

UNIVERSIDAD RICARDO PALMA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE
BAÑERA CASERA AUTOMATIZADA PARA CANES Y DE
BAJO COSTO**

TESIS
**PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO**

PRESENTADA POR

Bach. GARCIA BLONDET, CARLOS PIERO

Asesor: Dr. Ing. HUAMANÍ NAVARRETE, PEDRO FREDDY

LIMA-PERÚ

2022

METADATOS COMPLEMENTARIOS

Datos de autor

1. GARCIA BLONDET, CARLOS PIERO
DNI
70166789

Datos de asesor

1. Huamani Navarrete, Pedro Freddy
DNI
10032682

Datos del jurado

1. González Prado, Julio Cesar; 07702235; 0000-0003-0384-7015
2. Terukina Oshiro, Nelly Luz; 07808963; 000-0002-9654-7961
3. Hinojosa Sánchez, Raúl Sixto; 08605661; 0000-0002-6507-0224
4. Rivas León, Javier Hipólito; 10250991; 0000-0002-8365-4346

Datos de la investigación

Campo del conocimiento OCDE: 2.02.01

Código del Programa: 712097

DEDICATORIA

Dedico esta tesis en primer lugar a Dios y a Nuestra Señora por regalarme el don de la vida y todas las gracias que recibo. En segundo lugar, a mis Padres Carlos Garcia y Rosa Blondet por darme la oportunidad de darme estudios, brindarme su apoyo incondicional en cada etapa de mi vida y entregarme su amor incondicional. También a mis hermanos Daniel y Ximena por el aliento y el apoyo en todo momento y finalmente a mi novia Carolina Higa por el gran empuje que siempre da en cada proyecto que me propongo a realizar y en especial en este proyecto de tesis. A cada uno dedico este trabajo con mucho amor.

Carlos Piero Garcia Blondet

AGRADECIMIENTO

Agradezco en primer lugar a Dios por brindarme todo lo que tengo y por ello poder desarrollarme. Así como también a mis padres Carlos y Rosa por darme la oportunidad de estudiar, a mis hermanos Daniel y Ximena y mi novia Carolina por todo su apoyo.

También agradezco a mi asesor Pedro Huamaní por su apoyo y consejos para el desarrollo de mi Tesis.

Carlos Piero Garcia Blondet

ÍNDICE GENERAL

ÍNDICE DE FIGURAS	ix
RESUMEN	ix
ABSTRACT	x
INTRODUCCIÓN	1
CAPÍTULO I: PLANTEAMIENTO Y DELIMITACIÓN DEL PROBLEMA	2
1.1. Formulación del problema	2
1.1.1. Problema General.....	2
1.1.2. Problemas Específicos.....	2
1.2. Objetivos	3
1.2.1. Objetivo General	3
1.2.2. Objetivos Específicos.....	3
1.3. Importancia y justificación	3
1.4. Limitaciones del estudio	3
1.4.1. Espacial.....	4
1.4.2. Económico	4
1.4.3. Implementación.....	4
CAPÍTULO II: MARCO TEÓRICO / FUNDAMENTOS TEÓRICOS	5
2.1. Marco Histórico	5
2.2. Investigaciones relacionadas con el Tema	5
2.3. Estructura teórica y científica que sustenta el estudio.....	10
2.3.1. Sistemas de Control.....	10
2.3.2. Sensor de Temperatura DS18B20	12
2.3.3. Sensor de Nivel HC-SR04.....	14
2.3.4. Arduino Uno	16
2.3.5. LabVIEW	18
2.3.6. Electroválvula	19
2.3.7. Electrobomba.....	20
2.3.8. Módulo Relé	21
2.3.9. Resistencia Calefactora de Inmersión.....	22
2.4. Definición de términos básicos	22
2.5. Diseño de la Investigación.....	23
2.5.1. Variables de investigación	23

2.5.2. Tipo y Método de investigación.....	24
2.5.3. Técnicas e Instrumentos de recolección de datos	24
2.5.4. Procedimiento para la recolección de datos	25
CAPÍTULO III: DISEÑO DEL PROTOTIPO DE BAÑERA.....	26
3.1. Dimensionamiento del Prototipo de Bañera para Canes	26
3.2. Diseño e Implementación de la Estructura del Prototipo de Bañera	29
3.2.1. Diseño e Implementación de la Parte Estructural	29
3.2.2. Implementación de la parte de Plomería.....	34
3.2.3. Implementación de la parte Eléctrica.....	36
3.2.4. Implementación de la parte Electrónica	38
3.3. Implementación del controlador ON/OFF de nivel.....	52
3.4. Implementación y sintonización del controlador PID de temperatura.....	55
3.4.1. Implementación del controlador PID.....	55
3.4.2. Sintonización del controlador PID	57
3.5. Implementación del controlador secuencial	61
3.6. Implementación de la interfaz de usuario de la bañera casera automatizada para canes	62
CAPÍTULO IV: PRUEBAS Y RESULTADOS.....	64
4.1. Pruebas y resultados del controlador ON/OFF de nivel	64
4.1.1. Análisis del resultado para un SP= 20L	64
4.1.2. Análisis del resultado para un SP= 40L	64
4.1.3. Análisis del resultado para un SP=60L.....	65
4.1.4. Análisis del resultado para un SP=80L.....	66
4.2. Pruebas y resultados del controlador PID de temperatura.....	67
4.2.1. Análisis del Resultado para un SP=25 °C:	67
4.2.2. Análisis del resultado para un SP= 30 °C:	68
4.2.3. Análisis del resultado para un SP= 37 °C:	69
4.3. Pruebas y resultados del controlador secuencial.....	71
4.3.1. Análisis del resultado para T= 90 Segundos	71
4.3.2. Análisis del resultado para T= 120 Segundos.....	72
4.3.3. Análisis del resultado para T=150 Segundos	73
4.4. Pruebas y resultados finales	74
4.5. Presupuesto.....	78
CONCLUSIONES	80

RECOMENDACIONES.....	82
REFERENCIAS BIBLIOGRÁFICAS.....	83
ANEXOS.....	86

ÍNDICE DE FIGURAS

Figura N° 01: Respuesta real de la planta para un Set Point de 28 °C	06
Figura N° 02: Diagrama de Bloques Control PID	10
Figura N° 03: Gráfica de un Control ON/OFF	11
Figura N° 04: Sensor DS18B20.....	12
Figura N° 05: Gráfico de rango de Tolerancia del Sensor DS18B20.....	13
Figura N° 06: Pines de Conexión del sensor DS18B20	14
Figura N° 07: Sensor de Ultrasonido HC-SR04.....	15
Figura N° 08: Tarjeta Arduino Uno.....	17
Figura N° 09: Diagrama de Bloques.....	18
Figura N° 10: Panel Frontal	19
Figura N° 11: Electroválvula	20
Figura N° 12: Eletrobomba de 0.5hp.....	20
Figura N° 13: Módulo Réle	21
Figura N° 14: Resistencia de Inmersión	22
Figura N° 15: Representación pictográfica del Prototipo de Bañera.....	27
Figura N° 16: Diagrama P&ID del Prototipo de Bañera.....	27
Figura N° 17: Diagrama de Flujo del Prototipo de Bañera.....	28
Figura N° 18: Diseño en 3D de Base del Prototipo de Bañera	30
Figura N° 19: Implementación de la Base del Prototipo de Bañera	30
Figura N° 20: Diseño 3D de la Cabina del Prototipo de Bañera.....	31
Figura N° 21: Implementación de la Cabina del Prototipo de Bañera.....	32
Figura N° 22: Diseño 3D de la Bañera	33
Figura N° 23: Implementación de la Estructura de PVC del Prototipo de Bañera	33
Figura N° 24: Tanque de agua	34
Figura N° 25: Ingreso de agua al Prototipo de Bañera.....	34
Figura N° 26: Entrada y Salida de la Electrobomba de agua.....	35
Figura N° 27: Circuito de distribución ingreso de agua y shampoo.....	36
Figura N° 28: Esquema Eléctrico de los circuitos eléctricos	37
Figura N° 29: Fuente DC de 12V.....	38
Figura N° 30: Sensor de Nivel Ultrasónico.....	38
Figura N° 31: Sensor de Nivel y Temperatura	39
Figura N° 32: Sensor Temperatura Ambiental	40

Figura N° 33: Electroválvulas implementadas en el Prototipo	41
Figura N° 34: Módulo Relé para las Electroválvulas.....	41
Figura N° 35: Electrobomba de Agua.....	42
Figura N° 36: Relé estado sólido y Módulo Relé para Electrobomba.....	43
Figura N° 37: Resistencia implementada en el tanque.....	44
Figura N° 38: Diagrama de bloques del circuito dimmer.....	45
Figura N° 39: Esquemático del circuito dimmer	45
Figura N° 40: Esquemático circuito detector cruce por cero	46
Figura N° 41: Esquemático circuito de potencia	46
Figura N° 42: Tarjetas PCB de control y Potencia para el dimmer.....	47
Figura N° 43: Termostato de Respaldo.....	47
Figura N° 44: Tarjeta Interfaz Sensores y Actuadores.....	48
Figura N° 45: Tarjeta de Adquisición de Datos Arduino	49
Figura N° 46: Arduino y Filtro Pasabajo	51
Figura N° 47: Implementación final del prototipo de bañera para canes	51
Figura N° 48: Funcionamiento del Sensor de Ultrasonido en LabVIEW	52
Figura N° 49: Gráfica Volúmen vs Litros de los datos Registrados.....	53
Figura N° 50: Programación en LabVIEW del Controlador ON/OFF de Nivel.....	54
Figura N° 51: Panel Frontal de la Implementación del Controlador ON/OFF de Nivel.	55
Figura N° 52: Funcionamiento de Sensor de Temperatura en LabVIEW	55
Figura N° 53: Programación en LabVIEW del Controlador PID de Temperatura.....	56
Figura N° 54: Panel Frontal Implementación del Controlador PID de Temperatura.....	56
Figura N° 55: Tabla Parámetros de Ganancia controlador PID	57
Figura N° 56: Panel Frontal Implementación Controladores ON/OFF y PID.....	58
Figura N° 57: Controlador PID con parámetros $K_p=0.2$, $T_i=0.05$ y $T_d=0$	59
Figura N° 58: Controlador PID con parámetros $K_p=0.2$, $T_i=0.001$ y $T_d=0$	60
Figura N° 59: Controlador PID con parámetros $K_p=0.5$, $T_i=0.001$ y $T_d=0.02$	61
Figura N° 60: Programación Controlador Secuencial	61
Figura N° 61: Panel Frontal de la Implementación del controlador Secuencial.....	62
Figura N° 62: Implementación de la Interfaz para el Usuario.....	63
Figura N° 63: Respuesta del Controlador ON/OFF de Nivel $SP=20L$	64
Figura N° 64: Respuesta del Controlador ON/OFF de Nivel $SP=40L$	65
Figura N° 65: Respuesta del Controlador ON/OFF de Nivel $SP=60L$	65
Figura N° 66: Respuesta del Controlador ON/OFF de Nivel $SP=80L$	66

Figura N° 67: Gráfica del Controlador ON/OFF.....	66
Figura N° 68: Condiciones Iniciales para Controlador PID	67
Figura N° 69: Respuesta del controlador PID de Temperatura SP=25°C.....	68
Figura N° 70: Gráfica del Controlador PID para SP=25°C.....	68
Figura N° 71: Respuesta del controlador PID de Temperatura SP=30°C.....	69
Figura N° 72: Gráfica del Controlador PID para SP=30°C.....	69
Figura N° 73: Respuesta del controlador PID de Temperatura SP=37°C.....	70
Figura N° 74: Gráfica del Controlador PID para SP=37°C.....	70
Figura N° 75: Panel Frontal Inicial del Control Secuencial	71
Figura N° 76: Respuesta del Controlador Secuencial para un T=90 Segundos	72
Figura N° 77: Respuesta del Controlador Secuencial para un T=120 Segundos	72
Figura N° 78: Respuesta del Controlador Secuencial para un T=150 Segundos	73
Figura N° 79: Controlador Secuencial - Secuencia "Mojado"	74
Figura N° 80: Configuración inicial para bañado de "Ramón"	75
Figura N° 81: Control de Nivel y Temperatura para el Bañado de "Ramón"	75
Figura N° 82: "Ramón" antes de iniciar el bañado	76
Figura N° 83: "Ramón" en proceso de bañado.....	76
Figura N° 84: Proceso de bañado finalizado	77

ÍNDICE DE TABLAS

Tabla N° 1: Tabla Resoluciones DS18B20.....	13
Tabla N° 2: Tabla Resistencias Pull-Up para el sensor DS18B20.....	14
Tabla N° 3: Especificaciones Técnicas del Arduino Uno	17
Tabla N° 4: Distribución eléctrica para actuadores	36
Tabla N° 5: Distribución de interruptores termomagnéticos	37
Tabla N° 6: Distribución de Pines en Arduino Uno	50
Tabla N° 7: Registro de Datos Distancia y Volúmen	53
Tabla N° 8: Resumen de Secuencia de Lavado.....	62
Tabla N° 9: Resumen de Resultados del Controlador ON/OFF	67
Tabla N° 10: Resumen de Resultados del Controlador PID de Temperatura	71
Tabla N° 11: Resumen de Resultados del Controlador Secuencial	73
Tabla N° 12: Tabla de Presupuesto.....	78

RESUMEN

El presente trabajo de investigación describe el diseño e implementación de un prototipo de bañera casera automatizada para canes y de bajo costo, con la finalidad de controlar el nivel y la temperatura de agua necesitada; así como también, la secuencia automatizada para el proceso de bañado de un can. Para ello, el prototipo diseñado contó con 3 etapas de implementación, la primera fue la construcción de la estructura de la bañera que consistió en una base de madera, una cabina de planchas de acrílico y la estructura de la bañera realizado con tubos de PVC. Luego, se implementó la etapa de plomería, eléctrica y electrónica, esta última consistió en la instalación de sensores de nivel y temperatura, tres electroválvulas, una resistencia sumergible, una electrobomba, el módulo Arduino que tiene la función de tarjeta interfaz entre los sensores y actuadores con el software LabVIEW, y tarjetas intermedias o interfaz entre los sensores y el módulo Arduino. Finalmente, la última etapa consistió en la programación y puesta en marcha del control ON/OFF de nivel de agua, control PID de temperatura de agua y control Secuencial del bañado del can; esta etapa se realizó por medio de una interfaz gráfica de usuario mediante el software LabVIEW. Asimismo, para la implementación del controlador ON/OFF se logró encontrar una ecuación que satisfaga la conversión de centímetros entregados por el sensor a la unidad de litros, con ello el controlador obtuvo una buena respuesta con un error de 0.42 litros y el tiempo de llenado fue de 7.18 minutos. Por otro lado, la sintonización del controlador PID de temperatura se realizó por el método de prueba y error donde las constantes de ganancia del controlador fueron: $K_p=0.5$, $T_i=0.001$ y $T_d=0.02$; donde se observó que el tiempo para alcanzar la temperatura deseada fue de 45.35 minutos. Finalmente, el controlador secuencial para el proceso del bañado del can tomó 115 segundos donde se consumió un total de 26.22 litros de agua.

Palabras claves: LabVIEW, Módulo Arduino, Control ON/OFF, Control PID, Control Secuencial.

ABSTRACT

This research work describes the design and implementation of a low-cost, automated home bathtub prototype for dogs, in order to control the level and temperature of water needed; as well as the automated sequence for the bathing process of a dog. For this, the prototype designed had 3 implementation stages, the first was the construction of the structure of the bathtub that consisted of a wooden base, a cabin made of acrylic sheets and the structure of the bathtub made with PVC pipes. Then, the plumbing, electrical and electronic stage was implemented, the latter consisted of the installation of level and temperature sensors, three solenoid valves, a submersible resistance, an electric pump, the Arduino module that has the function of an interface card between the sensors and actuators with LabVIEW software, and intermediate cards or interface between the sensors and the Arduino module. Finally, the last stage consisted in the programming and start-up of the ON / OFF control of the water level, PID control of the water temperature and sequential control of the bathing of the dog; this stage was carried out through a graphical user interface using LabVIEW software. Likewise, for the implementation of the ON / OFF controller, it was possible to find an equation that satisfies the conversion of centimeters delivered by the sensor to the unit of liters, with this the controller obtained a good response with an error of 0.42 liters and the filling time it was 7.18 minutes. On the other hand, the tuning of the PID temperature controller was performed by the test and error method where the controller gain constants were: $K_p = 0.5$, $T_i = 0.001$ and $T_d = 0.02$; where it was observed that the time to reach the desired temperature was 45.35 minutes. Finally, the sequential controller for the dog bathing process took 115 seconds where a total of 26.22 liters of water was consumed.

Keywords: LabVIEW, Arduino module, ON/OFF Control, PID Control, Sequential Control.

INTRODUCCIÓN

Para el desarrollo de la presente tesis de investigación “Diseño e implementación de un prototipo de bañera casera automatizada para canes y de bajo costo”, se consideraron los siguientes antecedentes Fuente (2017) *Diseño e Implementación de un módulo educativo para el control de temperatura*. Así mismo, Bohórquez, Fonseca y Gutiérrez (2017) *Sistema didáctico para el control de nivel con tanques acoplados*. Finalmente, la investigación de Vizcaíno (2017) *Diseño e implementación de un sistema de calentamiento y ahorro de agua en Domicilio*.

De igual forma, para la investigación se ha formulado el problema general de la siguiente manera: ¿Cómo se diseñará e implementará un prototipo de bañera casera automatizada para canes, usando el software LabVIEW con interfaz Arduino y a un bajo costo?, interrogante que a través de la investigación se ha dado respuesta.

Y como también, el motivo fundamental de la investigación considera controlar la variable independiente, control PID, control ON/OFF y control Secuencial, y como repercuten en la variable dependiente que es el prototipo de bañera casera automatizada; con ello se desea brindar una mejor manera de ayudar a bañar a canes ahorrando el consumo de agua y energía; así como también dando al can un mejor confort.

Así mismo este proyecto de investigación se estructuró de la siguiente manera. En el Capítulo 1 se presenta la formulación del problema, el problema general y los problemas específicos, el objetivo general y los específicos, así como también la de señalar la importancia y justificación del proyecto y, por último, se menciona las limitaciones. En el Capítulo 2, se presentan investigaciones y antecedentes que sustentan las variables independientes y dependientes del presente trabajo de investigación, así como la base teórica. Luego, en el Capítulo 3, se describe el desarrollo e implementación del prototipo de bañera, donde se detalla cada etapa que tuvo el proyecto para el cumplimiento de los objetivos planteados. Igualmente, en el Capítulo 4, se muestran las pruebas realizadas y los resultados de cada etapa de control. Y, para finalizar, se redactan las recomendaciones, conclusiones y se describen las referencias bibliográficas utilizadas para la presente investigación.

CAPÍTULO I: PLANTEAMIENTO Y DELIMITACIÓN DEL PROBLEMA

1.1. Formulación del problema

En muchos hogares que adoptan o tienen a alguna mascota en particular, como los canes, el baño se puede tornar difícil, debido a que a estos animales les incomoda el agua, ya sea por la temperatura, la forma como se está realizando el baño o que hayan tenido una mala experiencia con el baño que los pone nerviosos.

Parte importante del problema de estos baños es el exceso de consumo de agua y el alto consumo eléctrico. Debido a eso se plantea implementar un prototipo de bañera automatizada para canes, utilizando sensores comerciales y un tipo de control PID para controlar la temperatura y un control ON/OFF para controlar la cantidad de agua a utilizar.

1.1.1. Problema General

¿Cómo se diseñará e implementará un prototipo de bañera casera automatizada para canes, usando el software LabVIEW con interfaz Arduino y a un bajo costo?

1.1.2. Problemas Específicos

- a) ¿Cómo se diseñará e implementará un prototipo de bañera para canes con dimensiones de 80cm de largo x 60cm de ancho y 65 cm, de tal manera que sea soportada por una estructura de madera y cubierto de planchas de acrílico?
- b) ¿Cómo se implementará y sintonizará un control PID para controlar la temperatura del agua en el prototipo de bañera automatizada usando el software LabVIEW y el módulo Arduino como interfaz?
- c) ¿Cómo se implementará un control ON/OFF para controlar el nivel de agua a utilizar en el prototipo de bañera automatizada usando el software LabVIEW y el módulo Arduino como interfaz?
- d) ¿Cómo se realizará la automatización del prototipo de bañera para canes, utilizando un control secuencial para la apertura y cerrado de 2 válvulas solenoides?

1.2. Objetivos

1.2.1. Objetivo General

Diseñar e implementar un prototipo de bañera casera automatizada para canes, usando el software LabVIEW con interfaz Arduino y a un bajo costo.

1.2.2. Objetivos Específicos

- a) Diseñar e Implementar un prototipo de bañera para canes con dimensiones de 80 cm de largo x 60 cm de ancho y 70 cm, de tal manera que sea soportada por una estructura de madera y cubierto de planchas de acrílico.
- b) Implementar y sintonizar, algorítmicamente, un control PID para controlar la temperatura del agua en el prototipo de bañera automatizada usando el software LabVIEW y el módulo Arduino como interfaz.
- c) Implementar un control ON/OFF para controlar el nivel de agua a utilizar en el prototipo de bañera automatizada usando el software LabVIEW y el módulo Arduino como interfaz.
- d) Realizar la automatización del prototipo de bañera casera para canes, utilizando un control secuencial para la apertura y cerrado de 2 válvulas solenoides.

1.3. Importancia y justificación

Debido al alto consumo de agua y los altos niveles de estrés y ansiedad que provoca un baño en los canes, se planteó este proyecto de tesis que tuvo como finalidad el diseño e implementación de un prototipo de bañera casera automatizada para canes, y de bajo costo. Asimismo, la realización de este proyecto de tesis se justificó por la facilidad de adquisición y montaje de los sensores de temperatura y nivel; además, por la viabilidad de la implementación y sintonización de un controlador PID, así como la implementación del control ON/OFF, ambos en el software LabVIEW.

1.4. Limitaciones del estudio

Las limitaciones de este proyecto de tesis planteado, en cuanto a la implementación del prototipo de bañera para canes, fueron clasificadas de la siguiente manera:

1.4.1. Espacial

Debido a la situación sanitaria actual se tuvo algunas limitaciones en cuanto a instrumentos y herramientas para la implementación, tales como osciloscopio, fuentes de voltaje y la adquisición de ciertos sensores y actuadores.

1.4.2. Económico

Otra limitación se encontró en la parte económica, por esa razón no se llegó a utilizar componentes electrónicos y controladores del tipo industrial. Mas bien, se utilizaron del tipo comercial y por ello presentaron restricciones de funcionamiento en situaciones hostiles, tal como lo indican sus respectivas hojas técnicas.

1.4.3. Implementación

La dimensión de la bañera se adaptó para bañar a canes de razas pequeñas como Shitzu y alguna otra raza en general que no excedan a 60cm de largo y 50 cm de alto. Asimismo, dicha dimensión fue limitada para evitar el uso excesivo de agua en el momento de la realización de las pruebas, así como también en el costo del material utilizado en la construcción de la misma.

Este trabajo de tesis se limitó solamente a la implementación de la etapa de bañado, y no de la etapa de secado.

Además, se limitó a la implementación de los controladores PID y ON/OFF de forma algorítmica utilizando el software LabVIEW.

CAPÍTULO II: MARCO TEÓRICO / FUNDAMENTOS TEÓRICOS

2.1. Marco Histórico

El bañado de canes es una actividad que se realiza cada vez que el animal lo requiera o que haya sido recomendado por parte del veterinario. Por lo cual, tradicionalmente, el baño era realizado en casa por los propios dueños y con poca frecuencia, sea utilizando un lavadero, una ducha de casa, un jardín, una cubeta o algún medio adaptado para tal situación. Asimismo, tales baños eran ejecutados sin limitar la cantidad de agua y el uso de los útiles de aseo; así como también sin considerar la incomodidad que manifestaban este grupo de mascotas. Con el transcurrir de los años, surge la carrera Medicina Veterinaria y con ello los Consultorios Médicos Veterinarios capaces de atender de la manera más apropiada a los animales, sea en caso de enfermedad, accidentes o como también en situaciones de aseo y belleza. Por esa razón, y por el avance tecnológico en la actualidad se han desarrollados diferentes tipos de bañeras automatizadas con características particulares y muy especializadas, sea con hidromasajes, bañados medicados entre otros; sin embargo, la adquisición de estas bañeras resulta muy caro para el estándar de la población. Por ello, este proyecto de tesis plantea una bañera de bajo costo.

2.2. Investigaciones relacionadas con el Tema

Fuentes (2017), en su tesis: Diseño e Implementación de un módulo educativo para el control de temperatura

Logró adicionar un lazo de control de temperatura al módulo de control de nivel existente del Centro de Tecnologías Avanzadas de Manufactura (CETAM) de la Universidad Pontificia Católica del Perú. Para la implementación del lazo de control de temperatura el autor agregó: resistencias calefactoras (actuadores) y un sensor de temperatura LM35. Así como también se implementaron circuitos de acondicionamiento para las resistencias, debido a que funcionan con energía AC. El control implementado para el lazo de control fue un control PID de temperatura, donde se obtuvo los modelos matemáticos para poder sintonizar el control PID. Finalmente se desarrolló una interfaz gráfica de usuario y se implementó el algoritmo de control PID mediante el software LabVIEW. En las pruebas realizadas el autor utilizó un Set Point de 28 °C y sintonizó el control PID con los siguientes parámetros: $K_p= 8$, $T_i= 200$ y $T_d=8$. Con estos valores

consiguió eliminar el sobreimpulso en la respuesta, un error en estado estable de cero y se pudo mantener el tiempo de establecimiento de 300s, como se muestra en la Figura N°1.

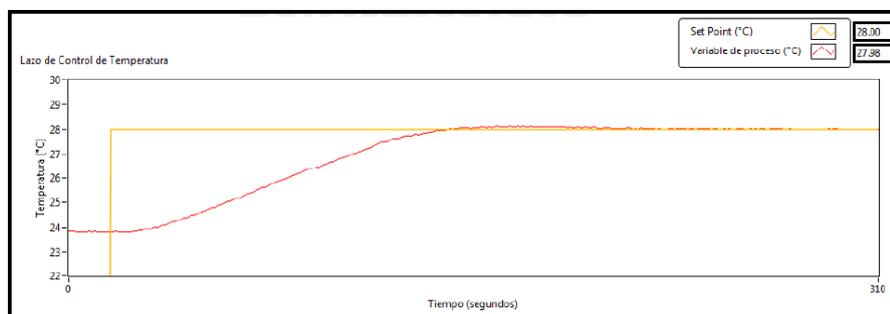


Figura N°1 Respuesta real de la planta para un Set Point de 28 °C

Fuente: Fuentes, 2017, p.60

El autor concluye que se pudo lograr implementar correctamente el control PID alcanzando los requerimientos de diseño de sobreimpulso, tiempo de establecimiento y en error en estado estable.

Bohorquez, Fonseca y Gutiérrez (2017), es su tesis: Sistema didáctico para el control de nivel con tanques acoplados

Logran implementar un sistema de 3 tanques acoplados, los cuales los autores denominaron N1, N2 y tanque reservorio. A este sistema se implementó un control ON/OFF de nivel y un control PID de flujo, los cuales fueron realizados en los tanques N1 y N2. Estos tanques se unieron por medio de tuberías, 1 válvula automática (válvula solenoide) y por 3 válvulas de manuales.

El control PID de flujo utilizó: 1 electrobomba sumergible instalada en el tanque reservorio, cuya función fue la de llevar el líquido hacia el tanque N1 y 1 un sensor de flujo YF-S201 para medir el flujo del líquido que está ingresando al tanque. Los autores utilizaron el software Matlab para encontrar las constantes de ganancia del controlador PID, las cuales fueron: $K_p = 805.402$, $K_i = 119.014$ y $K_d = 1361.4$. Con estas constantes el sistema pudo responder correctamente.

Por otro lado, para el control ON/OFF de nivel se utilizó 1 sensor de ultrasonido HC-SR04 para determinar el nivel del líquido en los tanques N1 y N2. El Set Point utilizado para los tanques fueron: Tanque N1: 9 Litros y Tanque N2: 7.5 Litros. Es preciso, entender que este sensor de ultrasonido mide en distancia, por ello los autores utilizaron una fórmula que convertía distancia en cm a litros.

Esta fórmula fue hallada con el software Excel e ingresada posteriormente, al programa realizado en LabVIEW.

El sistema fue implementado en una interfaz gráfica con el software LabVIEW y se utilizó un módulo Arduino como interfaz entre los sensores y la computadora.

Los autores concluyen que se logró realizar exitosamente la tesis, y a su vez tendrá gran aporte para los alumnos que están aprendiendo sistemas de control industrial.

Vizcaíno (2017) en su trabajo de titulación: Diseño e implementación de un sistema de calentamiento y ahorro de agua en Domicilio

Logró diseñar un sistema de calentamiento de agua para una ducha utilizando un control PID de temperatura. El autor propone este control para minimizar el consumo eléctrico y de agua. El control PID de temperatura fue implementado en un módulo Arduino para controlar la temperatura de agua por medio un calentador de AC monofásico, para ello, fue necesario elaborar una etapa de electrónica de potencia para controlar la potencia de la resistencia. Para el control de potencia se utilizaron 2 etapas: La primera etapa fue la de cruce por cero, esto para poder sincronizar el microcontrolador y la fase AC, y así determinar el momento en que se activa el triac de la etapa 2, que es la etapa de potencia donde se encuentra la resistencia calefactora. Asimismo, el autor pudo sintonizar el control PID de temperatura utilizando el método empírico donde los valores de ganancia del controlador fueron $K_p=20$, $K_i=4$ y $K_d=0$ y el Set Point usado fue de 38 °C. Los resultados que el autor obtuvo fueron que la respuesta fue suficientemente rápida y sin elongación. El tiempo que la respuesta estabilizó fue de 20s.

El autor concluye que el control PID de temperatura logró alcanzar la temperatura deseada en 20 S, así como también se logró minimizar el desperdicio de agua de 3.99L a 0.66L.

Ayala y Donoso (2015), en su tesis: Diseño, modelación, simulación e implementación de un sistema de control de temperatura para la ducha eléctrica

Lograron implementar un sistema de control de lazo cerrado de temperatura y flujo con fin de ahorrar agua en las duchas eléctricas. El sistema propuesto por los autores constó de lo siguiente:

- Un controlador de temperatura de la marca Omega quien se encargó de realizar el control de temperatura de la ducha.
- Una termocupla tipo T como sensor de temperatura para el controlador
- Un sensor PIR para detectar la presencia de una persona y poder activar al sistema.
- Una servoválvula construida con una válvula de bola y un servomotor, para controlar el caudal del ingreso de agua a la ducha eléctrica.
- Un relé de estado Sólido para controlar la resistencia eléctrica de la ducha. La resistencia de la ducha tuvo un consumo de 5.5KW.
- Un módulo de interfaz entre el controlador de temperatura y todos los actuadores. Este módulo de interfaz contó con un microcontrolador Atmega 328. El microcontrolador fue programado en Arduino IDE.

De acuerdo a estudios que realizaron los autores, encontraron que el tiempo promedio de tomar una ducha es de 10 minutos y la temperatura promedio del agua para una ducha es de 35 °C. Por lo que, para su proyecto se tomó como Set Point 35 °C. El controlador fue programado para ser un controlador P y el valor de la ganancia usada fue de 2.5. Los autores lograron reducir hasta en un 30% en ahorro de agua usado para una ducha y un ahorro energético, por los que los autores concluyen que se obtuvo un resultado satisfactorio.

Agudelo y Hernández (2015) en su artículo: Control PID de nivel de agua usando LabVIEW y Arduino

Lograron diseñar e implementar una planta donde el objetivo principal fue la de controlar el nivel de agua. La Planta diseñada constó de 2 tanques de agua, uno sobre otro, además contó con un sensor de ultrasonidos, una servoválvula y una bomba de agua sumergible. El control que implementaron los autores fue un control PID, el cual se realizó en el programa LabVIEW utilizando la tarjeta Arduino como módulo interfaz entre la computadora, el sensor y actuadores. Los autores hicieron uso del software Matlab para la sintonización del control PID utilizando el Toolkit: "PID Tuning" luego de haber obtenido una estimación de

la función de transferencia también utilizando el Matlab con los valores de las mediciones manuales que realizaron previamente. Las constantes de las ganancias obtenidas del PID Tuning, fueron $K_p = 113.951$, $K_i = 1.3631$ y $K_d = 394.131$, estos valores fueron ingresados en el controlador PID que fue programado en el LabVIEW. Durante las pruebas reales realizadas los autores se percataron que el sistema conmutaba demasiado rápido, debido a que el controlador reaccionaba ante cualquier cambio que realizaba el sensor, sin embargo, se pudo solucionar ya que dieron con el problema que generaba esta acción del controlador. El causante fue la constante de la acción derivativa, ya que era demasiado alta, se cambió a un valor cercano a cero, ahora el valor para el $K_d = 1 \cdot 10^{-11}$, y con este cambio el sistema respondía mejor a las perturbaciones.

Los autores concluyeron que se pudo lograr realizar la comunicación entre la tarjeta Arduino como tarjeta de adquisición de datos y el LabVIEW y los beneficios que estas herramientas permiten ayudar a desarrollar y optimizar otros cualquier proceso industrial.

Matías y Ruíz (2014) en su tesis: Diseño e implementación de un sistema programable para la regulación de la temperatura de una regadera

Lograron implementar un sistema de control de temperatura para una regadera. Este sistema consistió en regular la temperatura del agua manipulando las válvulas para lograr obtener una temperatura deseada por el usuario y de esta forma evitar el desperdicio de agua que puede llegar hasta de 6 litros. Los autores implementaron 4 rango de temperatura para que el usuario pudiera escoger. Por ello, los autores propusieron que el diseño de la regadera sea de manera electrónica a través de 2 electroválvulas y 2 servomotores para lograr el control de la temperatura del agua. La etapa de control fue implementada en un módulo Arduino, el cual controló las electroválvulas y los servomotores, así como también del sensor de temperatura. El control consistió en hacer la mezcla de agua caliente con agua fría hasta llegar a la temperatura de confort o desea, de no llegar a esta temperatura se abre la electroválvula 1 y hace que el agua llegue a un reservorio. Cuando la temperatura alcance la temperatura desea entonces se apertura la electroválvula 2 y se inicia la ducha. Por otro lado, los 2 servomotores fueron implementados a 2 válvulas de paso donde uno se conectó a la toma de

agua caliente y la otra al agua fría. Los servomotores fueron programados para que abrirán a un ángulo determinado según la temperatura que el usuario ingreso. Los autores concluyeron que con la implementación de este proyecto se puede lograr realizar un gran ahorro de agua evitando de esta forma el desperdicio trayendo así un impacto ecológico positivo.

2.3. Estructura teórica y científica que sustenta el estudio

2.3.1. Sistemas de Control:

a) Control PID:

Según Delgado y Quinte (2019), un controlador PID es un método de control que actúa sobre un actuador para mantener estable un proceso, es decir para mantener el valor de una variable que es medido por un sensor en un valor deseado. Este controlador es muy utilizado en las industrias, debido a ser un controlador robusto y tiene muchas aplicaciones.

La estructura del controlador PID, está compuesta por 3 acciones: Proporcional, Integral y Derivativo. Estos son los denominados P, I, PI, PD y PID.

En la Figura N°2, se puede observar el diagrama de bloques de un proceso retroalimentado simple. Este proceso consta de: una Señal de entrada (Set Point), Controlador PID, Actuador, la Planta o proceso y el Sensor.

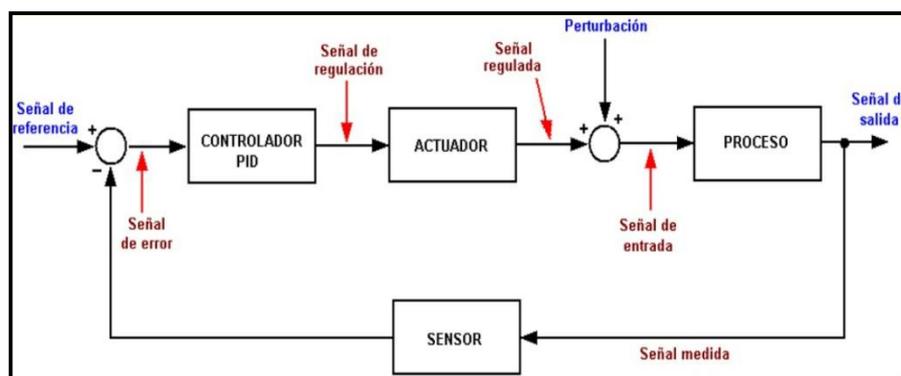


Figura N°2: Diagrama de Bloques Control PID

Fuente: Delgado y Quinte, 2018, p.19

En cuanto a la acción de este tipo de controlador PID, se afirma que reúne las ventajas de cada una de las tres acciones de control individual; y, está definir por la expresión señala en la ecuación N° 1.

$$u(t) = K_p e(t) + \frac{K_p}{T_I} \int_0^t e(r) dr + K_p T_D \frac{de(t)}{dt}$$

Ecuación N°1: Salida para el Control Proporcional Derivativo Integral

Con función de transferencia resultante:

$$C_{PID}(s) = K_P \left(1 + \frac{1}{T_I \cdot s} + T_D \cdot s \right)$$

Ecuación N°2: Función de transferencia Proporcional Integral Derivativo

Donde, K_p representar ganancia proporcional, T_i tiempo integral y T_d tiempo derivativo.

b) Control ON/OFF

Según Arteaga y Tobar (2014), un controlador ON/OFF se caracteriza por ser un sistema de dos posiciones que actúa sobre el elemento de actuación para el encendido o apagado como lo señala el nombre de este controlador. Este sistema es uno de los más utilizados en el ámbito doméstico e industrial por el bajo costo y su manejo.

Además, en un controlador ON/OFF, la señal de referencia o Set Point es el valor ingresado al sistema para ser comparado con el valor real de la variable del proceso, y este decidirá si enciende o no el actuador que conforma la planta. A continuación, en la Figura N°3 se observa el comportamiento del controlador ON/OFF.

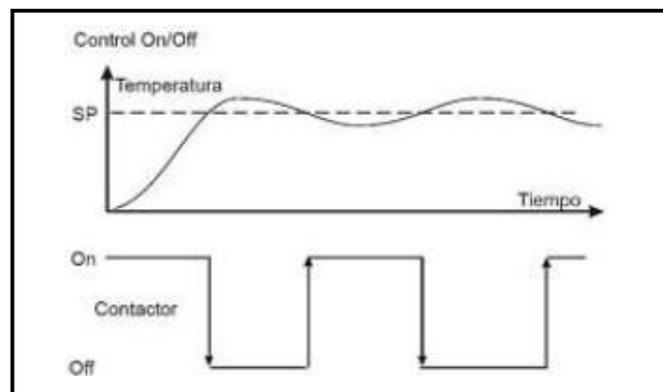


Figura N°3: Gráfica de un Controlador ON/OFF

Fuente: Arteaga y Tobar, 2014, p.26

2.3.2. Sensor de Temperatura DS18B20

Según Martínez (2018), un sensor de temperatura es un dispositivo capaz de transformar los cambios de temperatura de entorno, que serán medidos de diferentes formas según el tipo de funcionamiento y transformarlo a energía eléctrica. Posteriormente esta señal será enviada y procesada a otro dispositivo o computadora para poder visualizar o ser almacenada.

El sensor de temperatura DS18B20 cuenta con 2 modelos: sumergible y en seco, tal como se observa en la Figura N°4.

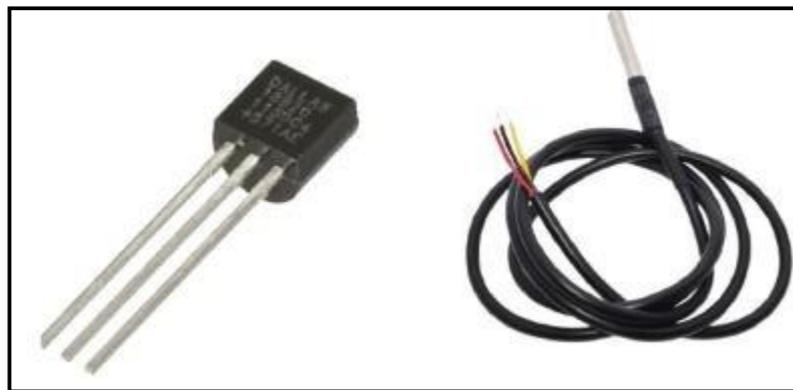


Figura N°4: Sensor DS18B20: En Seco y Sumergible

Fuente: Martínez,2017, P.18

Particularmente, para este trabajo de tesis, se optó por el sensor de temperatura sumergible por la característica del proyecto. Asimismo, este sensor posee un rango de medida comprendido entre -55 y 125 °C como temperaturas más extremas. Sin embargo, no en todo este rango tiene el mismo comportamiento, ya que la tolerancia al error es diferente en ciertos rangos, por ejemplo, entre -10 y 85 °C la tolerancia es de ± 5 grados. Para el resto de las temperaturas entre -55 °C y 125 °C el error es de ± 2 grados. Además, se debe tener en cuenta que el sensor tiene errores debido a factores externos, al ruido inherente en los circuitos eléctricos y alteraciones en medio físico. A continuación, en la Figura N°5 se muestra una representación gráfica del rango de tolerancia del sensor utilizado en este trabajo de tesis.

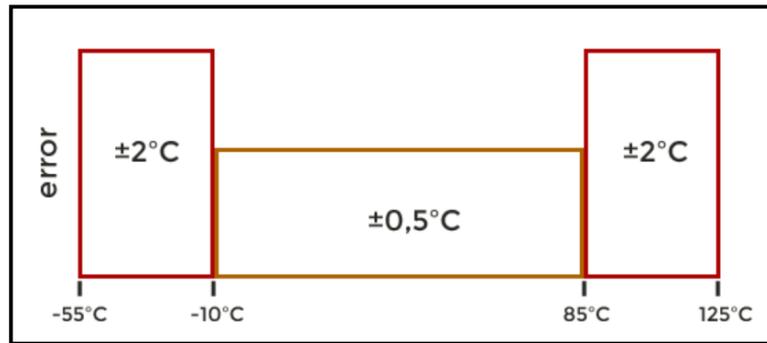


Figura N°5: Gráfico de rango de Tolerancia del Sensor DS18B20

Fuente: <https://programarfacil.com/blog/arduino-blog/ds18b20-sensor-temperatura-arduino/>

Otra característica importante de este sensor es la resolución, la cual nos permite trabajar en un rango variable de resoluciones, ajustando la precisión (variación mínima medible entre dos temperaturas). Además, el rango varía entre 9 y 12 bits, tal como se muestra en la Tabla N°1. Y como también, esta resolución se podrá colocar en la programación.

Tabla N°1: Tabla Resoluciones DS18B20

Resolución	Temperatura (°C)
9-bit	0,5
10-bit	0,25
11-bit	0,125
12-bit	0,0625

Fuente: Martínez,2017, P.19

Por otra parte, otra característica específica de este sensor es el uso del protocolo de comunicación 1-wire. La característica principal del protocolo 1-wire con este sensor es la de permitir trabajar con múltiples sensores a través de un solo cable conectado a un pin de entrada de Arduino, necesitando, además de una sola resistencia conectada a los múltiples sensores, con lo cual evitará contar con muchos circuitos de interfaz y de mucho cableado.

Complementariamente, está compuesto por un encapsulado impermeable y por el cableado que nos permite conectar al microcontrolador. Este cableado está formado por tres pines de salida (ver la Figura N°6):

V_{dd}: Cable Rojo. Por este cable se alimenta al sensor desde 3 VDC a 5.5 VDC.

GND: Cable Negro. Por este se conectará el sensor a tierra.

DQ: Cable Amarillo. Es el pin que transfiere los datos al programa a través del protocolo OneWire.



Figura N°6: Pines de conexión del sensor DS18B20

Fuente: Martínez,2017, P.19

Finalmente, para la conexión del pin de datos del sensor (DQ) se deberá tener en cuenta la conexión de una resistencia pull-up, esto es para controlar el bus de comunicación. La resistencia que se vaya a utilizar dependerá de la longitud del cable. Para el caso de este proyecto de tesis, la resistencia de Pull-up indicada es la de 4.7k Ω . Ver la Tabla N°2.

Tabla N°2: Tabla Resistencias Pull-Up para el sensor DS18B20

RESISTENCIA PULL-UP	DISTANCIA DEL CABLE (METROS)
4,7 k Ω	De 0 m a 5 m
3,3 k Ω	De 5 m a 10 m
2,2 k Ω	De 10 m a 20 m
1,2 k Ω	De 20 m a 50 m

Fuente: <https://programarfacil.com/blog/arduino-blog/ds18b20-sensor-temperatura-arduino/>

2.3.3. Sensor de Nivel HC-SR04

Según Hurtado (2020), un sensor de nivel es un dispositivo electrónico que permite medir la altura de un líquido que se encuentra dentro de un contenedor, tanque o pozo. Respecto a los sensores para líquidos, estos funcionan por medio de lecturas directas como una sonda, instrumento con flotador, nivel decristal o en base a medidas indirectas o inductivas como los de membranas y

de presión; también se pueden utilizar características conductivas de los líquidos para realizar mediciones, por ello se tienen los sensores tipos capacitivos, conductivos, resistivos, de radiación entre otros. Para el caso de esta Tesis, se utilizó el sensor de ultrasonido HC-SR04 (ver la Figura N°7) por ser un sensor de bajo costo que permite determinar la distancia de un objeto en un rango de 2 a 450cm. Además, es un dispositivo de bajo consumo energético y de pequeño tamaño. Este sensor cuenta con dos transductores: un emisor y un receptor piezoeléctricos. De esta manera, el principio de funcionamiento del sensor es generar un tren de pulsos mediante el pin TRIG, la onda de sonidos viajará por el aire y rebotará al encontrar el objeto, el sonido de retorno será escuchado por el receptor y el pin ECHO cambiará a Alto (5V) por un tiempo igual al que demoró la onda desde que fue emitida hasta que fue detectada. Este tiempo será medido por el microcontrolador y de esta manera se podría calcular la distancia del objeto mediante la ecuación 3.

$$d = \frac{1}{2} V \cdot t$$

Ecuación N°3: Distancia en función de la velocidad y el tiempo

Donde:

V: Velocidad del sonido en el aire (343 m/s).

t: Tiempo transcurrido entre la emisión y recepción del pulso.

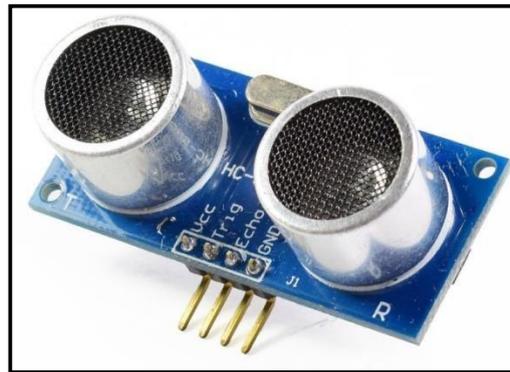


Figura N°7: Sensor de Ultrasonido HC-SR04

Fuente: <https://www.naylampmechatronics.com/sensores-proximidad/10-sensor-ultrasonido-hc-sr04.html>

A continuación, se indican las principales características técnicas del Sensor de ultrasonido HC-SR04.

- Voltaje de operación: 5V DC

- Corriente de reposo: < 2mA
- Corriente de trabajo: 15mA
- Rango de medición: 2cm a 450cm
- Precisión: +- 3mm
- Ángulo de apertura: 15°
- Frecuencia de ultrasonido: 40KHz
- Duración mínima del pulso de disparo TRIG (nivel TTL): 10 μ S
- Duración del pulso ECO de salida (nivel TTL): 100-25000 μ S
- Dimensiones: 45mm x 20mm x 15mm
- Tiempo mínimo de espera entre una medida y el inicio de otra 20ms (recomendable 50ms)

2.3.4. Arduino Uno

Según Huayta y Sueña (2019), Arduino es una plataforma de computación de código abierto basada en una sencilla placa electrónica con entradas y salidas (E/S), analógicas y digitales, y en un entorno de desarrollo que implementa el lenguaje Processing/Wiring. La tarjeta cuenta con un microprocesador Atmel, el cual es un chip sencillo y de bajo costo que permite el desarrollo de diversos proyectos.

Dentro del catálogo general de los modelos de las tarjetas Arduino, se utilizó para este trabajo de Tesis el módulo Arduino Uno como interfaz entre la computadora y el sistema, debido a que este módulo es de bajo costo y permite tener una participación como una tarjeta de adquisición de datos que facilita la lectura de sensores y enviarlos a una interfaz gráfica como el LabVIEW. Ver la Figura N°8.



Figura N°8: Tarjeta Arduino Uno

Fuente: Huayta y Sueña,2018, P.44

Según Sierra (2017), la Tarjeta Arduino Uno está basado en el microcontrolador Atmega328P. Tiene 14 pines digitales de Entrada/Salida, de los cuales 6 pueden ser usados como salidas PWM. Además, cuenta con 6 entradas analógicas (solo de entrada), un cristal de cuarzo de 16 MHz, una conexión USB, un conector de poder, un encabezado ICSP y un botón de reset o reinicio.

A continuación, la Tabla N°3 muestra las especificaciones del hardware del Arduino Uno:

Tabla N°3: Especificaciones Técnicas del Arduino Uno

Microcontrolador	ATmega328P
Voltaje de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límite)	6-20V
Pins de E/S digitales	14 (de los cuales 6 proporcionan salida PWM)
PWM Digital I/O Pins	6
Clavijas de entrada analógica	6
Corriente DC por Pin de E/S	20 mA
Corriente DC para 3.3V Pin	50 mA
Memoria flash	32 KB (ATmega328P) de los cuales 0.5 KB utilizados por el cargador de arranque.
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Velocidad de reloj	16 MHz
LED incorporado	13
Longitud	68.6 mm
Anchura	53.4 mm
Peso	25 g

Fuente: <https://store.arduino.cc/usa/arduino-uno-rev3>

2.3.5. LabVIEW

Vergara (2012), menciona que LabVIEW es un lenguaje de programación gráfico para diseñar sistemas de control, instrumentación o de adquisición de datos. LabVIEW permite diseñar interfaces de usuario mediante una consola interactiva basada en software.

Este software es compatible con herramientas de desarrollo similares y puede trabajar con programas de otras áreas de aplicación. Una de sus principales ventajas es la permitir integración con hardware, específicamente con tarjetas de medición, adquisición y procesamiento de datos. Además, LabVIEW permite crear interfaces, diagramas o paneles frontales simulando leds, diales, botones, etc.

Para el caso de la presente tesis se optó por el software LabVIEW por ser de lenguaje de programación gráfica, y así mismo por contar con la facilidad de integrar como hardware de adquisición de datos al módulo Arduino, cuya función es la de adquirir los datos de los sensores y enviarlas a la programación hecha en LabVIEW.

El software cuenta con las siguientes herramientas importantes:

- a) Diagrama de bloques: En esta sección del software se realiza la programación de bloques. A continuación, la Figura N°9 muestra un ejemplo de programación en esta sección.

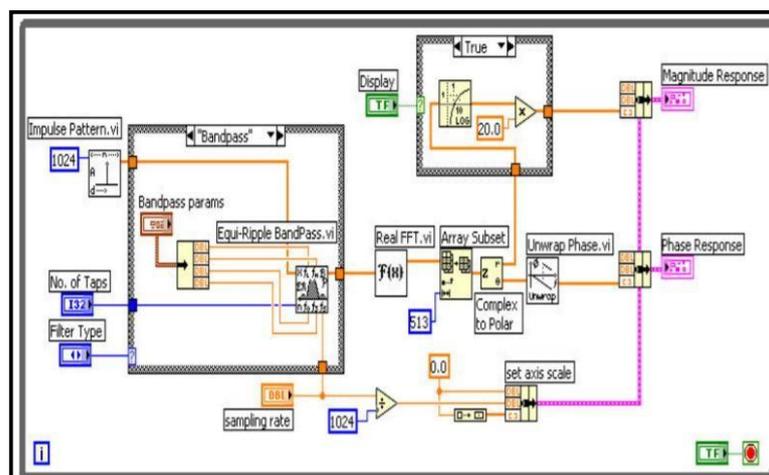


Figura N°9: Diagrama de Bloques

Fuente: Vergara B, 2012, P.6

b) Panel Frontal: En esta sección, es donde los datos se pueden apreciar, se manipulan y se controlan. A continuación, la Figura N°10 muestra un ejemplo de interface gráfica desarrollada en esta sección.

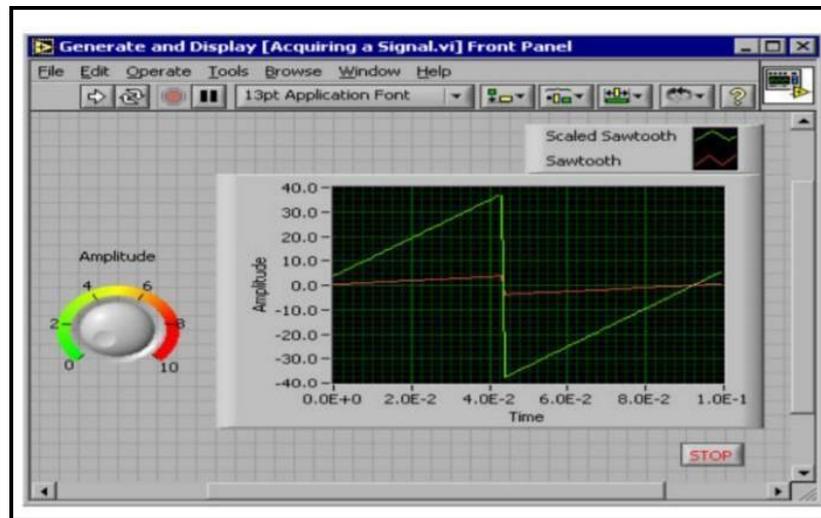


Figura N°10: Panel Frontal

Fuente: Vergara B, 2012, P.7

2.3.6. Electroválvula

Una electroválvula es un componente electromecánico controlado eléctrica o electrónicamente que permite, o no el paso de un fluido. Este dispositivo está compuesto por una válvula mecánica y una bobina solenoide. La válvula dejará pasar el fluido cuando se le aplica una corriente eléctrica a la bobina solenoide.

Para el caso de esta tesis se usó una electroválvula alimentada por 12 VDC, la cual dejó ingresar agua al prototipo de bañera en el momento que el microcontrolador la activó. Seguidamente, la Figura N°11 muestra una fotografía de la electroválvula empleada en este proyecto de tesis.



Figura 11: Electroválvula
Fuente: Elaboración Propia

2.3.7. Electrobomba

Una electrobomba o bomba de agua es una máquina eléctrica que transforma energía eléctrica a energía mecánica para poder transportar agua de un punto a otro, aplicándoles un flujo y una presión. En este trabajo de tesis se utilizó una bomba de agua periférica de 0.5 HP de potencia, para poder abastecer de agua a la bañera. Seguidamente, la Figura N°12 muestra una fotografía de la electrobomba empleada e inmediatamente después de indican sus principales características.



Figura N°12: Electrobomba de 0.5Hp
Fuente: Elaboración Propia

- Altura máxima: 35 m
- Caudal máximo: 35 l/m
- Voltaje de alimentación: 220 Vac
- Frecuencia: 60Hz
- Revolución: 3450 rpm

2.3.8. Módulo Relé

Matías y Ruiz (2014), explican que un relé es un dispositivo electromecánico que funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes.

Igualmente, la gran ventaja de los relés es la separación eléctrica entre la corriente de acoplamiento, la corriente del electroimán, y de los circuitos controlados por contacto, por ello es posible poder manejar altos voltajes con pequeñas tensiones de control.

Para la presente Tesis se utilizaron módulos relé controlados por un microcontrolador Arduino para controlar las electroválvulas del prototipo de bañera. Estos módulos cuentan con una etapa de asilamiento en las entradas de control para reducir el ruido percibido entre el circuito de control y la carga. Además, tiene la ventaja de no contar con un circuito adicional, por ello solo se conecta directamente al elemento de control y a la carga. A continuación, la Figura N°13 muestra una fotografía del módulo relé empleado en este proyecto de tesis.

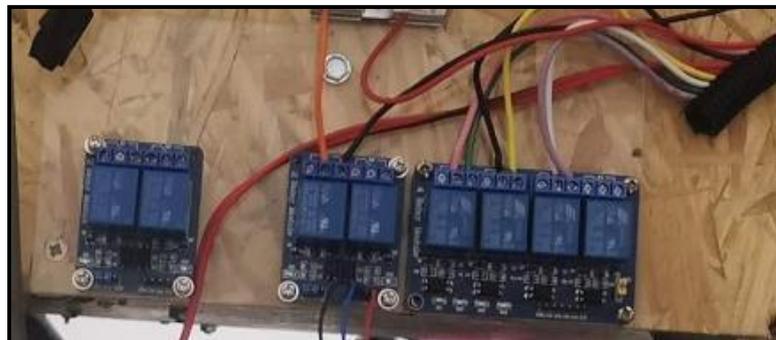


Figura N°13: Módulo Relé

Fuente: Elaboración Propia

2.3.9. Resistencia Calefactora de Inmersión

Según Quiñones (2020), una resistencia de inmersión es un dispositivo eléctrico cuya función principal es la de elevar la temperatura de algún fluido que se encuentre en contacto directo con este elemento, es decir la resistencia transfiere calor al líquido a calentar. El fluido puede ser: agua, aceite, material viscoso, disoluciones ácidas o básicas, etc.

Además, están elaboradas por lo general de alguna clase de filamento o cable, la cual al recibir un flujo de corriente este es transformado en calor, el cual será aprovechado para poder calentar el fluido requerido. Dentro de los principales usos de este dispositivo son las termas eléctricas que comúnmente encontramos en los hogares para las duchas.



Figura N°14: Resistencia de Inmersión

Fuente: Elaboración Propia

La Figura N° 14 muestra la fotografía de la resistencia calefactora de inmersión que se utilizó en este proyecto de tesis, cuyas características principales se indican a continuación:

- Voltaje de alimentación: 220 VAC
- Frecuencia: 60Hz
- Potencia: 3 KW
- Amperaje de consumo: 13.64 A
- Diámetro del tubo: 8mm
- Longitud: 203mm

2.4. Definición de términos básicos

- a) Bañera automatizada para canes: Es una máquina autónoma diseñada para el lavado y secado de animales pequeños como perros. Asimismo, está diseñado para el cuidado del animal y su confort, ya que cuenta con un sistema de hidromasaje

que pulveriza el agua a temperatura y presión idóneas para ofrecer una mayor calidad en el proceso de lavado al perro.

- b) Sensor: Elemento o dispositivo que convierte una señal física (Temperatura, presión, longitud, entre otras) a una señal eléctrica para poder ser procesado por un controlador.
- c) Reservorio de agua: Es una estructura de forma rectangular cuya función principal es la de almacenar agua.
- d) Agitador: Es un instrumento cuya función principal es la de mezclar o revolver una sustancia.
- e) Termostato mecánico: Es un instrumento cuya función principal es de activar o desactivar un circuito en función al cambio de temperatura.
- f) Válvula de bola: Es un mecanismo de llave de paso cuya función es regular el flujo de un líquido o sustancia.
- g) Tubo de PVC: Es un elemento elaborado de un material llamado cloruro de polivinilo. La forma de este elemento es cilíndrica. Es usado en su mayor parte para realizar trabajos de plomería por su alta resistencia al agua.
- h) Plancha OSB: Oriented Strand Board, es un tablero echo a partir de virutas de madera orientada las cuales son sometidas a presiones y temperaturas muy alta y de esa forma se compacta formando los tableros. Es mayormente usado en la construcción de viviendas.
- i) Acrílico: Es un tipo de plástico que es ampliamente utilizado en construcciones por sus buenas propiedades de poder permanecer a la intemperie sin sufrir algún daño, además de la flexibilidad respecto a otro tipo de plástico por lo que lo hace fácil de trabajar.

2.5. Diseño de la Investigación

2.5.1. Variables de investigación

En este trabajo de investigación, se consideraron las siguientes variables de estudio:

- **Variable Independiente**

Control PID

Control ON/OFF

Control Secuencial

• **Variable Dependiente**

Prototipo de Bañera Casera Automatizada

2.5.2. Tipo y Método de investigación

El tipo de investigación es aplicada y tecnológica. En cuanto al método de investigación, es empírico y experimental porque se midieron las variables temperatura y nivel para lograr el control secuencial del prototipo de bañera.

2.5.3. Técnicas e Instrumentos de recolección de datos

Los instrumentos utilizados para la recolección de datos fueron:

a) Sensor de ultrasonido HC-SR04: se encargó de enviar un tren de pulsos digitales y recibir una señal digital, que posteriormente fue adquirida por el módulo Arduino UNO y enviada a la Laptop para ser procesada por el Controlador implementado algorítmicamente en el software LabVIEW. De esta manera, la interconexión entre el software LabVIEW, el módulo Arduino y el sensor de ultrasonido permitió la recolección de datos experimentales del tipo distancia.

b) Sensor de temperatura DS18B20: se encargó de medir la temperatura en un formato digital, utilizando el protocolo 1 wire para comunicarse con el microcontrolador del módulo Arduino UNO. De esta manera, la interconexión entre el software LabVIEW, el módulo Arduino y el sensor de temperatura permitió la recolección de datos experimentales en la unidad de grados centígrados.

c) Arduino uno: Se encargó de ser la tarjeta interfaz entre el software LabVIEW y los sensores y actuadores del prototipo. El módulo Arduino recolectó la información enviada por los sensores de temperatura y ultrasonido para luego ser enviada al LabVIEW en donde se realizaría todo el procesamiento, así como también este módulo se encargó de activar los dispositivos actuadores cuando el LabVIEW se lo ordene, según la programación realizada.

Respecto a la técnica utilizada en este trabajo de tesis, fue la observación experimental y el análisis empírico lo cual permitió sintonizar e implementar los controladores PID, ON/OFF y Secuencial.

2.5.4. Procedimiento para la recolección de datos

Para el procedimiento de la recolección de datos para las variables nivel y temperatura se siguió los siguientes pasos:

- a) Se realizó la programación para el sensor de ultrasonido en la tarjeta Arduino. Donde se validó el correcto funcionamiento de este sensor.
- b) Se realizó la programación para la recepción de los datos del sensor de ultrasonido en el LabVIEW. Se corroboró que el LabVIEW esté recibiendo correctamente los datos del sensor.
- c) Una vez que se pudo hacer la comunicación entre el LabVIEW y el Arduino para el sensor de ultrasonido se realizó la implementación del controlador ON/OFF en el LabVIEW y se realizaron pruebas con para este control.
- d) Se realizó la programación para el sensor de temperatura en la tarjeta Arduino. Donde se validó el correcto funcionamiento de este sensor.
- e) Se realizó la programación para la recepción de los datos del sensor de temperatura en el LabVIEW. Se corroboró que el LabVIEW esté recibiendo correctamente los datos del sensor.
- f) Una vez que se pudo hacer la comunicación entre el LabVIEW y el Arduino para el sensor de temperatura se realizó la implementación y sintonización del controlador PID de temperatura en el LabVIEW, donde finalmente se realizaron las pruebas solo con este controlador.
- g) Se juntan de los controladores ON/OFF y PID. Donde el programa inicia con el control ON/OFF de nivel y luego el control PID de temperatura.
- h) Se hizo la programación del control secuencial en el LabVIEW y en la tarjeta Arduino.
- i) Por último, se realizó la programación de todos los controlares en un solo programa, donde estarán el control ON/OFF, control PID y control Secuencial.

CAPÍTULO III: DISEÑO DEL PROTOTIPO DE BAÑERA

En este capítulo se mostrarán todas las etapas del desarrollo de la implementación del prototipo de bañera. Se iniciará describiendo la etapa de armado del prototipo y posteriormente la implementación de cada uno de los controladores de nivel, temperatura y secuencial. En esta última parte se describirán los pasos tomados para el desarrollo de la implementación de cada etapa de control.

3.1. Dimensionamiento del Prototipo de Bañera para Canes

El prototipo de bañera propuesto está diseñado para bañar a canes controlando la cantidad de agua, la temperatura del agua para el confort del perro y el tiempo de bañado; de esta manera, se logra reducir el consumo de agua y de energía. Para ello, se determinó utilizar los siguientes controlares: ON/OFF para controlar el nivel de agua en el tanque, PID para controlar la variable temperatura, y secuencial para controlar el ciclo de lavado del can. Estos controladores fueron implementados en el software LabVIEW donde se desarrolló una interfaz gráfica para controlar y monitorear de manera sencilla el prototipo de bañera propuesto, igualmente se utilizó el módulo Arduino como interfaz entre sensores, actuadores y los controladores.

Además, el prototipo de bañera para canes fue construido en base a una estructura de madera, acrílico y tubos de PVC.

A continuación, en la Figura N°15 se muestra la representación pictográfica del prototipo de bañera de canes propuesto en este trabajo de tesis. Asimismo, en dicha figura se aprecian los principales elementos utilizados para el control del nivel de agua, control de la variable temperatura y el control secuencial de bañado.

Por otro lado, en la Figura N°16, se observa el diagrama P&ID del sistema de control para el prototipo de bañera para canes, según la Norma ISA-5.1.

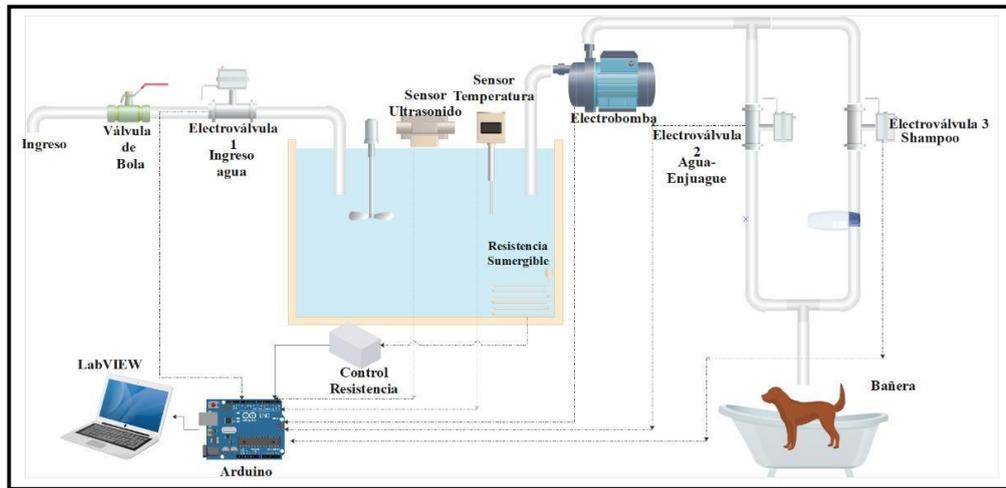


Figura N°15: Representación pictográfica del Prototipo de Bañera

Fuente: Elaboración Propia usando el software EdrawMax

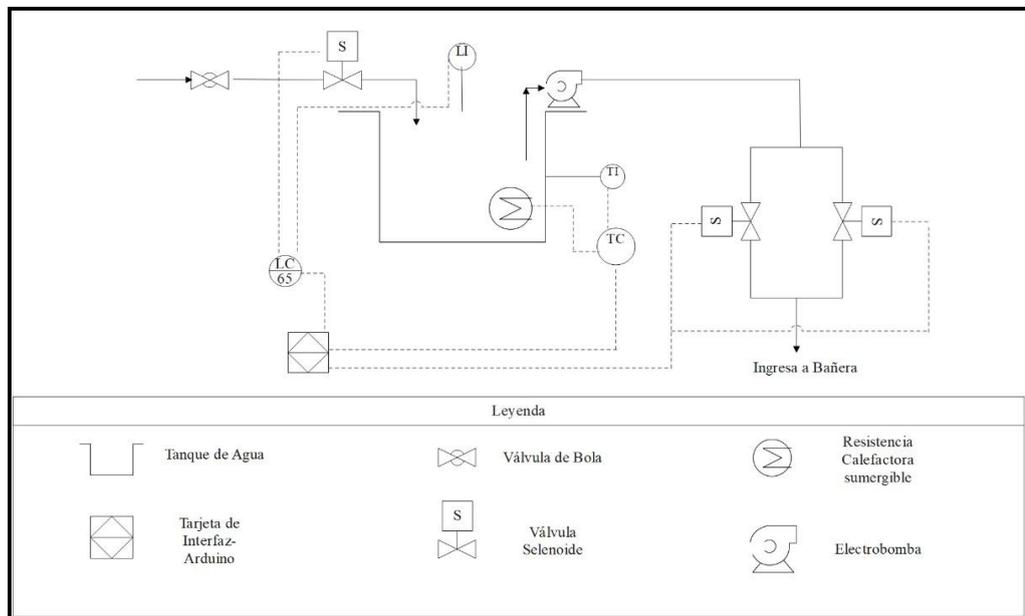


Figura N°16: Diagrama P&ID del Prototipo de Bañera

Fuente: Elaboración Propia usando el software EdrawMax

Seguidamente, en la Figura N° 17, se muestra el diagrama de flujo del funcionamiento general del prototipo de bañera para canes.

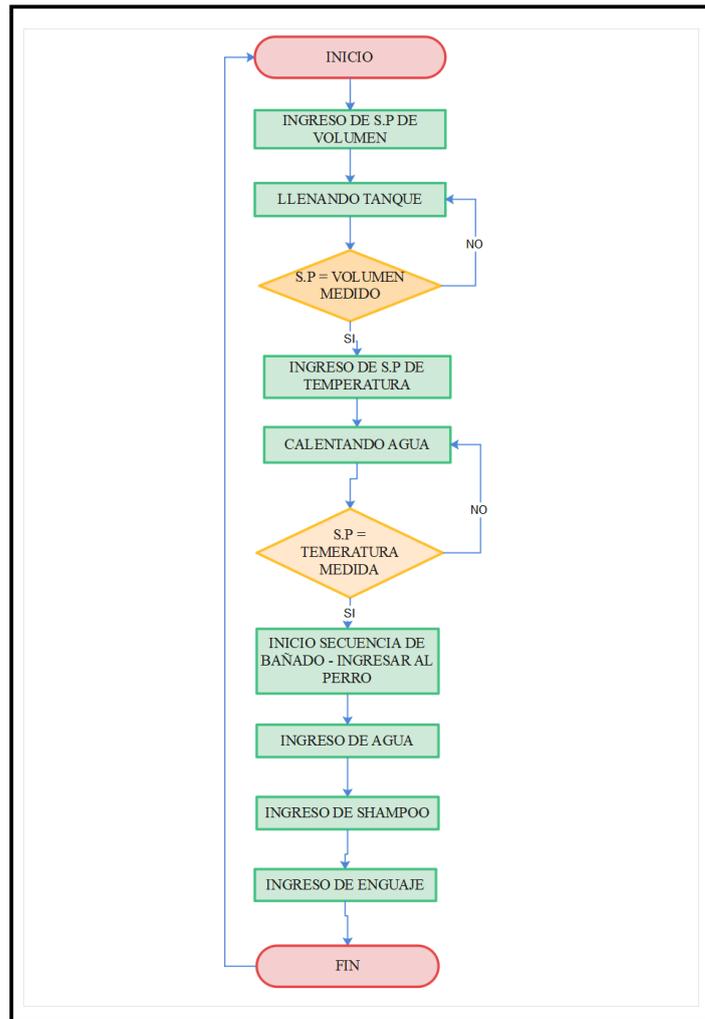


Figura N°17: Diagrama de Flujo del Prototipo de Bañera

Fuente: Elaboración Propia

Tal como se aprecia, el funcionamiento del prototipo de bañera inicia con el control ON/OFF de nivel, por tal razón la interfaz de control del LabVIEW solicita al usuario el ingreso del valor del volumen de agua requerido con el que se llenará el tanque, para el caso de este trabajo de tesis el volumen fue de 80L. La electroválvula 1 (ver la Figura N° 15) que es controlada por el controlador de nivel, es la encargada de permitir el ingreso de agua al tanque y estará activa mientras no alcance el Set Point ingresado por el usuario. Cuando el volumen de agua logre alcanzar el volumen solicitado, la electroválvula 1 se desactiva e inicia el funcionamiento del controlador PID de temperatura y al igual que en el control ON/OFF, la interfaz de control del LabVIEW también solicita al usuario ingresar la temperatura de Set Point; para este trabajo de Tesis la temperatura deseada fue de 37°C. Asimismo, para elevar la temperatura del agua hasta el valor deseado, se utilizó una resistencia calefactora sumergible.

Por último, al alcanzar la temperatura deseada para el prototipo de bañera, se activa el control secuencial de bañado para el can. Es en este momento en el que la mascota ingresa a la bañera y se da inicio al bañado. La secuencia de lavado tiene 3 etapas: la primera etapa llamada ingreso de agua, que inicia activando la electrobomba de agua y la electroválvula 2 (ver la Figura N° 15) para empapar al can por 15 segundos. Luego, la etapa 2 llamada ingreso Shampoo, encargada de desactivar la electroválvula 2 y activar la electroválvula 3 (ver la Figura N° 15) para que el Shampoo se vierta sobre el can. Finalmente, la etapa 3 llamada enjuague del can y encargada de realizar tal acción. Al completar las 3 etapas, el ciclo de bañado se completa y la interfaz se desactiva.

3.2. Diseño e Implementación de la Estructura del Prototipo de Bañera

El diseño e implementación de la estructura de la bañera se consideró dividirlo en las siguientes etapas: Parte estructural, Parte de plomería, Parte eléctrica y Parte electrónica.

3.2.1. Diseño e Implementación de la Parte Estructural

La parte estructural del prototipo de bañera se realizó dividiéndolo en cuatro partes: la base, la cabina de la bañera, la estructura de la Bañera de PVC y el tanque de agua.

a) La base

La base del módulo de la bañera se construyó de madera tipo OSB, esto debido a que las propiedades de este tipo de material permiten con mayor facilidad la construcción del módulo, ya que no requiere de mayor trabajo para armarlo con herramientas comunes que se tiene en el hogar, tal como un destornillador, martillo, tornillos, taladro, etc. A continuación, en la Figura N° 18, se muestra una representación tridimensional de la base del prototipo de bañera implementada en el software Autodesk Inventor, que a su vez ayudó a la implementación física de la misma.

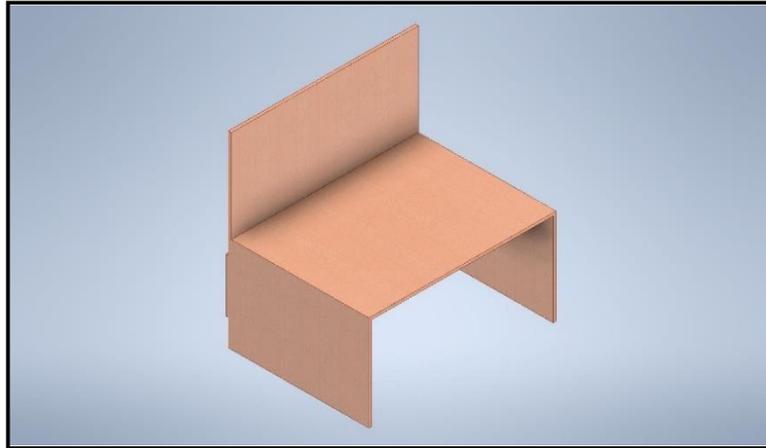


Figura N°18: Diseño en 3D de la base del prototipo de bañera

Fuente: Elaboración Propia usando el software Autodesk Inventor

Asimismo, las dimensiones de la base fueron:

- a) Patas: Se utilizó 2 planchas de Largo= 90cm y Alto=60cm.
- b) Mesa: Se utilizó 1 plancha de Largo= 120cm y Ancho=90cm.
- c) Soporte Posterior: Se utilizó 1 plancha de Largo=120cm y Alto=37cm.
- d) Soporte Superior: Se utilizó 1 plancha de Largo=120cm y Alto=60cm.

El espesor de la plancha de OSB utilizado para la base fue de 11mm. Seguidamente, la Figura N° 19, muestra una fotografía de la base del prototipo de bañera en mención.



Figura N°19: Implementación de la base del prototipo de bañera

Figura: Elaboración Propia

b) Cabina de la Bañera

Representa a la cabina donde ingresa el can para ser bañado, y es donde se sentirá cómodo y protegido evitando que el agua usada se rebalse. Para la construcción de esta cabina se utilizaron planchas de acrílico transparente formando una especie de caja. Se le colocó 2 puertas para poder ingresar al can y luego cerrarlo, para que el agua no salga. A continuación, en la Figura N° 20, se muestra una representación tridimensional de la cabina de bañera implementada en el software Autodesk Inventor, que a su vez ayudó a la implementación física de la misma.

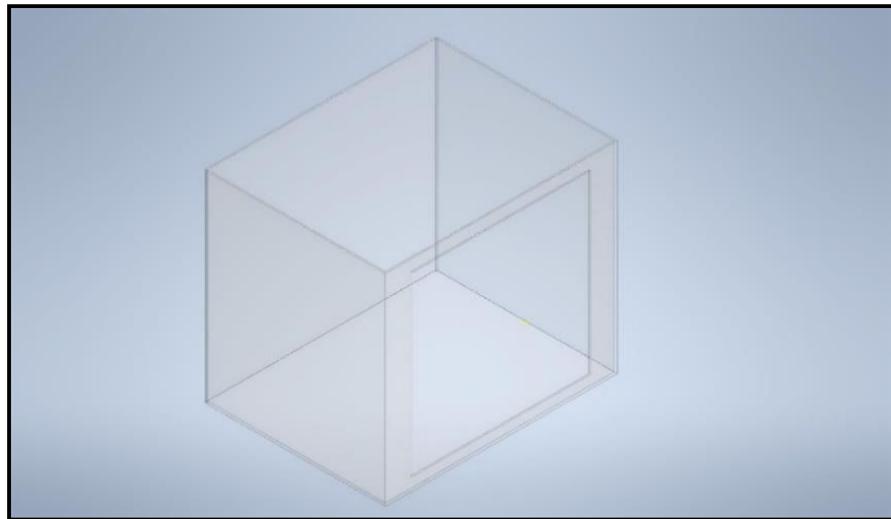


Figura N°20: Diseño 3D de la cabina del prototipo de bañera

Fuente: Elaboración Propia usando el software Autodesk Inventor

La dimensión de la cabina fue de: Largo 90cm, Ancho 70cm y Alto 80cm. Seguidamente, la Figura N° 21, muestra una fotografía de la implementación de la cabina del prototipo de bañera en mención.



Figura N°21: Implementación de la cabina del prototipo de bañera

Fuente: Elaboración Propia

c) Estructura de la Bañera de PVC

La estructura de bañera se construyó utilizando tubos de PVC, tri codos para las esquinas y un conector Tee. Se realizó de esta forma, debido a que se deseaba que el agua alcance al can en más de un punto, por tal razón se realizaron agujeros de 2 mm en los 3 tubos superiores con una distancia entre cada uno de 1 cm. A continuación, en la Figura N° 22, se muestra una representación tridimensional de la estructura de bañera implementada en el software Autodesk Inventor, que a su vez ayudó a la implementación física de la misma.

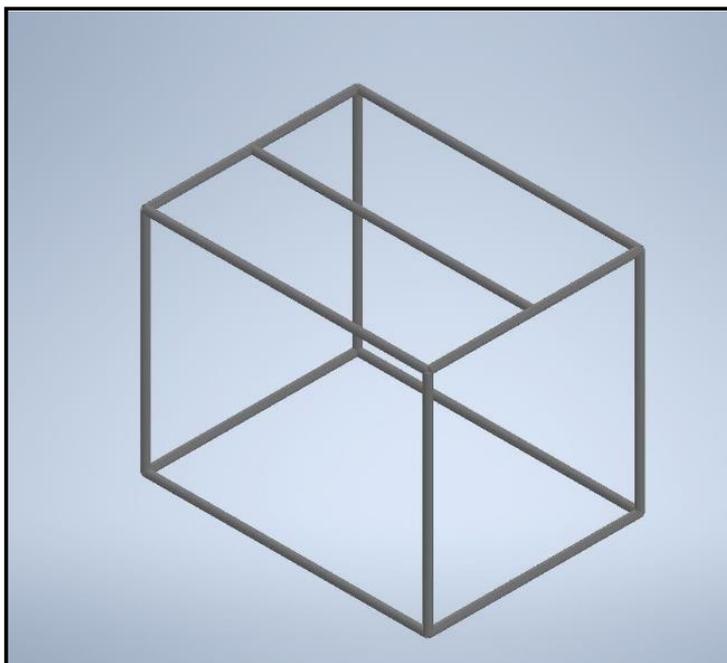


Figura N°22: Diseño 3D de la bañera

Fuente: Elaboración Propia usando el software Autodesk Inventor

Las dimensiones de estructura de la bañera son: Largo 80cm, Ancho 60cm y Alto 65cm. Además, se utilizaron tubos de PVC de 1/2". Seguidamente, la Figura N° 23 muestra una fotografía de la implementación de la estructura de PVC del prototipo de bañera en mención.

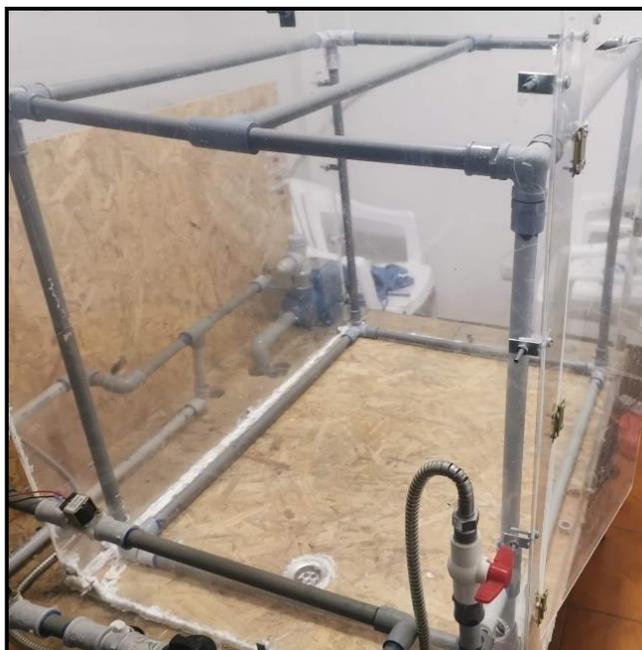


Figura N°23: Implementación de la estructura de PVC del prototipo

Fuente: Elaboración Propia

d) Tanque de agua

Se eligió un tanque de agua de dimensión (Largo: 77.5 cm, Ancho: 47.4 cm, Alto: 45.7 cm), y se ubicó en la parte inferior de la estructura de la base de la bañera. Además, se añadió un depósito de dimensión (Largo: 59.7 cm, Ancho: 46,4 cm, Alto: 34,7 cm) donde se depositó el drenaje de la bañera. Seguidamente, la Figura N° 24 muestra una fotografía del tanque y el depósito de agua utilizados.



Figura N°24: Tanque de agua

Fuente: Elaboración Propia

3.2.2. Implementación de la parte de Plomería

La implementación de las tuberías para el prototipo de bañera se realizó con tubos de PVC en su mayor parte de 1/2". En la Figura N° 25 se observa la tubería de ingreso al tanque. Además, esta implementación cuenta con una llave de paso para el ingreso del agua al prototipo. Así como también, se cuenta con la Electroválvula 1, la cual es controlada por el controlador ON/OFF.



Figura N°25: Ingreso de agua al Prototipo de Bañera

Fuente: Elaboración Propia

Luego, para el ingreso del agua hacia la bañera se cuenta con una electrobomba de agua, la cual es responsable de impulsar el agua desde el tanque hasta la bañera. Asimismo, se utilizaron conexiones de 1" para el ingreso de la bomba y luego en la salida de la bomba se utilizaron tuberías de 1/2". En la Figura N° 26 se observa esta distribución.



Figura N°26: Entrada y Salida de la Electrobomba de agua

Fuente: Elaboración Propia

Finalmente, se implementó el circuito para el ingreso de agua y Shampoo al prototipo la bañera. Por un lado, el ingreso de agua es directo mediante una electroválvula, la cual se activa cuando el sistema de control ordene abrirse permitiendo el ingreso del agua. Además, para el ingreso de Shampoo la bañera cuenta un pequeño contenedor con este producto de limpieza, el cual se mezcla con el agua que es impulsada por la propia bomba también de agua. A continuación, la Figura N° 27 muestra la implementación de las electroválvulas de ingreso de agua y Shampoo.

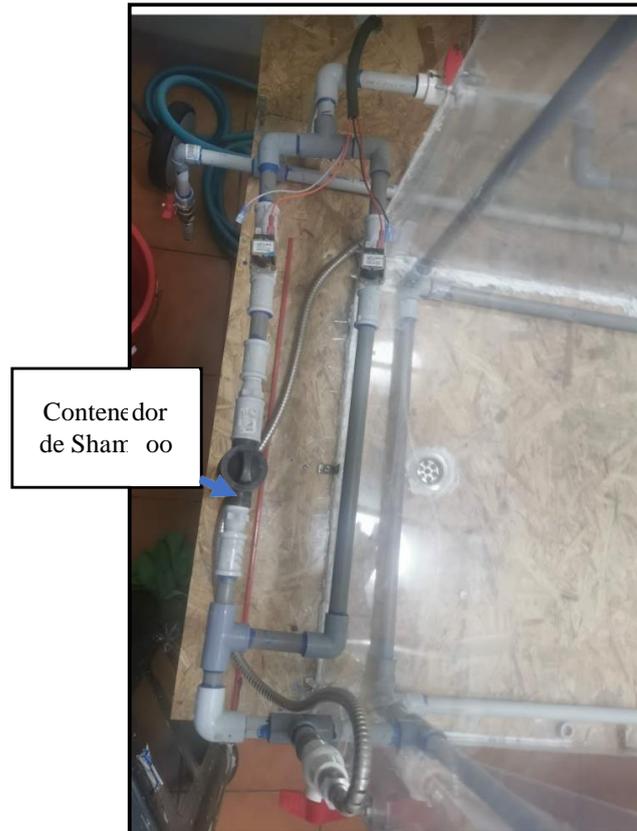


Figura N°27: Circuito de distribución ingreso de agua y shampoo

Fuente: Elaboración Propia

3.2.3. Implementación de la parte Eléctrica

El prototipo de bañera cuenta con elementos que trabajan con diferentes tipos de voltajes, es decir se tienen elementos que trabajan en corriente AC y otras en DC.

a) Para las corrientes AC. La Tabla N°4 muestra la distribución eléctrica.

Características de potencia:

Alimentación general: 220 VAC monofásico, 60Hz.

Tabla N°4: Distribución eléctrica para actuadores

Aparato Eléctrico	Cant.	Potencia (Kw)	Amperaje (A)
Electrobomba (0.5 HP)	1	0.370	1.68
Resistencia de inmersión	1	3.000	13.64
Válvula solenoide ½” 12VDC 8W	3	0.024	0.11
Corriente total (A)			15.43

Fuente: Elaboración propia

En cuanto a la selección de interruptores termomagnéticos de protección monofásicos, cada aparato eléctrico formó un circuito: factor de seguridad con 25% del valor del amperaje. Ver la Tabla N° 5.

Tabla N°5: Distribución de interruptores termomagnéticos

Circuitos	Amperaje (A)	Amperaje (A) + 25%	Interruptor de protección
Circuito-1	1.68	1.89	6A
Circuito-2	13.64	17.05	20A
Circuito-3	0.11	0.14	6A
Protección			
Interruptor general (IGA)	15.43	19.08	25A
Interruptor diferencial (ID)			25A

Fuente: Elaboración Propia

La figura N°28 muestra el esquema eléctrico de los circuitos eléctricos.

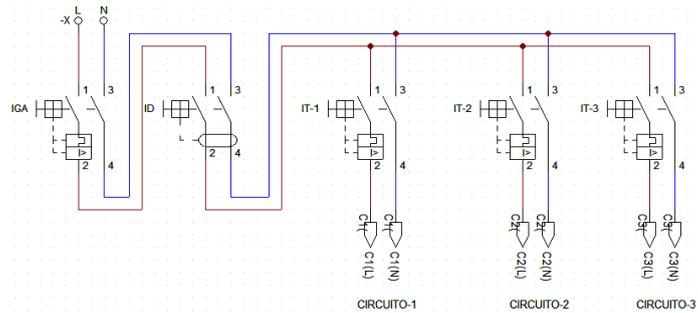


Figura N°28: Esquema Eléctrico de los circuitos eléctricos

Fuente: Elaboración Propia usando el software CadeSIMU V4

Para las corrientes DC, se utilizó una fuente de 12v la cual alimentó a las 3 electroválvulas del prototipo, los sensores de nivel y temperatura, así como a la tarjeta dimmer para el control de la resistencia calefactora sumergible. A continuación, la Figura N° 29 muestra la de fuente de 12v utilizada en el proyecto.



Figura N°29: Fuente DC de 12V

Fuente: Elaboración Propia

3.2.4. Implementación de la parte Electrónica

La implementación de los elementos de electrónica para el prototipo de la bañera contempló la instalación de sensores, actuadores, tarjetas de interfaz y tarjeta de microcontrolador.

a) Sensores:

1. Sensor de nivel: Se consideró el sensor de nivel ultrasónico (HC-S04), el cual fue colocado en la parte superior del tanque, específicamente en la tapa de este, con la finalidad que las ondas de ultrasonido tengan una mejor propagación. Se tuvo que colocar una capa de silicona para evitar que la humedad ingrese al circuito y dañe el sensor. A continuación, la Figura N°30 se muestra el sensor de ultrasonido implementado en el proyecto.



Figura 30: Sensor de nivel ultrasónico

Fuente: Elaboración Propia

2.Sensor de temperatura de agua: Se usó un sensor sumergible de temperatura (DS18B20) y se colocó al interior del tanque para que tenga un mejor contacto con el agua. Seguidamente, la Figura N°31 muestra el sensor de temperatura y el sensor de nivel.

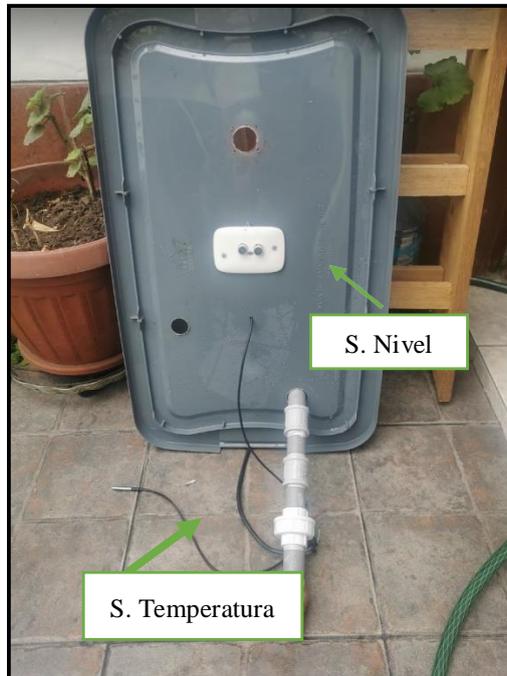


Figura N°31: Sensor de nivel y temperatura

Fuente: Elaboración Propia

3.Sensor de ambiente: Se utilizó un sensor de temperatura ambiental (DTH22), para tener información del ambiente donde se encuentra instalado el prototipo. Se debe tener en cuenta que este sensor solo es de monitorización. A continuación, la Figura N°32 muestra la implementación del sensor de temperatura ambiental.

b) Actuadores:

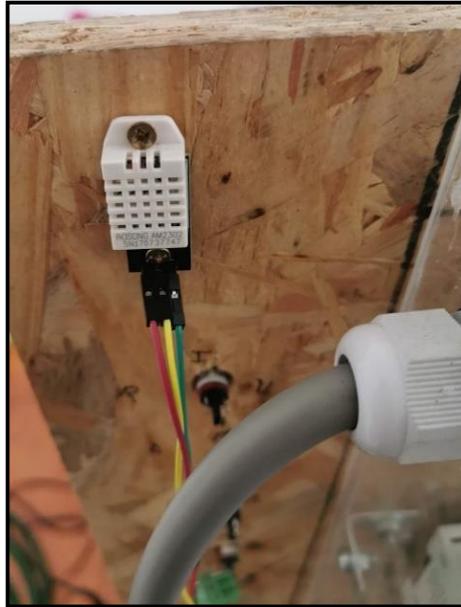


Figura N°32: Sensor temperatura ambiental

Fuente: Elaboración Propia

Los actuadores utilizados para el prototipo de bañera fueron:

1. Electroválvula: Se usaron 3 electroválvulas para el trabajo de tesis. Como se puede observar en la Figura N° 33, la distribución fue la siguiente: Electroválvula 1: Ingreso a Tanque y fue usada para el control ON/OFF. Electroválvula 2 y Electroválvula 3 fueron usadas para el control secuencial, para realizar el bañado del can. Estos actuadores se instalaron en las tuberías de 1/2" utilizando conectores de PVC.

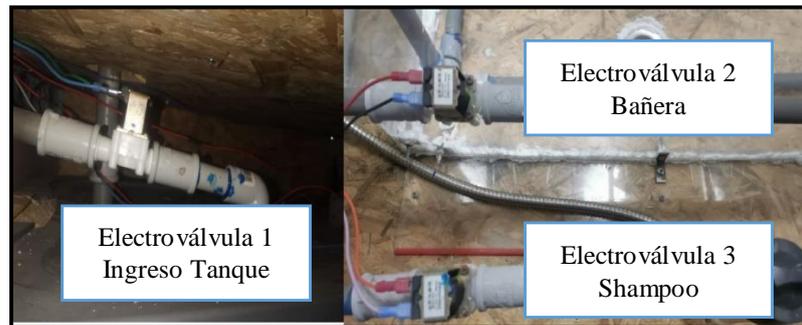


Figura N°33: Electroválvulas implementadas en el prototipo

Fuente: Elaboración Propia

Debido a que las electroválvulas tienen un voltaje de trabajo de 12Vdc se requirió utilizar un elemento de potencia que pudiera entregar ese voltaje a las electroválvulas, y a la vez ser controladas por el módulo Arduino. Por ello, se utilizó un módulo Relé para el control de las electroválvulas. Seguidamente, en la Figura N° 34 se muestra el módulo relé para las electroválvulas.

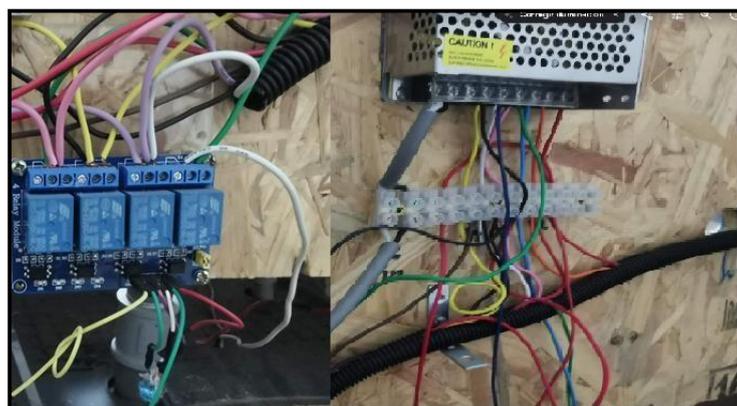


Figura N°34: Módulo Relé para las electroválvulas

Fuente: Elaboración Propia

2. Electrobomba de agua: Se instaló al nivel del piso, esto para que pudiera impulsar el agua desde el tanque hacia la bañera. A continuación, en la Figura N° 35 se muestra la bomba de agua.



Figura N°35: Electrobomba de agua

Fuente: Elaboración Propia

La electrobomba es controlada por medio del Arduino, sin embargo, al ser un elemento de potencia, es decir trabaja con voltaje de 220VAC, fue necesario contar con un elemento que ayude activarlo sin afectar al microcontrolador. Por ello, se utilizó un relé de estado sólido, y se implementó un módulo relé para activar el relé de estado sólido, debido a que se requirió controlarse por un microcontrolador. La Figura N° 36 muestra el relé de estado sólido.



Figura N°36: Relé estado sólido y módulo Relé para electrobomba

Fuente: Elaboración Propia

3. Resistencia calefactora sumergible o de inmersión: Fue instalada en la parte inferior del tanque debido a que el cuerpo de la resistencia debe estar en contacto con el agua, porque de otra forma podría dañarse. Se tuvo que colocar un adaptador llamado “Bulkhead Fitting” que cumple la función de colocar la resistencia y al mismo tiempo evitar que el agua del tanque se filtre. La Figura N° 37 muestra la resistencia sumergible instalada en el tanque de agua.



Figura N°37: Resistencia implementada en el tanque

Fuente: Elaboración Propia

Tal como fue descrito, la resistencia sumergible cumple la función de llevar la temperatura del agua hasta al valor deseado por el usuario. Además, se indicó que este dispositivo funciona con energía alterna a 220v AC; es decir, es un elemento de potencia, por ello se diseñó un circuito de dimmer para que el módulo Arduino pueda controlar la potencia de la resistencia calefactora según la orden del controlador PID.

El diseño del circuito dimmer utilizado fue el control de disparo de un triac para regular la potencia de la resistencia, el cual se activó a partir de la sincronización de la señal producida por un circuito por cero y el módulo Arduino. Se puede observar en la Figura N°38 el diagrama del circuito dimmer diseñado para el trabajo de tesis.

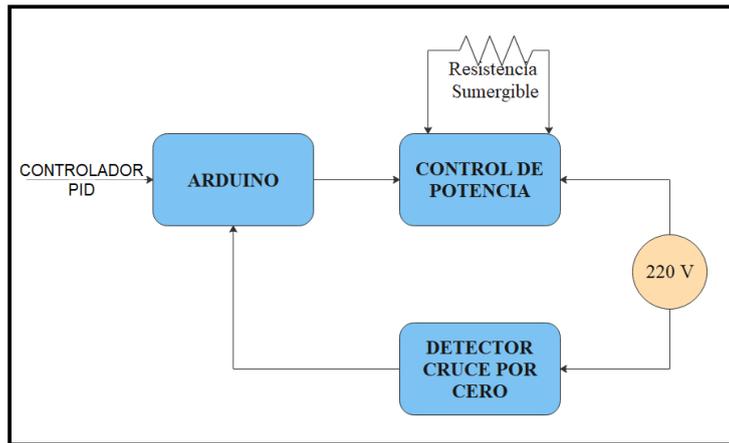


Figura N°38: Diagrama del circuito dimmer

Fuente: Elaboración Propia

En la Figura N°39 se puede observar el esquemático del circuito dimmer.

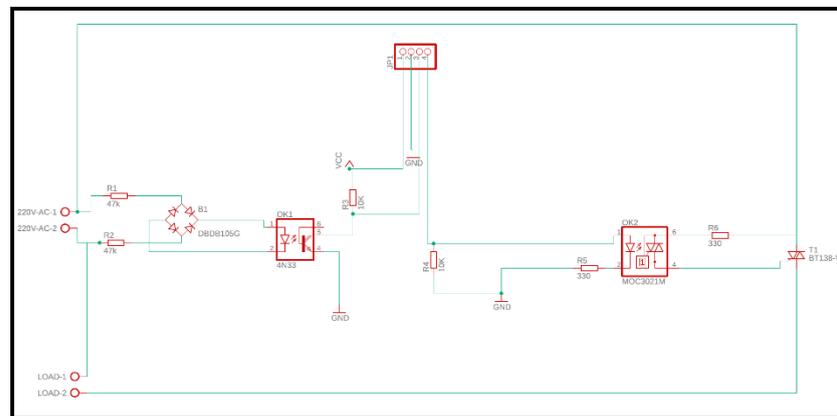


Figura N°39: Esquemático del circuito dimmer

Fuente: Elaboración Propia usando el software Eagle

El circuito dimmer, cuenta con 2 etapas: Detector cruce por cero y control de potencia.

El circuito detector cruce por cero fue diseñado para indicarle al Arduino cuando la onda de 220v AC cruza por cero para activar al Triac y por ello controla la potencia disipada por la resistencia.

El circuito utilizó un puente de diodos que realizó la rectificación de la onda de entrada, y un optoacoplador 4N25 cuya función fue de aislar la señal AC del Arduino, además de generar el pulso digital cuando la onda AC cruza por cero, la cual fue detectada por el Arduino. En la Figura N°40 se muestra el esquemático del circuito de detector por cero diseñado.

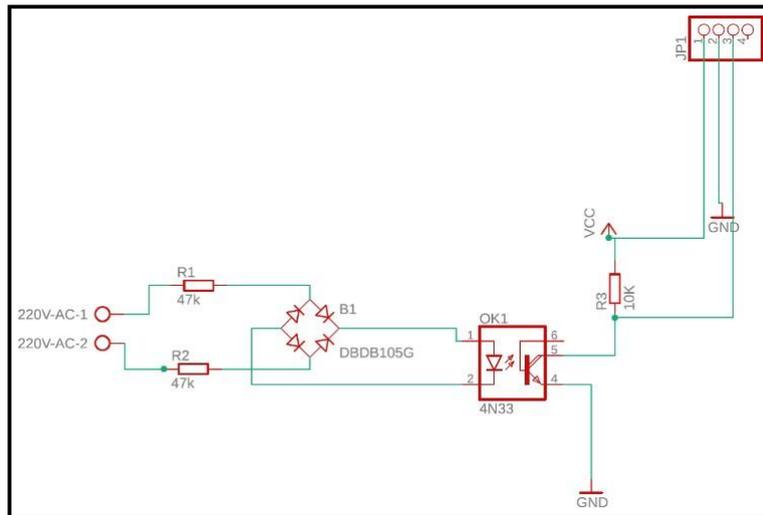


Figura N°40: Esquemático circuito de detector cruce por cero

Fuente: Elaboración Propia usando el software Eagle

Por otro lado, el circuito de potencia fue diseñado para controlar la potencia disipada por la resistencia cuando el Arduino envía la señal de activación al Triac. El circuito de potencia utilizó un circuito que aísla la etapa de potencia y el Arduino, por ello se usó un optoacoplador MOC3021; de esta manera, se protege la parte de control y la parte de potencia. El otro elemento importante del circuito es el Triac BT138. Se puede observar en la Figura N°41 el esquemático del circuito de potencia.

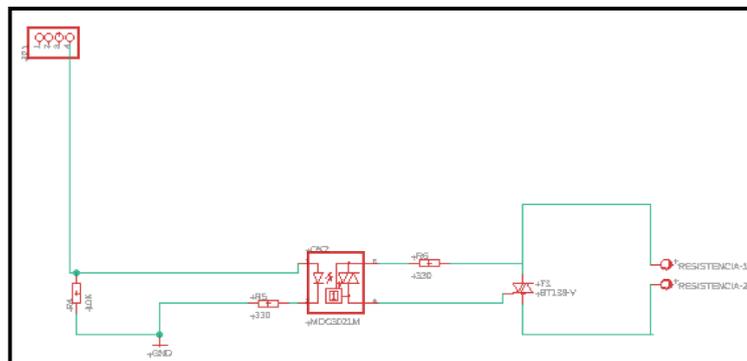


Figura N°41: Esquemático circuito de potencia

Fuente: Elaboración Propia usando el software Eagle

Para este trabajo de tesis se elaboró una tarjeta PCB para cada uno de los circuitos explicados. Se puede observar en la Figura N°42 el diseño de las tarjetas PCB implementadas para el prototipo.

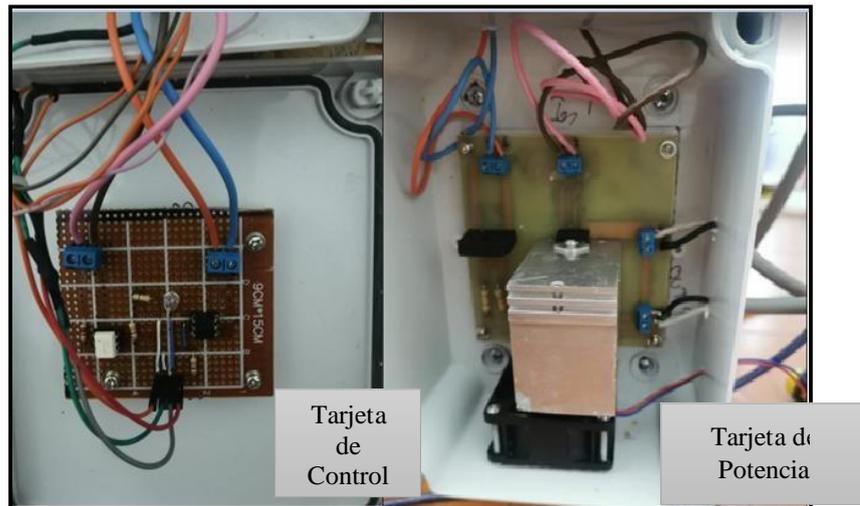


Figura N°42: Tarjetas PCB de Control y Potencia para el dimmer

Fuente: Elaboración Propia

4. Termostato: Se instaló un termostato para la resistencia sumergible como un respaldo de seguridad para la bañera de canes. El termostato se encargará de cortar la energía de la resistencia, en caso la temperatura del agua siga subiendo, por lo que se consideró como temperatura de corte 40 °C, ya que el termostato esta graduado cada 5 grados centígrados. La Figura N° 43 muestra el termostato instalado en la resistencia.



Figura N°43: Termostato de respaldo

Fuente: Elaboración Propia

5. Tarjeta de interfaz: Se diseñó una tarjeta PCB de interfaz entre los sensores y actuadores con el módulo Arduino, esto porque se deseaba contar con una mejor distribución de cableado desde los dispositivos hasta la tarjeta de control. Además, todos los sensores y ciertos actuadores utilizaron la misma tensión de 5V para trabajar; por ello se implementó un módulo regulador de tensión, debido a que la fuente de tensión brinda 12v y los sensores se alimentan a 5V. Se puede observar en la Figura N°44 la tarjeta de interfaz de los sensores implementada en el prototipo.

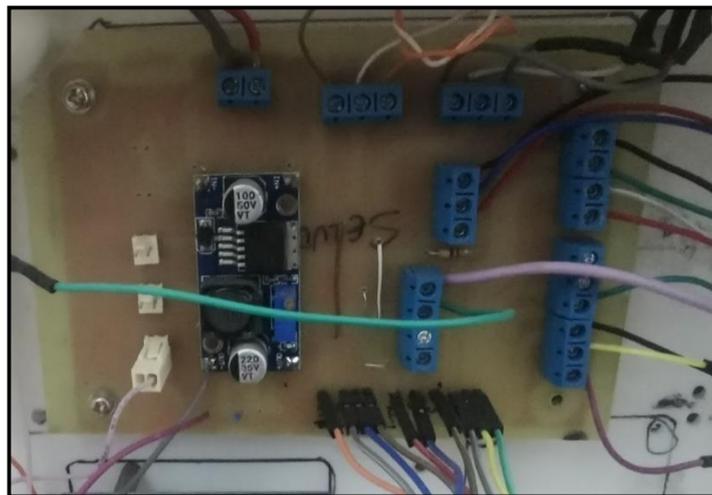


Figura N°44: Tarjeta interfaz sensores y actuadores

Fuente: Elaboración Propia

6. Módulo Arduino: Para este trabajo de tesis se utilizó un módulo Arduino como dispositivo de interfaz entre el software LabVIEW y el prototipo, es decir como una tarjeta de adquisición de datos. Esta tarjeta se encargó de interpretar las señales recibida por los sensores y enviadas al LabVIEW para que las pueda interpretar y procesar. Finalmente, el LabVIEW envía una señal para que el Arduino actúe sobre los actuadores según el proceso que está realizando. Se puede observar en la Figura N°45 la tarjeta de adquisición de datos Arduino implementada para el prototipo.

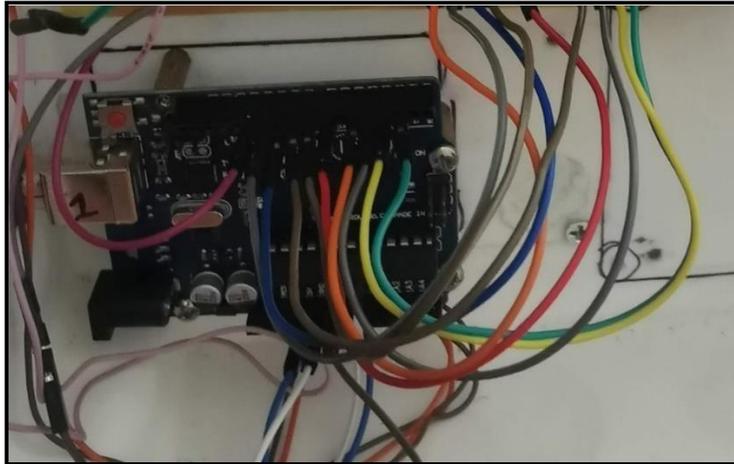


Figura N°45: Tarjeta de adquisición de datos arduino

Fuente: Elaboración Propia

Además, la tarjeta Arduino fue programada para que pueda recibir y enviar los datos de los sensores al LabVIEW por el puerto serial. Por ello, se usó el Arduino IDE donde se escribió el programa, así como también se emplearon librerías de programación para hacer más sencillo el programa final, tales como el “One Wire” utilizado para el sensor de temperatura DB18B20 y “ProtoThreads” que facilita la ejecución de varios programas en simultaneo. Esta última librería ayuda a que el Arduino constantemente envíe los datos de los sensores y reciba la orden del LabVIEW al mismo tiempo. Además, en la programación se realizó la distribución de los pines para los sensores y actuadores según la Tabla N°6.

Tabla N°6: Distribución de Pines en Arduino Uno

PIN	TIPO	FUNCIÓN
0	-	-
1	-	-
2	DATA	TRIGGER
3	DATA	ECHO
4	DATA	DS18B20
5	DATA	DTH
6	-	LIBRE
7	ON/OFF	ELECTROBOMBA
8	ON/OFF	E.V 1 (AGUA GENERAL)
9	PWM	RESISTENCIA
10	ON/OFF	E.V 2 (AGUA. ENJUAGE)
11	ON/OFF	E.V 3 (SHAMPOO)
12	ON/OFF	LED RESISTENCIA/MOTOR
13	-	LIBRE

Fuente: Elaboración Propia

Adicionalmente, se utilizó otro módulo Arduino para el control del dimmer. Es decir, para poder controlar la potencia disipada de la resistencia por el dimmer se usó un segundo módulo Arduino, que fue controlado por el Arduino principal el cual está conectado al LabVIEW. Este control se basó en la conversión de la señal PWM a Analógica enviada por el principal Arduino y leída por el segundo Arduino. Este último módulo fue programado para controlar a la resistencia mediante el método del disparo del triac. Y, asimismo, se utilizó un filtro pasabajo del tipo RC construido con una resistencia de $4.7K\Omega$ y un condensador de $10\mu F$, para convertir la señal PWM enviada por el Arduino principal a una analógica de 0-5V que fue leída con el segundo módulo. En la Figura N°46 se puede observar el Arduino y el filtro pasabajo implementado.

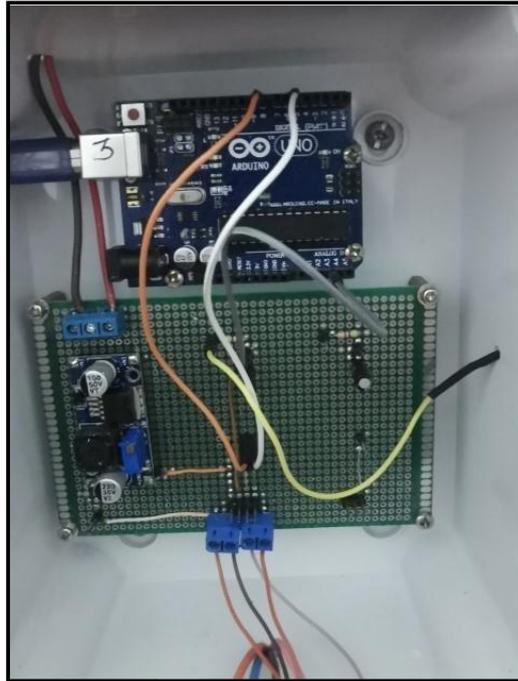


Figura N°46: Arduino y filtro pasabajo

Fuente: Elaboración Propia

Finalmente, en la Figura N°47, se puede observar la implementación final del prototipo de bañera para canes. En ella se puede observar el tanque de agua, el depósito de drenaje, el sistema de ingreso de agua, la cabina para la bañera y la bañera de PVC, donde ingresa el can para el bañado.

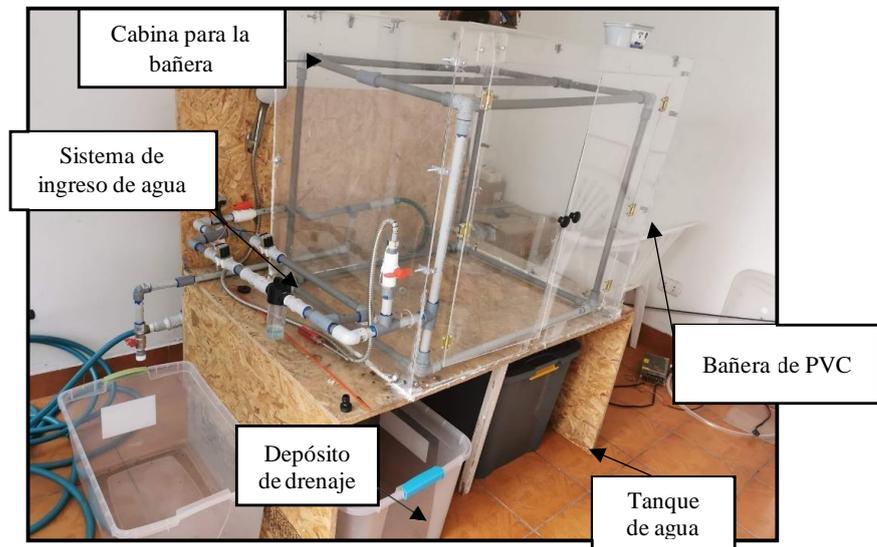


Figura N°47: Implementación final del prototipo de bañera para canes

Fuente: Elaboración Propia

3.3. Implementación del controlador ON/OFF de nivel

Se inició la implementación del controlador ON/OFF de nivel con la programación del sensor de ultrasonido en el módulo Arduino donde se validó el funcionamiento del sensor. Luego, se realizó una programación en LabVIEW para recibir la información enviada por el sensor, como se puede observar en la Figura N°48.

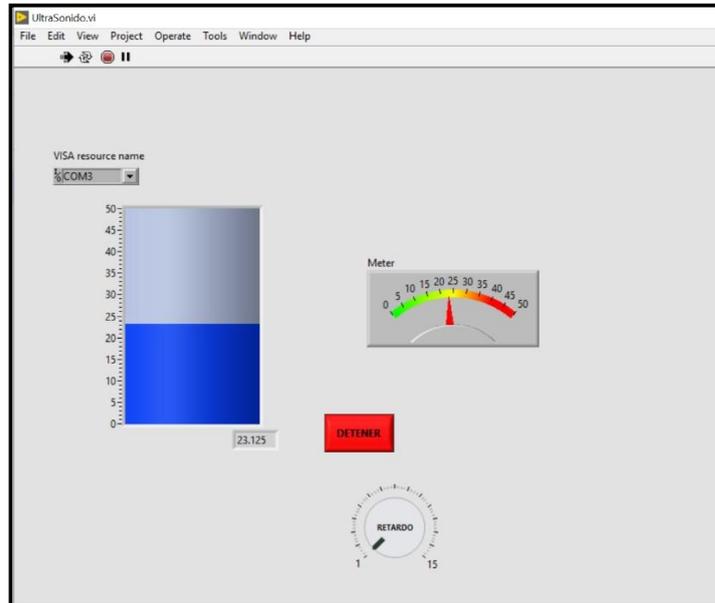


Figura N°48: Funcionamiento de sensor de ultrasonido en LabVIEW

Fuente: Elaboración Propia usando el software LabVIEW

Después de lograr la comunicación entre el módulo Arduino y el LabVIEW, se realizó la implementación del controlador ON/OFF de nivel; para ello, fue necesario hallar una ecuación que convierte la señal enviada por el sensor en cm a litros. Para ello, se necesitó adquirir datos mientras se llenaba el tanque de forma manual a través de un recipiente con capacidad de 5L. Estos datos fueron registrados en una hoja de cálculo donde se encontró la ecuación que satisface a la conversión de distancia en cm a volumen en litros. En la Tabla N°7 se puede observar la tabla con los datos registrados.

Tabla N°7: Registro de datos distancia y Volumen

N	Distancia(cm)	Volumen (L)
0	39	0
1	37	5
2	36	10
3	34	15
4	32	20
5	30	25
6	28	30
7	26	35
8	25	40
9	23	45
10	22	50
11	20	55
12	19	60
13	17	65
14	15	70
15	14	75
16	13	80

Fuente: Elaboración Propia

Asimismo, en la Figura N°49 se observa la gráfica correspondiente a los datos registrados y con la respectiva ecuación.

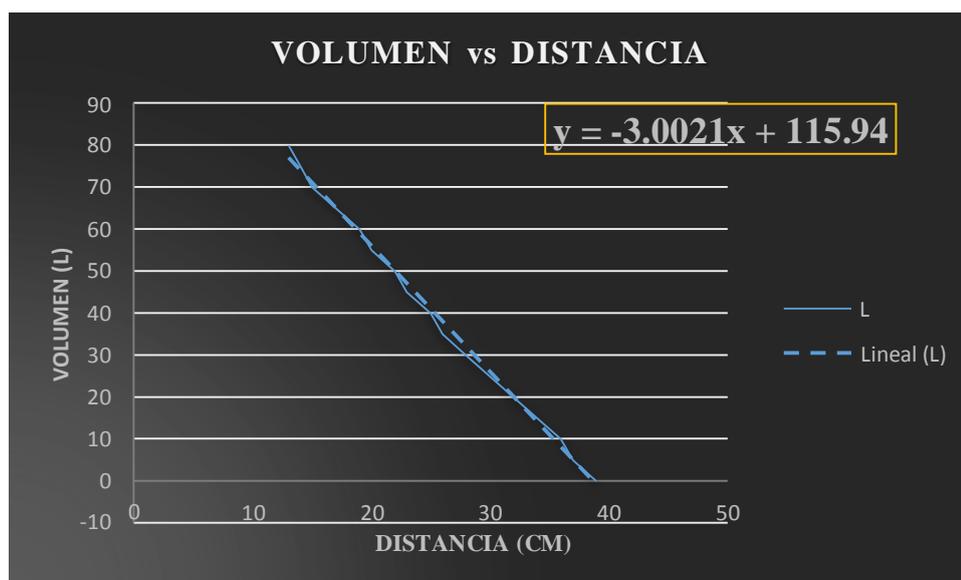


Figura N°49: Gráfica Volumen vs Litros de los datos registrados

Fuente: Elaboración Propia usando el software MS Excel

Con la obtención de la ecuación se pudo completar la programación e implementación del controlador ON/OFF en el software LabVIEW. Se puede ver en la Figura N°50 la ventana de diagrama de bloques de la programación del controlador ON/OFF de nivel.

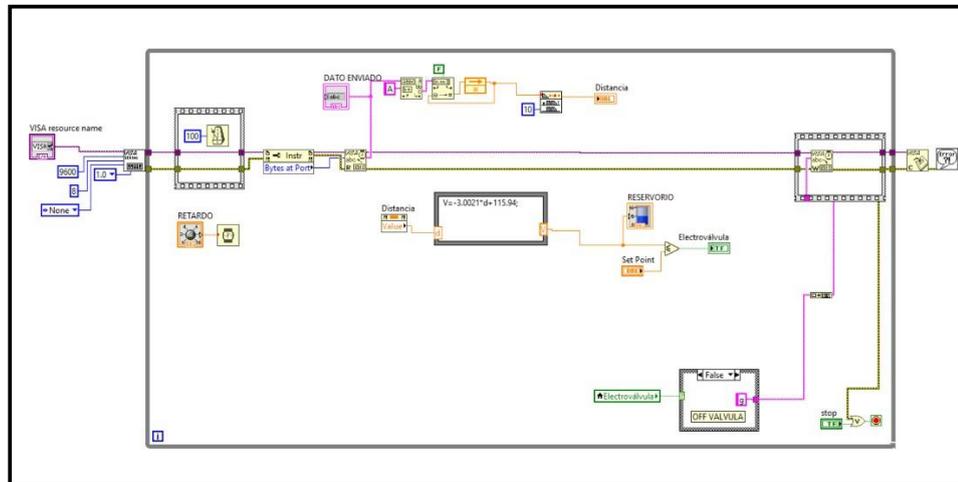


Figura N°50: Programación en LabVIEW del controlador ON/OFF de nivel

Fuente: Elaboración Propia usando el software LabVIEW

En cuanto al funcionamiento, este consistió en adquirir la señal del sensor de ultrasonido, mediante el módulo Arduino y enviada al LabVIEW. Luego, se ingresó el Set Point deseado para que el software haga la comparación del valor recibido con el valor deseado. Cuando el valor ingresado fue mayor al del medido, entonces el LabVIEW envió una señal al módulo Arduino para que active la electroválvula 1 e ingrese agua al tanque. En cuando el valor medido alcanza al valor deseado, la electroválvula 1 se desactiva evitando que ingrese más líquido al tanque. Se puede observar en la Figura N°51 la ventana panel frontal implementado para el control ON/OFF en el LabVIEW.

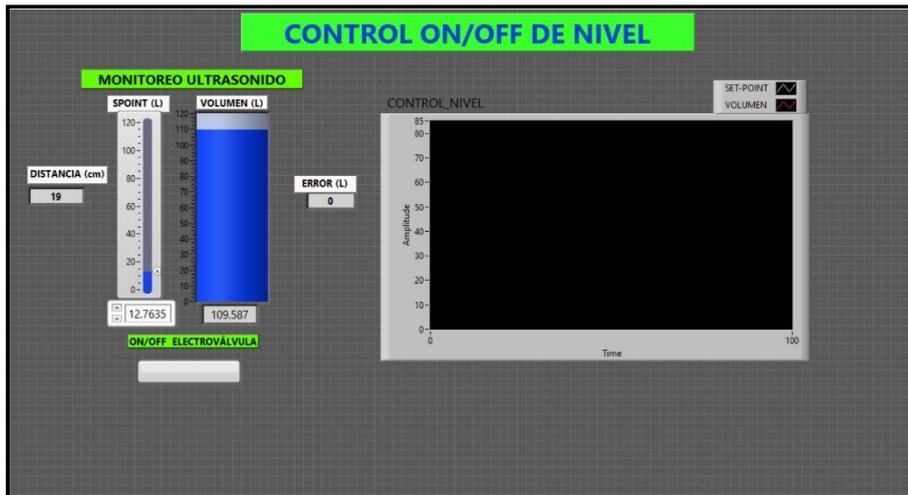


Figura N°51: Panel frontal de la implementación del controlador ON/OFF de nivel

Fuente: Elaboración Propia

3.4. Implementación y sintonización del controlador PID de temperatura

El desarrollo del controlador PID de temperatura se realizó en 2 partes: Implementación del Controlador PID y Sintonización del Controlador PID. A continuación, se describe cada uno de ellos:

3.4.1. Implementación del controlador PID

Inició con la programación en el módulo Arduino para leer los datos enviados por el sensor de temperatura DS18B20 y con ello validar el correcto funcionamiento del sensor. Luego, se realizó un programa en el software LabVIEW para poder recoger la información enviada por el sensor de temperatura, tal como se puede observar en la Figura N° 52.

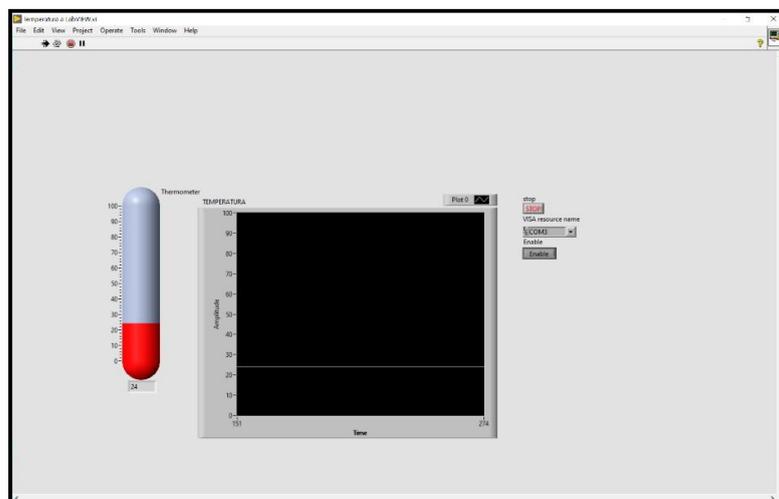


Figura N°52: Funcionamiento de sensor de temperatura en LabVIEW

Fuente: Elaboración Propia usando el software LabVIEW

Seguidamente, se realizó la implementación del controlador PID de temperatura. Para facilitar la programación se utilizó el bloque PID de la librería de controladores del software LabVIEW.

Se puede observar en la Figura N° 53 la ventana de diagrama de bloques de la programación del controlador PID de temperatura.

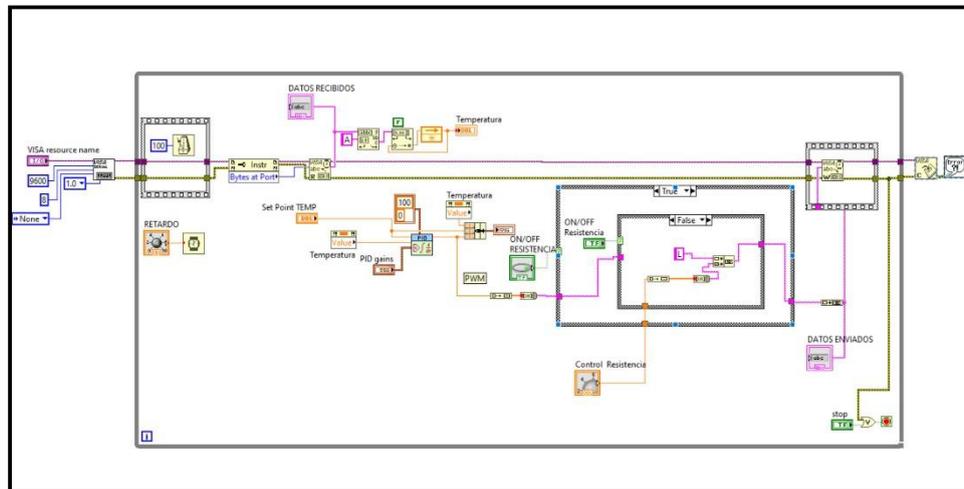


Figura N°53: Programación en LabVIEW del controlador PID de temperatura

Fuente: Elaboración Propia usando el software LabVIEW

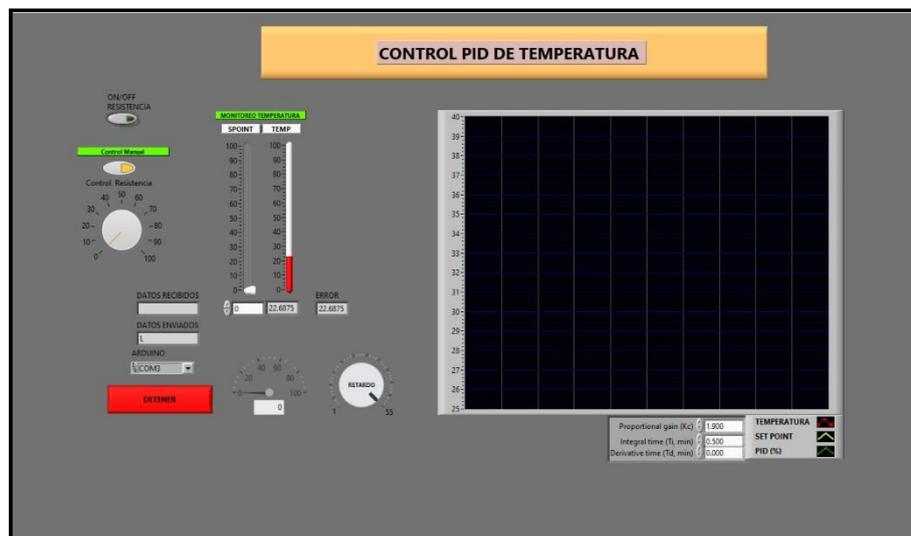


Figura N°54: Panel frontal implementación del controlador PID de temperatura

Fuente: Elaboración Propia usando el software LabVIEW

En la Figura N°54 se muestra el panel frontal implementado para el controlador PID de temperatura, donde se observa la temperatura del agua medida por el sensor, el Set Point, las ganancias del controlador PID y la salida del controlador PID.

El controlador inicia su funcionamiento cuando el valor de temperatura de Set Point es mayor al valor de temperatura medida, enviando una señal a la resistencia sumergible para que empiece a calentar el agua del tanque para así alcanzar la temperatura deseada.

La salida del controlador PID es de 0% a 100% la cual es enviada al módulo Arduino como una señal PWM que controla a la resistencia sumergible, y disipando la potencia requerida por el controlador.

Para el correcto funcionamiento del controlador implementado fue necesario antes realizar la sintonización de este para encontrar las correctas ganancias de K_p , T_i y T_d . A continuación, se muestra el método de sintonización del controlador PID de Temperatura para el presente trabajo de tesis.

3.4.2. Sintonización del controlador PID

Para la sintonización del controlador PID de temperatura del presente trabajo de tesis se usó el método de “prueba y error”, teniendo como referencia los parámetros de las ganancias K_p , T_i y T_d de la tabla de la Figura N°55.

Parámetro Controlador	Probable Mínimo	Probable Máximo	Valor Típico	Unidades
Presión Líquidos y Fluidos				
Ganancia	0.2	1.5	0.5	ganancia
Integral	0.05	0.5	0.2	minutos
Derivativa	0	0	0	minutos
Filtro	0	0.2	0.02	minutos
Muestreo	0.5	2	1	segundos
Temperatura				
Ganancia	0.5	4	1	ganancia
Integral	0.2	1	0.5	minutos
Derivativa	0	0.25	0.1	minutos
Filtro	0	0.1	0	minutos
Muestreo	1	5	2	segundos

Figura N°55: Tabla parámetros de ganancia controlador PID

Fuente: <https://instrumentacionycontrol.net/que-valores-iniciales-usar-en-un-pid/>

Se consideró la implementación de un nuevo programa donde se añadió el controlador ON/OFF de nivel al panel frontal del controlador PID de temperatura, esto para monitorear el nivel de agua del tanque mientras se realiza el control de temperatura. Se puede observar en la Figura N° 56 el panel

frontal con la implementación de los controladores ON/OFF de nivel y PID de temperatura.

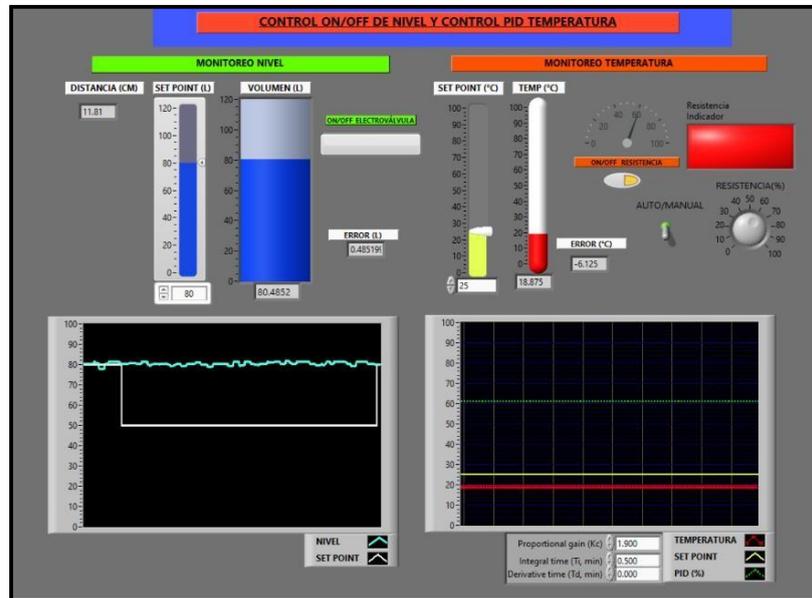


Figura N°56: Panel frontal implementación controladores ON/OFF y PID

Fuente: Elaboración Propia usando el software LabVIEW

Se inició la sintonización del controlador PID usando los siguientes valores: $K_p=0.2$, $T_i=0.05$ y $T_d=0$ con un Set point de $22\text{ }^\circ\text{C}$. Se muestra en la ecuación N°4, la ecuación del controlador PID con los valores iniciales:

$$C_{pid} = 0.2 \left(1 + \frac{1}{0.05s} \right)$$

Ecuación N°3: Ecuación controlador PID valores iniciales

Se observó que la respuesta del controlador con estos parámetros fue de lenta reacción, ya que la resistencia aumentaba su potencia de manera muy pausada. Este comportamiento hace que llegar a la temperatura deseada tome más tiempo.

Se puede observar en la Figura N° 57 el controlador PID con los parámetros configurados y la potencia de la resistencia que se encuentra en 10% y subiendo lentamente.

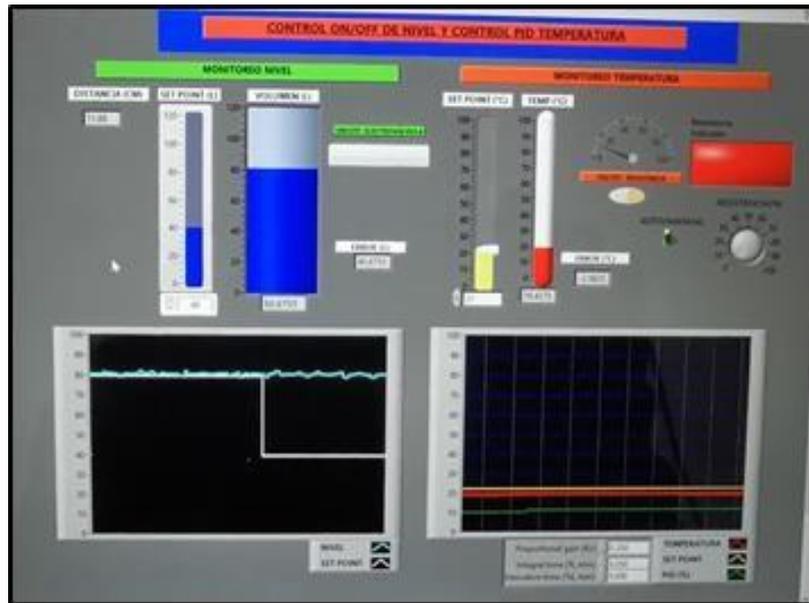


Figura N° 57: Controlador PID con parámetros $K_p= 0.2$, $T_i=0.05$ y $T_d= 0$

Fuente: Elaboración Propia usando el software LabVIEW

Al notar el comportamiento del controlador, se decidió cambiar la ganancia de la parte integral, esto porque se deseaba una acción más rápida del controlador ante el error acumulado, y para que la resistencia se active más rápido y llegue a su máxima potencia. Se cambió la ganancia integral de 0.05 a 0.001. Se puede observar en la Figura N° 58 como la salida del controlador tiene una mejor respuesta ante el error respecto al Set-Point, el cual en esta segunda prueba fue de 25 °C y las ganancias de la parte proporcional y derivativo se mantuvieron con los mismos valores que en el primer paso.

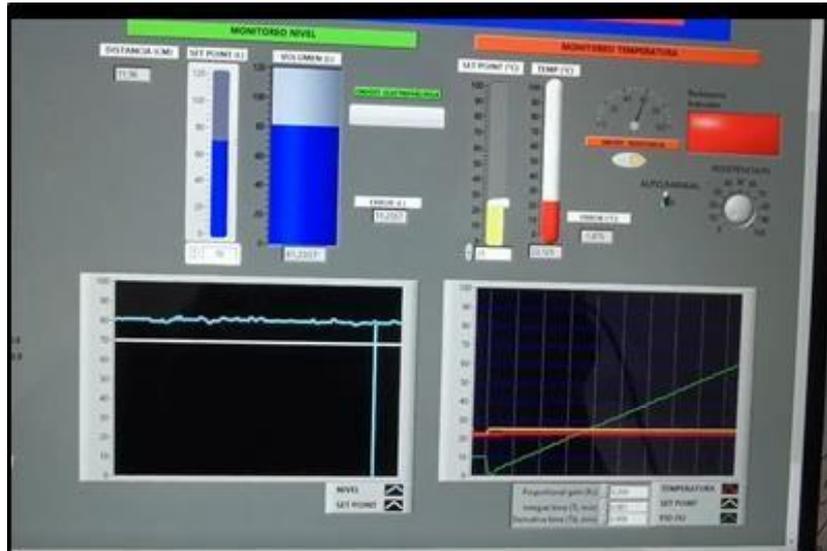


Figura N° 58: Controlador PID con parámetros $K_p= 0.2$, $T_i=0.001$ y $T_d= 0$

Fuente: Elaboración Propia usando el software LabVIEW

Por último, como siguiente prueba se modificó la ganancia proporcional y ganancia derivativa como ajuste final para lograr que el controlador tenga una mejor respuesta. Se encontró que las ganancias finales para el controlador fueron: $K_p= 0.5$, $T_i=0.001$ y $T_d= 0.02$. Así mismo, se muestra en la ecuación N°5, la ecuación final para el controlador PID y la Figura N°59 la salida del controlador con los valores finales.

$$C_{pid} = 0.5 \left(1 + \frac{1}{0.001s} + 0.02s \right)$$

Ecuación N°4: Ecuación controlador PID valores finales

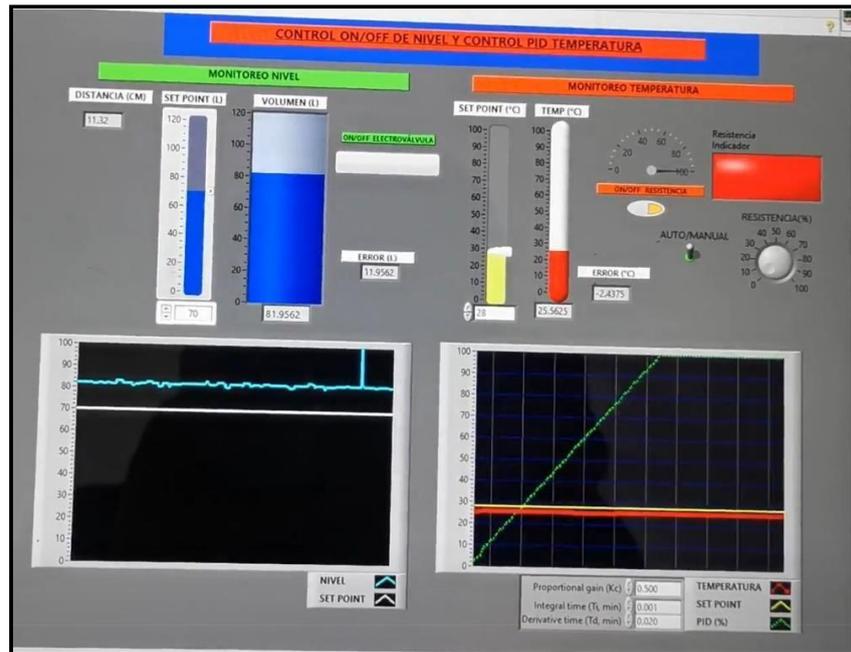


Figura N° 59: Controlador PID con parámetros $K_p=0.5$, $T_i=0.001$ y $T_d=0.02$

Fuente: Elaboración Propia usando el software LabVIEW

3.5. Implementación del controlador secuencial

La implementación del controlador secuencial consistió en realizar un conjunto de secuencias para el proceso del bañado del can. Se inició con una programación en el módulo Arduino para controlar la activación o desactivación de las electroválvulas 2 y 3. Luego, se realizó la programación en el LabVIEW donde se utilizó la librería de estructura de secuencia. Las secuencias programadas fueron: “Mojado”, “Ingreso de Shampoo” y “Enjuague” en ese orden, la cuales pasan de una a otra según el tiempo (en segundos) ingresado por el usuario.

En la Figura N°60 se puede observar la programación realizada en el LabVIEW.

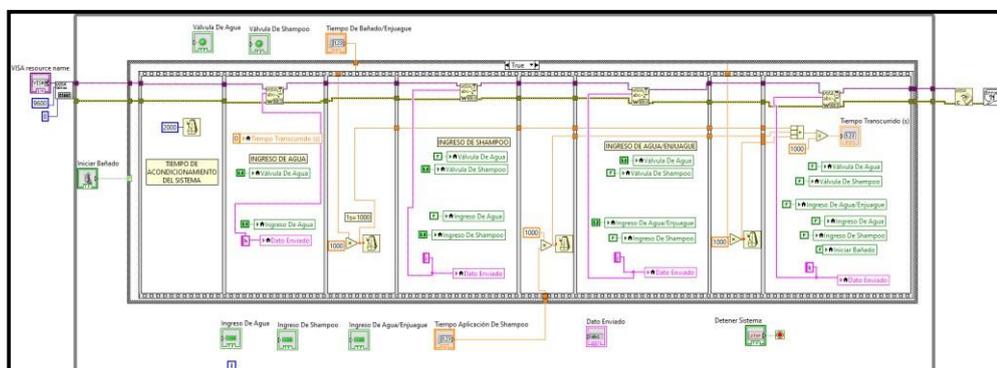


Figura N°60: Programación controlador secuencia

Fuente: Elaboración Propia usando el software LabVIEW

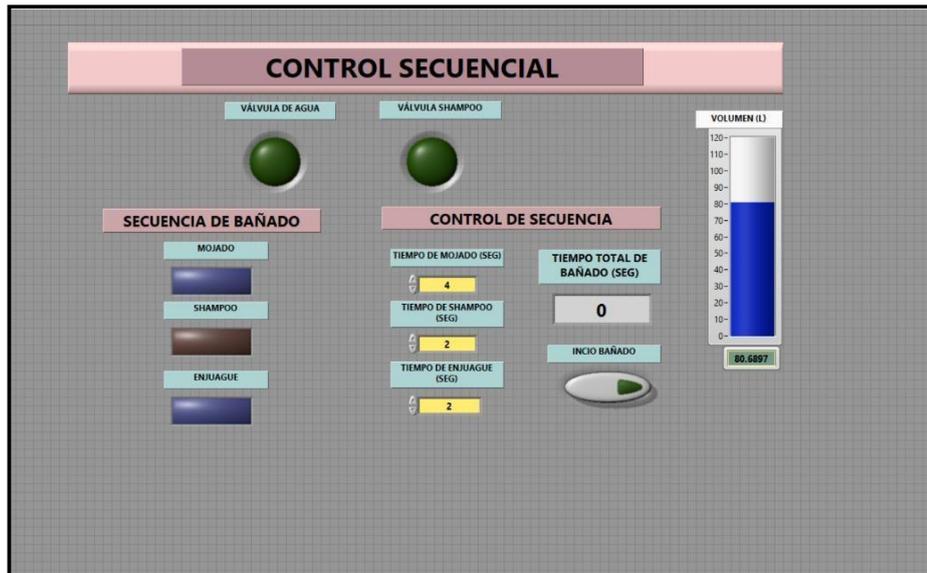


Figura N°61: Panel frontal de la implementación del controlador Secuencial

Fuente: Elaboración Propia usando el software LabVIEW

Se puede observar en la Figura N°61 el panel frontal implementado para el control secuencial.

La programación del controlador secuencial en LabVIEW consistió en asignar una letra a cada secuencia: (h): Mojar, (i): Shampoo y (j): Enjuague, que fueron enviadas hacia el módulo Arduino. Luego, el módulo Arduino recibe esta señal y la interpreta activando o desactivando la electroválvula correspondiente a cada secuencia. Se puede observar en la Tabla N° 8 la relación de secuencias para la activación y desactivación de las electroválvulas según el orden que le corresponde. Además, se puede observar que la electrobomba está activada durante toda la secuencia de bañado, debido a que se necesita transportar el agua del tanque a la bañera.

Tabla N°8: Resumen de Secuencia de lavado

Letra	Secuencia	Electroválvula 2	Electroválvula 3	Bomba de Agua
h	Inicio- Mojar Can	Encendida	Apagada	Encendida
i	Shampoo	Apagada	Encendida	Encendida
j	Enjuague	Encendida	Apagada	Apagada
k	Fin de Ciclo de Bañado	Apagada	Apagada	Apagada

Fuente: Elaboración Propia

3.6. Implementación de la interfaz de usuario de la bañera casera automatizada para canes

Por último, se realizó la implementación de una interfaz de usuario, donde se integró los controladores ON/OFF de nivel, PID de temperatura y el controlador secuencial. Se puede observar en la Figura N°62 la interfaz de usuario para la bañera. Además,

en los anexos 1 y 2 se puede observar la programación final en el módulo Arduino, así como también la programación final en LabVIEW.

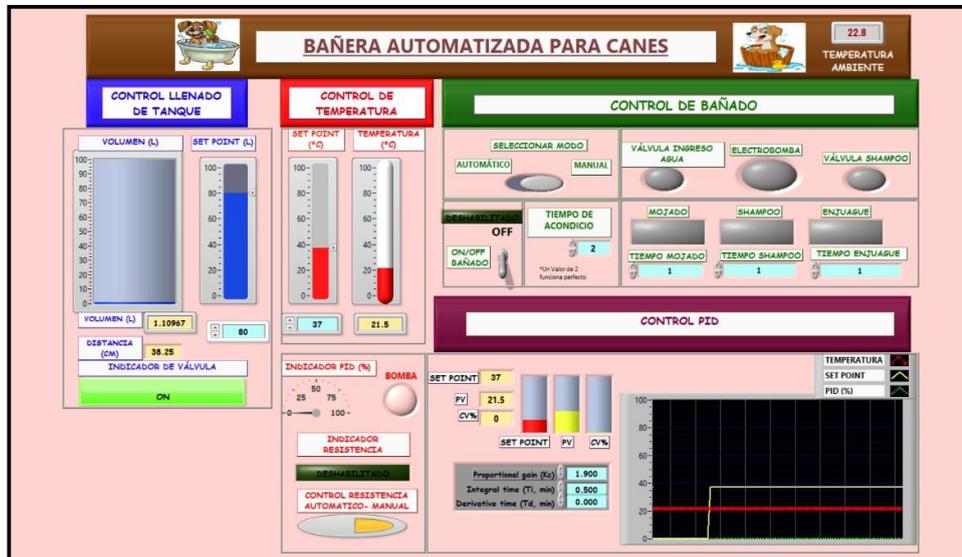


Figura N°62: Implementación de la interfaz para el usuario

Fuente: Elaboración Propia usando el software LabVIEW

CAPÍTULO IV: PRUEBAS Y RESULTADOS

En este capítulo se mostrarán las pruebas y resultados de cada uno de los controladores por separado: Control ON/OFF de nivel, Control PID de temperatura y el controlador secuencial. Por último, se mostrarán las pruebas finales con los 3 controladores unidos y la prueba del baño para un can.

4.1. Pruebas y resultados del controlador ON/OFF de nivel:

Se realizaron 4 pruebas con el Controlador ON/OFF utilizando 4 Set Point diferentes, estos fueron: 20 litros, 40 litros, 60 litros y 80 litros; cada prueba inició con el tanque vacío. A continuación, se muestran las pruebas realizadas:

4.1.1. Análisis del resultado para un SP= 20L:

Se inició con un Set Point de 20 litros, tal como se puede observar en la Figura N° 63. El tiempo de llenado del tanque de agua fue de 1.4 minutos con un error final de 1.8 litros respecto al Set Point.

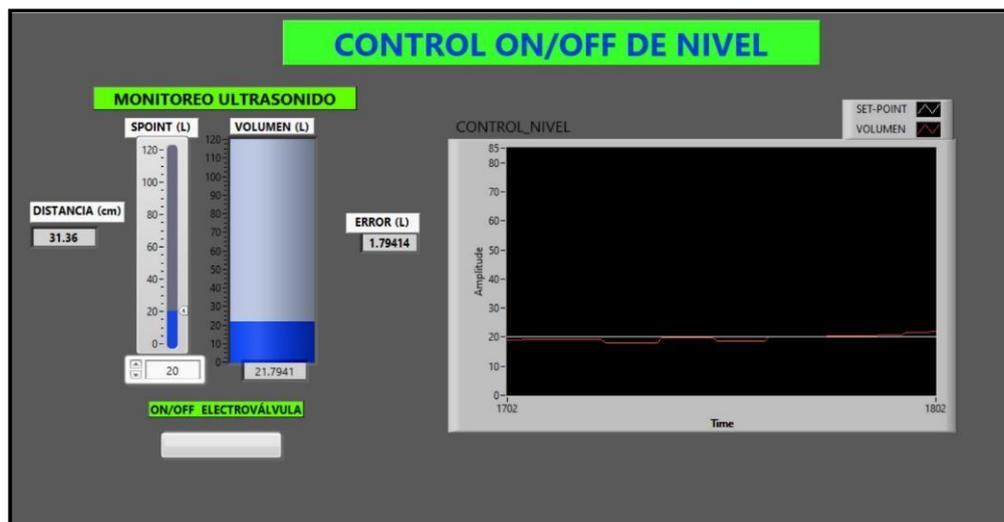


Figura N° 63: Respuesta del controlador ON/OFF de nivel SP=20L

Fuente: Elaboración Propia usando el software LabVIEW

4.1.2. Análisis del resultado para un SP= 40L:

Luego se seleccionó un Set Point de 40 litros, el tiempo que tomó para el llenado del tanque fue de 3.15 minutos. El error final fue de 1.39 litros respecto al Set point. Se puede observar en la Figura N° 64 la respuesta del controlador ON/OFF de nivel.

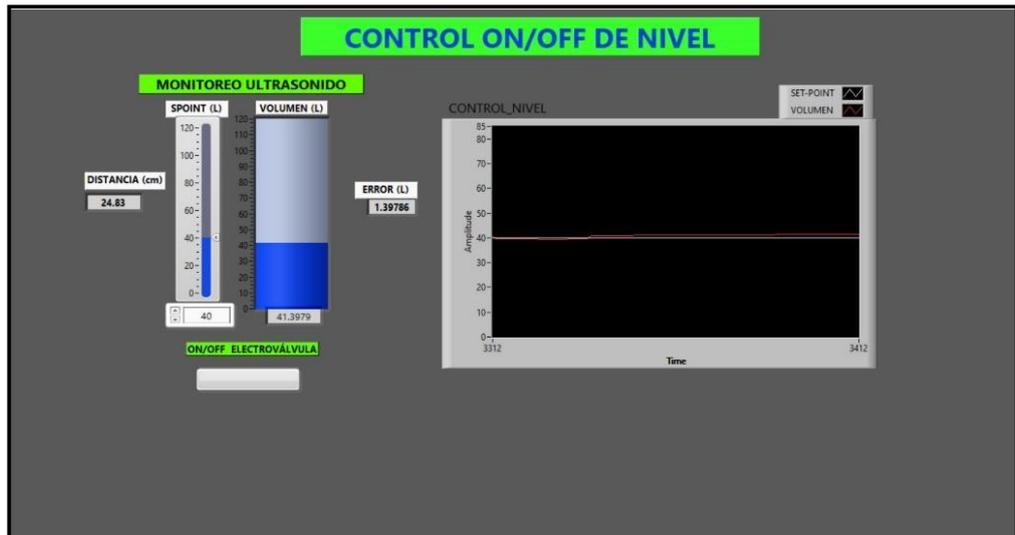


Figura N° 64: Respuesta del controlador ON/OFF de nivel SP=40L

Fuente: Elaboración Propia usando el software LabVIEW

4.1.3. Análisis del resultado para un SP=60L:

La siguiente prueba se realizó con un Set Point de 60 litros, el tiempo de llenado fue de 5.2 minutos. El error para el Set Point de esta prueba fue de 0.13 litros. Se puede observar en la Figura N° 65 la respuesta del controlador de Nivel.

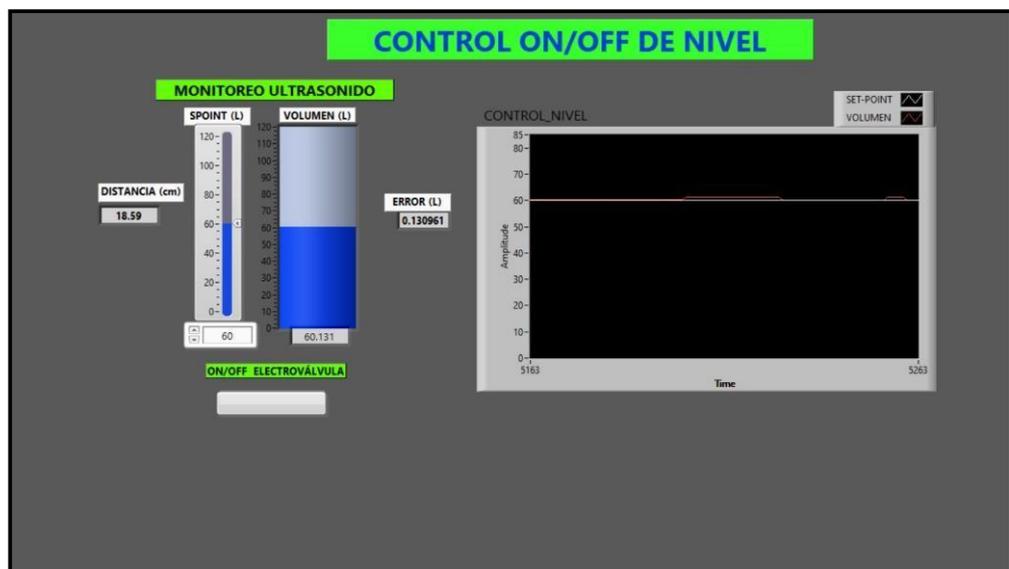


Figura N° 65: Respuesta del controlador ON/OFF de nivel SP=60L

Fuente: Elaboración Propia el software LabVIEW

4.1.4. Análisis del resultado para un SP=80L:

La última prueba realizada fue con un Set Point de 80 litros, donde el tiempo de llenado fue de 7.18 minutos. En esta prueba el error del controlador fue de 0.48 Litros. Se puede observar en la Figura N° 66 la respuesta del controlador usando un Set Point de 80 Litros.

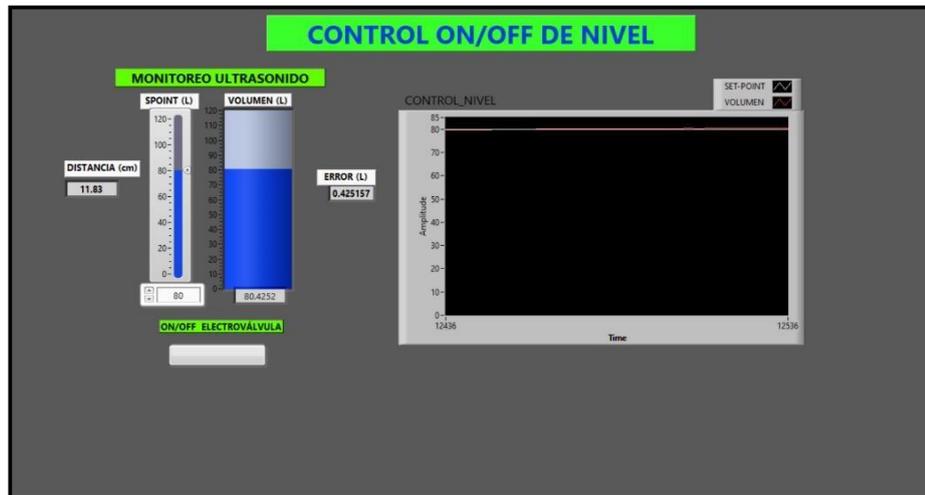


Figura N° 66: Respuesta del controlador ON/OFF de nivel SP=80L

Fuente: Elaboración Propia usando el software LabVIEW

Asimismo, se puede observar en la Figura N° 67, la gráfica del comportamiento del control ON/OFF de nivel para el llenado del tanque con un Set Point de 80 litros. En él se muestra que en los primeros 20 litros el sensor es un poco inestable en las lecturas y esto se debe a que se percibe mucho movimiento del agua por la caída de este dentro del tanque vacío. Por otra parte, las mediciones del sensor luego de 20 litros se estabilizan y las lecturas se observan con mejor comportamiento.

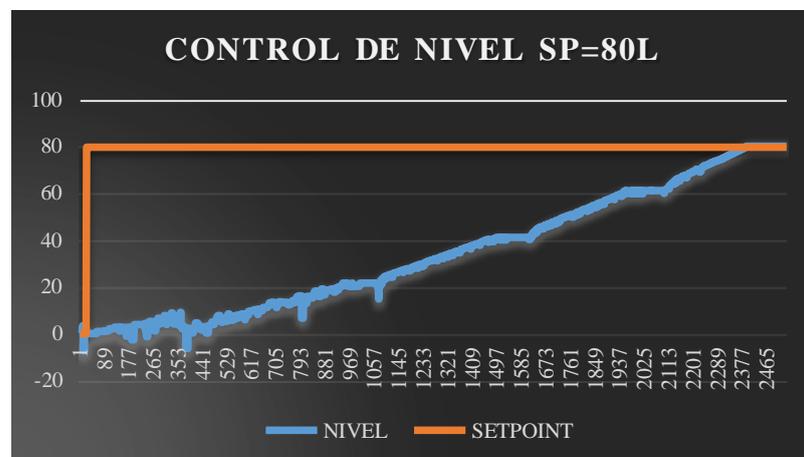


Figura N° 67: Gráfica del controlador ON/OFF de nivel

Fuente: Elaboración Propia usando el software MS Excel

Tabla N°9: Resumen de Resultados del controlador ON/OFF de nivel

LITROS (SET POINT)	VALOR MEDIDO	ERROR	TIEMPO (Min)
20	21.79	1.79	1.4
40	41.39	1.39	3.15
60	60.13	0.13	5.2
80	80.42	0.42	7.18
ERROR PROMEDIO		0.9325	

Fuente: Elaboración Propia

Finalmente, en la Tabla N°9 se observan los resultados obtenidos del controlador ON/OFF de nivel, donde se encontró que se tiene un error promedio de 0.93 litros. Además, el flujo promedio de agua al tanque es de 12.03 L/min.

4.2. Pruebas y resultados del controlador PID de temperatura:

Se realizaron 3 pruebas con el controlador PID de temperatura utilizando 3 Set Point: 25 °C, 30 °C y 37 °C donde cada prueba se inició con el tanque a 80 L y a una temperatura inicial de 19.43 °C, tal como se observa en la Figura N°68. A continuación, se muestran las pruebas realizadas.

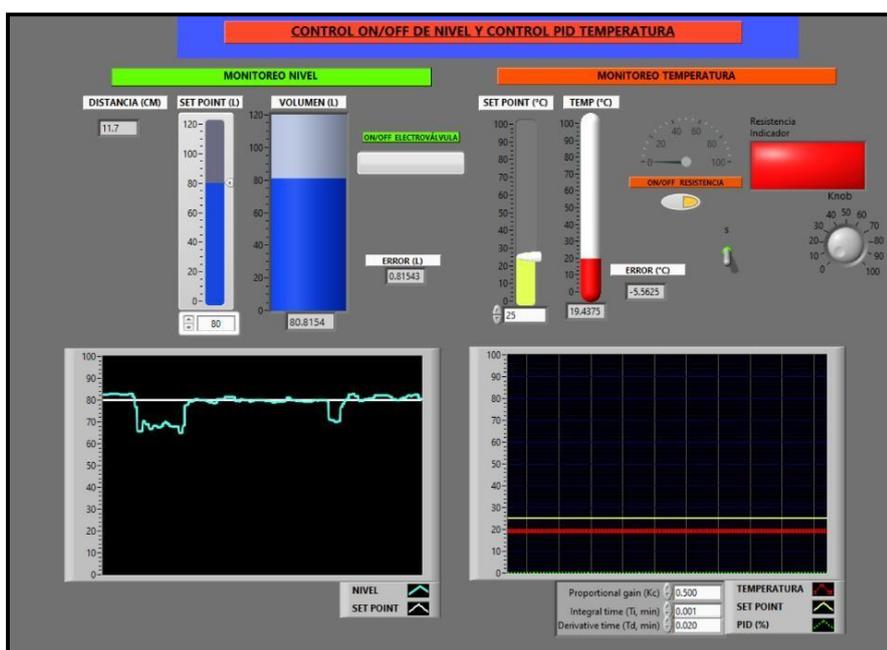


Figura N°68: Condiciones iniciales para controlador PID de temperatura

Fuente: Elaboración Propia usando el software LabVIEW

4.2.1. Análisis del Resultado para un SP=25 °C:

En la primera prueba realizada se utilizó un Set Point de 25 °C, tal como muestra la Figura N° 69. El tiempo que tomó para alcanzar la temperatura

deseada fue de 19.35 minutos y un error respecto al Set point de $-0.0625\text{ }^{\circ}\text{C}$. Además, se puede observar en la Figura N°70 la gráfica de la respuesta del controlador PID.

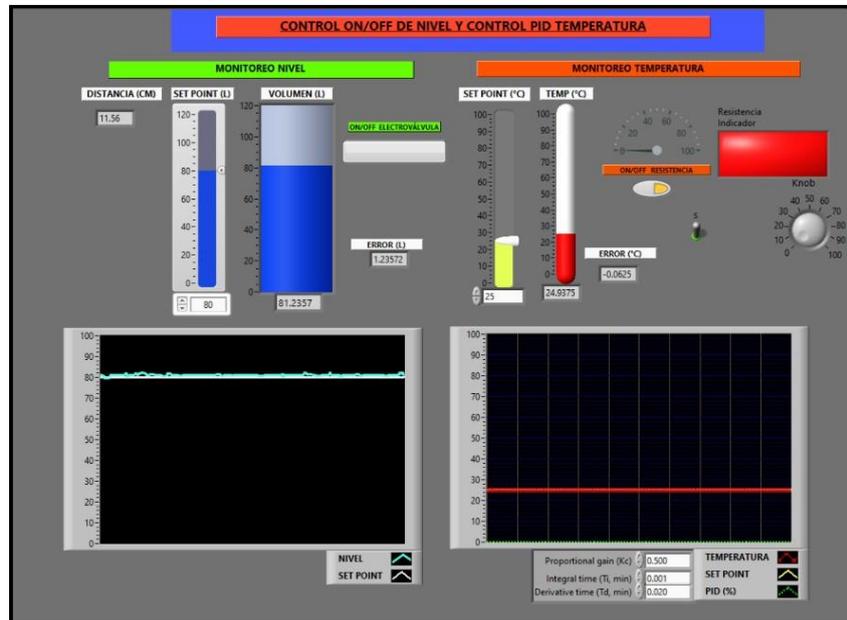


Figura N°69: Respuesta del controlador PID de temperatura SP=25 °C

Fuente: Elaboración Propia usando el software LabVIEW

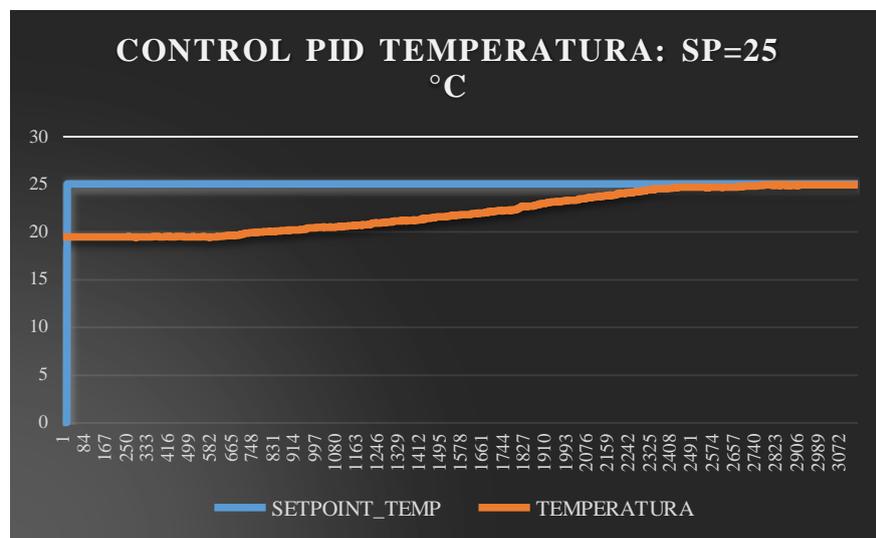


Figura 70: Gráfica del controlador PID de temperatura para SP=25 °C

Fuente: Elaboración Propia usando el software MS Excel

4.2.2. Análisis del resultado para un SP= 30 °C:

En la segunda prueba realizada se utilizó un Set Point de 30 °C, tal como muestra la Figura N° 71. El tiempo que tomó para alcanzar la temperatura deseada fue de 30.51 minutos y el error respecto al Set point fue de $0.1875\text{ }^{\circ}\text{C}$.

Además, se puede observar en la Figura N°72 la gráfica de la respuesta del controlador PID.

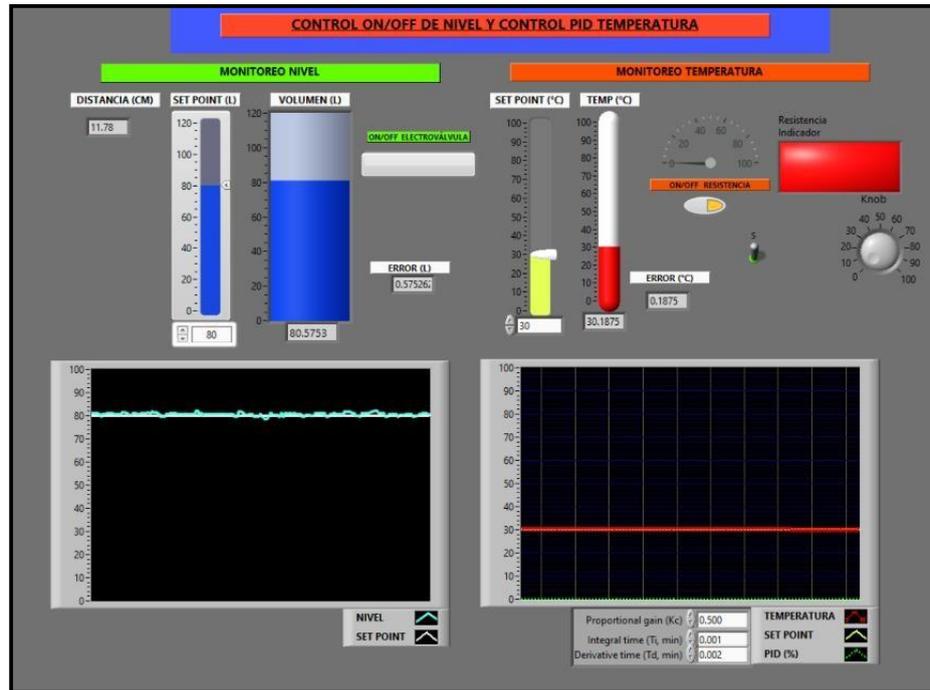


Figura 71: Respuesta del controlador PID de temperatura SP=30 °C

Fuente: Elaboración Propia usando el software LabVIEW

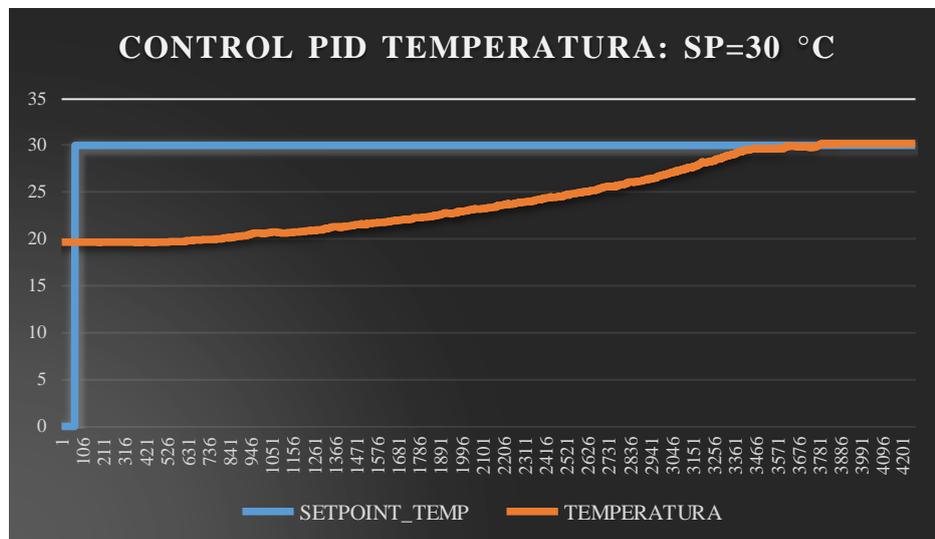


Figura N° 72: Gráfica del controlador PID de temperatura para SP=30 °C

Fuente: Elaboración Propia usando el software MS Excel

4.2.3. Análisis del resultado para un SP= 37 °C:

Para la tercera prueba se utilizó un Set Point de 37 °C, tal como se muestra en la Figura N° 73. El tiempo que tomó el controlador para alcanzar la temperatura de Set Point fue de 45.35 minutos, y el error final respecto a la temperatura

deseada fue de 0.0625 °C. Además, en la Figura N° 74 se muestra la gráfica de respuesta del controlador.

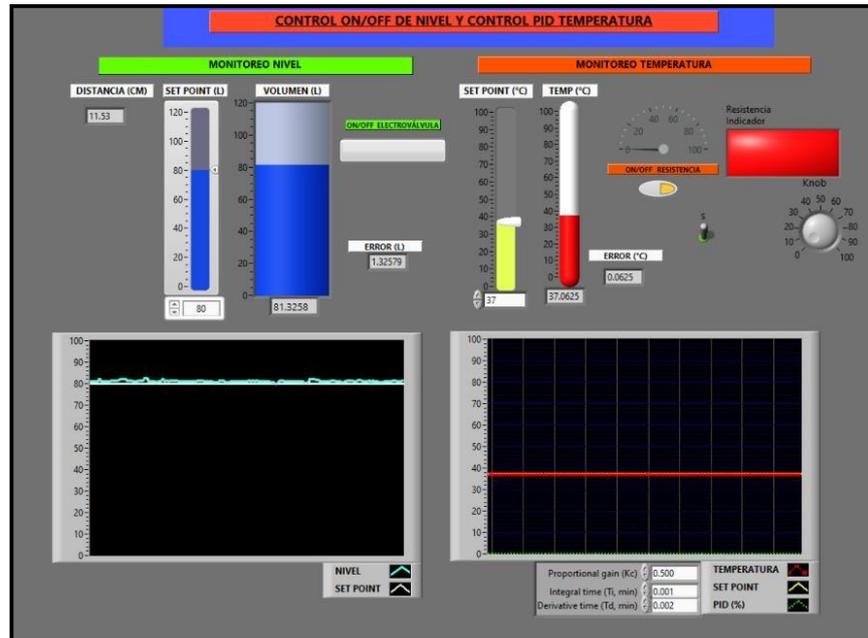


Figura 73: Respuesta del Controlador PID de Temperatura SP=37 °C

Fuente: Elaboración Propia usando el software LabVIEW

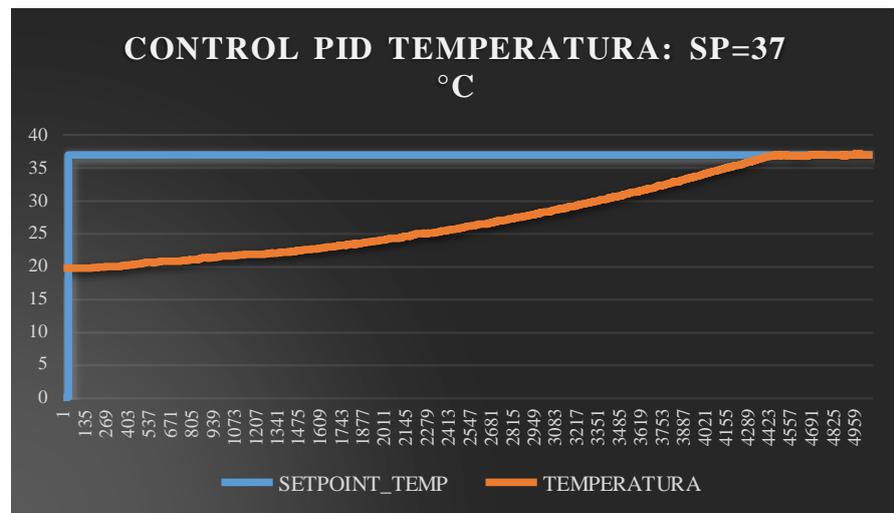


Figura N° 74: Gráfica del Controlador PID de temperatura para SP=30 °C

Fuente: Elaboración Propia usando el software MS Excel

Tabla N°10: Resumen de resultados del controlador PID de temperatura

TEMPERATURA (SET POINT)	VALOR MEDIDO	ERROR	TIEMPO (Min)
25	24.9375	-0.0625	19.75
30	30.1875	0.1875	30.51
37	37.0625	0.0625	45.35

Fuente: Elaboración Propia

Finalmente, en la Tabla N°10 se observan los resultados obtenidos con el controlador PID de temperatura donde se muestran los errores obtenidos. Para una temperatura de 37 °C, con la cual se trabajó para el bañado del can, se obtuvo un error de 0.0625 °C respecto al Set-Point y el tiempo que tomó para estabilizar a la temperatura deseada fue de 45.25 minutos.

4.3. Pruebas y resultados del controlador secuencial:

Se realizaron 3 pruebas con el controlador secuencial utilizando diferentes tiempos: 90 segundos, 120 segundos y 150 segundos. Estas pruebas representan el tiempo completo de bañado. Además, cada una se realizó con el tanque lleno con 80 litros de agua. En la Figura N° 75, se muestra el panel frontal inicial del control secuencial.



Figura N°75: Panel Frontal inicial del control secuencial

Fuente: Elaboración Propia usando el software LabVIEW

A continuación, se muestran las pruebas realizadas:

4.3.1. Análisis del resultado para T= 90 Segundos:

Para la primera prueba realizada se configuró el controlador para un tiempo total de bañado de 90 segundos. Donde el tiempo de “Mojado” fue de 15 segundos, tiempo de “Shampoo” 15 segundos y el “Enjuague” fue de 60 segundos. Para esta prueba se consumieron 16.33 Litros de agua. En la Figura N° 76, se observa la respuesta del control secuencial.

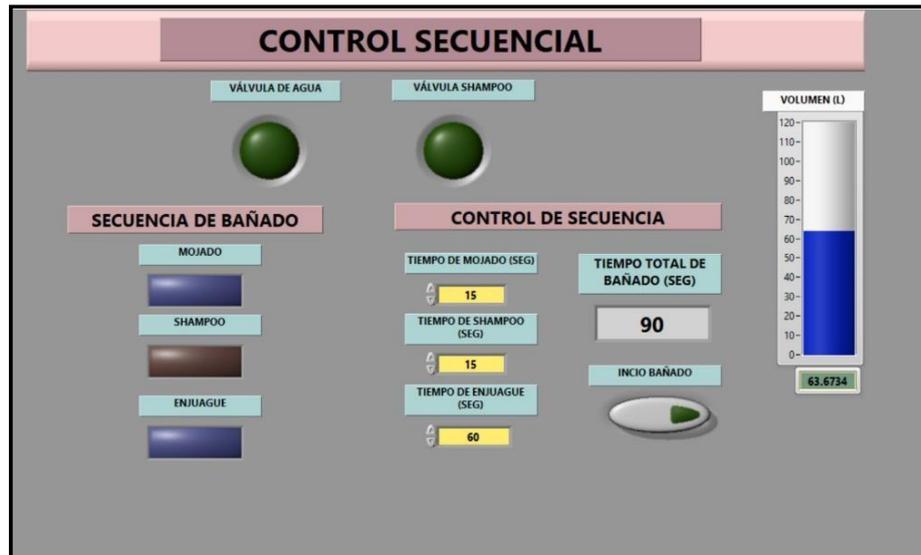


Figura N°76: Respuesta del controlador secuencial para un T=90 segundos

Fuente: Elaboración Propia usando el software LabVIEW

4.3.2. Análisis del resultado para T= 120 Segundos:

En la segunda prueba realizada se configuró el controlador para un tiempo total de bañado de 120 segundos. Donde el tiempo de “Mojado” fue de 30 segundos, tiempo de “Shampoo” 30 segundos y el “Enjuague” fue de 60 segundos. Para esta prueba se consumieron 28.64 Litros de agua. En la Figura N° 77, se observa la respuesta del control secuencial.



Figura N°77: Respuesta del controlador secuencial para un T=120 segundos

Fuente: Elaboración Propia usando el software LabVIEW

4.3.3. Análisis del resultado para T=150 Segundos:

En la última prueba realizada se configuró el controlador secuencial para un tiempo total de 150 segundos. Donde el tiempo de “Mojado” fue de 30 segundos, tiempo de “Shampoo” 45 segundos y el “Enjuague” fue de 75 segundos. Para esta prueba se consumieron 34.54 Litros de agua. En la Figura N° 78, se observa la respuesta del control secuencial.

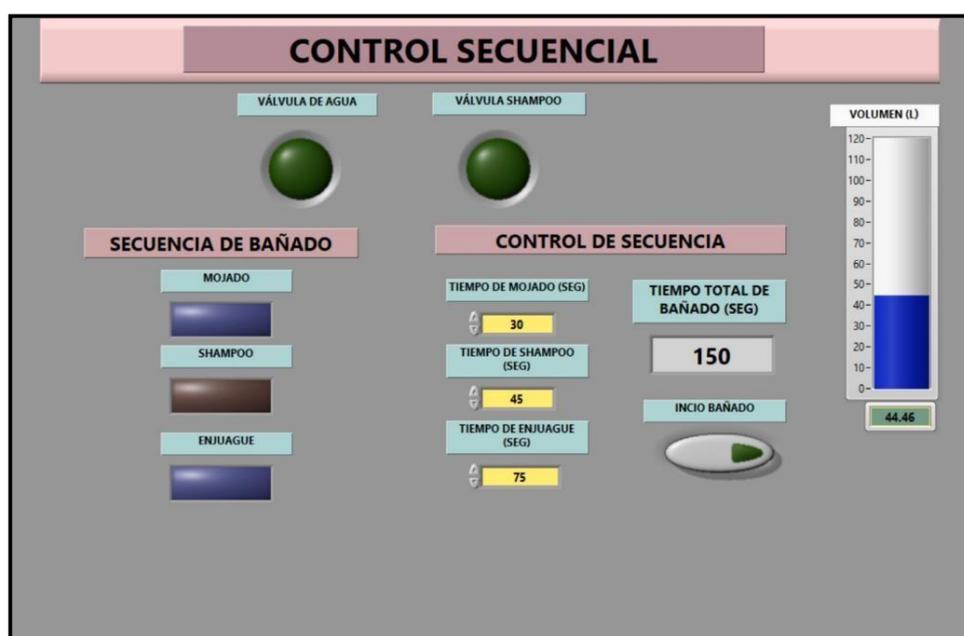


Figura N°78: Respuesta del controlador Secuencial para un T=150 segundos

Fuente: Elaboración Propia usando el software LabVIEW

Finalmente, en la Tabla N° 11 se observan los resultados obtenidos del controlador secuencial donde se muestra la cantidad de agua consumida en función del tiempo programado, así como la cantidad de baños posibles a realizarse. Esta información fue necesaria para las pruebas finales. Además, en la Figura N° 79 se muestra el funcionamiento del controlador secuencial en la secuencia de “Mojado”.

Tabla N°11: Resumen de resultados del controlador secuencial

TIEMPO (S)	VOLUMEN CONSUMIDO (L)	CANTIDAD DE BAÑADO
90	16.33	4
120	28.64	3
150	34.54	2

Fuente: Elaboración Propia



Figura N° 79: Controlador Secuencial- Secuencia “Mojado

Fuente: Elaboración Propia

4.4. Pruebas y resultados finales

Para la prueba final donde se realizó el baño al can de nombre “Ramón”, se consideraron las siguientes condiciones, debido a que se consiguió buenos resultados tanto para el bañado como para la conservación de agua para poder bañar a 2 canes más.

Nivel de tanque: 80L

Temperatura de agua: 37 °C

Para el bañado se configuró un total de 110 segundos de la siguiente manera:

- a) Tiempo de Mojado: 20 Segundos
- b) Tiempo de Shampoo: 40 Segundos
- c) Tiempo de Enjuagado: 50 Segundos

Se puede observar en la Figura N°80 la configuración inicial para el bañado del can “Ramón”.

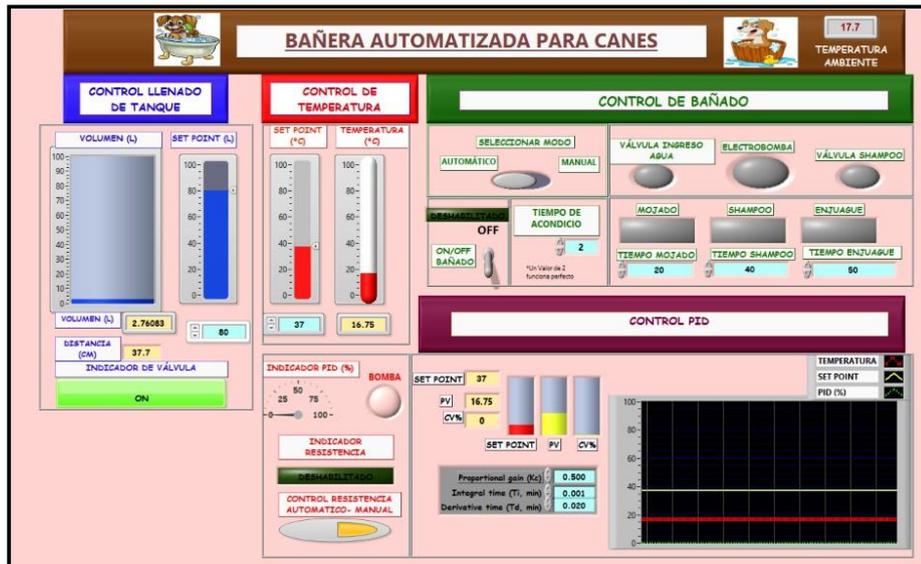


Figura N°80: Configuración inicial para bañado del can “Ramón”

Fuente: Elaboración Propia usando el software LabVIEW

El programa inicia con el llenado del tanque de agua a un Set Point de 80L, el cual tomó 7.20 minutos y se llenó hasta 84.83L. Al finalizar el llenado del tanque de agua, el programa inició con el calentamiento del agua, con un Set Point de 37°C. El tiempo para alcanzar la temperatura deseada fue de 45 minutos. Luego, la temperatura se estabilizó en 37.25°C, tal como se puede observar en la Figura N°81.

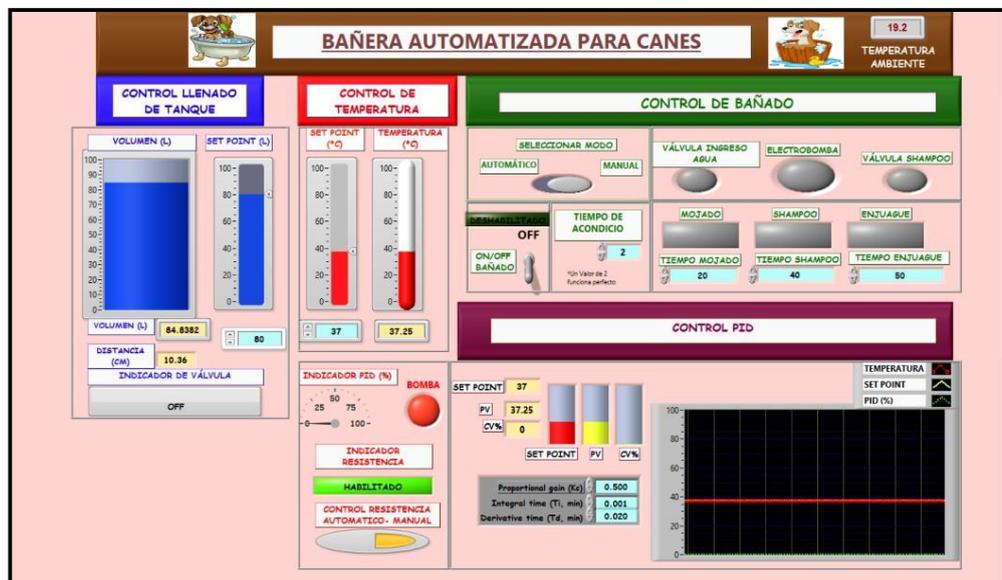


Figura N°81: Control de nivel y temperatura para el bañado del can “Ramón”

Fuente: Elaboración Propia usando el software LabVIEW

En la Figura N°82 se muestra al can “Ramón” antes de ingresar a la bañera e iniciar su proceso de bañado.

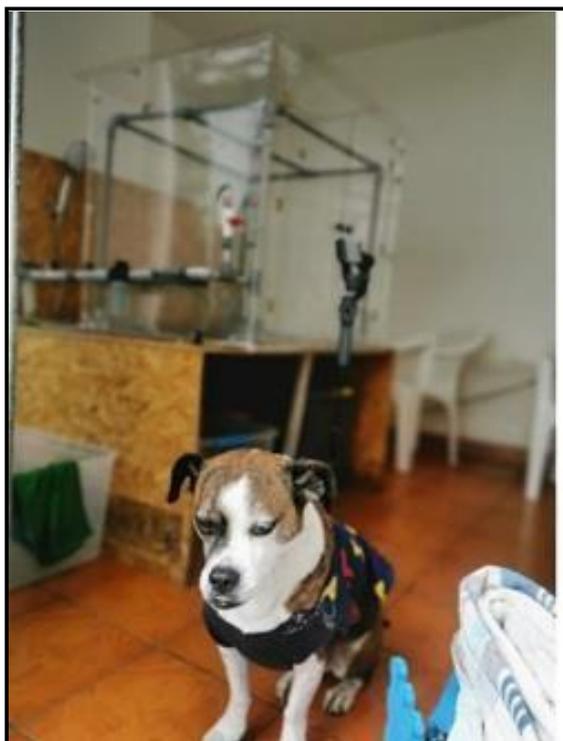


Figura N°82: Can “Ramón” antes de iniciar el bañado

Fuente: Elaboración Propia

Luego, cuando la temperatura del agua llegó al Set Point configurado, el programa indicó al usuario que la bañera se encuentra lista para ingresar al can y dar inicio al bañado. En la Figura N°83 se puede observar el proceso de bañado del can “Ramón”, particularmente en esta parte se encontraba en la secuencia “Mojado”.

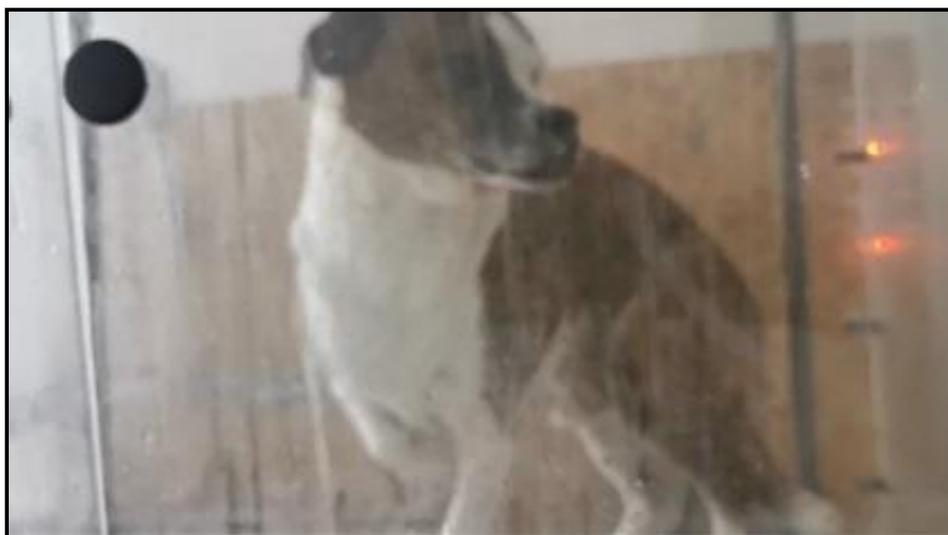


Figura N°83: Can “Ramón” en proceso de bañado

Fuente: Elaboración Propia

Asimismo, en la Figura N° 84 se observa que el bañado concluyó satisfactoriamente.



Figura N°84: Proceso de bañado finalizado

Fuente: Elaboración Propia

4.5. Presupuesto

En la Tabla N° 12 se detalla el costo de los materiales para la implementación del prototipo de bañera automatizada para canes.

Tabla N° 12: Tabla de Presupuesto

N°	Materiales	Un. Medida	Cant.	P. Unt. S/.	P. Total S/.
Electrónica					S/ 845.10
1	Sensor DB18B20	Unid	2	8.50	17.00
2	Sensor HC-SR04	Unid	3	6.50	19.50
3	Arduino Uno	Unid	2	30.00	60.00
4	Válvula solenoide de 1/2"	Unid	3	25.00	75.00
5	Tarjeta dimmer (PBC)	Unid	1	30.00	30.00
6	Tarjeta interfaz (PBC)	Unid	1	20.00	20.00
7	Resistencia sumergible 2.5Kw con termóstato	Unid	1	550.00	550.00
8	Indicadores Led	Unid	5	1.00	5.00
9	Cables jumpers	Juego	2	6.00	12.00
10	Módulo relé x 4	Unid	1	13.60	13.60
11	Módulo relé x 2	Unid	2	6.50	13.00
12	Relé estado sólido	Unid	1	30.00	30.00
13	Motor eléctrico 12v para mezclador	Unid	1	30.00	30.00
Construcción Prototipo					S/1,750.90
1	Tubo de PVC de 1/2"	Unid	5	12.00	60.00
2	Tee de PVC 90°	Unid	20	0.60	12.00
3	Tricodos galvanizados de 1/2"	Unid	8	10.00	80.00
4	Codos de PVC 1/2"	Unid	30	1.00	30.00
5	Adaptadores de PVC 1/2" (Hembra)	Unid	20	0.60	12.00
6	Adaptadores de PVC 1/2" (Macho)	Unid	20	0.60	12.00
7	Adaptador de manguera	Unid	1	6.00	6.00
8	Válvula de paso de 1/2"	Unid	3	25.00	75.00
9	Electrobomba de agua de 1/2 HP	Unid	1	220.00	220.00
10	Accesorios de PVC para conexiones bomba	Unid	1	25.00	25.00
11	Planchas de acrílico	Unid	3	120.00	360.00
12	Plancha de madera OSB	Unid	1	50.00	50.00
13	Reservorio de agua 120 l	Unid	1	60.00	60.00
14	Reservorio de agua 68 l	Unid	1	38.90	38.90
15	Pernos 3/8"	Unid	100	1.00	100.00
16	Tuerca 3/8"	Unid	100	1.00	100.00
17	Válvula check de 1/2"	Unid	1	35.00	35.00
18	Tubo de silicona aislante	Unid	5	9.00	45.00
19	Ángulo 1"	Unid	70	1.00	70.00

20	Regadera tipo ducha	Unid	1	30.00	30.00
21	Dispensador de shampoo	Unid	1	30.00	30.00
22	Bulkhead para resistencia	Unid	1	150.00	150.00
23	Bulkhead para salida de electrobomba	Unid	1	150.00	150.00
Eléctrico					S/ 287.70
1	Llave térmica 2 x 63A	Unid	1	50.00	50.00
2	Llave térmica 2 x 32A	Unid	1	40.00	40.00
3	Llave térmica 2 x 15 A	Unid	1	20.00	20.00
4	Llave térmica 2 X10 A	Unid	1	20.00	20.00
5	Riel para llave	Unid	1	8.60	8.60
6	Cable vulcanizado 14 AWG	m	3	4.40	13.20
7	Cable vulcanizado 16 AWG	m	3	3.30	9.90
8	Cable GPT 18 AWG	m	10	1.50	15.00
9	Caja de paso 20x20x0.80 cm	Unid	3	30.00	90.00
10	Enchufe	Unid	1	5.00	5.00
11	Cinta aislante	Unid	4	4.00	16.00
12	Fuente de alimentación de 220v a 12v	Unid	1	100.00	100.00
Servicio de Ingeniería					S/1,500.00
1	Servicio de instalación y puesta en funcionamiento	Día	10	S/150.00	S/1,500.00
TOTAL				S/	4,383.70

Fuente: Elaboración Propia

CONCLUSIONES

1. Se logró diseñar e implementar una bañera casera para canes con una base de madera tipo OSB, una cabina de planchas de acrílico y una estructura de PVC para el bañado del can. La bañera de canes tuvo una dimensión de 80cm de largo, 60 cm de ancho y 65 cm de alto. Además, este prototipo de bañera contó con un sensor de ultrasonido y un sensor de temperatura; también contó con 3 electroválvulas, una resistencia eléctrica y una electrobomba. Se utilizó un módulo Arduino como interfaz entre los componentes anteriores y la computadora donde fue implementado el controlador en LabVIEW de la bañera. Asimismo, se instalaron 2 tanques de agua donde uno de ellos se usó para realizar los controles de Nivel y Temperatura del agua usada para el bañado del can, y el otro tanque fue usado para recolectar los residuos del agua usada luego del bañado del can.
2. Asimismo, se logró la implementación y sintonización del controlador PID de temperatura utilizando un sensor sumergible de temperatura y una resistencia de inmersión para la calefacción del agua. El controlador PID fue implementado en el software LabVIEW utilizando un módulo Arduino como interfaz entre el sensor, la resistencia y el controlador. Se logró sintonizar el controlador usando el método de prueba y error donde se encontró que, para un buen comportamiento del controlador de temperatura, las ganancias fueron: $K_p= 0.5$, $T_i=0.001$ y $T_d= 0.02$. El controlador implementado logró calentar el agua a un Set Point de 37°C en un tiempo de 45.35 minutos, donde se logró estabilizar a una temperatura de 37.0625°C y con un error de 0.0625°C .
3. Así como también, se realizó la implementación del controlador ON/OFF de nivel utilizando un sensor de ultrasonido para el llenado de un tanque de agua. El controlador ON/OFF fue implementado en el software LabVIEW utilizando un módulo Arduino como interfaz de comunicación entre el sensor y el controlador, con ello se halló una ecuación que convierta los datos enviados del sensor de centímetros a litros. Además, dicho controlador logró llenar el tanque de agua a un Set Point de 80 litros en un tiempo de 7.18 minutos y con un error de 0.420 litros de agua.

4. Finalmente, se logró implementar el control secuencial para el prototipo de bañera de canes utilizando válvulas solenoides como actuadores. El controlador Secuencial fue implementado en el software LabVIEW, utilizando el módulo Arduino como tarjeta interfaz entre el controlador y los actuadores. Con el controlador secuencial implementado, por tiempo, se logró realizar un correcto funcionamiento de cada uno de los procesos “Mojado”, “Ingreso Shampoo” y “Enjuagado”. Igualmente, para la realización de las pruebas se implementó una interfaz amigable para el usuario, donde se unieron los 3 controladores anteriores y se realizó satisfactoriamente el bañado de un can, el cual no demostró incomodidad ni inseguridad.

RECOMENDACIONES

1. Como mejora para la presente investigación se recomienda cambiar el controlador ON/OFF de nivel por un controlador PID de nivel, para tener un control más preciso con el llenado del tanque; para ello, también se recomendaría utilizar un sensor de flujo y una servoválvula. Además, que se puede tener un monitoreo completo del consumo de agua utilizado para todos los procesos de bañado.
2. Para disminuir el tiempo de calefacción del agua en el tanque, se recomienda usar una resistencia sumergible de mayor potencia a la usada en el presente trabajo de investigación.
3. Si se desea implementar este prototipo para uso comercial, se recomienda instalar la etapa de secado. La etapa de secado podrá llevar un control de flujo de aire y temperatura para no incomodar al can mientras es secado.

REFERENCIAS BIBLIOGRÁFICAS

1. Agudelo, J. y Hernández, M. (2015), *Control PID de nivel de agua usando LabVIEW y Arduino*. (Artículo de Investigación). Recuperado de [https://www.academia.edu/32372926/Control_PID_Nivel_de_agua_usando_Lab View_y_Arduino_Control_de_procesos_asistidos_por_PC](https://www.academia.edu/32372926/Control_PID_Nivel_de_agua_usando_Lab_View_y_Arduino_Control_de_procesos_asistidos_por_PC).
2. Arduino (2021). *Arduino UNO*. Visitado en enero del 2021. Recuperado de: <https://store.arduino.cc/usa/arduino-uno-rev3>
3. Arteaga, D. y Tobar, E (2014). *Diseño e implementación de módulos de entrenamiento con aplicaciones de control PID de un caldero de una cámara de estabilidad utilizando la interface NI MYDAQ y el software LabVIEW*. [Tesis de Grado, Universidad Politécnica Salesiana]. Repositorio institucional de la Universidad Politécnica Salesiana
<https://dspace.ups.edu.ec/bitstream/123456789/7385/1/UPS-GT000767.pdf>
4. Ayala, L. y Donoso, R. (2015), *Diseño, modelación, simulación e implementación de un sistema de control de temperatura para la ducha eléctrica*. [Tesis de Grado, Escuela Superior Politécnica de Chimborazo]. Repositorio institucional de la Escuela Superior Politécnica de Chimborazo
<http://dspace.esoch.edu.ec/handle/123456789/4499>
5. Bohórquez, K., Fonseca, D. y Gutiérrez, S. (2017). *Sistema didáctico para el control de nivel con tanques acoplados* [Tesis de Grado, Universidad Católica de Colombia]. Repositorio institucional de la Universidad Católica de Colombia
<https://repository.ucatolica.edu.co/bitstream/10983/15297/1/Documento%20Tanques%20Acoplados%2012-12-2017%20Versi%C3%B3n%20Final.pdf>
6. Carhuaz, C. y Castillo, R. (2018). *Determinación de los parámetros de sintonía PID para controlar el nivel en la planta del laboratorio de operaciones y procesos unitarios-FIQ*. [Tesis de Grado, Universidad Nacional del Centro del Perú]. Repositorio institucional de la Universidad Nacional del Centro del Perú
<http://repositorio.uncp.edu.pe/bitstream/handle/20.500.12894/3774/Carhuaz%20Lozano%20Su%20Su%20Castillo.pdf?sequence=1&isAllowed=y>
7. Delgado, C. y Quinte, J. (2018). *Implementación de un módulo de control de razón con un controlador PI para controlar la proporción de mezclado de dos*

- flujos*. [Tesis de Grado, Universidad Ricardo Palma]. Repositorio institucional de la Universidad Ricardo Palma <https://repositorio.urp.edu.pe/bitstream/handle/URP/2716/CDELGADO%26JQUINTE.pdf?sequence=1&isAllowed=y>
8. Fuentes, Y. (2017). *Diseño e Implementación de un módulo educativo para el control de temperatura*. [Tesis de Grado, Pontificia Universidad Católica del Perú]. Repositorio institucional de la Pontificia Universidad Católica del Perú <https://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/8629>
 9. Huayta, J. y Sueña, E. (2017). *Diseño e Implementación de un sistema SCADA para el control de nivel de agua para uso domótico mediante redes industriales*. [Tesis de Grado, Universidad Nacional del Altiplano Puno]. Recuperado de Repositorio institucional de la Universidad Nacional del Altiplano Puno http://repositorio.unap.edu.pe/bitstream/handle/UNAP/4185/Huayta_Sucasaca_Jose_Carlos_Sua%c3%b1a_Humpire_Elmer_Wilson.pdf?sequence=1&isAllowed=y
 10. Hurtado, I. (2020). *Implementación de un prototipo para un Sistema de adquisición de datos de temperatura de nivel de aguas en tiempo real de una planta de tratamiento de aguas residuales*. [Tesis de Grado, Universidad de las Américas- Campus Granados]. Repositorio institucional de la Universidad de las Américas-Campus Granados <http://dspace.udla.edu.ec/handle/33000/12734>
 11. Instrumentación y Control (2021). ¿Qué valores iniciales usar en un PID? Recuperado el 15 de junio del 2021, de <https://instrumentacionycontrol.net/que-valores-iniciales-usar-en-un-pid/>
 12. Martínez, J. (2018). *Desarrollo de un sistema de monitorización de temperaturas en tiempo real para intercambiador de calor de doble tubo con sondas sumergibles de temperatura DS18B20 usando el microcontrolador Arduino*. [Tesis de Grado, Universidad Politécnica de Cartagena]. Repositorio institucional de la Universidad Politécnica de Cartagena <https://repositorio.upct.es/bitstream/handle/10317/6932/tfg-martinez.pdf?sequence=1&isAllowed=y>.
 13. Matías, L. y Ruiz, A. (2014). *Diseño e Implementación de un sistema Programable para la Regulación de la Temperatura en una Regadera*. [Tesis de

- Grado, Instituto Politécnico Nacional]. Repositorio institucional del Instituto Politécnico Nacional
<https://tesis.ipn.mx/bitstream/handle/123456789/15605/I.C.E.%2047-14.pdf?sequence=1&isAllowed=y>.
14. Naylampmechatronics (2021). Sensor de ultrasonido HC-S04. Recuperado el 15 de enero del 2021, de <https://www.naylampmechatronics.com/sensores-proximidad/10-sensor-ultrasonido-hc-sr04.html>
 15. Del Valle, L. (2021)). Sensor de temperatura DS18B20. Programa Fácil. <https://programarfacil.com/blog/arduino-blog/ds18b20-sensor-temperatura-arduino/>
 16. Quiñones, A. (2020). *Diseño de un calentador eléctrico con mecanismo de encendido automático por flujo de agua*. [Tesis de Grado, Universidad Nacional del Altiplano de Puno]. Repositorio institucional de la Universidad Nacional del Altiplano de Puno
http://repositorio.unap.edu.pe/bitstream/handle/UNAP/14483/Qui%c3%b1onez_Terrazas_Angel_Enrique.pdf?sequence=1&isAllowed=y.
 17. Sierra, L. (2017). *Diseño del sistema de medición y despliegue de temperaturas con sensor DS18B20 mediante el protocolo de transmisión 1 Wire*. [Tesis de Grado, Universidad de San Carlos de Guatemala]. Repositorio institucional de la Universidad de San Carlos de Guatemala
<http://www.repositorio.usac.edu.gt/7963/1/Luis%20Antonio%20Sierra%20Garc%C3%ADa.pdf>
 18. Vergara, B (2012). *Realización de un sistema SCADA para el control de imagen utilizando IMAQ de LabVIEW y cámara de objetos para el laboratorio de mecatrónica*. [Tesis de Grado, Escuela Superior Politécnica de Chimborazo]. Repositorio institucional de la Escuela Superior Politécnica de Chimborazo
<http://dspace.epoch.edu.ec/bitstream/123456789/3069/1/25T00203.pdf>
 19. Vizcaíno, D. (2017). *Diseño e Implementación de un Sistema de Calentamiento y Ahorro de Agua en Domicilios* [Tesis de Grado, Universidad San Francisco de Quito]. Repositorio institucional de la Universidad San Francisco de Quito
<https://repositorio.usfq.edu.ec/bitstream/23000/6495/1/131232.pdf>

ANEXOS

Anexo 1: Programación en Arduino

```
PROYECTO_FINAL §
//LIBRERIAS
#include <pt.h>
static struct pt pt1; //Para la función Ultrasonido-Distancia
static struct pt pt2;
static struct pt pt3; //PARA TEMPERATURA AMBIENTE
#include "DHT.h" //cargamos la libreria DHT
#define DHTTYPE DHT22 //Se selecciona el DHT22(hay otros DHT)
#include <OneWire.h>
#include <DallasTemperature.h>
//ULTRASONIDO
#define Trigger 2
#define Echo 3
//DS16B20
#define ONE_WIRE_BUS 4
//DHT22
#define DHTPIN 5 //Seleccionamos el pin en el que se conectará el sensor
//SALIDAS
#define BOMBA 6
#define ElectroBomba 7
#define Valvula 8
#define Resistencia 9 //PWM
#define ValvulaAgua 10
#define ValvulaShampoo 11
#define LEDResistencia 12
//.....
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensorDS18B20(oneWire);
DHT dht(DHTPIN, DHTTYPE); //Se inicia una variable que será usada por Arduino para comunicarse con el sensor
//.....
//.....
void setup()
{
  Serial.begin(9600);
  sensorDS18B20.begin();
  dht.begin();
  PT_INIT(spt1);
  PT_INIT(spt2);
  PT_INIT(spt3);
}
void loop()
{
  protothreadDISTANCIA_TEMPERATURA(spt1); //Función Distancia-Ultrasonido
  protothreadVALVULAREsistencia(spt2);
  dht(spt3); //Funcion temperatura ambiente
}
//.....
//.....

//-----Distancia-Ultrasonido-Temperatura-----

static int protothreadDISTANCIA_TEMPERATURA(struct pt 'pt)
{
  static unsigned long Time=0;
  PT_BEGIN(pt);

  //void setup ()
  pinMode(Echo, INPUT);
  pinMode(Trigger, OUTPUT);
  digitalWrite(Trigger, LOW);
  //
}
```

PROYECTO_FINAL §

```

//
while(1)
{
    Time= millis();
    PT_WAIT_UNTIL(pt, millis()-Time>100); //Tiempo de espera para enviar datos

    //*****Programación sensor de ULTRASONIDO*****
    double t; //timepo que demora en llegar el eco
    double d; //distancia en centimetros
    digitalWrite(Triquer, HIGH);
    delayMicroseconds(10);
    digitalWrite(Triquer,LOW);
    t = pulseIn(Echo, HIGH); //obtenemos el ancho del pulso
    d = t/59.0;//escalamos el tiempo a una distancia en cm
    String Sensor1=String(d);
    String DISTANCIA=String("A"+Sensor1);
    if(d>255||d<=0){d=255;}

    //*****Programación sensor de TEMPERATURA*****
    sensorDS18B20.requestTemperatures();
    String Sensor2 = String(sensorDS18B20.getTempCByIndex(0),DEC);
    String TEMPERATURA=String("B"+Sensor2);

    //*****ENVIO DE DATOS A LABVIEW*****
    Serial.println(DISTANCIA);
    Serial.println(TEMPERATURA);

}
PT_END(pt);
}

//-----Activar VÁLVULA-RESISTENCIA-----

static int protothreadVALVULaresistencia(struct pt *pt)
{
    static unsigned long Time=0;
    PT_BEGIN(pt);

    //void setup ()
    pinMode(ValvulaAgua,OUTPUT);
    digitalWrite(ValvulaAgua,HIGH);
    pinMode(ValvulaShampoo,OUTPUT);
    digitalWrite(ValvulaShampoo,HIGH);
    pinMode(ElectroBomba,OUTPUT);
    digitalWrite(ElectroBomba,HIGH);
    pinMode(BOMBA,OUTPUT);
    digitalWrite(BOMBA,HIGH);

    pinMode(Valvula,OUTPUT);
    digitalWrite(Valvula,HIGH);
    pinMode(Resistencia,OUTPUT);
    digitalWrite(Resistencia,LOW);

    //
    while(1)
    {
        Time= millis();
        PT_WAIT_UNTIL(pt, millis()-Time>10); //Tiempo lectura de datos
    }
}

```

PROYECTO_FINAL §

```

if(Serial.available()>0)
{
    char comando = Serial.read();
    char Dato = Serial.read();
    int Dato2=map(Dato,0,100,0,255);

    if(comando=='L')
        {analogWrite(Resistencia,Dato2);} //ON RESISTENCIA
    else if(comando=='f')
        {digitalWrite(Valvula,LOW);
        analogWrite(Resistencia,0);} //ON VALVULA
    else if(comando=='g')
        {digitalWrite(Valvula,HIGH);} //OFF VALVULA
    else if(comando=='n')//INDICADOR TANQUE
        {digitalWrite(ValvulaAgua,HIGH);
        digitalWrite(ElectroBomba,HIGH);
        digitalWrite(ValvulaShampoo,HIGH);}
    else if(comando=='h')
        {digitalWrite(ValvulaAgua,LOW);
        digitalWrite(ElectroBomba,LOW);
        digitalWrite(Valvula,HIGH);
        analogWrite(Resistencia,0);} //ON VALVULA AGUA - ON MOTOR
    else if(comando=='i')
        {digitalWrite(ValvulaShampoo,LOW);
        digitalWrite(ValvulaAgua,HIGH);
        digitalWrite(ElectroBomba,LOW);
        digitalWrite(Valvula,HIGH);
        analogWrite(Resistencia,0);} //ON VALVULA SHAMPOO - OFF VALVULA AGUA/MOTOR
    else if(comando=='j')
        {digitalWrite(ValvulaAgua,LOW);
        digitalWrite(ElectroBomba,LOW);
        digitalWrite(ValvulaShampoo,HIGH);
        digitalWrite(Valvula,HIGH);
        analogWrite(Resistencia,0);} //ON VALVULA AGUA/MOTOR - OFF VALVULA SHAMPOO
    else if(comando=='k')
        {digitalWrite(ValvulaAgua,HIGH);
        digitalWrite(ElectroBomba,HIGH);
        digitalWrite(ValvulaShampoo,HIGH);
        digitalWrite(Valvula,HIGH);
        analogWrite(Resistencia,0);} //OFF VALVULA AGUA Y SHAMPOO/MOTOR
    else if(comando=='m')
        {digitalWrite(ValvulaAgua,HIGH);
        digitalWrite(ElectroBomba,HIGH);
        digitalWrite(ValvulaShampoo,HIGH);
        digitalWrite(Valvula,HIGH);
        analogWrite(Resistencia,0);} //OFF VALVULA AGUA Y SHAMPOO/MOTOR
    else if (comando == 'o')
        {digitalWrite(BOMBA,LOW);}
    else if (comando == 'p')
        {digitalWrite(BOMBA,HIGH);}
    }
}
PT_END(pt);
}
static int dht (struct pt *pt){
    static unsigned long Tiem=0;
    PT_BEGIN(pt);
    while (1){
        Tiem=millis();
        PT_WAIT_UNTIL(pt,millis()-Tiem>100);
        float t = dht.readTemperature(); //Se lee la temperatura
        String temp=String(t);
        String TEMPERAT=String("C"+temp);
        Serial.println(TEMPERAT); //Envio de datos hacia LabVIEW
        PT_END(pt);
    }
}

```

Anexo 2: Programación LabVIEW Final

