



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN**

**DESARROLLO DE UNA PLANTA DE LABORATORIO PARA EL**  
**CONTROL DE PROCESOS TÉRMICOS, DE VELOCIDAD,**  
**POSICIÓN Y DESPLAZAMIENTO ANGULAR UTILIZANDO**  
**CONTROLADORES DE 32 BITS**

**Trabajo de Integración Curricular**

**Tipo:** Proyecto Técnico

Presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN**

**AUTOR:**

**DANIEL MAURICIO CEVALLOS CALVOPIÑA**

Riobamba – Ecuador

2023



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN**

**DESARROLLO DE UNA PLANTA DE LABORATORIO PARA EL  
CONTROL DE PROCESOS TÉRMICOS, DE VELOCIDAD,  
POSICIÓN Y DESPLAZAMIENTO ANGULAR UTILIZANDO  
CONTROLADORES DE 32 BITS.**

**Trabajo de Integración Curricular**

**Tipo:** Proyecto Técnico

Presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN**

**AUTOR:** DANIEL MAURICIO CEVALLOS CALVOPIÑA

**DIRECTOR:** Ing. JORGE LUIS PAUCAR SAMANIEGO MSc.

Riobamba – Ecuador

2023

© 2023, Daniel Mauricio Cevallos Calvopiña

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, Daniel Mauricio Cevallos Calvopiña, declaro que el presente Trabajo de Integración Curricular es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este Trabajo de Integración Curricular; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 17 de noviembre de 2023



**Daniel Mauricio Cevallos Calvopiña**

**050389756-3**


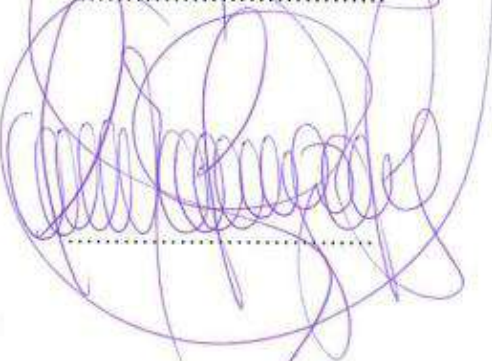

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN**

El Tribunal del Trabajo de Integración Curricular certifica que: El Trabajo de Integración Curricular; Tipo: Proyecto Técnico, **DESARROLLO DE UNA PLANTA DE LABORATORIO PARA EL CONTROL DE PROCESOS TÉRMICOS, DE VELOCIDAD, POSICIÓN Y DESPLAZAMIENTO ANGULAR UTILIZANDO CONTROLADORES DE 32 BITS**, realizado por el señor : **DANIEL MAURICIO CEVALLOS CALVOPIÑA**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

Ing. Pablo Eduardo Lozada Yánez  
**PRESIDENTE DEL TRIBUNAL**

Ing. Jorge Luis Paucar Samaniego  
**DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR**

Ing. Paul Patricio Romero Riera  
**ASESOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR**

FIRMA	FECHA
	2023-11-17
	2023-11-17
	2023-11-17

## **DEDICATORIA**

A ti, mi Dios Jehová, pues tú haces que todo sea posible. A mi señor padre Edison Alfonso Cevallos Pozo por ser mi apoyo, mi guía y mi ejemplo de hombre trabajador. A mi señora madre Silvia Patricia Calvopiña Chacón por ser la mejor madre, amiga, consejera y por cultivar en mí el valor de nunca darme por vencido sea cual sea la circunstancia que atraviere. A mis hermanos David, Abigail, Karla y mi sobrino Jorge por ser la fuerza que me impulsa a salir adelante día con día. A mi novia Josselyn Lisbeth Taipe Sinchiguano por ser la mejor mujer que puedo tener a mi lado.

**Daniel**

## **AGRADECIMIENTO**

A ti mi Dios, por las bendiciones infinitas que recibo de tu parte. A mis padres por todo el amor infinito, cariño, apoyo y esfuerzo que me brindan para triunfar y cumplir mis metas. A mis hermanos por estar ahí presentes dándome fuerza para superar los obstáculos que se me atraviesan. A mi sobrino por llenarme de alegría con sus ocurrencias. A mi novia por su amor, su cariño y su apoyo incondicional. A mis abuelitos, tíos y primos por ayudarme en todo este proceso. A mis amigos Xavier Molina, Fernando Tarco, Luis Tenorio y Jhony Chango por demostrarme que la verdadera amistad existe. A cada uno de mis docentes que a lo largo de mi carrera universitaria dentro de mi querida Escuela Superior Politécnica de Chimborazo que me han brindado sus conocimientos para formarme como profesional y en especial a mi director de trabajo de integración curricular el Ing. Jorge Luis Paucar por brindarme su tiempo y su guía para la elaboración de este proyecto.

**Daniel**

## ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS.....	xii
ÍNDICE DE ILUSTRACIONES.....	xiv
ÍNDICE DE ANEXOS.....	xviii
RESUMEN.....	xix
SUMMARY.....	xx
INTRODUCCIÓN.....	1

### CAPÍTULO I

<b>1. DIAGNÓSTICO DEL PROBLEMA.....</b>	<b>2</b>
1.1. Antecedentes.....	2
1.2. Planteamiento del problema.....	3
1.2.1. <i>Sistematización del problema</i> .....	3
1.3. Justificación teórica.....	4
1.3.1. <i>Justificación aplicativa</i> .....	5
1.4. Objetivo general.....	6
1.4.1. <i>Objetivos específicos</i> .....	7
1.5. Metodología.....	7

### CAPÍTULO II

<b>2. MARCO TEÓRICO.....</b>	<b>9</b>
2.1. Entrenador de planta de control “EPC” o plantas de control de procesos.....	9
2.1.1. <i>Experimentos que contiene el EPC</i> .....	10
2.2. Sistema de control.....	10
2.2.1. <i>Elementos de un sistema de control</i> .....	11
2.2.2. <i>Clasificación de los sistemas de control</i> .....	11
2.2.3. <i>Ventajas y desventajas de los sistemas de control</i> .....	12
2.3. Sistemas de control incorporados en EPC.....	13
2.4. Tarjetas de desarrollo.....	14
2.4.1. <i>Comparativa de tarjetas de desarrollo</i> .....	15
2.5. Comunicación serial ESP.....	16
2.5.1. <i>Tipos de comunicación aplicados en ESP</i> .....	16



2.5.2.	<i>Comparativa de tipos de comunicación serial en ESP</i> .....	17
2.6.	<b>Elementos de adquisición de datos</b> .....	18
2.6.1.	<i>Sensor de temperatura</i> .....	18
2.6.1.1.	<i>Comparativa de sensores de temperatura</i> .....	18
2.6.2.	<i>Encoders ópticos infrarrojos</i> .....	19
2.6.3.	<i>Comparativa de encoders ópticos infrarrojos</i> .....	19
2.7.	<b>Elementos de actuación</b> .....	20
2.7.1.	<i>Motor DC</i> .....	20
2.7.1.1.	<i>Comparativa de motores DC</i> .....	20
2.7.2.	<i>Motor a pasos (Stepper)</i> .....	21
2.7.2.1.	<i>Comparativa de motores a pasos</i> .....	22
2.7.3.	<i>Led de alta potencia</i> .....	23
2.8.	<b>Software de programación gráfica utilizado en EPC</b> .....	23
2.9.	<b>Fuente de alimentación usado en EPC</b> .....	25

### CAPÍTULO III

3.	<b>MARCO METODOLÓGICO</b> .....	26
3.1.	<b>Requerimiento de la planta de laboratorio</b> .....	26
3.2.	<b>Concepción de la arquitectura general del prototipo</b> .....	27
3.2.1.	<i>Etapas de adquisición de datos</i> .....	27
3.2.2.	<i>Etapas de comunicación</i> .....	28
3.2.3.	<i>Etapas de control y visualización</i> .....	28
3.2.4.	<i>Etapas de actuadores</i> .....	29
3.3.	<b>Diseño de la arquitectura de las etapas de la PLCP</b> .....	29
3.3.1.	<i>Etapas de adquisición y comunicación de datos</i> .....	29
3.3.2.	<i>Etapas de comunicación, control y actuación</i> .....	30
3.4.	<b>Selección del hardware para el PSES</b> .....	30
3.4.1.	<i>ESP32 38 Pines ESP WROOM 32</i> .....	31
3.4.2.	<i>Fuente conmutada MP1584</i> .....	32
3.4.3.	<i>Driver DRV8825</i> .....	32
3.4.4.	<i>Driver TB6612FNG</i> .....	33
3.4.5.	<i>Relé de estado sólido G3MB-202P</i> .....	34
3.4.6.	<i>Motorreductor JAG25-370</i> .....	35
3.4.7.	<i>Led de potencia de 10W</i> .....	36
3.4.8.	<i>Motor NEMA 17HD48002H-22B</i> .....	37

3.4.9.	<i>Encoder óptico infrarrojo H2010</i> .....	37
3.5.	<b>Esquema de conexión de la planta de laboratorio</b> .....	38
3.5.1.	<i>Esquema de conexión de la planta de laboratorio conformado por las etapas de adquisición de datos, comunicación, control y visualización y etapa de actuadores ...</i> .....	38
3.6.	<b>Diseño del software para la PLCP</b> .....	41
3.6.1.	<i>Software de desarrollo para la planta de laboratorio</i> .....	41
3.6.2.	<i>Software requerido para la comunicación de la planta de laboratorio</i> .....	42
3.6.2.1.	<i>Creación de programa en Arduino IDE</i> .....	42
3.6.2.2.	<i>Creación de programas en LabVIEW</i> .....	43
3.6.3.1.	<i>Programa software correspondiente al control de posición angular</i> .....	43
3.6.3.2.	<i>Programa software correspondiente al control manual de velocidad del motor DC</i> .	46
3.6.3.3.	<i>Programa software correspondiente al control manual de activación de SSR</i> .....	48
3.6.3.4.	<i>Programa software correspondiente al control de temperatura manual</i> .....	50
3.6.3.5.	<i>Programa software correspondiente al control PID de temperatura</i> .....	52
3.6.3.6.	<i>Programa software correspondiente al control PID de velocidad del motor DC</i> .....	54
3.7.	<b>Interfaz de control y visualización de la información</b> .....	56
3.7.1.	<i>Interfaz de la práctica de control de posición angular</i> .....	56
3.7.2.	<i>Interfaz de control manual de velocidad del motor DC</i> .....	57
3.7.3.	<i>Interfaz de control de activación de SSR</i> .....	58
3.7.4.	<i>Interfaz de control de temperatura manual</i> .....	59
3.7.5.	<i>Interfaz de control PID de temperatura</i> .....	59
3.7.6.	<i>Interfaz de control PID de velocidad del motor DC</i> .....	60
3.8.	<i>Estructura del prototipo</i> .....	61

## CAPÍTULO IV

4.	<b>MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS</b> .....	64
4.1.	<b>Pruebas de la etapa de adquisición de datos</b> .....	66
4.1.1.	<i>Validación de los elementos de sensado de la etapa de adquisición</i> .....	66
4.2.	<b>Pruebas de la etapa de comunicación</b> .....	68
4.2.1.	<i>Repetibilidad en la comunicación entre la etapa de adquisición, comunicación y control y visualización</i> .....	68
4.2.2.	<i>Pruebas de funcionamiento de las prácticas a realizarse con la planta de laboratorio de control de procesos</i> .....	69
4.2.2.1.	<i>Prueba de funcionamiento de la práctica de control de posición angular</i> .....	70

4.2.2.2.	<i>Prueba de funcionamiento de la práctica de control manual de velocidad</i>	71
4.2.2.3.	<i>Prueba de funcionamiento de la práctica control de activación de SSR</i>	72
4.2.2.4.	<i>Prueba de funcionamiento de la práctica de control de temperatura manual</i>	74
4.2.2.5.	<i>Prueba de funcionamiento de la práctica de control PID de temperatura</i>	75
4.2.2.6.	<i>Prueba de funcionamiento de la práctica de control PID de velocidad del motor DC</i>	76
4.3.	<b>Obtención de las funciones de transferencia de cada una de las plantas que componen la PLCP</b>	78
4.3.1.	<b><i>Función de transferencia de la planta de temperatura</i></b>	78
4.3.1.1.	<i>Identificación del sistema en Matlab de la planta de temperatura</i>	78
4.3.1.2.	<i>Ajuste de ganancias del controlador PID de la planta de temperatura</i>	80
4.3.2.	<b><i>Función de transferencia de la planta del Motor DC</i></b>	81
4.3.2.1.	<i>Identificación del sistema en Matlab de la planta del motor DC</i>	82
4.3.2.2.	<i>Ajuste de ganancias del controlador PID de la planta de temperatura</i>	84
4.4.	<b>Consumo energético de la PLCP</b>	85
4.5.	<b>Análisis económico para la construcción del PSES</b>	86
4.6.	<b>Funcionamiento general de la PLCP</b>	87
4.7.	<b>Manual de usuario de la PLCP</b>	89
4.7.1.	<i>Requerimientos de hardware</i>	89
4.7.2.	<i>Requerimientos de software</i>	89
4.7.3.	<i>Prerrequisitos para realizar las practicas</i>	89
4.8.	<b>Manual de prácticas de laboratorio</b>	90
4.8.1.	<i>Practica 1</i>	91
4.8.2.	<i>Practica 2</i>	98
4.8.3.	<i>Practica 3</i>	103
4.8.4.	<i>Practica 4</i>	107
4.8.5.	<i>Practica 5</i>	111
4.8.6.	<i>Practica 6</i>	117

## CAPÍTULO V

5.	<b>CONCLUSIONES Y RECOMENDACIONES</b>	122
5.1.	<b>Conclusiones</b>	122
5.2.	<b>Recomendaciones</b>	124

**BIBLIOGRAFÍA**  
**ANEXOS**

## ÍNDICE DE TABLAS

<b>Tabla 2-1:</b>	Experimentos incluidos en la EPC de National Instruments .....	10
<b>Tabla 2-2:</b>	Elementos de un sistema de control.....	11
<b>Tabla 2-3:</b>	Ventajas y desventajas de los sistemas de control en lazo abierto.....	12
<b>Tabla 2-4:</b>	Ventajas y desventajas de los sistemas de control en lazo cerrado.....	13
<b>Tabla 2-5:</b>	Comparativa de protocolos de comunicación serial aplicados en Arduino .....	15
<b>Tabla 2-6:</b>	Comparativa de protocolos de los tipos de comunicación serial en ESP.....	17
<b>Tabla 2-7:</b>	Comparativa de sensores de temperatura.....	18
<b>Tabla 2-8:</b>	Comparativa de encoders ópticos infrarrojos.....	19
<b>Tabla 2-9:</b>	Comparativa entre Motores DC.....	21
<b>Tabla 2-10:</b>	Comparativa de motores a pasos.....	22
<b>Tabla 3-1:</b>	Especificaciones técnicas principales de ESP32.....	31
<b>Tabla 3-2:</b>	Fuente conmutada MP1584 .....	32
<b>Tabla 3-3:</b>	Especificaciones técnicas del Driver DRV8825 .....	33
<b>Tabla 3-4:</b>	Especificaciones técnicas del Driver TB6612FNG.....	34
<b>Tabla 3-5:</b>	Especificación técnica del relevador de estado sólido G3MB-202P.....	35
<b>Tabla 3-6:</b>	Datos técnicos de la luz led de 10W .....	36
<b>Tabla 4-1:</b>	Validación del sensor de temperatura DS18B20 .....	67
<b>Tabla 4-2:</b>	Comunicación entre la etapa de adquisición, de comunicación y módulo de control y visualización .....	69
<b>Tabla 4-3:</b>	Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de posición angular.....	70
<b>Tabla 4-4:</b>	Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de velocidad manual.....	72
<b>Tabla 4-5:</b>	Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de activación de SSR.....	73
<b>Tabla 4-6:</b>	Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de activación de control de temperatura manual .....	74
<b>Tabla 4-7:</b>	Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de activación de control PID de temperatura .....	75
<b>Tabla 4-8:</b>	Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control PID de velocidad.....	76

<b>Tabla 4-9:</b>	Datos experimentales encontrados para los controladores PID de las plantas de temperatura y motor DC .....	85
<b>Tabla 4-10:</b>	Consumo energético según la practica realizada en la PLCP .....	85
<b>Tabla 4-11:</b>	Análisis económico para la construcción del PLCP .....	86

## ÍNDICE DE ILUSTRACIONES

<b>Ilustración 2-1:</b>	Entrenador de planta de control.....	9
<b>Ilustración 2-2:</b>	Representación de un sistema.....	10
<b>Ilustración 2-3:</b>	Sistema de control en lazo abierto.....	11
<b>Ilustración 2-4:</b>	Sistema de control en lazo cerrado.....	12
<b>Ilustración 2-5:</b>	Diagrama de bloques de un controlador On/Off.....	13
<b>Ilustración 2-6:</b>	Diagrama de bloques de un controlador PID en un sistema de lazo cerrado .....	14
<b>Ilustración 2-7:</b>	Logo de software de ingeniería LabVIEW.....	24
<b>Ilustración 2-8:</b>	DAQ 6009 LabVIEW con entradas análogas y digitales.....	24
<b>Ilustración 3-1:</b>	Arquitectura general de la planta de laboratorio para el control de procesos .....	28
<b>Ilustración 3-2:</b>	Diagrama de bloques de la etapa de adquisición y comunicación.....	30
<b>Ilustración 3-3:</b>	Etapas de comunicación, control y visualización.....	30
<b>Ilustración 3-4:</b>	ESP32.....	31
<b>Ilustración 3-5:</b>	Fuente conmutada MP1584.....	32
<b>Ilustración 3-6:</b>	Driver DRV8825.....	33
<b>Ilustración 3-7:</b>	Driver para motor TB6612FNG.....	34
<b>Ilustración 3-8:</b>	Relé de estado sólido G3MB-202P.....	35
<b>Ilustración 3-9:</b>	Motorreductor JAG25-370.....	36
<b>Ilustración 3-10:</b>	Motor NEMA 17HD48002H-22B.....	37
<b>Ilustración 3-11:</b>	Encoder óptico infrarrojo H2010.....	38
<b>Ilustración 3-12:</b>	Diseño PCB de la conexión electrónica de las etapas de adquisición de datos, comunicación, control y visualización y etapa de actuadores.....	40
<b>Ilustración 3-13:</b>	Esquema de conexión electrónica de la planta de laboratorio (Etapa de adquisición y comunicación).....	40
<b>Ilustración 3-14:</b>	Esquema de conexión electrónica de la PLCP (Etapa de activación de actuadores).....	41
<b>Ilustración 3-15:</b>	Diagrama de flujo del programa de control de posicionamiento angular..	45
<b>Ilustración 3-16:</b>	Diagrama de flujo del control manual de velocidad del motor DC.....	47
<b>Ilustración 3-17:</b>	Diagrama de flujo del control de activación de SSRs.....	49
<b>Ilustración 3-18:</b>	Diagrama de flujo del control de temperatura On/Off.....	51
<b>Ilustración 3-19:</b>	Diagrama de flujo del control PID de temperatura.....	53
<b>Ilustración 3-20:</b>	Diagrama de flujo del control PID de velocidad de un motor DC.....	55

<b>Ilustración 3-21:</b>	Panel frontal de la práctica de control de posición angular .....	57
<b>Ilustración 3-22:</b>	Panel frontal de la práctica de control manual de velocidad del motor DC58	
<b>Ilustración 3-23:</b>	Panel frontal de la práctica de control de activación de SSR .....	58
<b>Ilustración 3-24:</b>	Panel frontal de la práctica de control de temperatura manual .....	59
<b>Ilustración 3-25:</b>	Panel frontal de la práctica de control de control PID de temperatura .....	60
<b>Ilustración 3-26:</b>	Panel frontal de la práctica de control PID de velocidad del motor DC....	61
<b>Ilustración 3-27:</b>	Caja de conexiones Heyiarbeit de plástico ABS a prueba de polvo impermeable IP65 .....	62
<b>Ilustración 3-28:</b>	Diseño 3D del Montaje de placa PCB dentro de la caja de conexiones Heyiarbeit .....	62
<b>Ilustración 4-1:</b>	Arquitectura general de la planta de laboratorio.....	64
<b>Ilustración 4-2:</b>	Obtención de las funciones de transferencia de la planta de temperatura .	79
<b>Ilustración 4-3:</b>	Función de transferencia tf2 obtenida de la planta de Temperatura .....	79
<b>Ilustración 4-4:</b>	Función de transferencia tf2 obtenida de la planta de Temperatura .....	80
<b>Ilustración 4-5:</b>	Ganancias de controlador de la función de transferencia tf1 obtenida de la planta de Temperatura .....	81
<b>Ilustración 4-6:</b>	Ganancias de controlador de la función de transferencia tf2 obtenida de la planta de Temperatura .....	81
<b>Ilustración 4-7:</b>	Obtención de las funciones de transferencia de planta del Motor DC.....	82
<b>Ilustración 4-8:</b>	Función de transferencia tf1 obtenida de la planta del Motor DC.....	83
<b>Ilustración 4-9:</b>	Función de transferencia tf2 obtenida de la planta del Motor DC.....	83
<b>Ilustración 4-10:</b>	Ganancias de controlador de la función de transferencia tf1 obtenida de la planta de Temperatura .....	84
<b>Ilustración 4-11:</b>	Ganancias de controlador de la función de transferencia tf1 obtenida de la planta de Temperatura .....	84
<b>Ilustración 4-12:</b>	Implementación de la planta de laboratorio de procesos (PLCP) finalizada .....	88
<b>Ilustración 4-13:</b>	Posición inicial del indicador en el encoder infrarrojo H2010 .....	91
<b>Ilustración 4-14:</b>	Vista superior de la PLCP.....	92
<b>Ilustración 4-15:</b>	Conexión de cable de alimentación y comunicación.....	93
<b>Ilustración 4-16:</b>	Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de posición .....	94
<b>Ilustración 4-17:</b>	Mensaje de error en la comunicación serial en la práctica de control de posición.....	95
<b>Ilustración 4-18:</b>	Ingreso de valores de TIEMPO( $\mu$ s) y $\text{ÁNGULO}(\text{°})$ .....	96



<b>Ilustración 4-19:</b>	Visualización de TIEMPO( $\mu$ s) ingresado y pulsos generados para un desplazamiento de $180^\circ$ .....	96
<b>Ilustración 4-20:</b>	Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de velocidad manual.....	99
<b>Ilustración 4-21:</b>	Mensaje de error en la comunicación serial en la práctica de control de velocidad manual .....	100
<b>Ilustración 4-22:</b>	Ingreso de valores PWM en el control numérico .....	101
<b>Ilustración 4-23:</b>	Velocidad en RPM generada por la señal PWM ingresada al sistema ....	101
<b>Ilustración 4-24:</b>	Gráfica Velocidad (RPM) vs Tiempo.....	102
<b>Ilustración 4-25:</b>	Controles booleanos y paleta de selección en panel frontal de la práctica de control de velocidad manual .....	104
<b>Ilustración 4-26:</b>	Mensaje de error en la comunicación serial en la práctica de control de activación de SSRs .....	105
<b>Ilustración 4-27:</b>	Selección y activación de relé SSR .....	106
<b>Ilustración 4-28:</b>	Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de temperatura manual .....	108
<b>Ilustración 4-29:</b>	Mensaje de error en la comunicación serial en la práctica de control de temperatura manual .....	109
<b>Ilustración 4-30:</b>	Ingreso del valor PWM en el control numérico.....	109
<b>Ilustración 4-31:</b>	Incremento en el valor de la temperatura en función al valor de PWM ..	110
<b>Ilustración 4-32:</b>	Gráfica Temperatura ( $^\circ$ C) vs Tiempo .....	110
<b>Ilustración 4-33:</b>	Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control PID de temperatura .....	113
<b>Ilustración 4-34:</b>	Mensaje de error en la comunicación serial en la práctica de control PID temperatura .....	114
<b>Ilustración 4-35:</b>	Valor de Setpoint ingresado y de temperatura actual .....	115
<b>Ilustración 4-36:</b>	Gráfica de Temperatura vs Tiempo de la señal de temperatura ante la acción del controlador PID sintonizado mediante Autotuning de LabVIEW .....	115
<b>Ilustración 4-37:</b>	Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control PID de velocidad.....	118
<b>Ilustración 4-38:</b>	Mensaje de error en la comunicación serial en la práctica de control PID de velocidad.....	119
<b>Ilustración 4-39:</b>	Valor de Setpoint ingresado y velocidad actual.....	120
<b>Ilustración 4-40:</b>	Gráfica de velocidad (RPM) vs Tiempo de la señal de temperatura ante la acción del controlador PID sintonizado mediante Autotuning de LabVIEW .....	121

<b>Ilustración 4-41:</b>	Frenado de eje del motor para generar perturbación en el proceso de velocidad.....	121
--------------------------	------------------------------------------------------------------------------------	-----

## ÍNDICE DE ANEXOS

- ANEXO A:** HOJA DE DATOS DEL ESP32 38 PINES ESP WROOM 32
- ANEXO B:** HOJA DE DATOS DEL ESP32 DRIVER M1584
- ANEXO C:** HOJA DE DATOS DEL DRIVER DRV8825
- ANEXO D:** HOJA DE DATOS DEL DRIVER TB6612FNG
- ANEXO E:** HOJA DE DATOS DEL RELEVADOR DE ESTADO SÓLIDO G3MB-202P
- ANEXO F:** HOJA DE DATOS DEL MOTORREDUCTOR JGA25-370
- ANEXO G:** HOJA DE DATOS DE LUZ LED DE 10W
- ANEXO H:** HOJA DE DATOS MOTOR NEMA 17HD48002H-22B
- ANEXO I:** DIAGRAMA DE BLOQUES CORRESPONDIENTES A LAS INTERFACES DE CONTROL Y VISUALIZACIÓN DE CADA UNA DE LAS PRACTICAS A DESARROLLARSE EN LA PLCP AGRAMA DE BLOQUES CORRESPONDIENTES A LAS INTERFACES DE CONTROL Y VISUALIZACIÓN DE LA INFORMACIÓN
- ANEXO J:** HOJA DE DATOS DE LA CAJA DE CONEXIONES HEYIARBEIT DE PLÁSTICO ABS
- ANEXO K:** EVIDENCIA DEL MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS
- ANEXO L:** PROMACIÓN REALIZADA EN ARDUINO IDE 2.1.1

## RESUMEN

El objetivo de este proyecto de integración curricular consistió en diseñar una planta de laboratorio para el control de procesos térmicos, de velocidad, posición y desplazamiento angular, haciendo uso de controladores de 32 bits. Se emplearon métodos teóricos que proporcionaron la base conceptual necesaria, así como métodos empíricos que permitieron validar y evaluar la funcionalidad del trabajo. La planta de laboratorio está compuesta por cuatro etapas, la etapa de adquisición integrada por los sensores, la etapa de comunicación implementada en una tarjeta de desarrollo ESP32, la etapa de control y visualización constituida por programas realizados en LabVIEW referentes a cada práctica y la etapa de actuadores donde se ejecutan las ordenes enviadas por la tarjeta de desarrollo, donde los actuadores están conformados por un motor DC y un motor a pasos y una luz led de potencia. De las pruebas realizadas se obtuvo que el error relativo máximo dado por el sensor de temperatura DS18B20 es de 2.44%, determinando que la recolección de información ejecutada por este sensor es de calidad. En la prueba de repetibilidad en la comunicación entre las etapas de adquisición, comunicación y control y visualización y etapa de actuadores en cada una de las prácticas realizadas, se determinó un coeficiente de variación del 0%, lo que indica una estabilidad de datos, confirmando que cada práctica se ejecutó de manera correcta y confiable. En lo que respecta a la alimentación, se verificó que la PLCP tiene un consumo energético real de 1876.12 mA en su estado activo máximo y de 266.63 mA en su estado activo mínimo, debido a los diversos actuadores utilizados en cada práctica. Se recomienda imprescindiblemente leer el manual de usuario antes de utilizar la planta de laboratorio, ya que cada proceso presenta características y modos de funcionamiento distintos propios de cada práctica.

**Palabras clave:** <PLANTA DE LABORATORIO>, <TARJETA DE DESARROLLO>, <SENSORES>, <ACTUADORES>, <COMUNICACIÓN>, <ADQUISICIÓN>, <CONTROL DE PROCESOS >, <LABVIEW(SOFTWARE)>.



  
1858-DBRA-UPT-2023

## SUMMARY

This curricular integration research project aimed to design a laboratory plant to control thermal processes, speed, position, and angular displacement using 32-bit controllers. Theoretical methods were employed to provide the necessary conceptual foundation and empirical methods to validate and evaluate the functionality of the work. The laboratory plant consists of four stages: the acquisition stage, which includes sensors; the communication stage, implemented ESP32 development board; the control and visualization stage, consisting of LabVIEW programs specific to each experiment; and the actuator stage, where commands sent by the development board are executed where the actuators include a DC motor, a stepper motor, and a high-power LED light. From the tests conducted, it was determined that the maximum relative error provided by the DS18B20 temperature sensor is 2.44%, indicating that the data collection performed by this sensor is of high quality. In the repeatability test in the communication between the stages of acquisition, communication and control, and visualization and actuator stage in each of the conducted experiments, a coefficient of variation of 0% was determined, which indicates data stability, confirming that each experiment was carried out correctly and reliably. Regarding the power supply, it was verified that the PLCP has a real energy consumption of 1876.12 mA in its maximum active state and 266.63 mA in its minimum active state, attributed to the various actuators used in each experiment. It is highly recommended to read the user manual before using the laboratory plant, as each process has different characteristics and operational modes specific to each experiment.

**Keywords:** <LABORATORY PLANT>, <DEVELOPMENT BOARD>, <SENSORS>, <ACTUATORS>, <COMMUNICATION>, <ACQUISITION>, <PROCESS CONTROL>, <LABVIEW (SOFTWARE)>.



Lenin Iván Lara Olivo  
0602546103

## **INTRODUCCIÓN**

El presente trabajo de investigación curricular se compone de cuatro capítulos que se detallan de la siguiente manera: el primer capítulo aborda el diagnóstico de la problemática, el segundo se centra en el marco teórico, el tercero en el enfoque metodológico, y finalmente, el cuarto capítulo se dedica al marco de análisis e interpretación de los resultados.

En el primer capítulo, se hace referencia a diversos puntos iniciales que sirven como entrada para el desarrollo del tema propuesto como lo son: antecedentes, planteamiento del problema, justificación teórica y aplicativa, tomando en cuenta los objetivos a desarrollarse en el presente proyecto, según los requerimientos de la planta de laboratorio para el control de procesos.

En el segundo capítulo se realiza una revisión bibliográfica orientada a entrenadores de plantas de control, sistemas de control, comunicación serial, elementos de adquisición y actuación y software de programación. Con el objetivo de seleccionar los elementos, tipos y familias de mejores características, haciendo que el usuario obtenga una comprensión teórica más profunda acerca del tema.

En el tercer capítulo se tratan puntos muy importantes para el desarrollo de una planta de laboratorio para el control de procesos (PLCP), tal como: planteamiento de los requerimientos, concepción de la arquitectura del prototipo, diseño de la arquitectura de cada uno de sus etapas, selección del hardware, esquemas de conexión, diseño de software, interfaces de control, visualización de la información y estructura del prototipo.

Finalmente, en el cuarto capítulo se realiza el análisis e interpretación de resultados posterior a la construcción de la PLCP, para ello se desarrolla la caracterización de las etapas de: adquisición de datos, comunicación, control y visualización y finalmente la etapa de actuadores y actuación.

# CAPÍTULO I

## 1. DIAGNÓSTICO DEL PROBLEMA

El presente capítulo cuenta con los siguientes apartados:

### 1.1. Antecedentes

Según (Rodríguez, 2013), la falta de acceso a entrenadoras de plantas de control, limita los conocimientos de los estudiantes en el diseño y modelado de sistemas de control automático, generando falta de experiencia en prácticas relacionadas a los campos de teoría de control, robótica, automatización industrial e instrumentación.

Según (Sunta y Yáñez, 2021) mencionaron que para el estudiante o docente representa un gran problema el no contar con un equipo compacto y de fácil uso que requiera una conexión simple, y que además permita la interacción de numerosos sensores y actuadores de una forma segura, ligera y confiable. Debido a que en el mercado actual se ofrece entrenadores de plantas de control convencionales, que dejan de ser funcionales si no cuentan con una tarjeta de adquisición de datos (DAQ).

En Ibarra-Ecuador estudiantes de la Universidad Técnica del Norte, realizaron el diseño y la construcción de módulos didácticos para el laboratorio virtual de instrumentación industrial (Altamirano y Méndez 2016), haciendo uso de elementos eléctricos y electrónicos más comunes en la industria. Emplearon programas como LabVIEW para la creación de la interfaz de control y ALTIUM DESIGNER para el desarrollo de las placas de circuitos impreso (PCB). Estos módulos permitieron realizar prácticas de control de temperatura, velocidad, control de posición y el control de un relé para accionamientos On/Off. Los autores indicaron que las practicas antes mencionadas se realizaron de una manera sencilla para comprobando el correcto funcionamiento de cada uno de los elementos.

Un estudiante de la Escuela Politécnica Nacional de la ciudad de Quito (Iza, 2015) realizo como proyecto de titulación el diseño y la implementación de un módulo de instrumentación electrónica. El autor menciona que el objetivo principal del proyecto fue enlazar el módulo de instrumentación electrónica con LabVIEW por medio del microcontrolador ATXMEGA32A4 encargado del manejo de las entradas y salidas analógicas. Donde además se utilizó un Circuito Integrado Programable (PIC) PIC18F2550 el cual cumplía con la función gestionar la

comunicación USB con el computador. Dado que el módulo estaba diseñado para estudiantes, se proporcionó un manual de usuario que resultó fundamental para la correcta operación de los dispositivos y accesorios incorporados en el módulo.

(Sunta y Yáñez, 2021) desarrollaron en la Universidad Politécnica Salesiana sede en Quito un entrenador inalámbrico de plantas de control basado en IoT (Internet de las cosas). Haciendo uso de microcontroladores NODEMCU ESP8226 para el control de los diferentes aspectos de las plantas a través de comunicación Wifi. Empleando una tarjeta de desarrollo Raspberry Pi 3 que actúa como servidor, procesando datos y tomando decisiones según algoritmos personalizados. El servidor también cuenta con una interfaz gráfica para control y observación de información. Cabe mencionar que los autores comentaron que el sistema ofrece una manera efectiva de aprender y experimentar con control de plantas utilizando tecnología inalámbrica e IoT.

Como se ha estudiado existen varios proyectos de investigación que cuentan con un mismo enfoque, pero con distintas arquitecturas, por lo que dotan de ventajas y desventajas diferentes. Es por eso que se debe realizar un análisis de cada aspecto que conforma la realización de este trabajo de integración curricular, tanto para su diseño y su futura implementación.

En la actualidad se pretende contar con una planta de laboratorio para el control de procesos, la cual este suficientemente equipada para llevar a cabo prácticas de temperatura, velocidad y posición, con el propósito de adecuarlo al medio tecnológico actual. Buscando generar innovaciones que mejoren los equipos comerciales existentes a un menor costo, con mejores características que los equipos convencionales y que sea de fácil de utilización por parte del usuario.

## **1.2. Planteamiento del problema**

¿Cuáles son los beneficios de desarrollar una planta de laboratorio para la adquisición de señales físicas, que nos permita controlar y supervisar las variables de los procesos térmico, de velocidad y desplazamiento angular usando microcontroladores de 32 bits?

### ***1.2.1. Sistematización del problema***

- ¿Qué marco teórico y metodología se utilizó en trabajos similares ya realizados y cuál fue su alcance y requerimientos cumplidos?



- ¿Qué elementos de hardware son necesarios para implementar y simular el proceso térmico, de velocidad y posición angular?
- ¿Qué algoritmos se deben implementar sobre un microcontrolador de 32 bits que permita controlar y supervisar las variables del proceso térmico, de velocidad y posición angular mediante una interfaz virtual?
- ¿Cómo se puede validar el correcto funcionamiento de la planta de laboratorio?
- ¿Qué procedimiento se debe seguir para simular los tres procesos establecidos en una práctica de laboratorio de control discreto?

### **1.3. Justificación teórica**

Actualmente, cada día se incrementa la aceleración del avance de las ciencias y el desarrollo tecnológico, debido a este avance se ha recurrido a reemplazar la lógica cableada en los sistemas de automatización y control de contactos por la lógica programada, ya que un proceso controlado brinda mejores garantías de calidad, seguridad y competitividad. Se requiere entonces que los estudiantes adquieran conocimientos teóricos y prácticos de instrumentación y programación, los cuales estén fomentados a la utilización de nuevos sistemas de control en procesos industriales y de nuevos dispositivos tecnológicos (Alcaciega y Simba, 2020).

Los autores (Iza, 2015) y (Sunta y Yáñez, 2021) mencionan en sus proyectos de titulación que la utilización de microcontroladores de 32 Bits son una gran opción para aplicaciones que involucren comunicación USB ya que muchos fabricantes de microcontroladores incorporan módulos USB en sus dispositivos para facilitar el desarrollo de aplicaciones de instrumentación, permitiendo un procesamiento y transferencia de datos más eficiente, lo que es esencial en el intercambio de información entre la planta entrenadora de control con el computador, para posteriormente realizar el control de actuadores y lectura de sensores.

En los laboratorios de la Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo, se encuentran disponibles entrenadores de plantas de control (EPC) de National Instruments, diseñados para llevar a cabo prácticas relacionadas con la medición y control de temperatura, velocidad y posicionamiento. Destacando que estos equipos requieren la presencia de un dispositivo de adquisición de datos (DAQ) por sus siglas en inglés (Data Acquisition). La función principal de la DAQ es la de medir, registrar y analizar los datos generados por los sensores e instrumentos ubicados en cada planta de control. Es esencial entender que tanto el EPC como la DAQ son componentes interdependientes, es decir que, si falta alguno de estos dos

dispositivos, el usuario no podrá realizar ninguna de las prácticas diseñadas para el entrenador de planta de control.

La crisis económica que está atravesando el país es un problema significativo para el gobierno, la sociedad y las universidades públicas, especialmente para aquellas que no se dedican a la investigación y producción. Esto se debió a los recortes constantes en el presupuesto de las instituciones públicas. Es por eso que (Caicedo et al. 2021) enfatizaron en la necesidad de implementar estrategias adecuadas para hacer frente a estos recortes, para así poder afrontar la crisis económica que impide la compra de nuevos equipos, laboratorios, insumos, materiales, etc.

Por todo lo antes mencionado surge la necesidad de diseñar una planta de laboratorio para el control de procesos usando microcontroladores de 32 Bits, la cual reforzará los conocimientos del usuario en las áreas de sistemas de control e instrumentación a través de prácticas de temperatura, velocidad y posición. Presentando un menor costo de fabricación y una mayor facilidad de uso, abriendo así paso a futuras mejoras en el prototipo.

### ***1.3.1. Justificación aplicativa***

Considerando la necesidad de desarrollar una planta de laboratorio para el control de procesos, la cual basa su funcionamiento en la realización de cuatro etapas fundamentales las cuales son: “adquisición de datos”, “comunicación”, “control y visualización” y además la etapa de actuadores.

La etapa de adquisición de datos estaría compuesta por los dispositivos de sensado, los que se encargarían de la recepción y envío de los estados de los actuadores. La cual en conjunto tendría tres elementos de adquisición de datos, un sensor de temperatura, un encoder infrarrojo y un encoder de cuadratura.

La etapa de comunicación sería la encargada de establecer la comunicación bidireccional entre la planta de laboratorio de control de procesos (PLCP) con el software LabVIEW, en este se elegiría la práctica a realizar según sea el caso, en donde la tarjeta de desarrollo que para nuestro caso será el ESP32 definiría que entradas y que salidas de este dispositivo dispondrá para tal proceso o para tal práctica.

La etapa de control y visualización correspondería a la interfaz del software de programación gráfica, quien actuará como herramienta de visualización y control, pues por medio de ella se

observará y se gestionará el proceso que se está ejecutando, donde se podría observar la interacción entre el software y hardware, lo que permitiría al usuario visualizar gráficos y resultados obtenidos de cada proceso.

La etapa de actuadores tendría como tarea fundamental transformar señales de control en movimientos físicos u otras acciones mecánicas. El proceso básico de esta etapa implicaría los siguientes pasos: primero, se enviarían señales de control al actuador desde el sistema, posteriormente, el actuador convertiría la señal eléctrica en energía mecánica o energía térmica.

Las cuatro etapas nombradas anteriormente estarían siempre obligadas a mantener una comunicación USB entre la PLCP con el computador. Ya que si alguna de las etapas llegaría a fallar el proceso seleccionado no podría desarrollarse de manera correcta, debido a que un proceso dependería del otro.

Se necesita hacer uso de programas independientes destinados a cada practica correspondientes a control de temperatura, velocidad y posición. Los cuáles serán realizados en el software de LabVIEW, estos nos servirán como interfaz gráfico de control y visualización de la información generada en cada proceso.

Se requiere la utilización de programas independientes diseñados específicamente para el control de temperatura, velocidad y posición en cada práctica. Estos programas se desarrollarán en el software LabVIEW y funcionarán como interfaces gráficas para controlar y visualizar la información generada en cada proceso.

Respecto a cuestiones de alimentación eléctrica, sería necesario tener en consideración que la fuente de voltaje debe ser escogida de acuerdo a un análisis de consumo energético. Ya que, al elegir una fuente de voltaje inadecuada, se pueden generar problemas o causar fallas. Un voltaje demasiado alto puede dañar los componentes o causar fallas en el sistema, mientras que un voltaje insuficiente puede llevar a un funcionamiento deficiente del sistema en cada proceso.

#### **1.4. Objetivo general**

Desarrollar una planta de laboratorio para la adquisición de señales físicas, manipulación y supervisión de variables controladas y manipuladas de un proceso térmico, de un proceso de velocidad y desplazamiento angular con base en un microcontrolador de 32 bits como interfaz de comunicación y control.

### ***1.4.1. Objetivos específicos***

- Investigar la metodología utilizada en trabajos previos, sus alcances y requerimientos.
- Diseñar el hardware de la planta analizando los sensores, actuadores, circuitos de acondicionamiento de señal, circuitos de adquisición de datos, microcontroladores, fuentes de alimentación y comunicación.
- Implementar los algoritmos que permitan una comunicación bidireccional PC-microcontrolador para supervisar y controlar las variables de los procesos.
- Validar el correcto funcionamiento de la planta basado en criterios de funcionalidad utilizando pruebas exhaustivas que aseguren la capacidad de cumplir las especificaciones requeridas.
- Desarrollar un manual de usuario que permita desarrollar una práctica de laboratorio térmica, de velocidad y desplazamiento angular.

### **1.5. Metodología**

Para el desarrollo del presente trabajo de integración curricular es necesario hacer uso de manera combinada de las siguientes técnicas y métodos de investigación:

Dentro de los métodos teóricos se tiene: revisión documental, la cual implica analizar y sintetizar la información existente, para el estudio de trabajos de titulación, blogs, páginas de internet, libros y cualquier otro tipo de información relacionada al tema

Los métodos teóricos son esenciales en la investigación académica ya que proporcionan la base conceptual y analítica necesaria para comprender, explicar y avanzar en el conocimiento en un campo específico. También influyen en la dirección y el enfoque de la investigación, lo que les otorga un papel central en el proceso de investigación.

Al emplear estas herramientas, los investigadores descomponen problemas en sus componentes esenciales, logrando así comprender y prever resultados. Además, este enfoque brinda la oportunidad de generar nuevas perspectivas y descubrimientos en disciplinas que abarcan desde la ciencia y la ingeniería hasta las ciencias sociales y humanas. De esta manera, los métodos teóricos proporcionan un sólido cimiento para la exploración y comprensión del mundo que nos rodea.

Los métodos empíricos que se emplearán, se incluyen la recopilación y análisis de datos a partir de observaciones o experiencias concretas. Estos métodos se basarán en la evidencia empírica, es decir, datos que se obtienen a través de la observación directa, experimentación, encuestas, entrevistas u otras técnicas de recolección de datos. A partir de esta recopilación de datos se podrá extraer conclusiones respaldadas por evidencia concreta. Esta forma de validar resulta esencial para evaluar la funcionalidad de nuestro trabajo de integración curricular.

## CAPÍTULO II

### 2. MARCO TEÓRICO

En el presente capítulo se realiza una breve descripción acerca de los conceptos de entrenador de planta de control e instrumentación, sistemas de control de procesos, tipos de controladores, comunicaciones, tarjetas de desarrollo, sensores de temperatura, velocidad y posición, motores DC, software de programación y de instrumentación virtual.

#### 2.1. Entrenador de planta de control “EPC” o plantas de control de procesos

“Es una placa electrónica que incluye varios sensores y actuadores típicos en los sistemas de instrumentación y control tales como temperatura, velocidad, posición, señales analógicas de corriente continua, alterna, digital, y tren de pulsos” (High, 2009, p.4).

El propósito de este equipo es simplificar la comprensión de conceptos relacionados con teoría de control e instrumentación, al ofrecer al usuario una serie de experimentos prácticos listos para utilizar. De esta manera, se reduce el tiempo necesario para diseñar y construir componentes electrónicos, se garantiza la compatibilidad de los sensores con los experimentos, y se brinda una experiencia directa con las características y problemáticas de sistemas físicos reales, como el ruido, la precisión y el acoplamiento AC/DC, en lugar de depender únicamente de simulaciones por computadora (High Lights, 2009).



**Ilustración 2-1:** Entrenador de planta de control

**Fuente:** (High, 2009).

### 2.1.1. Experimentos que contiene el EPC

En la *Tabla 2-1* describe brevemente un resumen de los experimentos con las que cuenta la EPC mencionada en el anterior apartado.

**Tabla 2-1:** Experimentos incluidos en la EPC de National Instruments

Experimento	Características
Control de Temperatura	Mediante un bombillo halógeno simula un dispositivo de calentamiento.
Control de Velocidad de Motor DC	El motor es regulado mediante una señal de voltaje de corriente continua (DC) .
Control de Posición de Motor Stepper	A medida que se activan las líneas digitales, el motor se mueve girando paso a paso.
Análisis de Sonido	La señal de sonido adquirida será procesada, analizada mediante espectros y otras mediciones.
Relé de Propósito General:	Usado para aplicaciones de conmutación de propósito general.

Fuente: (High, 2009).

Realizado por: Cevallos, Daniel, 2023.

Cada una de las practicas presentadas en la *Tabla 1-2* consta de una breve explicación del hardware utilizado y uno o más programas de aplicación desarrollados en el software de instrumentación virtual LabVIEW los cuales son de código abierto para que puedan ser usados como base para la creación de nuevos programas con mayor complejidad (High, 2009, p.5).

## 2.2. Sistema de control

Un sistema de control automático se configura a partir de elementos interconectados con la habilidad de autorregularse. Dicho de otro modo, se establece una estructura donde los componentes se organizan de manera que el sistema pueda operar en modo automático. En el caso de un sistema o componente específico que pueda ser controlado, se introduce una señal de entrada  $r(t)$  con el propósito de obtener una respuesta o salida  $y(t)$ . Esta disposición puede ser visualizada mediante la representación gráfica de bloques interconectados (Hernández, 2010, p.3).



**Ilustración 2-2:** Representación de un sistema

Realizado por: Cevallos, Daniel, 2023.

El problema de control se resume en elegir y adaptar un conjunto particular de componentes de manera que, al conectarse entre sí, el sistema resultante presente un comportamiento específico (Hernández 2010, p.4).

### 2.2.1. Elementos de un sistema de control

Los sistemas de control son utilizados en una amplia variedad de aplicaciones para regular y mantener el funcionamiento adecuado de sistemas y procesos. A continuación, en la *Tabla 2-2* se describen los elementos fundamentales de un sistema de control:

**Tabla 2-2:** Elementos de un sistema de control

<b>Entrada:</b> Es el punto donde se recoge información sobre el proceso a través del uso de sensores
<b>Sistema:</b> Interpreta la señal de llegada, la compara con cierta señal de referencia y genera otra señal ajustada de acuerdo a los valores de referencia
<b>Salida:</b> Es el resultado de las decisiones realizadas por el controlador, y ejecutadas por los actuadores presentes en el proceso

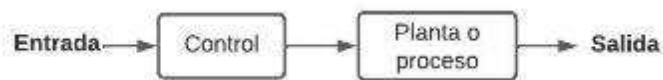
**Fuente:** (CEIISA, 2016).

**Realizado por:** Cevallos, Daniel, 2023.

### 2.2.2. Clasificación de los sistemas de control

Los sistemas de control se clasifican en sistemas de lazo abierto y sistemas de lazo cerrado.

**Sistema de control lazo abierto.** - Un sistema de control en bucle abierto, también conocido como sistema de control sin retroalimentación, se caracteriza por no recibir información sobre el valor actual de la variable del producto o proceso que se desea controlar (Brunete, San Segundo y Herrero 2020a).



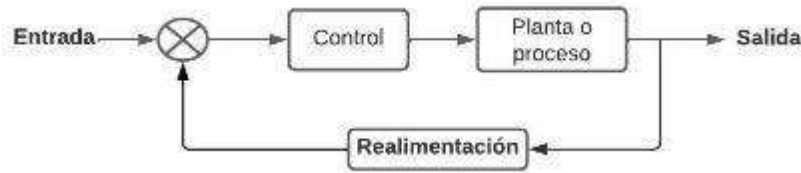
**Ilustración 2-3:** Sistema de control en lazo abierto

**Realizado por:** Cevallos, Daniel, 2023.

“La capacidad que tales sistemas tienen para ejecutar una acción con exactitud depende de su calibración. En general, los sistemas de lazo abierto están regulados por base de tiempo” (Hernández, 2010, p.4).



**Sistema de control en lazo cerrado.** - Conocidos como sistemas de control con retroalimentación, se caracterizan por recibir información en sus entradas sobre el valor de la variable que están controlando. Debido a esto, se dice que estos sistemas están realimentados, ya que utilizan la retroalimentación para ajustar y mejorar continuamente el control del sistema (Brunete, San Segundo y Herrero 2020b).



**Ilustración 2-4:** Sistema de control en lazo cerrado

**Realizado por:** Cevallos, Daniel, 2023.

El concepto de retroalimentación implica realizar una comparación; en este caso, se compara la salida real del sistema con el comportamiento deseado. Si el sistema requiere ajustes, se aplica una acción correctiva para corregir el proceso que está siendo controlado (Hernández 2010).

### 2.2.3. Ventajas y desventajas de los sistemas de control

Los sistemas de control ofrecen una serie de ventajas y desventajas dependiendo de su aplicación y complejidad. La *Tabla 2-3* presenta ventajas y desventajas de los sistemas de control en lazo abierto.

**Tabla 2-3:** Ventajas y desventajas de los sistemas de control en lazo abierto

Sistema en lazo abierto	
Ventajas	Desventajas
Son más simples y requieren menos componentes y circuitos	No cuentan con retroalimentación para corregir errores o compensar perturbaciones externas.
Tienden a ser más económicos en términos de costos de diseño, construcción y operación.	No pueden adaptarse automáticamente a cambios en las condiciones o en los requisitos de control.
Al no tener que esperar retroalimentación, los sistemas en lazo abierto pueden responder de manera más rápida.	No se puede medir o corregir el error entre la salida real y la salida deseada.
Dado que no hay retroalimentación en el sistema, los sistemas en lazo abierto son inherentemente estables en condiciones normales de operación.	Pueden ser menos estables en sistemas complejos o en presencia de retroalimentación no lineal.

**Fuente:** (Pineda, 2014).

**Realizado por:** Cevallos, Daniel, 2023.

La *Tabla 2-4* presenta ventajas y desventajas de los sistemas de control en lazo cerrado.

**Tabla 2-4:** Ventajas y desventajas de los sistemas de control en lazo cerrado

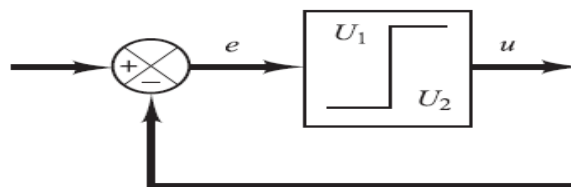
Sistema en lazo cerrado	
Ventajas	Desventajas
Tienen la capacidad de medir la salida real y compararla con la referencia deseada.	Son más complejos debido a la inclusión de sensores, retroalimentación y algoritmos de control.
Son más robustos y estables frente a perturbaciones y cambios en las condiciones del sistema.	Tienden a tener un mayor costo en comparación con los sistemas en lazo abierto.
Tienen la capacidad de adaptarse a cambios en el entorno o en los requisitos del control.	Existe la posibilidad de que se produzcan inestabilidades si no se diseñan y ajustan adecuadamente
Recibe información de la retroalimentación, los sistemas pueden corregir errores y desviaciones en la salida real.	Dependen en gran medida de la retroalimentación para ajustar y corregir el sistema.
Reducen la sensibilidad a perturbaciones externas, ya que el control se ajusta continuamente en función de la retroalimentación.	Requieren de una cuidadosa sintonización y ajuste de los parámetros del controlador para lograr un rendimiento óptimo.

Realizado por: Cevallos, Daniel, 2023.

### 2.3. Sistemas de control incorporados en EPC

Los entrenadores de plantas de control pueden usar una variedad de controladores dependiendo de la aplicación y los requisitos específicos de la planta en estudio. Algunos de los sistemas de control más comunes utilizados en las prácticas o experimentos son el control On/Off y control PID.

**Control On/Off.** - En un sistema de control de dos posiciones, el dispositivo que realiza la acción de control tiene únicamente dos opciones fijas, que generalmente son encendido y apagado. El control de dos posiciones o encendido y apagado es un método sencillo y económico, por lo cual su aplicación es amplia en sistemas de control, tanto en entornos industriales como domésticos (Ogata, 2010).

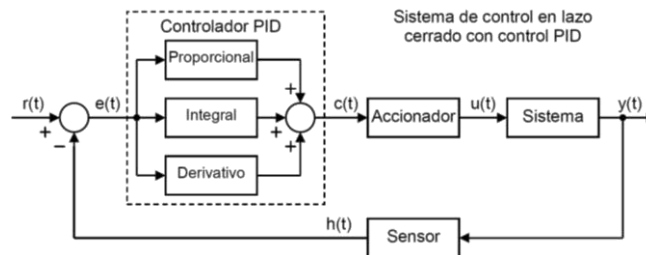


**Ilustración 2-5:** Diagrama de bloques de un controlador On/Off

Fuente: (Ogata, 2010).

**Control PID.** - La acción de control proporcional-integral-derivativa es la integración de la acción de control proporcional, la acción de control integral y la acción de control derivativa. Esta combinación se conoce como PID y ofrece las ventajas de cada una de las acciones de control por separado (Ogata, 2010).

La función de transferencia de un control PID está representada por:



**Ilustración 2-6:** Diagrama de bloques de un controlador PID en un sistema de lazo cerrado

**Fuente:** (Ogata, 2010).

## 2.4. Tarjetas de desarrollo

Una Tarjeta de desarrollo es un componente electrónico que incorpora un microcontrolador o dispositivo lógico, junto con elementos como puertos, conectores y reguladores. Estos elementos permiten a los usuarios acceder de manera sencilla y rápida a los periféricos para llevar a cabo aplicaciones y la realización de pruebas (Castro y Osorio, 2011). Entre las más importantes se destacan las siguientes:

**Arduino.** - Arduino es una plataforma de hardware de código abierto que se basa en placas programables para la creación de dispositivos digitales. En el mercado existen varios modelos de placas Arduino, que difieren en tamaño, procesadores, conectores y capacidades. Sin embargo, todas comparten una característica fundamental: incorporan un procesador programable con memoria RAM y almacenamiento flash, así como pines de entrada y salida para la comunicación con otros dispositivos, sensores o elementos tanto en forma analógica como digital. Además, estas placas cuentan con algún tipo de conexión USB o similar que permite programar el dispositivo y proporcionarle energía. Estas capacidades permiten que estas pequeñas placas puedan cargar código desde un ordenador y ejecutarlo, incluso de forma autónoma siempre que dispongan de una fuente de alimentación (Romero, 2021, p.2).

**Raspberry Pi.** - Es un mini ordenador altamente funcional que dispone de una placa de dimensiones reducidas. Viene equipada con todas las conexiones habituales que se encuentran en un ordenador estándar, como puertos USB, conectividad a internet y red, salidas de audio y video, entre otras. Al añadir los periféricos necesarios, como el mouse, teclado y una salida de video HDMI, podemos conectarla fácilmente a un televisor o monitor, lo que nos permite utilizarla tanto como un potente ordenador de sobremesa como un versátil centro multimedia. Gracias a su flexibilidad y asequible precio, Raspberry Pi ha superado su objetivo inicial y se ha expandido su uso a otros campos, como la robótica y la domótica (Lucas, 2019, p.2).

**FPGA.** - Son extremadamente versátiles y encuentran una amplia gama de aplicaciones debido a su naturaleza programable, que permite adaptarlos a tareas específicas. Estos chips se utilizan en diversas áreas, desde naves espaciales y coches inteligentes hasta servidores de telecomunicaciones. Su interior suele incluir diferentes bloques de procesamiento y conectores que permiten configurar su lógica de programación para realizar operaciones complejas o actuar como simples puertas lógicas. En resumen, podríamos afirmar que los FPGA son aplicables en cualquier entorno que demande procesamiento de datos, y en ciertas ocasiones, pueden comportarse como microprocesadores, asumiendo la denominación de "Soft Microprocessors", en función de su programación y configuración específica (Romero, 2021, p.1).

**ESP.** – Es un conjunto de microcontroladores confiables en ambientes industriales y ESP32 es uno de los más conocidos en la industria. La serie ESP32 está pensada para operar en rangos de temperatura desde los -40 °C, hasta los 125 °C. Se recomienda su uso para dispositivos móviles, ya que tiene un ultra bajo consumo de energía. Además, el circuito incluye antena, amplificador de potencia, poco susceptible al ruido electrónico, filtros y módulos para administrar el consumo de potencia. Finalmente, para comunicarse ESP32 cuenta con módulos basados en protocolos wifi, bluetooth, SPI, I2C y UART (Mier, 2020,p .6).

#### **2.4.1. Comparativa de tarjetas de desarrollo**

La *Tabla 5-2* muestra las características más importantes de las de tarjetas de desarrollo que pueden ser usadas para realización de nuestro proyecto.

**Tabla 2-5:** Comparativa de protocolos de comunicación serial aplicados en Arduino

<b>Parámetro</b>	<b>Arduino</b>	<b>Raspberry Pi</b>	<b>FPGA</b>	<b>ESP</b>
Procesador	Atmel AVR	ARM Cortex	Intel Cyclone	Tensilica Xtensa LX
Entrada de voltaje	7-12 V	5V	5-12 V	3.3-5 V
Consumo energético	15-93 mA	100 mA-3A	40-80 mA	20-80 mA

Redes incorporadas	Ninguna	ETH, Wifi	No	Wifi
RAM	2-8 KB	256 MB - 8 GB	75-1355 KB	80-520 KB
Almacenamiento	Eepron (1-4 KB)	MicroSD	Flash 512 KB -1.6 MB	Flash 4-16 MB
Precio	\$8-65	\$20-200	\$10-500	\$8-20

**Realizado por:** Cevallos, Daniel, 2023.

En función de la *Tabla 2-5* la tarjeta de desarrollo a ser usada en nuestro trabajo de integración curricular es la ESP, debido a que presenta mejores características como: mayor capacidad de memoria, almacenamiento, dimensiones, redes de comunicación y precio. Además de las características antes mencionadas, también sería una excelente opción para proyectos que requieran una comunicación serial debido a su integración de UART, mediante la cual existiría una comunicación bidireccional y una interacción entre dispositivos.

## 2.5. Comunicación serial ESP

La comunicación serial se refiere a un método de intercambio de datos entre dispositivos donde la información se transmite y recibe un bit a la vez a través de un canal de comunicación. La comunicación serial se emplea para enviar y recibir datos entre la placa Arduino y otros dispositivos externos, como un ordenador u otros dispositivos periféricos. Esta forma de comunicación permite la transferencia de datos bidireccional entre el microcontrolador y los dispositivos externos, facilitando tareas como la sincronización de información y el control de dispositivos externos desde la placa Arduino (Castaño, 2019, p.6).

### 2.5.1. Tipos de comunicación aplicados en ESP

**UART.** - El protocolo UART (Universal Asynchronous Receiver-Transmitter) es un estándar de comunicación serial ampliamente utilizado en dispositivos electrónicos para la transmisión de datos. A diferencia de otros protocolos seriales, no requiere una señal de reloj compartida entre los dispositivos comunicantes. En su lugar, se basa en el envío y recepción de datos de forma asíncrona, lo que significa que los datos se transmiten sin un reloj de sincronización. (Rohde y Schwarz, 2023, p.2).

La velocidad de transmisión de datos en un enlace UART se especifica en baudios (bps) y puede configurarse según los requisitos de la aplicación. Las velocidades de baudios comunes incluyen 9600, 19200, 115200, etc. Tanto el dispositivo transmisor como el receptor deben configurarse para operar a la misma velocidad de baudios para garantizar una comunicación adecuada (ARDUINO, 2022, p.21).

**I2C.** - Es un protocolo de comunicación serial y un puerto utilizado para la transferencia de bits entre dos dispositivos digitales. Define la estructura de la trama de datos y las conexiones físicas necesarias. El puerto I2C se compone de dos cables de comunicación, SDA y SCL. Una de las características destacadas de este protocolo es su capacidad para conectar hasta 127 dispositivos esclavos utilizando estos dos cables, con velocidades que pueden ser de 100, 400 y 1000 Kbits/s. El protocolo I2C es ampliamente utilizado en la comunicación con sensores digitales debido a sus ventajas, como la confirmación de datos recibidos dentro de la misma trama, a diferencia del puerto Serial. La capacidad de conectar múltiples dispositivos al mismo bus es una de las principales ventajas del I2C (HeTPro, 2017,p .3).

**SPI.** - Conocido como Interfaz Serial para Periféricos, posibilita la transferencia simultánea y bidireccional de datos a alta velocidad (de hasta 10 MB) en distancias cortas (de hasta 30 cm). Esto permite una transferencia rápida y eficiente de información entre dispositivos, lo cual resulta especialmente adecuado para aplicaciones que requieren una velocidad de transmisión elevada, como la comunicación con pantallas LCD, memorias flash, sensores y otros periféricos. Además, es ampliamente utilizado en microcontroladores, como Arduino para establecer la comunicación entre distintos componentes en un sistema electrónico (Machicao, 2017, p.7).

### 2.5.2. Comparativa de tipos de comunicación serial en ESP

La *Tabla 2-6* detalla las características más importantes de los tipos de comunicación aplicados en ESP.

**Tabla 2-6:** Comparativa de protocolos de los tipos de comunicación serial en ESP

<b>Parámetro</b>	<b>UART</b>	<b>I2C</b>	<b>SPI</b>
Señal de reloj	No	Si	Si
Tipo de Comunicación	Asíncrona	Síncrona	Síncrona
Conexión múltiples dispositivos	No	Si	Si
Velocidad de transmisión	Baja	Baja	Alta

**Realizado por:** Cevallos, Daniel, 2023.

Con base a la *Tabla 2-6* el protocolo de comunicación que hemos seleccionado es el UART como el protocolo más adecuado ante los demás protocolos. Esto se debe a que uno de los requisitos fundamentales para el diseño de nuestra planta de laboratorio destinada al control de procesos, es establecer una comunicación serial de 9600 baudios entre la planta de laboratorio y el computador, siendo esto posible por el protocolo antes mencionado y seleccionado.

## 2.6. Elementos de adquisición de datos

La adquisición de datos en un entrenador de planta de control se refiere al proceso de recopilación y registro de información relevante proveniente de los sensores y actuadores presentes en dicho entrenador (High, 2009 p.43). El proceso de adquisición de datos en un entrenador de planta de control generalmente está compuesto por los siguientes elementos:

### 2.6.1. Sensor de temperatura

Un sensor de temperatura es un dispositivo que identifica cambios en la temperatura de un gas, líquido u otro material, una vez medida esta magnitud la convierte en una señal eléctrica que se envía a un sistema electrónico el cual provoca ajustes específicos sobre él, para regular la temperatura de manera adecuada según las condiciones establecidas para el correcto funcionamiento del proceso (Saldias, 2018, p.4).

#### 2.6.1.1. Comparativa de sensores de temperatura

La *Tabla 2-7* se exponen las características más relevantes de sensores de temperatura que son usado con más frecuencia en proyectos y en EPC.

**Tabla 2-7:** Comparativa de sensores de temperatura

Parámetro	LM35	DTH11	DS18B20
Entrada de voltaje	5 V (recomendado)	3-5 V	3-5 V
Consumo energético	60 $\mu$ A	0.5 mA-2.5 mA	1 $\mu$ A-1.5 mA
Rango de medición	-55°C a +150°C	0°C a 50°C	-55°C a 125°C
Precisión	$\pm 0.5^\circ\text{C}$	$\pm 2.0^\circ\text{C}$	$\pm 0.5^\circ\text{C}$
Tipo de salida	Analógica	Analógica	Digital
Precio	\$2	\$3	\$3

**Realizado por:** Cevallos, Daniel, 2023.

Según la *Tabla 2-7*, el sensor de temperatura más adecuado es el DS18B20. Entre sus características destacadas se encuentran una precisión de  $\pm 0.5^\circ\text{C}$ , un bajo consumo de corriente y un amplio rango de medición desde  $-55^\circ\text{C}$  hasta  $+125^\circ\text{C}$ . Sin embargo, su atributo más sobresaliente es su interfaz digital, que permite la conexión de varios sensores en un único bus de datos, simplificando significativamente el diseño y tamaño de los circuitos. Por estas razones, el

sensor DS18B20 se posiciona como la elección óptima para nuestras necesidades de medición de temperatura.

### 2.6.2. Encoders ópticos infrarrojos

Un encoder óptico está formado por dos dispositivos optoelectrónicos, uno que funciona como emisor o fuente de luz, y el otro como receptor. Normalmente, se utiliza una combinación de fotodiodo y fototransistor como este par optoelectrónico (Corona, Abarca y Mares, 2014).

La determinación de la posición o velocidad de un motor es de gran relevancia en el campo de la robótica. Para lograrlo, hay diversas opciones disponibles, y una de las más comunes es el empleo de encoders ópticos. Estos dispositivos tienen la función de convertir el movimiento angular o lineal del motor en pulsos eléctricos, permitiendo así su interpretación por parte del controlador del sistema (Naylamp Mechatronics, 2023).

### 2.6.3. Comparativa de encoders ópticos infrarrojos

La *Tabla 2-8* indica las características más relevantes e importantes de los encoders ópticos usados para medir velocidad y posicionamiento en motores de corriente continua y motores a pasos.

**Tabla 2-8:** Comparativa de encoders ópticos infrarrojos

<b>Parámetro</b>	<b>FC-03</b>	<b>H2010</b>
Entrada de voltaje	3-5 V	3-5 V
Consumo energético	15 mA	15 mA
Tipo de salida	Analógica y Digital	Digital
Ranura de lectura	5 mm	10 mm
Dimensiones	32x14x7 mm	27x20x17mm
Peso	8 g	2.5 g
Precio	\$2.8	\$2.5

**Realizado por:** Cevallos, Daniel, 2023.

Fundamentándonos *en la Tabla 2-8* elegiremos el encoder óptico H2010 ya que cuenta con una ranura de lectura de mayor longitud a comparación del encoder FC-03 sobrepasándolo por 5 mm, permitiéndonos cubrir una mayor área del objeto que será sentido al atravesar por el sensor tipo herradura con el que cuenta el encoder seleccionado. Por sus dimensiones y peso lo hacen perfecto para ser integrado a nuestro sistema electrónico.



## 2.7. Elementos de actuación

En la composición interna de un entrenador de planta de control los actuadores son los dispositivos encargados de convertir una señal de control en una acción física o movimiento. A continuación, mencionaremos los elementos de actuación presentes en las plantas de control:

### 2.7.1. Motor DC

El motor de corriente continua, también conocido como motor DC, es un tipo de dispositivo electromotor utilizado principalmente para convertir energía eléctrica en energía mecánica. La mayoría de las variantes de motores DC se basan en fuerzas magnéticas y cuentan con componentes internos electrónicos o electromecánicos. Un rasgo distintivo de los motores de corriente continua convencionales es la presencia de un conmutador, que alterna periódicamente la dirección del flujo de corriente dentro del motor. Esto permite que el motor gire en una dirección específica y es fundamental para su funcionamiento (Harmonic, 2023, p.5).

**Motor reductor 25GA370.** - Los motores de DC con engranajes se pueden definir como una extensión del motor. Estos motores con engranajes tienen un conjunto de engranajes conectados al motor ayudándole a aumentar el par y reducir la velocidad. Usando la combinación correcta de engranajes en un motorreductor, su velocidad puede reducirse o aumentarse en cualquier cifra deseable según su aplicación (Electrostore, 2019, p.9).

**Motor reductor DJAG25-370.** - Este motorreductor es perfecto para la creación de robots y proyectos que requieran un rendimiento óptimo en términos de fuerza y velocidad. Está equipado con un motor DC de alta potencia de 6V, junto con una caja de engranajes que proporciona una velocidad de 100 RPM. La caja de engranajes tiene un diámetro de 25 mm y un eje de salida con forma de D de 4 mm. Además, incorpora un decodificador en cuadratura o un encoder de efecto Hall, lo que facilita la lectura de los pulsos por cada revolución. Con todas estas características, este motorreductor es una opción ideal para aplicaciones que demanden un control preciso y un rendimiento alto (Electrostore 2019, p.5).

#### 2.7.1.1. Comparativa de motores DC

En la *Tabla 2-9* se realiza una comparativa respecto a los motores DC antes mencionados que pueden ser seleccionados para la realización de nuestro proyecto.

**Tabla 2-9:** Comparativa entre Motores DC

<b>Parámetros</b>	<b>Motor reductor 25GA370</b>	<b>Motor reductor DJAG25-370</b>
Entrada de voltaje	6-12 V	6-12 V
Consumo energético	60 mA-0.45 A	130 mA-1.2 A
Pulsos por revolución	No	341.2 PPR
Revoluciones	620 Rpm	280 Rpm
Torque	0.22 Kg-cm	10 Kg-cm
Peso	83.9 g	88 g
Encoder	No	Hall de 2 fases
Precio	\$14.99	\$29.99

**Realizado por:** Cevallos, Daniel, 2023.

En función al análisis de la *Tabla 2-9* el motor DC a seleccionar para el desarrollo físico del presente trabajo de integración curricular es el motor reductor DJAG25-370 que a pesar de su elevado costo, cuenta en su estructura con un encoder de cuadratura el cual nos proporcionara información en tiempo real sobre la posición y la velocidad del eje del motor, permitiéndonos un control más preciso sobre estas variables, ahorrándonos tiempo en la instalación de otro tipo de encoder y además facilitándonos la implementación de este motor en nuestra planta de laboratorio por su tamaño acorde a la dimensiones del case donde será alojado.

### **2.7.2. Motor a pasos (Stepper)**

Cuando se emplean motores de corriente continua, uno de los problemas principales es la falta de precisión al iniciar y detener el motor. Para abordar esta cuestión, se han desarrollado motores con excitación independiente, como el motor paso a paso. En este tipo de motor, la alimentación de las bobinas se realiza mediante una fuente de alimentación externa. De esta manera, para ajustar su velocidad, simplemente se modifica y controla la corriente que se aplica a las bobinas. Estos dispositivos electromagnéticos transforman pulsos digitales en movimiento de rotación mecánica, ya sea en incrementos o en forma continua. La cantidad de rotación está directamente relacionada con el número de pulsos generados, mientras que la velocidad de rotación se determina por la frecuencia de dichos pulsos. Estos pulsos, en última instancia, se definen por un ángulo preestablecido que se puede controlar mediante un dispositivo programable (CLR, 2016,p .4).

**Motor NEMA 17HD48002H-22B.** - El motor paso a paso NEMA 17 que estamos considerando es de tipo bipolar. Presenta un ángulo de paso de 1.8°, lo que equivale a 200 pasos por vuelta, y

cada bobina tiene una corriente de 1.5A. Además, es capaz de soportar una carga de hasta 4.58 kg/cm. Este motor es ampliamente empleado en aplicaciones como Impresoras 3D o máquinas CNC de uso general. Su diseño incluye un bastidor con una sección plana que garantiza un ajuste óptimo (SATYCSA, 2023, p.4).

**Motor NEMA 17 42HB34F08AB.** - Estos motores paso a paso son excelentes para proyectos que demanden movimiento, así como para máquinas como CNC y routers. El modelo 42HB34F08AB es un motor paso a paso bipolar de 4 hilos, que ofrece un ángulo de paso de 1.8°, proporcionando un movimiento muy preciso. Teniendo un ángulo de paso de 1.8° (equivalente a 200 pasos por revolución), cada fase de este motor requiere una corriente de 0.84A, teniendo un torque de 2.65 kg/cm (Electronilab, 2020, p.4).

### 2.7.2.1. Comparativa de motores a pasos

Debido a la amplia variedad de motores disponibles en el mercado, con distintas características, precios y marcas, hemos realizado una comparativa en la *Tabla 2-10* de los motores paso a paso mencionados anteriormente. A partir de esta comparación, seleccionaremos el motor que mejor cumpla con nuestras necesidades para su posterior uso en nuestro diseño.

**Tabla 2-10:** Comparativa de motores a pasos

Parámetros	NEMA 17HD48002H-22B	NEMA 17 42HB34F08AB
Voltaje de operación	12 V	4.83 V
Consumo energético	1.7 A	0.84 A
Angulo de paso	1.8°	1.8°
Pasos por vuelta	200	200
Torque	4 Kg-cm	2.56 Kg-cm
Resistencia de fase	1.5 ohms	5.75 ohms
Precio	\$18	\$18

**Realizado por:** Cevallos, Daniel, 2023.

Con base a la *Tabla 2-10* se escoge el motor NEMA 17HD48002H-22B por su modelo robusto, fácil conexión mediante cuatro cables y por su voltaje de operación, ya que todos los elementos actuadores que hemos escogido trabajan en un rango de 12V, lo que nos facilita su conexión a la misma fuente de voltaje que alimentara a todos los actuadores, evitándonos el uso de reguladores de voltaje adicionales que aumentarían el tamaño del circuito de conexiones.

### ***2.7.3. Led de alta potencia***

Son una clase especial de diodos emisores de luz que se caracterizan por tener una capacidad de emisión de luz mucho mayor en comparación con los Led convencionales. Estos dispositivos se utilizan en diversas aplicaciones que requieren una intensidad lumínica significativa y un alto rendimiento energético. Es importante tener en cuenta que los LED de alta potencia, al emitir una gran cantidad de luz, también pueden generar un mayor calor, por lo que es esencial proporcionar una adecuada disipación térmica para garantizar su correcto funcionamiento y durabilidad. Además, al elegir Led de alta potencia, es fundamental tener en cuenta la elección de una fuente de alimentación adecuada para asegurar un rendimiento óptimo y seguro (UNIT-ELECTRONICS, 2019, p.5).

### **2.8. Software de programación gráfica utilizado en EPC**

En la programación visual, los componentes del lenguaje de programación se presentan como bloques gráficos, lo que la hace también conocida como programación gráfica. Estos bloques tienen una apariencia y etiquetado que facilita la identificación de las tareas que pueden resolver dentro del flujo del programa. Los pictogramas se utilizan para guiar al usuario, lo que significa que no se requieren estructuras muy complejas ni un alto nivel de abstracción para programar de manera efectiva (IONOS, 2020,p .4).

**LabVIEW.** - Es un sistema de desarrollo de software y entorno de programación gráfica creado por National Instruments, utilizado ampliamente en adquisición de datos, control y automatización en entornos de laboratorios e ingeniería. Su programación visual basada en diagramas de bloques facilita su accesibilidad incluso para aquellos sin experiencia en programación. LabVIEW destaca por su capacidad para interactuar con dispositivos y equipos de medición y control, lo que lo convierte en una herramienta poderosa para monitoreo, control de procesos, pruebas automatizadas y desarrollo de sistemas embebidos. Además, cuenta con una comunidad activa de usuarios y una biblioteca de herramientas predefinidas, lo que lo hace una solución flexible y robusta para aplicaciones en ingeniería y ciencias en diversas industrias (National Instrument, 2023, p.4).



**Ilustración 2-7:** Logo de software de ingeniería LabVIEW

**Fuente:** (National Instrument, 2023).

**Toolbox de LabVIEW.** - es una colección de complementos que extienden la funcionalidad de LabVIEW al proporcionar componentes predefinidos, funciones personalizadas, paletas de herramientas adicionales, controladores de hardware específicos y otras utilidades. Estas herramientas permiten a los desarrolladores de LabVIEW expandir y optimizar su entorno de desarrollo, ahorrar tiempo y mejorar la eficiencia en el diseño de sistemas de control y adquisición de datos.

**Adquisición de datos en DAQ.** – la adquisición de datos (DAQ) es el proceso de medir un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido. Un sistema DAQ consiste de sensores, hardware de medidas DAQ y una PC con el software programable LabVIEW. La adquisición continua de datos es esencial en muchas aplicaciones, como control de procesos industriales, sistemas de monitorización en tiempo real, adquisición de datos en laboratorios de investigación y muchas otras áreas. Mediante CDC (Continuous Data Capture) nos permite capturar datos de manera continua y en tiempo real desde dispositivos de hardware, lo que es esencial en muchas aplicaciones de automatización, control y adquisición de datos. Asegurando que los datos se capturen en intervalos regulares sin pausas, lo que es fundamental para el análisis y el control de sistemas en tiempo real (National Instrument, 2023, p.6).



**Ilustración 2-8:** DAQ 6009 LabVIEW con entradas analógicas y digitales

**Fuente:** (Designand-Mechatronics, 2013, p.5)

## 2.9. Fuente de alimentación usado en EPC

Cada circuito o dispositivo electrónico requiere una fuente de energía para su funcionamiento. La función principal de la fuente de alimentación es convertir el voltaje alterno de la red eléctrica doméstica en un voltaje continuo adecuado para el equipo. La Fuente de alimentación AC/DC 12VDC 3A es perfecta para suministrar energía a diversos dispositivos como tarjetas Arduino, microcontroladores, motores DC, luces LED y otros. Su diseño compacto con carcasa de plástico, entrada de enchufe AC tipo A, un cable de 90 cm y un conector de salida tipo DC-Plug la hacen muy conveniente. Esta fuente DC de 12V/1A forma parte de una serie de fuentes de alimentación conmutadas AC/DC (Naylamp Mechatronics, 2023a).



**Ilustración 2-9:** Fuente de alimentación AC/DC 12VDC

Fuente: (Naylamp, 2023).

## CAPÍTULO III

### 3. MARCO METODOLÓGICO

El presente capítulo cuenta con los siguientes apartados:

#### 3.1. Requerimiento de la planta de laboratorio

Con base a la revisión bibliográfica presentada y revisada en el capítulo anterior se pueden definir los tipos de requerimientos por parte del diseñador que se muestran a continuación:

- La comunicación se establecerá entre la planta de laboratorio y el computador será de tipo serial con una velocidad de transmisión de datos de 9600 baudios.
- Se creará una comunicación bidireccional entre la PLCP y el computador para enviar y recibir información entre sí. Permitiendo que el computador envíe comandos a la planta de laboratorio para configurar su funcionamiento, iniciar o detener procesos, modificar parámetros, etc.
- Será necesaria una etapa de adquisición de datos, una etapa de comunicación, una etapa de control y visualización y por último una etapa de actuadores.
- La etapa de adquisición de datos tendrá que estar compuesta por un sensor de temperatura, un encoder de cuadratura y un encoder infrarrojo.
- La tarea de gestionar la etapa de comunicación estará a cargo de una tarjeta de desarrollo perteneciente a la familia ESP, encargada de adquirir y enviar datos de los procesos.
- La etapa de control y visualización será controlada mediante programas diseñados en el software de programación gráfica LabVIEW, específico para este propósito.
- La etapa de actuadores deberá estar constituida por un motor DC, un motor paso a paso y un LED de alta potencia.
- Se realizarán seis prácticas, de las cuales 4 trabajaran en control On/Off y los dos restantes trabajaran en un control PID.

- Cada uno de los procesos trabajarán de manera independiente para evitar fallos en su funcionamiento.
- La PLCP deberá estar equipada con una alerta acústica que notificará al usuario sobre la conexión o desconexión de la comunicación serial con el computador.
- Esta deberá ser de fácil uso, para que el usuario no tenga problemas al usarla y al ejecutar cada uno de los programas.
- La placa electrónica tendrá que estar aislada mediante el uso de un CASE, de tal manera que no pueda tener contacto con el usuario y así evitar cualquier desperfecto eléctrico, pero si deberá permitir la visibilidad de todos sus elementos.
- Deberá contar con un manual de usuario y manual de prácticas para que cada uno de los procesos se realicen de la mejor manera.
- Tendrá que ser de menor costo en comparación a equipos comerciales de similares características.

### **3.2. Concepción de la arquitectura general del prototipo**

La *Ilustración 3.1* representa la propuesta para el desarrollo de la planta de laboratorio, el cual se basa en el desarrollo cuatro etapas fundamentales las cuales son: adquisición de datos, comunicación, control y visualización y además la etapa de actuadores.

#### ***3.2.1. Etapa de adquisición de datos***

Está conformado por todos aquellos dispositivos de sensado ubicados en cada uno de los actuadores. Los cuales en conjunto tienen tres elementos de adquisición de datos, el primero es un sensor de temperatura ubicado en el disipador de calor de la luz led el cual tiene un rango de medición que parte de la temperatura ambiente hasta un valor máximo de 50° C, el segundo es un encoder infrarrojo posicionado a un lado de la rueda graduada, este elemento se encarga de indicar la posición inicial de donde parte el movimiento giratorio del motor a pasos, el tercer elemento también es un encoder pero en este caso de cuadratura el cual nos permite saber la dirección y la velocidad en la que este girando el motor DC.

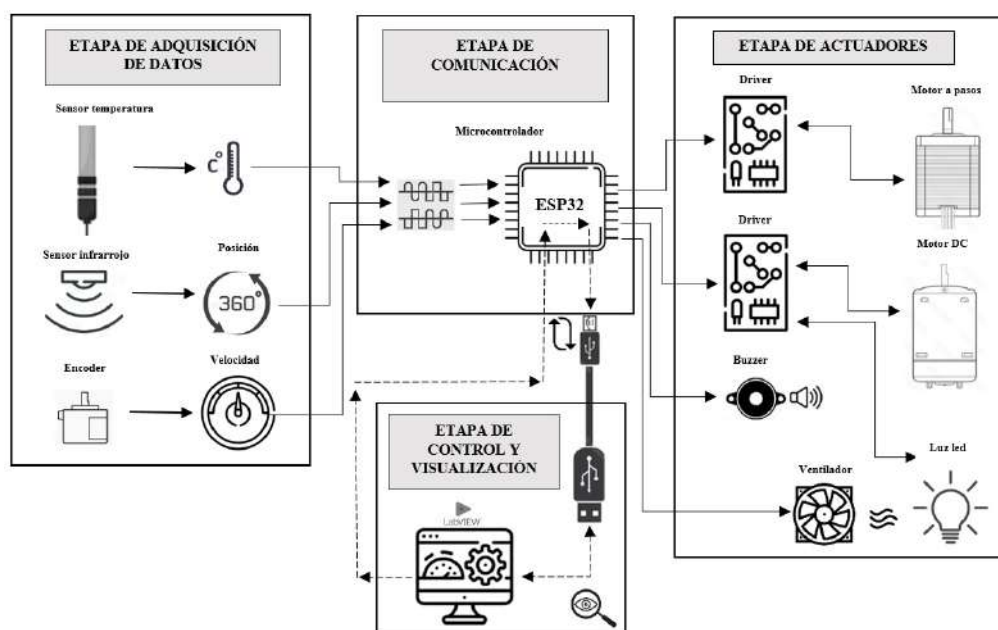


### 3.2.2. Etapa de comunicación

Se encarga de establecer la comunicación bidireccional con el software LabVIEW, en este se elige la práctica realizar y la tarjeta de desarrollo que en nuestro caso será el ESP32 define que entradas y que salidas dispondrá para tal proceso o para tal práctica, cabe mencionar que se hace uso de los 2 procesadores disponibles en la tarjeta de desarrollo, el primer procesador se encarga de la comunicación con LabVIEW y el segundo procesador se encarga de la medición del sensor de temperatura y el encoder de cuadratura esto para evitar el retardo en el procesamiento principal o en el uso del primer procesador debido a que los sensores digitales requieren una conversión de datos.

### 3.2.3. Etapa de control y visualización

Corresponde a la interfaz del software de programación gráfica, quien actúa como herramienta de visualización y control, pues por medio de ella se observa y se gestiona el proceso que se está ejecutando, observando la interacción entre el software y hardware que para este caso es la planta de laboratorio de control de procesos, permitiendo al usuario visualizar gráficos y además observar los resultados obtenidos en cada proceso, también para cada practica se desarrolla un programa específico el cual realiza la comunicación serial con la placa de desarrollo para adquirir la información de los sensores y actuadores correspondientes.



**Ilustración 3-1:** Arquitectura general de la planta de laboratorio para el control de procesos

Realizado por: Cevallos, Daniel, 2023.

### ***3.2.4. Etapa de actuadores***

La etapa de actuadores tiene como tarea fundamental transformar señales de control en movimientos físicos u otras acciones mecánicas. El proceso básico de esta etapa implica los siguientes pasos: primero, se envían señales de control al actuador desde el sistema, posteriormente, el actuador convierte la energía eléctrica en energía mecánica o energía térmica, generando movimiento o produciendo energía en forma de calor. Por último, en las prácticas que requieren precisión, como lo son el control PID de velocidad y de temperatura los actuadores proporcionar la información de retroalimentación sobre su estado actual para ser ajustados en tiempo real según sus valores de Setpoint.

### **3.3. Diseño de la arquitectura de las etapas de la PLCP**

A continuación, se presenta una descripción de los diagramas de bloques correspondientes a las diferentes etapas de la planta de laboratorio de control de procesos.

#### ***3.3.1. Etapa de adquisición y comunicación de datos***

La *Ilustración 3-2* representa la estructura de la misma, que está conformada por tres partes fundamentales: la primera es la etapa de recepción de datos establecida por cada uno de los sensores según el caso, el segundo es el bloque de adquisición de datos, quien recoge dichos datos, los procesa y a través de la tercera etapa (bloque de comunicación) los envía a la siguiente etapa que es la encargada del control y visualización de toda esta información que por medio de la interfaz gráfica desarrollada en LabVIEW se crea una comunicación bidireccional entre la planta de laboratorio y los sistemas de control implementados en el programa permitiéndonos enviar y recibir datos de temperatura, velocidad y posición.

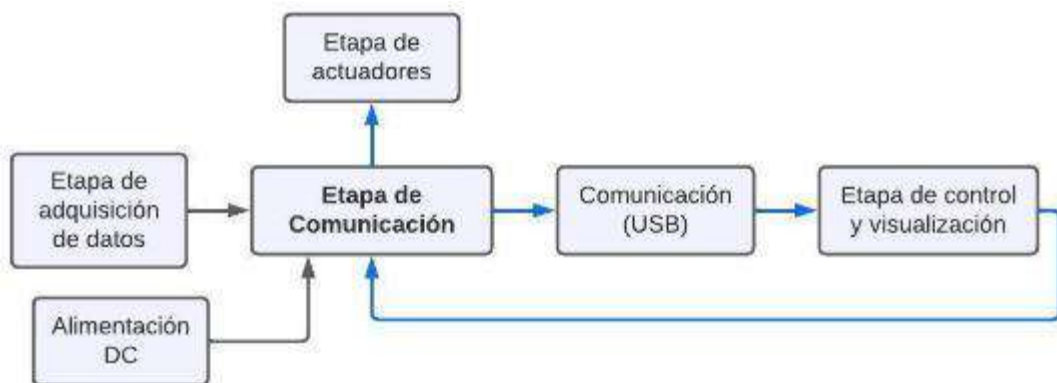


**Ilustración 3-2:** Diagrama de bloques de la etapa de adquisición y comunicación

Realizado por: Cevallos, Daniel, 2023.

### 3.3.2. Etapa de comunicación, control y actuación

La *Ilustración 3-3* representa la estructura por bloques de las etapas de comunicación, control y actuación, conformada por un bloque de comunicación, a través del cual interactúa la información enviada por los sensores desde el bloque de adquisición de datos con el bloque de comunicación, para luego ser enviada hacia la etapa de control y visualización donde se determina qué tarea realizar en base a los datos que se han obtenido y posteriormente activar los actuadores según sea caso puede ser el motor DC, motor a pasos o la luz LED respectivamente.



**Ilustración 3-3:** Etapa de comunicación, control y visualización

Realizado por: Cevallos, Daniel, 2023.

## 3.4. Selección del hardware para el PLCP

En este literal se presenta detalladamente los elementos utilizados para la implementación de una planta de laboratorio para el control de procesos, donde se realiza una rápida descripción de sus características y especificaciones técnicas de funcionamiento más importantes.

### 3.4.1. ESP32 38 Pines ESP WROOM 32

La placa de desarrollo ESP WROOM 32 es una herramienta poderosa para el prototipado rápido de proyectos IoT. Incluye el SoM ESP-WROOM-32 basado en el SoC ESP32, convertor USB-serial CP2102 para programación USB, reguladores de voltaje y leds indicadores. El ESP32 mejora al ESP8266 en comunicación y procesamiento, con Wifi, Bluetooth y BLE. Su CPU de dos núcleos a 240Mhz es independiente, y tiene periféricos internos como sensores táctiles, efecto Hall, amplificadores, tarjeta SD, Ethernet, SPI, UART, I2S e I2C. Se aplica en servidores web, procesamiento digital, webcams, robótica, domótica, entre otros (Naylamp, 2021, p.5).



**Ilustración 3-4:** ESP32

Realizado por: Cevallos, Daniel, 2023

En la Tabla 3-1 se describe las especificaciones técnicas de funcionamiento principales del módulo ESP32. Para mayor información véase al Anexo A.

**Tabla 3-1:** Especificaciones técnicas principales de ESP32

Especificación técnica	Valor
Voltaje de alimentación	5VCC
Voltaje de Entradas/Salidas	3.3VCC
Consumo energético	260 mA
Frecuencia de Reloj	240MHz
Pines Digitales GPIO	24
Pines PWM	12
Pines Analógicos ADC	18(3.3VCC)
Conectividad	Wifi/Bluetooth

Dimensiones	55x28mm
Peso	6.8g

Realizado por: Cevallos, Daniel, 2023

### 3.4.2. Fuente conmutada MP1584

Es un módulo buck regulable de tamaño reducido. El módulo tiene un regulador de conmutación reductor de alta frecuencia integrado con un MOSFET de potencia de alto voltaje interno de alto lado. El módulo puede proporcionar una corriente máxima de 3 Amperios y tiene un amplio rango de tensión de entrada que se adapta a diversas aplicaciones de reducción (Components101, 2023).



**Ilustración 3-5:** Fuente conmutada MP1584

Realizado por: Cevallos, Daniel, 2023

En la *Tabla 3-2* se describe las especificaciones técnicas de funcionamiento principales de la fuente conmutada MP1584. Para mayor información véase al Anexo B.

**Tabla 3-2:** Fuente conmutada MP1584

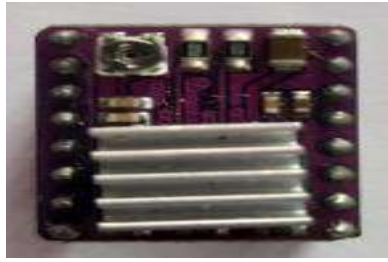
Especificación técnica	Valor
Voltaje de entrada	4.5V-28VCC
Voltaje de salida	0.8V-20VCC
Consumo energético	0.125 mA
Máxima corriente de salida	3 A
Frecuencia de conmutación	150kHz-1.5MHz

Realizado por: Cevallos, Daniel, 2023.

### 3.4.3. Driver DRV8825

El controlador DRV8825 posibilita la regulación de motores bipolares paso a paso con una capacidad de hasta 2.5A. Fundamentado en el componente DRV88225 de Texas Instruments, representa un avance en relación al chip A4988. Es frecuentemente empleado en conjunto con placas de control de impresoras 3D y máquinas CNC de código abierto como RAMPS y CNC

Shield. El DRV8825 se adapta a los pines del controlador A4988, permitiendo su utilización como una alternativa directa que proporciona un mejor desempeño (Naylamp Mechatronics, 2021).



**Ilustración 3-6:** Driver DRV8825

**Realizado por:** Cevallos, Daniel, 2023

En la *Tabla 3-3* se describe las especificaciones técnicas de funcionamiento principales del driver DRV8825. Para mayor información véase al Anexo C.

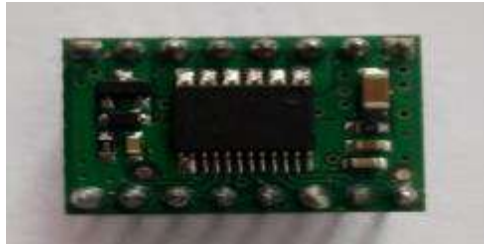
**Tabla 3-3:** Especificaciones técnicas del Driver DRV8825

<b>Especificación técnica</b>	<b>Valor</b>
Voltaje de alimentación-potencia(VMOT)	8.2V-45VCC
Voltaje de control lógico	3.3V-5VCC
Consumo energético	1 mA
Máxima corriente de salida	1.5A por bobina
Frecuencia de conmutación	150kHz-1.5MHz
Resoluciones por paso	full-step, half-step, 1/4, 1/8, 1/16 y 1/32

**Realizado por:** Cevallos, Daniel, 2023

#### **3.4.4. Driver TB6612FNG**

El controlador de motores DC TB6612FNG incorpora dos puentes H y tiene la capacidad para operar con hasta dos motores de corriente continua, sosteniendo una corriente constante de 1.2A (con la capacidad de alcanzar picos de hasta 3.2A). Se pueden emplear dos señales de entrada (denominadas IN1 e IN2) para regular el funcionamiento del motor en cuatro modalidades distintas: movimiento en el sentido de las agujas del reloj, movimiento en dirección contraria, frenado rápido y detención completa. Cada uno de los dos motores puede ser controlado de manera independiente, ya que las salidas A y B pueden ser ajustadas por separado. La velocidad de giro de cada motor se puede modificar utilizando una señal de modulación por ancho de pulso (PWM) con una frecuencia máxima de 100 kHz. Al colocar el pin STBY en estado alto, se activa el modo de espera del motor (Naylamp Mechatronics, 2021).



**Ilustración 3-7:** Driver para motor TB6612FNG

**Realizado por:** Cevallos, Daniel, 2023

En la *Tabla 3-4* se describe las especificaciones técnicas de funcionamiento principales del driver TB6612FNG. Para mayor información véase al Anexo D.

**Tabla 3-4:** Especificaciones técnicas del Driver TB6612FNG

<b>Especificación técnica</b>	<b>Valor</b>
Voltaje de potencia (VMOT)	5VCC - 15VCC
Voltaje de operación	2.7VCC - 5.5VCC
Consumo energético	2.2 mA
Máxima corriente de salida	1.2A (picos de hasta 3A)
Potencia máxima de disipación:	1W
Número de canales	2 Motores

**Realizado por:** Cevallos, Daniel, 2023

#### **3.4.5. Relé de estado sólido G3MB-202P**

El módulo de interruptor electrónico de 2 canales G3MB-202P, conocido como relé de estado sólido, desempeña la función de controlar el flujo de corriente eléctrica al recibir una pequeña corriente en sus terminales de activación. A diferencia de los relés electromecánicos, este dispositivo opera sin componentes móviles. Su principal utilidad radica en la conmutación ágil y silenciosa de cargas de corriente alterna. Asimismo, puede reemplazar de manera efectiva a los relés electromecánicos convencionales. El funcionamiento de este relé se basa en una tensión de 5V, proveniente de un microcontrolador u otra fuente, que alimenta un optoacoplador. Este compuesto por un transistor (receptor) y un diodo (emisor), está físicamente aislado. Entre sus aplicaciones destacan su uso como interruptor para sensores, como salida para PLC y en sistemas de conmutación de carga (Novatronic, 2020).



**Ilustración 3-8:** Relé de estado sólido G3MB-202P

**Realizado por:** Cevallos, Daniel, 2023.

En la *Tabla 3-5* se describe las especificaciones técnicas de funcionamiento principales del relevador de estado sólido G3MB-202P. Para mayor información véase al Anexo E.

**Tabla 3-5:** Especificación técnica del relevador de estado sólido G3MB-202P

<b>Especificación técnica</b>	<b>Valor</b>
Voltaje de entrada	5VCC
Consumo energético	6.5 Ma
Corriente de entrada	160 Ma
Corriente máxima de salida a la carga	2A por canal
Voltaje de salida a la carga	110 a 240 VAC
Señal de activación	0 lógico
Dimensiones	33mm x55mmx25mm

**Realizado por:** Cevallos, Daniel, 2023

#### **3.4.6. Motorreductor JAG25-370**

Este motorreductor resulta óptimo para la creación de robots y proyectos que demanden altas prestaciones en cuanto a fuerza y velocidad. Incluye un motor de corriente continua DC de 6-12V con una potencia significativa, así como una caja de engranajes que proporciona una velocidad de 100 revoluciones por minuto (RPM). La caja de engranajes tiene un diámetro de 25 mm y el eje de salida se presenta en una configuración en forma de D con un diámetro de 4 mm. Además, se encuentra equipado con un decodificador en cuadratura o un codificador de efecto Hall, lo que simplifica la lectura de los pulsos generados por cada revolución (Electrostore, 2019, p.6).





**Ilustración 3-9:** Motorreductor JAG25-370

**Realizado por:** Cevallos, Daniel, 2023.

Para observar las características de este motor DC debemos revisar la Tabla 9-2, donde se realizó una comparativa entre motores, en la que elegimos el motor antes nombrado. Para mayor información dirigirse al anexo F.

### 3.4.7. Led de potencia de 10W

Son dispositivos eléctricos semiconductores, también conocidos como diodos, capaces de producir luz cuando una pequeña corriente eléctrica los atraviesa. La abreviatura Led proviene del término en inglés "Light emitting diodo", que se traduce al español como "Diodo emisor de luz". Los leds de alta potencia son versiones más avanzadas que incorporan diversas opciones de óptica para controlar el flujo de luz y se fabrican con una potencia superior a 1 W. Estos elementos se emplean ampliamente en diversos dispositivos electrónicos, como iluminación en acuarios, espacios exteriores y calles, así como en publicidad comercial al aire libre y como indicadores en paneles frontales. Gracias a su versatilidad, ofrecen mayor flexibilidad en el diseño y permiten la creación de efectos de color (UNIT-ELECTRONICS, 2019, p.5).

En la *Tabla 3-6* se encuentran las características más importantes de la luz Led que puede ser considerada para la realización de nuestro proyecto. Para mayor información véase al Anexo G.

**Tabla 3-6:** Datos técnicos de la luz led de 10W

Parámetros	Luz Led 10W
Entrada de voltaje	9-12 V
Consumo energético	900 Ma
Potencia	10W
Flujo luminoso	900-1000 LM
Temperatura de color	6000-6500 K
Dimensiones	20 x 29 mm

**Realizado por:** Cevallos, Daniel, 2023.

### 3.4.8. Motor NEMA 17HD48002H-22B

Este motor paso a paso NEMA 17 funciona en modo bipolar, presenta un ángulo de paso de  $1.8^\circ$  (equivalente a 200 pasos por vuelta) y cada bobina tiene una corriente de 1.5A. Además, tiene la capacidad de soportar una carga de hasta 4.58 kg/cm. Este tipo de motor es comúnmente empleado en aplicaciones como impresoras 3D y máquinas CNC de uso general. La estructura incluye una superficie plana para asegurar un ajuste más preciso (SATYCSA, 2023, p.5).

Para observar las características de este motor a paso debemos revisar la *Tabla 10-2*, donde se realizó una comparativa entre motores a pasos, en la que elegimos el motor antes mencionado. Para mayor información dirigirse al Anexo H.



**Ilustración 3-10:** Motor NEMA 17HD48002H-22B

**Realizado por:** Cevallos, Daniel, 2023.

### 3.4.9. Encoder óptico infrarrojo H2010

Es un sensor tipo optoacoplador, también conocido como interruptor de ranura o sensor de horquilla, que se utiliza para detectar la presencia o interrupción de un objeto en su campo de detección. Ha sido extensivamente aplicado en la identificación de la velocidad rotacional del motor, así como en la cuantificación de pulsos y la determinación de límites de posición, entre otras funcionalidades. Los sistemas de optorranura presentan una disposición en la cual el emisor y el receptor de luz son ubicados en un encapsulado o una placa de circuito impreso (PCB), con la finalidad de asegurar la alineación óptima. Este módulo optoacoplador tipo herradura, es ideal para usarse como encoder, o sensor de inicio y fin de carrera (Carrod, 2018, p.53).



**Ilustración 3-11:** Encoder óptico infrarrojo H2010

**Realizado por:** Cevallos, Daniel, 2023.

Para observar las características de este encoder debemos revisar la *Tabla 8-2*, donde se realizó una comparativa entre encoders ópticos, en la que elegimos el encoder antes citado debido a las destacadas características que muestra.

### **3.5. Esquema de conexión de la planta de laboratorio**

A continuación, se detallan las conexiones realizadas para las etapas de: adquisición de datos, comunicación, control y visualización y etapa de actuadores.

#### ***3.5.1. Esquema de conexión de la planta de laboratorio conformado por las etapas de adquisición de datos, comunicación, control y visualización y etapa de actuadores.***

Respecto a la toma de alimentación eléctrica de 12VCC, se conecta el conector macho de la fuente de alimentación externa en el conector hembra ubicado en la estructura, dotando de corriente eléctrica al circuito.

El modulo regulador de voltaje DC-DC MP1584 se encuentra conectado a los pines de 12VCC y GND del voltaje de entrada, entregando un voltaje de salida de +5VCC en su terminal OUT+ para alimentar los diferentes circuitos y dispositivos que trabajan en este rango.

El módulo ESP32 conecta su terminal GND con el terminal OUT- y su terminal de 5VCC con el terminal OUT+ del regulador DC-DC MP1584 respectivamente.

Los terminales de salida A1 y A2 del driver TB6612FNG irán conectados a los pines del motor DC el cual será activado por la entrada PWMA del terminal I033 del ESP32.

El encoder A y el encoder B de fase del motor DC se conectan a los terminales I035 Y IO32 del ESP32, alimentando este motor desde los terminales A2 Y A1 del driver TB6612FNG con el terminal M+ y M- del motor, energizando cada uno de los encoder con el terminal de 3.3VCC del

ESP32. Desde el módulo ESP32 conectamos su terminal de salida IO22 enable con el terminal 1 enable del driver DRV8825.

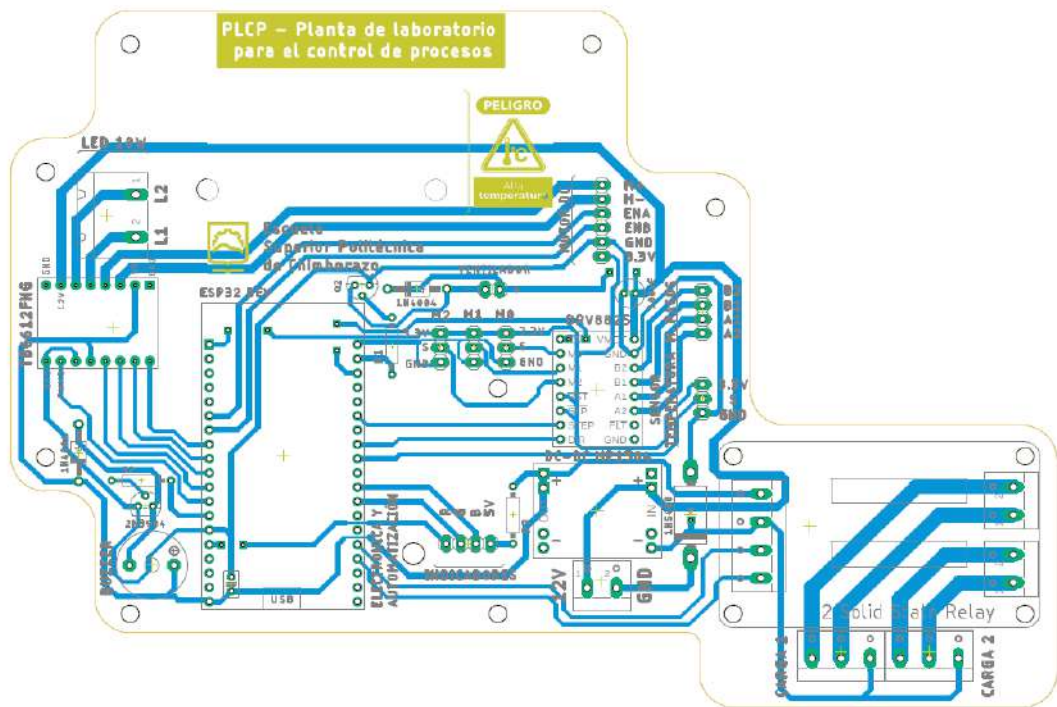
El terminal 5 y 6 pertenecientes al driver DRV8825 van conectados al pin 1 del ESP32 que les provee un voltaje de 3.3VCC.

Los terminales de salida B1 y B2 del driver TB6612FNG irán conectados a los pines de la luz led la cual será activada por la entrada PWMB del terminal IO14 del ESP32.

Para saber el número de pasos que se traslada el motor, conectamos el terminal IO21 desde el módulo ESP32 al terminal 7 del driver DRV8825 y de la misma forma para conocer la dirección o el sentido de giro del motor a pasos, conectamos el terminal IO19 con el terminal 8 del drive antes mencionado.

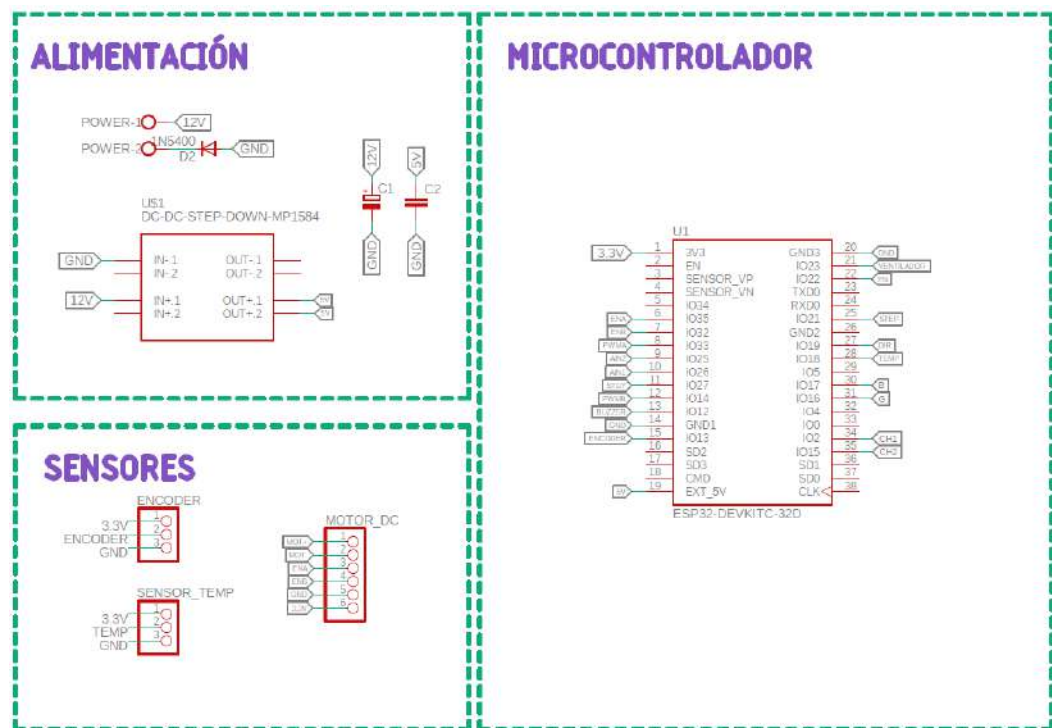
Así mismo los terminales M0, M1, M2 del driver DRV8825 encargados del control de micro stepping están unidos a espadines macho en los cuales se puede realizar la configuración de resolución de paso, que parte desde un valor de 1/2 paso hasta el valor de 1/32 para tener un valor más preciso en su movimiento.

El motor a pasos cuenta con dos bobinas que deben ser energizadas para que este pueda funcionar, donde los terminales B2, B1 y A2, A1 del driver DRV8825 son los encargados de controlar la activación del motor a pasos, alimentándolo de 12VCC por medio de su terminal VMOT y conexión a tierra por el terminal GND2.



**Ilustración 3-12:** Diseño PCB de la conexión electrónica de las etapas de adquisición de datos, comunicación, control y visualización y etapa de actuadores

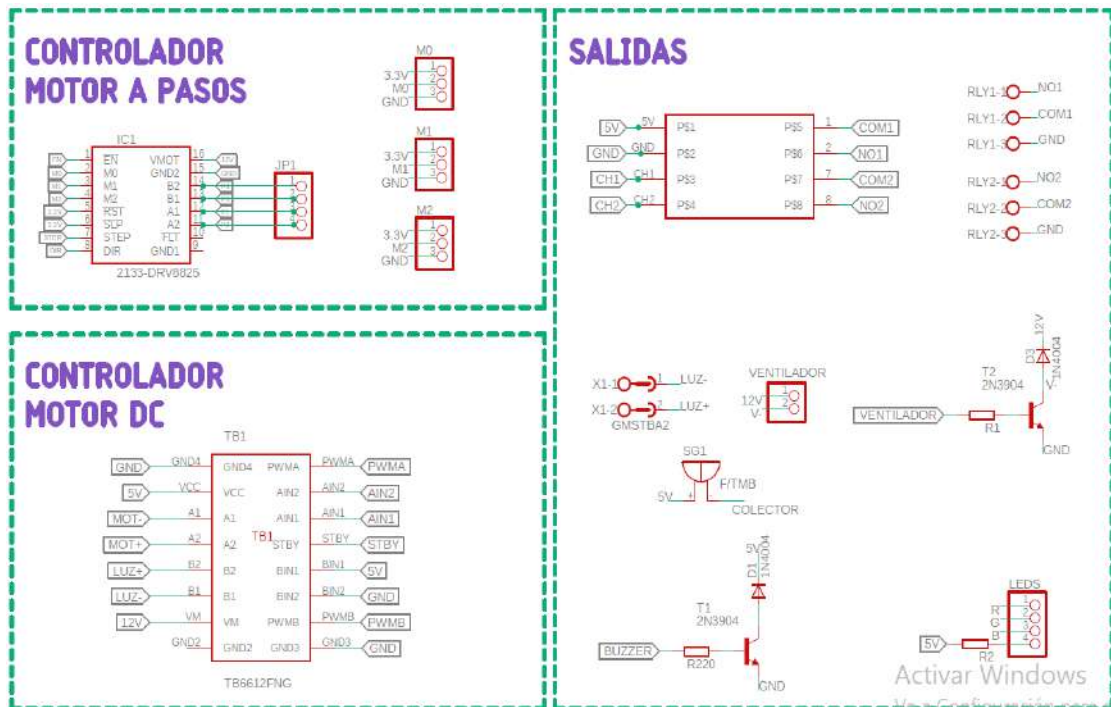
Realizado por: Cevallos, Daniel, 2023.



**Ilustración 3-13:** Esquema de conexión electrónica de la planta de laboratorio (Etapa de adquisición y comunicación)

Realizado por: Cevallos, Daniel, 2023.

Como lo muestra la *Ilustración 3-13*, se realizó el diseño de dicho circuito en PCB para la posterior construcción de la etapa de adquisición y comunicación.



**Ilustración 3-14:** Esquema de conexión electrónica de la PLCP (Etapa de activación de actuadores)

**Realizado por:** Cevallos, Daniel, 2023.

Como lo muestra la *Ilustración 3-14*, realizamos el diseño de dicho circuito en PCB para su posterior construcción de la etapa de activación de actuadores.

### 3.6. Diseño del software para la PLCP

En el siguiente apartado se presenta el software de desarrollo seleccionado para el trabajo en colectivo con los elementos hardware antes mencionados, así también se desarrolla los diferentes diagramas de flujo los cuales describen el proceso realizado dentro del entorno de Arduino IDE y además el entorno de programación gráfico desarrollado en LabVIEW donde interactúan cada una de las diferentes etapas como lo son: adquisición de datos, comunicación, control y visualización y etapa de actuadores.

#### 3.6.1. Software de desarrollo para la planta de laboratorio

Se utilizaron 3 herramientas de desarrollo de software las cuales son descritas a continuación:

**Arduino IDE.** - Es un conjunto de herramientas de programación integradas desarrollado por Arduino para sus componentes de electrónica de código abierto. Este sistema permite redactar códigos destinados a placas basadas en Arduino, eliminar programas, aplicar correcciones y reemplazar cualquier software previo. También incluye funciones de compilación que posibilitan la detección de errores. Además, características como autocompletado, formateo automático, gestión simplificada y esquematización rápida contribuyen a agilizar el proceso de desarrollo (Softonic, 2023,p .6). Para la creación del (sketch) necesario para la planta de laboratorio se utilizó la versión Arduino IDE 2.1.0.

**EAGLE.** – Se trata de una herramienta software de diseño electrónico automatizado (EDA) que facilita a los diseñadores de placas de circuito impreso (PCB) la interconexión fluida entre diagramas esquemáticos, la disposición de componentes, el trazado de rutas en las PCB y la disponibilidad de una extensa biblioteca. Para generar los variados esquemas esquemáticos y PCB requeridos para el laboratorio, se empleará la versión EAGLE 9.6.2 (Direct-Industry, 2023).

**LabVIEW.** – Es una plataforma de programación visual de alto nivel, concebida para llevar a cabo aplicaciones relacionadas con sistemas de control, captura de datos, análisis y medición. Su enfoque se basa en una interfaz gráfica en lugar de líneas de código, lo que facilita a los usuarios comprender el flujo de la programación. También nos permite abordar la programación orientada a objetos y manipular archivos XML. La versión utilizada para desarrollar todos los programas que trabajarán en conjunto con la PLCP fue LabVIEW 2021 (National-Instrument, 2023,p .5).

### ***3.6.2. Software requerido para la comunicación de la planta de laboratorio***

Para establecer una comunicación entre la planta de laboratorio y el interfaz gráfico en LabVIEW, es necesario la creación de un programa que trabaje como intermediario en la lectura, adquisición y envío de datos, tal como se menciona a continuación:

#### ***3.6.2.1. Creación de programa en Arduino IDE***

Es imprescindible la creación de un programa que nos permita mantener una comunicación bidireccional entre la planta de laboratorio y LabVIEW, donde a través de este medio podremos obtener datos mediante cada uno de los sensores que conforman la etapa de adquisición de datos y luego estos ser enviados a la siguiente etapa de control y visualización.

### 3.6.2.2. Creación de programas en LabVIEW

La interfaz gráfica de usuario (GUI) de un VI permitirá la interacción del usuario con el entorno gráfico de LabVIEW compuesto por botones, controles deslizantes, gráficos, tablas, indicadores, entre otros elementos visuales. Estos elementos permitirán la entrada de datos, la visualización de resultados y la configuración de parámetros obtenidos a través de la comunicación serial con la planta de laboratorio.

### 1.1.1 3.6.3. Programas software de la planta de laboratorio

A continuación, se presentan los diferentes programas software correspondientes a cada una de las practicas realizadas que serán mencionadas posteriormente y también se realiza una descripción para los diagramas de flujo que las componen.

#### 3.6.3.1. Programa software correspondiente al control de posición angular

La *Ilustración 3-15* representa el diagrama de flujo el cual describe el proceso del control de posicionamiento angular, el cual se lo describe de la siguiente manera:

#### **Inicialización**

- A continuación, se establecen las librerías que se emplearán en cada uno de los programas vinculados a las prácticas detalladas en esta sección.

**OneWire.h.** - Protocolo de comunicación que permite la comunicación entre dispositivos utilizando solo un cable de datos (Open Webinars, 2015).

**DallasTemperature.h.** - Simplifica la comunicación y el manejo de los sensores de temperatura, permitiendo obtener mediciones precisas de temperatura con facilidad (Zuno 2023).

**ToneESP32.h.** - Envía una frecuencia PWM al 50% del ciclo de trabajo a un pin específico para generar un tono en un zumbador piezoeléctrico con una duración opcional (Thomas, 2021, p.8).

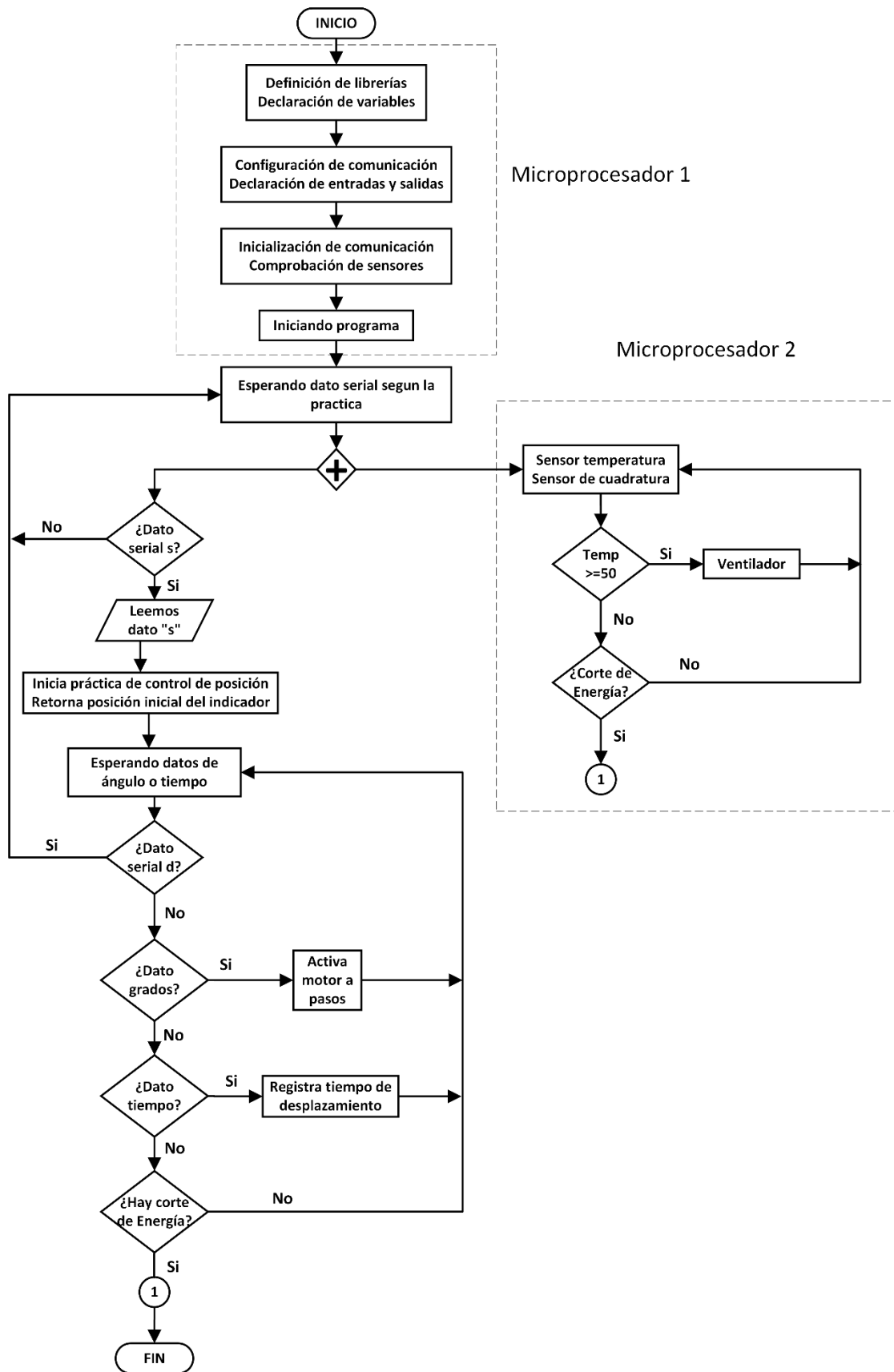
- Se declaran las variables globales que serán utilizadas
- Declaración de terminales de entrada como de salida que servirán para la adquisición y envío de datos
- Inicialización de comunicación serial
- Comprobación del estado de los sensores según el caso



- Configuración de parámetros para la comunicación serial a través de la identificación por medio de la letra distintiva de cada programa.
- Activación del buzzer, el cual nos indica que existe una comunicación entre la planta de laboratorio y el computador
- Inicialización y conexión de la planta de laboratorio con el computador por comunicación serial

### **Ciclo de repetición**

- Se pregunta si la PLCP está recibiendo un dato que indica que proceso o que practica fue seleccionada a través de su comunicación serial, de no ser así se realiza un nuevo intento de conexión, esperando hasta que nuevamente un dato sea enviado.
- Si la lectura realizada fue correcta se procede a iniciar el control de posicionamiento en su punto inicial de trabajo.
- Esperamos los datos de tiempo y ángulo enviados desde LabVIEW, estos valores son ingresados al programa de la práctica correspondiente al posicionamiento angular según la conveniencia del usuario acorde a los parámetros preestablecidos de la PLCP.
- Se pregunta nuevamente si están llegando datos, sea de ángulo o de tiempo, si el caso es positivo se activa el motor a pasos y si no continua con el proceso esperando él envió de algún dato desde la práctica de posicionamiento angular.
- Se pregunta si existe conexión de energía en ser esto verdadero el proceso continúa de forma indefinida, caso contrario el programa cierra completamente.



**Ilustración 3-15:** Diagrama de flujo del programa de control de posicionamiento angular

Realizado por: Cevallos, Daniel, 2023.

### 3.6.3.2. Programa software correspondiente al control manual de velocidad del motor DC

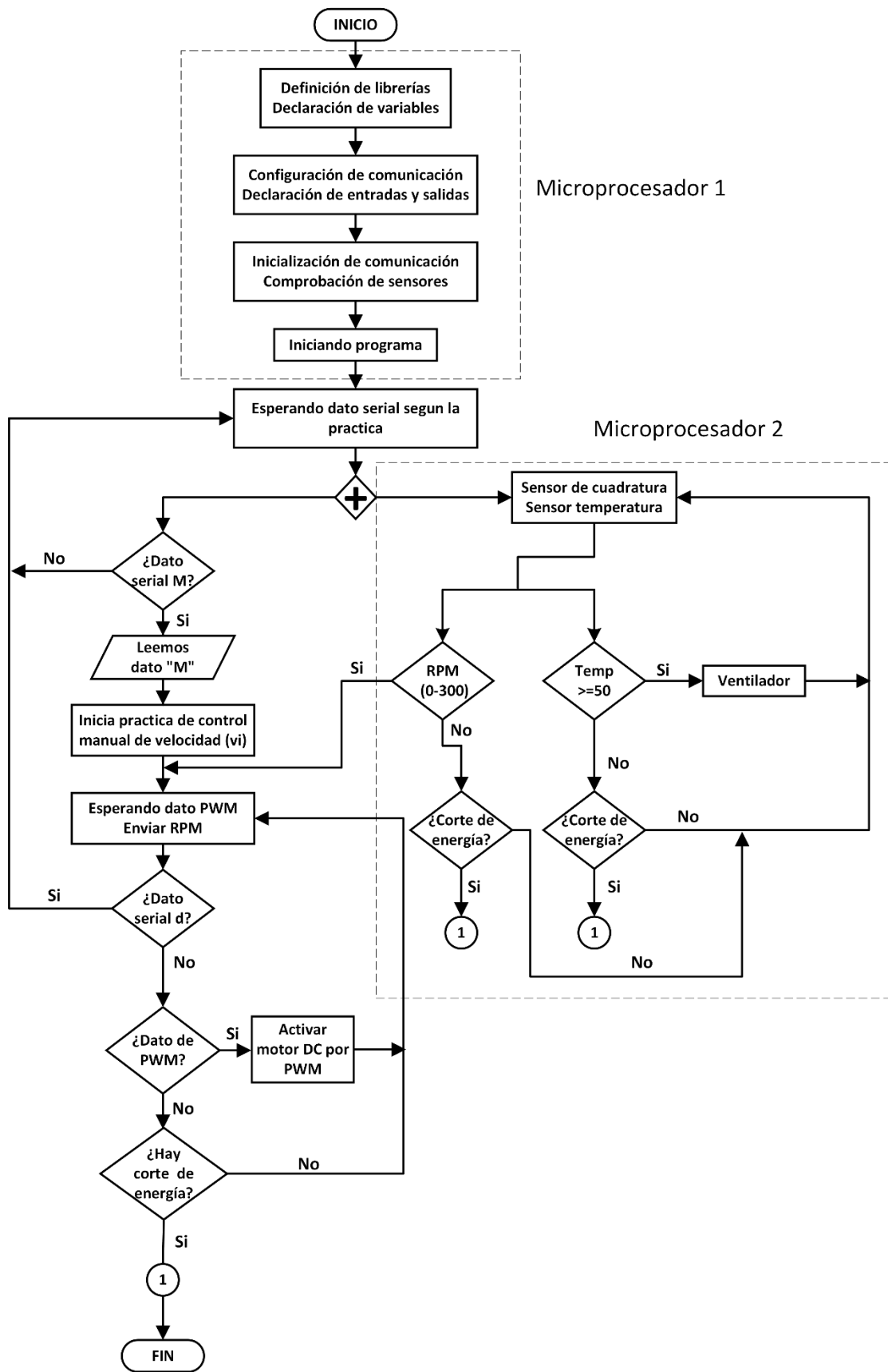
La *Ilustración 3-16* representa el diagrama de flujo el cual describe el proceso de control manual de velocidad del motor DC, el cual se lo describe de la siguiente manera:

#### **Inicialización**

- Se definen las siguientes librerías
- Se declaran las variables globales que serán utilizadas.
- Declaración de terminales de entrada como de salida que servirán para la adquisición y envío de datos
- Inicialización de comunicación serial
- Comprobación del estado de los sensores según el caso
- Configuración de parámetros para la comunicación serial a través de la identificación por medio de la letra distintiva de cada programa.
- Activación del buzzer, el cual nos indica que existe una comunicación entre la planta de laboratorio y el computador.
- Inicialización y conexión de la planta de laboratorio con el computador por comunicación serial.

#### **Ciclo de repetición**

- Se pregunta si la planta de laboratorio está recibiendo un dato que indica que proceso o que practica fue seleccionada a través de su comunicación serial, de no ser así se realiza un nuevo intento de conexión, esperando hasta que nuevamente un dato sea enviado.
- Si la lectura realizada fue correcta se procede a iniciar el control manual de velocidad de un motor DC.
- Esperamos el dato de PWM ingresado desde el programa correspondiente a la práctica anteriormente mencionada según la conveniencia del usuario acorde a los parámetros preestablecidos de la planta de laboratorio.
- Se pregunta nuevamente si están recibiendo datos, si el caso es positivo y llega el valor de PWM el motor DC se activa y si no continua con el proceso.
- Se pregunta si existe conexión de energía en ser esto verdadero el proceso continúa de forma indefinida, caso contrario el programa cierra completamente.



**Ilustración 3-16:** Diagrama de flujo del control manual de velocidad del motor DC

Realizado por: Cevallos, Daniel, 2023.

### 3.6.3.3. Programa software correspondiente al control manual de activación de SSR

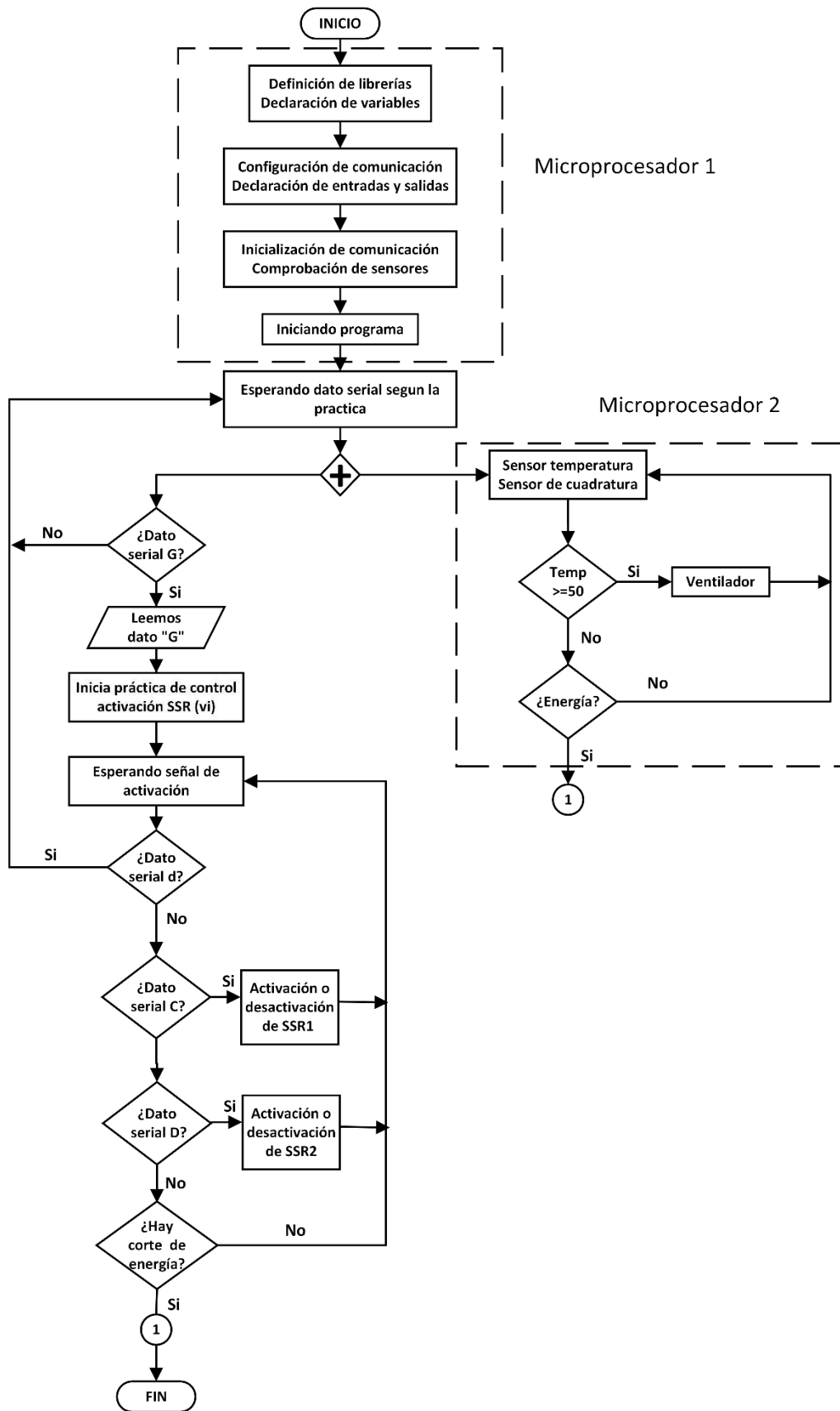
La *Ilustración 3-17* representa el diagrama de flujo el cual describe el proceso del control de activación de SSR el cual se lo describe de la siguiente manera:

#### **Inicialización**

- Se definen las siguientes librerías
- Se declaran las variables globales que serán utilizadas.
- Declaración de terminales de entrada como de salida que servirán para la adquisición y envío de datos
- Inicialización de comunicación serial
- Comprobación del estado de los sensores según el caso
- Activación del buzzer, el cual nos indica que existe una comunicación entre la planta de laboratorio y el computador
- Inicialización y conexión de la planta de laboratorio con el computador por comunicación serial.

#### **Ciclo de repetición**

- Se pregunta si la planta de laboratorio está recibiendo un dato que indica que proceso o que practica fue seleccionada a través de su comunicación serial, de no ser así se realiza un nuevo intento de conexión, esperando hasta que nuevamente un dato sea enviado.
- Si la lectura realizada fue correcta se procede a iniciar el control manual de los relés de estado sólido.
- Esperamos el dato de activación seleccionado desde el programa correspondiente a la práctica anteriormente mencionada según la conveniencia del usuario acorde a los parámetros preestablecidos de la planta de laboratorio.
- Se pregunta nuevamente si están llegando datos, si el caso es positivo y llega el valor de activación los relés se activarán y si no continua con el proceso.
- Se pregunta si existe conexión de energía en ser esto verdadero el proceso continúa de forma indefinida, caso contrario el programa cierra completamente.



**Ilustración 3-17:** Diagrama de flujo del control de activación de SSRs

Realizado por: Cevallos, Daniel, 2023.

#### 3.6.3.4. Programa software correspondiente al control de temperatura manual

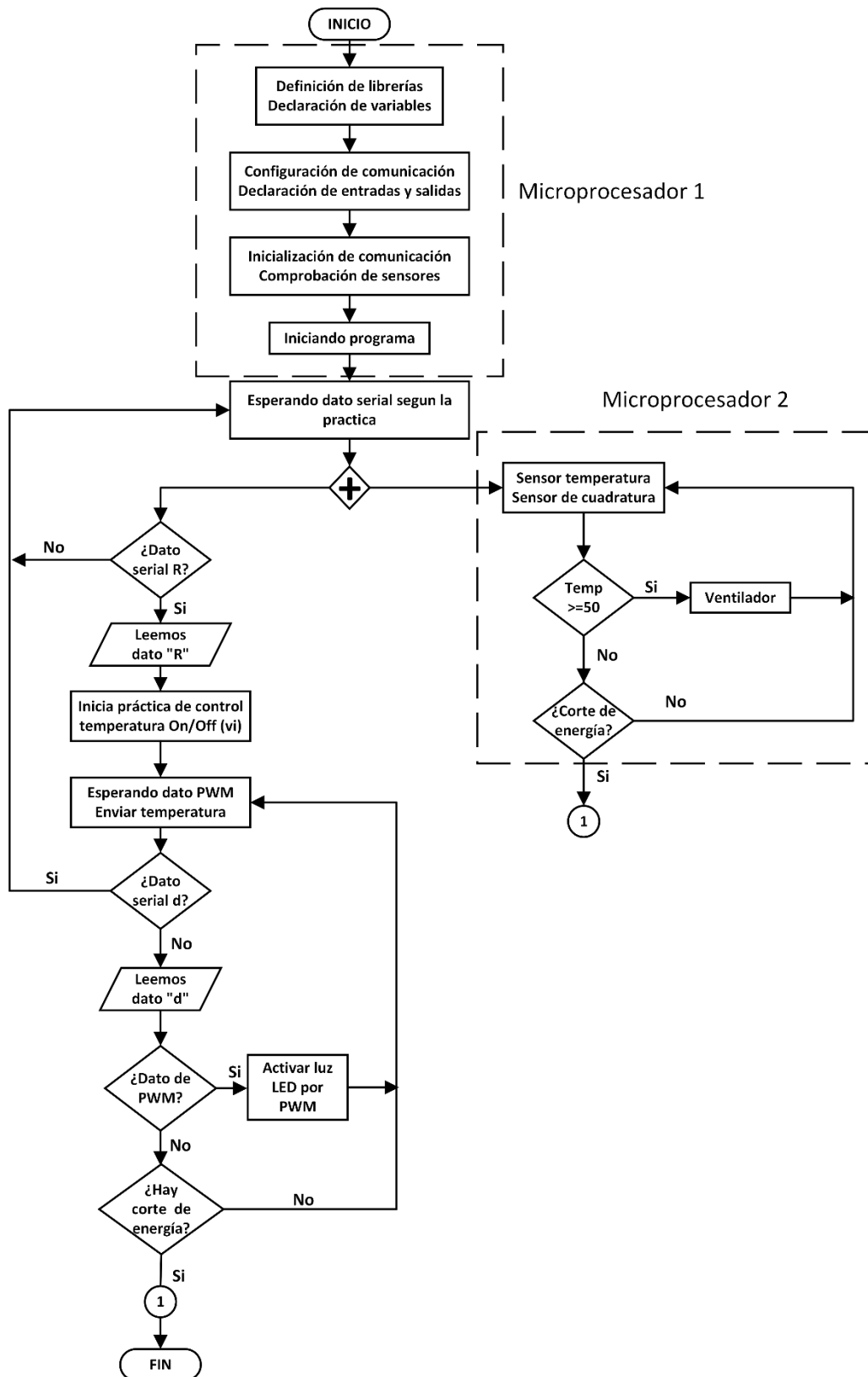
La *Ilustración 3-18* representa el diagrama de flujo el cual describe el proceso del control de temperatura On/Off, el cual se lo describe de la siguiente manera:

##### **Inicialización**

- Se definen las siguientes librerías
- Se declaran las variables globales que serán utilizadas.
- Declaración de terminales de entrada como de salida que servirán para la adquisición y envío de datos
- Inicialización de comunicación serial
- Comprobación del estado de los sensores según el caso
- Establecimiento de parámetros para la comunicación serial mediante la identificación por medio de la letra identificadora de cada programa.
- Activación del buzzer, el cual nos indica que existe una comunicación entre la planta de laboratorio y el computador.
- Inicialización y conexión de la planta de laboratorio con el computador por comunicación serial.

##### **Ciclo de repetición**

- Se pregunta si la planta de laboratorio está recibiendo un dato que indica que proceso o que practica fue seleccionada a través de su comunicación serial, de no ser así se realiza un nuevo intento de conexión, esperando hasta que nuevamente un dato sea enviado.
- Si la lectura realizada fue correcta se procede a iniciar la práctica de control de temperatura On/Off de acuerdo a la parametrización propuesto por el usuario.
- Esperamos el valor de PWM enviado desde LabVIEW, este valor es ingresado al programa correspondiente según la conveniencia del usuario acorde a los parámetros preestablecidos de la planta de laboratorio.
- Se realiza una nueva consulta para verificar la llegada de los datos relativos al PWM. En caso afirmativo, se procede a iniciar el proceso de aumento de la temperatura. Dependiendo de la situación, el ventilador se activará para reducir la temperatura si se superan los 50°C.
- Se pregunta nuevamente si existe conexión de energía, en ser esto verdadero el proceso continúa de forma indefinida, caso contrario el programa cierra completamente.



**Ilustración 3-18:** Diagrama de flujo del control de temperatura manual

Realizado por: Cevallos, Daniel, 2023.



### 3.6.3.5. Programa software correspondiente al control PID de temperatura

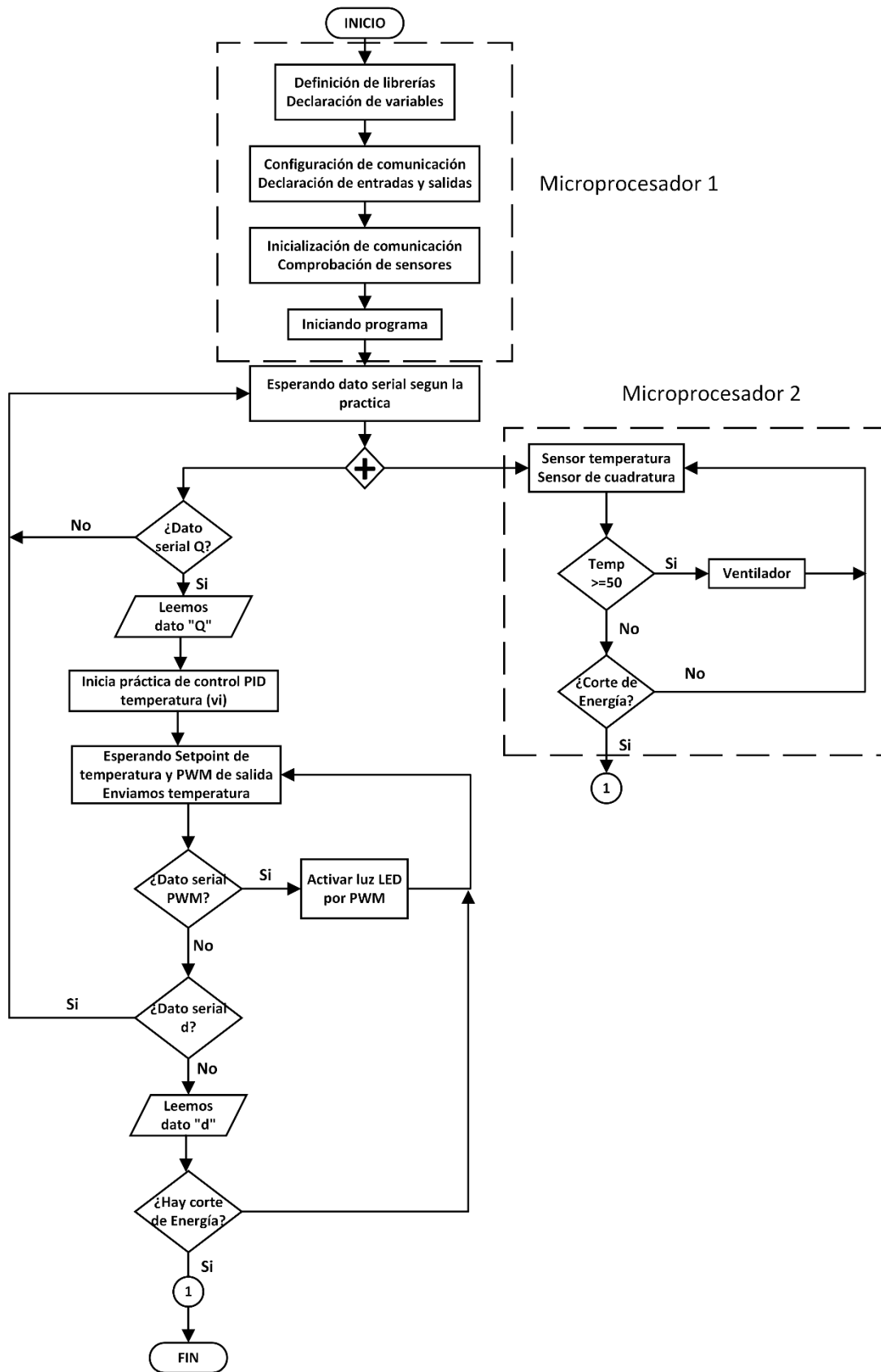
La *Ilustración 3-19* representa el diagrama de flujo el cual describe el proceso del control PID de temperatura, el cual se lo describe de la siguiente manera:

#### **Inicialización**

- Se definen las siguientes librerías
- Se declaran las variables globales que serán utilizadas.
- Declaración de terminales de entrada como de salida que servirán para la adquisición y envío de datos
- Inicialización de comunicación serial
- Comprobación del estado de los sensores según el caso
- Establecimiento de parámetros para la comunicación serial mediante la identificación por medio de la letra identificadora de cada programa.
- Activación del buzzer, el cual nos indica que existe una comunicación entre la planta de laboratorio y el computador.
- Inicialización y conexión de la planta de laboratorio con el computador por comunicación serial.

#### **Ciclo de repetición**

- Se pregunta si la planta de laboratorio está recibiendo un dato que indica que proceso o que practica fue seleccionada a través de su comunicación serial, de no ser así se realiza un nuevo intento de conexión, esperando hasta que nuevamente un dato sea enviado.
- Si la lectura realizada fue correcta se procede a iniciar la práctica de control PID de temperatura de acuerdo a la parametrización y SetPoint propuestos por el usuario.
- Esperamos el valor de Setpoint enviado desde LabVIEW, este valor es ingresado al programa correspondiente según la conveniencia del usuario acorde a los parámetros preestablecidos de la planta de laboratorio.
- Se realiza una nueva consulta para verificar la llegada de los datos relativos al PWM. En caso afirmativo, se procede a iniciar el proceso de aumento o disminución de la temperatura. Dependiendo de la situación, el ventilador se activará para reducir la temperatura; en caso contrario, se desactivará para aumentarla.
- Se pregunta nuevamente si existe conexión de energía, en ser esto verdadero el proceso continúa de forma indefinida, caso contrario el programa cierra completamente.



**Ilustración 3-19:** Diagrama de flujo del control PID de temperatura

Realizado por: Cevallos, Daniel, 2023.

### 3.6.3.6. Programa software correspondiente al control PID de velocidad del motor DC

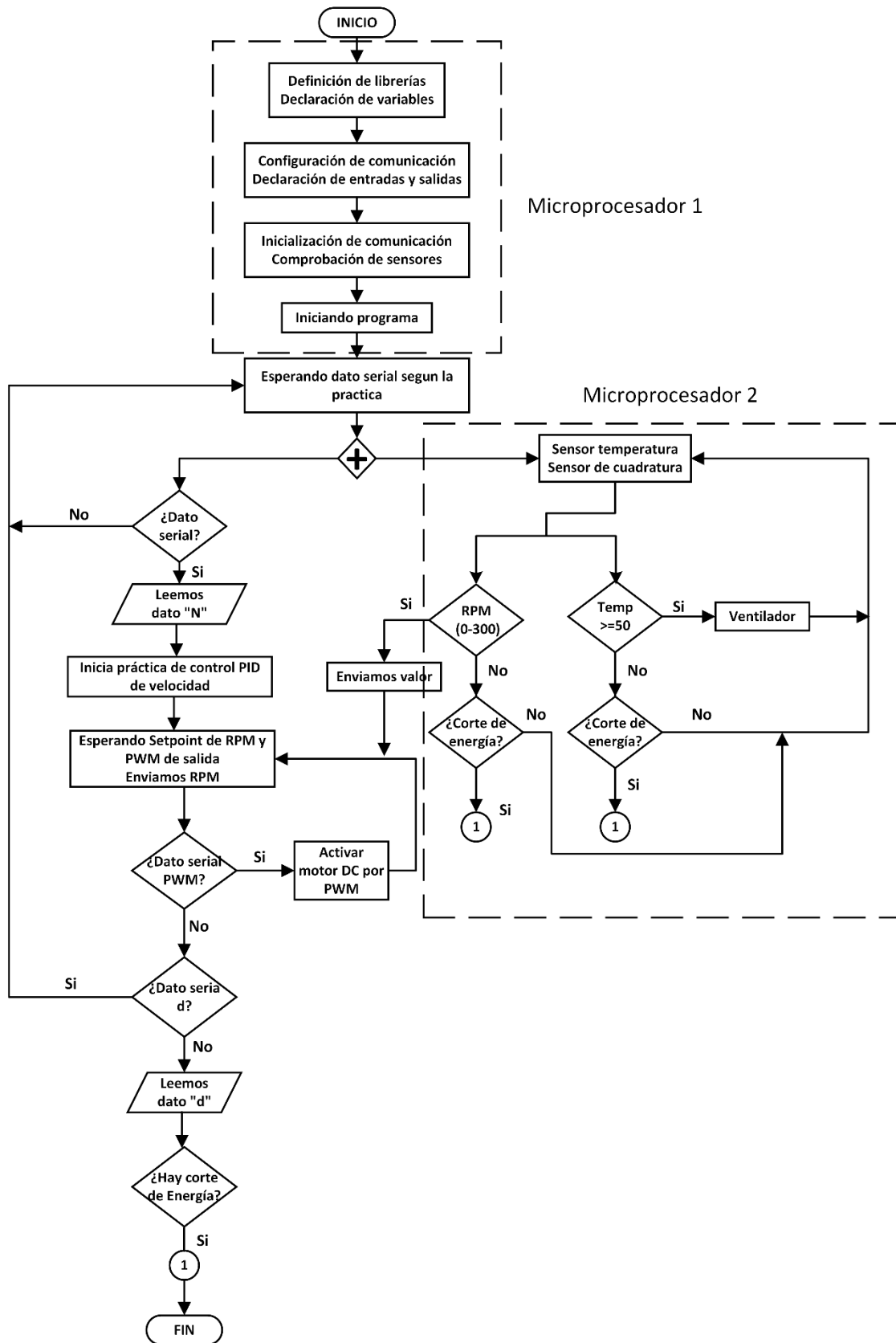
La *Ilustración 3-20* representa el diagrama de flujo el cual describe el proceso del control PID de velocidad del motor DC, el cual se lo describe de la siguiente manera:

#### **Inicialización**

- Se definen las siguientes librerías
- Se declaran las variables globales que serán utilizadas.
- Declaración de terminales de entrada como de salida que servirán para la adquisición y envío de datos
- Inicialización de comunicación serial
- Comprobación del estado de los sensores según el caso
- Establecimiento de parámetros para la comunicación serial mediante la identificación por medio de la letra identificadora de cada programa.
- Activación del buzzer, el cual nos indica que existe una comunicación entre la planta de laboratorio y el computador.
- Inicialización y conexión de la planta de laboratorio con el computador por comunicación serial.

#### **Ciclo de repetición**

- Se pregunta si la planta de laboratorio está recibiendo un dato que indica que proceso o que practica fue seleccionada a través de su comunicación serial, de no ser así se realiza un nuevo intento de conexión, esperando hasta que nuevamente un dato sea enviado.
- Si la lectura realizada fue correcta se procede a iniciar la práctica de control PID de velocidad del motor DC.
- Ingresamos el valor de Setpoint que para esta práctica es la velocidad expresada en RPM, este valor es ingresado al programa según la conveniencia del usuario acorde a los parámetros preestablecidos de la planta de laboratorio.
- Verifica si está llegando el dato de PWM de salida, si el caso es positivo el motor DC se activa, aumentando o disminuyendo su velocidad según corresponda.
- Nuevamente se pregunta si existe conexión de energía en ser esto verdadero el proceso continúa de forma indefinida, caso contrario el programa cierra completamente.



**Ilustración 3-20:** Diagrama de flujo del control PID de velocidad de un motor DC

Realizado por: Cevallos, Daniel, 2023.

Para mayor información acerca de toda la programación mostrada anteriormente se realizó en Arduino IDE 2.1.1 que se encuentra adjunto en el Anexo J.

### **3.7. Interfaz de control y visualización de la información**

Está conformado por los distintos programas relacionados con las prácticas llevadas a cabo en el entorno de programación gráfica de LabVIEW. Cada práctica, de manera individual, desempeña el papel de una herramienta para el control y la visualización de la información en los procesos térmicos, de velocidad y de posicionamiento angular. Esto se logra a través de los datos ingresados por el usuario y los datos transmitidos desde las diferentes etapas que componen la planta de laboratorio para el control de procesos (PLCP).

#### ***3.7.1. Interfaz de la práctica de control de posición angular***

En la *Ilustración 3-21*, se muestra el panel frontal correspondiente a la práctica de control de posición angular. En este panel, se ubican los botones que permiten seleccionar el puerto serial, establecer la conexión, desconectarse y detener el proceso. Para ingresar los datos, como el tiempo y el ángulo, contamos con la opción de emplear controles numéricos tipo slide o introducirlos manualmente por medio del control numérico también provisto en esta práctica, para así aumentar la precisión en la elección del valor. Junto a esto, se incluyen indicadores visuales que nos facilitan la observación de los valores ingresados y generados por el sistema, que en este contexto abarcan el ángulo, el tiempo de desplazamiento, los pulsos requeridos para la traslación y la dirección de rotación.

Para poder observar la programación en diagrama de bloque de cada una de las interfaces dirigirse al ANEXO I.



**Ilustración 3-21:** Panel frontal de la práctica de control de posición angular

Realizado por: Cevallos, Daniel, 2023.

### 3.7.2. Interfaz de control manual de velocidad del motor DC

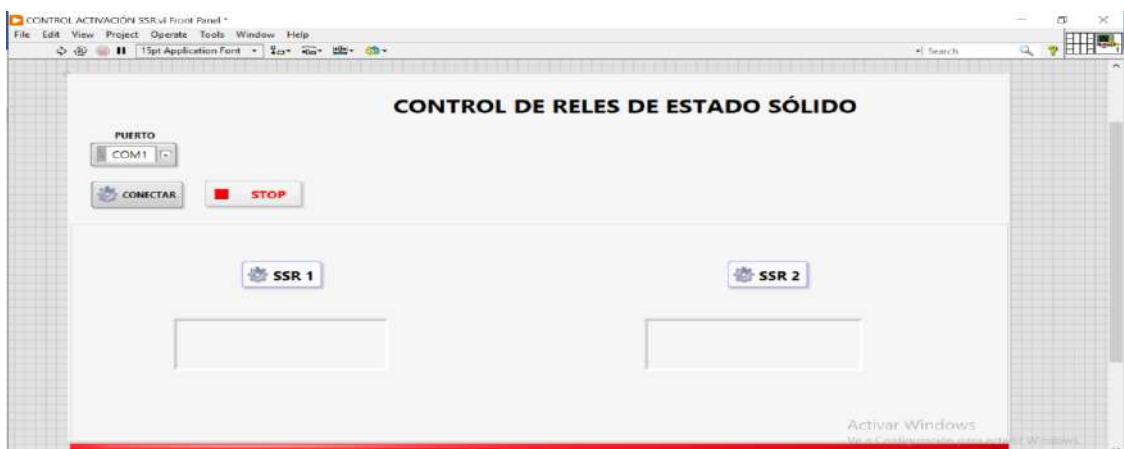
El panel frontal relacionado con la práctica de control manual de la velocidad del motor DC se observa en la *Ilustración 3-22*. En este panel, se encuentran dispuestos botones que posibilitan la elección del puerto serial, el establecimiento de la conexión, la desconexión y la detención del proceso. Para ingresar el valor de PWM, se utiliza un control numérico tipo slide que permite ajustar el nivel desde 0 hasta un máximo de 255; también incluye otro control numérico que hace posible introducir valores más exactos dentro del rango mencionado. Además, se incluye un indicador numérico y una gráfica que facilitan la visualización del valor de RPM generada por el motor debido a la variación del rango PWM.



**Ilustración 3-22:** Panel frontal de la práctica de control manual de velocidad del motor DC  
**Realizado por:** Cevallos, Daniel, 2023.

### 3.7.3. Interfaz de control de activación de SSR

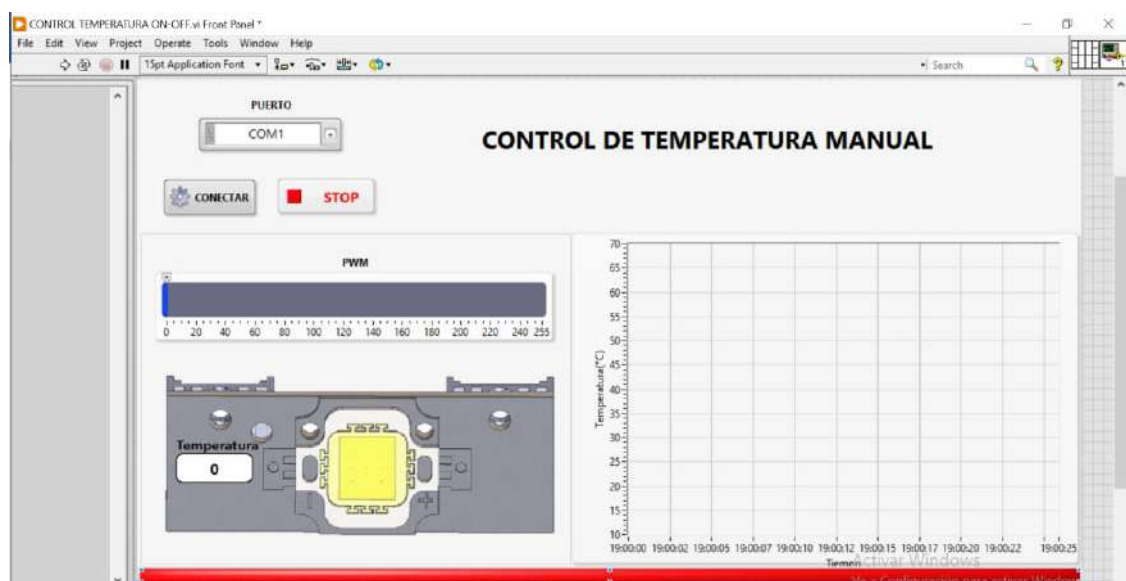
El panel frontal relacionado con la práctica de control de activación de SSR se aprecia en la *Ilustración 3-23*. En este panel, se encuentran ubicados botones que posibilitan la elección del puerto serial, el establecimiento de la conexión, la desconexión y la detención del proceso. Para seleccionar ya sea el SSR 1 o a su vez el SSR2, se utiliza un botón de tipo control booleano que permite activar o desactivar el SSR seleccionado. El cual nos indicara su estado activo por medio de un indicador tipo booleano, si la acción es positiva se torna de color amarillo caso contrario permanece en el mismo estado.



**Ilustración 3-23:** Panel frontal de la práctica de control de activación de SSR  
**Realizado por:** Cevallos, Daniel, 2023.

### 3.7.4. Interfaz de control de temperatura manual

El panel frontal relacionado con la práctica de control de temperatura On/Off se visualiza en la *Ilustración 3-24*. En este panel, se encuentran botones que posibilitan la elección del puerto serial, el establecimiento de la conexión, la desconexión y la detención del proceso. Para ingresar el valor de PWM, se utiliza un control numérico tipo slide que permite ajustar el nivel desde 0 hasta un máximo de 255; también incluye otro control numérico que hace posible introducir valores más exactos dentro del rango mencionado, hasta llegar a los 50°C. Además, se incluye un indicador numérico y una gráfica que facilitan la visualización del valor de temperatura en °C generada por la luz LED, debido a la variación del rango PWM.



**Ilustración 3-24:** Panel frontal de la práctica de control de temperatura manual

**Realizado por:** Cevallos, Daniel, 2023.

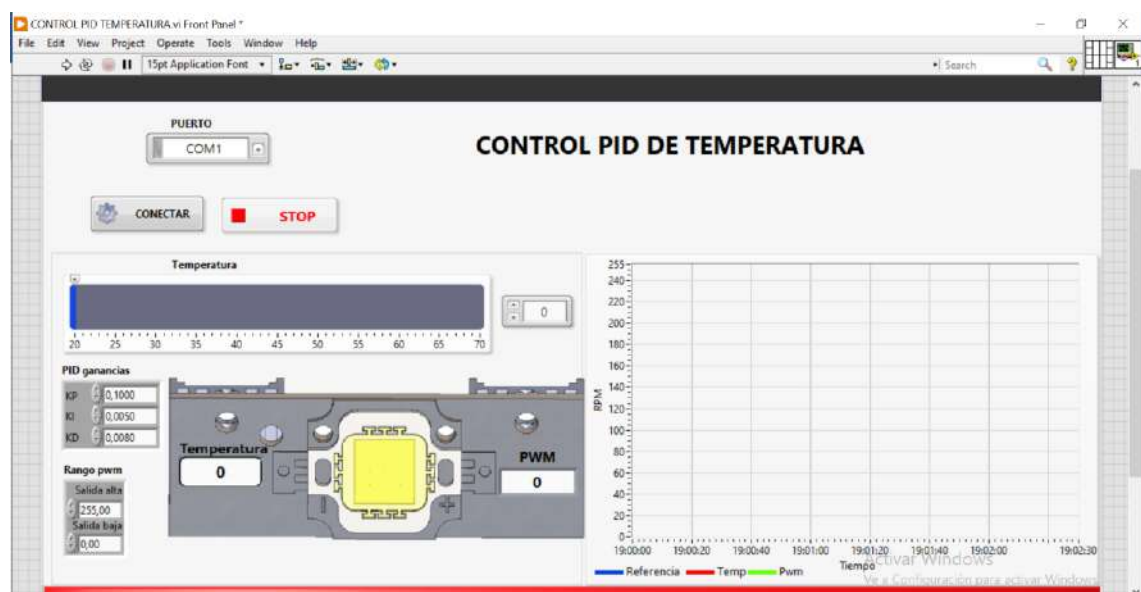
### 3.7.5. Interfaz de control PID de temperatura

En la *Ilustración 3-25* se presenta el panel frontal correspondiente a la práctica de control PID de temperatura. En este entorno gráfico se incluye un conjunto de botones de control que permiten llevar a cabo diversas acciones. Entre estas se cuentan la selección del puerto serial, el establecimiento y la interrupción de la conexión, así como la detención completa del proceso.

En esta configuración, es posible ingresar el valor de referencia (Setpoint) mediante dos tipos de controles numéricos. Uno de ellos es de tipo slider (deslizante), que posibilita ajustar el valor deseado al mover el cursor. El otro control numérico ofrece la capacidad de introducir un valor más preciso dentro del rango de temperatura, que se extiende desde 20°C hasta un límite máximo



de 50°C. Mediante otro control numérico, se tienen la posibilidad de ingresar los valores correspondientes a los parámetros del controlador PID. Estos parámetros son cruciales para adecuar el comportamiento del controlador a las características específicas del sistema bajo control. Adicionalmente, se ha incorporado un indicador numérico que refleja la evolución de la temperatura. Este valor se ajustará en función de las circunstancias, aumentando o disminuyendo hasta alcanzar el valor de referencia (Setpoint). Junto a este indicador, se incluye una gráfica que facilita la visualización de la temperatura en °C. Esta gráfica es generada por una luz LED y permite observar tanto las fluctuaciones causadas por perturbaciones externas como la estabilización resultante de la acción de control aplicada al sistema. Esto último es una consecuencia de las variaciones en el rango de modulación por ancho de pulso (PWM).



**Ilustración 3-25:** Panel frontal de la práctica de control de control PID de temperatura

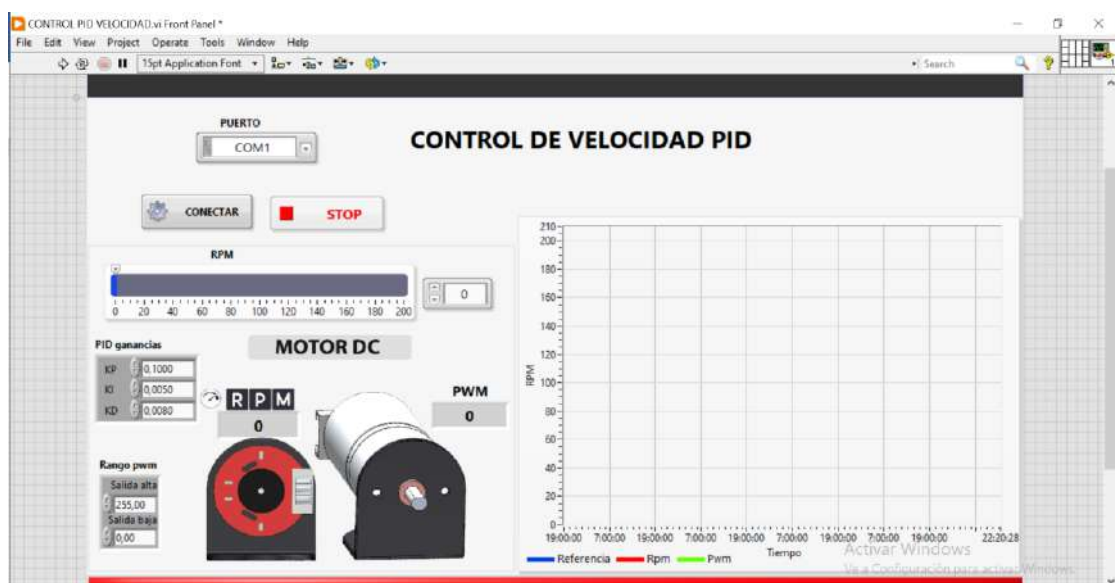
Realizado por: Cevallos, Daniel, 2023.

### 3.7.6. Interfaz de control PID de velocidad del motor DC

En la *Ilustración 3-26* se presenta el panel frontal correspondiente a la práctica de control PID de velocidad del motor DC. En este entorno gráfico se incluye un conjunto de botones de control que permiten llevar a cabo diversas acciones. Entre estas se cuentan la selección del puerto serial, el establecimiento y la interrupción de la conexión, así como la detención completa del proceso.

En esta configuración, es posible ingresar el valor de referencia (Setpoint) mediante dos tipos de controles numéricos. Uno de ellos es de tipo slider (deslizante), que posibilita ajustar el valor deseado al mover el cursor. El otro control numérico ofrece la capacidad de introducir un valor de velocidad más preciso dentro del rango que se extiende desde 0 hasta un límite máximo de 300

RPM. Mediante otro control numérico, se tienen la posibilidad de ingresar los valores correspondientes a los parámetros del controlador PID. Estos parámetros son cruciales para adecuar el comportamiento del controlador a las características específicas del sistema bajo control. Adicionalmente, se ha incorporado un indicador numérico que refleja el incremento o decremento de la velocidad. Este valor se ajustará en función de las circunstancias, aumentando o disminuyendo hasta alcanzar el valor de referencia (Setpoint). Junto a este indicador, se incluye una gráfica que facilita la visualización de la velocidad en RPM. Esta gráfica es generada por la acción del motor que permite observar tanto las variaciones causadas por perturbaciones externas como la estabilización resultante de la acción de control aplicada al sistema. Esto último es una consecuencia de las variaciones en el rango de modulación por ancho de pulso (PWM).



**Ilustración 3-26:** Panel frontal de la práctica de control PID de velocidad del motor DC

**Realizado por:** Cevallos, Daniel, 2023.

Para tener una mayor comprensión acerca del funcionamiento correcto de las practicas detallas anteriormente por favor dirigirse al Anexo I.

### 3.8. Estructura del prototipo

La estructura que elegimos para albergar nuestra placa PCB y demás elementos eléctricos y electrónicos es la caja de conexiones Heyiarbeit construida en plástico ABS a prueba de polvo con un grado de protección "IP65", es una caja diseñada para proporcionar un recinto seguro y protegido para componentes eléctricos y electrónicos en diversas aplicaciones (Amazon, 2023, p.4).



**Ilustración 3-27:** Caja de conexiones Heyiarbeit de plástico ABS a prueba de polvo impermeable IP65

**Fuente:** (Amazon,2023,p .2).

Antes de adquirir este producto, revisamos las especificaciones técnicas proporcionadas por el fabricante y nos aseguramos que cumpla con nuestras necesidades específicas en términos de tamaño, características de protección y opciones de montaje. Y para asegurar todo lo antes mencionado realizamos el diseño 3D en SolidWorks, acorde a las dimensiones y especificaciones de cada elemento, comprobando que la caja de conexiones es una opción adecuada para la instalación de nuestra placa electrónica y demás elementos.



**Ilustración 3-28:** Diseño 3D del Montaje de placa PCB dentro de la caja de conexiones Heyiarbeit

**Realizado por:** Cevallos, Daniel, 2023.

Tras soldar todos los componentes eléctricos y electrónicos en la placa PCB, se llevó a cabo la instalación cuidadosa dentro de la caja de conexiones. Además, aseguramos de manera precisa tanto el motor DC como el motor a pasos, fijando cada elemento con precisión para prevenir cualquier posible fallo en su operación. Una vez culminado este procedimiento, el ensamblaje definitivo se presenta tal como se muestra en la *Ilustración 3-29*.



**Ilustración 3-29:** Montaje final de la placa PCB y demás elementos dentro de la caja de conexiones Heyiarbeit.

**Realizado por:** Cevallos, Daniel, 2023.

## CAPÍTULO IV

### 4. MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

En el siguiente capítulo se realiza diferentes tipos de pruebas orientadas a las etapas que conforman la arquitectura general de la planta de laboratorio, tal como, adquisición de datos, comunicación, control y visualización de la información y etapa de actuadores. Se los puede observar en la *Ilustración 4-1*.



**Ilustración 4-1:** Arquitectura general de la planta de laboratorio

**Realizado por:** Cevallos, Daniel, 2023.

La determinación del tamaño de muestra es un desafío clave en investigación de muestreo. Implica consideraciones teóricas y prácticas. Desde la teoría, se debe definir el parámetro a estimar, el método de muestreo y el estimador. Desde la práctica, influyen factores como presupuesto, tiempo, objetivos, información deseada, personal y calidad de materiales. Estos aspectos determinan el tamaño de muestra necesario para una investigación efectiva (Montesinos et al., 2010,p .6).

Así también los autores plantean que la cantidad de muestras necesarias para realizar un análisis estadístico depende de varios factores, como el tipo de análisis, el nivel de confianza y precisión, la variabilidad de los datos y el tamaño del efecto que estás investigando. No hay un número fijo o universal de muestras que se aplique a todas las situaciones. Por lo antes mencionado y tomando en cuenta el factor del tiempo que disponemos para realizar el trabajo de integración curricular hemos decidimos tomar aproximadamente 30 muestras, para las validaciones por medio del cálculo estadístico del error relativo.

El error relativo es una medida que se utiliza para evaluar la precisión de una medición o cálculo en relación con el valor verdadero o de referencia de una magnitud. Para encontrar este valor se calcula dividiendo el error absoluto (la diferencia entre el valor medido y el valor verdadero) entre el valor verdadero y se expresa como un porcentaje o fracción. El error relativo proporciona información sobre la importancia relativa de la desviación en comparación con el valor verdadero, lo que puede ser útil para comprender la exactitud de una medición (Posadas, 2023, p.5).

(Santo y Lecumberry, 2005, p. 14-15) mencionan que el error relativo ( $Er$ ) es una magnitud adimensional para comparar la incertidumbre de una medición con la propia medida y evaluar la importancia de dicha incertidumbre. Esta expresión de incertidumbre facilita la comparación de la calidad de mediciones realizadas con distintos instrumentos o métodos. El error relativo porcentual ( $Er \%$ ) se emplea como indicador de calidad del resultado. Estableciendo que, si el  $Er \%$  es menor al 1%, el resultado experimental es considerado bueno. Si se encuentra entre el 5% y el 10%, se considera aceptable, mientras que un  $Er \%$  superior al 10% indica un resultado poco confiable.

Una muestra de 10 repeticiones proporciona suficiente información para calcular estadísticas descriptivas, como la desviación estándar y la variabilidad, de manera más confiable. Cuantas más muestras se tomen, mayor será la confianza en la representación de la variabilidad real del proceso. Al mismo tiempo, tomar un número razonable de muestras evita extremos anómalos y proporciona una imagen más precisa de la repetibilidad en lugar de depender de un número limitado de mediciones (Senar, 1999, p. 53-61).

Luego, se procede a calcular el coeficiente de variación ( $CV$ ), el cual es un indicador que permite establecer comparaciones entre distintos casos o poblaciones, al mismo tiempo que establece una relación entre el tamaño de la media aritmética y la variabilidad de la variable. El coeficiente de variación se expresa en porcentaje, ya que esta es la forma más efectiva de presentarlo. Sin embargo, también es posible expresarlo en valores de 0 a 1. Es importante tener en cuenta que, en ciertas distribuciones de probabilidad, este coeficiente puede alcanzar o incluso superar el valor de 1 (SDELSOL, 2022, p .5).

Para su interpretación, si el  $CV$  es próximo a 0 (0%) quiere decir que la muestra es compacta por lo que existe poca variabilidad de datos, si el  $CV$  supera a 0.3 (30%) se traduce a que su media es poco representativa, finalmente, si el valor del  $CV$  tiende a 1 (100%) significa que los datos son muy dispersos por lo que su media pierde confiabilidad (Requena, 2016, p .2).

Según (Vargas, 2007, p. 48-49) proporciona una vía para interpretar los valores posibles del coeficiente de variación (CV). Si el coeficiente de variación es inferior al 30%, esto indica que los datos de la muestra presentan una relativa homogeneidad. Si el CV se encuentra en el rango de 30% a 70%, se considera que los datos de la muestra son heterogéneos, lo que implica que la media no es representativa. Por último, si el CV supera el 70%, los datos de la muestra muestran una gran variabilidad o heterogeneidad, lo que también hace que la media carezca de representatividad

La evidencia correspondiente a las pruebas realizadas para este capítulo se encuentra adjuntas en el Anexo K.

#### **4.1. Pruebas de la etapa de adquisición de datos**

En esta sección, se llevan a cabo pruebas específicas en los componentes de sensado que componen la etapa de adquisición de datos. El propósito de esta prueba es evaluar la sensibilidad mediante el cálculo y obtención del error absoluto, así como la calidad a través del error relativo. Donde también se realiza el análisis de la confiabilidad y calidad del componente sometido a la prueba mediante el coeficiente de variación (CV), obtenido tras realizar un análisis de repetibilidad.

##### ***4.1.1. Validación de los elementos de sensado en la etapa de adquisición***

A continuación, se realizan pruebas para la validación de los elementos de sensado que componen el módulo de adquisición de datos.

**Validación del sensor de temperatura DS18B20.** - Se lo realiza mediante pruebas orientadas a la práctica de laboratorio de control de temperatura On/Off, con el objetivo de determinar el criterio de calidad de dicho sensor de temperatura al momento de realizar dicha práctica.

Para ello, se ingresa un valor de temperatura en el caso de trabajar en lazo cerrado o un valor de PWM en modo de control On/Off mediante la etapa de control y visualización para que por medio de este, la luz led se encienda permitiendo elevar la temperatura en el disipador de calor hasta llegar al valor ingresado y con ayuda de un equipo patrón, en nuestro caso un termómetro tipo pistola como puede observarse en el Anexo K, se toma un valor cualquiera de temperatura en el disipador de calor ubicado en el LED, tomamos un valor medido de 36°C que se encuentra en el rango de temperatura de (20-50°C) en el que trabajan las practicas orientadas a control de

temperatura, véase en la *Tabla 4-1*, finalmente se calcula el error absoluto y error relativo para un posterior análisis de resultados.

**Tabla 4-1:** Validación del sensor de temperatura DS18B20

N.º de muestra	Valor de $T^{\circ}$ generado por el sistema ( $^{\circ}\text{C}$ )	Valor de $T^{\circ}$ medido por el termómetro ( $^{\circ}\text{C}$ )	Error Absoluto	Error Relativo
1	42	42	0	0
2	42	41	0	2.44
3	42	42	0	0
4	42	41	1	2.44
5	42	42	0	0
6	42	42	0	0
7	42	43	-1	2.33
8	42	42	0	0
9	42	41	1	2.44
10	42	42	0	0
11	42	43	-1	2.33
12	42	41	1	2.44
13	42	42	0	0
14	42	43	-1	2.33
15	42	42	0	0
16	42	42	0	0
17	42	42	0	0
18	42	41	1	2.44
19	42	41	1	2.44
20	42	42	0	0
21	42	43	-1	2.33
22	42	41	1	2.44
23	42	41	1	2.44
24	42	39	-1	2.33
25	42	41	1	2.44
26	42	42	0	0
27	42	42	0	0
28	42	42	0	0
29	42	43	-1	2.33
30	42	43	-1	2.33

**Realizado por:** Cevallos, Daniel, 2023.

En la *Tabla 4-1* se observa como resultado de la validación del sensor del sensor DS18B20 en cuanto a su temperatura de medición el Er máximo encontrado es de 2.44%, por lo que, según



definición de autores, al encontrarse el valor en el rango de  $5\% < Er < 10\%$ , significa que se obtuvo un resultado experimental con criterio de calidad aceptable.

## **4.2. Pruebas de la etapa de comunicación**

En el siguiente apartado se realiza pruebas de repetibilidad en la comunicación de la PLCP con la etapa de adquisición de datos, de comunicación y etapa de control y visualización.

### ***4.2.1. Repetibilidad en la comunicación entre la etapa de adquisición, comunicación y control y visualización***

Para cuantificar las muestras registradas en las pruebas correspondientes a esta sección, una correcta comunicación entre etapas adquisición, comunicación y control y visualización tras la activación de los sensores en un primer intento se la considera como “1”, caso contrario se la representa con un “0”. Con esto se plantea:

**Comunicación entre la etapa de adquisición, de comunicación y etapa de control y visualización.** - El objetivo de esta prueba es determinar la confiabilidad y calidad de la comunicación entre etapas mediante el cálculo del coeficiente de variación (CV) tras realizar un análisis de repetibilidad.

Para que se establezca una comunicación entre las etapas antes mencionadas la comunicación entre el ESP y la etapa de control y visualización (LabVIEW) deben mantener una comunicación serial de 9600 baudios de velocidad para la transmisión de datos, ya que este valor fue asignado y declarado como requerimiento en cada una de las practicas a desarrollarse, es por eso que este valor no puede cambiar o modificarse. Desde el interfaz de la etapa de control y visualización seleccionamos el puerto serial del computador con el que van a trabajar conjuntamente con la planta de laboratorio, si la comunicación serial es exitosa, una alarma auditiva nos indicara que si existe conexión serial y el programa se ejecuta de manera correcta partiendo de valores iniciales en cada una de las prácticas y si el caso no es favorable en la pantalla nos aparecerá un mensaje indicándonos que existe un error en comunicación serial. Para que todo el proceso antes mencionado funcione correctamente, la PLCP debe mantenerse conectada al computador a través de un cable micro USB de un metro de longitud para este caso, caso contrario no existirá conexión alguna.

La prueba de repetibilidad comunicación entre la etapa de adquisición, de comunicación y etapa de control y visualización se observa en la *Tabla 4-2*.

**Tabla 4-2:** Comunicación entre la etapa de adquisición, de comunicación y módulo de control y visualización

N.º de muestra	Practica a realizarse	Parámetros			
		Velocidad de transmisión	Distancia (m)	Intento de conexión	Respuesta del Sistema
1	Control de velocidad On/Off	9600 Bd	1	1	1
2	Control de temperatura On/Off	9600 Bd	1	1	1
3	Control de Posición	9600 Bd	1	1	1
4	Control de SSR	9600 Bd	1	1	1
5	Control PID de temperatura	9600 Bd	1	1	1
6	Control PID de velocidad	9600 Bd	1	1	1
7	Control de velocidad On/Off	9600 Bd	1	1	1
8	Control de temperatura On/Off	9600 Bd	1	1	1
9	Control de Posición	9600 Bd	1	1	1
10	Control de SSR	9600 Bd	1	1	1
	Media				1
	Desviación estándar ( $\sigma$ )				0
	Coefficiente de variación estándar (CV)				0%

Realizado por: Cevallos, Daniel, 2023.

De la *Tabla 4-2* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variabilidad de datos, teniendo así valores estables. Producto a ello se dice que la comunicación entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización es confiable y de calidad.

#### ***4.2.2. Pruebas de funcionamiento de las prácticas a realizarse con la planta de laboratorio de control de procesos***

Para cuantificar las muestras registradas en las pruebas correspondientes a esta sección, una correcta comunicación entre etapas adquisición, comunicación y control y visualización tras la activación de los sensores en un primer intento se la considera como “1”, caso contrario se la representa con un “0”. Con esto se plantea:

**Repetibilidad del funcionamiento de las etapas de adquisición, comunicación y control y visualización y etapa de actuadores en cada una de las prácticas.** - El objetivo de esta prueba

es determinar la confiabilidad y calidad de la comunicación entre etapas mediante el cálculo del coeficiente de variación (CV) tras realizar un análisis de repetibilidad.

Una vez establecida la comunicación serial a 9600 baudios entre las de etapas de adquisición, de comunicación, control y visualización, ejecutaremos cada una de las practicas, activando cada uno de los actuadores según corresponda. Si se ejecuta alguna práctica de control de velocidad se activará el motor DC, en el caso de temperatura se pondrá en funcionamiento la luz LED, cuando se activa el motor a pasos podemos observar su desplazamiento en función al tiempo y el valor de grados sexagesimales y en el caso de control de SSR comprobaremos su correcto funcionamiento al activar una carga AC.

#### 4.2.2.1. Prueba de funcionamiento de la práctica de control de posición angular

El objetivo de esta prueba es determinar la confiabilidad del funcionamiento de la práctica antes nombrada mediante el cálculo del coeficiente de variación tras realizar un análisis de repetibilidad.

En la interfaz de control y visualización de la práctica de control de posición ingresamos un valor de tiempo de desplazamiento y un valor de ángulo, una vez ingresado cada uno de estos valores los enviamos, para que posteriormente se active el motor a pasos, observando su funcionamiento y comprobando que el valor de ángulo desplazado es correcto mediante el uso de un equipo patrón que para este caso es una rueda graduada tipo brújula con un rango de 0° a 360° como podemos evidenciar en el ANEXO K.

La prueba de repetibilidad de esta práctica se observa en la *Tabla 4-3*.

**Tabla 4-3:** Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de posición angular.

N.º de muestra	Fecha/Hora	Parámetros				
		Practica a realizarse	Actuador	Dato de control (Angulo)	Intento de activación	Respuesta del Sistema
1	11/07/2023 10:10:10	Control de posición angular	Motor a pasos	180°	1	1
2	11/07/2023 10:20:00	Control de posición angular	Motor a pasos	180°	1	1
3	11/07/2023 10:30:00	Control de posición angular	Motor a pasos	180°	1	1

4	11/07/2023 10:40:00	Control de posición angular	Motor a pasos	180°	1	1
5	11/07/2023 10:50:00	Control de posición angular	Motor a pasos	180°	1	1
6	11/07/2023 11:00:00	Control de posición angular	Motor a pasos	180°	1	1
7	11/07/2023 11:10:10	Control de posición angular	Motor a pasos	180°	1	1
8	11/07/2023 11:20:00	Control de posición angular	Motor a pasos	180°	1	1
9	11/07/2023 11:30:00	Control de posición angular	Motor a pasos	180°	1	1
10	11/07/2023 11:40:10	Control de posición angular	Motor a pasos	180°	1	1
		Media				1
		Desviación estándar ( $\sigma$ )				0
		Coeficiente de variación estándar (CV)				0%

**Realizado por:** Cevallos, Daniel, 2023.

De la *Tabla 4-3* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variación de datos, teniendo así valores estables. Producto a ello se dice que la comunicación entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores en la práctica de control de posicionamiento angular es confiable.

#### 4.2.2.2. Prueba de funcionamiento de la práctica de control manual de velocidad

El objetivo de esta prueba es determinar la confiabilidad del funcionamiento de la práctica antes nombrada mediante el cálculo del coeficiente de variación tras realizar un análisis de repetibilidad.

En la interfaz de control y visualización de la práctica de control de control manual de velocidad ingresamos un valor de PWM, una vez ingresado este valor, el ESP32 se encarga de activar las salidas correspondientes y procede a activar el motor DC. Observando su funcionamiento y comprobando el valor de velocidad producido en el eje del motor con el uso de un equipo patrón que para este caso es tacómetro laser como podemos observar en el ANEXO K.

La prueba de repetibilidad de esta práctica se observa en la *Tabla 4-4*.

**Tabla 4-4:** Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de velocidad manual.

N.º de muestra	Fecha/Hora	Parámetros				
		Practica a realizarse	Actuador	Dato de control (PWM)	Intento de activación	Respuesta del Sistema
1	11/07/2023 12:10:10	Control de velocidad manual	Motor DC	60	1	1
2	11/07/2023 12:20:00	Control de velocidad manual	Motor DC	60	1	1
3	11/07/2023 12:30:00	Control de velocidad manual	Motor DC	60	1	1
4	11/07/2023 12:40:00	Control de velocidad manual	Motor DC	60	1	1
5	11/07/2023 12:50:00	Control de velocidad manual	Motor DC	60	1	1
6	11/07/2023 13:00:00	Control de velocidad manual	Motor DC	60	1	1
7	11/07/2023 13:10:10	Control de velocidad manual	Motor DC	60	1	1
8	11/07/2023 13:20:00	Control de velocidad manual	Motor DC	60	1	1
9	11/07/2023 13:30:00	Control de velocidad manual	Motor DC	60	1	1
10	11/07/2023 13:40:10	Control de velocidad manual	Motor DC	60	1	1
		Media				1
		Desviación estándar ( $\sigma$ )				0
		Coefficiente de variación estándar (CV)				0%

**Realizado por:** Cevallos, Daniel, 2023.

De la *Tabla 4-4* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variación de datos, teniendo así valores estables. Producto a ello se dice que la comunicación entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores en la práctica de control de velocidad manual es confiable.

#### 4.2.2.3. Prueba de funcionamiento de la práctica control de activación de SSR

En la interfaz de control y visualización de la práctica antes mencionada, por medio de un botón virtual seleccionamos un de los dos SSR, una vez seleccionado el relé, el ESP32 se encarga de activar las salidas correspondientes y procede a activar el relé de estado sólido seleccionado.

La prueba de repetibilidad de esta práctica se observa en la *Tabla 4-5*.

**Tabla 4-5:** Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de activación de SSR.

N.º de muestra	Fecha/Hora	Parámetros				
		Practica a realizarse	Actuador	Dato de control (V)	Intento de activación	Respuesta del Sistema
1	11/07/2023 14:00:10	Control de activación de SSR	SSR	3.3	1	1
2	11/07/2023 14:10:00	Control de activación de SSR	SSR	3.3	1	1
3	11/07/2023 14:20:00	Control de activación de SSR	SSR	3.3	1	1
4	11/07/2023 14:30:00	Control de activación de SSR	SSR	3.3	1	1
5	11/07/2023 14:40:00	Control de activación de SSR	SSR	3.3	1	1
6	11/07/2023 14:50:00	Control de activación de SSR	SSR	3.3	1	1
7	11/07/2023 15:00:00	Control de activación de SSR	SSR	3.3	1	1
8	11/07/2023 15:10:00	Control de activación de SSR	SSR	3.3	1	1
9	11/07/2023 15:20:00	Control de activación de SSR	SSR	3.3	1	1
10	11/07/2023 15:30:10	Control de activación de SSR	SSR	3.3	1	1
		Media				1
		Desviación estándar ( $\sigma$ )				0
		Coeficiente de variación estándar (CV)				0%

Realizado por: Cevallos, Daniel, 2023.

De la *Tabla 4-5* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variación de datos, teniendo así valores estables. Producto a ello se dice que la comunicación

entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores en la práctica de control de activación de SSR es confiable.

#### 4.2.2.4. Prueba de funcionamiento de la práctica de control de temperatura manual

En la interfaz de control y visualización de la práctica antes mencionada, ingresamos un valor de PWM el que enciende la luz LED el cual genera calor en el disipador de calor hasta un valor de temperatura máximo de 50°C, si llega a sobrepasar este valor el ventilador se activara automáticamente para disminuir la temperatura hasta un valor de 20°.

La prueba de repetibilidad de esta práctica se observa en la Tabla 4-6.

**Tabla 4-6:** Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de activación de control de temperatura manual

N.º de muestra	Fecha/Hora	Parámetros				
		Practica a realizarse	Actuador	Dato de control (PWM)	Intento de activación	Respuesta del Sistema
1	12/07/2023 09:20:00	Control manual de temperatura	Luz LED	200	1	1
2	12/07/2023 09:25:00	Control manual de temperatura	Luz LED	200	1	1
3	12/07/2023 09:30:00	Control manual de temperatura	Luz LED	200	1	1
4	12/07/2023 09:35:00	Control manual de temperatura	Luz LED	200	1	1
5	12/07/2023 09:40:00	Control manual de temperatura	Luz LED	200	1	1
6	12/07/2023 09:45:00	Control manual de temperatura	Luz LED	200	1	1
7	12/07/2023 09:50:00	Control manual de temperatura	Luz LED	200	1	1
8	12/07/2023 09:55:00	Control manual de temperatura	Luz LED	200	1	1
9	12/07/2023 10:00:00	Control manual de temperatura	Luz LED	200	1	1
10	12/07/2023 10:05:00	Control manual de temperatura	Luz LED	200	1	1
		Media				1
		Desviación estándar ( $\sigma$ )				0

	Coefficiente de variación estándar (CV)			0%
--	-----------------------------------------	--	--	----

Realizado por: Cevallos, Daniel, 2023.

De la *Tabla 4-6* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variación de datos, teniendo así valores estables. Producto a ello se dice que la comunicación entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores en la práctica de control manual de temperatura es confiable y de calidad.

#### 4.2.2.5. Prueba de funcionamiento de la práctica de control PID de temperatura

En la interfaz de control y visualización de la práctica antes mencionada, ingresamos un valor de temperatura de Setpoint, el cual se ajusta automáticamente para mantener la señal en el valor de referencia que enciende la luz LED el cual genera calor en el disipador de calor hasta un valor de temperatura máximo de  $50^{\circ}\text{C}$ , si excede este valor ventilador se activara automáticamente para disminuir la temperatura y mantenerla en el valor de referencia, caso contrario el sistema aumenta el valor de PWM para elevar la temperatura.

La prueba de repetibilidad de esta práctica se observa en la *Tabla 4-7*.

**Tabla 4-7:** Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control de activación de control PID de temperatura

N.º de muestra	Fecha/Hora	Parámetros				
		Practica a realizarse	Actuador	Dato de control Setpoint ( $T^{\circ}$ )	Intento de activación	Respuesta del Sistema
1	12/07/2023 10:20:00	Control PID de temperatura	Luz LED	$35^{\circ}$	1	1
2	12/07/2023 10:30:00	Control PID de temperatura	Luz LED	$35^{\circ}$	1	1
3	12/07/2023 10:40:00	Control PID de temperatura	Luz LED	$35^{\circ}$	1	1
4	12/07/2023 10:50:00	Control PID de temperatura	Luz LED	$35^{\circ}$	1	1
5	12/07/2023 11:00:00	Control PID de temperatura	Luz LED	$35^{\circ}$	1	1
6	12/07/2023 11:10:00	Control PID de temperatura	Luz LED	$35^{\circ}$	1	1



7	12/07/2023 11:20:00	Control PID de temperatura	Luz LED	35°	1	1
8	12/07/2023 11:30:00	Control PID de temperatura	Luz LED	35°	1	1
9	12/07/2023 11:40:00	Control PID de temperatura	Luz LED	35°	1	1
10	12/07/2023 11:50:00	Control PID de temperatura	Luz LED	35°	1	1
		Media				1
		Desviación estándar ( $\sigma$ )				0
		Coeficiente de variación estándar (CV)				0%

Realizado por: Cevallos, Daniel, 2023.

De la *Tabla 4-7* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variación de datos, teniendo así valores estables. Producto a ello se dice que la comunicación entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores en la práctica de control PID de temperatura es confiable y de calidad.

#### 4.2.2.6. Prueba de funcionamiento de la práctica de control PID de velocidad del motor DC

En la interfaz de control y visualización de la práctica antes mencionada, ingresamos un valor de RPM en el Setpoint, el cual se ajusta automáticamente para mantener la señal de velocidad en el valor de referencia que enciende el motor DC el cual alcanza un valor de máximo de 300 RPM. Si el motor recibe una perturbación externa en su eje y este disminuye su velocidad, el sistema automáticamente eleva el valor de PWM para volver a mantener la velocidad en el valor de referencia, caso contrario el sistema disminuye el valor de PWM para disminuir la velocidad y mantenerla estable.

La prueba de repetibilidad de esta práctica se observa en la *Tabla 4-8*.

**Tabla 4-8:** Prueba de comunicación, adquisición, control, visualización y activación de actuadores en la práctica de control PID de velocidad.

N.º de muestra	Fecha/Hora	Parámetros				
		Practica a realizarse	Actuador	Setpoint (RPM)	Intento de activación	Respuesta del Sistema
1	11/07/2023 12:10:10	Control PID de velocidad	Motor DC	100	1	1

2	11/07/2023 12:20:00	Control PID de velocidad	Motor DC	100	1	1
3	11/07/2023 12:30:00	Control PID de velocidad	Motor DC	100	1	1
4	11/07/2023 12:40:00	Control PID de velocidad	Motor DC	100	1	1
5	11/07/2023 12:50:00	Control PID de velocidad	Motor DC	100	1	1
6	11/07/2023 13:00:00	Control PID de velocidad	Motor DC	100	1	1
7	11/07/2023 13:10:10	Control PID de velocidad	Motor DC	100	1	1
8	11/07/2023 13:20:00	Control PID de velocidad	Motor DC	100	1	1
9	11/07/2023 13:30:00	Control PID de velocidad	Motor DC	100	1	1
10	11/07/2023 13:40:10	Control PID de velocidad	Motor DC	100	1	1
		Media				1
		Desviación estándar ( $\sigma$ )				0
		Coefficiente de variación estándar (CV)				0%

**Realizado por:** Cevallos, Daniel, 2023.

De la *Tabla 4-8* se obtiene un coeficiente de variación  $CV=0\%$ , por lo que, según definición de autores al ser el CV igual a  $0\%$  significa que se tiene muestras compactas, por lo que no existe variación de datos, teniendo así valores estables. Producto a ello se dice que la comunicación entre la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores en la práctica de control PID de velocidad es confiable y de calidad.

De la *Tabla 4-3* a la *Tabla 4-8* se realizó una prueba de repetibilidad de acuerdo a los datos obtenidos en cada práctica, donde se calculó un coeficiente de variación (CV) de  $0\%$ , lo que, de acuerdo a la interpretación de los autores citados al inicio de este capítulo nos indica que las muestras son uniformes y que además no existe variación en los datos, lo que implica que los valores son constantes. Como resultado de esto, se afirma que la etapa de adquisición de datos, de comunicación y etapa de control y visualización y etapa de actuadores presente en cada una de las practicas mencionadas en esta sección funciona de manera adecuada, lo que nos lleva a concluir que el funcionamiento de cada práctica es confiable y de gran calidad.

### **4.3. Obtención de las funciones de transferencia de cada una de las plantas que componen la PLCP**

En esta sección se detalla paso a paso los procesos que se realizaron para obtener las funciones de transferencia de cada planta, ya que estas son herramientas muy importantes que nos permite conocer el comportamiento de determinado proceso.

#### ***4.3.1. Función de transferencia de la planta de temperatura***

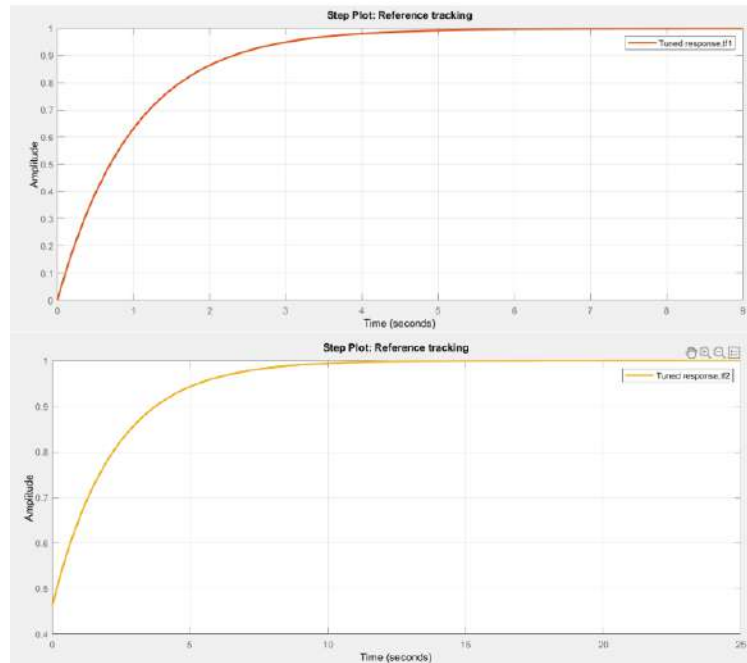
En la etapa de actuadores de la PLCP, contamos con una luz LED de alta potencia que se encarga de proveer calor al sensor de temperatura. Adicionalmente, incorporamos un ventilador de 12 V cuyo propósito es disminuir la temperatura en el disipador de calor.

Para establecer el modelo matemático del proceso térmico, se emplea el microcontrolador usado en este proyecto para establecer la comunicación serial y en LabVIEW desarrollamos el diagrama de bloques que utilizamos para aplicar los distintos datos de PRBS a través del microcontrolador a nuestra planta. A partir de ello almacenamos los datos de salida de la planta y generamos una función de transferencia en Matlab por medio de la herramienta de identificador de sistemas.

Cabe destacar que esta señal PRBS se generó previamente en Matlab la cual tiene un rango de variación entre 0 y 12. Los datos generados desempeñan un papel esencial al ajustar la salida de PWM que alimenta la luz LED, generando así un aumento de temperatura en esta fuente lumínica. La medición de esta temperatura se lleva a cabo utilizando el sensor DS18B20. El ESP32, en su rol de procesador de señales eléctricas, recopila los datos mencionados, emitiendo a través del puerto serial el valor de temperatura en grados centígrados [°C] cada 100 milisegundos. Este proceso se repite 128 veces, correspondiendo al tamaño muestral definido por la señal PRBS.

##### ***4.3.1.1. Identificación del sistema en Matlab de la planta de temperatura***

Para adquirir la respuesta de la señal, se utilizan los datos que se habían registrado con antelación en una hoja de cálculo y se introducen en el programa IDENT de Matlab. Esto se realiza con el propósito de obtener tanto las representaciones gráficas del sistema como sus correspondientes funciones de transferencia como se muestra a continuación:



**Ilustración 4-2:** Obtención de las funciones de transferencia de la planta de temperatura  
**Realizado por:** Cevallos, Daniel, 2023.

De la planta de temperatura se obtuvo dos funciones de transferencia la primera (tf1) con dos polos y un cero, la segunda solo un polo, (tf2).

```

>> tf1

tf1 -

From input "u1" to output "y1":
  0.07161 s + 0.001418
-----
s^2 + 0.02136 s + 3.369e-08

Name: tf1
Continuous-time identified transfer function.

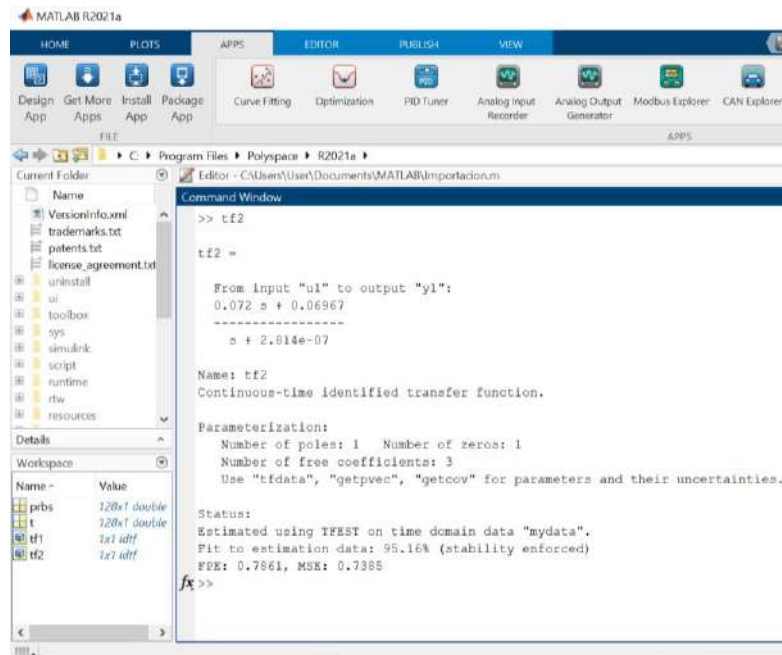
Parameterization:
Number of poles: 2   Number of zeros: 1
Number of free coefficients: 4
Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.

Status:
Estimated using TFEST on time domain data "mydata".
Fit to estimation data: 94.82% (stability enforced)
FPE: 0.929%, MSE: 0.0462

fx >>

```

**Ilustración 4-3:** Función de transferencia tf1 obtenida de la planta de Temperatura  
**Realizado por:** Cevallos, Daniel, 2023.

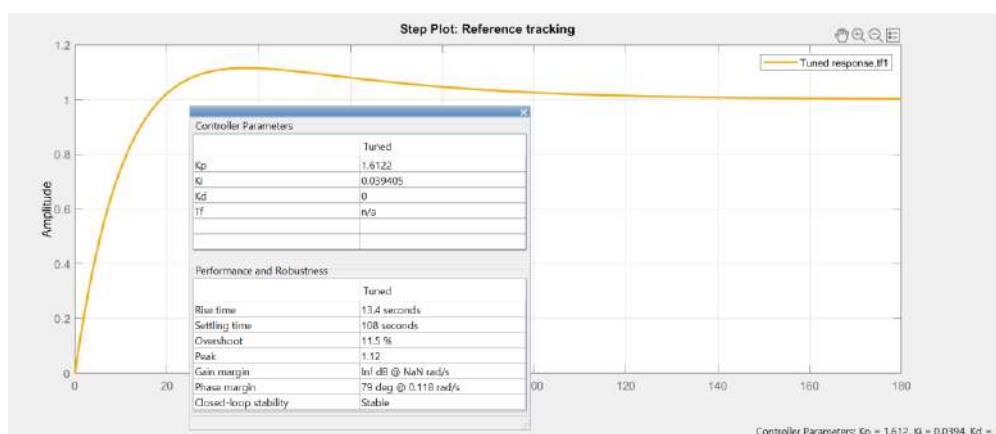


**Ilustración 4-4:** Función de transferencia tf2 obtenida de la planta de Temperatura  
**Realizado por:** Cevallos, Daniel, 2023.

Como se observa en la *Ilustración 4-3* y la *Ilustración 4-4* la primera función de transferencia tf1 tiene un 94.82% de aproximación y tf2 es tiene 95.16%, siendo así tf2 la función de transferencia más aproximada.

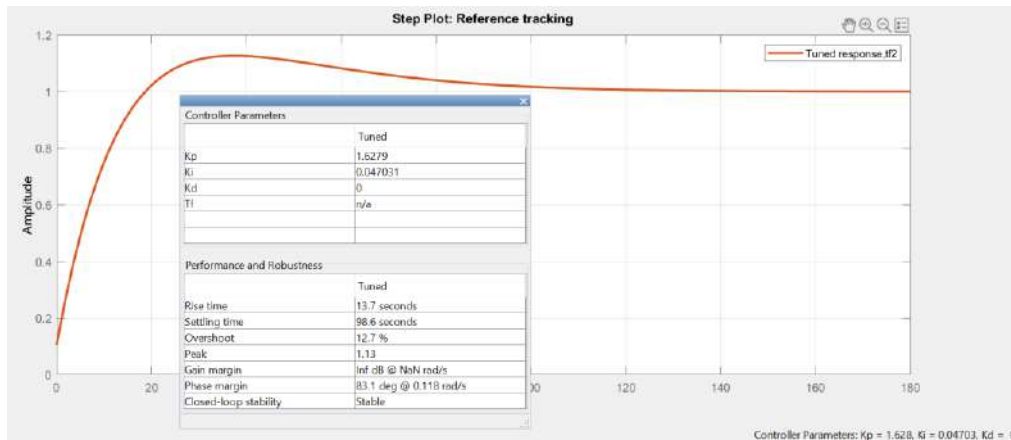
#### 4.3.1.2. Ajuste de ganancias del controlador PID de la planta de temperatura

La aplicación PID Tuner modifica las constantes de un controlador PID de manera automática para un sistema de control de una única entrada y una única salida (SISO), con el objetivo de alcanzar un balance entre el desempeño y la estabilidad. Una vez usada esta herramienta encontramos los siguientes valores de las ganancias para nuestro controlador.



**Ilustración 4-5:** Ganancias de controlador de la función de transferencia  $tf1$  obtenida de la planta de Temperatura

**Realizado por:** Cevallos, Daniel, 2023.



**Ilustración 4-6:** Ganancias de controlador de la función de transferencia  $tf2$  obtenida de la planta de Temperatura

**Realizado por:** Cevallos, Daniel, 2023.

#### 4.3.2. Función de transferencia de la planta del Motor DC

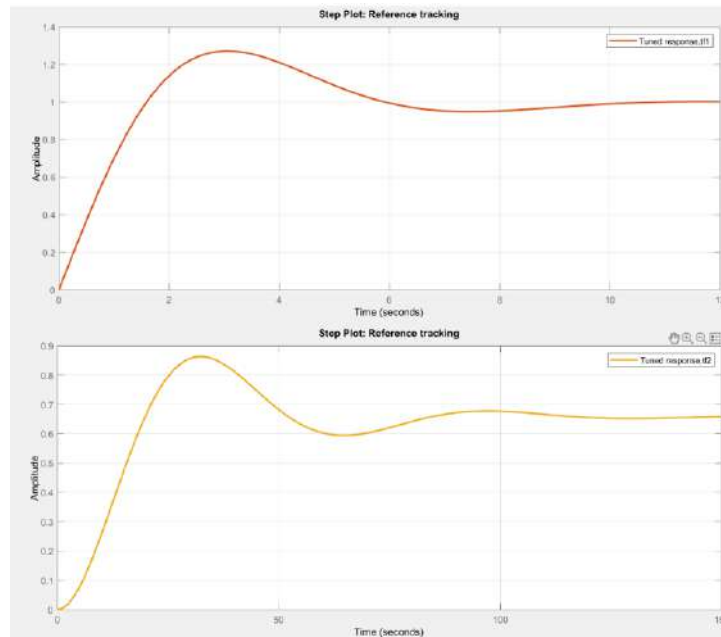
En la etapa de actuadores de la PLCP, contamos con una luz LED de alta potencia que se encarga de proveer calor al sensor de temperatura. Adicionalmente, incorporamos un ventilador de 12 V cuyo propósito es disminuir la temperatura en el disipador de calor.

Para establecer el modelo matemático del proceso térmico, se emplea el microcontrolador usado en este proyecto para establecer la comunicación serial y en LabVIEW desarrollamos el diagrama de bloques que utilizamos para aplicar los distintos datos de PRBS a través de Arduino a nuestra planta. A partir de ello almacenamos los datos de salida de la planta y generamos una función de transferencia en Matlab por medio de la herramienta de identificador de sistemas.

Cabe destacar que esta señal PRBS se generó previamente en Matlab la cual tiene un rango de variación entre 2 y 12. Los datos generados desempeñan un papel esencial al ajustar la salida de PWM que alimenta al motor DC, generando así un aumento o decremento de la velocidad. La medición de velocidad se lleva a cabo utilizando el encoder de cuadratura provisto en el mismo motor. El Arduino, en su rol de procesador de señales eléctricas, recopila los datos mencionados, emitiendo a través del puerto serial el valor de velocidad en RPM. Este proceso se repite 128 veces, correspondiendo al tamaño muestral definido por la señal PRBS.

#### 4.3.2.1. Identificación del sistema en Matlab de la planta del motor DC

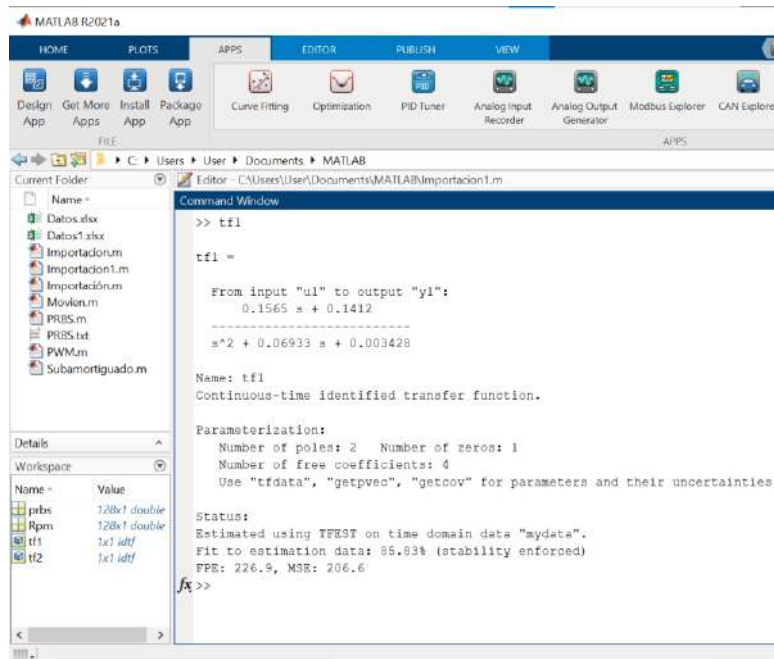
Para adquirir la respuesta de la señal, se utilizan los datos que se habían registrado con antelación en una hoja de cálculo y se introducen en el programa IDENT de Matlab. Esto se realiza con el propósito de obtener tanto las representaciones gráficas del sistema como sus correspondientes funciones de transferencia como se muestra a continuación:



**Ilustración 4-7:** Obtención de las representaciones gráficas del sistema en la planta de planta del Motor DC

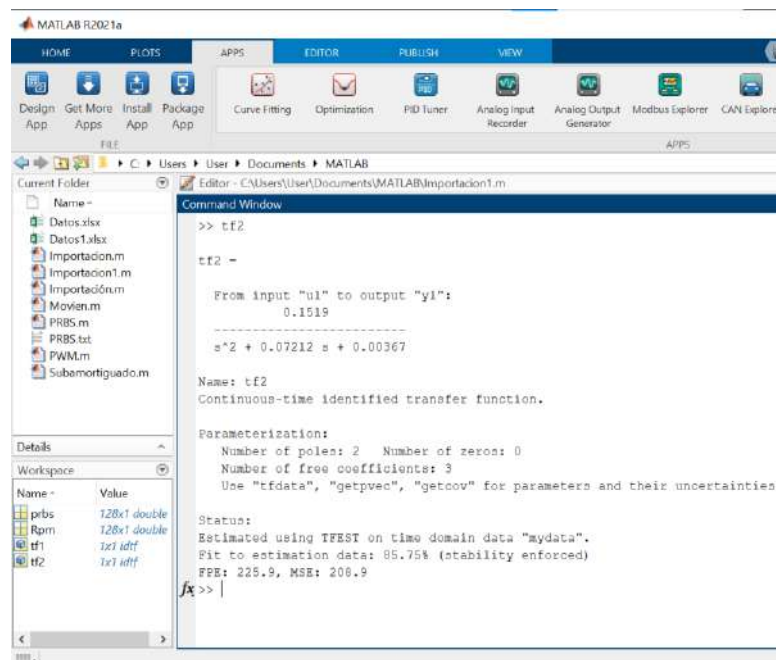
**Realizado por:** Cevallos, Daniel, 2023.

De la planta de planta Motor DC se obtuvo dos funciones de transferencia la primera (tf1) con dos polos y un cero, la segunda solo 2 polos y ningún cero, (tf2).



**Ilustración 4-8:** Función de transferencia tf1 obtenida de la planta del Motor DC

Realizado por: Cevallos, Daniel, 2023.



**Ilustración 4-9:** Función de transferencia tf2 obtenida de la planta del Motor DC

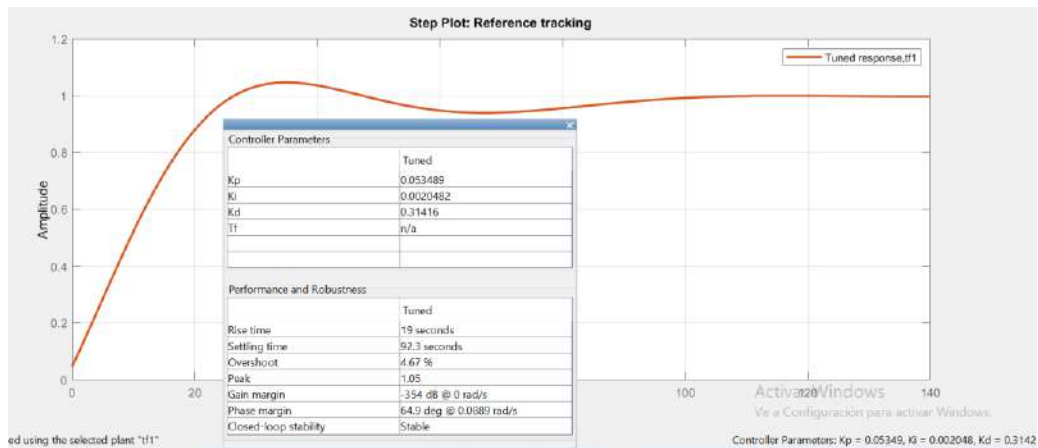
Realizado por: Cevallos, Daniel, 2023.

Como se observa en la *Ilustración 4-8* y la *Ilustración 4-9* la primera función de transferencia tf1 tiene un 85.83% de aproximación y tf2 es tiene 85.77%, siendo así tf1 la función de transferencia más aproximada.



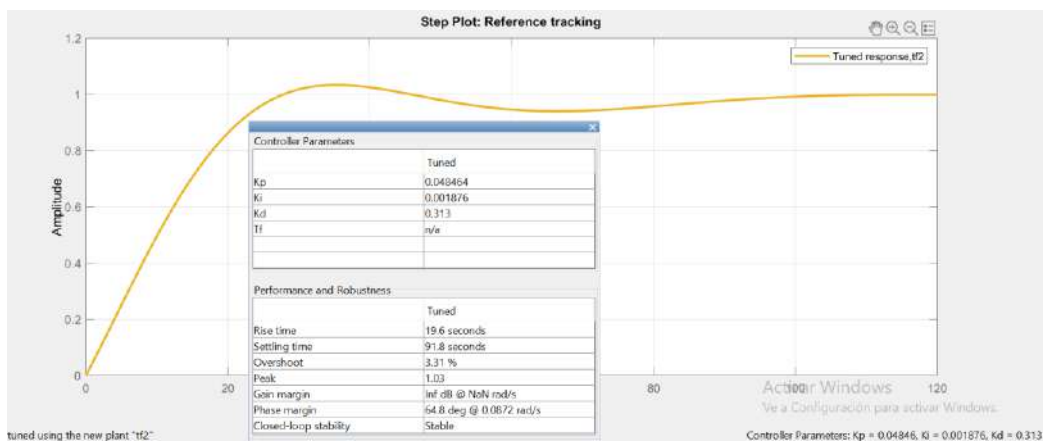
#### 4.3.2.2. Ajuste de ganancias del controlador PID de la planta de temperatura

La aplicación PID Tuner modifica las constantes de un controlador PID de manera automática para un sistema de control de una única entrada y una única salida (SISO), con el objetivo de alcanzar un balance entre el desempeño y la estabilidad. Una vez usada esta herramienta encontramos los siguientes valores de las ganancias para nuestro controlador.



**Ilustración 4-10:** Ganancias de controlador de la función de transferencia tf1 obtenida de la planta de Temperatura

Realizado por: Cevallos, Daniel, 2023.



**Ilustración 4-11:** Ganancias de controlador de la función de transferencia tf2 obtenida de la planta de Temperatura

Realizado por: Cevallos, Daniel, 2023.

Una observación que debemos hacer sobre PID Tuner es que esta aplicación asume que el modelo de la planta es preciso y adecuado. Si bien el modelo no refleja fielmente el comportamiento real de la planta, los valores de sintonización generados pueden no ser óptimos para el sistema en el

mundo real. Por esta razón es más recomendable validar los valores de sintonización en un entorno experimental o real para asegurarse de que el sistema se comporte como se espera.

Por esta razón usamos el método experimental para encontrar los valores de las ganancias de los controladores PID de las plantas de temperatura y del motor DC, los cuales nos llevan a que los sistemas se comporten como nosotros lo esperamos. En la *Tabla 4-9* se observan los datos experimentales encontrados de las ganancias de los controladores que serán usados en la practicas de control PID de temperatura y de control de velocidad.

**Tabla 4-9:** Datos experimentales encontrados para los controladores PID de las plantas de temperatura y motor DC

Planta	KP	KI	KD
Temperatura (Luz Led )	0.1000	0.0050	0.0080
Motor DC	0.2000	0.0050	0.0010

Realizado por: Cevallos, Daniel, 2023.

#### 4.4. Consumo energético de la PLCP

Para verificar el consumo de energía del sistema, se llevaron a cabo mediciones de corriente utilizando el multímetro digital Proskit modelo MT-1236. De esta manera, se confirmaron los valores de funcionamiento que se detallan en las hojas de datos. Estos datos son de utilidad para determinar la fuente de alimentación eléctrica adecuada para la PLCP en función de la carga total que el sistema absorbe. La *Tabla 4-10* proporciona un análisis detallado de los valores consumidos por cada componente del sistema.

**Tabla 4-10:** Consumo energético según la práctica realizada en la PLCP

Practica	Consumo total independiente (mA)
Control de posición angular	1876.12
Control manual de velocidad	1462.32
Control de activación de SSR	266.63
Control de temperatura manual	1162.33
Control PID de temperatura	1278.24
Control PID de velocidad	1762.33

Realizado por: Cevallos, Daniel, 2023

Es importante destacar que todas las prácticas realizadas en la PLCP funcionan de manera independiente. Esto se refleja en la *Tabla 4-10*, donde se observa que la práctica que implica el control de posición angular registra el mayor consumo energético, con un valor de 1876.12 mA, mientras que la práctica relacionada con el control de activación de SSRs presenta un consumo menor, de 0.27 A. En base a estos datos, seleccionamos una fuente de alimentación comercial con una capacidad de 2 A para garantizar el funcionamiento adecuado de la PLCP.

#### 4.5. Análisis económico para la construcción del PSES

En la *Tabla 4-11* se presenta el costo total requerido para la construcción de la PLCP, para ello se hace mención a los costos unitarios pertenecientes a cada elemento necesario para la construcción del mismo y la cantidad requerida de dichos elementos. Los costos expuestos a continuación corresponden a su adquisición dentro de Ecuador.

**Tabla 4-11:** Análisis económico para la construcción del PLCP

Componente	Cantidad	Costo unitario (USD)	Costo total (USD)
Esp32	1	\$ 25.99	\$ 25.99
Fuente conmutada MP1584	1	\$ 1.20	\$ 1.20
Driver DRV8825	1	\$ 3.50	\$ 3.50
Driver TB6612FNG	1	\$ 10.00	\$ 10.00
Sensor DS18B20		\$ 2.99	\$ 2.99
Módulo Relé de estado sólido G3MB-202P	1	\$ 7.99	\$ 7.99
Motorreductor JAG25-370	1	\$ 30.50	\$ 30.50
Led de potencia de 10W	1	\$ 1.00	\$ 1.00
Motor NEMA 17HD48002H- 22B	1	\$ 17.99	\$ 17.99
Encoder óptico infrarrojo H2010	1	\$ 3.50	\$ 3.50
Caja de conexiones Heyiarbeit	1	\$ 30.00	\$ 30.00
Placa PCB	1	\$ 27.00	\$ 27.00
Impresiones 3D	1	\$ 6.00	\$ 6.00
Acrílico	1	\$ 2.00	\$ 2.00
Fuente de alimentación 2A	1	\$ 7.50	\$ 7.50
Brújula	1	\$ 8.50	\$ 8.50
Ventilador 12V	1	\$ 10.00	\$ 10.00
Buzzer	1	\$ 1.00	\$ 1.00
Cable micro USB	1	\$ 1.00	\$ 1.00
Elementos eléctricos y electrónicos	1	\$ 8.00	\$ 8.00

Otros elementos	1	\$ 10.00	\$ 10.00
COSTO TOTAL DE LA PLCP			\$ 215.16

**Realizado por:** Cevallos, Daniel, 2023.

En la *Tabla 4-11* se muestra el análisis económico del costo total de construcción de la PLCP que detallan sus componentes, la cantidad y sus respectivos precios de la implementación del sistema. Donde se observa que el sistema implementado tiene un costo total de 215.16 dólares americanos (USD).

#### **4.6. Funcionamiento general de la PLCP**

Una vez habiendo realizadas las pruebas en cada una de las etapas que conforma la PLCP de manera individual, como siguiente paso se procedió a la implementación de la planta de laboratorio de control de procesos por completo véase en la *Ilustración 4-12*.



**Ilustración 4-12:** Implementación de la planta de laboratorio de procesos (PLCP) finalizada

**Realizado por:** Cevallos, Daniel, 2023.

Posteriormente, encontrándose el PLCP ensamblado en su totalidad se realizaron pruebas orientadas al funcionamiento del sistema para ello:

En repetidas ocasiones se puso en funcionamiento cada elemento de la etapa de adquisición de datos por individual, incorporado en cada actuador, las respuestas a cada una de estas pruebas resultaron ser favorables, obteniendo los valores de temperatura, velocidad y posición por parte de la etapa de adquisición.

La comunicación entre la etapa de adquisición, de comunicación y módulo de control y visualización y etapa de actuadores en cada una de las practicas llevadas a cabo respondieron de manera correcta, estableciendo una comunicación serial entre la PLCP y el computador donde los

programas realizados en LabVIEW trabajaron como medio de control y visualización, enviando, recibiendo y controlando cada uno de los valores relacionados a cada una de las seis prácticas.

Continuamente se realizaron pruebas de: establecimiento de comunicación serial PLCP-computador, control y visualización de información, activación y desactivación de actuadores, teniendo resultados adecuados por cada prueba aplicada.

Los valores de temperatura, velocidad y posición angular generados por la PLCP en cada uno de sus procesos correspondientes se compararon con instrumentos patrones. Esto permitió verificar que los valores obtenidos por el sistema coincidieron con los valores medidos.

#### **4.7. Manual de usuario de la PLCP**

Para llevar a cabo las prácticas, es necesario:

##### ***4.7.1. Requerimientos de hardware***

- Planta de laboratorio de control de procesos (PLCP)
- Computador
- Fuente de poder AC/DC de 12V, 2000 mA
- Cable micro USB para la conexión de PLC-Computador
- Destornillador para bornera
- Cargas de tensión alterna de 100VAC hasta 220VAC con corriente de hasta 2<sup>a</sup>

##### ***4.7.2. Requerimientos de software***

- LabVIEW 2021 o versiones superiores
- NI-VISA para controlar y configurar los puertos serie y USB
- Recomendado: Toolkit de diseño e identificación de sistemas de control

##### ***4.7.3. Prerrequisitos para realizar las practicas***

- Conocimientos de programación en LabVIEW fundamental.
- Instrumentos para conocer sobre las opciones de capacitación.
- Conocimientos básicos de sistemas de adquisición de datos y sensores
- Para las prácticas de control, conocimientos básicos de teoría de control moderno

#### **4.8. Manual de prácticas de laboratorio**

Cada una de estas prácticas ofrece una breve descripción del equipo utilizado y uno o más programas de aplicación creados en LabVIEW. Estos programas se proporcionan con su código fuente abierto, lo que permite emplearlos como punto de partida para desarrollar aplicaciones más avanzadas o como ejemplos de estudio.

Es posible emplear los programas proporcionados, aunque se aconseja crear los programas desde cero con el fin de optimizar el proceso de aprendizaje y enriquecer las experiencias, comprensión de conceptos e ideas para futuras prácticas.

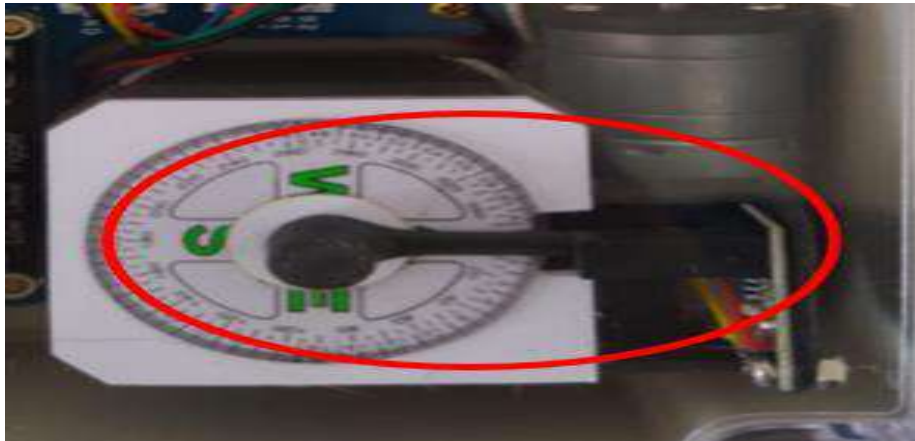
#### 4.8.1. Práctica 1

**Tema:** Control de posición angular

**Objetivo:** Realizar el control de posición de un motor a pasos

#### Marco teórico

El EPC incluye un motor de pasos unipolar (Stepper) de cuatro entradas digitales disponibles en los terminales A, A', B y B'. Conforme se activan las líneas digitales en determinadas secuencias, el motor gira paso a paso. La PLCP también incluye un encoder infrarrojo el cual detecta la posición inicial del indicador como se observa en la *Ilustración 4-13*.



**Ilustración 4-13:** Posición inicial del indicador en el encoder infrarrojo H2010

**Realizado por:** Cevallos, Daniel, 2023.

En todas las prácticas, a excepción de la práctica de control de activación de SSRs que se usan cargas de corriente alterna, no es necesario conectar cables desde la tarjeta de adquisición de datos hacia los actuadores as entradas o salidas como es normal en las plantas entrenadoras de control (EPC) de National Instruments. Esto se debe a que todos los elementos, como los actuadores, la tarjeta de desarrollo y los componentes electrónicos, están integrados en una tarjeta PCB evitando el uso de cableado extra como se evidencia en la *Ilustración 4-14*.





**Ilustración 4-14:** Vista superior de la PLCP

**Realizado por:** Cevallos, Daniel, 2023.

### **Equipos y materiales**

- Planta de laboratorio para el control de procesos (PLCP)
- Fuente de poder AC/DC de 12 V, 1200 mA
- Computador con software LabVIEW instalado

### **Desarrollo**

En esta práctica se utiliza un control numérico para especificar la posición angular a la que debe apuntar el indicador provisto en el eje del motor a pasos. Dada las especificaciones de este equipo, la posición final tendrá una precisión de  $1.1^\circ$  debido a la resolución del paso.

1. Realizar las siguientes conexiones entre la PLCP y el computador:
  - a. Conectamos la fuente de alimentación DC de 12 V a la PLCP y el cable de comunicación micro USB como se muestra en la Ilustración 15-4.

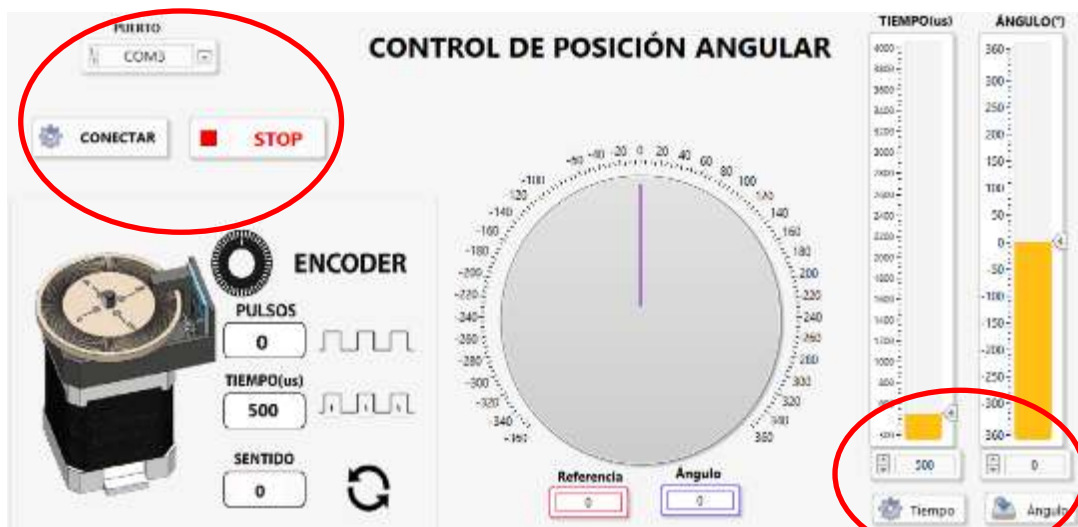


**Ilustración 4-15:** Conexión de cable de alimentación y comunicación

**Realizado por:** Cevallos, Daniel, 2023.

- b. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
2. Abrir el programa **CONTROL DE POSICIÓN.vi**. El panel frontal presenta los siguientes objetos
    - a. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
    - b. Un control numérico tipo Slide **TIEMPO( $\mu$ s)** y para especificar el tiempo entre pasos. Incluye un control tipo numérico para colocar valores exactos.
    - c. Un control numérico tipo Slide **ANGULO ( $^{\circ}$ )** para especificar la posición angular en grados a la cual debe apuntar el indicador del motor a pasos. Incluye un control tipo numérico para colocar valores exactos.
    - d. Un botón **Tiempo** que envía el valor de tiempo entre pasos ingresado por el control numérico.
    - e. Un botón **Ángulo** que envía el valor de ángulo entre pasos ingresado por el control numérico.

- f. Un botón de control booleano denominado **CONECTAR** permite iniciar la comunicación entre LabVIEW y el ESP32. Esto se logra identificando la práctica en ejecución y activando o desactivando las entradas y salidas mediante la programación en la tarjeta de desarrollo
- g. Un botón de control booleano, llamado **DESCONECTAR**, se muestra cuando el programa ha comenzado su ejecución. Este botón finaliza la comunicación entre LabVIEW y la PLCP al desactivar la ejecución de la programación en el ESP32.
- h. Una paleta de selección llamada **PUERTO** que nos permite elegir el puerto de comunicación al que está conectada la PLCP.

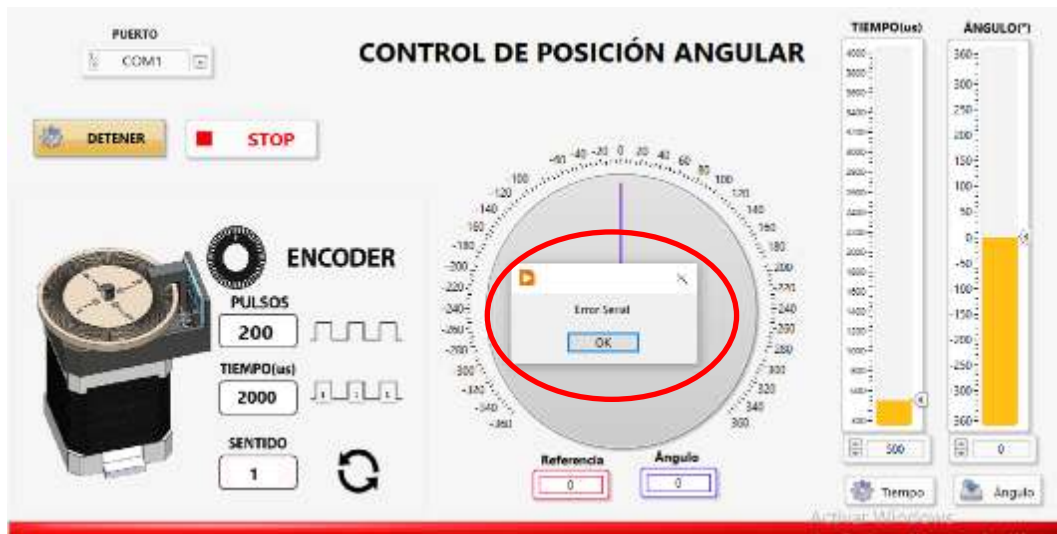


**Ilustración 4-16:** Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de posición

Realizado por: Cevallos, Daniel, 2023.

3. Correr el programa
  - a. Para seleccionar el puerto de comunicación al que está conectada la PLCP, utilizamos la paleta de selección **PUERTO**. En este caso, el puerto es COM3. Si no aparece en la lista, podemos hacer clic en el botón **Refresh** dentro de la misma paleta para actualizar la lista de puertos y elegir el puerto correcto.
  - b. Al hacer clic en el botón "RUN" en la barra de herramientas de LabVIEW, iniciamos la ejecución del programa y establecemos la comunicación con el PLC. Si no se establece la comunicación con éxito, se generará un mensaje de Error Serial y aparecerá una barra de

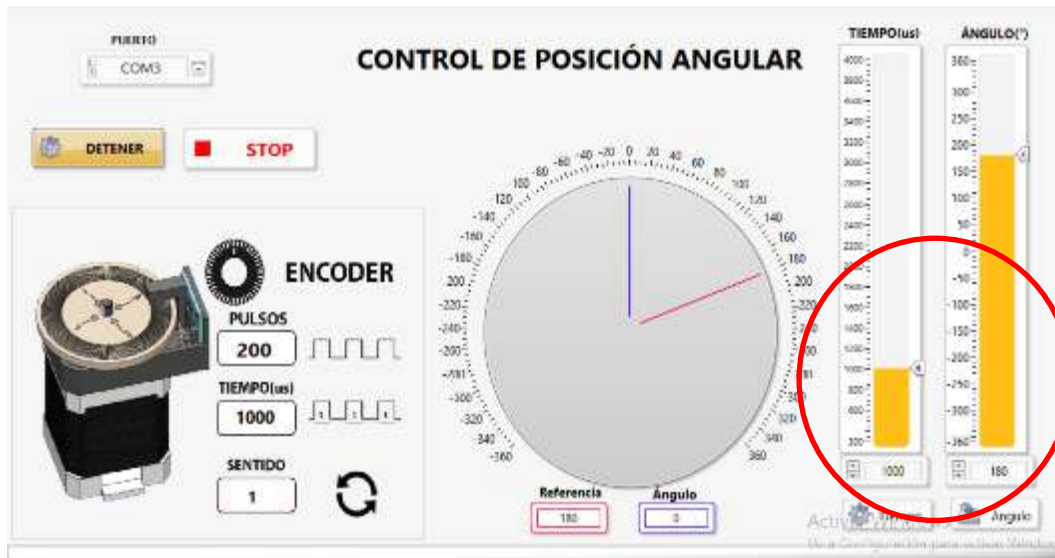
alerta de color rojo en la parte inferior del panel frontal, como se ilustra en la *Ilustración 4-17*. Esto indica que no se ha logrado establecer la comunicación con el puerto seleccionado.



**Ilustración 4-17:** Mensaje de error en la comunicación serial en la práctica de control de posición

**Realizado por:** Cevallos, Daniel, 2023.

- c. Si la posición del indicador ubicado en el eje del motor a pasos no se encuentra en la posición inicial  $0^\circ$ , este automáticamente volverá a su posición de inicio al presionar el botón **CONECTAR**.
- d. Seleccionar un **TIEMPO(μs)** entre pasos de (300-4000)  $\mu\text{s}$  y un **ÁNGULO(°)** de desplazamiento de (-360 a 360) mediante el control numérico Slide. Mientras más alto sea este valor, la velocidad del desplazamiento del indicador será baja caso contrario, si el valor es menor la velocidad de desplazamiento será mayor. Para este caso seleccionamos un valor de 2000  $\mu\text{s}$ , véase en la *Ilustración 4-18*.

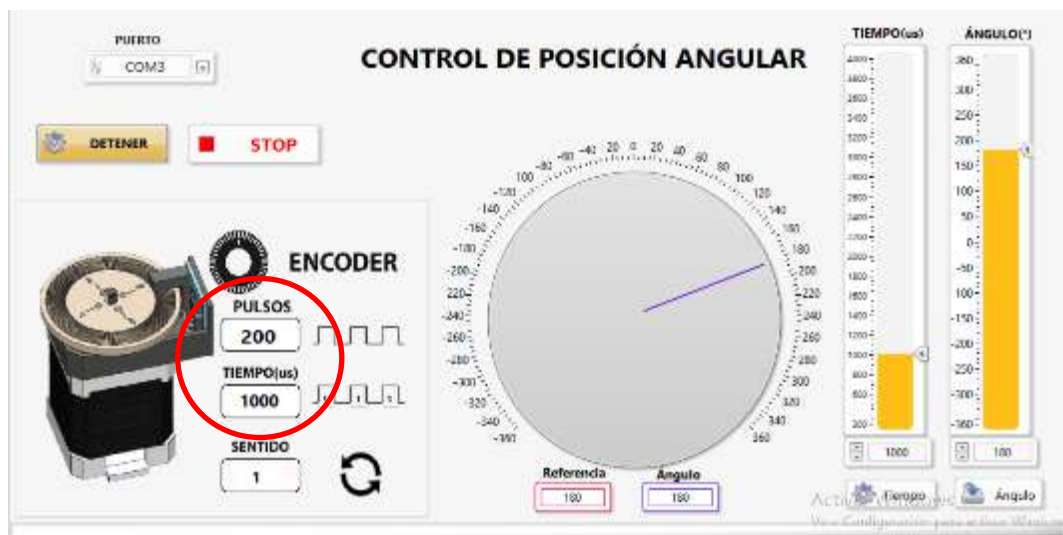


**Ilustración 4-18:** Ingreso de valores de TIEMPO( $\mu$ s) y  $\text{ÁNGULO}(\text{°})$

Realizado por: Cevallos, Daniel, 2023.

- e. Cuando enviamos el valor del tiempo, este se muestra en la parte izquierda del panel frontal. Además, al enviar el valor del ángulo, el motor a pasos se activa automáticamente, mostrando el número de pulsos o pasos necesarios para el desplazamiento, que en este caso es de 200 pulsos, como se ilustra en la *Ilustración 4-19*.

Z



**Ilustración 4-19:** Visualización de TIEMPO( $\mu$ s) ingresado y pulsos generados para un desplazamiento de  $180^\circ$

Realizado por: Cevallos, Daniel, 2023.

4. Abrir el diagrama de bloques para analizar el algoritmo

Dirigirse al ANEXO I para observar la programación en diagramas de bloque.

5. Cerrar el programa sin guardar cambios

**NOTA:** Tanto al inicio como al término de la comunicación entre LabVIEW y el PLC, se emitirá una alerta acústica en cada una de las prácticas descritas en este manual.

#### **4.8.2. Practica 2**

**Tema:** Control manual de velocidad

**Objetivo:** Realizar el control de velocidad de un motor DC

#### **Marco teórico**

La PLCP incluye un motor de corriente continua (Motor DC) en cuyo eje está acoplado un encoder de cuadratura encargado de medir la velocidad. El motor es controlado por una señal PWM que puede variar entre 0 y 255.

Es posible emplear los programas proporcionados, no obstante, se recomienda crear los programas desde cero con el propósito de lograr una comprensión más profunda de su funcionamiento, enriquecer la experiencia de desarrollo y fomentar la generación de ideas para mejorarlos e incorporar nuevas funcionalidades.

#### **Desarrollo**

En esta práctica se utilizará un control de tipo numérico en el Panel Frontal para controlar la señal de alimentación del Motor DC, y así controlar manualmente la velocidad.

1. Realizar las siguientes conexiones entre la PLCP y el computador:
  - a. Conectamos la fuente de alimentación DC de 12 V a la PLCP y el cable de comunicación micro USB como se muestra en la Ilustración 15 -4.
  - b. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
2. Abrir el programa **CONTROL MANUAL VELOCIDAD.vi**. El panel frontal presenta los siguientes objetos
  - a. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.

- b. Un control numérico tipo Slide **PWM** para especificar el valor de la señal entregada hacia el motor DC. Incluye un control tipo numérico para colocar valores exactos.
- c. Un botón de control booleano denominado **CONECTAR** permite iniciar la comunicación entre LabVIEW y el ESP32. Esto se logra identificando la práctica en ejecución y activando o desactivando las entradas y salidas mediante la programación en la tarjeta de desarrollo.
- d. Un botón de control booleano, llamado **DESCONECTAR**, se muestra cuando el programa ha comenzado su ejecución. Este botón finaliza la comunicación entre LabVIEW y la PLCP al desactivar la ejecución de la programación en el ESP32.
- e. Una paleta de selección llamada **PUERTO** que nos permite elegir el puerto de comunicación al que está conectada la PLCP.
- f. El botón de **STOP** detiene el programa. Es importante detener el programa utilizando este botón y no el botón de abortar de la barra de herramientas porque así se asegura que al parar el programa se detenga correctamente el motor DC y la comunicación.

En la *Ilustración 4-20* se muestran los controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de velocidad manual.



**Ilustración 4-20:** Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de velocidad manual

**Realizado por:** Cevallos, Daniel, 2023.



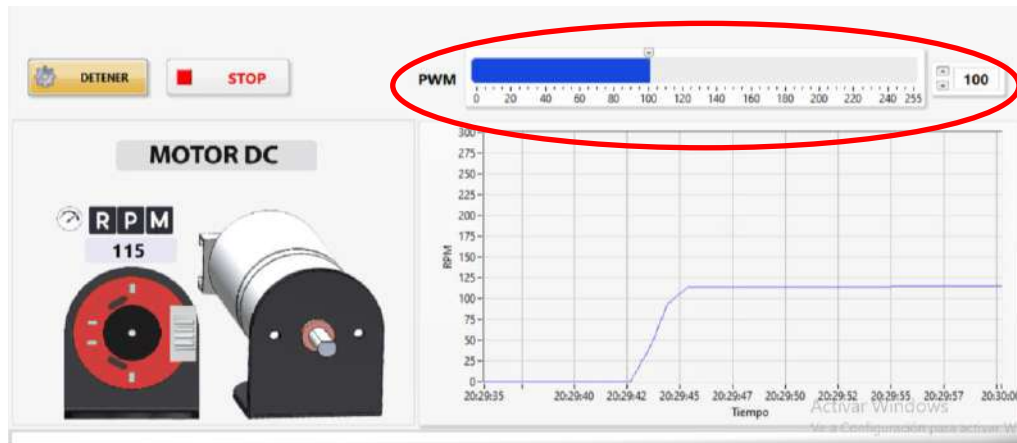
3. Correr el programa
  - a. Para seleccionar el puerto de comunicación al que está conectada la PLCP, utilizamos la paleta de selección **PUERTO**. En este caso, el puerto es COM3. Si no aparece en la lista, podemos hacer clic en el botón Refresh dentro de la misma paleta para actualizar la lista de puertos y elegir el puerto correcto.
  - b. Al hacer clic en el botón **RUN** en la barra de herramientas de LabVIEW, iniciamos la ejecución del programa y establecemos la comunicación con el PLC. Si no se establece la comunicación con éxito, se generará un mensaje de Error Serial y aparecerá una barra de alerta de color rojo en la parte inferior del panel frontal, como se ilustra en la *Ilustración 4-21*. Esto indica que no se ha logrado establecer la comunicación con el puerto seleccionado.



**Ilustración 4-21:** Mensaje de error en la comunicación serial en la práctica de control de velocidad manual

**Realizado por:** Cevallos, Daniel, 2023.

- c. Pulsamos el botón **CONECTAR** el cual inicia la comunicación entre LabVIEW y la PLCP. Identificando la práctica en ejecución, activando o desactivando las entradas y salidas mediante la programación dentro de la tarjeta de desarrollo.
- d. Seleccionar un valor de PWM entre (0-255) que para este es un valor de 100 el cual se lo puede observar en la *Ilustración 4-22*, generando una velocidad de 115 RPM como se muestra en la *Ilustración 4-23*. Mientras más alto sea este valor, la velocidad del motor aumentará caso contrario, si el valor es menor la velocidad disminuirá.



**Ilustración 4-22:** Ingreso de valores PWM en el control numérico

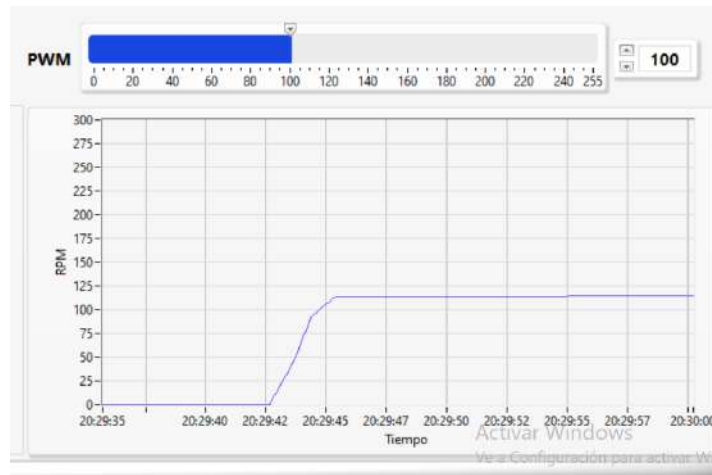
Realizado por: Cevallos, Daniel, 2023.



**Ilustración 4-23:** Velocidad en RPM generada por la señal PWM ingresada al sistema

Realizado por: Cevallos, Daniel, 2023

La simulación de la práctica de control de velocidad manual nos genera una gráfica (velocidad vs tiempo) en la que podemos observar el comportamiento de la velocidad ante el estímulo de la señal PWM como se indica en la *Ilustración 4-24*.



**Ilustración 4-24:** Gráfica Velocidad (RPM) vs Tiempo

**Realizado por:** Cevallos, Daniel, 2023

4. Abrir el diagrama de bloques para analizar el algoritmo

Dirigirse al ANEXO I para observar la programación en diagramas de bloques.

5. Cerrar el programa sin guardar cambios

### 4.8.3. Practica 3

**Tema:** Control de activación de SSR

**Objetivo:** Realizar el control de activación de SSR

#### **Marco teórico**

La PLCP cuenta con un módulo de relé de estado sólido (SSR, por sus siglas en inglés, Solid State Relay) el cual nos permite controlar la conmutación de energía eléctrica alterna. A diferencia de los relés electromecánicos tradicionales, los SSRs no tienen partes móviles y utilizan componentes de estado sólido como optoacopladores, tiristores o transistores para lograr la conmutación.

El relé funciona con lógica inversa; es decir

- Cuando se envía un estado bajo (0V) al canal 1 o canal 2 del relé, éste se activa conmutando el terminal SW1 o SW2.
- Cuando se envía un estado alto (5V) al canal 1 o canal 2 del relé, éste se desactiva la conmutación el terminal SW1 o SW2.

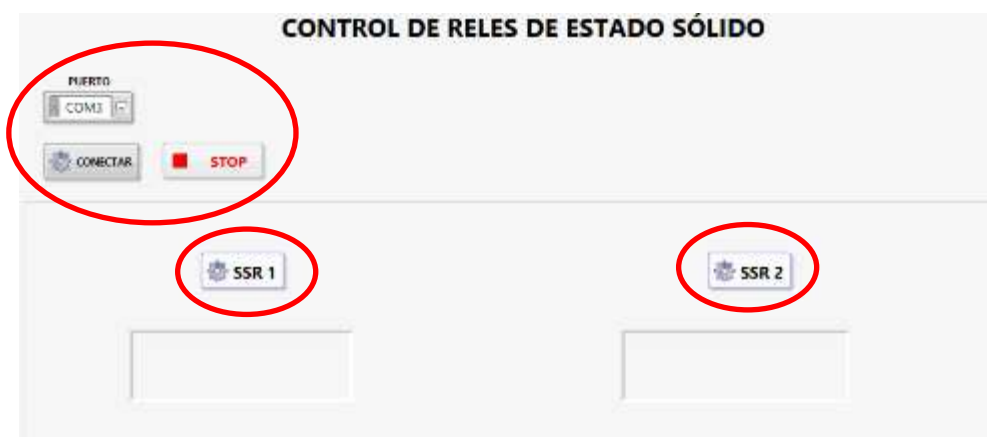
#### **Desarrollo**

En esta práctica se utilizará un control de tipo numérico en el Panel Frontal para controlar la señal de alimentación del Motor DC, y así controlar manualmente la velocidad.

1. Realizar las siguientes conexiones entre la PLCP y el computador:
  - a. Conectamos la fuente de alimentación DC de 12 V a la PLCP y el cable de comunicación micro USB como se muestra en la Ilustración 15 -4.
  - b. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
2. Abrir el programa CONTROL ACTIVACIÓN SSR.vi. El panel frontal presenta los siguientes objetos

- a. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
- b. Dos controles tipo booleano **SSR1** y **SSR2** nos permite activar y desactivar cada uno de los SSRs conmutando los terminales SW1 Y SW2.
- c. Un botón de control booleano denominado **CONECTAR** permite iniciar la comunicación entre LabVIEW y el ESP32. Esto se logra identificando la práctica en ejecución y activando o desactivando las entradas y salidas mediante la programación en la tarjeta de desarrollo.
- d. Un botón de control booleano, llamado **DESCONECTAR**, se muestra cuando el programa ha comenzado su ejecución. Este botón finaliza la comunicación entre LabVIEW y la PLCP al desactivar la ejecución de la programación en el ESP32.
- e. Una paleta de selección llamada **PUERTO** que nos permite elegir el puerto de comunicación al que está conectada la PLCP.
- f. El botón de **STOP** detiene el programa. Es importante detener el programa utilizando este botón y no el botón de abortar de la barra de herramientas porque así se asegura que al parar el programa se detenga correctamente el motor DC y la comunicación.

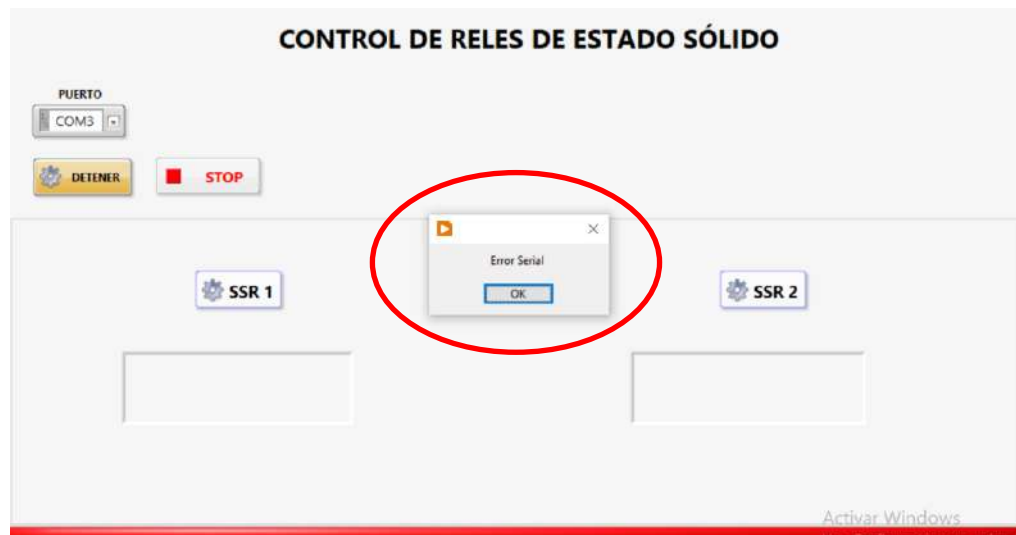
En la *Ilustración 4-25* se muestran los controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de activación de SSRs.



**Ilustración 4-25:** Controles booleanos y paleta de selección en panel frontal de la práctica de control de velocidad manual

### 3. Correr el programa

- a. Para seleccionar el puerto de comunicación al que está conectada la PLCP, utilizamos la paleta de selección **PUERTO**. En este caso, el puerto es COM3. Si no aparece en la lista, podemos hacer clic en el botón Refresh dentro de la misma paleta para actualizar la lista de puertos y elegir el puerto correcto.
- b. Al hacer clic en el botón **RUN** en la barra de herramientas de LabVIEW, iniciamos la ejecución del programa y establecemos la comunicación con el PLC. Si no se establece la comunicación con éxito, se generará un mensaje de Error Serial y aparecerá una barra de alerta de color rojo en la parte inferior del panel frontal, como se ilustra en la *Ilustración 4-26*. Esto indica que no se ha logrado establecer la comunicación con el puerto seleccionado.



**Ilustración 4-26:** Mensaje de error en la comunicación serial en la práctica de control de activación de SSRs

**Realizado por:** Cevallos, Daniel, 2023.

- c. Pulsamos el botón **CONECTAR** el cual inicia la comunicación entre LabVIEW y la PLCP. Identificando la práctica en ejecución, activando o desactivando las entradas y salidas mediante la programación dentro de la tarjeta de desarrollo.
- d. Por medio de los botones SSR1 Y SSR2 seleccionamos que relé activar, podemos hacerlo uno por uno como es este caso. Cuando los relés se activen, un indicador tipo booleano se tornará de color amarillo indicándonos su estado activo como se aprecia en la *Ilustración 4-27*.



**Ilustración 4-27:** Selección y activación de relé SSR

**Realizado por:** Cevallos, Daniel, 2023

4. Abrir el diagrama de bloques para analizar el algoritmo

Dirigirse al ANEXO I para observar la programación en diagramas de bloque.

5. Cerrar el programa sin guardar cambios

#### 4.8.4. Practica 4

**Tema:** Control de temperatura manual

**Objetivo:** Realizar el control de temperatura de forma manual

#### **Marco teórico**

En esta práctica, se realizará el control manual del encendido de una luz LED de alta potencia desde el computador. La luz LED actuará como una fuente de calor y, a través de un sensor de temperatura integrado en este, será posible conocer el valor de temperatura. El encendido de la luz LED es controlado por una señal PWM que puede variar entre 0 y 255.

Es posible emplear los programas proporcionados, no obstante, se recomienda crear los programas desde cero con el propósito de lograr una comprensión más profunda de su funcionamiento, enriquecer la experiencia de desarrollo y fomentar la generación de ideas para mejorarlos e incorporar nuevas funcionalidades.

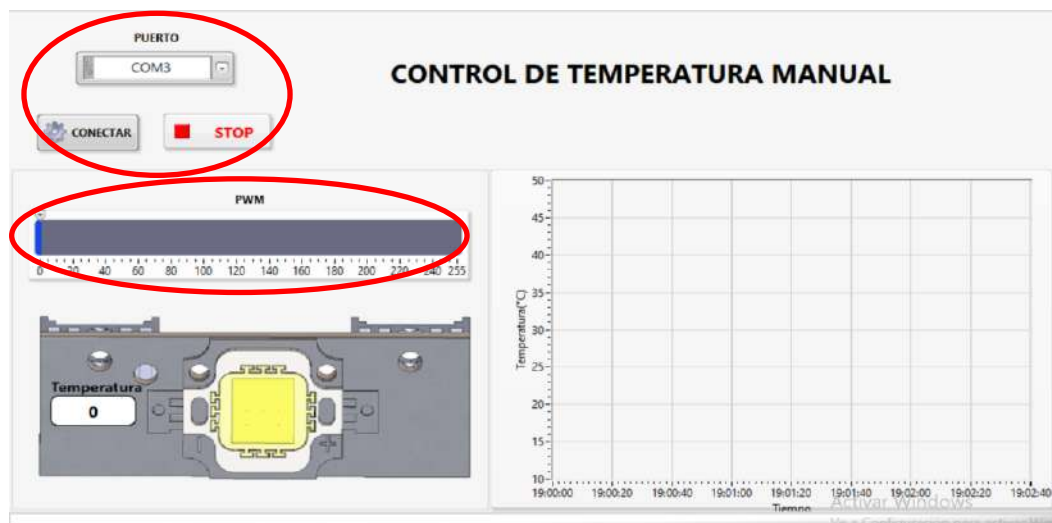
#### **Desarrollo**

1. Realizar las siguientes conexiones entre la PLCP y el computador:
  - a. Conectamos la fuente de alimentación DC de 12 V a la PLCP y el cable de comunicación micro USB como se muestra en la *Ilustración 4-15*.
  - b. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
2. Abrir el programa CONTROL MANUAL DE TEMPERATURA.vi. El panel frontal presenta los siguientes objetos
  - a. Un control numérico tipo Slide **PWM** para especificar el valor de la señal entregada hacia la luz LED. Incluye un control tipo numérico para colocar valores exactos.



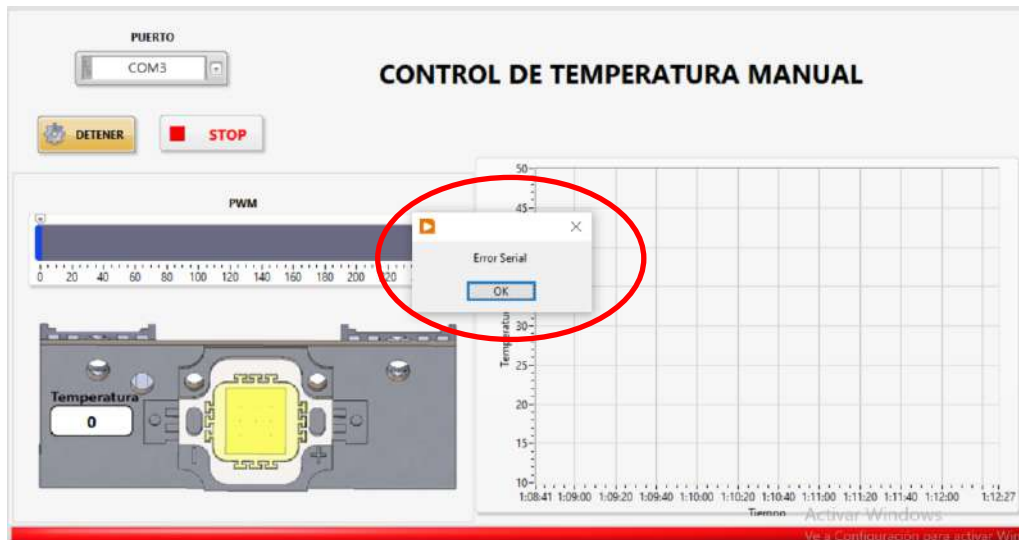
- b. Un botón de control booleano denominado **CONECTAR** permite iniciar la comunicación entre LabVIEW y el ESP32. Esto se logra identificando la práctica en ejecución y activando o desactivando las entradas y salidas mediante la programación en la tarjeta de desarrollo.
- c. Un botón de control booleano, llamado **DESCONECTAR**, se muestra cuando el programa ha comenzado su ejecución. Este botón finaliza la comunicación entre LabVIEW y la PLCP al desactivar la ejecución de la programación en el ESP32.
- d. Una paleta de selección llamada **PUERTO** que nos permite elegir el puerto de comunicación al que está conectada la PLCP.
- e. El botón de **STOP** detiene el programa. Es importante detener el programa utilizando este botón y no el botón de abortar de la barra de herramientas porque así se asegura que al parar el programa se detenga correctamente el motor DC y la comunicación.

En la *Ilustración 4-28* se muestran los controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de temperatura manual.



**Ilustración 4-28:** Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control de temperatura manual

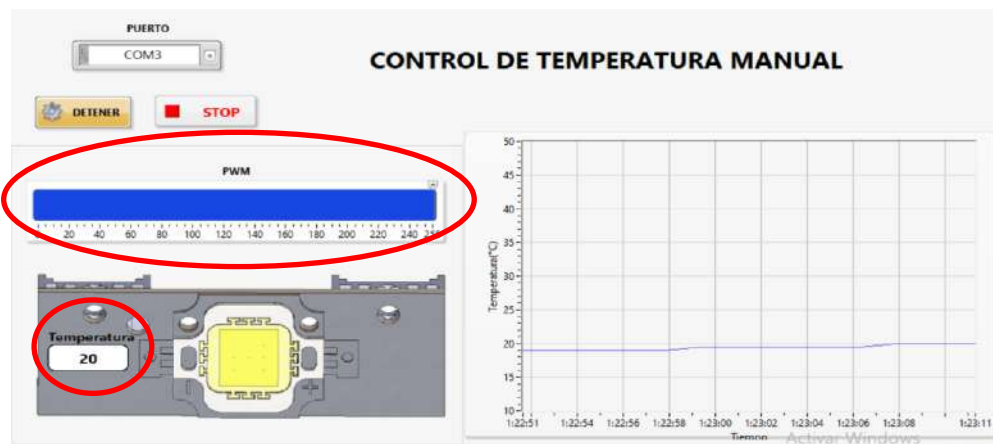
Realizado por: Cevallos, Daniel, 2023.



**Ilustración 4-29:** Mensaje de error en la comunicación serial en la práctica de control de temperatura manual

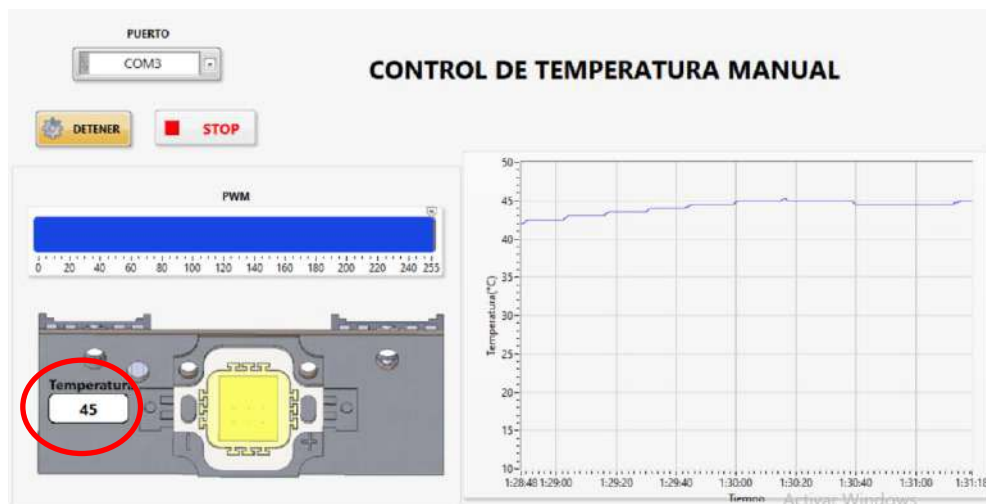
**Realizado por:** Cevallos, Daniel, 2023.

- a. Pulsamos el botón **CONECTAR** el cual inicia la comunicación entre LabVIEW y la PLCP. Identificando la práctica en ejecución, activando o desactivando las entradas y salidas mediante la programación dentro de la tarjeta de desarrollo.
- b. Seleccionar un valor de PWM entre (0-255) que para este es un valor de 255 y una temperatura inicial de 20° como se puede observar en la *Ilustración 4-30*, generando un valor de temperatura que aumenta conforme pasa el tiempo como se muestra en la *Ilustración 4-31* pudiendo llegar hasta un valor de 50°C como límite. Mientras más alto sea el valor de PWM, la temperatura en disipador aumentara rápidamente caso contrario, si el valor es menor la temperatura ira aumentando lentamente.



**Ilustración 4-30:** Ingreso del valor PWM en el control numérico

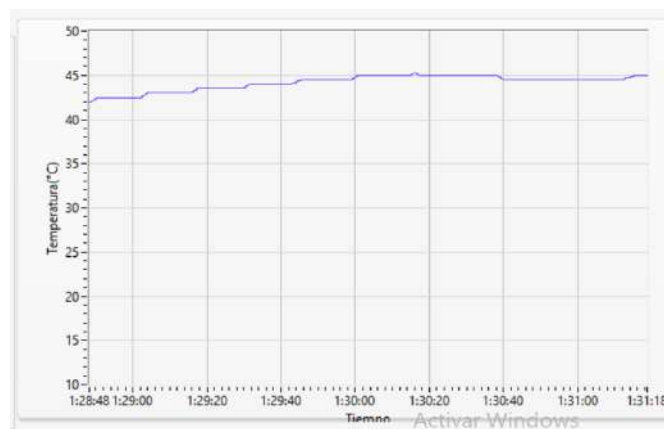
**Realizado por:** Cevallos, Daniel, 2023



**Ilustración 4-31:** Incremento en el valor de la temperatura en función al valor de PWM

Realizado por: Cevallos, Daniel, 2023.

La simulación de la práctica de control de temperatura manual nos genera una gráfica (Temperatura vs tiempo) en la que podemos observar el comportamiento de la temperatura ante el estímulo de la señal PWM como se indica en la *Ilustración 4-32*.



**Ilustración 4-32:** Gráfica Temperatura (°C) vs Tiempo

Realizado por: Cevallos, Daniel, 2023.

3. Abrir el diagrama de bloques para analizar el algoritmo

Dirigirse al ANEXO I para observar la programación en diagramas de bloque.

4. Cerrar el programa sin guardar cambios

#### 4.8.5. Practica 5

**Tema:** Control PID de temperatura

**Objetivo:** Realizar el control PID de temperatura mediante (PWM)

#### **Marco teórico**

En esta práctica se mejorará el control de temperatura utilizando un controlador PID. Puesto que el bombillo halógeno tiene una operación tipo On/Off, El PWM será la señal de control de la luz LED de potencia.

La selección de los valores de las ganancias PID se puede obtener por varios métodos experimentales y analíticos. En este caso se trabajarán con diferentes valores de ganancias de  $K_p$ ,  $T_i$  y  $T_d$ , observando el comportamiento de la variable controlada Temperatura.

Los resultados observados en la práctica pueden variar con los expuestos en este manual. Estas variaciones se deberían a varios factores como tiempo de vida de la luz LED, contacto con el sensor, temperatura ambiente, y perturbaciones como ráfagas de aire.

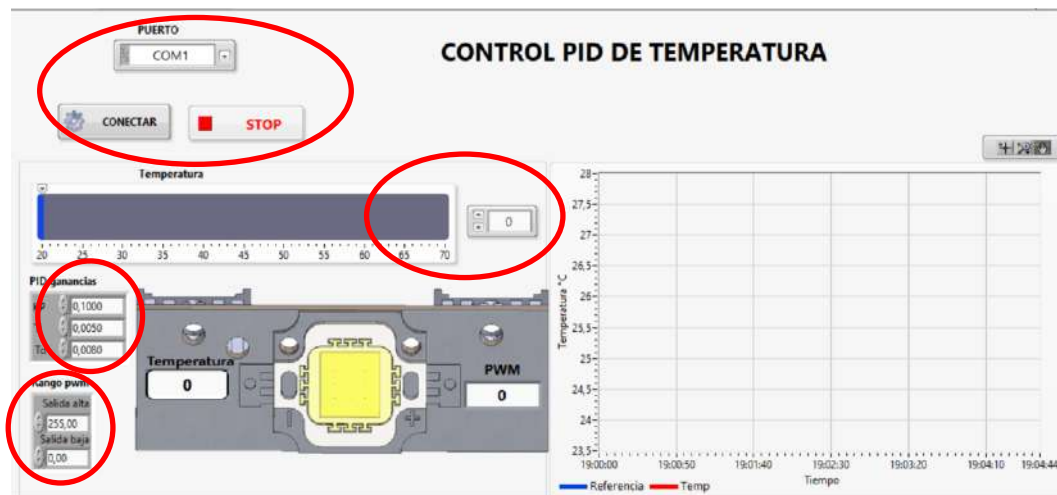
Estas prácticas se llevan a cabo con el ventilador apagado, y se activará posteriormente para ajustar la temperatura de acuerdo al valor de referencia ingresado en el programa, con el objetivo de disminuir la temperatura.

#### **Desarrollo**

1. Realizar las siguientes conexiones entre la PLCP y el computador:
  - a. Conectamos la fuente de alimentación DC de 12 V a la PLCP y el cable de comunicación micro USB como se muestra en la *Ilustración 4-15*.
  - b. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
2. Abrir el programa CONTROL PID DE TEMPERATURA.vi. El panel frontal presenta los siguientes objetos.

- a. Un control numérico tipo Slide **Temperatura** para especificar el valor de temperatura de referencia (Setpoint). Incluye un control tipo numérico para colocar valores exactos.
- b. Un botón de control booleano denominado **CONECTAR** permite iniciar la comunicación entre LabVIEW y el ESP32. Esto se logra identificando la práctica en ejecución y activando o desactivando las entradas y salidas mediante la programación en la tarjeta de desarrollo.
- c. Un botón de control booleano, llamado **DESCONECTAR**, se muestra cuando el programa ha comenzado su ejecución. Este botón finaliza la comunicación entre LabVIEW y la PLCP al desactivar la ejecución de la programación en el ESP32.
- d. Una paleta de selección llamada **PUERTO** que nos permite elegir el puerto de comunicación al que está conectada la PLCP.
- e. El botón **STOP** detiene el programa. Es importante detener el programa utilizando este botón y no el botón de abortar de la barra de herramientas porque así se asegura que al parar el programa se detenga correctamente el motor DC y la comunicación.
- f. Una paleta de selección llamada **PID ganancias** que nos permite ingresar el valor de las ganancias  $K_p$ ,  $T_i$  y  $T_d$  al sistema.
- g. Una paleta de selección llamada **Rango pwm** nos permite ingresar el valor de la salida alta y baja de PWM.

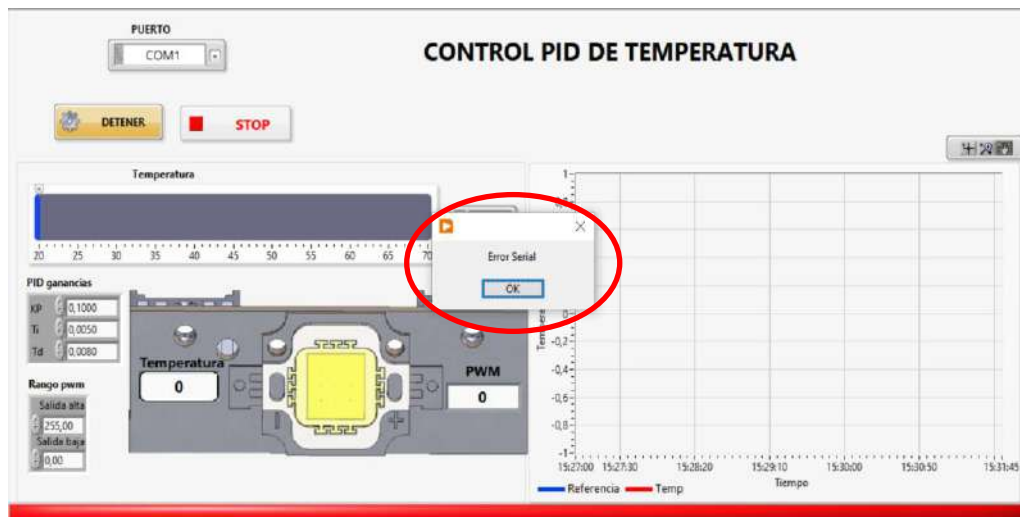
En la *Ilustración 4-33* se muestran los controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control PID de temperatura.



**Ilustración 4-33:** Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control PID de temperatura

**Realizado por:** Cevallos, Daniel, 2023.

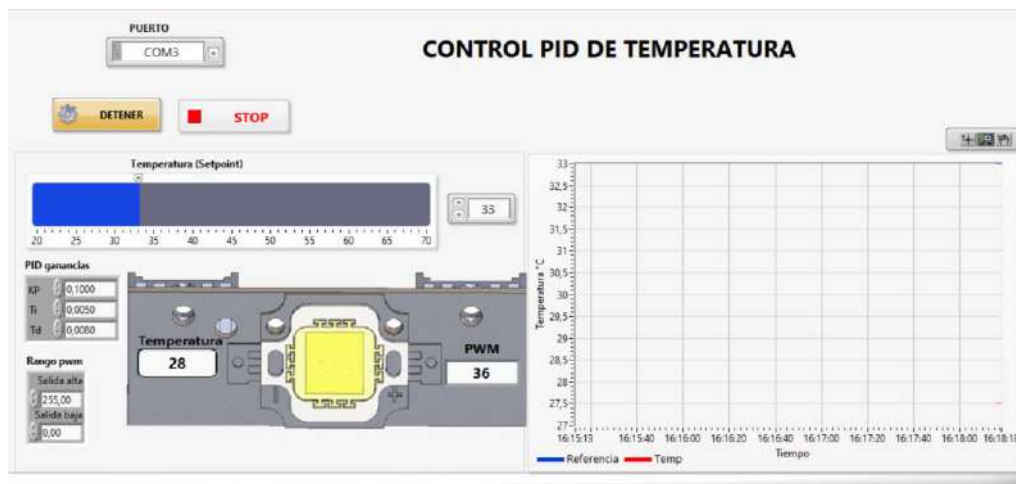
3. Correr el programa
  - a. Para seleccionar el puerto de comunicación al que está conectada la PLCP, utilizamos la paleta de selección **PUERTO**. En este caso, el puerto es COM3. Si no aparece en la lista, podemos hacer clic en el botón Refresh dentro de la misma paleta para actualizar la lista de puertos y elegir el puerto correcto.
  - b. Al hacer clic en el botón **RUN** en la barra de herramientas de LabVIEW, iniciamos la ejecución del programa y establecemos la comunicación con el PLC. Si no se establece la comunicación con éxito, se generará un mensaje de Error Serial y aparecerá una barra de alerta de color rojo en la parte inferior del panel frontal, como se ilustra en la *Ilustración 4-34*. Esto indica que no se ha logrado establecer la comunicación con el puerto seleccionado.



**Ilustración 4-34:** Mensaje de error en la comunicación serial en la práctica de control PID temperatura

**Realizado por:** Cevallos, Daniel, 2023.

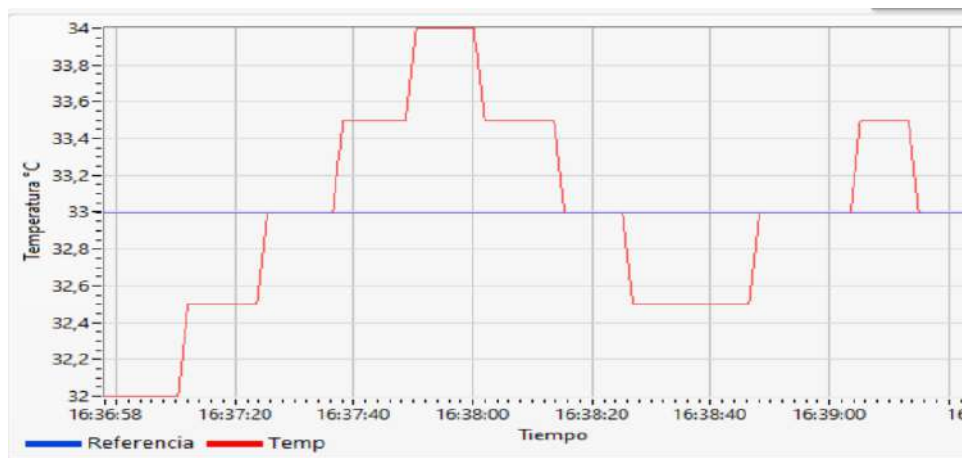
- c. Seleccionar un valor de temperatura de referencia entre (20-70) °C que para este caso 30°C, además ingresamos un valor de ganancias PID iguales a  $K_p=0,1000$ ,  $T_i=0,0050$  y  $T_d=0,0050$  donde estas constantes fueron encontradas de manera experimental.
- d. Pulsamos el botón **CONECTAR** el cual inicia la comunicación entre LabVIEW y la PLCP. Identificando la práctica en ejecución, activando o desactivando las entradas y salidas mediante la programación dentro de la tarjeta de desarrollo.
- e. En la *Ilustración 4-35* se aprecia que el proceso empieza su control en una temperatura de 28°C a la que se encuentra el disipador de calor de la luz de potencia, elevándose hasta el valor de referencia de 33°C.
- f. En un sistema PID se aplica una ganancia al error, consistente en la diferencia entre el valor deseado **Temperatura (Setpoint)** y el valor de la variable (en este caso, la **Temperatura**). Se puede observar la gráfica de la salida del PID en la *Ilustración 4-36*.
- g. Para esta práctica de control PID de temperatura el método utilizado para la sintonización del control PID es la autoafinación o (Autotuning). Esta herramienta nos permite identificar automáticamente los parámetros óptimos del controlador PID sin la necesidad de una intervención manual extensa. Esta funcionalidad es particularmente útil en aplicaciones de control donde el sistema es complejo o donde es difícil determinar manualmente los valores óptimos de los parámetros del controlador.



**Ilustración 4-35:** Valor de Setpoint ingresado y de temperatura actual

Realizado por: Cevallos, Daniel, 2023.

- h. La señal de temperatura controlada por la acción del control PID se la pueda observar en la *Ilustración 4-36*, la cual nos muestra una gráfica de Temperatura vs Tiempo de la señal, produciendo una respuesta amortiguada.



**Ilustración 4-36:** Gráfica de Temperatura vs Tiempo de la señal de temperatura ante la acción del controlador PID sintonizado mediante Autotuning de LabVIEW

Realizado por: Cevallos, Daniel, 2023.

De esta forma se han desarrollado la práctica de control PID de temperatura para sistemas térmicos. El equipo puede también utilizarse conjuntamente con el Toolkit de Identificación de Sistemas de LabVIEW para modelar matemáticamente el sistema térmico.

4. Abrir el diagrama de bloques para analizar el algoritmo



Dirigirse al ANEXO I para observar la programación en diagramas de bloque.

5. Cerrar el programa sin guardar cambios

#### 4.8.6. Practica 6

**Tema:** Control PID de velocidad del motor DC

**Objetivo:** Realizar el control PID de velocidad del motor DC mediante (PWM)

#### Marco teórico

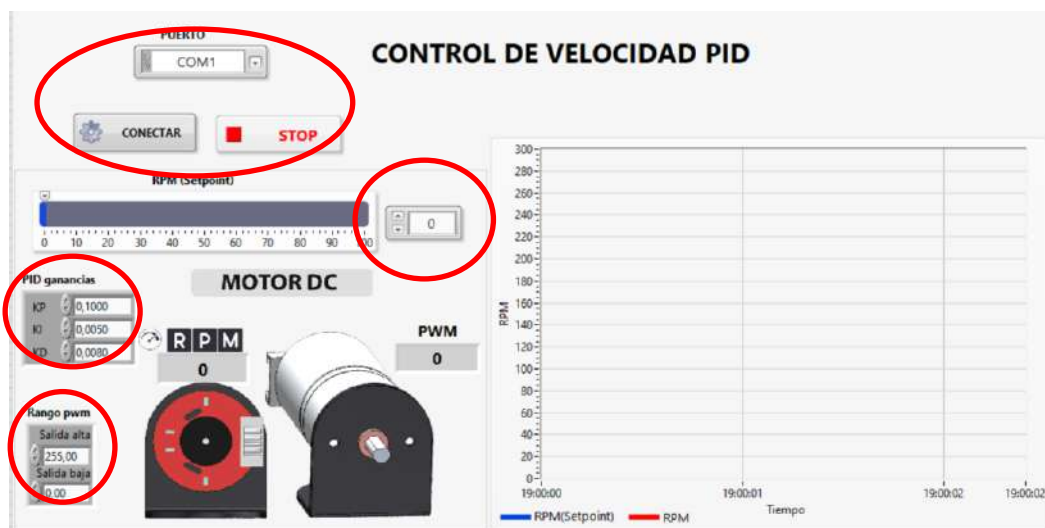
En numerosos sistemas en los que se requiere regular una variable, resulta imposible establecer un modelo matemático de la planta debido a la carencia de información detallada. Esto se ilustra en el ejemplo del Motor DC de la PLCP, ya que no existen especificaciones proporcionadas por el fabricante respecto a la resistencia de armadura, inductancia, amortiguamiento por fricción viscosa, inercia del eje, entre otros parámetros. Por lo tanto, se hace necesario emplear enfoques experimentales para adquirir un conjunto de valores óptimos para las ganancias del controlador PID.

#### Desarrollo

1. Realizar las siguientes conexiones entre la PLCP y el computador:
  - c. Conectamos la fuente de alimentación DC de 12 V a la PLCP y el cable de comunicación micro USB como se muestra en la *Ilustración 4-15*.
  - d. Conectamos el cable micro USB con el computador el cual nos permitirá establecer la comunicación serial UART.
2. Abrir el programa CONTROL PID DE VELOCIDAD.vi. El panel frontal presenta los siguientes objetos.
  - h. Un control numérico tipo Slide **RPM (Setpoint)** para especificar el valor de RPM de referencia. Incluye un control tipo numérico para colocar valores exactos.
  - i. Un botón de control booleano denominado **CONECTAR** permite iniciar la comunicación entre LabVIEW y el ESP32. Esto se logra identificando la práctica en ejecución y activando o desactivando las entradas y salidas mediante la programación en la tarjeta de desarrollo.

- j. Un botón de control booleano, llamado **DESCONECTAR**, se muestra cuando el programa ha comenzado su ejecución. Este botón finaliza la comunicación entre LabVIEW y la PLCP al desactivar la ejecución de la programación en el ESP32.
- k. Una paleta de selección llamada **PUERTO** que nos permite elegir el puerto de comunicación al que está conectada la PLCP.
- l. El botón **STOP** detiene el programa. Es importante detener el programa utilizando este botón y no el botón de abortar de la barra de herramientas porque así se asegura que al parar el programa se detenga correctamente el motor DC y la comunicación.
- m. Una paleta de selección llamada **PID ganancias** que nos permite ingresar el valor de las ganancias  $K_p$ ,  $T_i$  y  $T_d$  al sistema.
- n. Una paleta de selección llamada **Rango pwm** nos permite ingresar el valor de la salida alta y baja de PWM.

En la *Ilustración 4-37* se muestran los controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control PID de velocidad.



**Ilustración 4-37:** Controles numéricos, booleanos y paleta de selección en panel frontal de la práctica de control PID de velocidad

Realizado por: Cevallos, Daniel, 2023.

- 3. Correr el programa

- a. Para seleccionar el puerto de comunicación al que está conectada la PLCP, utilizamos la paleta de selección **PUERTO**. En este caso, el puerto es COM3. Si no aparece en la lista, podemos hacer clic en el botón Refresh dentro de la misma paleta para actualizar la lista de puertos y elegir el puerto correcto.
- b. Al hacer clic en el botón **RUN** en la barra de herramientas de LabVIEW, iniciamos la ejecución del programa y establecemos la comunicación con el PLC. Si no se establece la comunicación con éxito, se generará un mensaje de Error Serial y aparecerá una barra de alerta de color rojo en la parte inferior del panel frontal, como se ilustra en la Ilustración 38-4. Esto indica que no se ha logrado establecer la comunicación con el puerto seleccionado.



**Ilustración 4-38:** Mensaje de error en la comunicación serial en la práctica de control PID de velocidad

**Realizado por:** Cevallos, Daniel, 2023.

- c. S En este caso, hemos elegido un valor de RPM de referencia (Setpoint) de 100 RPM. Además, hemos definido las ganancias del controlador PID como sigue:  $K_p=0.2000$ ,  $T_i=0.0050$ , y  $T_d=0.0010$ . Estas constantes fueron determinadas a través de experimentación.
- d. Pulsamos el botón **CONECTAR** el cual inicia la comunicación entre LabVIEW y la PLCP. Identificando la práctica en ejecución, activando o desactivando las entradas y salidas mediante la programación dentro de la tarjeta de desarrollo.
- e. En la *Ilustración 4-39* se aprecia el proceso de control de velocidad, que inicia en un valor inicial de 0 RPM y elevándose hasta el valor de referencia de 100 RPM. A

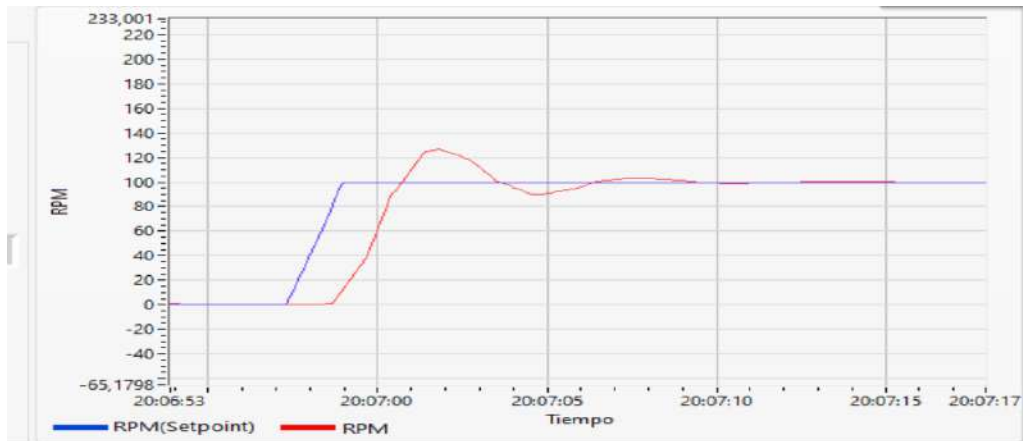
- f. En un sistema PID se aplica una ganancia al error, consistente en la diferencia entre el valor deseado **RPM (Setpoint)** y el valor de la variable (en este caso, la **RPM**). Se puede observar la gráfica de la salida del PID en la *Ilustración 4-40*.
- g. Para esta práctica de control PID de temperatura el método utilizado para la sintonización del control PID es la autoafinación (Autotuning). Esta herramienta nos permite identificar automáticamente los parámetros óptimos del controlador PID sin la necesidad de una intervención manual extensa. Esta funcionalidad es particularmente útil en aplicaciones de control donde el sistema es complejo o donde es difícil determinar manualmente los valores óptimos de los parámetros del controlador.



**Ilustración 4-39:** Valor de Setpoint ingresado y velocidad actual

**Realizado por:** Cevallos, Daniel, 2023.

- h. La señal de temperatura controlada por la acción del control PID se la pueda observar en la *Ilustración 4-40*, la cual nos muestra una gráfica de Temperatura vs Tiempo de la señal, produciendo una respuesta amortiguada.



**Ilustración 4-40:** Gráfica de velocidad (RPM) vs Tiempo de la señal de temperatura ante la acción del controlador PID sintonizado mediante Autotuning de LabVIEW

**Realizado por:** Cevallos, Daniel, 2023.

Con el propósito de inducir perturbaciones en el sistema, introducimos un objeto desde un lado de la PLCP como se visualiza en la *Ilustración 4-41*. Lo que provoca un frenado forzado en el eje y, como consecuencia, una disminución de la velocidad.



**Ilustración 4-41:** Frenado de eje del motor para generar perturbación en el proceso de velocidad

**Realizado por:** Cevallos, Daniel, 2023.

## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

#### 5.1. Conclusiones

- Se desarrolló y se diseñó una planta de laboratorio para el control de procesos térmicos, de velocidad, posición y desplazamiento angular utilizando controladores de 32 bits basándonos en una exhaustiva evaluación y selección de componentes clave, como sensores, drivers y actuadores. La elección adecuada del microcontrolador desempeña un papel fundamental en el funcionamiento óptimo del sistema. Además, la integración de una comunicación serial por protocolo UART garantiza la interconexión efectiva entre los diferentes elementos del sistema.
- A lo largo del desarrollo de este trabajo de integración curricular, hemos explorado detenidamente las metodologías empleadas en trabajos previos, analizando tanto sus alcances como los requerimientos que han guiado la ejecución de los mismos. Al hacerlo, obtuvimos una visión más amplia de cómo aplicar toda esta información para el diseño de hardware y software, para así desarrollar una planta de laboratorio de control de procesos eficiente y de alto rendimiento, capaz de cumplir con sus objetivos de manera precisa y confiable.
- Se estableció una comunicación bidireccional establecida a en una velocidad de 9600 baudios entre el ESP32 y LabVIEW debido a que este entorno de control y monitoreo es muy efectivo para una variedad de aplicaciones. Al establecer una conexión fluida y confiable, pudimos aprovechar las capacidades del ESP32 junto con la interfaz gráfica de LabVIEW para crear cada uno de los programas que corresponden a las prácticas que van desde el control de posición angular hasta el control PID de temperatura.
- Se determinó que el error relativo ( $E_r$ ) máximo dado por el sensor de temperatura DS18B20 es de 2.44%, por consiguiente, la recolección de información ejecutada por este sensor conduce a un criterio de calidad aceptable según los resultados experimentales.
- Tras realizar una prueba de repetibilidad en la etapa de comunicación entre la etapa de adquisición, de comunicación y módulo de control y visualización establecida a una velocidad de transmisión de dato de 9600 baudios se obtiene un coeficiente de variación  $CV=0\%$  lo que nos indica que esta etapa también se ejecuta de manera confiable y de calidad.

- Después de llevar a cabo una nueva prueba de repetibilidad en el funcionamiento de las prácticas a realizarse con la planta de laboratorio de control de procesos se obtuvo un  $CV=0\%$  significando estabilidad de datos indicándonos que cada práctica se ejecuta con calidad y de manera confiable.
- En lo que respecta a la alimentación, se verificó que la PLCP tiene un consumo energético real de 1876.12 mA en su estado activo máximo y de 266.63 mA en su estado activo mínimo. Estas variaciones se deben a las diferencias en el consumo energético de los diversos actuadores utilizados en cada práctica.
- Una vez habiendo realizado el análisis económico necesario para la construcción de la PLCP se determinó un costo total de \$ 215.16 siendo más económico que dispositivos comerciales de similares características, ya que solamente una DAQ 6008 de National Instruments tiene un costo aproximado de 350\$.
- La planta de laboratorio para el control de procesos diseñada en el presente trabajo de integración curricular podría ser usada como herramienta de aprendizaje en instituciones de educación superior donde se vean involucradas las asignaturas de control de procesos e instrumentación virtual debido a las características que presenta.



## 5.2. Recomendaciones

- En cuanto a Hardware, se pueden desarrollar e implementar más plantas de control en trabajos de integración futuros relacionados a este tema y así poder aprovechar todos los terminales y capacidades con las que cuenta el ESP32, y además utilizar los demás canales de comunicación con los que cuenta este microcontrolador como los son Wifi y Bluetooth, lo que permitiría la conexión a redes locales o remotas. Esto facilitaría el control y monitoreo a distancia, así como la interconexión con otros dispositivos y sistemas.
- Debemos tomar en cuenta el tiempo para la realización del proyecto de integración curricular, debido a que el momento de realizar las pruebas de funcionamiento siempre aparecen fallas o errores que deben ser corregidos a tiempo, ya que algunos errores pueden llevarnos días encontrarlos y eso dificultaría entregar nuestro trabajo en las fechas establecidas.
- Considerando que nuestra PLCP contiene los procesos principales de control, con este proyecto tratamos de fomentar al usuario el interés de adquirir mayores conocimientos en el área de sistemas de control motivándolo a crear nuevos diseños y la creación de nuevas plantas de control con diferentes actuadores.
- Utilizar equipos patrones de mejor precisión para la toma de datos de los elementos que conforman la etapa de adquisición y la etapa de actuadores ya que basados en estos datos podemos realizar en análisis estadístico de calidad.
- Las practicas deben ser efectuadas exactamente según las instrucciones y los parámetros exactos indicados en los capítulos anteriores, ya que si ingresamos un parámetro de manera errónea podemos presentar errores en la ejecución de cada programa.
- Es fundamental llevar a cabo las prácticas de acuerdo a las instrucciones y los parámetros precisos especificados en el manual de prácticas correspondiente. Cualquier introducción de valores incorrectos o un manejo inadecuado del equipo puede resultar en dificultades durante la ejecución.
- Es imprescindible leer el manual de usuario antes de utilizar la PLCP, ya que cada proceso presenta características y modos de funcionamiento distintos propios de cada práctica, evitando así futuros daños en sus componentes.

## BIBLIOGRAFÍA

**ALCACIEGA SIZA, C.O.; & SIMBA LEMA, D.A.** Diseño de un módulo didáctico para sistemas de control, en el laboratorio de Ingeniería Electromecánica de la Universidad Técnica de Cotopaxi [en línea] (Trabajo de titulación) (Pregrado). Universidad Técnica de Cotopaxi, Facultad de Ciencias de la Ingeniería y Aplicadas, Latacunga, Ecuador. 2020. pp. 1-168. [Consulta: 10 octubre 2023]. Disponible en: <http://repositorio.utc.edu.ec/handle/27000/6770>

**ALTAMIRANO, E.; & MÉNDEZ, P.** Diseño y construcción de módulos didácticos para el laboratorio virtual de instrumentación industrial, de la carrera de Ingeniería en Mantenimiento Eléctrico [en línea] (Trabajo de titulación) (Pregrado). Universidad Técnica del Norte, Facultad de Ingeniería en Ciencias Aplicadas, Ibarra, Ecuador. 2016. pp. 1-117. [Consulta: 8 mayo 2023]. Disponible en: <http://repositorio.utn.edu.ec/handle/123456789/7602>

**AMAZON.** Heyiarbeit Caja de conexiones de plástico ABS a prueba de polvo impermeable IP65 Caja de recinto eléctrico universal para proyecto gris con cubierta transparente de policarbonato 7.87 x 4.72 x 2.95 pulgadas (7.874 x 4.724 x 2.953 in) : Herramientas y Mejoras del Hogar. *Amazon.com* [en línea]. 2023. [Consulta: 14 agosto 2023]. Disponible en: [https://www.amazon.com/dp/B09B72TR3L?ref=ppx\\_pop\\_mob\\_ap\\_share&th=1](https://www.amazon.com/dp/B09B72TR3L?ref=ppx_pop_mob_ap_share&th=1)

**ARDUINO.CL.** Cómo configurar la comunicación UART en Arduino. [en línea]. 2022. [Consulta: 13 julio 2023]. Disponible en: <https://arduino.cl/como-configurar-la-comunicacion-uart-en-arduino/>

**BRUNETE, A.; et al.** *Introducción a la Automatización Industrial* [en línea]. Madrid-España: s.n. 2020a. [Consulta: 12 junio 2023]. Disponible en: [https://bookdown.org/alberto\\_brunete/intro\\_automatica/sistemas-de-control.html](https://bookdown.org/alberto_brunete/intro_automatica/sistemas-de-control.html)

**BRUNETE, A.; et al.** *Introducción a la Automatización Industrial* [en línea]. Madrid-España: s.n. 2020b. [Consulta: 12 junio 2023]. Disponible en: [https://bookdown.org/alberto\\_brunete/intro\\_automatica/sistemas-de-control.html](https://bookdown.org/alberto_brunete/intro_automatica/sistemas-de-control.html)

**CARROD ELECTRÓNICA.** Módulo Sensor Detector de Velocidad para Motor Tipo Herradura Breakout H2010. *Carrod* [en línea]. 2018. [Consulta: 9 agosto 2023]. Disponible en: <https://www.carrod.mx/products/optoranura-h21b1-switch-optico-salida-darlington-1>

**CASTAÑO, S.** Comunicación Serial con Arduino. *Control Automático Educación* [en línea]. 2019. [Consulta: 13 julio 2023]. Disponible en: <https://controlautomaticoeducacion.com/arduino/comunicacion-serial-con-arduino/>

**CASTRO, C.D.P.; & OSORIO, S.R.** “Tarjetas de Desarrollo: Herramientas para el diseño”. *Letras Con Ciencia Tecnológica* [en línea], 2011, (11), pp. 43-52. [Consulta: 16 julio 2023]. ISSN 2665-2544. Disponible en: <https://revistas.itc.edu.co/index.php/letras/article/view/104>

**CEISA.** *Control e Instrumentación Industrial: Elementos de un sistema de control* [blog]. 11 de enero 2016. [Consulta: 12 julio 2023]. Disponible en: <http://ceisa.blogspot.com/2015/01/elementos-de-un-sistema-de-control.html>

**CLR.** *Motor paso a paso: cuándo y por qué utilizarlo* [blog]. Junio 2016. [Consulta: 29 julio 2023]. Disponible en: <https://clr.es/blog/es/motor-paso-a-paso-cuando-utilizarlo/>

**COMPONENTS101.** Mini MP1584 DC-DC 3A Adjustable Buck Module. *Components101* [en línea]. 2023. [Consulta: 30 mayo 2023]. Disponible en: <https://components101.com/modules/mini-mp1584-dc-dc-3a-adjustable-buck-module>

**CORONA, L.; et al.** *Sensores y Actuadores* [en línea]. Azcapotzalco, México D. F.: GRUPO EDITORIAL PATRIA. 2014. ISBN 978-607-438-936-4. Disponible en: <https://media.oiiipdf.com/pdf/6cc029d2-741d-417b-ae90-8b7f3965b41f.pdf>

**DESIGNAND MECHATRONICS,.** *DAQ 6009 Labview Entradas análogas y Digitales* [blog]. 27 de abril 2013. [Consulta: 13 octubre 2023]. Disponible en: <http://designandmechatronics.blogspot.com/2013/03/daq-6009-labview-entradas-analogas-y.html>

**DIRECT INDUSTRY. EAGLE AUTODESK.** *Direct Industry* [en línea]. 2023. [Consulta: 12 junio 2023]. Disponible en: <https://www.directindustry.es/prod/autodesk/product-14521-1911712.html>

**ELECTRONILAB.** Motor Paso A Paso Nema 17 - 36.8 oz.in - 200 Pasos/vuelta. *Electronilab.co* [en línea]. 2020. [Consulta: 29 julio 2023]. Disponible en: <https://electronilab.co/tienda/motor-paso-a-paso-nema-17-25-oz-in-200-pasosvuelta/>

**GRUPO ELECTROSTORE.** MOTOR REDUCTOR 25D 12V DC 77RPM EJE 4MM 25GA370. [en línea]. 2019<sup>a</sup>. [Consulta: 29 julio 2023]. Disponible en: <https://grupoelectrostore.com/shop/motores/motor-reductor-25d-12v-dc-77rpm-eje-4mm-25ga370/>.

**GRUPO ELECTROSTORE.** MOTOR REDUCTOR 25D 12V DC 280RPM EJE 4MM ENCODER JGA25-370. [en línea]. 2019<sup>b</sup> [Consulta: 29 julio 2023]. Disponible en: <https://grupoelectrostore.com/shop/motores/motor-reductor-25d-12v-dc-280rpm-eje-4mm-encoder-jga25-370/>.

**HARMONIC DRIVE SE.** Motor DC. Harmonic Drive SE [en línea]. 2023. [Consulta: 28 julio 2023]. Disponible en: <https://harmonicdrive.de/es/glosario/motor-dc>

**HERNÁNDEZ, R.** *Introducción a los sistemas de control Conceptos, aplicaciones y simulación con MATLAB - Ricardo Hernández Gaviño* [en línea]. Naucalpan de Juárez-México: PEARSON EDUCACIÓN. 2010. [Consulta: 5 julio 2023]. Disponible en: [https://www.academia.edu/10902090/Introducci%C3%B3n\\_a\\_los\\_sistemas\\_de\\_control\\_Conceptos\\_aplicaciones\\_y\\_simulaci%C3%B3n\\_con\\_MATLAB\\_Ricardo\\_Hern%C3%A1ndez\\_Gavi%C3%B3n](https://www.academia.edu/10902090/Introducci%C3%B3n_a_los_sistemas_de_control_Conceptos_aplicaciones_y_simulaci%C3%B3n_con_MATLAB_Ricardo_Hern%C3%A1ndez_Gavi%C3%B3n).

**HETPRO.** I2C - Puerto, Introducción, trama y protocolo [en línea]. 2017. [Consulta: 13 julio 2023]. Disponible en: <https://hetpro-store.com/TUTORIALES/i2c/>

**HIGH LIGHTS.** *ENTRENADOR DE PLANTA DE CONTROL* [en línea]. 2009. Cuenca-Ecuador: s.n. Disponible en: <https://silو.tips/download/entrenador-de-planta-de-control-epc>.

**IONOS.** Programación visual: la entrada más sencilla al mundo digital. *IONOS* [en línea]. [Consulta: 30 julio 2023]. 2020. Disponible en: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/programacion-visual/>.

**IZA, O.** Diseño e implementación de un módulo de instrumentación electrónica, enlazado con Labview mediante interfaz USB [en línea] (Trabajo de titulación) (Pregrado). Escuela Politécnica Nacional, Facultad de Ingeniería Eléctrica y Electrónica, Quito, Ecuador. 2015. pp. 1-148. [Consulta: 8 mayo 2023]. Disponible en: <http://bibdigital.epn.edu.ec/handle/15000/11061>

**LUCAS, J.** Qué es Raspberry Pi. *OpenWebinars.net* [en línea].2019, [Consulta: 16 julio 2023]. Disponible en: <https://openwebinars.net/blog/que-es-raspberry-pi/>.

**MACHICAO, M.C.** “Características de las placas Arduino”. *bit@bit* [en línea], 2017, 2(4). pp. 1-6. [Consulta: 14 julio 2023]. Disponible en: <http://dicyt.uajms.edu.bo/revistas/index.php/bitabit/article/view/801>

**MIER, L.** ESP32, ESP82 y ESP8226: Microcontroladores ESP. *BitCuco* [en línea]. 2020. [Consulta: 16 julio 2023]. Disponible en: <https://bitcu.co/esp32-microcontroladores/>

**MONTESINOS, O.; et al.** *Muestreo estadístico. Tamaño de muestra y estimación de parámetros* [en línea]. 2010. Colima-Colombia: Universidad de Colima. 2010. [Consulta: 11 julio 2023]. ISBN 978-607-7565-62-8. Disponible en: [https://www.researchgate.net/publication/343499141\\_Muestreo\\_estadistico\\_Tamano\\_de\\_muestra\\_y\\_estimacion\\_de\\_parametros](https://www.researchgate.net/publication/343499141_Muestreo_estadistico_Tamano_de_muestra_y_estimacion_de_parametros).

**NATIONAL INSTRUMENT.** Descargar LabVIEW. *NI* [en línea]. [Consulta: 13 octubre 2023]. 2023a. Disponible en: <https://www.ni.com/es/support/downloads/software-products/download.labview.html>

**NATIONAL INSTRUMENT.** ¿Qué es LabVIEW? *NI*[en línea]. 2023b. [Consulta: 13 junio 2023]. Disponible en: <https://www.ni.com/es-cr/shop/labview.html>

**NATIONAL INSTRUMENT.** Question about USB CDC with LabVIEW. *NI* [en línea]. 2023c. [Consulta: 13 octubre 2023]. Disponible en: <https://forums.ni.com/t5/LabVIEW/Question-about-USB-CDC-with-LabVIEW/td-p/4302824>.

**NAYLAMP MECHATRONICS.** DRV8825 Driver motor paso a paso. Naylamp Mechatronics - Perú [en línea]. 2021a. [Consulta: 30 mayo 2023]. Disponible en: <https://naylampmechatronics.com/driver-pap-stepper/126-driver-pap-drv8825.html>.

**NAYLAMP MECHATRONICS.** NodeMCU-32 38-pin ESP32 Wi-Fi. Naylamp Mechatronics - Perú [en línea]. 2021b. [Consulta: 30 mayo 2023]. Disponible en: <https://naylampmechatronics.com/espressif-esp/1011-nodemcu-32-38-pin-esp32-wifi.html>.

**NAYLAMP MECHATRONICS.** Fuente de alimentación AC/DC 12V 1A (adaptador). Naylamp Mechatronics-Perú [en línea]. 2023a. [Consulta: 2 agosto 2023]. Disponible en: <https://naylampmechatronics.com/fuentes-switching-ac-dc/831-fuente-de-alimentacion-dc-12v-1a.html>.

**NAYLAMP MECHATRONICS.** Sensor Encoder Infrarrojo FC-03. Naylamp Mechatronics - Perú [en línea]. 2023b. [Consulta: 19 julio 2023]. Disponible en: <https://naylampmechatronics.com/sensores-proximidad/240-sensor-encoder-infrarrojo-fc-03-moch22a.html>.

**NOVATRONIC.** Relé de Estado Solido 2 Canales. *Novatronic*. [en línea]. 2020. [Consulta: 8 agosto 2023]. Disponible en: <https://novatronic.com/index.php/product/rele-de-estado-solido-2-canales/>.

**OGATA, K.** *Ingeniería De Control Moderna* [en línea]. Quinta ed. Madrid-España: PEARSON EDUCACIÓN. 2010. [Consulta: 13 julio 2023]. ISBN 978-84-8322-660-5. Disponible en: [https://www.academia.edu/9814191/Ingenieria\\_de\\_Control\\_Moderna\\_Ogata\\_5ed](https://www.academia.edu/9814191/Ingenieria_de_Control_Moderna_Ogata_5ed).

**OPEN WEBINARS.** Tutorial Arduino: Uso de sistema OneWire. *OpenWebinars.net* [en línea]. 2015. [Consulta: 13 junio 2023]. Disponible en: <https://openwebinars.net/blog/tutorial-arduino-uso-de-sistema-onewire/>.

**PINEDA, K.** Implementación de un Módulo de Entrenamiento para Control de Lazos Abiertos y Cerrados Utilizando LabVIEW para el Laboratorio de Mecatrónica de la Facultad de Mecánica. [en línea] (Trabajo de titulación) (Pregrado). ESPOCH, Riobamba, Ecuador. 2014. pp. 1-103. [Consulta: 8 mayo 2023]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/3063>.

**POSADAS, A.** Determinación de errores y tratamiento de datos. *Studocu* [en línea]. [Consulta: 15 agosto 2023]. 2023. Disponible en: <https://www.studocu.com/latam/document/universidad-santa-paula/quimica-organica/determinacion-de-errores-y-tratamiento-de-datos/27061475>

**RODRÍGUEZ, W.** Planta electrónica para entrenamiento de sistemas de control utilizando la interface NI MYDAQ y el software LABVIEW 2010 [en línea] (Trabajo de titulación) (Pregrado). Universidad Politécnica Salesiana, Quito, Ecuador. 2013. pp. 1-243. [Consulta: 8 mayo 2023]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/4502>

**ROHDE; & SCHWARZ.** Entendiendo el UART. *Rohde-Schwarz* [en línea]. 2023. [Consulta: 13 julio 2023]. Disponible en: [https://www.rohde-schwarz.com/lat/productos/prueba-y-medicion/essentials-test-equipment/digital-oscilloscopes/entendiendo-el-uart\\_254524.html](https://www.rohde-schwarz.com/lat/productos/prueba-y-medicion/essentials-test-equipment/digital-oscilloscopes/entendiendo-el-uart_254524.html).

**ROMERO, J.** ¿Qué es Arduino y para qué sirve? *GEEKNETIC* [en línea]. 2021a. [Consulta: 16 julio 2023]. Disponible en: <https://www.geeknetic.es/Arduino/que-es-y-para-que-sirve>.

**ROMERO, J.** ¿Qué es un FPGA y para qué sirve? - Definición. *GEEKNETIC* [en línea]. 2021b. [Consulta: 16 julio 2023]. Disponible en: <https://www.geeknetic.es/FPGA/que-es-y-para-que-sirve>.

**SALDIAS, A.** Tipos de Sensores de Temperatura y sus diferencias. *SRC Sistemas de Regulación y Control* [en línea]. 2018. [Consulta: 8 mayo 2023]. Disponible en: <https://srcsl.com/tipos-sensores-temperatura/>.

**SANTO, M.; & LECUMBERRY, G.** *El proceso de medición: análisis y comunicación de datos experimentales*. 1a.ed. Río Cuarto: Universidad Nacional de Río Cuarto. 2005. ISBN 978-950-665-344-6.

**SATYCSA.** Motor a pasos Nema 17 4.58Kg/cm 1.5A. *Satycsa* [en línea]. 2023. [Consulta: 8 agosto 2023]. Disponible en: <https://satycsa.com.mx/motor-a-pasos-nema-17-4-58kg-cm-1-5a/>.

**SDELSOL.** Coeficiente de variación ¿Qué es? *Sdelsol* [en línea]. 2022. [Consulta: 15 agosto 2023]. Disponible en: <https://www.sdelsol.com/glosario/coeficiente-de-variacion/>.

**SENAR, J.C.** “La medición de la repetibilidad y el error de medida”. *Etologia*, vol. 17 (1999) pp. 54-64.

**SOFTONIC.** Arduino IDE. [en línea]. 2023. [Consulta: 12 junio 2023]. Disponible en: <https://arduino.softonic.com/>.

**SUNTA, P.; & YÁNEZ, D.** Desarrollo de un entrenador inalámbrico de plantas de control mediante una herramienta de programación IOT [en línea] (Trabajo de titulación) (Pregrado). Universidad Politécnica Salesiana, Quito, Ecuador. 2021. pp. 1-91. [Consulta: 23 julio 2023]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/19668>

**THOMAS COUNTZ.** Arduino Tone for ESP32. *Thomas Countz* [en línea]. 2021. [Consulta: 13 junio 2023]. Disponible en: <https://thomascountz.com/2021/02/21/arduino-tone-for-esp32>.

**UNIT ELECTRONICS.** Led Blanco Frio de potencia 10W, 50W y 100W. *Uelectronics* [en línea]. 2019. [Consulta: 30 julio 2023]. Disponible en: <https://uelectronics.com/producto/led-blanco-frio-de-potencia-10w-50w-y-100w/>.

**VARGAS, V.** *Estadística descriptiva para ingeniería ambiental con SPSS* [en línea]. S.l.: Editora Viviana Vargas Franco. 2007. [Consulta: 15 agosto 2023]. Disponible en: <https://repositorio.unal.edu.co/handle/unal/52004>.

**ZUNO.** DallasTemperature. *Z-Wave.Me* [en línea]. 2023. [Consulta: 13 junio 2023]. Disponible en: <https://z-uno.z-wave.me/Reference/DallasTemperature/>





## 4 Electrical Characteristics

### 4.1 Absolute Maximum Ratings

Stresses above those listed in Table 4-1 *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only and normal operation of the device at these or any other conditions beyond those indicated in Section 4.2 *Recommended Power Supply Characteristics* is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Table 4-1. Absolute Maximum Ratings

Parameter	Description	Min	Max	Unit
VDDA, VDD3P3, VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO	Allowed input voltage	-0.3	3.6	V
$I_{output}^1$	Cumulative IO output current	—	1200	mA
$T_{STORE}$	Storage temperature	-40	150	°C

<sup>1</sup> The product proved to be fully functional after all its IO pins were pulled high while being connected to ground for 24 consecutive hours at ambient temperature of 25 °C.

### 4.2 Recommended Power Supply Characteristics

Table 4-2. Recommended Power Supply Characteristics

Parameter	Description	Min	Typ	Max	Unit
VDDA, VDD3P3_RTC <sup>note 1</sup> , VDD3P3, VDD_SDIO (3.3 V mode) <sup>note 2</sup>	Voltage applied to power supply pins per power domain	2.3/3.0 <sup>note 3</sup>	3.3	3.6	V
VDD3P3_CPU	Voltage applied to power supply pin	1.8	3.3	3.6	V
$I_{VDD}$	Current delivered by external power supply	0.5	—	—	A
$T$ <sup>note 4</sup>	Operating temperature	-40	—	125	°C

- When writing eFuse, VDD3P3\_RTC should be at least 3.3 V.
- VDD\_SDIO works as the power supply for the related IO, and also for an external device. Please refer to the Appendix IO\_MUX of this datasheet for more details.
  - VDD\_SDIO can be sourced internally by the ESP32 from the VDD3P3\_RTC power domain:
    - When VDD\_SDIO operates at 3.3 V, it is driven directly by VDD3P3\_RTC through a 6 Ω resistor, therefore, there will be some voltage drop from VDD3P3\_RTC.
    - When VDD\_SDIO operates at 1.8 V, it can be generated from ESP32's internal LDO. The maximum current this LDO can offer is 40 mA, and the output voltage range is 1.65 V ~ 2.0 V.
  - VDD\_SDIO can also be driven by an external power supply.
  - Please refer to Section 2.3.1 Power Scheme, for more information.
- Chips with a 3.3 V flash or PSRAM in-package: this minimum voltage is 3.0 V;
  - Chips with no flash or PSRAM in-package: this minimum voltage is 2.3 V;
  - For more information, see Section 1 ESP32 Series Comparison.
- The operating temperature of ESP32-U4WDH ranges from -40 °C to 105 °C, due to the in-package flash.
  - The operating temperature of ESP32-D0WDR2-V3 ranges from -40 °C to 85 °C, due to the in-package PSRAM.
  - For other chips that have no in-package flash or PSRAM, their operating temperature is -40 °C ~ 125 °C.

## ANEXO B: HOJA DE DATOS DEL ESP32 DRIVER M1584



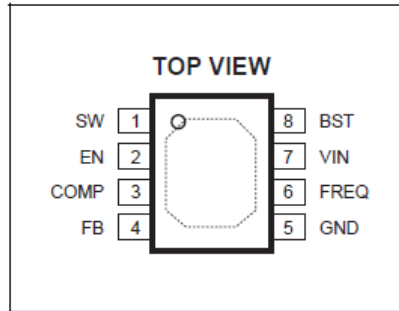
MP1584 – 3A, 1.5MHz, 28V STEP-DOWN CONVERTER

### ORDERING INFORMATION

Part Number*	Package	Top Marking	Free Air Temperature (T <sub>A</sub> )
MP1584EN	SOIC8E	MP1584EN	-20°C to +85°C

\* For Tape & Reel, add suffix -Z (e.g. MP1584EN-Z);  
For RoHS Compliant Packaging, add suffix -LF. (e.g. MP1584EN-LF-Z)

### PACKAGE REFERENCE



### ABSOLUTE MAXIMUM RATINGS <sup>(1)</sup>

Supply Voltage (V <sub>IN</sub> )	-0.3V to +30V
Switch Voltage (V <sub>SW</sub> )	-0.3V to V <sub>IN</sub> + 0.3V
BST to SW	-0.3V to +6V
All Other Pins	-0.3V to +6V
Continuous Power Dissipation (T <sub>A</sub> = +25°C) <sup>(2)</sup>	2.5W
Junction Temperature	150°C
Lead Temperature	260°C
Storage Temperature	-65°C to +150°C

**Recommended Operating Conditions <sup>(3)</sup>**

Supply Voltage V <sub>IN</sub>	4.5V to 28V
Output Voltage V <sub>OUT</sub>	0.8V to 25V

Operating Junct. Temp (T<sub>J</sub>) ..... -20°C to +125°C

Thermal Resistance <sup>(4)</sup>	θ <sub>JA</sub>	θ <sub>JC</sub>
SOIC8E	50	10

#### Notes:

- Exceeding these ratings may damage the device.
- The maximum allowable power dissipation is a function of the maximum junction temperature T<sub>J(MAX)</sub>, the junction-to-ambient thermal resistance θ<sub>JA</sub>, and the ambient temperature T<sub>A</sub>. The maximum allowable continuous power dissipation at any ambient temperature is calculated by P<sub>D(MAX)</sub>=(T<sub>J(MAX)</sub>-T<sub>A</sub>)/θ<sub>JA</sub>. Exceeding the maximum allowable power dissipation will cause excessive die temperature, and the regulator will go into thermal shutdown. Internal thermal shutdown circuitry protects the device from permanent damage.
- The device is not guaranteed to function outside of its operating conditions.
- Measured on JE5D61-7, 4-layer PCB.

### ELECTRICAL CHARACTERISTICS

V<sub>IN</sub> = 12V, V<sub>EN</sub> = 2.5V, V<sub>COMP</sub> = 1.4V, T<sub>A</sub> = +25°C, unless otherwise noted.

Parameter	Symbol	Condition	Min	Typ	Max	Units
Feedback Voltage	V <sub>FB</sub>	4.5V < V <sub>IN</sub> < 28V	0.776	0.8	0.824	V
Upper Switch On Resistance	R <sub>DS(ON)</sub>	V <sub>BST</sub> - V <sub>SW</sub> = 5V		150		mΩ
Upper Switch Leakage		V <sub>EN</sub> = 0V, V <sub>SW</sub> = 0V, V <sub>IN</sub> = 28V		1		μA
Current Limit			4.0	4.7		A
COMP to Current Sense Transconductance	G <sub>CS</sub>			9		A/V
Error Amp Voltage Gain <sup>(5)</sup>				200		V/V
Error Amp Transconductance		I <sub>COMP</sub> = ±3μA	40	60	80	μA/V
Error Amp Min Source current		V <sub>FB</sub> = 0.7V		5		μA
Error Amp Min Sink current		V <sub>FB</sub> = 0.9V		-5		μA
VIN UVLO Threshold			2.7	3.0	3.3	V
VIN UVLO Hysteresis				0.35		V
Soft-Start Time <sup>(5)</sup>		0V < V <sub>FB</sub> < 0.8V		1.5		ms
Oscillator Frequency	f <sub>FREQ</sub>	R <sub>FREQ</sub> = 100kΩ		900		kHz
Shutdown Supply Current		V <sub>EN</sub> = 0V		12	20	μA
Quiescent Supply Current		No load, V <sub>FB</sub> = 0.9V		100	125	μA
Thermal Shutdown				150		°C
Thermal Shutdown Hysteresis				15		°C
Minimum Off Time <sup>(5)</sup>				100		ns
Minimum On Time <sup>(5)</sup>				100		ns
EN Up Threshold			1.35	1.5	1.65	V
EN Hysteresis				300		mV

#### Note:



5) Guaranteed by design.

## ANEXO C: HOJA DE DATOS DEL DRIVER DRV8825

### TECHNICAL SPECIFICATION

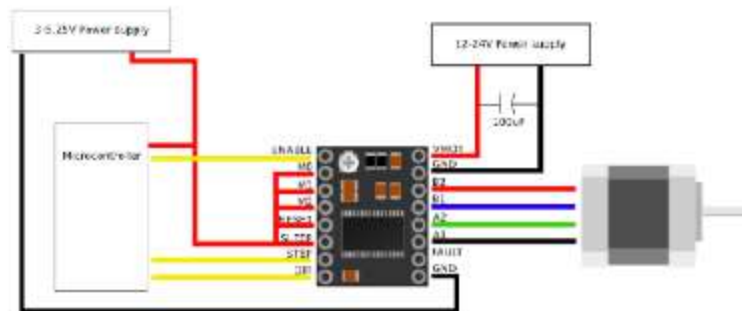
Controller	DRV8825
Operating Voltage (logic)	3-5.25V
Operating Voltage (vmot)	12-24V
Max current	2.5A
Dimensions	20.4x15.6mm

### MAJOR FEATURES

	<p><b>DRV8825</b> Powerful DRV8825 with</p> <ul style="list-style-type: none"> <li>• High current driver capable up to 2.5A</li> <li>• Six different step resolutions: full-step, half-step, 1/4-step, 1/8-step, 1/16-step, and 1/32-step</li> <li>• Protection against over-temperature and over-current</li> <li>• No logic voltage required</li> </ul>
	<p><b>Adjustable current</b> Using the trimpot on the board you can easily turn the current up or down. Turn left to lower the current, right to output a higher current.</p>

### EXAMPLE CIRCUIT

The following diagram lists the pins and show an example circuit.



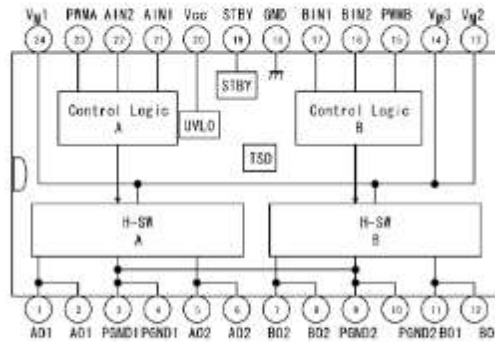
Name	Description
Enable	Enable/disable the stepper driver HIGH - Disable LOW - Enable *
M0 - M2	Step resolution setting, see chapter 'step resolution configuration'
RESET	Enable/disable the H-bridge output * LOW - Disable * HIGH - Enable
SLEEP	Enable/disable low-power sleep mode LOW - Sleep * HIGH - Active
STEP	LOW → HIGH, move one step
DIR	LOW / HIGH switches direction
VMOT	Motor power (12-24V)
GND	System ground
FAULT	LOW when the stepper driver is in fault condition. You can provide 5V on this pin for compatibility with stepstick A4988

# ANEXO D: HOJA DE DATOS DEL DRIVER TB6612FNG

**TOSHIBA**

TB6612F

## Block Diagram



## Pin Functions

No.	Pin Name	I/O	Function
1	AO1	O	ch A output 1
2	AO1	O	ch A output 1
3	PGND1	—	Power GND 1
4	PGND1	—	Power GND 1
5	AO2	O	ch A output 2
6	AO2	O	ch A output 2
7	BO2	O	ch B output 2
8	BO2	O	ch B output 2
9	PGND2	—	Power GND 2
10	PGND2	—	Power GND 2
11	BO1	O	ch B output 1
12	BO1	O	ch B output 1
13	VM2	—	Motor supply
14	VM2	—	Motor supply
15	PAWA	I	ch A PWM input(250 kΩ pull-down at internal)
16	BIN2	I	ch B input 2(200 kΩ pull-down at internal)
17	BIN1	I	ch B input 1(200 kΩ pull-down at internal)
18	GND	—	Small signal GND
19	STBY	I	"L" = standby(200 kΩ pull-down at internal)
20	Vcc	—	Small signal supply
21	AIN1	I	ch A input 1(200 kΩ pull-down at internal)
22	AIN2	I	ch A input 2(200 kΩ pull-down at internal)
23	PAWA	I	ch A PWM input(200 kΩ pull-down at internal)
24	VM1	—	Motor supply

**TOSHIBA**

TB6612FNG

## Electrical Characteristics (unless otherwise specified, Ta = 25°C, Vcc = 3 V, VM = 5 V)

Characteristics	Symbol	Test Condition	Min	Typ	Max	Unit
Supply current	$I_{CC}(3 V)$	STBY = Vcc = 3 V, VM = 5 V	—	1.1	1.8	mA
	$I_{CC}(5.5 V)$	STBY = Vcc = 5.5 V, VM = 5 V	—	1.5	2.2	mA
	$I_{CC}(I_{CP})$	STBY = 2 V	—	—	1	μA
	$I_{CC}(I_{STR})$		—	—	1	μA
Control input voltage	$V_{IH}$	—	Vcc+0.7	—	Vcc+0.2	V
	$V_{IL}$	—	-0.2	—	Vcc-0.3	V
Control input current	$I_{IH}$	V <sub>IN</sub> = 3 V	0	15	25	μA
	$I_{IL}$	V <sub>IN</sub> = 5 V	—	—	1	μA
Standby input voltage	$V_{IH}(I_{CP})$	—	Vcc+0.7	—	Vcc+0.2	V
	$V_{IL}(I_{STR})$	—	-0.2	—	Vcc-0.3	V
Standby input current	$I_{I}(I_{CP})$	V <sub>IN</sub> = 3 V	0	15	25	μA
	$I_{I}(I_{STR})$	V <sub>IN</sub> = 5 V	—	—	1	μA
Output maximum voltage	$V_{OH}(I_{OL})$	I <sub>O</sub> = 1 A, Vcc = VM = 5 V	—	0.5	0.7	V
	$V_{OH}(I_{OR})$	I <sub>O</sub> = 0.5 A, Vcc = VM = 5 V	—	0.15	0.21	V
Output leakage current	$I_{OL}$	VM = V <sub>OUT</sub> = 15 V	—	—	1	μA
	$I_{OR}$	VM = 15 V, V <sub>OUT</sub> = 2 V	-1	—	—	μA
Regenerative diode VF	$V_{FD}$	I <sub>D</sub> = 1 A	—	1	1.1	V
	$V_{FR}$	I <sub>D</sub> = 1 A	—	1	1.1	V
Low-voltage detecting voltage	UVLO	(Design target only)	—	1.0	—	V
Recovering voltage	UVLC	(Design target only)	—	2.2	—	V
Response speed	$t_r$	(Design target only)	—	24	—	ns
	$t_f$	(Design target only)	—	41	—	ns
	Dead time	Prevention protect time (Design target only)	—	50	—	ns
Thermal shutdown circuit operating temperature	TSD	(Design target only)	—	175	—	°C
Thermal shutdown hysteresis	ΔTSD	(Design target only)	—	20	—	°C

## ANEXO E: HOJA DE DATOS DEL RELEVADOR DE ESTADO SÓLIDO G3MB-202P

# OMRON®

## Solid State Relay

## G3MB

- Switches 2-A loads at 25°C
- DC input with AC output
- Space-saving design, ideal for high-density PCB applications
- Bottom is approximately three times smaller than Omron's G3M relay
- UL 508 recognized, CSA certified



### Ordering Information

To Order: Select the part number and add the desired coil voltage rating, (e.g., G3MB-202P-DC12).

Isolation	Rated load voltage	Zero cross function	Rated input voltage	Part number
				Snubber circuit/No indicator
Phototriac	2 A at 100 to 120 VAC	No	5 VDC	G3MB-102PL-DC5
			12 VDC	G3MB-102PL-DC12
			24 VDC	G3MB-102PL-DC24
	2 A at 100 to 240 VAC	Yes	5VDC	G3MB-202P-DC5
			12 VDC	G3MB-202P-DC12
			24 VDC	G3MB-202P-DC24
		No	5 VDC	G3MB-202PL-DC5
			12 VDC	G3MB-202PL-DC12
			24 VDC	G3MB-202PL-DC24

### Specifications

#### ■ INPUT RATINGS

Ambient temperature 25° (77°F)

Type	Rated voltage	Operating voltage range	Impedance	Voltage Level	
				Must operate voltage	Must release voltage
G3MB-102PL	5 VDC	4 to 6 VDC	440 Ω, ±20%	4 VDC max.	1 VDC min.
G3MB-202P	12 VDC	9.60 to 14.40 VDC	1 kΩ, ±20%	9.60 VDC max.	1 VDC min.
G3MB-202PL	24 VDC	19.20 to 28.80 VDC	2.20 kΩ, ±20%	19.20 VDC max.	1 VDC min.

Note: Each model has 5 VDC, 12 VDC, and 24 VDC input versions.

#### ■ OUTPUT RATINGS

Type	Applicable load			
	Rated load voltage	Load voltage range	Load current	Surge current
G3MB-102PL	100 to 120 VAC, 50/60 Hz	75 to 132 VAC, 50/60 Hz	0.10 to 2 A	30 A (60 Hz, 1 cycle)
G3MB-202P	120 to 240 VAC, 50/60 Hz	75 to 264 VAC, 50/60 Hz	0.10 to 2 A	30 A (60 Hz, 1 cycle)
G3MB-202PL	100 to 240 VAC, 50/60 Hz	75 to 264 VAC, 50/60 Hz	0.10 to 2 A	30 A (60 Hz, 1 cycle)

## ANEXO F: HOJA DE DATOS DEL MOTORREDUCTOR JGA25-370



### JGA25-370 Geared Motor

SKU 114090046

Please use this motor as an alternative to JGR37-371 and Encoder Geared Motor JGA25-371

What is the Geared motor?

The geared motor uses a gear set to convert the original high speed and low torque of the motor to a low speed and high torque state. So what are the benefits of geared motors? Under the same voltage conditions, you can manually clamp the motor to stop it, but once it is a gear motor, it is more difficult to stop the motor with an external force because the "force" of the motor becomes larger. Therefore, when you use a geared motor, you will find it is slower than a motor that does not slow down, but it can provide a larger load. Geared motors are typically used where high torque is required, such as an elevator, which will carry more than a dozen people upstairs, which will require a lot of torque. Of course, there will be some energy loss during deceleration, but it will still bring a lot of convenience to our lives.

Introduction:

This Motor not encoder. Mainly used in robot platform and car provides power. Good quality and long lifetime, high torque and low noise.

If you need an encoder, you can choose JGA25-371 Geared Motor with Encoder.

Specification:

Voltage V	No-load		Maximum efficiency pointed				Blockage	
	speed r/min	electric current A	speed r/min	electric current A	Torque Kg.cm	Power W	Torque Kg.cm	electric current A
6	190	0.2	133	0.5	0.75	1.1	4.0	2.1
12	350	0.1	245	0.65	1.4	2.4	5.2	2.2

size:



L-21

Part List

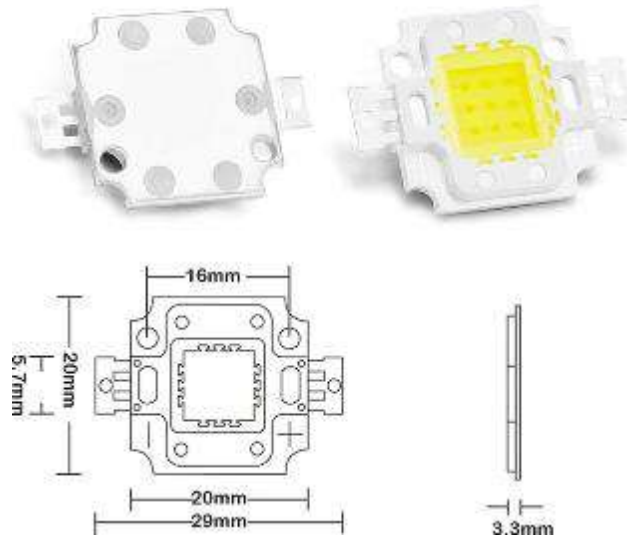
1 x JGA25-370 Geared Motor

ECCN/HTS

ECCN	EAR99
HS CODE	8501101000
UPC	

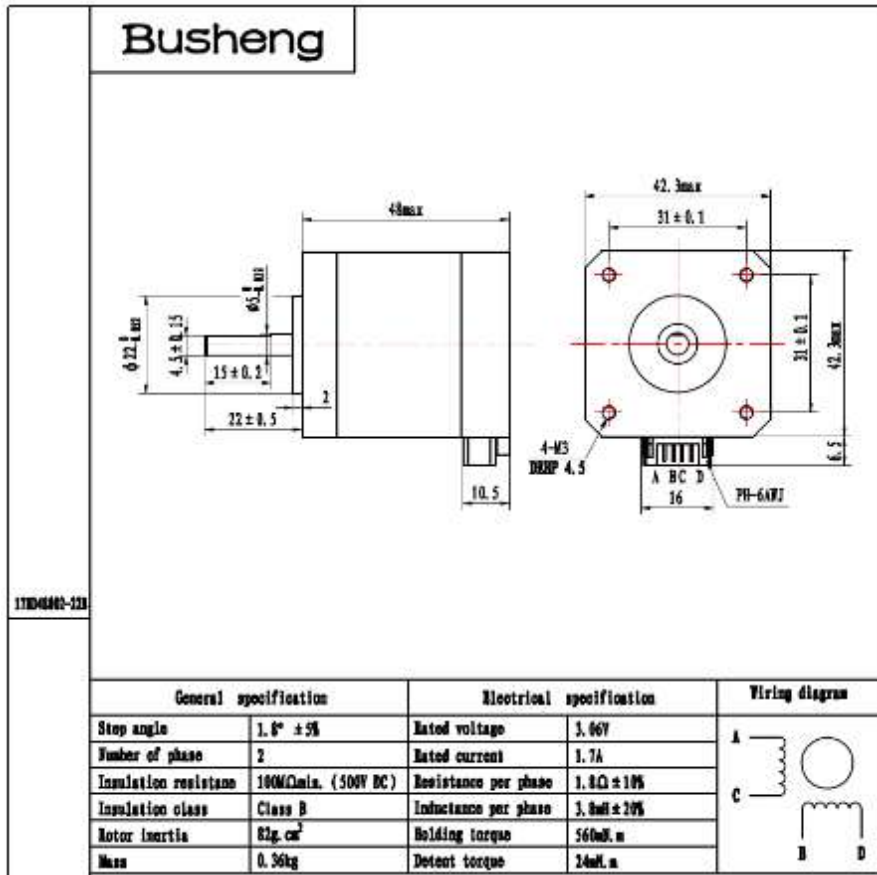
## ANEXO G: HOJA DE DATOS DE LUZ LED DE 10W

### ▼ 10W LED Size



Power	Color	Color Temperature Wavelength	Forward Voltage	Forward Current	Luminous Flux
10W (3x3)	Warm White	3000-3500K	9-11V	840-1000mA	900-1000LM
	Natural White	4000-4500K	9-11V	840-1000mA	900-1000LM
	White	8000-8500K	9-11V	840-1000mA	900-1000LM
	Cool White	10000-15000K	9-11V	840-1000mA	900-1000LM
	Red	620-625nm	6-7V	840-1000mA	350-450LM
	Blue	460-470nm	9-11V	840-1000mA	50-100LM
	Cyan	480nm	9-11V	840-1000mA	120-130 LM
	Green	520-525nm	8-10V	840-1000mA	800-800LM
	Yellow	580-592nm	8-7V	840-1000mA	450-500LM
	Orange	600-605nm	9-11V	840-1000mA	600-700LM
	UV/Purple	395-400nm	8-12V	800-800mA	/
	Royal Blue (Grow LED)	440-445nm	9-11V	840-1000mA	90-100LM
	Deep RED (Grow LED)	660nm	6-7V	840-1000mA	350-450LM
	Full Spectrum (Grow LED)	380-840nm	9-11V	840-1000mA	200-300LM
	IR(730nm)	730-740nm	6-7V	840-1000mA	/
	IR(850nm)	850nm	5-6V	840-1000mA	/
	IR(940nm)	940nm	4-5.5V	840-1000mA	/
RGB	R: 620-630nm G: 520-525nm B: 460-465nm	R: 8-7V G: 9-11V B: 9-11V	R: 280-350mA G: 280-350mA B: 280-350mA	R: 90-100LM G: 120-150LM B: 30-40LM	

**ANEXO H: HOJA DE DATOS MOTOR NEMA 17HD48002H-22B**





# ANEXO I: DIAGRAMA DE BLOQUES CORRESPONDIENTES A LAS INTERFACES DE CONTROL Y VISUALIZACIÓN DE CADA UNA DE LAS PRACTICAS A DESARROLLARSE EN LA PLCP

Diagrama de bloques de la interfaz de la práctica de control de posición angular

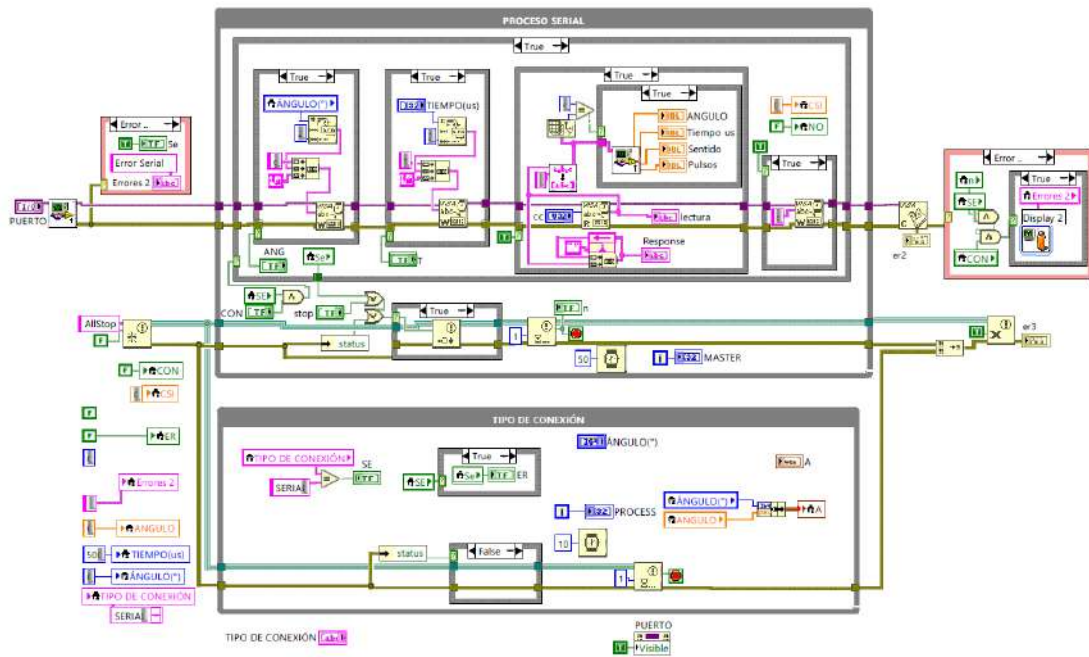


Diagrama de bloques de la interfaz de control manual de velocidad del motor DC

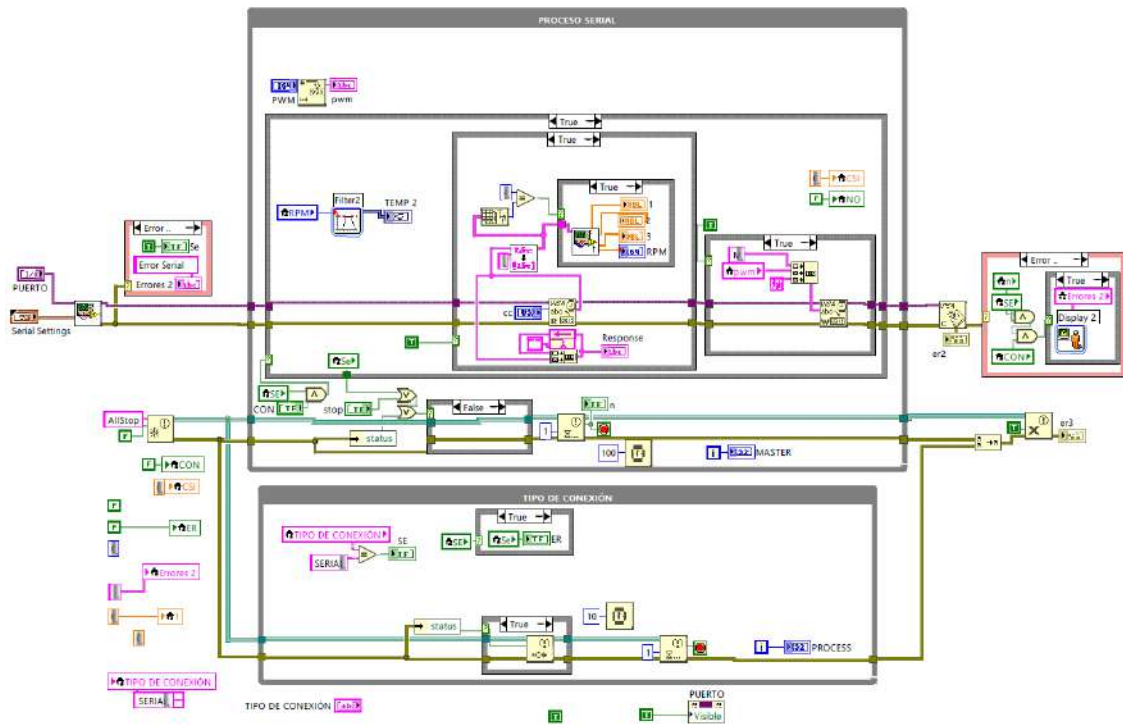


Diagrama de bloques de interfaz de control de activación de SSR

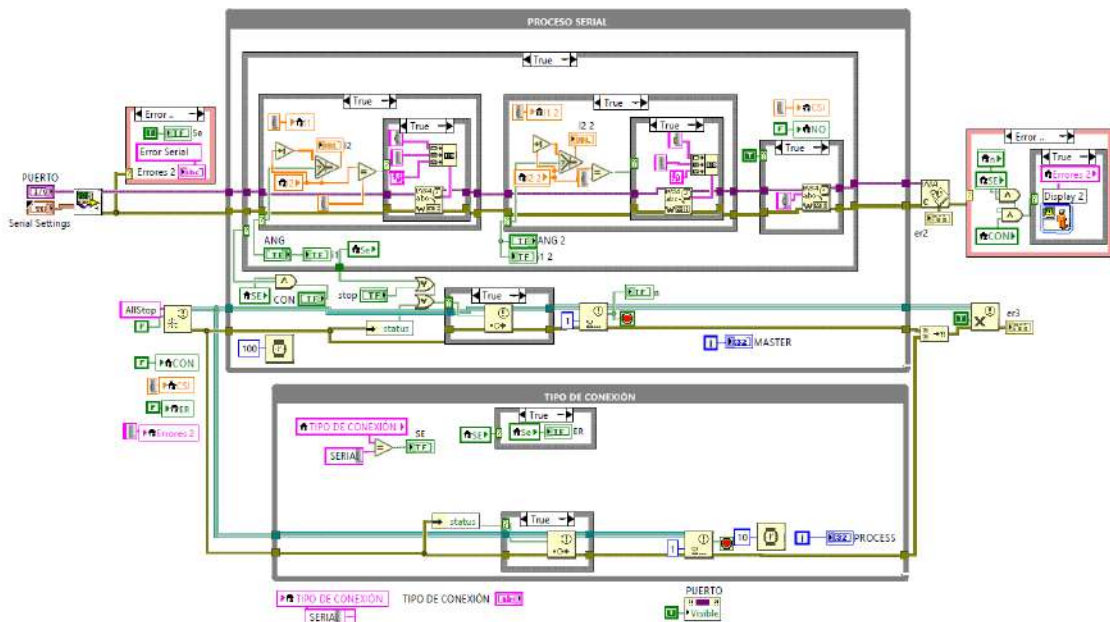


Diagrama de bloques de la interfaz de control de temperatura manual

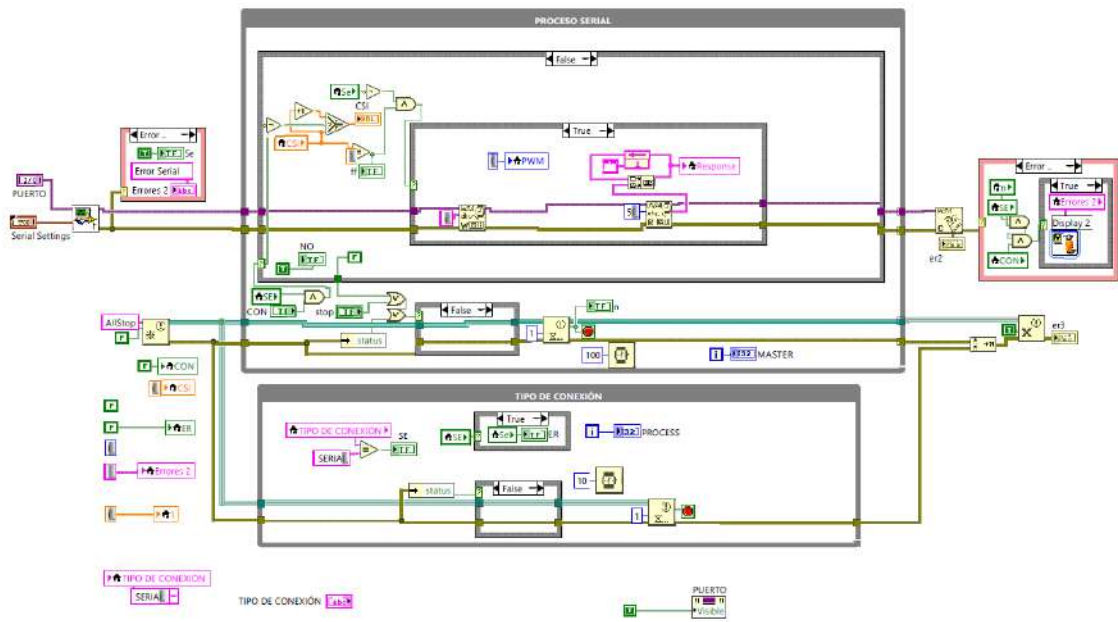
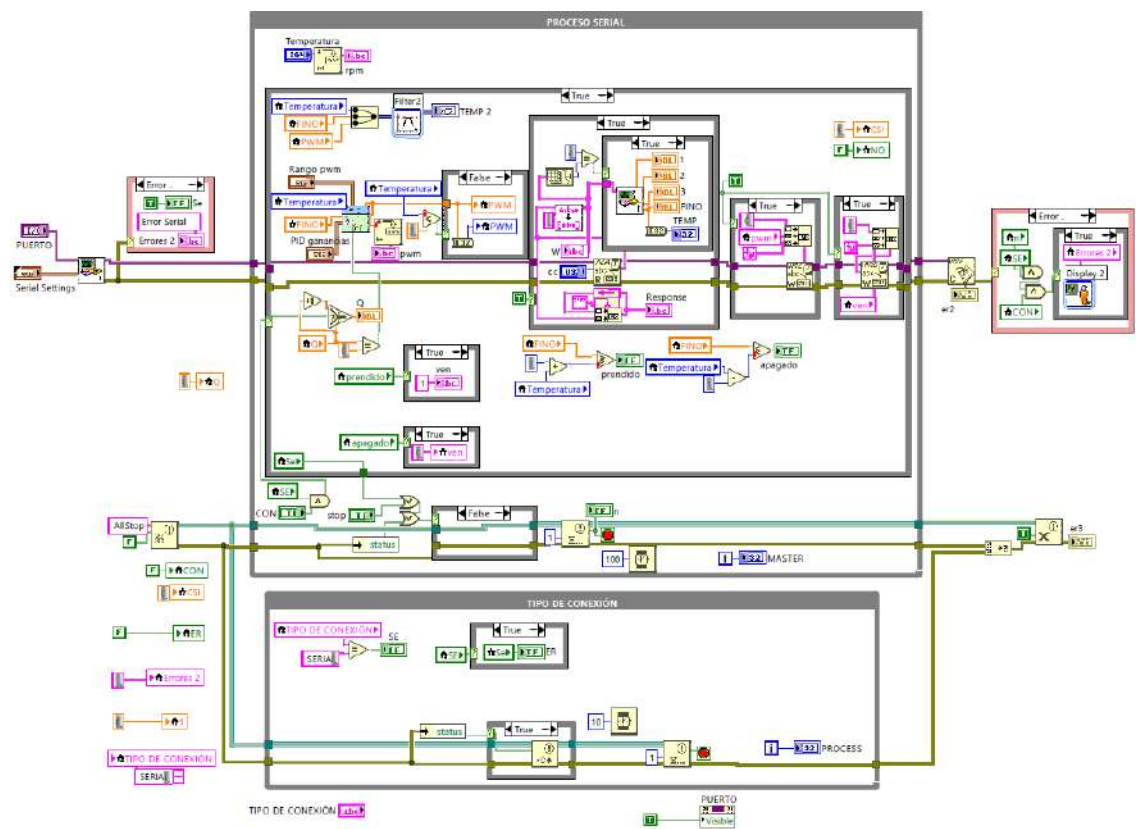
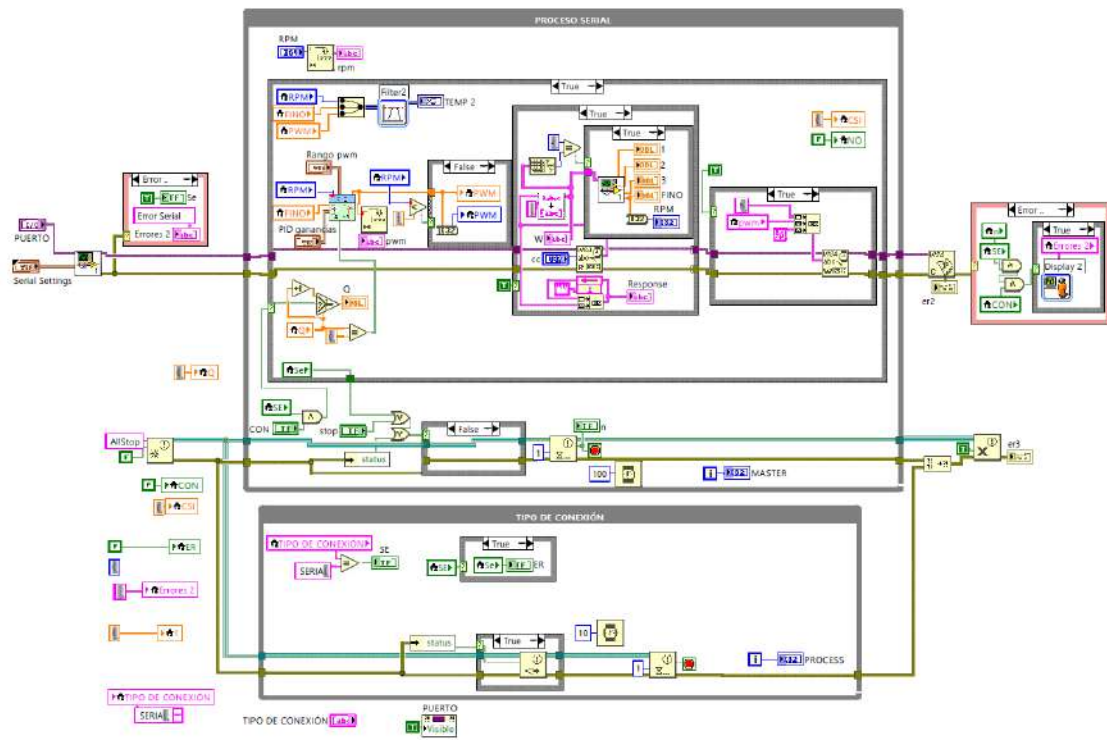


Diagrama de bloques de la interfaz de control PID de temperatura



# Diagrama de bloques de la interfaz de control PID de velocidad del motor DC



## ANEXO J: HOJA DE DATOS DE LA CAJA DE CONEXIONES HEYIARBEIT DE PLÁSTICO ABS

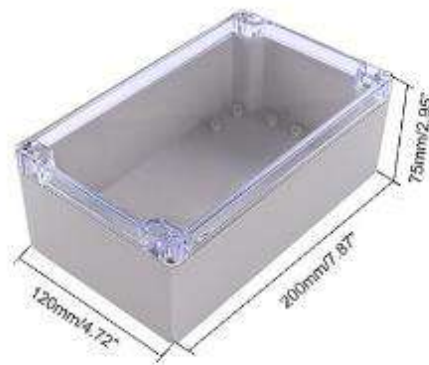
<b>Material</b>	Acrilonitrilo butadieno estireno Plástico Goma
<b>Marca</b>	Heyiarbeit
<b>Estilo</b>	Protección
<b>Tipo de acabado</b>	Pulido
<b>Tipo de instalación</b>	Atornillar

---

### Sobre este artículo

- Dimensiones - Tamaño exterior de la caja eléctrica (aprox. ): 200 x 120 x 75 mm / 7.87" x 4.72" x 2.95" (LxWxH); Nombre del producto: caja de conexiones; grado de protección impermeable: IP65; Material: ABS; Color principal: Gris claro (para el cuerpo de la caja), transparente (para la cubierta de la caja)
- Ventaja: la caja del proyecto es fácil de operar y reparar, sin corte de cable, más conveniente y a prueba de humedad, protector solar, anticorrosión, duradera para años de uso
- Material: esta caja de proyecto de gabinete está hecha de plástico ABS de calidad, duradero y práctico, resistente, material de calidad y diseño único siempre mantiene tus dispositivos nuevos y limpios
- Aplicación: esta caja de proyecto es adecuada para interiores y exteriores eléctricos comunicaciones, aeropuerto, estación de metro, caja de botones, terminal pequeño, chasis de señal, sensor, dispositivo de medición y otros equipos
- Contenido del paquete: 1 caja de conexiones, 4 tornillos (no inoxidable), 1 ojal de goma

### Product Size



## ANEXO K: EVIDENCIA DEL MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

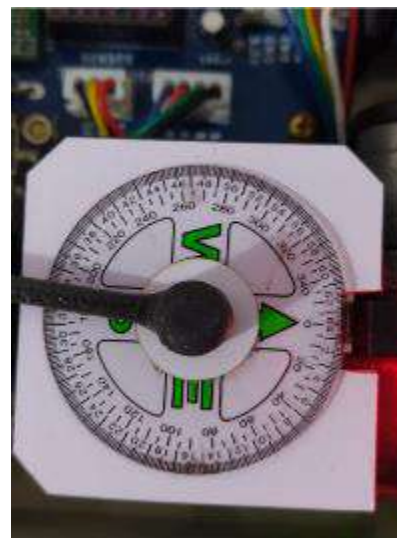
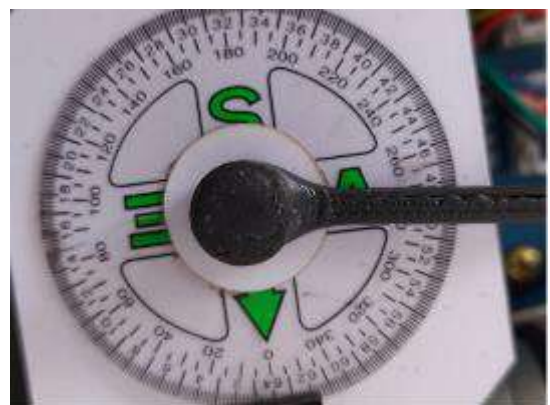
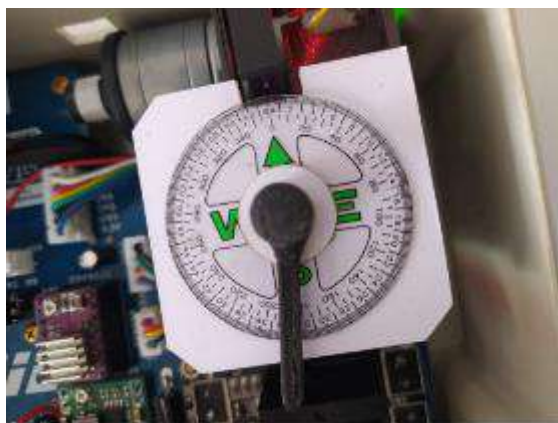
Toma de datos de temperatura usando un termómetro tipo pistola



Toma de datos de velocidad en RPM mediante el uso de un tacómetro laser



Toma de datos de velocidad de grados sexagesimales mediante el uso de una rueda graduada tipo brújula





# Comprobación del correcto funcionamiento de la activación de SSRs usando cargas AC



## ANEXO L: PROMOCIÓN REALIZADA EN ARDUINO IDE 2.1.1

// LIBRERIAS

#include <OneWire.h>

#include <DallasTemperature.h>

#include <ToneESP32.h>

#define SENSOR\_PIN 18 // ESP32 pin GIOP17 connected to DS18B20 sensor's DATA pin

#define STBY 27 // STBY MOTOR H

#define EN 22 // ENABLE PASOS

#define CHA1 2 // SSR 1

#define CHA2 15 // SSR2

#define ven 23 // ventilador

#define BUZZER\_PIN 12 // buzzer

#define BUZZER\_CHANNEL 0

#define AIN1 26 // sentido A

#define AIN2 25 // sentido B

//=====

// TONO DE BUZZER

ToneESP32 buzzer(BUZZER\_PIN, BUZZER\_CHANNEL); //objeto buzzer pin y canal

//=====

// SENSOR DE TEMPERATURA

OneWire oneWire(SENSOR\_PIN);

DallasTemperature DS18B20(&oneWire);

bool MANUALTEMP = false;

bool PIDTEMP = false;

bool ventilador = false;

bool undatoven = false;

bool undatoven2 = false;

//=====

// PROCESOS PARALELOS

TaskHandle\_t taskSensor; //proceso paralelo para lectura de temperatura

TaskHandle\_t taskCuadratura; //proceso paralelo para lectura sensor cuadratura

//=====

// CONTROL ANGULAR Y VELOCIDAD

float tempC = 0; // temperatura en grados c

```

float tempF = 0; // temperatura en grados f
int motor = 0; // pwm recibido para motor dc
//=====
// CONTROL ANGULAR Y VELOCIDAD
String v1 = ""; //angulo
String v2 = ""; // tiempo
String v3 = ""; //sentido
String v4 = ""; // pasos
//=====
// CONTROL ANGULAR Y VELOCIDAD
int target_angle = 0; //angulo objetivo
int current_angle = 0; //angulo actual
int steps_per_rev = 400; //200
int dir_pin = 19; //pin direccion de giro
int step_pin = 21; //pin pasos
int timest = 500; //tiempo de pasos
//=====
int contador = 0; //contador de pulsos encoder
int sensor = 13; //pin sensor encoder
unsigned long previousMillis = 0; //valor anterior contador de desconexion serial
const long interval = 1000; //timer 1 segundo
int steps_to_move = 0; //variable de converion de angulo a pasos
int pasosm = 0; //pasos para motor
bool GIRAR = false; //habilitador para movimiento angular
bool seriac = false; //impresion 1 vez estado conexion
bool SERIALCONNECT = false; //variable de conexion serial
//=====
int sentido = 0; // 0 izquierda, 1 derecha
String data = ""; // cadena de lectura serial
//=====
bool CONTROLA = false; //Control de angulo
//=====CONTROL ON/OFF TEMPERATURA
const int PWM_CHANNEL = 0; //canal para señal pwm 1
const int PWM_CHANNEL2 = 1; //canal para señal pwm 2
const int PWM_FREQ = 10000; // Frecuencia en Hz
const int PWM_RESOLUTION = 8; // Resolución en bits (1-16)

```

```

const int PWM_PIN = 14;    //pin de conexion para luz
const int PWM_PIN2 = 33;  //pin conexion motor
int luz = 0;              //valor de pwm para foco recibido

String b = ""; //String de dato de temperatura a enviar.
//=====ssr
int ssr1 = 0;            //valor de ssr1: 1=on y 0=off
int ssr2 = 0;            //valor de ssr2: 1=on y 0=off
volatile int encoderPos = 0;    //interrupciones detectadas
volatile unsigned long lastTime2 = 0; //tiempo anterior para medicion rpm
volatile float rpm = 0.0;      // valor velocidad de motor dc
String x = "";                // String para envio e datos de velocidad
int desco = 0;                //contador para desconexion inicial
bool contar = false;          //habilitador de contador desconexion inicial
int serialno = 0;             //perdida de conexion a los 2 segundos
bool contar2 = false;         //habilitador de desconexion 2
bool iniposi = false;

//=====Conteo de pulsos encoder

void IRAM_ATTR interrupcion() {
  contador++;
  //Serial.println(contador);
}

//===== Conteo de pulsos encoder cuadratura
void isr() {

  static boolean A_old = false;
  static boolean B_old = false;

  boolean A = digitalRead(32);
  boolean B = digitalRead(35);

  if (!A_old && A) {

```

```

if (!B) {
    encoderPos--;
} else {
    encoderPos++;
}
}

if (!B_old && B) {
    if (A) {
        encoderPos--;
    } else {
        encoderPos++;
    }
}

A_old = A;
B_old = B;
}

void taskSensorFunc(void* parameter) {

    for (;;) {
        DS18B20.requestTemperatures();    // obtener temperatura
        tempC = DS18B20.getTempCByIndex(0); // obtener valor en °C
        tempF = tempC * 9 / 5 + 32;        // convertir °C to °F
        b = "1,2,3," + String(tempC);

        ventila();
        //Serial.println(tempC);

        vTaskDelay(100 / portTICK_PERIOD_MS); // delay for 100 ms
    }
}

```

```

void taskcuadra(void* parameter) {

for (;;) {
    unsigned long currentTime2 = millis();
    if (currentTime2 - lastTime2 >= 1000) {
        // Medir RPM cada segundo
        float revs = encoderPos / 2.0;           // Cada pulso del encoder representa medio ciclo
        rpm = (revs / (currentTime2 - lastTime2)) * 280.0; // RPM = (revoluciones / tiempo) * 60000
cada 60 segundos
        rpm = abs(rpm);
        x = "1,2," + String(revs) + "," + String(rpm);
        lastTime2 = currentTime2; //igualar tiempos
        encoderPos = 0;           // Reiniciar el contador de pulsos
    }
    vTaskDelay(100 / portTICK_PERIOD_MS); // delay for 100 milliseconds
}
}
//=====

```

```

void setup() {

Serial.begin(9600); // velocidad de comunicacion con labview
//Serial.setTimeout(100); // tiempo de espera para datos
pinMode(ven, OUTPUT);
digitalWrite(ven, LOW); //ventilador pagado

pinMode(AIN1, OUTPUT); //sentido 1 motor dc
pinMode(AIN2, OUTPUT); //sensito 2 motor dc
//motor gira en un sentido definido
digitalWrite(AIN1, HIGH); // 1
digitalWrite(AIN2, LOW); // 0

pinMode(CHA1, OUTPUT); //canal 1 ssr
pinMode(CHA2, OUTPUT); //canal 2 ssr

```

```

//Apagar SSR1-2
digitalWrite(CHA1, HIGH);
digitalWrite(CHA2, HIGH);

pinMode(32, INPUT_PULLUP); //cuadratura a
pinMode(35, INPUT_PULLUP); //cuadratura b

pinMode(SENSOR_PIN, INPUT); //temperatura
pinMode(sensor, INPUT); //encoder

pinMode(dir_pin, OUTPUT); // pin direccion motor pasos
pinMode(step_pin, OUTPUT); // pin pasos
pinMode(EN, OUTPUT); // habilitador pasos
digitalWrite(EN, HIGH); // drive apagado

pinMode(STBY, OUTPUT); // habilitador puente H
digitalWrite(STBY, LOW); //punte h desactivado

pinMode(ven, OUTPUT); //ventilador
digitalWrite(ven, LOW);
//pines encoder cuadratura
attachInterrupt(digitalPinToInterrupt(32), isr, CHANGE);
attachInterrupt(digitalPinToInterrupt(35), isr, CHANGE);
//Tono para sistema OK

Serial.println("INICIANDO SISTEMA-PLANTA");
delay(1000);
//toneconnect();
toneinicio(); //Tono de encendido
attachInterrupt(sensor, interrupcion, RISING);
//PWM SALIDA LUZ
ledcSetup(PWM_CHANNEL, PWM_FREQ, PWM_RESOLUTION);
// Asignamos el pin GPIO al canal PWM
ledcAttachPin(PWM_PIN, PWM_CHANNEL);
//PWM SALIDA MOTOR DC
ledcSetup(PWM_CHANNEL2, PWM_FREQ, PWM_RESOLUTION);

```

```

// Asignamos el pin GPIO al canal PWM
ledcAttachPin(PWM_PIN2, PWM_CHANNEL2);

//Procesos paralelos

xTaskCreatePinnedToCore(taskSensorFunc, "Sensor Task", 1000, NULL, 1, &taskSensor, 0);
xTaskCreatePinnedToCore(taskcuadra, "cuadratura Task", 1000, NULL, 1, &taskCuadratura,
1);

//Apagar buzzer
//buzzer.noTone();
}
//=====
//=====
void loop() {

if (contar == false && SERIALCONNECT == false) {
    desco += 1;
}
if (desco == 1000) {
    Serial.println("SIN CONEXIONES");
    contar = true;
    desco = 1001;
    detener();
    alarm();
}
//FUNCIONES DE ESCUCHA
CONNECT_SERIAL();
}

//=====
void ventila() {
if (MANUALTEMP) {
    if (tempC >= 50) {
        digitalWrite(ven, HIGH);
    } else {

```



```

    digitalWrite(ven, LOW);
}
} else {
    if (!PIDTEMP) {
        if (tempC >= 35) {
            digitalWrite(ven, HIGH);
        } else {
            digitalWrite(ven, LOW);
        }
    }
}
}

//=====
void CONNECT_SERIAL() {

    if (Serial.available() > 0) {
        serialno = 0;
        contar = false;
        contar2 = false;
        desco = 0;

        data = Serial.readStringUntil('\n');

        //Serial.println(data);

        if (data == "d") {
            SERIALCONNECT = false;
            seriac = false;
            Serial.println("desConectado");
            detener();
        }

        if (data.substring(0, 1) == "B") { // BUZZER
            alarm2();

```

```

delay(200);
buzzer.noTone();
}

if (data.substring(0, 1) == "s") { // MODO CONTROL ANGULAR
    SERIALCONNECT = true;
    buzzer.noTone();
    digitalWrite(STBY, HIGH);
    digitalWrite(ven, LOW);
    digitalWrite(EN, LOW); //PRENDIDO
    v1 = String(current_angle);
    v2 = String(timest);
    v3 = String(sentido);
    v4 = String(pasosm);
    String a = v1 + "," + v2 + "," + v3 + "," + v4;
    Serial.println(a);
    CONTROLA = true;
    MANUALTEMP = false;
    PIDTEMP = false;
}

if (data.substring(0, 1) == "R") { // MODO CONTROL MANUAL TEMPERATURA
    buzzer.noTone();
    luz = data.substring(1).toInt(); // Leer el número
    ledcWrite(PWM_CHANNEL, luz);
    ledcWrite(PWM_CHANNEL2, 0);
    Serial.println(b);
    digitalWrite(STBY, HIGH);
    digitalWrite(EN, HIGH); //APAGADO
    MANUALTEMP = true;
    PIDTEMP = false;
    SERIALCONNECT = true;
    CONTROLA = false;
}

if (data.substring(0, 1) == "Q") { // MODO CONTROL PID TEMPERATURA
    //sensorT();
    buzzer.noTone();
}

```

```

digitalWrite(STBY, HIGH);
digitalWrite(EN, HIGH);
luz = data.substring(1).toInt(); // Leer el número
Serial.println(b);
ledcWrite(PWM_CHANNEL2, 0);
ledcWrite(PWM_CHANNEL, luz);
SERIALCONNECT = true;
PIDTEMP = true;
MANUALTEMP = false;
CONTROLA = false;
if (tempC >= 40) {
    digitalWrite(ven, HIGH);
} else {
    digitalWrite(ven, LOW);
}
}

if (data.substring(0, 1) == "q") { // MODO CONTROL PID TEMPERATURA ventilador
    //sensorT();

    int ve = data.substring(1).toInt(); // Leer el número
    if (ve == 1) {
        if (!undatoven) {
            undatoven2 = false;
            ventilador = true;
            //digitalWrite(ven, HIGH);
            undatoven = true;
        }
    }
    if (ve == 0) {
        if (!undatoven2) {
            undatoven = false;
            ventilador = false;
            //digitalWrite(ven, HIGH);
            undatoven2 = true;
        }
    }
}

```

```

    // digitalWrite(ven, LOW);
}
}

if (PIDTEMP) {
    if (ventilador) {
        digitalWrite(ven, HIGH);
    } else {
        digitalWrite(ven, LOW);
    }
}

if (data.substring(0, 1) == "G") { // MODO SSR
    buzzer.noTone();
    PIDTEMP = false;
    digitalWrite(STBY, LOW);
    digitalWrite(EN, HIGH);
    MANUALTEMP = false;
    PIDTEMP = false;
    SERIALCONNECT = true;
    CONTROLA = false;
}

if (data.substring(0, 1) == "M") { // MODO CONTROL MANUAL VELOCIDAD
    motor = data.substring(1).toInt(); // Leer el número
    buzzer.noTone();
    Serial.println(x);
    digitalWrite(STBY, HIGH);
    digitalWrite(EN, HIGH);
    ledcWrite(PWM_CHANNEL, 0);
    ledcWrite(PWM_CHANNEL2, motor);
    ledcWrite(PWM_CHANNEL, 0);
    SERIALCONNECT = true;
    MANUALTEMP = false;
    PIDTEMP = false;
    CONTROLA = false;
}

```

```

}
if (data.substring(0, 1) == "N") { // MODO CONTROL PID VELOCIDAD
    motor = data.substring(1).toInt(); // Leer el número
    buzzer.noTone();
    Serial.println(x);
    digitalWrite(STBY, HIGH);
    digitalWrite(EN, HIGH);
    ledcWrite(PWM_CHANNEL, 0);
    ledcWrite(PWM_CHANNEL2, motor);
    ledcWrite(PWM_CHANNEL, 0);
    SERIALCONNECT = true;
    MANUALTEMP = false;
    PIDTEMP = false;
    CONTROLA = false;
}

```

```

if (data.substring(0, 1) == "C") { // ssr1
    ssr1 = data.substring(1).toInt(); // ssr2
    buzzer.noTone();
    if (ssr1 == 1) {
        digitalWrite(CHA1, LOW);
    }
    if (ssr1 == 0) {
        digitalWrite(CHA1, HIGH);
    }
}
if (data.substring(0, 1) == "D") { // Si el primer caracter es 'L'
    buzzer.noTone();
    ssr2 = data.substring(1).toInt(); // Leer el número
    if (ssr2 == 1) {
        digitalWrite(CHA2, LOW);
    }
    if (ssr2 == 0) {
        digitalWrite(CHA2, HIGH);
    }
}

```

```

}

// PROCESO DE CONTROL DE ANGULO
if (CONTROLA) {
  // control de posicion inicial
  if (!iniposi) {
    INI_POSI();
    current_angle=0;
  }

  if (seriac == false) {
    Serial.println("Conectado");
    seriac = true;
  }

  if (data.substring(0, 1) == "g") { // Si el primer caracter es 'g' (grados)
    GIRAR = true;
    target_angle = data.substring(1).toInt(); // Leer el número de grados
  }
  if (data.substring(0, 1) == "t") { // Si el primer caracter es 't' (tiempo)
    timest = data.substring(1).toInt(); // Leer el número de tiempo
  }
  if (GIRAR) {
    // Calcula numero necesario de pasos
    steps_to_move = abs((target_angle - current_angle) * steps_per_rev / 360);
    Serial.println("pasos: " + String(steps_to_move));
    // Dirección basada en angulo objetivo
    if (steps_to_move > 0) {
      pasosm = steps_to_move;
    }
    if (target_angle > current_angle) {
      sentido = 1;
      //Serial.println("horario");
      digitalWrite(dir_pin, HIGH);
    }
  }
}

```

```

if (target_angle < current_angle) {
    sentido = -1;
    //Serial.println("anti horario");
    digitalWrite(dir_pin, LOW);
}

// Moviendo motor al objetivo
for (int i = 0; i < steps_to_move; i++) {
    digitalWrite(step_pin, HIGH);
    delayMicroseconds(timest);
    digitalWrite(step_pin, LOW);
    delayMicroseconds(timest);
}
// Actualizar el angulo actual
current_angle = target_angle;
//Calculo los pulsos del encoder

GIRAR = false;
v1 = String(current_angle);
v2 = String(timest);
v3 = String(sentido);
v4 = String(pasosm);
String a = v1 + "," + v2 + "," + v3 + "," + v4;
Serial.println(a);
}
} else {
if (contar2 == false) {
    unsigned long currentMillis = millis();

    if (currentMillis - previousMillis >= interval) {
        // save the last time you blinked the LED
        previousMillis = currentMillis;
        serialno += 1;
    }
}
}

```

```

if (serialno == 2) {
  desco = 0;
  SERIALCONNECT = false;
  contar2 = true;
  serialno = 1001;
  iniposi = false;
  digitalWrite(CHA2, HIGH);
  digitalWrite(CHA1, HIGH);
  PIDTEMP = true;
  MANUALTEMP = false;
}
}
}

void INI_POSI() {
  digitalWrite(dir_pin, LOW);
  // Moviendo motor al objetivo
  for (int i = 0; i < 900; i++) {
    if (contador >= 1) {
      iniposi = true;
      Serial.println("detener motor ");
      contador = 0;
      break;
    } else {
      digitalWrite(step_pin, HIGH);
      delayMicroseconds(timest);
      digitalWrite(step_pin, LOW);
      delayMicroseconds(timest);
    }
  }
}

void toneinicio() {

  buzzer.tone(NOTE_G4, 90);
}

```



```
buzzer.tone(NOTE_E4, 90);
buzzer.tone(NOTE_G4, 90);
buzzer.tone(NOTE_A4, 120);
delay(120);
buzzer.noTone();
}

void toneconnect() {
  ToneESP32 buzzer(BUZZER_PIN, BUZZER_CHANNEL);
  buzzer.tone(NOTE_G6, 90);
  buzzer.tone(NOTE_E7, 90);
  buzzer.tone(NOTE_G7, 90);
  buzzer.tone(NOTE_A7, 120);
  delay(120);
  buzzer.noTone();
}

void tonenoconnect() {
  //no se conecta
  ToneESP32 buzzer(BUZZER_PIN, BUZZER_CHANNEL);
  buzzer.tone(NOTE_F3, 120);
  buzzer.tone(NOTE_F4, 120);
  buzzer.tone(NOTE_D3, 120);
  buzzer.tone(NOTE_D4, 120);
  buzzer.tone(NOTE_DS3, 120);
  buzzer.tone(NOTE_DS4, 120);
  delay(120);
  buzzer.noTone();
}

void alarm() {
  ToneESP32 buzzer(BUZZER_PIN, BUZZER_CHANNEL);

  buzzer.tone(NOTE_B5, 150);
  buzzer.tone(NOTE_G5, 150);
  delay(120);
```

```
buzzer.noTone();  
}  
  
void alarm2() {  
  ToneESP32 buzzer(BUZZER_PIN, BUZZER_CHANNEL);  
  buzzer.tone(NOTE_B5, 150);  
  buzzer.tone(NOTE_B5, 150);  
  buzzer.tone(NOTE_G5, 150);  
  buzzer.tone(NOTE_B5, 150);  
  buzzer.tone(NOTE_G5, 150);  
  buzzer.tone(NOTE_B5, 150);  
  buzzer.tone(NOTE_G5, 150);  
  delay(120);  
  buzzer.noTone();  
}  
  
void detener() {  
  digitalWrite(STBY, LOW);  
  digitalWrite(EN, HIGH);  
}
```



ESCUELA SUPERIOR POLITÉCNICA DE  
CHIMBORAZO

DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL  
APRENDIZAJE



UNIDAD DE PROCESOS TÉCNICOS  
REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 09/ 11 / 2023

<b>INFORMACIÓN DE LOS AUTORES</b>
<b>Nombres – Apellidos:</b> DANIEL MAURICIO CEVALLOS CALVOPIÑA
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> INFORMÁTICA Y ELECTRÓNICA
<b>Carrera:</b> ELECTRÓNICA Y AUTOMATIZACIÓN
<b>Título a optar:</b> INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN
<b>f. Analista de Biblioteca responsable:</b>  Ing. Fernanda Arévalo M.

