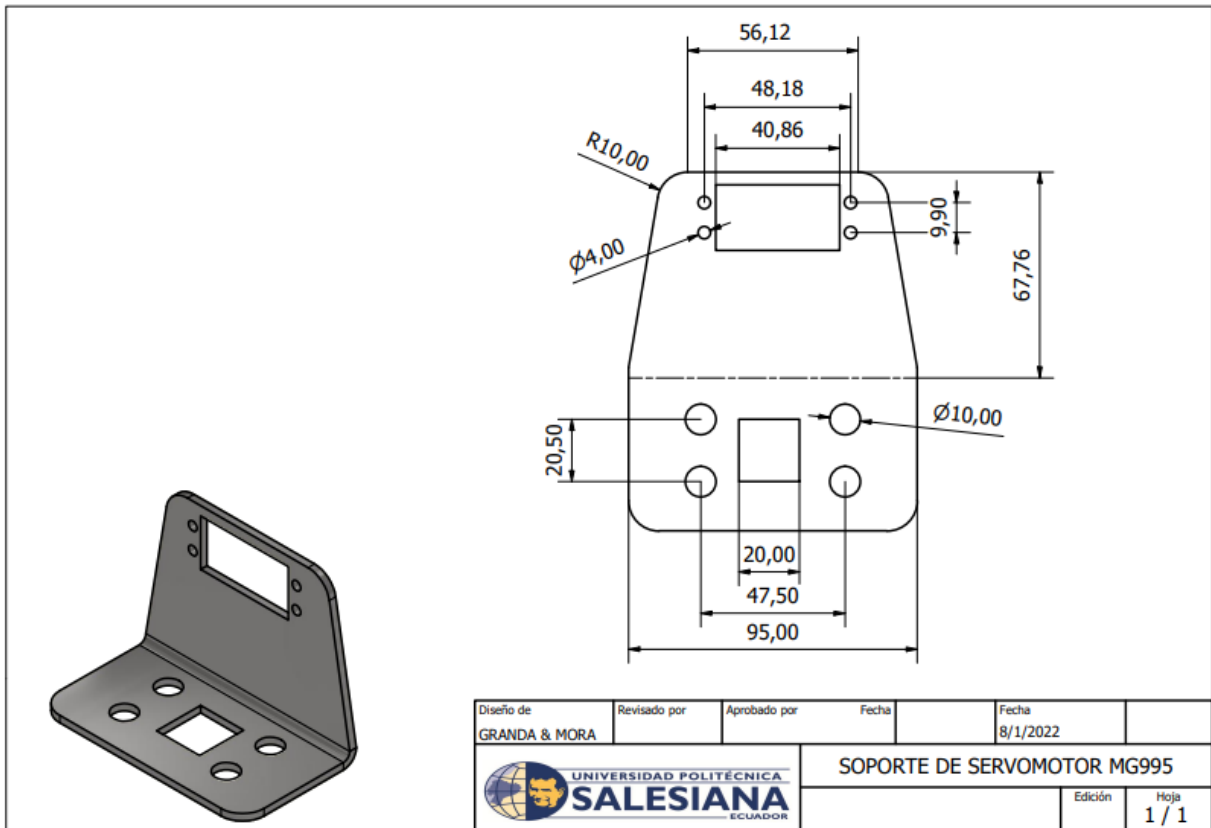


Figura 72. Lámina acotada del soporte del servomotor, por C. Granda y C. Mora, Inventor Autodesk

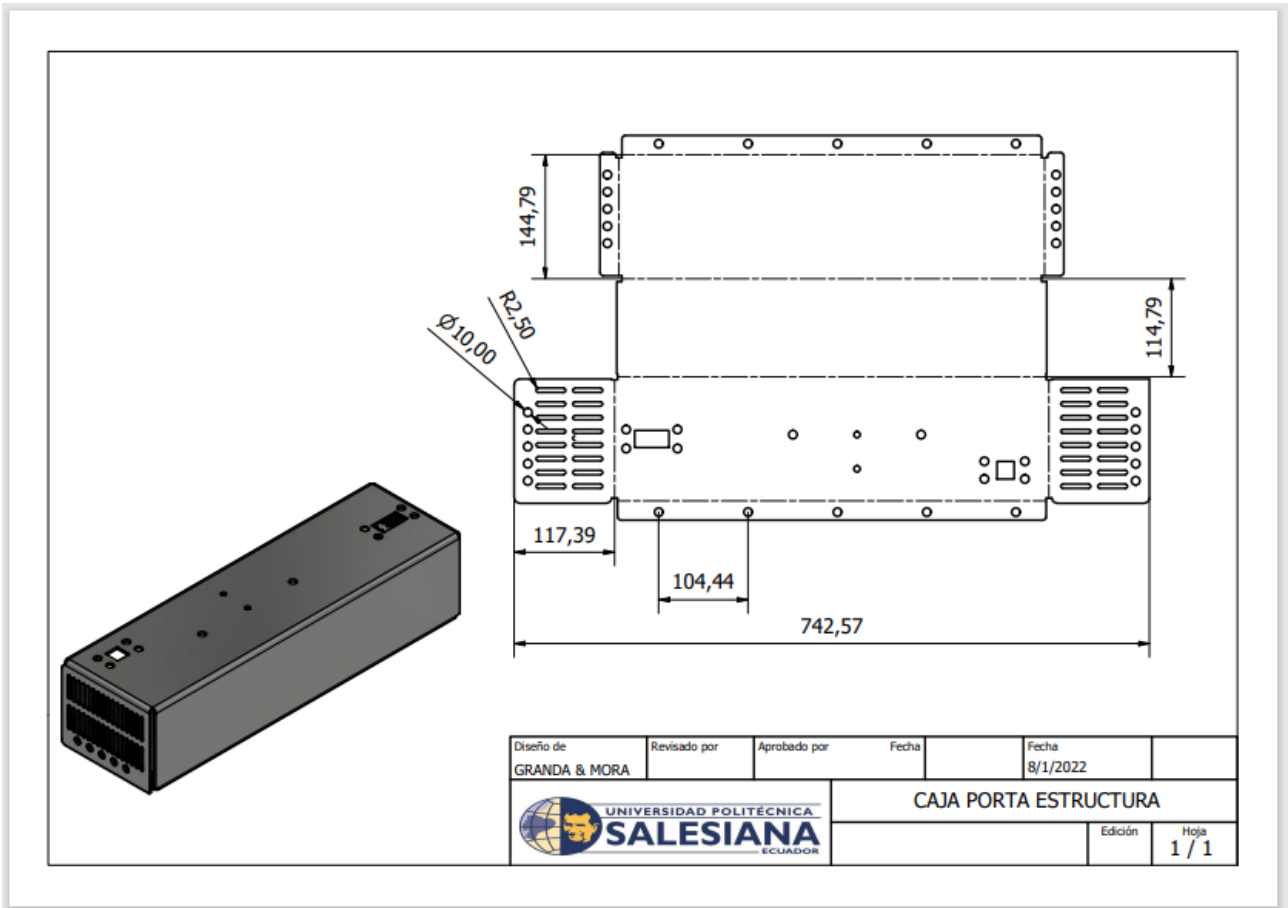


Figura 73. Lámina acotada de la caja porta estructura, por C.Granda y C.Mora, Inventor Autodesk

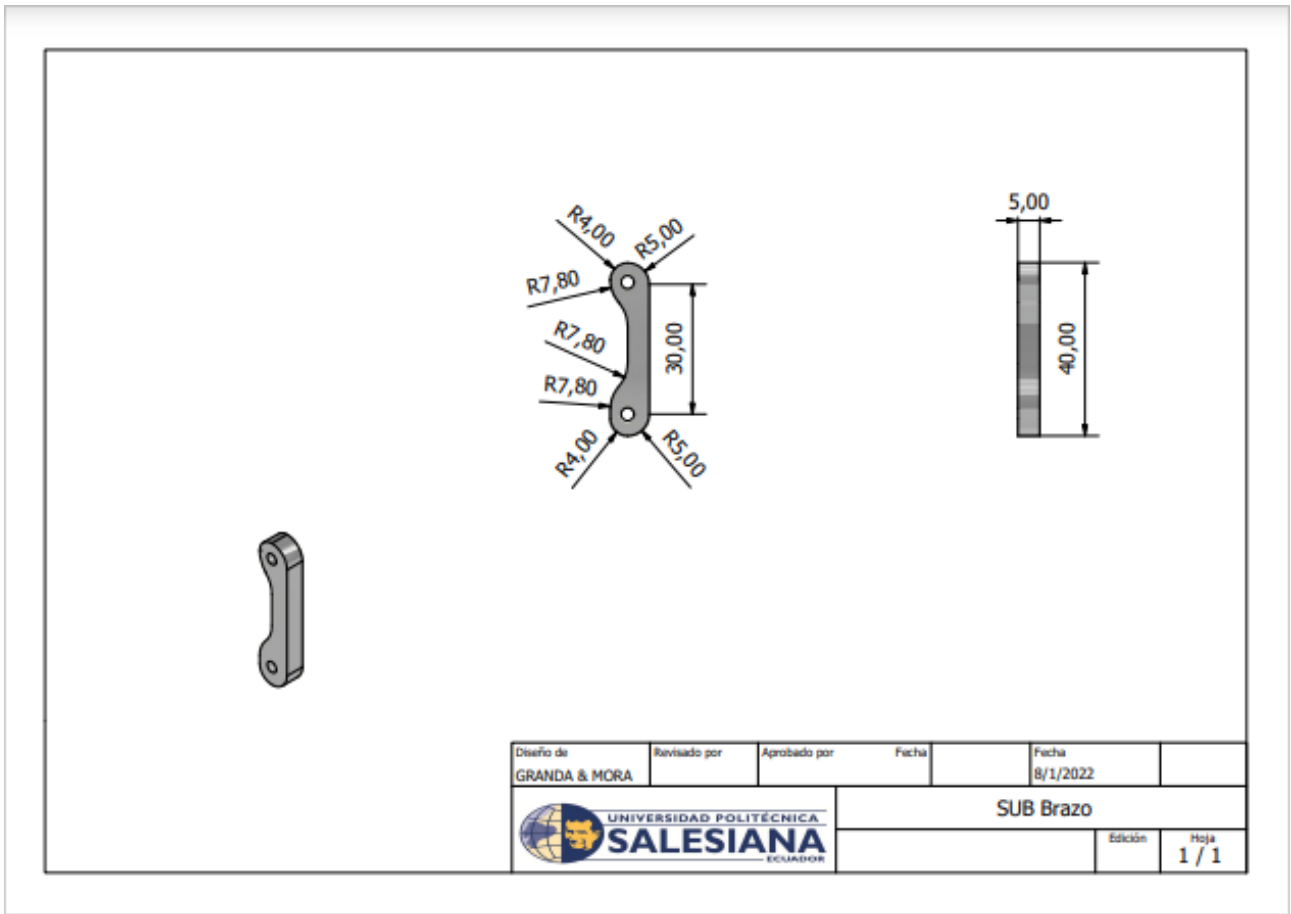


Figura 74. Lámina acotada de los sub brazos, por C.Granda y C.Mora, Inventor Autodesk

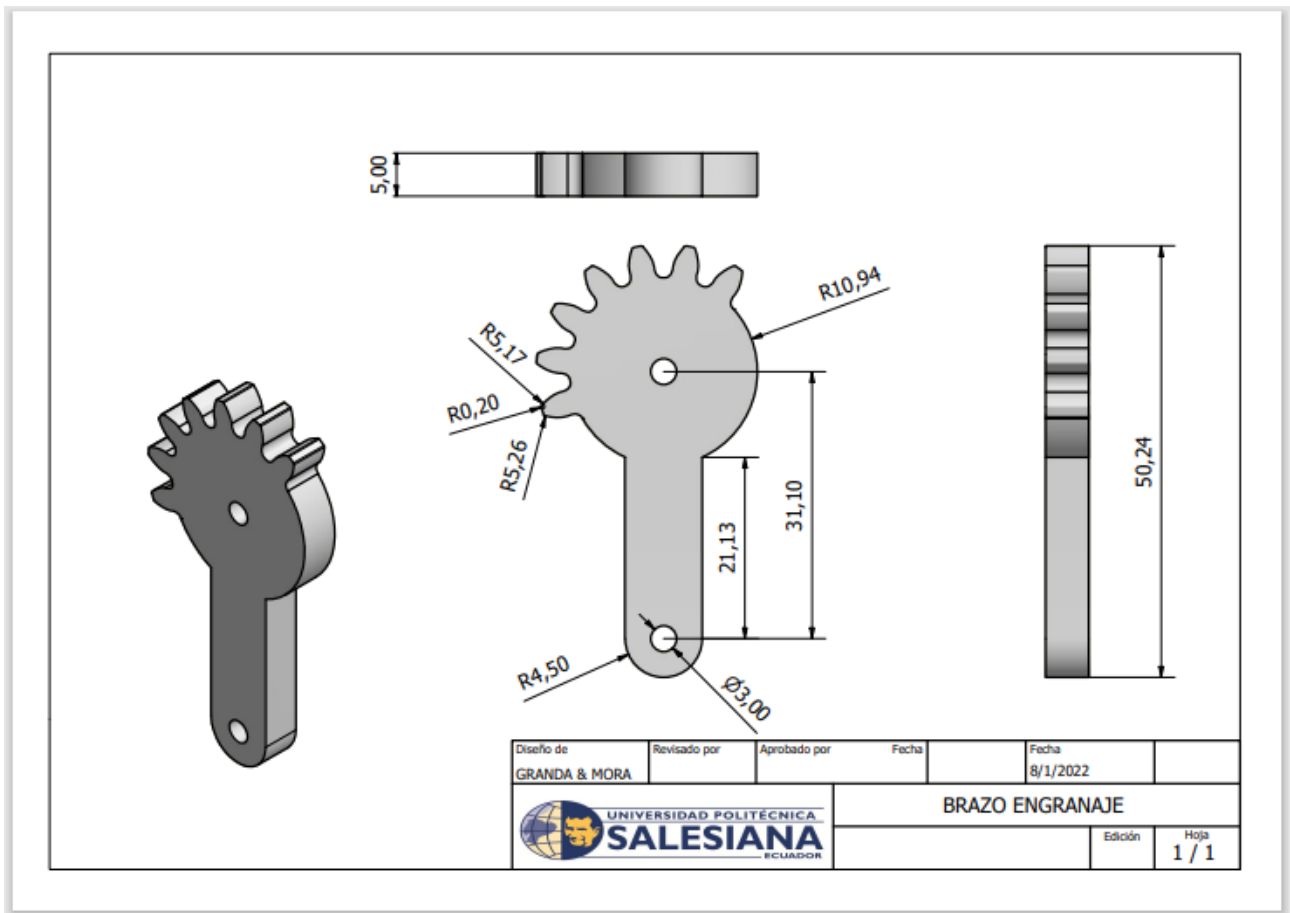


Figura 75. Lámina acotada del brazo, por C.Granda y C.Mora, Inventor Autodesk



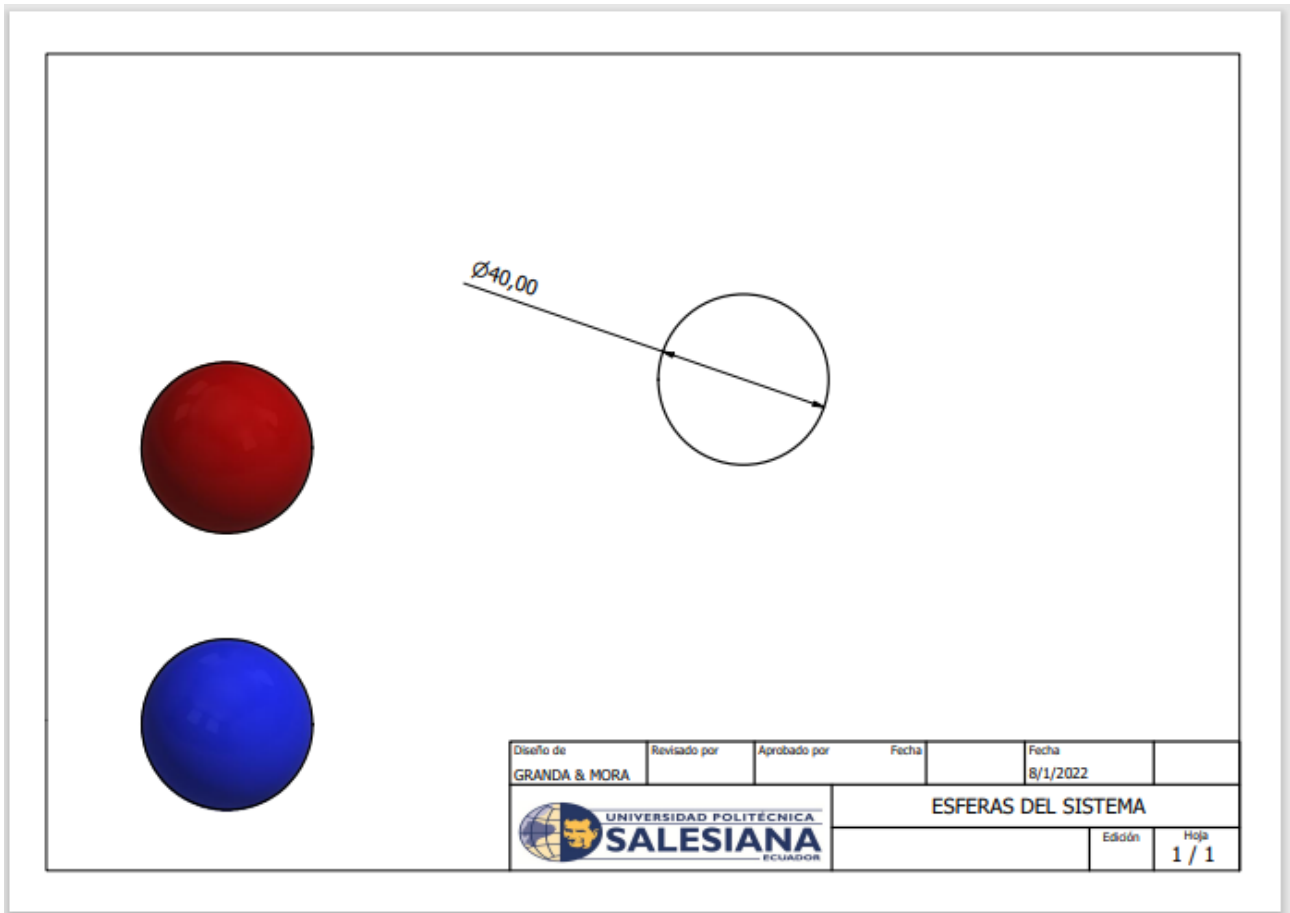


Figura 76. Lámina acotada de las esferas del sistema, por C.Granda y C.Mora, Inventor Autodesk

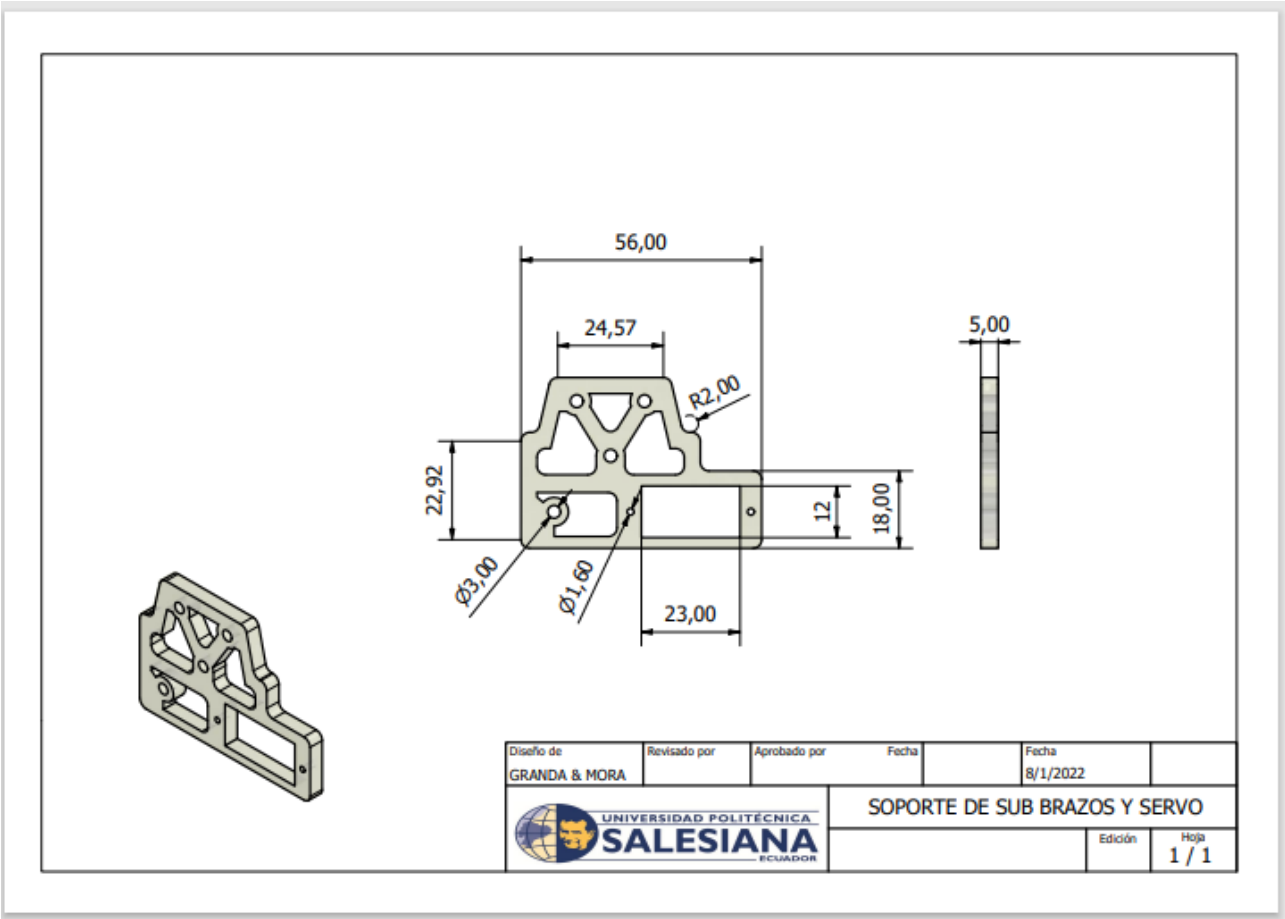


Figura 77. Lámina acotada del soporte de los sub brazos y servo, por C.Granda y C.Mora, Inventor Autodesk

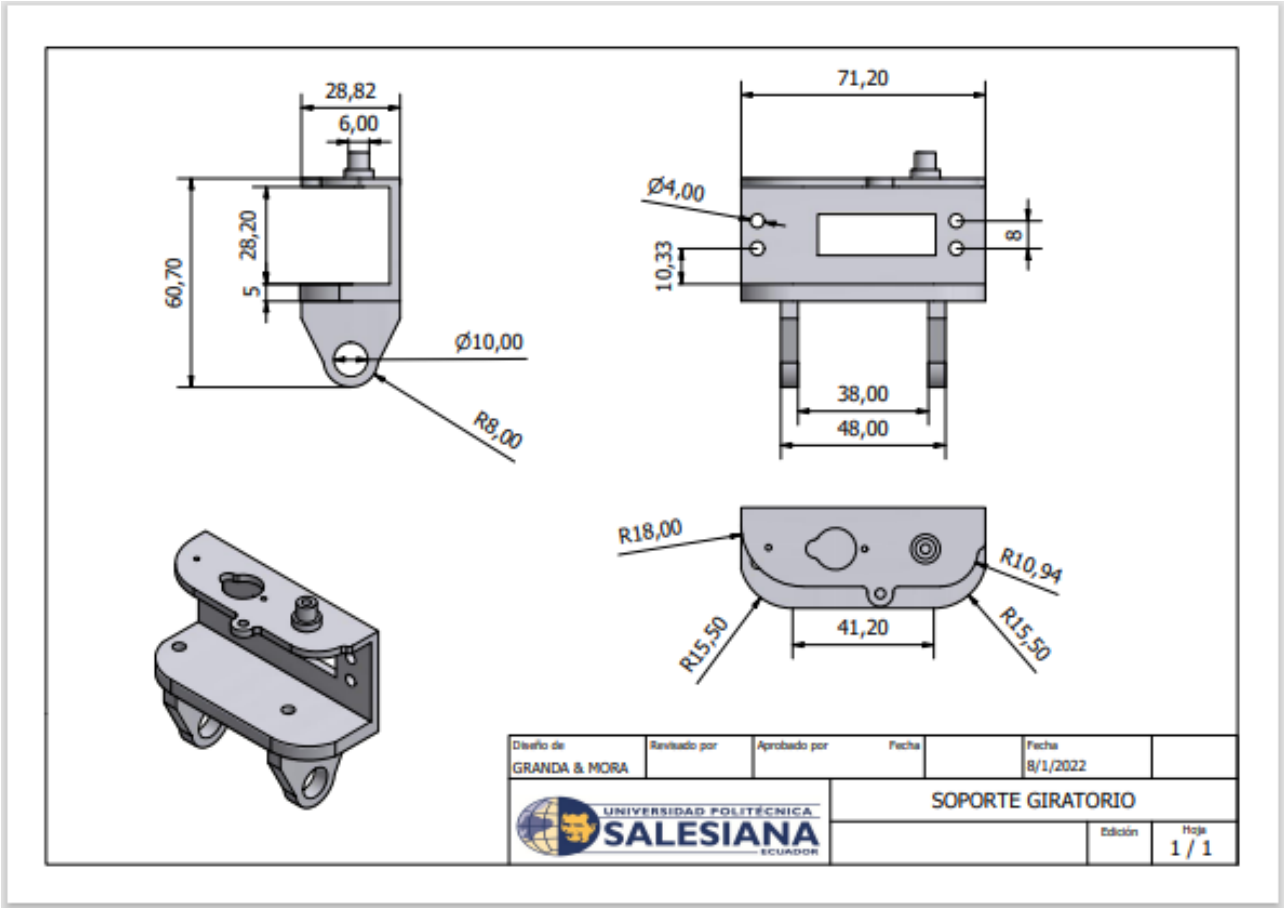


Figura 78. Lámina acotada del soporte del sistema de abertura, por C.Granda y C.Mora, Inventor Autodesk

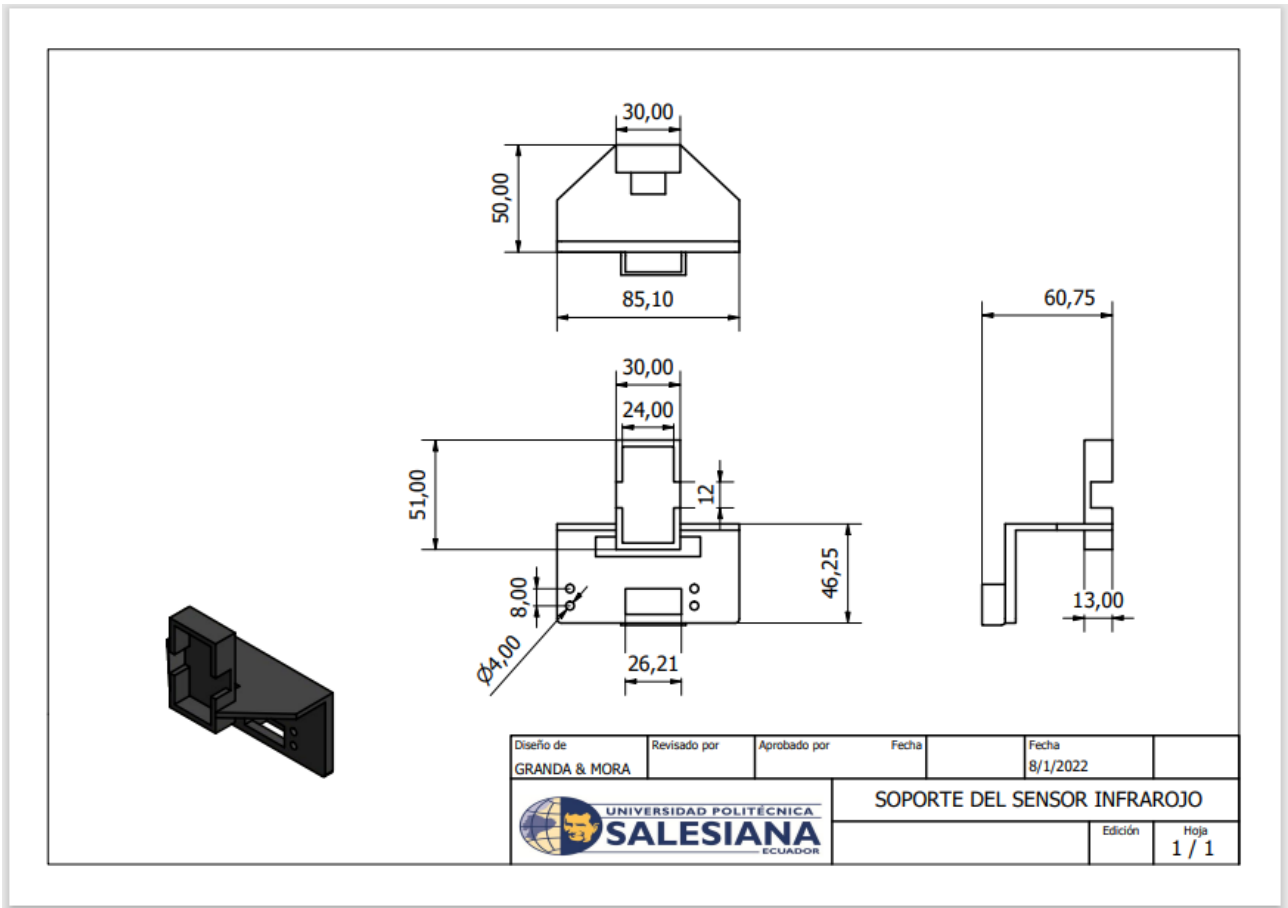


Figura 79. Lámina acotada del soporte del sensor infrarrojo, por C.Granda y C.Mora, Inventor Autodesk

Planto en 2d para la realización del corte láser en acrílico del primer prototipo funcional.

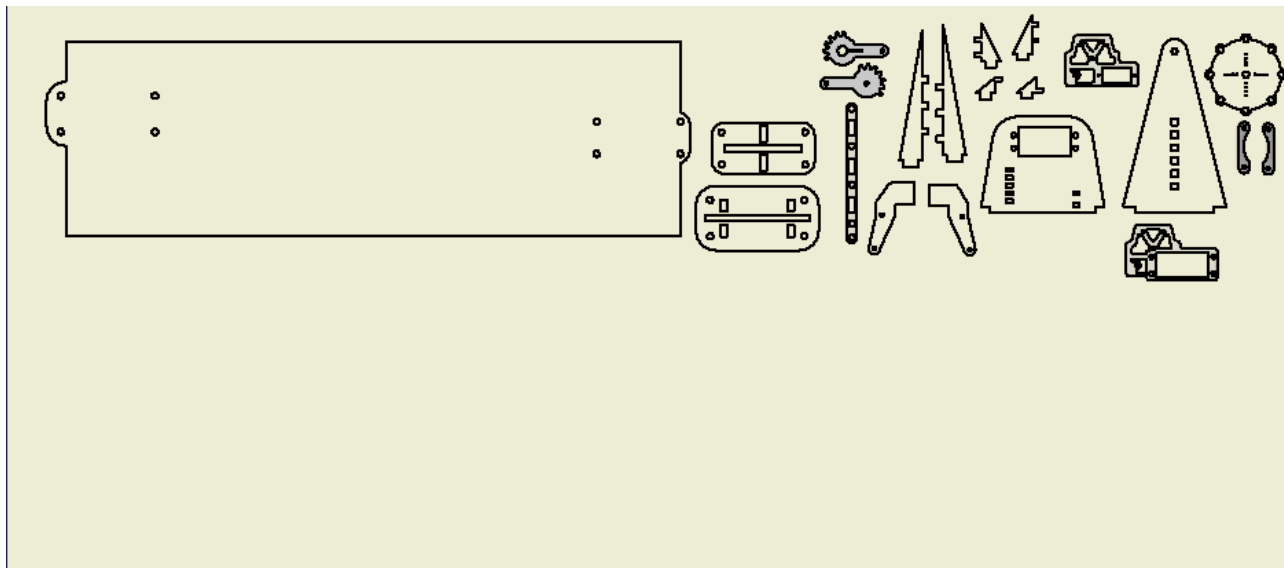


Figura 80. Lámina del contorno del prototipo en acrílico, por C.Granda y C.Mora, Inventor Autodesk

APÉNDICE B  
EQUIPOS DE MANUFACTURA



Figura 81. Corte láser del prototipo en acrílico, por C.Granda y C.Mora

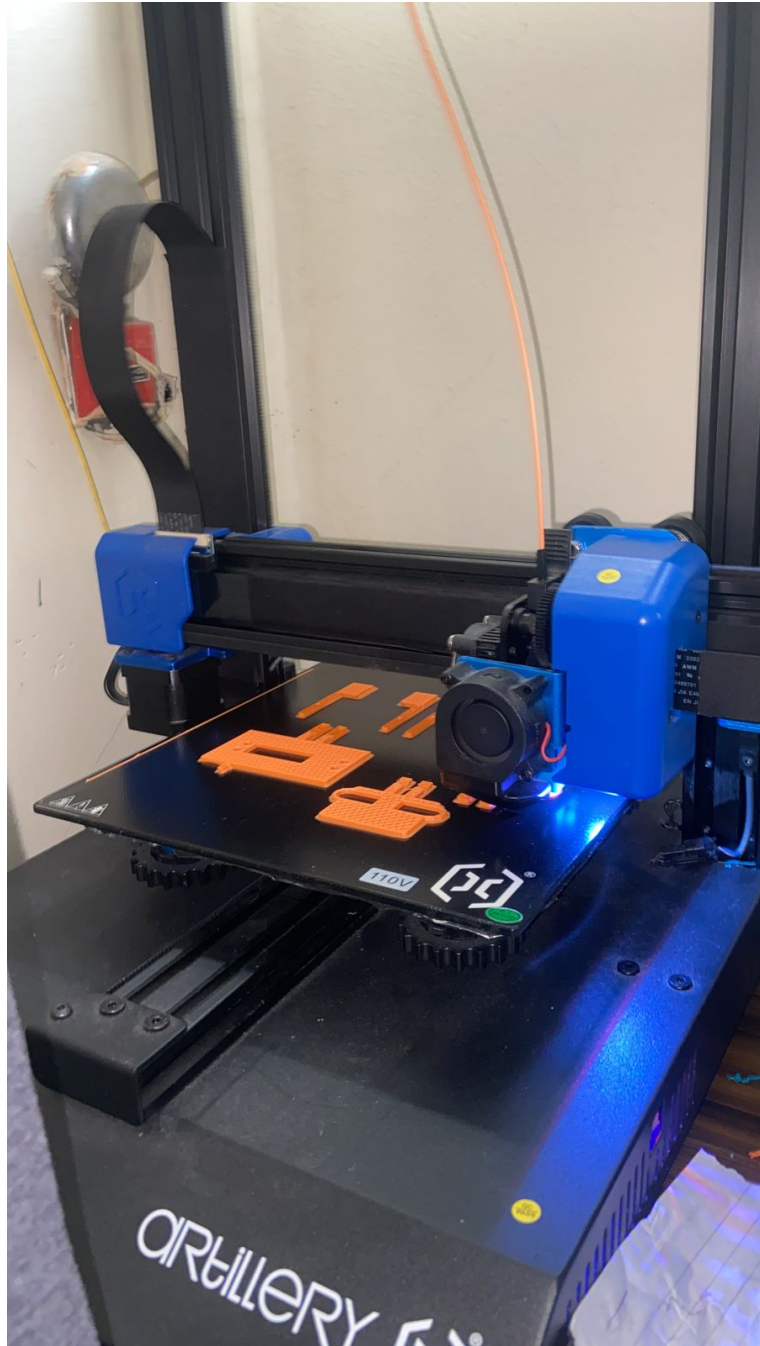


Figura 82. Impresión en 3D de las piezas.



Figura 83. Corte láser de la estructura metálica



## APÉNDICE C HERRAMIENTAS LABVIEW

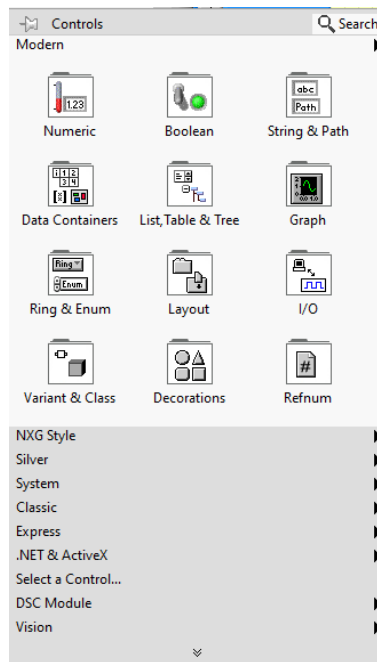


Figura 84. Menú de funciones, estructuras de control e indicadores del panel frontal en LabVIEW [15].

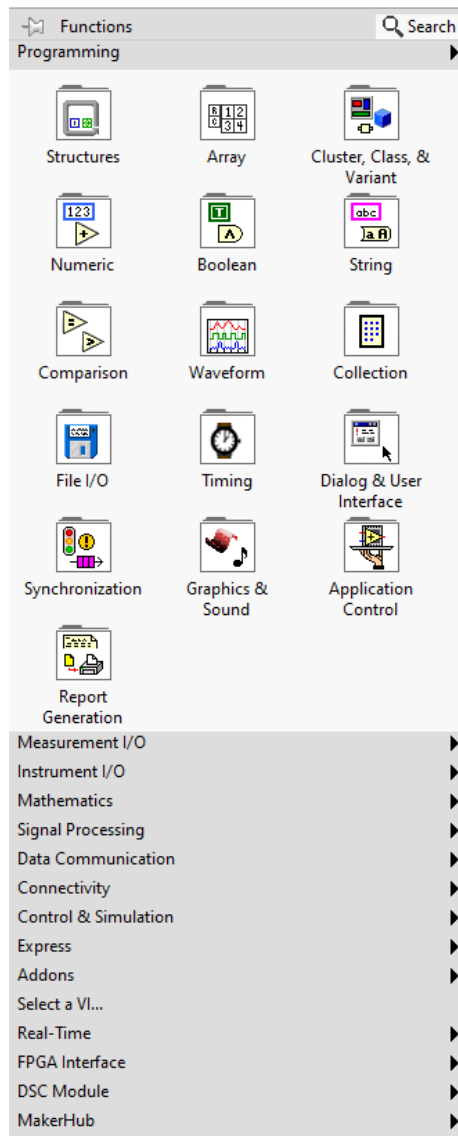


Figura 85. Paleta de funciones del diagrama de bloques del VI[15].

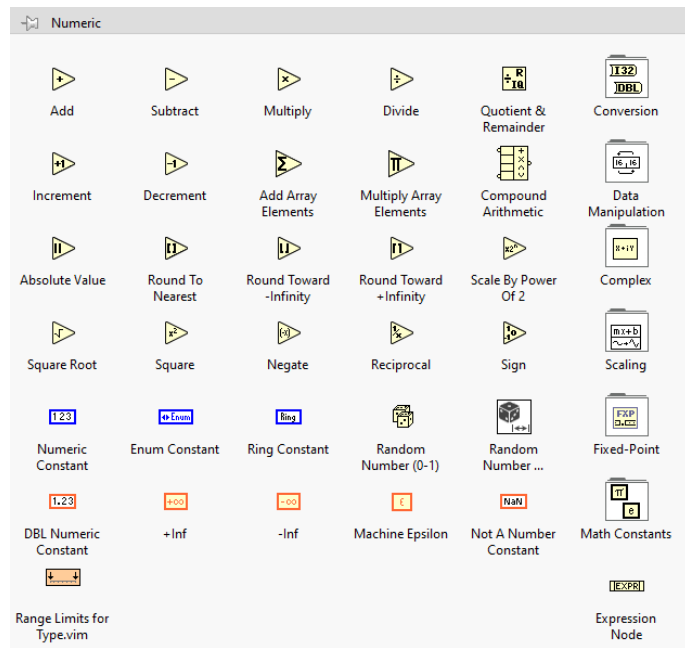


Figura 86. Paleta de funciones numéricas [15].

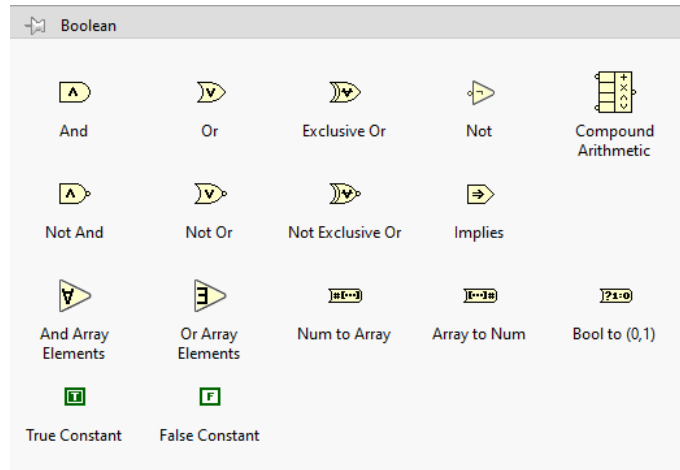


Figura 87. Paleta de funciones lógicas [15].

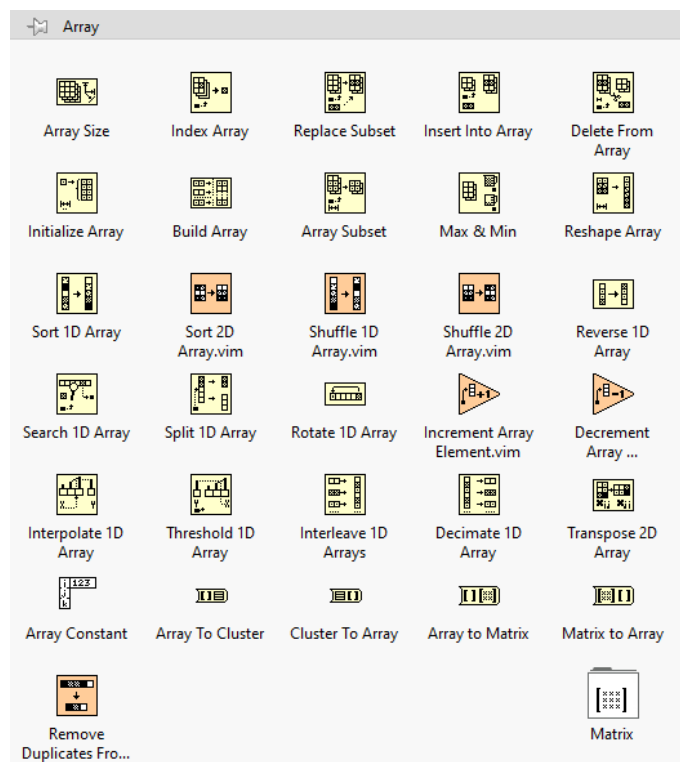


Figura 88. Paleta de funciones de hileras(arrays) [15].

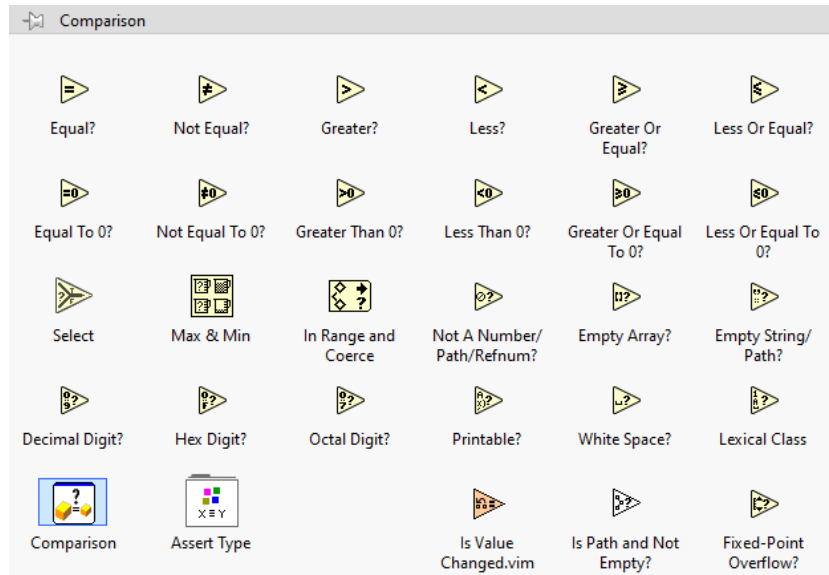


Figura 89. Paleta de funciones de comparación [15].

## APÉNDICE D PROGRAMACIÓN EN LABVIEW

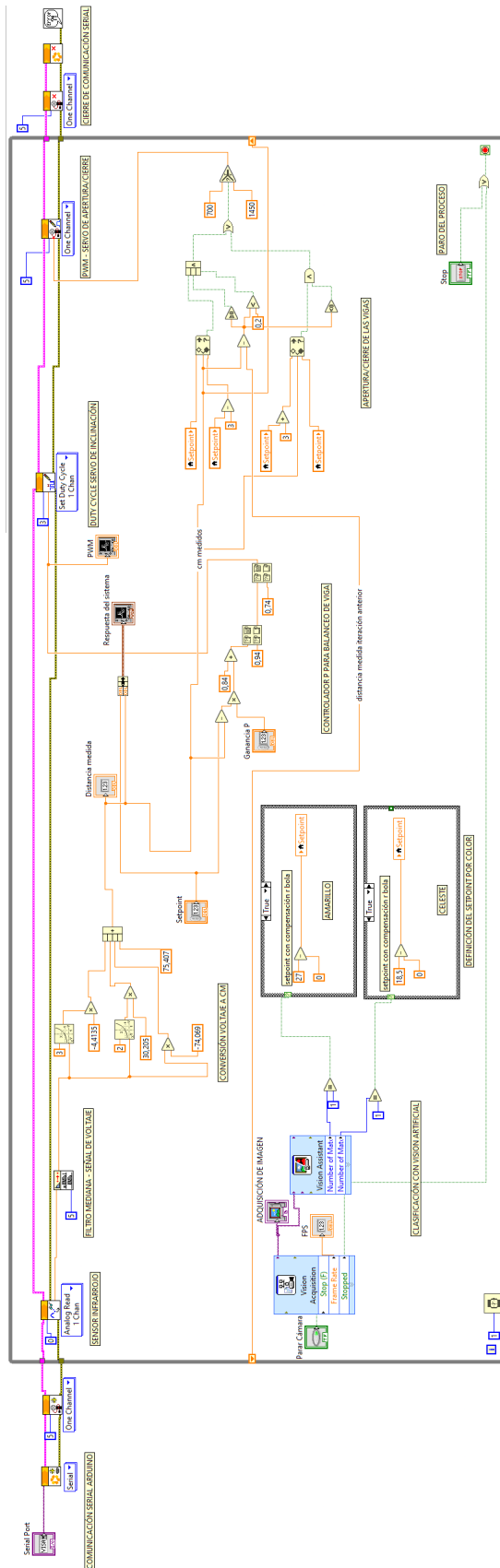


Figura 90. Diagrama de bloques completo del controlador PID en LabVIEW