



ESPE

**UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA**

UNIDAD DE GESTIÓN DE  TECNOLOGÍAS

**IMPLEMENTACIÓN DE UN CONTROL AUTOMÁTICO DE
NIVEL MEDIANTE INDUSTRIINO PARA PRÁCTICAS DE
LABORATORIO EN EL MÓDULO DIDÁCTICO PCT 3.**

AUTOR:

JIMÉNEZ GUAMÁN, CARLOS STEVEN

**DEPARTAMENTO DE TECNOLOGÍA ELÉCTRICA Y
ELECTRÓNICA**

**CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN
& AVIÓNICA**

**MONOGRAFÍA PREVIO A LA OBTENCIÓN DEL TÍTULO DE
TECNÓLOGO EN ELECTRÓNICA MENCIÓN
INSTRUMENTACIÓN & AVIÓNICA**

TUTOR(A):

ING. CAJAS BUENAÑO, MILDRED LISSETH

LATACUNGA

07-09-2020



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN Y AVIÓNICA

CERTIFICACIÓN

Certifico que la monografía, “**IMPLEMENTACIÓN DE UN CONTROL AUTOMÁTICO DE NIVEL MEDIANTE INDUSTRIUINO PARA PRÁCTICAS DE LABORATORIO EN EL MÓDULO DIDÁCTICO PCT 3**” fue realizado por el/los señor/señores señor **JIMÉNEZ GUAMÁN, CARLOS STEVEN** la cual ha sido revisada y analizada en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 7 de septiembre de 2020

Firma:

Ing. Cajas Buenaño, Mildred Lisseth

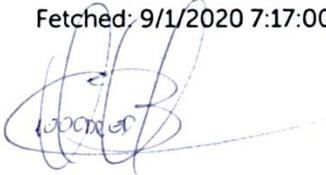
C. C.: 050349760-4

Document Information

Analyzed document	TESIS_CAPÍTULO II_III_IV.pdf (D78463053)
Subn	
Submitted by	
Submitter email	csjimnezg@espe.edu.ec
Similarity	7%
Analysis address	mlcajas.espe@analysis.orkund.com

Sources included in the report

W	URL. http://lcr.uns.edu.ar/fcr/images/Introduccion%20a%20Los%20Sistemas%20de%20Control.pdf Fetched: 9/1/2020 7:17:00 PM		2
W	URL. http://www.alegsa.com.ar/Dic/sistema_de_control.php Fetched: 9/1/2020 7:17:00 PM		1
SA	Universidad de las Fuerzas Armadas ESPE / CAPITULOS I_II_III_IV MMAX.docx Document CAPITULOS I_II_III_IV MMAX.docx (D25559530) Submitted by: alexmallitasig1993@gmail.com Receiver: zaproano.espe@analysis.orkund.com		2
W	URL. https://industriino.com/ Fetched: 9/1/2020 7:17:00 PM		1
W	URL. https://docplayer.es/40593009-Universidad-politecnica-salesiana-sede-quito-faculta ... Fetched: 7/23/2020 4:27:12 AM		1
W	URL. https://instrumentacionuc.wixsite.com/facultad-ingenieria/copia-de-2 Fetched: 9/1/2020 7:17:00 PM		7
SA	Universidad de las Fuerzas Armadas ESPE / tesis 6 de julio del 2016.docx Document tesis 6 de julio del 2016.docx (D21201180) Submitted by: wesanchez@espe.edu.ec Receiver: wesanchez.espe@analysis.orkund.com		3
W	URL. http://vam-comunicaciones-ferroviarias.blogspot.com/2012/12/que-es-carril-din-what ... Fetched: 9/1/2020 7:17:00 PM		1
W	URL. https://concepto.de/fuente-de-alimentacion/ Fetched: 9/1/2020 7:17:00 PM		1



Ing. Cajas Buenaño, Mildred Lisseth
C. C.: 050349760-4
DIRECTOR(A)

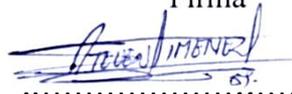


DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN Y
AVIÓNICA

RESPONSABILIDAD DE AUTORÍA

Yo, **JIMÉNEZ GUAMÁN, CARLOS STEVEN**, con cédula de ciudadanía n°172532248-9, declaro que el contenido, ideas y criterios de la monografía: **“IMPLEMENTACIÓN DE UN CONTROL AUTOMÁTICO DE NIVEL MEDIANTE INDUSTRIINO PARA PRÁCTICAS DE LABORATORIO EN EL MÓDULO DIDÁCTICO PCT 3”** es de mi/nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 7 de septiembre de 2020

Firma

.....

Jiménez Guamán, Carlos Steven

C.C.: 172532248-9



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN
Y AVIÓNICA

AUTORIZACIÓN DE PUBLICACIÓN

Yo, **JIMÉNEZ GUAMÁN, CARLOS STEVEN** autorizo a la Universidad de las Fuerzas Armadas ESPE publicar la monografía: “**IMPLEMENTACIÓN DE UN CONTROL AUTOMÁTICO DE NIVEL MEDIANTE INDUSTRIUINO PARA PRÁCTICAS DE LABORATORIO EN EL MÓDULO DIDÁCTICO PCT 3**” en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 7 de septiembre de 2020

Firma

.....

Jiménez Guamán, Carlos Steven

C.C.: 172532248-9

DEDICATORIA

El presente proyecto está dedicado a una gran compañera, quien fue mi mentora académica e influyó en gran manera en mi vida estudiantil y hacia toda la comunidad universitaria cuyos anhelos aún están por cumplirse, el camino siempre es complejo pero cada día es un paso menos para lograr los objetivos y metas propuestos.

El ser humano siempre en su afán por mejorar cada día busca nuevas maneras de realizar procesos de manera más eficiente y yo, al igual que muchos en este camino, dedico este logro a cada persona que estuvo apoyándome tanto en las aulas como fuera de ellas, además de manera especial a mi familia y amigos que me ayudaron a mantener mi estabilidad emocional.

Carlos Steven Jiménez Guamán

AGRADECIMIENTO

Quiero agradecer a Dios por darme la oportunidad de seguir creciendo en el ámbito profesional, a los ingenieros con los cuales tuve la oportunidad de compartir un momento tanto en las aulas como fuera de ellas, a mi familia la cual puso su confianza en mí y estuvo apoyándome día a día.

Sería letanía describir a todas las personas que me apoyaron en mi vida estudiantil y personal a las cuales agradezco en gran manera, gracias por permitirme ser parte de un instante de su vida.

Carlos Steven Jiménez Guamán

ÍNDICE DE CONTENIDOS	
CARÁTULA	1
CERTIFICACIÓN	2
ANÁLISIS URKUND	3
RESPONSABILIDAD DE AUTORÍA	4
AUTORIZACIÓN DE PUBLICACIÓN	5
DEDICATORIA	6
AGRADECIMIENTO	7
ÍNDICE DE CONTENIDOS	8
ÍNDICE DE TABLAS	11
ÍNDICE DE FIGURAS	12
RESUMEN	14
ABSTRACT	15
CAPÍTULO I	
PLANTEAMIENTO DEL PROBLEMA	16
1.1 Tema	16
1.2 Antecedentes	16
1.3 Planteamiento del problema	17
1.4 Justificación	18
1.5 Objetivos	19
1.6 Alcance	19

CAPÍTULO II

MARCO TEÓRICO	20
2.1 Control automático	20
2.2 Sistema de control	21
2.3 Tipos de sistemas de control	21
2.4 Control Fuzzy	23
2.5 Técnica de Lógica difusa	24
2.6 Industruino	25
2.7 Arduino Nano	26
2.8 Medidores de nivel de líquidos	27
2.9 Rotámetro	31
2.10 Carril DIN	35
2.11 Electroválvula	35
2.12 Fuentes de alimentación	36

CAPÍTULO III

DESARROLLO DEL TEMA	40
3.1 Características de los elementos	41
3.1.1 Módulo Arduino IND.	41
3.1.2 Arduino Nano	43
3.1.3 Sensor TEACH UB500- 18GM75-I-V15	43
3.1.4 Electro-válvula	44

	10
3.1.5 Bomba JABSCO	45
3.2 Características del proceso	45
3.2.1 P&ID del módulo	45
3.2.2 Control Fuzzy	47
3.2.3 Estructura de programación	52
3.3 Proceso de automatización del módulo de nivel PCT-3	54
3.4 Diseño del panel frontal y Ensamblaje de panel frontal	56
3.5 Transmisor Arduino	58
3.6 Programación en la plataforma de arduino Ind.	60
3.7 Montaje final de módulo de nivel PCT-3	106
CAPÍTULO IV	
CONCLUSIONES Y RECOMENDACIONES	108
REFERENCIAS BIBLIOGRÁFICAS	109
ANEXOS	111

ÍNDICE DE TABLAS

Tabla 1 Industruino- Zona Analógica	27
Tabla 2 Industruino -Zona Digital	27
Tabla 3 Industruino -Zona de Micro-Controlador	28
Tabla 4 Características de Arduino Nano	28
Tabla 5 Características de sensor teach	29
Tabla 6 Elementos de Indicación.	29
Tabla 7 Características de Servo-Motor (electro-válvula)	30
Tabla 8 Características de Bomba	30
Tabla 9 Nomenclatura de P&ID	31
Tabla 10 Reglas difusas	35

ÍNDICE DE FIGURAS

Figura. 1 Sistema de control Automático	20
Figura. 2 Tipos de entrdas	21
Figura. 3 Control de lazo abierto	22
Figura. 4 Etapa para toma de decisiones	23
Figura. 5 Módulo Industruino	25
Figura. 6 Zonas principales de Indutruino	26
Figura. 7 Arduino Nano	27
Figura. 8 Mirilla de vidrio	28
Figura. 9 Sensor ultrasónico Teach	31
Figura. 10 Partes de un rotámetro	31
Figura. 11 Carril Din	35
Figura. 12 Electro-válvula	36
Figura. 13 Fuente de alimentación lineal	37
Figura. 14 Fuente de alimentación conmutada	38
Figura. 15 Diagrama de conexiones Indutruino	40
Figura. 16 P&ID de módulo PCT 3	46
Figura. 17 Fusificación de set-point	47
Figura. 18 Fusificación del sensor ultrasónico Teach	48
Figura. 19 Fusificación de variable “error”	49
Figura. 20 Inferencias difusas	51
Figura. 21 Desfusificación para control de actuadores	51
Figura. 22 Estructura de programación	53
Figura. 23 Desmontaje del módulo PCT 3	54
Figura. 24 Montaje de los instrumentos de medición	55

Figura. 25 Electro-válvula didáctica (servo-válvula)	56
Figura. 26 Diseño de panel frontal	57
Figura. 27 Panel frontal ensamblado (vista interna)	58
Figura. 28 Panel frontal ensamblado (vista externa)	58
Figura. 29 Diseño de transmisor Arduino	59
Figura. 30 Entorno Arduino	61
Figura. 31 Gestor de librerías en Arduino	61
Figura. 32 Entorno de Arduino	62
Figura. 33 Módulo completo con panel abierto	106
Figura. 34 Módulo completo con panel cerrado	107

RESUMEN

En el laboratorio de Instrumentación Virtual de la Unidad de Gestión de Tecnologías existe una cantidad limitada de módulos didácticos de control de nivel, los cuales tienen un sistema de control mediante PLC'S, al momento no se cuenta con un sistema Industruino para que los estudiantes puedan realizar sus prácticas. Es así que se desarrolló el trabajo de titulación "Implementación de un control automático de nivel mediante Industruino para prácticas de laboratorio en el módulo didáctico PCT 3", con el propósito de mejorar el desempeño teórico – práctico de los estudiantes de la carrera de tecnología en Electrónica, Mención Instrumentación y Aviónica. Se desarrolló actuadores para el módulo PCT 3, como electro-válvulas didácticas, un panel frontal, con el cual se facilita la recopilación y verificación de datos de manera didáctica. Las electro-válvulas, una a la salida y una a la entrada del tanque principal, controlan el nivel de líquido manteniendo un flujo. Comparado con el control tradicional, el control difuso tiene dos ventajas prácticas, una es que el modelo matemático del proceso a controlar no es requerido y otra es que se obtiene un controlador no lineal desarrollado empíricamente sin complicaciones matemáticas, en realidad los desarrollos matemáticos de este tema todavía están en sus inicios. Se utiliza el conocimiento y experiencia de un operador humano para construir un controlador que emule el comportamiento de tal persona, en este caso se emula el comportamiento de un estudiante.

PALABRAS CLAVE:

- **MÓDULO INDUSTRIINO**
- **CONTROL DE NIVEL**
- **ELECTRO-VÁLVULAS**
- **CONTROL DIFUSO**

ABSTRACT

In the Virtual Instrumentation laboratory of the Technology Management Unit there are a limited number of didactic modules for level control, which have a control system by means of PLC's. At the moment there is no industrial system for students to carry out their practices. Therefore, the degree work "Implementation of an automatic level control by means of Industruino for laboratory practices in the didactic module PCT 3" was developed, with the purpose of improving the theoretical and practical performance of the students of the Electronics, Instrumentation and Avionics Technology career. Actuators were developed for the PCT 3 module, as didactic electro-valves, a front panel, with which the collection and verification of data is facilitated in a didactic way. The electro-valves, one at the output and one at the input of the main tank, control the liquid level while maintaining a flow. Compared to traditional control, diffuse control has two practical advantages, one is that the mathematical model of the process to be controlled is not required and the other is that a non-linear controller empirically developed without mathematical complications is obtained, in fact the mathematical developments of this topic are still in their infancy. It uses the knowledge and experience of a human operator to build a controller that emulates the behavior of such a person, in this case emulates the behavior of a student.

KEYWORDS:

- **INDUSTRUINO MODULE**
- **CONTROL THE LIQUID LEVEL**
- **ELECTRO-VALVULAS**
- **DIFFUSE CONTROL**

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

1.1. Tema

"IMPLEMENTACIÓN DE UN CONTROL AUTOMÁTICO DE NIVEL MEDIANTE INDUSTRIUINO PARA PRÁCTICAS DE LABORATORIO EN EL MÓDULO DIDÁCTICO PCT 3."

1.2. Antecedentes

Los controladores lógicos programables se basan en un microprocesador que usa instrucciones guardadas en la memoria programable para implementar lógica secuencial, temporización, conteo y funciones aritméticas a través de entradas y salidas tanto digitales como analógicas para controlar máquinas y procesos.

Según (Gómez, 2005) con anterioridad a los PLC's, se utilizaban controladores basados en relés electromecánicos, contadores, temporizadores, etc., hacia 1960 Richard Morley desarrolla una compañía que sustituye el control relé por microcomputadores y en 1968 se formalizan las primeras especificaciones del primer PLC con el objetivo de superar las limitaciones operacionales como: Procesamiento en tiempo real, composición modular y programación basada en lenguaje LADDER.

En el trabajo perteneciente a los autores (Xi CHEN, Yanbo Che, KWE Cheng, 2009) se describe que los controladores lógicos programables PLCs han sido ampliamente utilizados en los campos de control industrial debido a sus méritos como: estructura simple, funciones flexibles, alta confiabilidad y fuerte habilidad en la comunicación.

En la actualidad para que un sistema sea automático debe tener incorporado un PLC o un Relé Lógico Programable el cual se encarga de realizar el proceso, pero estos tienen un costo elevado.

Arduino es una plataforma electrónica de código abierto basada en hardware y software fácil de usar. Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirlo en una salida: activar un motor,

encender un LED. Para hacerlo, utiliza el lenguaje de programación Arduino (basado en el cableado) y el software Arduino (IDE), basado en el procesamiento. Con los años, Arduino ha sido el cerebro de miles de proyectos, desde objetos cotidianos hasta instrumentos científicos complejos. Una comunidad mundial de creadores (estudiantes, aficionados, artistas, programadores y profesionales) se ha reunido en torno a esta plataforma de código abierto, sus contribuciones se han sumado a una increíble cantidad de conocimiento accesible que puede ser de gran ayuda para principiantes y expertos por igual. (Arduino, 2005).

Industruino es una nueva plataforma a nivel industrial con bajos costos de adquisición la cual se está abriendo paso entre las marcas más conocidas de Relés Lógicos Programables y PLC's, uno de los trabajos pioneros en el Ecuador con esa plataforma es el trabajo de titulación "DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA NEUTRALIZADOR DEL PH DEL AGUA BASADO EN INDUSTRIUINO PARA LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE EN GUAYAQUIL CARRERA DE INGENIERÍA ELECTRÓNICA" desarrollado por (Cajamarca Velasco & Velasco Herrera, 2018), en el cual se describe el proceso de automatización para la neutralización del ph de un tanque de agua mediante sensores, actuadores, etc., basado en la plataforma de Industruino como controlador del proceso.

Se propone realizar la "IMPLEMENTACIÓN DE UN CONTROL AUTOMÁTICO DE NIVEL MEDIANTE INDUSTRIUINO PARA PRÁCTICAS DE LABORATORIO EN EL MÓDULO DIDÁCTICO PCT 3", para lo cual se realizará una readecuación de la estación de nivel implementando una electroválvula didáctica más, un rotámetro, un carril Din, un módulo Ind. I/O, las conexiones tanto eléctricas como conducciones de agua necesarias además de un tablero de control con su respectiva señalización.

1.3. Planteamiento del problema

En la actualidad el laboratorio de Instrumentación Virtual de la Unidad de Gestión de Tecnologías de la Universidad de las Fuerzas Armadas ESPE, está dotado de varios equipos entre ellos, sensores, estación de caudal, módulo electro neumático, PLC S7-1200, Relé lógico Programable etc., pero este no cuenta con un módulo Industruino para la ejecución de prácticas por lo cual, se pretende mejorar la tecnología mediante la

implementación de un dispositivo actualizado como es la placa base Ind I/O de nivel industrial.

La falta de implementación de esta tecnología causará que los alumnos de la carrera Electrónica Mención Instrumentación y Aviónica tengan dificultades para acoplar nuevas soluciones tecnológicas en su vida profesional, ya que no tienen conocimiento acerca de este nuevo dispositivo pues esta nueva plataforma va desarrollándose a medida que el alumno aún está en su formación académica y al momento de desarrollarse en el campo profesional no conocerán qué es o cómo se usa esta nueva plataforma.

Por tanto, para preparar profesionales capaces de manejar nuevas tecnologías que permitan mantener los equipos operativos dentro de un proceso industrial a la par con la tecnología que se desarrolla, es necesario que conozcan las nuevas tecnologías desde su formación en las aulas, con el objetivo de mejorar su aprendizaje y desenvolverse de mejor manera dentro del campo laboral.

1.4. Justificación

El manejo de equipos actualizados es de vital importancia para preparar profesionales calificados capaces de realizar diferentes actividades que puedan enriquecer su conocimiento en su vida laboral, ya que las nuevas tecnologías han permitido dar soluciones eficaces en el campo industrial.

El desarrollo de este proyecto permitirá que los alumnos conozcan una nueva plataforma de automatización, de qué se trata, cómo se usa, las diferentes funciones que posee y de esta manera se familiaricen con la tecnología actual realizando prácticas las cuales permitirá mejorar su nivel de estudio y ser profesionales capaces de solucionar los distintos problemas existentes en la industria.

Los principales beneficiarios serán directamente los estudiantes de la carrera de Electrónica Mención Instrumentación y Aviónica, ya que se fortalecerá los conocimientos teóricos y prácticos adquiridos en las aulas, permitiendo así alcanzar un nivel de aprendizaje alto, en el ámbito profesional, con el propósito de formar

profesionales con mano de obra calificada y cualificada para así aportar al desarrollo del país.

1.5.Objetivos

1.5.1. Objetivo general

Implementar un control automático de nivel mediante Industruino para prácticas de laboratorio en el módulo didáctico PCT 3.

1.5.2. Objetivos específicos

- Investigar las características del módulo Ind I/O de nivel industrial mediante las hojas técnicas para aplicarlas en un control de nivel.
- Determinar los parámetros que se van a configurar en el módulo mediante hojas técnicas para implementar el control automático de la estación de nivel.
- Diseñar e implementar dos electro-válvulas para el control de nivel de los tanques de la estación.
- Realizar pruebas de funcionamiento para desarrollar un manual de usuario de la correcta operación de la estación.

1.6. Alcance

Este proyecto tiene como finalidad realizar la implementación de un control automático en la estación de nivel. Para su control se utilizará un módulo Ind I/O 1286 o más actual.

Los dispositivos adquiridos serán donados al Laboratorio de Instrumentación Virtual en la Unidad de gestión de Tecnologías de la Universidad de las Fuerzas Armadas para fines educativos los cuales ayudarán a fomentar el conocimiento en cada uno de los estudiantes y docentes.

CAPÍTULO II

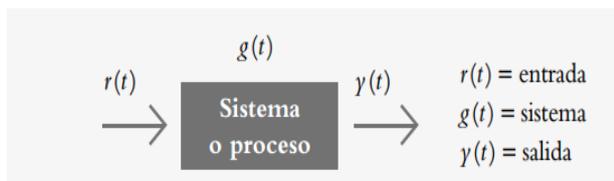
MARCO TEÓRICO

2.1. Control automático

Según (Hernández, 2017), un sistema de control automático es una interconexión de elementos que forman una configuración denominada sistema, de tal manera que el arreglo resultante es capaz de controlar se por sí mismo. Un sistema o componente del sistema susceptible de ser controlado, al cual se le aplica una señal $r(t)$ a manera de entrada para obtener una respuesta o salida $y(t)$, puede representarse mediante bloques.

Figura. 1

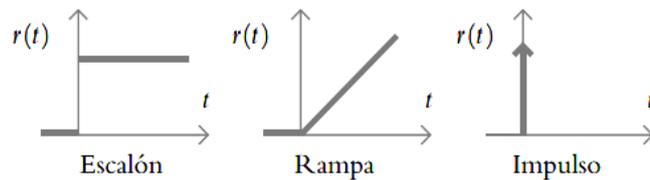
Sistema de control automático



Nota: El gráfico representa la interconexión de elementos que forman un sistema, (r) es la variable de entrada, (s) es el sistema de control este se establece de acuerdo a las necesidades del proceso en el cual se aplique el control automático, (y) son los actuadores que manipularan físicamente el proceso estos pueden ser electro-válvulas, motores, bombas, etc. Tomado de Hernández, 2017.

El vínculo entrada-salida es una relación de causa y efecto con el sistema, por lo que el proceso por controlar (también denominado planta) relaciona la salida (señales típicas) con la entrada (actuadores). Las entradas típicas aplicadas a los sistemas de control son:

- Escalón
- Rampa
- Impulso

Figura. 2*Tipos de entradas*

Nota: La gráfica escalón indica un comportamiento o una referencia constantes introducidos al sistema, la gráfica rampa supone una referencia con variación continua en el tiempo, y la gráfica impulso se caracteriza por ser una señal de prueba con magnitud muy grande y duración muy corta. Tomado de Hernández, 2017.

. La función respuesta impulso o función de transferencia es la representación matemática del sistema. Básicamente, el problema de control consiste en seleccionar y ajustar un conjunto específico de elementos tal que, al interconectarse, el sistema resultante deberá comportarse de una manera específica.

2.2. Sistema de control

Según (Mecafenix, 2019), un sistema de control es un grupo de componentes electrónicos, mecánicos, neumáticos, hidráulicos, etc. Que se utilizan en conjunto para lograr un objetivo deseado. Para que se pueda considerar como un sistema de control por lo menos debe de contar con tres elementos esenciales que son: Una variable a controlar, un actuador y un punto de referencia (set-point).

2.3. Tipos de sistemas de control

Según (Alegsa, 2016), sistema o subsistema que está constituido por un conjunto de componentes que regulan el comportamiento de un sistema (o de sí mismos) para lograr un objetivo. Cualquier sistema (organizaciones, seres vivos o máquinas) puede tener distintos sistemas de control. Existen varios tipos de sistemas de control sin embargo los tipos comunes se detallan a continuación.

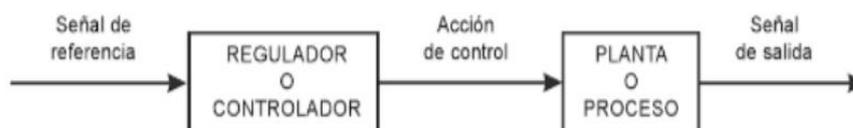
- **Sistema de control de lazo abierto**

La salida se genera dependiendo de la entrada. En otras palabras, cuando la salida para ser controlada, no se compara con el valor de la señal de entrada o señal de referencia.

En este sistema de control sólo se actúa sobre la señal de entrada, dando como resultado una señal de salida independiente a la señal de entrada, aunque basada en la primera. No existe entonces retroalimentación hacia el controlador para ajustar la acción de control. En otras palabras, la señal de salida nunca se convierte en señal de entrada para el controlador.

Figura. 3

Control de lazo abierto



Nota: En el gráfico podemos visualizar que la salida del control no es realimentada a la entrada de control, esto es la característica principal de un control de lazo abierto.

Tomado de Cajamarca Velasco & Velasco Herrera, 2018.

- **Sistemas de control de lazo cerrado**

La salida depende de las consideraciones y correcciones realizadas por la retroalimentación. En otras palabras, cuando la salida para ser controlada, se compara con la señal de referencia. La señal de salida que es llevada junto a la señal de entrada, para ser comparada, se denomina señal de feedback o de retroalimentación.

En este caso el sistema tiene una acción de control en función de la señal de salida. La señal de salida sirve para retroalimentar el sistema para ajustar la acción de control en consecuencia. (Hernández, 2017)

2.4. Control Fuzzy

Según (Brio & Sanz Molina, 2002), el control automático surge para liberar al hombre de tareas repetitivas, donde la complejidad del sistema a controlar es elevada o la operación es riesgosa, puede haber una gran cantidad de motivos por la cual se opta por el control automático.

El control manual es llevado a cabo por personas que conocen (aunque sea de manera aproximada) el proceso a controlar y saben cómo debe ser el resultado de su control y cómo lograrlo, en la industria, estas personas (operarios) cuentan con experiencia y conocimiento suficiente para cumplir con los objetivos de control. Este concepto de experiencia o base de conocimiento es muy importante en sistemas de control difuso.

La idea es muy simple, se trata de determinar de manera lógica que se debe hacer para lograr los objetivos de control de mejor manera posible a partir de una base de conocimiento proporcionada por un operador humano, sin esta base no es posible desarrollar una aplicación y que esta funcione de manera correcta. (Brio & Sanz Molina, 2002)

Figura. 4

Etapa para toma de decisiones



Nota: El gráfico muestra el proceso de inferencia difusa de manera resumida. Tomado de Brio & Sanz Molina, 2002.

2.5. Técnica de Lógica difusa

Las técnicas lógicas Fuzzy se han aplicado ampliamente en todos los aspectos en la sociedad actual, para implementar la técnica de lógica difusa a una aplicación real se requiere tres pasos:

- **Fusificación**

Es la puerta de entrada al sistema de inferencia difusa, el proceso de convertir los datos clásicos o los datos nítidos en datos difusos o funciones de membrecía (MF). Las Funciones de Membrecía representan el grado de pertenencia de un elemento a conjunto u otro, formando un subconjunto.

- **Proceso de inferencia difusa**

Los controladores difusos usan reglas, estas son proposiciones que permiten expresar el conocimiento que se dispone determinando la relación entre antecedentes y consecuentes. A estas se le conocen como Reglas Difusas y se asocian mediante operaciones lógicas (AND, OR, etc.).

Las reglas difusas suelen representarse por tablas, la edición de estas determina cuál será el comportamiento del controlador difuso, de esta manera se emula el conocimiento y experiencia del operario. Cuando se evalúan las reglas se obtienen tantos conjuntos difusos como reglas existen.

- **Desfusificación**

Es un proceso usado para convertir un conjunto difuso en un número real. El sistema de inferencia difusa usa diferentes métodos para obtener una conclusión a partir de la información de entrada. La salida difusa es obtenida por la etapa de inferencia difusa, pero el dato de salida del sistema debe ser un número real.

En la mayoría de los casos, las variables de entrada son más de una dimensión real aplicaciones, y es necesario realizar una fusificación o desarrollar una membrecía. La entrada original y la salida terminal deben ser variables nítidas, pero el proceso intermedio es un proceso de inferencia difuso. La razón por la que uno necesita cambiar

una variable nítida a una difusa es que, desde el punto de vista del control difuso o la intuición de un ser humano, no existe ninguna variable absolutamente nítida en nuestro mundo real.

2.6. Industruino

Es un controlador industrial compatible con Arduino Leonardo que puede montarse sobre un carril DIN, con entradas/salidas suficientes para conectar todo tipo de sensores y actuadores. Además, tiene una placa de prototipado en su interior para añadir componentes personalizados. Es un autómata programable que tiene como objetivo ofrecer una PLC potente con la flexibilidad y sencillez de Arduino. (Robologs, 2015)

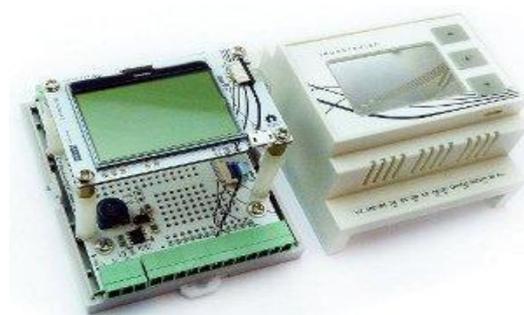
2.6.1 Tipos de Industruino

- **PROTO**

El Proto Baseboard significa "Prototipado". El Industruino estándar viene con este zócalo de prototipo. Ofrece mucho espacio y flexibilidad para añadir sus propios componentes. Tiene una gran área de prototipado, un puerto de expansión IDC de 14 pines y un regulador de voltaje de salida de 8-28 Vin \rightarrow 5V/2A. Este tablero puede ser usado si tienes algún pequeño circuito que actualmente vive en un tablero junto al Arduino. Colocándola en esta placa base se asegura de que su proyecto parezca un producto acabado y esté protegido del mundo exterior.

Figura. 5

Módulo Industruino



Nota: El gráfico nos muestra el módulo Industruino retirado el case de protección.

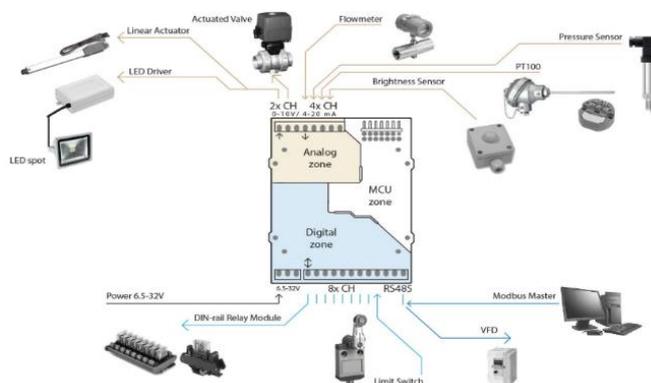
Tomado de Robologs, 2015

- **IND I/O**

El zócalo Ind.I/O significa "I/O de nivel industrial". El tablero ofrece una abundancia de opciones de interfaz, y tiene fuentes de alimentación aisladas para cada una de sus tres zonas funcionales, la placa incluye 8ch 24V I/O, 4ch 0-10V/4-20mA 18 bit ADC, 2ch 0-10V/4-20mA 12 bit DAC, transceptor aislado RS485, zonas de potencia aisladas.

Figura. 6

Zonas principales de Industruino



Nota: el gráfico nos muestra los diferentes actuadores que podemos colocar de acuerdo a la zona del módulo Industruino. Tomado de Industruino Home, 2019.

Este Baseboard es una solución de interfaz para salvar la brecha entre la compatibilidad de Arduino y el mundo de los PLC, robustos sensores y actuadores industriales. Todos los periféricos de campo están aislados del micro controlador mediante aisladores digitales y se comunican mediante el protocolo i2c. La placa se suministra con las librerías de Arduino, lo que le permite interactuar fácilmente con los sensores y actuadores de nivel industrial, utilizando sus herramientas de software favoritas.

2.7. Arduino Nano

El Arduino Nano es una placa pequeña, completa y fácil de usar basado en ATmega328P (Arduino Nano). Se puede alimentar a través de la conexión USB mini,

una fuente no regulada de 6-20 V por el pin 30 o una fuente regulada de 5v por el pin 27, la alimentación se selecciona automáticamente dependiendo de la más alta.

Cuenta con 14 pines digitales los cuales se pueden usar como entrada o salida. Cada pin proporciona o recibe un máximo de 40 mA y tiene una resistencia interna pull-up (desconectada por defecto) de 20-50 kohm. (Cajamarca Velasco & Velasco Herrera, 2018)

Figura. 7

Arduino Nano



Nota: el gráfico muestra el módulo arduino Nano. Tomado de Cajamarca Velasco & Velasco Herrera, 2018.

2.8. Medidores de nivel de líquidos

Según (Veto y Compañía Ltda., 2019), para la medición del nivel de líquidos, podemos hablar de tres métodos de medición principales: medición directa, Medición de presión hidrostática y Medición de las características eléctricas del líquido. A continuación, desglosamos cada una de estos métodos:

2.8.1. Medición directa

- **Medición por varilla**

Se trata de una varilla o regla graduada, de la longitud conveniente para introducirla dentro del depósito. La determinación del nivel se efectúa por la lectura directa en donde la varilla está mojada por el líquido. En el momento de la lectura el

tanque debe estar abierto a presión atmosférica. Se emplea en tanques de agua a presión atmosférica y de bajo volumen. (Veto y Compañía Ltda., 2019)

- **Medición por cinta y plomada**

Se emplea en depósitos o pozos donde es más difícil que la varilla llegue al fondo. Consiste en una cinta graduada y un plomo en la punta. Puede utilizarse midiendo también el punto donde la cinta está mojada, o emplear la técnica que utilizan los pozómetros, que en lugar de una plomada poseen un sensor que emitirá una señal acústica y/o lumínica al tomar contacto con la superficie del agua, el nivel se obtiene por la diferencia entre la profundidad total del pozo y la cantidad de metros que el sensor debió bajar para tomar contacto con el agua. (Veto y Compañía Ltda., 2019)

- **Visor de vidrio**

Consiste en un tubo de vidrio con su extremo inferior conectado al tanque generalmente mediante tres válvulas (dos de cierre de seguridad en los extremos del tubo, para impedir el escape del líquido en caso de rotura del cristal y una de purga). Funciona por principio de vasos comunicantes. El nivel de vidrio va acompañado de una regla graduada. Se emplea para presiones hasta 7 bar. A presiones más elevadas el vidrio es grueso, de sección rectangular y está protegido por una armadura metálica. (Veto y Compañía Ltda., 2019)

Figura. 8

Mirilla de vidrio



Nota: el gráfico muestra la mirilla de vidrio graduada. Tomado de John Moncrieff Ltd, 2019.

- **Manométrico**

Consiste en un manómetro conectado directamente a la parte inferior del tanque. El manómetro mide la presión debida a la altura de líquido que existe entre el nivel del tanque y el eje del instrumento. La medición está limitada a tanques abiertos y el nivel viene influido por las variaciones de densidad del líquido. (Veto y Compañía Ltda., 2019)

- **Burbujeo**

Mediante un regulador de caudal se hace pasar por un tubo (sumergido en el depósito hasta el nivel mínimo), un pequeño caudal de aire o gas inerte hasta producir una corriente continua de burbujas. La presión requerida para producir el flujo continuo de burbujas equivale a la medida de la columna de líquido. Este sistema es muy ventajoso en aplicaciones con líquidos corrosivos o con materiales en suspensión, ya que el fluido no penetra en el medidor ni en la línea de conexión. (Veto y Compañía Ltda., 2019)

- **Transmisor hidrostático**

Consiste en un diafragma en contacto con el líquido del tanque, que permite medir la presión hidrostática en un punto del fondo del tanque. En un tanque abierto esta presión es proporcional a la altura del líquido en ese punto y a su peso específico. El diafragma forma parte del transmisor electrónico de presión hidrostática. Estos dispositivos poseen una alta exactitud, el material del diafragma debe ser el adecuado para el tipo de líquido que se trate. (Veto y Compañía Ltda., 2019)

2.8.2. Medición de las características eléctricas del líquido

- **Medidor conductivo**

Según (Veto y Compañía Ltda., 2019), consiste en uno o varios electrodos y un relé eléctrico o electrónico que es excitado cuando el líquido moja a dichos electrodos. El líquido debe ser lo suficientemente conductor como para excitar el circuito electrónico.

El relé electrónico dispone de un temporizador de retardo que impide su enclavamiento ante una ola del nivel del líquido o ante cualquier perturbación momentánea o bien en su lugar se disponen dos electrodos poco separados enclavados eléctricamente en el circuito.

El instrumento se usa como alarma o control de nivel alto y bajo, utiliza relés eléctricos o electrónicos, en función de la conductividad del líquido. Es versátil, sin partes móviles, su campo de medida es grande con la limitación física de la longitud de los electrodos. El líquido contenido en el tanque debe tener un mínimo de conductividad y si su naturaleza lo exige, la corriente debe ser baja para evitar el deterioro.

- **Medidor Led**

Está compuesto por un cable multipar de 3 pares y 6 hilos conectados a un indicador de nivel. Uno de estos cables lleva adosado una plomada en su extremo y debe tener el largo suficiente para alcanzar el fondo del estanque, los 5 restantes se deben cortar a tramos equidistantes unos de otros y pelarse en la punta. Dado que estos cables son sensibles al contacto del agua, transmiten una señal al indicador que mostrará mediante luces Led cada uno de los niveles alcanzados. (Veto y Compañía Ltda., 2019)

- **Medidor capacitivo**

El medidor de nivel capacitivo mide la capacidad del condensador formado por el electrodo sumergido en el líquido y las paredes del tanque. La capacidad del conjunto depende linealmente del nivel del líquido. En fluidos no conductores se emplea un electrodo normal y la capacidad total del sistema se compone de la del líquido, la del gas superior y la de las conexiones superiores. En fluidos conductores el electrodo está aislado usualmente con teflón interviniendo las capacidades adicionales entre el material aislante y el electrodo en la zona del líquido y del gas. (Veto y Compañía Ltda., 2019)

- **Medidor ultrasónico**

Se basa en la emisión de un impulso ultrasónico a una superficie reflectante y la recepción del eco del mismo en un receptor. El retardo en la captación del eco depende del nivel del tanque. Este sistema se utiliza para todo tipo de tanque y líquido o lodo. Pueden usarse en áreas clasificadas. Son sensibles a la densidad de los fluidos y dan

señales erróneas cuando la superficie del nivel del líquido no es nítida (por ej.: líquido que forme espuma), ya que se crean falsos ecos de los ultrasonidos. (Veto y Compañía Ltda., 2019)

Figura. 9

Sensor ultrasónico Teach



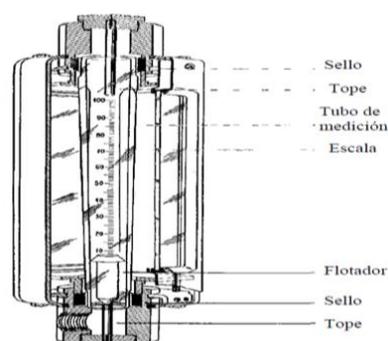
Nota: El gráfico muestra el sensor ultrasónico con su botón de configuración para establecer los puntos máximo y mínimo de censado. Tomado de Avnet Company, 2019.

2.9. Rotámetro

Según (Andira, 2019), esta herramienta se utiliza, principalmente, para controlar el rendimiento de bombas y válvulas, para la medición de capilares en equipos de refrigeración, para calcular el consumo de combustible en calderas, dosificar aditivos y muchas aplicaciones más.

Figura. 10

Partes de un rotámetro



Nota: En el gráfico muestra la constitución interna del rotámetro. Tomado de Cerven & Naranjo, 2017.

2.9.1. Elementos

Según (Cerven & Naranjo, 2017) los elementos del rotámetro son los que se presenta a continuación:

- **Tubo de medición**

Este tiene una forma de cono truncado, que por lo general se modifica ligeramente para obtener una relación lineal exacta. El ángulo (α) del tubo suele ser pequeño del orden de los 2 a 3°, lo cual hace despreciable el factor de escala (a) en la ecuación. Casi todos los tubos de los rotámetros llevan por dentro guías que permiten que el flotador se mantenga centrado. Esto ya que, un movimiento irregular no centrado del flotador, puede producir errores en la medida y en todo caso una difícil lectura de su posición.

Estas guías pueden ser de dos tipos. La forma más común es el uso de canales sobre las paredes del tubo de medición con el fin de guiar al flotador por sus costados. Otras veces el flotador está perforado y el tubo lleva una guía central en forma de un eje fino sobre el cual desliza el flotador.

- **Flotador**

El flotador de un rotámetro es un elemento que tiene formas variadas y se fabrica de diversos materiales, según el fluido a medir. El material de fabricación debe cumplir con una serie de criterios a saber:

- a) Ser más pesados que el fluido del proceso.
- b) Resistir convenientemente a la corrosión.
- c) Permitir un buen deslizamiento sobre las guías.

En función de esto los materiales más utilizados son acero inoxidable, monel y níquel. Sin embargo, es también posible conseguir flotadores de latón o bronce y en algunos casos de materiales compuestos. Este último sobre todo cuando se trata de medición de flujo de aire, por ser éste de baja densidad.

En los rotámetros se utilizan flotadores de formas diversas, esto determina la influencia de los cambios de viscosidad del fluido en el comportamiento del rotámetro; puesto que el coeficiente de descarga está influenciado por la viscosidad del fluido. Las formas más comunes son:

- a) **Flotador esférico:** Para bajos caudales y poca precisión, con una influencia considerable de la viscosidad.
- b) **Flotador cilíndrico con borde plano:** Para caudales medios y elevados con una influencia media de la viscosidad.
- c) **Flotador cilíndrico con borde saliente, con la cara inclinada de frente al flujo:** Este flotador presenta menor influencia de la viscosidad del fluido.
- d) **Flotador cilíndrico con bordes salientes, contra el flujo:** Es el que presenta la menor influencia de la viscosidad del fluido.

Según el material con que se fabrica el tubo el rotámetro puede ser de visión directa en cuyo caso el material del tubo suele ser vidrio, que puede o no resistir altas temperaturas (Pyrex), o cualquier otro material transparente como acrílico. En la figura F30 se ilustra el rotámetro de vidrio. (Cerven & Naranjo, 2017)

- **Escala**

La escala puede venir grabada sobre una regla metálica la cual se monta lateralmente con el tubo de medición, o directamente sobre el propio tubo. La escala del rotámetro es casi lineal, especialmente si el ángulo del cono es pequeño (de 2 a 3°) la rangeabilidad máxima de este instrumento es de 10:1.

- **Ventajas**

- a) Costo bajo.
- b) Simple.
- c) Pérdidas de carga baja.
- d) Para aplicaciones de líquidos, gases o vapor.
- e) No requiere fuente de alimentación.

- **Desventajas**

- a) Las unidades en algunos casos son voluminosas.

- b) El costo se incrementa considerablemente con operaciones extras (corazas de protección, montaje en tablero).
- c) Transmisión no disponible como estándar.
- d) Las incrustaciones de suciedad pueden volver difícil la lectura.
- e) Requiere calibración específica para cada fluido.

- **Funcionamiento**

Su funcionamiento se basa en la medición del desplazamiento vertical de un elemento sensible, cuya posición de equilibrio depende del caudal circulante que conduce simultáneamente a un cambio en el área del orificio de pasaje del fluido. De esta manera, la diferencia de presiones que actúan sobre el elemento móvil permanece prácticamente constante.

- **Fuerza de gravedad**

La fuerza que equilibra a un rotámetro la constituye la fuerza de gravedad que actúa sobre el elemento sensible, construido, por lo general, de forma cilíndrica con un disco en su extremo. El tubo contiene varios orificios laterales por donde circula fluido que induce una rotación alrededor de su eje.

- **Forma**

Los rotámetros están hechos, en su mayoría, de un tubo de vidrio con un extremo angosto hacia abajo. En el interior del tubo se encuentra un flotador. Bajo la acción de la corriente de líquido o gas, dicho flotador se desplaza verticalmente para indicar sobre una escala graduada el caudal circulante.

La corriente fluida que se dirige de abajo hacia arriba a través del tubo cónico provoca la elevación del flotador hasta una altura en la que el área anular (entre las paredes del tubo y el cuerpo del flotador) adquiere una dimensión donde el flotador se equilibra y mantiene estable a una altura que corresponde a un determinado valor de caudal circulante. (Andira, 2019)

2.10. Carril DIN

Según (Martín, 2019), Un carril DIN es una barra metálica normalizada de 35 mm de ancho con una sección transversal en forma de sombrero. Es muy empleado para instalaciones eléctricas, así como para equipamiento de comunicaciones con equipos embarcados o en armarios a la intemperie. La robustez de estos equipos, con rangos extendidos de temperatura y carcasa para este fin, es la máxima de su instalación.

Figura. 11

Carril Din



Nota: El gráfico muestra el carril Din instalado como soporte de elementos. Tomado de Martín, 2019.

2.11. Electroválvula

Según (Distritec, 2013), Las electroválvulas son dispositivos que responden a pulsos eléctricos. Gracias a la corriente que circula a través del solenoide es posible abrir o cerrar la válvula controlando, de esta forma, el flujo de fluidos. Al circular corriente por solenoide genera un campo magnético que atrae el núcleo móvil y al finalizar el efecto del campo magnético, el núcleo vuelve a su posición, en la mayoría de los casos, por efecto de un resorte.

Las electro-válvulas se utilizan en gran número de sistemas y rubros industriales que manejan fluidos como el agua, el aire, el vapor, aceites livianos, gases neutros y otros. En particular, las electroválvulas suelen implementarse en lugares de difícil acceso

ya que pueden ser accionadas por medio de acciones eléctricas. También son utilizadas en vacío o hasta en altas presiones y altas temperaturas.

Figura. 12

Electro-válvula



Nota: El gráfico muestra diferentes tipos de electro-válvulas. Tomado de Distritec, 2013.

Las electroválvulas se aplican a surtidores automáticos de combustibles, irrigación de parques, fuentes de agua danzantes, dosificadores de líquidos o gases, regulación de niveles de líquidos, en máquinas envasadoras, lavaderos automáticos de autos, máquinas de limpieza, procesos de niquelado o galvanizado, en máquinas de café y en muchos lugares más.

2.12. Fuentes de alimentación

Según (Mecafenix, 2018), Una fuente de alimentación es un dispositivo utilizado para alimentar los circuitos de los aparatos electrónicos, como pueden ser, televisores, computadoras, impresoras, el cargador de tu celular. Existen diferentes tipos de fuentes de alimentación ya sea fijas o variables, o también pueden clasificarse por su tipo de funcionamiento. Cada uno con sus características, ventajas y desventajas.

2.12.1. Tipos de fuentes de alimentación

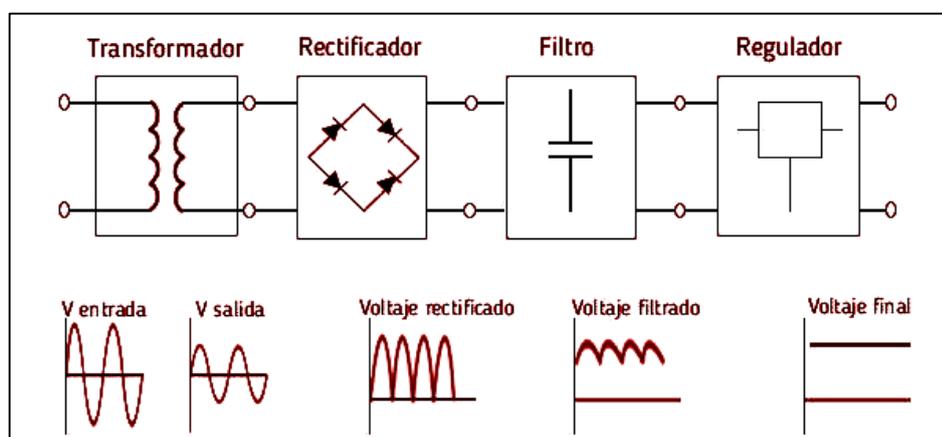
Existen dos tipos de fuentes de alimentación que se presentan a continuación:

- **Fuentes de alimentación lineales**

Según (Raffino, 2019), Siguen el esquema de transformador (reductor de tensión), rectificador (conversión de voltaje alterno a onda completa), filtro (conversión de onda completa a continua) y regulación (mantenimiento del voltaje de salida ante variaciones en la carga).

Figura. 13

Fuente de alimentación lineal



Nota: El gráfico muestra la transformación de la señal alterna a continua y como la señal va cambiando de acuerdo a la etapa en la que se encuentra, esta fuente se caracteriza por ser de bajo costo, pero poco eficiente. Tomado de Mecafenix, 2018.

- **Transformador**

El transformador se encarga de modificar el voltaje de entrada a un voltaje con mayor o menor amplitud en su salida, dependiendo el tipo de fuente a utilizar. Estos dispositivos solo pueden trabajar con voltaje alterno y es uno de los elementos principales dentro de una fuente de alimentación.

- **Rectificador**

El rectificador está construido por un puente de diodos, el cual recibe el voltaje del transformador y se encarga de convertirlo en voltaje de corriente directa.

- **Filtro**

El filtro se encarga de disminuir la variación de voltaje, a través de capacitores. Con este filtrado se genera un fenómeno llamado efecto rizo.

- **Regulación**

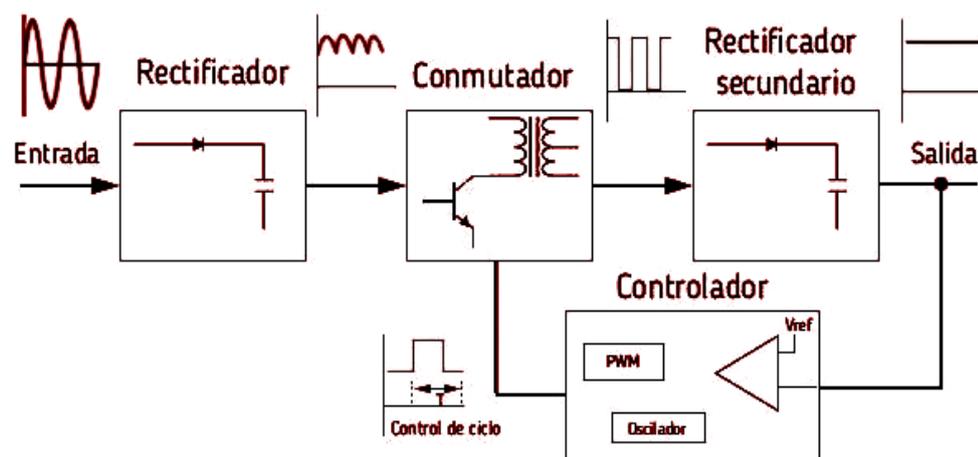
El regulador se encarga de recibir la señal proveniente del filtro para que en la salida se obtenga un voltaje continuo sin importar las variaciones de voltaje dependiendo el regulador es el voltaje de salida.

- **Fuentes de alimentación conmutadas**

Convierten la energía eléctrica por medio de conmutación de alta frecuencia sobre transistores de potencia. Las fuentes lineales son típicamente de regulación ineficiente, comparadas con fuentes conmutativas de similar potencia. Estas últimas son las más utilizadas cuando se requiere un diseño compacto y de bajo costo. (Raffino, 2019)

Figura. 14

Fuente de alimentación conmutada



Nota: El gráfico muestra la transformación de la señal alterna a continua dentro de una fuente conmutada. Tomado de Mecafenix, 2018.

- **Rectificación y filtrado**

En este bloque se rectifica y filtra el voltaje de corriente alterna convirtiéndolo en una señal continua y pulsante.

- **Conmutación**

Se encarga de convertir la señal pulsante en una onda cuadrada, la cual es introducida a un transformador. En este bloque existen diferentes tipos de configuraciones como son: Buck, Boost, Buck-Boost.

- **Rectificación y filtrado secundario:**

Se vuelve a rectificar y filtrar la salida del bloque anterior, para poder entregar una señal continua más lineal.

- **Controlador**

Se encarga de controlar la oscilación de la etapa de conmutación. Este bloque se compone de un oscilador de frecuencia fija, un voltaje de referencia, un comparador de voltaje y un modulador de ancho de pulso (PWM). El modulador recibe el pulso del oscilador y modifica su ciclo según la señal que envía el comparador. Este comparador examina el voltaje de salida de la rectificaron secundaria con el voltaje de referencia. (Mecafenix, 2018).

CAPÍTULO III

DESARROLLO DEL TEMA

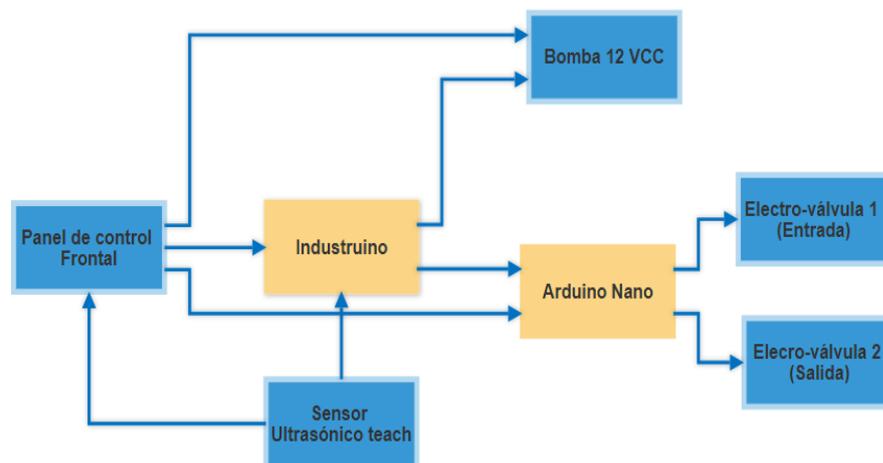
En este proyecto se presenta la implementación de un control automático de nivel mediante Industruino, cuenta con 2 electroválvulas, un sensor ultrasónico, una bomba de 12VCC, un rotámetro y un módulo arduino Ind.

La base del proyecto es el control de nivel, en este caso por medio de un control-Fuzzy se manipula la apertura de las electroválvulas tanto de entrada como de salida del tanque principal, para mantener el nivel mientras la bomba está activada, el objetivo es lograr un nivel estable con un flujo constante.

El elemento de mayor prioridad es el Industruino, al cual se conecta el sensor ultrasónico Teach, un arduino Nano, el panel frontal y 2 electros - válvulas (de entrada y salida).

Figura. 15

Diagrama de conexiones de Industruino



Nota: El gráfico muestra las conexiones realizadas en el módulo PCT 3, mediante un diagrama en bloque.

La señal del sensor ultrasónico es tomada y cuantificada para establecer el nivel de líquido en el tanque principal, el nivel se compara con los valores establecidos en el

panel frontal y de acuerdo al acondicionamiento programado en el módulo Industruino, se envía una señal a un arduino Nano, elemento funcionará como un transmisor para comunicar los actuadores (electro-válvulas) con el controlador (Industruino) permitiendo elevar o disminuir el nivel de líquido en el tanque.

Los valores principales se pueden visualizar en tiempo real en el LCD del módulo Industruino.

3.1 Características de los elementos

A continuación, se detallan las características técnicas de los principales elementos eléctricos como Industruino, arduino Nano, bomba Jabasco, del módulo didáctico PCT 3.

3.1.1 Módulo Arduino IND.

Tabla 1

Industruino- Zona Analógica

Zona Analógica	Características
Entradas analógicas	4
Salidas analógicas	2
Niveles analógicos	0-10 V/ 4-20 mA
Resolución de entradas analógicas	18 bit
Resolución de salidas analógicas	12 bit
Aislamiento de micro-controlador y zona digital	1 KV aislamiento

Nota: Esta tabla muestra las principales características de la zona analógica del módulo Industruino como son número de entradas y salidas analógicas, niveles analógicos aceptables, resolución de entradas analógicas, resolución de salidas analógicas y aislamiento del microcontrolador con la sección de potencia. Tomado del manual guía Industruino.

Tabla 2*Industruino -zona digital*

Zona Digital	Características
Rango de alimentación	6.5-32V
Canales Digitales I/O	8
Rango de tensión entradas digitales	3.3-32V
Rango de tensión salidas digitales	6.5-32V
Máxima potencia de salida digital	2.3 A short circuit
Corriente de salida total	6 Amp

Nota: Esta tabla muestra las principales características de la zona digital del Industruino como el rango de alimentación, canales digitales, etc.

Tabla 3*Industruino -zona de micro-controlador*

Zona Micro-Controlador	Características
Memoria Flash	128KB
SRAM	2/8 KB
EEPROM	4 KB
Reloj interno	16 MHz
Alimentación de Micro-controlador	5 Vcc
Librería para LCD	UC 1701
Botones panel frontal	3 botones (membrana)

Nota: Esta tabla muestra las principales características de la zona del micro-controlador del módulo Industruino. Tomado del manual guía de Industruino.

3.1.2 Arduino Nano

Tabla 4

Características de arduino Nano

Zona	Características
Micro-controlador	Atmega 238
Voltaje de operación	5 Vcc
Voltaje de alimentación	7-12 Vc
Puertos digitales I/O	14 (6 PWM)
Memoria Flash	32 KB
EEPROM	1KB
Frecuencia de trabajo	16 MHz
Resolución	10 bits

Nota: Esta tabla muestra las principales características del arduino Nano. Tomado del manual guía de arduino Nano.

3.1.3 Sensor TEACH UB500- 18GM75-I-V15

Tabla 5

Características de sensor Teach

Ítem	Características
IP Rating	IP 67
Voltaje de operación	4-20 mA
Voltaje de alimentación	10-30 Vcc
Resolución	0.13mm
Rango de detección	30-500 mm
Impedancia de carga	0 – 300 Ohm
Frecuencia del transductor	380 kHz
Influencia de la temperatura	+/- 1.5% del valor

Ítem	Características
Tipo de salida	Corriente

Nota: Esta tabla muestra las principales características de alimentación del sensor ultrasónico Teach. Tomado del manual guía del Sensor TEACH UB500- 18GM75-I-V15.

Tabla 6

Elementos de indicación.

Elementos	LED	Indicación
LED Amarillo	Permanente	Objeto en rango evaluación
	Intermitente	Función Teach-in objeto detectado
LED Rojo	Permanente	Avería
	Intermitente	Función Teach-in objeto no detectado

Nota: Esta tabla muestra como reconocer la calibración de sensor ultrasónico Teach. Tomado del manual guía Sensor TEACH UB500- 18GM75-I-V15.

3.1.4 Electro-válvula

Tabla 7

Características de servo-motor (electro-válvula)

Ítem	Características
Voltaje de alimentación	4.8-6.8 Vcc
Velocidad de funcionamiento (5.0V)	0.15seg/grados
Velocidad de funcionamiento (6.8V)	0.13seg/grados
Par de parada (5.0V)	21 kg/cm
Par de parada (6.8V)	25 kg/cm

Nota: Esta tabla muestra las principales características del servo motor. Tomado del manual guía del servo motor.

3.1.5 Bomba JABSCO

Tabla 8

Características de bomba

Ítem	Características
Voltaje de alimentación	12 Vcc
Consumo de Corriente	8 Amp.
Fusible recomendado	15 Amp.
Velocidad de Flujo	6.3 GPM
Tamaño del Puerto interno	1/2 ”
Tamaño del Puerto externo	1”
Case	Metálico

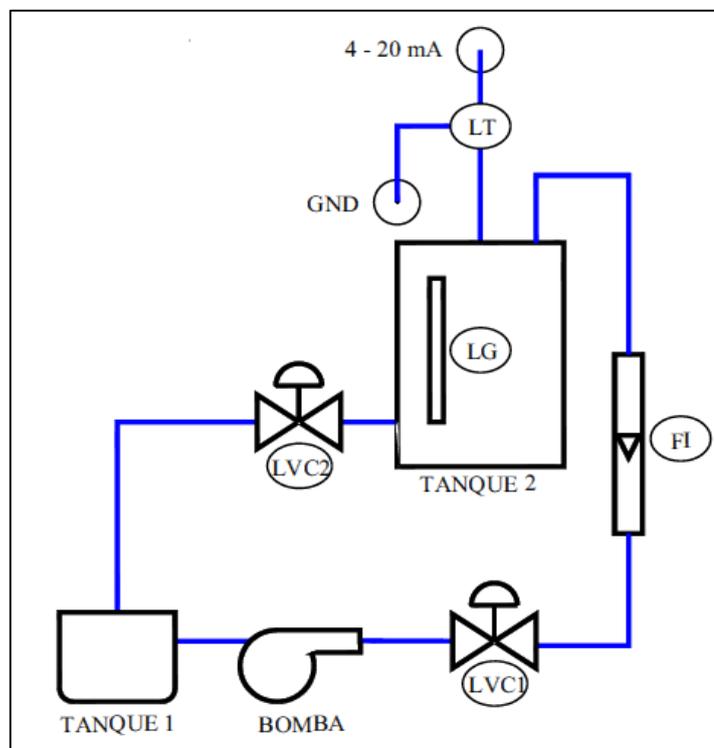
Nota: Esta tabla muestra las características principales de la Bomba Jabsco su voltaje de alimentación, consumo de corriente, etc.

3.2 Características del proceso

3.2.1 P&ID del módulo

P&ID es el acrónimo de "Piping and instrumentation diagram", es un diagrama detallado que muestra las tuberías y el equipo de procesos junto con los dispositivos de instrumentación y control. El diagrama de tuberías e instrumentación (P&ID) a menudo se considera el estándar de oro para el diseño, la operación y el mantenimiento adecuados de plantas en las industrias de procesos, incluidas las instalaciones químicas, de petróleo y gas, las operaciones mineras, la industria alimentaria, plantas de procesamiento y plantas de tratamiento de agua y aguas residuales.

El P&ID proporciona información importante para la fabricación e instalación de equipos y maquinaria, tuberías, instrumentación, arranque seguro y apropiado y operación correcta de la planta o estación.

Figura. 16*P&ID de módulo PCT 3*

Nota: El gráfico muestra la información sobre el conexionado y elementos instalados en la estación PCT 3.

Tabla 9*Nomenclatura de P&ID*

Nomenclatura	Significado	Equipo
LT	Level Transmitter	Sensor Teach
LG	Level Gauge	Mirilla de Vidrio
LVC1	Control Valve Level 1	Electro-válvula 1
LVC2	Control Valve Level 2	Electro-válvula 2
FI	Flow Indicator	Rotámetro

Nota: Esta tabla muestra la nomenclatura y el significado de los elementos contenidos en el P&ID, de igual manera se muestra el equipo que corresponde cada elemento.

3.2.2 Control Fuzzy

Un sistema de control difuso usa el conocimiento experto para generar una base de datos que darán al sistema la capacidad de tomar decisiones sobre ciertas acciones que se presentan en su funcionamiento, permite describir un conjunto de reglas que utilizaría una persona para controlar un proceso. El sistema de control usado es MIMO (Múltiples entradas – Múltiples Salidas).

Para implementar la técnica de lógica difusa a una aplicación real se requiere tomar en consideración las entradas (sensores o panel de ingreso de datos), la experiencia de usuario (operario) y las salidas (actuadores como: electro-válvulas, bomba), de acuerdo a estas consideraciones se siguen los siguientes parámetros:

- a) Fusificación
- b) Proceso de inferencia difusa
- c) Defusificación

- **Fusificación**

En la fusificación se asignan grados de pertenencia a cada una de las variables de entrada en este caso particular las variables son “Set-Point” (tomada del panel frontal) y “Sensor” (tomada del sensor ultrasónico), además se considera el “error” como una variable interna puesto que este resulta de la diferencia entre “Set-Point” y “Sensor”.

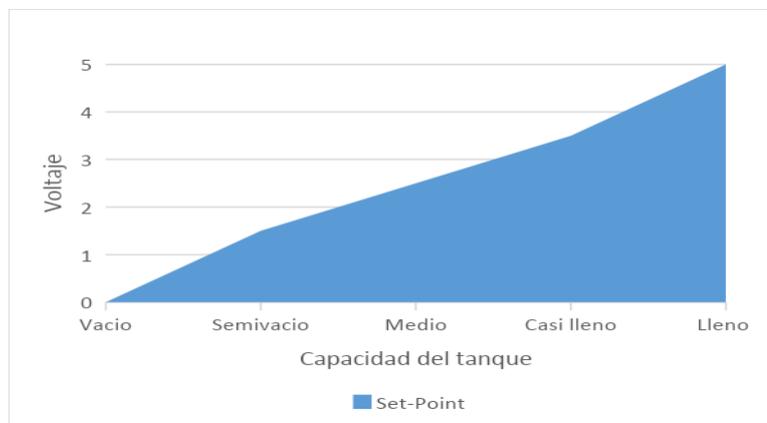
Mediante la Ley de Ohm se explica la manera de obtener el voltaje de entrada (Set-Point), el cual es mapeado, estableciendo 0 V como “vacío”, 1,5 V como “Semi-vacío”, 2,5 V como “Medio”, 3,5 V como “Casi lleno”, 5 V como “Lleno”. La función Mapeo nos permite relacionar los valores de voltaje con las premisas de capacidad de manera efectiva, asumiendo por diseño el valor de resistencia variable con un valor Máximo (5K) y un amperaje (0,001mA).

$$E = I \times R$$

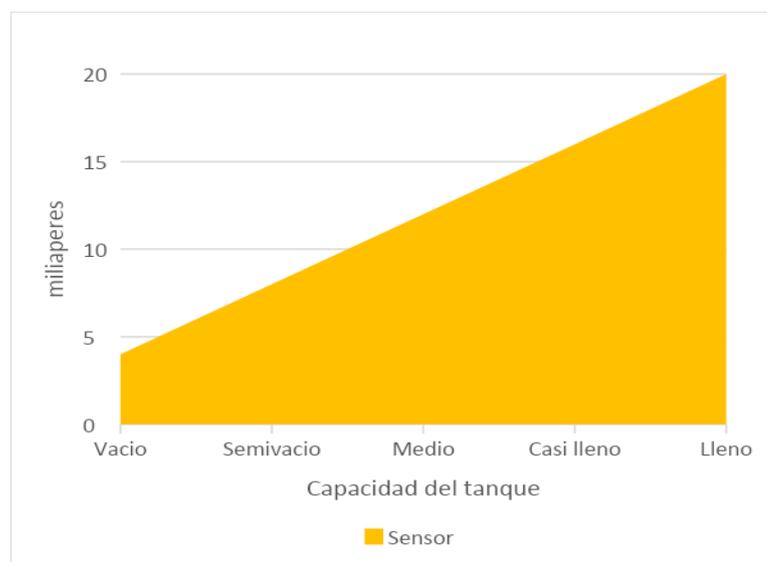
$$E = 0,001 \times R$$

$$E = 0,001 \times 5k$$

$$E = 5v$$

Figura. 17**Fusificación de set-point**

Nota: El gráfico muestra las premisas que se asignan a cada valor de voltaje de acuerdo a la experiencia del usuario de la estación PCT 3 para la fusificación de la variable "Set-Point" la cual es obtenida desde el panel frontal.

Figura. 18**Fusificación del sensor ultrasónico Teach**

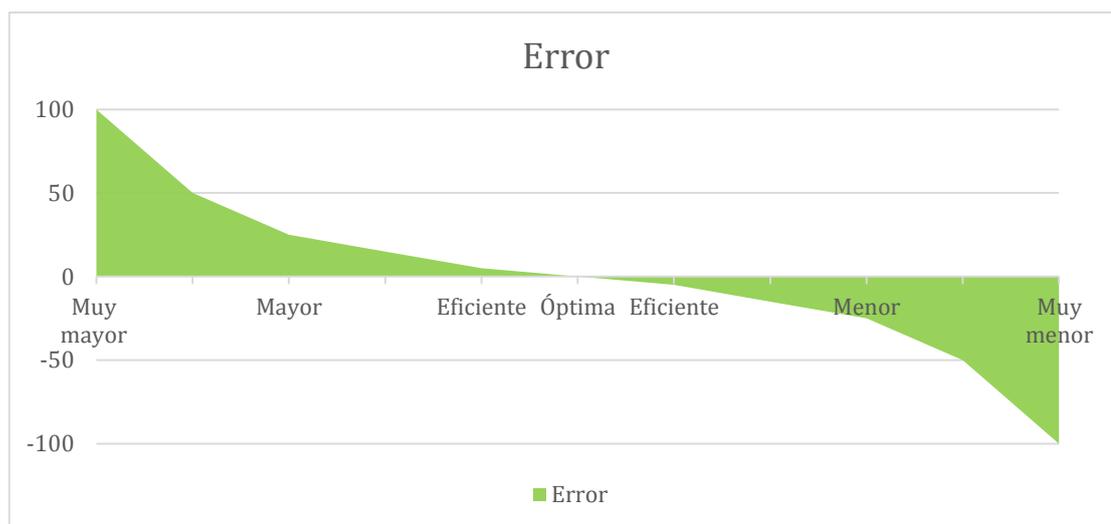
Nota: El gráfico muestra las premisas que se asignan a cada valor de corriente de acuerdo a la experiencia del usuario de la estación PCT 3 para la fusificación de la variable "snsor" obtenida del sensor ultrasónico Teach.

La manera de obtener la corriente de entrada (Sensor), es directa puesto que el fabricante lo especifica en el manual guía. El valor es mapeado mediante el transmisor arduino Nano y el controlador Industruino para minimizar los errores de lectura, estableciendo 4mA como “vacío”, 8mA como “Semi-vacío”, 12mA como “Medio”, 16mA como “Casi lleno”, 20mA como “Lleno”.

La función Mapeo nos permite relacionar los valores de Amperaje con las premisas de capacidad de manera efectiva, el resultado es un valor cuantitativo sin unidad de medida por lo cual se toma como referencia las premisas para la construcción de las reglas difusas y el valor numérico para la defusificación.

Figura. 19

Fusificación de variable “error”



Nota: El gráfico muestra las premisas que se asignan a cada valor de acuerdo a la experiencia del usuario de la estación PCT 3 para la fusificación de la variable “error” obtenida de la sustracción de la variable “Set-point” y “sensor”.

$$\text{Error} = \text{Set-Point} - \text{Sensor}$$

El “error” establece la eficiencia con la que el sistema se acopla al valor de nivel establecido (Set-Point), mientras más se acerque el valor de error a cero el control será

más eficiente. Para obtener un valor de error claro se debe mapear o establecer una relación directa entre los valores o premisas de referencia (Set-Point y Sensor).

- **Proceso de Inferencia difusa**

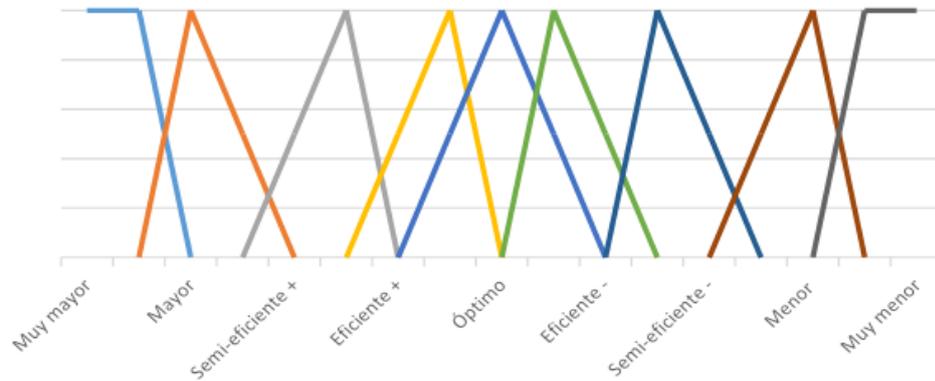
La inferencia relaciona los conjuntos difusos de entrada y salida para representar las reglas que definirán el sistema, mediante la base de conocimiento se asocia el dominio de la aplicación y los objetivos de control, en esta etapa se deben definir las reglas lingüísticas de control que decidirán la forma en la que debe actuar el sistema. Básicamente combina las funciones de membrecía con las reglas de control para derivar la salida borrosa.

Tabla 10

Reglas difusas

	Muy Mayor	Mayor	Eficiente +	Óptimo	Eficiente -	Menor	Muy Menor
Bomba	Activa	Activa	Activa	Activa	Activa	Activa	Activa
Electro - válvula 1	Abierta	Medio abierta	Constante	Constante	Medio abierta	Semi abierta	Semi abierta
Electro - válvula 2	Semi abierta	Semi abierta	Medio abierta	Constante	Constante	Medio abierta	Abierta

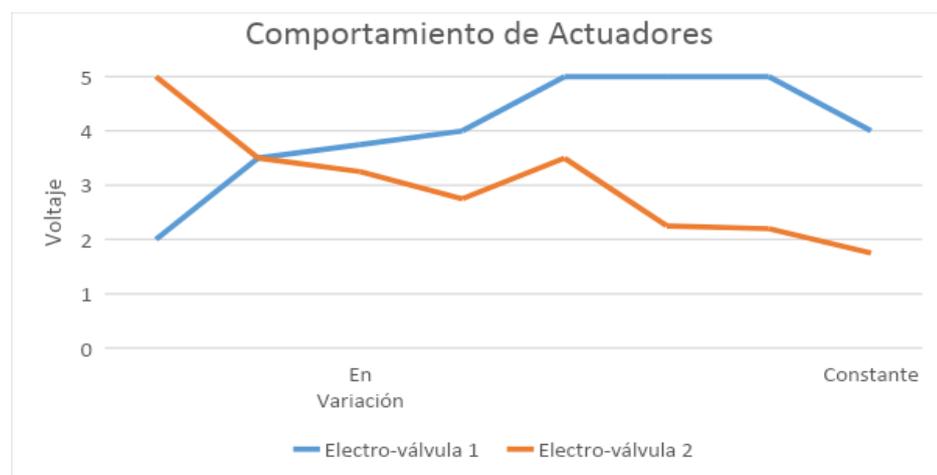
Nota: Esta tabla muestra las reglas difusas establecidas entre inferencias difusas y los actuadores, explica la acción que toman los actuadores dependiendo la premisa que tome la variable “error”. Las reglas difusas se establecen mediante programación dentro del módulo Industruino.

Figura. 20*Inferencias difusas*

Nota: El gráfico muestra la relación hecha con las premisas establecidas para el control automático.

- **Desfusificación**

La desfusificación adecua los valores difusos generados en la inferencia en valores crisp, estos son datos cuantificables que posteriormente se utilizarán en el proceso de control determinando comportamiento de los actuadores. Se utilizan diferentes métodos para determinar la salida en base a las entradas y a las reglas difusas establecidas.

Figura. 21*Desfusificación para control de actuadores*

Nota: El gráfico muestra el comportamiento de los actuadores es definido por el "error" y este por el nivel de líquido. Los electroválvulas actúan de manera difusa mientras están

en su estado de variación, pero al llegar a su estado constante, cuando el “error” se encuentra en su estado óptimo, estas tienen un comportamiento semi-lineal manteniendo el nivel deseado.

3.2.3 Estructura de programación

El programa principal se divide en 3 niveles, el nivel 1 es la visualización del salvapantallas, el nivel 2 es el nivel de control, el nivel 3 son las visualizaciones y actuadores de los cuales se detalla a continuación.

Antes de dar inicio al proceso de automatización del módulo nivel, de se debe verificar el nivel de agua en el tanque reservorio, luego verificar que el sensor ultrasónico se encuentre con los indicadores led en verde y tomate eso significa que el sensor está calibrado y listo para funcionar, después de eso se dirige a la pantalla del Industruino, en la primera ventana de interfaz se visualizará un mensaje de iniciación por 5 segundos, luego desplegará un menú el cual contiene 3 sub-menús AUTO - DEMO – SCREEN.

El menú SCREEN muestra la pantalla de iniciación nuevamente, y se quedará allí hasta presionar un botón.

El menú “DEMO” contiene diferentes funciones primarias del módulo las cuales se detallan a continuación.

- a) Bomba: en esta pantalla se puede encender o apagar la bomba del módulo de manera manual.
- b) Válvula 1: permite controlar la apertura de la electroválvula
- c) Válvula 2: permite controlar la apertura de la electroválvula
- d) Sensor de nivel: muestra la cantidad de agua que contiene el tanque principal en mili amperes siendo 4 mA el mínimo y 20 mA el máximo.
- e) Black: regresa al menú anterior.

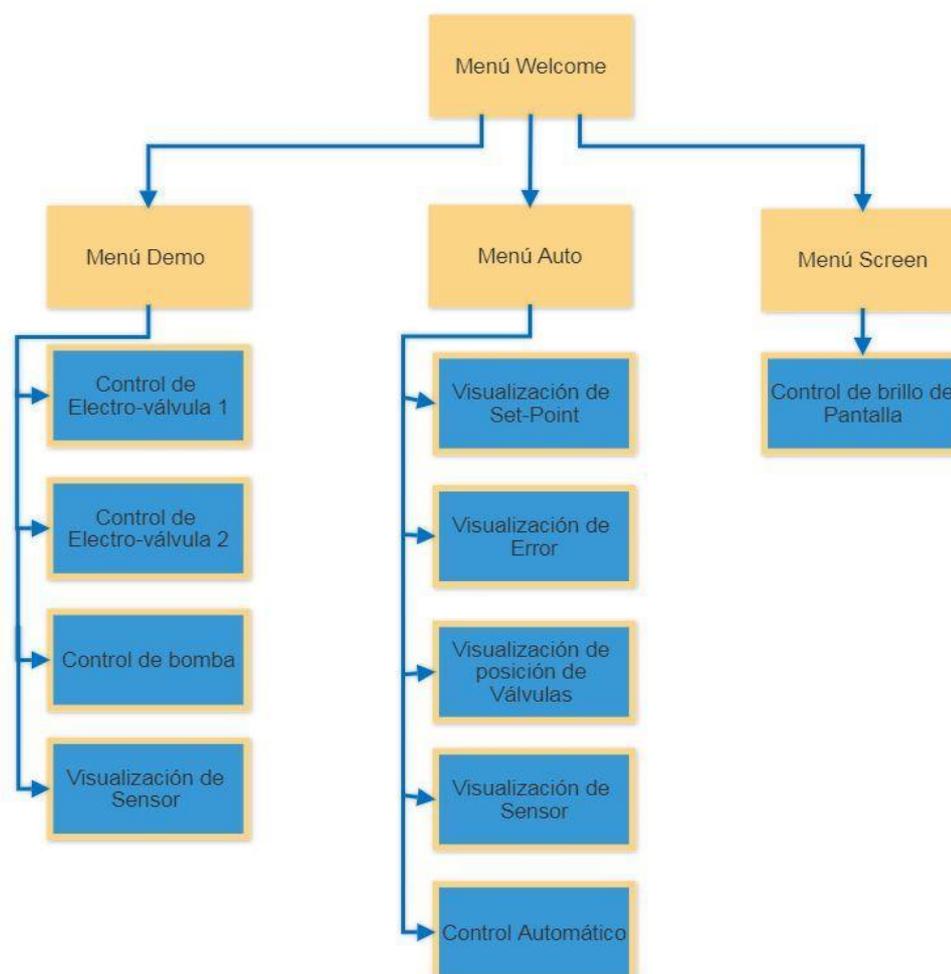
El menú “AUTO” se despliega una ventana con diferentes opciones, en la interfaz se puede comprobar los valores fijados del Set Point, la posición de las electroválvulas, además del nivel de agua medido por el sensor ultrasónico, se debe considerar que la electroválvula 1 no puede estar cerrada a más del 20%, ya que si el

flujo no es el adecuado la presión que ejerce la bomba puede provocar daños. Una vez que se verifique los datos se pasa a la etapa de control.

Etapa de control: Inicia encendiendo la bomba, se fija el valor del servo-válvula en estándar, se visualiza el valor de nivel, el valor del sensor, el valor Set-Point establecido, el porcentaje de error. A continuación, las servos-válvulas se moverán hasta lograr que el valor de error sea cero, el valor de nivel se obtiene de la resta del valor medido y del Set-Point ingresado.

Figura. 22

Estructura de programación



Nota: El gráfico muestra los niveles de programación y la estructura utilizada para acceder a los diferentes menús.

Mediante el uso de los potenciómetros logró fijar los niveles de Set-Point y la posición de las electroválvulas, los cuales serán visualizados en la pantalla. En base al módulo didáctico PCT 3 se realizaron las adecuaciones necesarias para el correcto funcionamiento.

3.3 Proceso de automatización del módulo de nivel PCT-3

Se realizó el desmontaje de las piezas obsoletas como el radiador y reductores de $\frac{1}{4}$ de pulgada correspondiente a la tubería anterior de módulo de nivel PCT-3, de la misma manera se hizo un corte en el parte horizontal del soporte para incluir el instrumento de medición de flujo (rotámetro).

Figura. 23

Desmontaje del módulo PCT 3.



Nota: El gráfico muestra el desmontaje de elementos del módulo PCT 3.

Una vez realizada las modificaciones del módulo se incluyen los elementos necesarios, los cuales van a realizar el control de nivel. Se comienza por colocar la bomba de 12V a 7.2A, acoplamos a la salida de la misma a la servo-válvula 1 la cual controlará el flujo de agua de esta manera llegará del tanque reservorio al tanque

principal, y a su vez pasando por el rotámetro el cual mide el nivel de flujo de agua de 40 lt/h a 400 lt/h.

El agua llega al tanque principal en el cual se encuentra un servo válvula 2 la cual controla el nivel de agua y este se puede visualizar en la mirilla de vidrio instalada en el tanque

Figura. 24

Montaje de los instrumentos de medición



Nota: El gráfico muestra el montaje de los instrumentos de medición (rotámetro y mirilla de vidrio).

Para tener un una servo-válvula didáctica se soldó un soporte metálico a una llave de paso con palanca ya que esta cuenta con una esfera perforada que al girar corta el flujo de líquido de manera proporcional, adicional al soporte se le colocó un servomotor de 25 kg/m para poder mover la palanca. Entre la llave de paso y el servomotor se colocó un acople plástico.

Figura. 25

Electro – válvula didáctica (servo-válvula).



Nota: El gráfico muestra la electro-válvula didáctica ensamblada.

3.4 Diseño del panel frontal y Ensamblaje de panel frontal

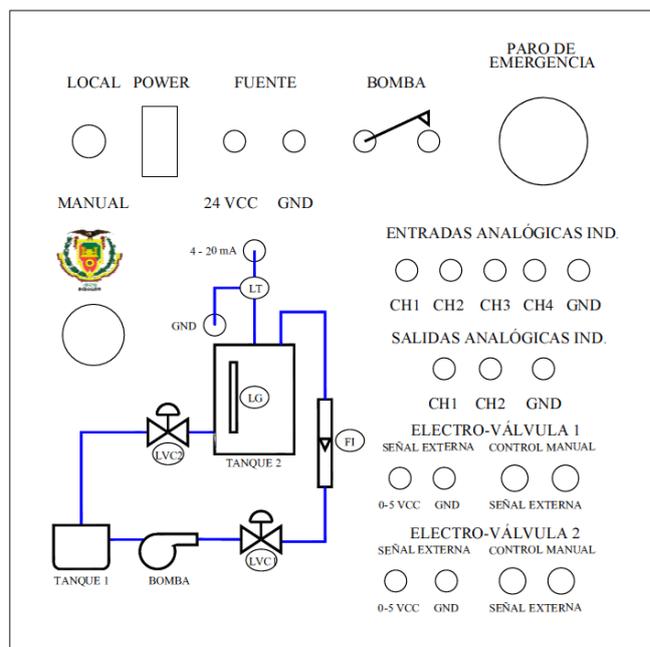
Para lograr una toma de datos más didáctica se diseñó un panel frontal (Véase el diagrama en el anexo A), el mismo que cuenta con:

- a) Switch 1: Permite seleccionar la manera de usar el panel en modo local o manual.
- b) Switch 2: Permite energizar el módulo didáctico.
- c) Bornera “Fuente”: Permite tener una fuente de 24VCD en el panel frontal
- d) Bornera “Bomba”: Permite encender la bomba de manera manual
- e) Pulsador de Emergencia: Permite detener la bomba en caso de emergencia
- f) Borneras de Sensor: Permiten medir directamente el sensor Ultrasónico de 4 – 20 mA
- g) Borneras “IN” Industruino: Permite usar las entradas analógicas del Industruino desde el panel frontal
- h) Borneras “OUT” Industruino: Permite usar las salidas analógicas del Industruino desde el panel frontal
- i) Borneras Electroválvula 1: Permite usar la electroválvula 1 de manera local, manual o ingresando una señal externa
- j) Borneras Electroválvula 2: Permite usar la electroválvula 2 de manera local, manual o ingresando una señal externa

- k) Switch 3: Permite seleccionar el método de entrada para la electro-válvula 1 de manera local o de una señal externa
- l) Switch 4 Permite seleccionar el método de entrada para el electro-válvula 2 de manera local o de una señal externa
- m) Potenciómetro 1: Permite controlar el electro-válvula 1 desde el panel frontal de forma manual.
- n) Potenciómetro 2: Permite controlar el electro-válvula 1 desde el panel frontal de forma manual.

Figura. 26

Diseño de panel frontal

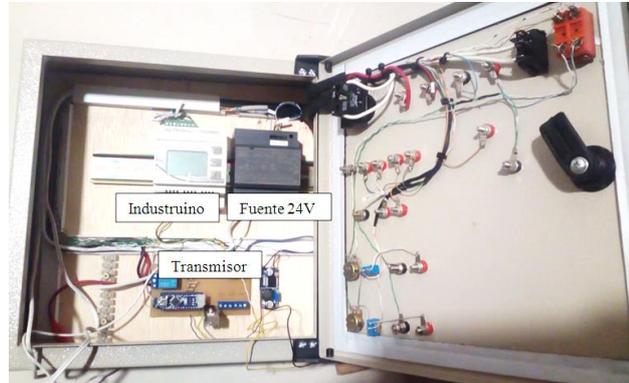


Nota: El gráfico nos muestra la distribución de elementos en el panel frontal, también los nombres de los diferentes puntos de prueba en el módulo PCT 3.

Una vez diseñado el panel frontal procedemos a montar los elementos, estos son distribuidos de manera estratégica para evitar problemas de espacio y manipulación. También realizamos el conexionado interno del panel soldando los conductores con los elementos correspondientes.

Figura. 27

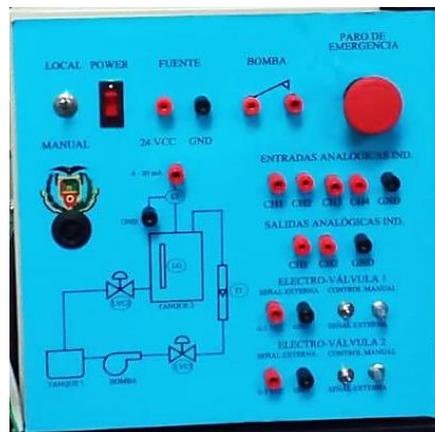
Panel frontal ensamblado (vista interna)



Nota: El gráfico nos muestra la ubicación de los elementos principales del panel frontal.

Figura. 28

Panel frontal ensamblado (vista externa)



Nota: El gráfico nos muestra la vista del panel frontal cerrado y colocado los elementos.

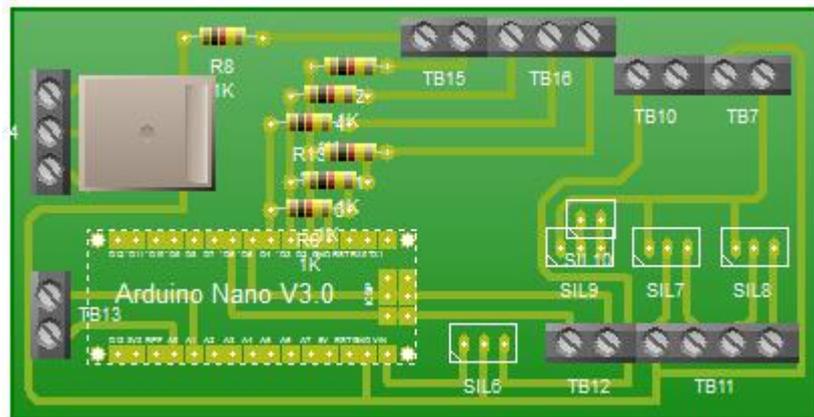
3.5 Transmisor Arduino

Para comunicar el panel frontal e Industruino con los actuadores (electro-válvulas) se diseñó un transmisor, el cual mediante un arduino Nano transforma la señal eléctrica recibida en una señal PWM, ya que el electro-válvula consta de un servomotor y este solo admite una señal PWM para moverse.

En la transformación se toma una señal eléctrica entre 0 – 5 voltios de energía continua y mediante la función mapeo se establece a 0 voltios como 0 grados y 5 voltios como 90 grados incluyendo todos los valores intermedios, además se establece una relación lineal entre el voltaje y la posición. (Véase el diagrama en el anexo B).

Figura. 29

Diseño de transmisor Arduino



Nota: El gráfico muestra el diseño del transmisor arduino Nano.

A continuación, se describe la programación del transmisor arduino Nano:

`#include <Servo.h>`: Librería para control de servomotor

`Servo válvula 1`: Define el objeto (válvula 1) que controla la función servo

`Servo válvula 2`: Define el objeto (válvula 2) que controla la función servo

`int potenciómetro = 0.5`: Variable para almacenar la posición del servo (válvula 1)

`int valpot1`: Variable para almacenar el voltaje de entrada del panel frontal o Industruino (válvula 1)

`int potnciomtro2 = 1`: Variable para almacenar la posición del servo (válvula 2)

`int valpot2`: Variable para almacenar el voltaje de entrada del panel frontal o Industruino (válvula 2)

```

void setup() {

    valvula1.attach(5): Define el pin de salida para la Señal PWM (válvula 1)

    valvula2.attach(6): Define el pin de salida para la Señal PWM (válvula 1)

}

void loop() {

    valpot1= analogRead(potnciomtro1): Lee y almacena el valor de entrada

    valpot1= map (valpot1, 0, 1023, 0, 90): Mapea el valor de entrada

    valvula2.write(valpot2): envía al servo a la posición definida por la variable

    delay(10 );

    valpot2= analogRead(potnciomtro2);

    valpot2= map (valpot2, 0, 1023, 0, 90);

    valvula1.write(valpot1):

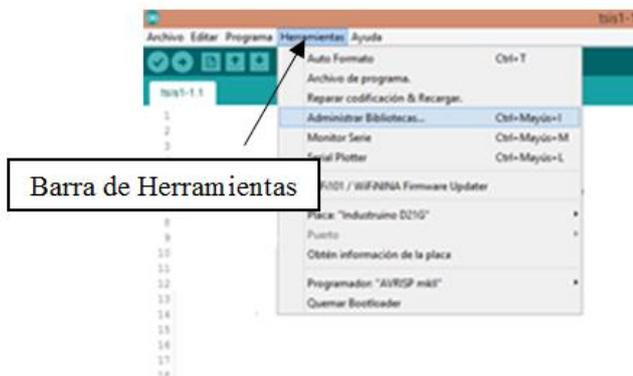
    delay(10 ):

}

```

3.6 Programación en la plataforma de arduino Ind.

Abrimos el entorno de arduino vamos a la barra de herramientas, administrar Bibliotecas y añadimos las librerías Fram by Industruino, Indio I/O, UC 1701 Industruino 128*64 LCD las cuales se descargan automáticamente al dar en la opción instalar.

Figura. 30*Entorno Arduino*

Nota: El gráfico muestra la barra de herramientas en el software Arduino.

Aparecerá una ventana del gestor de librerías y en la barra de búsqueda de esta colocamos las palabras “indus” de esta manera se visualizarán las librerías necesarias para el funcionamiento del control, instalamos las librerías y podemos continuar con la programación del Industruino.

Nota: es necesario que el software Arduino ID sea descargado de la página oficial de Arduino para poder descargar las librerías.

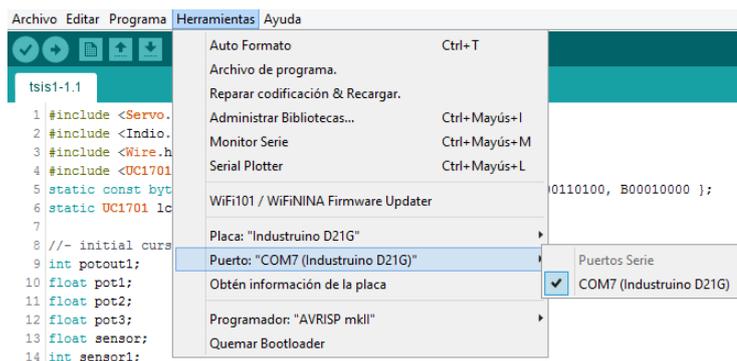
Figura. 31*Gestor de librerías en Arduino*

Nota: El gráfico muestra el gesto de librerías en el software Arduino.

Para instalar los controladores del Industruino basta con conectar el Industruino al computador mientras esté conectado al internet, el computador reconocerá la placa y asigna el puerto correspondiente. Esto podemos verificarlo en la barra de herramientas

Figura. 32

Entorno de Arduino



Nota: El gráfico muestra el puerto asignado al módulo Industruino en el software Arduino.

Instaladas las librerías y controladores procedemos a programar en el entorno de Arduino. La primera parte de la programación incluye las librerías necesarias y se declara una constante por defecto que permite navegar por la pantalla del Industruino.

`#include <Indio.h>`: permite controlar el Industruino

`#include <Wire.h>`: permite reconocer los puertos del Industruino

`#include <UC1701.h>`: permite utilizar la pantalla del Industruino

Se declara las variables a usar:

`static const byte glyph[] = { B00010000, B00110100, B00110000, B00110100, B00010000 }`: variables de escala grafica

`float pot1`: variable tipo flotante para el potenciómetro

`float pot2`: variable tipo flotante para el potenciómetro

`float pot3`: variable tipo flotante para el potenciómetro

`float sensor`: variable tipo flotante para sensor

`int sensor1`: variable tipo entero para sensor

`int sensor0`: variable tipo entero para sensor

`int valpot0`: variable tipo entero para potenciómetro

`int valpot1`: variable tipo entero para potenciómetro

`int valpot2`: variable tipo entero para potenciómetro

`int valpot3`: variable tipo entero para potenciómetro

`int potout1`: variable tipo entero para el potenciómetro

`float potout2`: variable tipo flotante para potenciómetro

`float potout3`: variable tipo flotante para el potenciómetro

`float nivl`: variable tipo flotante para almacenar nivel

`int coll = 0`: variable que determina el cursor - siempre se mantiene en 0 (lado izquierdo de la pantalla)

Nota: No todas las variables están definidas en esta sección pues cada subprograma consciente sus propias variables

Se utilizó parámetros para seleccionar diferentes niveles dentro del Industruino, esto nos ayuda a configurar los menús y submenús dentro de la programación.

`int channel = 0`: Contador se controla con los botones arriba y abajo del panel de membrana.

`int lastChannel = 0`: Variable que mantiene la pista del "canal" anterior. Se usa para detectar el cambio de estado.

Parámetros del nivel de menú inicial

`int MenuLevel = 0`: Variable que define la profundidad del árbol del menú

`int MenuID = 0`: Variable que define el identificador único de cada menú que reside en el mismo nivel de menú

`int channelUpLimit = 5`: Define el límite superior del contador de botones.

`int channelLowLimit = 0`: Define el límite inferior del contador de botones.

Parámetros iniciales para el "modo de edición de valores".

`int valueEditing = 0`: Flag para indicar si la interfaz está en "modo de edición de valores", desactivando así el movimiento del cursor.

`int row = 0`: Valor temporal para almacenar la posición actual del cursor mientras está en modo de edición de valores.

`int constrainEnc = 1`: Habilita o deshabilita la limitación del contador de la botonera a un límite inferior y superior.

`float valueEditingInc = 0`: Incremento de cada botón pulsado cuando se utiliza el 'modo de edición de valores'.

`float TargetValue = 0`: Es el valor objetivo a ser editado en el 'modo de edición de valores'.

El botón del panel de membrana define

`int buttonUpState = 0`: Define el estado de la entrada del botón "UP".

botón `intEnterState = 0`: Define el estado de la entrada del botón "Enter".

`int buttonDownState = 0`: Define el estado de la entrada del botón "Down".

`int lastBtnUp = 0`: Define el tiempo desde la última presión del botón "Up"

`int lastBtnEnt = 0`: Define el tiempo desde la última presión del botón "Enter"

`int lastBtnDown = 0`: Define el tiempo desde la última presión del botón "Down"

`int enterPresionado = 0`: Es el estado del botón "Enter" después del filtrado de rebotes: 1 = presionado 0 = sin presionar

`AdminActionTime = 0`: Mantiene un registro de la última actividad del botón

`int ButtonsAnalogValue = 0`: Valor leído de los botones del panel de membrana.

`int backlightIntensityDef = 5`: intensidad de la luz de fondo del LCD

`unsigned long lastLCDredraw = 0`: Lleva la cuenta de la última vez que la pantalla fue redibujada

Al iniciar el programa principal, existen diferentes opciones organizadas en forma de menú y submenús los cuales veremos a continuación.

Incluimos los puertos de potencia del Industruino y los designamos a medir Tensión o Corriente Eléctrica

```
void setup() {
  Indio.analogWriteMode(1, mA);
  Indio.analogWriteMode(2, mA);
  Indio.analogWrite(1, 0, true);
  Indio.analogWrite(2, 0, true);
}
```

Programación de los pines de control de Industruino como entradas o salidas

```
pinMode(buttonEnterPin, INPUT);
pinMode(buttonUpPin, INPUT);
pinMode(buttonDownPin, INPUT);
pinMode(backlightPin, OUTPUT);

analogWrite(backlightPin, (map(backlightIntensity, 5, 1, 255, 0))): controla la
intensidad de fondo del LCD
```

Inicializa el LCD en el modulo Arduino IND.

`lcd.begin()`: Establece la resolución de la pantalla LCD

```
for (int y = 0; y <= 7; y++) {
  for (int x = 0; x <= 128; x++) {
    lcd.setCursor(x, y);
    lcd.print(" ");
  }
}
```

`//debug`

`Serial.begin(9600)`: Habilita el puerto serie para la depuración de mensajes

Se inicializa y cargan los menús, de acuerdo al árbol de profundidad se van mostrando los menús de menor a mayor

```
void loop() {
```

`ReadButtons()`: Chequea los botones del panel frontal de Industruino

`Navigate()`: Actualiza los menús y realizar acciones

```
delay(50);
```

```
}
```

Se definen los parámetros del menú “Welcome”. Este es el primer menú a visualizarse

```
void MenuWelcome() {
```

`channel = 0`: Dirige la posición del cursor en la fila de inicio (fila superior) - controlado por el contador de la botonera

`channelUpLimit = 0`: Límite superior de la fila

`channelLowLimit = 0`: Límite inferior de la fila

`MenuLevel = 0`: Menú árbol de profundidad -> primer nivel

`MenuID = 0`: Id único de menú -> tiene que ser único para cada menú en el mismo nivel de menú.

`enterPressed = 0`: borra cualquier posible pulsación accidental de "Enter" que podría haber sido arrastrada desde el menú anterior

`lcd.clear()`: Limpia el LCD

Contenido que se visualiza el usuario en la pantalla

`lcd.setCursor(5, 0)`: Ubica el cursor al pixel necesario.

`lcd.print(" Universidad de las ")`: Imprime el texto en el LCD

`lcd.setCursor(5, 1)`: Ubica el cursor al pixel necesario.

`lcd.print(" Fuerzas Armadas")`: Imprime el texto en el LCD

`lcd.setCursor(5, 3)`: Ubica el cursor al pixel necesario.

`lcd.print(" Industruino")`: Imprime el texto en el LCD

`lcd.setCursor(5, 5)`: Ubica el cursor al pixel necesario.

`lcd.print(" Carlos Jimenez")`: Imprime el texto en el LCD

`lcd.setCursor(5, 6)`: Ubica el cursor al pixel necesario.

`lcd.print(" PCT 3 ")`: Imprime el texto en el LCD

`delay(5000);`

}

Se define los parámetros del sub-menú "MenuSelect", el mismo que controla las funciones indispensables para ingresar a diferentes funciones del módulo

```
void MenuSelect() { :
```

```
    Indio.digitalWrite(1,LOW):
```

```
    Indio.digitalWrite(2,LOW):
```

```
    Indio.digitalWrite(3,LOW):
```

```
    channel = 3:
```

```
    channelLowLimit = 3:
```

```
    channelUpLimit = 5:
```

```
    MenuLevel = 1:
```

```
    MenuID = 1:
```

```
    enterPressed = 0:
```

```
    lcd.clear(); limpia la pantalla
```

```
    ScrollCursor(): Habilitar el cursor móvil
```

```
Contenido que se visualiza el usuario en la pantalla
```

```
    lcd.setCursor(6, 0):
```

```
    lcd.print("Control de Nivel");
```

```
    lcd.setCursor(6, 1):
```

```
    lcd.print("Baseboard type:");
```

```
    lcd.setCursor(6, 3):
```

```
    lcd.print("AUTO");
```

```
    lcd.setCursor(6, 4):
```

```
    lcd.print("DEMO");
```

```

lcd.setCursor(6, 5):

lcd.print("SCREEN"):

}

```

Se define los parámetros del menú "MenuMain"

```

void MenuMain() { :

channel = 1:

channelUpLimit = 2:

MenuLevel = 1: Menú profundidad del árbol -> segundo nivel

MenuID = 1:

enterPressed = 0:

lcd.clear(); Limpia la pantalla

ScrollCursor():

Contenido que se visualiza el usuario en la pantalla

lcd.setCursor(6, 0):

lcd.print("Setup"):

lcd.setCursor(6, 1):

lcd.print("Demo"):

lcd.setCursor(6, 2):

lcd.print("Back"):

}

```

Se define los parámetros del menú “MenuSetup”, para controlar la intensidad de luz del LCD

```
void MenuSetup() {  
  
    channel = 0;  
  
    channelUpLimit = 2;  
  
    channelLowLimit = 0;  
  
    MenuID = 9;  
  
    MenuLevel = 3;  
  
    enterPressed = 0;  
  
    lcd.clear();  
  
    ScrollCursor();  
  
    lcd.setCursor(6, 0);  
  
    lcd.print("BackLight  ");  
  
    lcd.setCursor(65, 0);  
  
    lcd.print(backlightIntensity, 1);  
  
    lcd.setCursor(6, 1);  
  
    lcd.print("Reset param.");  
  
    lcd.setCursor(6, 2);  
  
    lcd.print("Back");  
  
}
```

Se define los parámetros del menú “MenuParametersReset”, para resetear todas las funciones básicas del módulo Arduino IND.

```
void MenuParametersReset() {  
  
    channel = 6;  
  
    channelUpLimit = 5;  
  
    channelLowLimit = 4;  
  
    MenuID = 10;  
  
    MenuLevel = 3;  
  
    enterPressed = 0;  
  
    lcd.clear();  
  
    ScrollCursor();  
  
    lcd.setCursor(6, 0);  
  
    lcd.print("Set system");  
  
    lcd.setCursor(6, 1);  
  
    lcd.print("to default");  
  
    lcd.setCursor(6, 2);  
  
    lcd.print("settings?");  
  
    lcd.setCursor(6, 4);  
  
    lcd.print("OK?");  
  
    lcd.setCursor(6, 5);  
  
    lcd.print("Cancel");  
  
}
```

Se define los parámetros del menú “MenuDemoProto”, este sub-menú activa las funciones demostrativas del control en Industruino.

```
void MenuDemoProto() {  
  
    Indio.digitalWrite(1,LOW);  
  
    Indio.digitalWrite(2,LOW);  
  
    Indio.digitalWrite(3,LOW);  
  
    channel = 0;  
  
    channelUpLimit = 4;  
  
    channelLowLimit = 0;  
  
    MenuID = 3;  
  
    MenuLevel = 2;  
  
    enterPressed = 0;  
  
    lcd.clear();  
  
    ScrollCursor();  
  
    lcd.setCursor(6, 0);  
  
    lcd.print("Bomba");  
  
    lcd.setCursor(6, 1);  
  
    lcd.print("Electrovalvula 1");  
  
    lcd.setCursor(6, 2);  
  
    lcd.print("Electrovalvula 2");  
  
    lcd.setCursor(6, 3);  
}
```

```
lcd.print("Sensor Nivel");  
  
lcd.setCursor(6, 4);  
  
lcd.print("Back");  
  
}
```

Se define los parámetros del menú “MenuInpot1”, para poder adquirir datos de voltaje desde los puertos análogos además de activarlos y desactivarlos

```
void MenuInpot1() {  
  
  Indio.setADCResolution(14);  
  
  Indio.digitalWrite(1,LOW);  
  
  Indio.digitalWrite(2,HIGH);  
  
  Indio.digitalWrite(3,LOW);  
  
  Indio.analogReadMode(3, V10);  
  
  Indio.analogReadMode(4, V10);  
  
  Indio.analogWriteMode(1, V10);  
  
  Indio.analogWriteMode(2, false);  
  
  channel = 4;  
  
  channelUpLimit = 4;  
  
  channelLowLimit = 4;  
  
  MenuID = 23;  
  
  MenuLevel = 3;  
  
  enterPressed = 0;
```

```

lcd.clear();

ScrollCursor();

MenuInpot1Live();

}

```

Se define los parámetros del menú “MenuInpot1Live”, para controlar la electroválvula1 por medio de voltaje desde el panel frontal.

```

void MenuInpot1Live() {

Indio.digitalWrite(1,LOW);

Indio.digitalWrite(2,HIGH);

Indio.digitalWrite(3,LOW);

lcd.setCursor(6, 0);

lcd.print(" Valvula 1 ");

lcd.setCursor(6, 1);

lcd.print("CH3      ");

pot2 = Indio.analogRead(3);

lcd.setCursor(35, 1);

lcd.print(pot2);

Indio.analogWrite(1, pot2,true);

Indio.analogWrite(2, pot3,false);

lcd.setCursor(67, 1);

lcd.print("V");

```

```
lcd.setCursor(6, 4);  
  
lcd.print("Back  ");  
  
}
```

Se define los parámetros del menú “MenuInpot2”, para poder adquirir datos de voltaje desde los puertos análogos además de activarlos y desactivarlos

```
void MenuInpot2() {  
  
    pot2=0;  
  
    Indio.setADCResolution(14);  
  
    Indio.digitalWrite(1,LOW);  
  
    Indio.digitalWrite(2,LOW);  
  
    Indio.digitalWrite(3,HIGH);  
  
    Indio.analogReadMode(3, V10);  
  
    Indio.analogReadMode(4, V10);  
  
    Indio.analogWriteMode(2, V10);  
  
    Indio.analogWriteMode(1, false);  
  
    channel = 4;  
  
    channelUpLimit = 4;  
  
    channelLowLimit = 4;  
  
    MenuID = 24;  
  
    MenuLevel = 3;  
  
    enterPressed = 0;
```

```

lcd.clear();

ScrollCursor();

MenuInpot1Live();

}

```

Se define los parámetros del menú “MenuInpot1Live”, para controlar la electroválvula1 por medio de voltaje desde el panel frontal.

```

void MenuInpot2Live() {

pot3=0;

Indio.digitalWrite(1,LOW);

Indio.digitalWrite(2,LOW);

Indio.digitalWrite(3,HIGH);

lcd.setCursor(6, 0);

lcd.print(" Valvula 2 ");

lcd.setCursor(6, 1);

lcd.print("CH4      ");

pot3 = Indio.analogRead(4);

lcd.setCursor(35, 1);

lcd.print(Indio.analogRead(4),2);

Indio.analogWrite(2, pot3,true);

Indio.analogWrite(1, pot2,false);

lcd.setCursor(67, 1);

```

```
lcd.print("V");  
  
lcd.setCursor(6, 4);  
  
lcd.print("Back ");  
  
}
```

Se define los parámetros del primer sub-menú “MenuDemoInd”, para ingresar a diferentes funciones demostrativas del módulo permitiendo comprobar las señales básicas que recibe.

Esta función es vista por el usuario como el ítem “DEMO” en el LCD del Industruino,

```
void MenuDemoInd() {  
  
  Indio.digitalMode(1, OUTPUT);  
  
  Indio.digitalWrite(1,LOW);  
  
  Indio.analogWrite(1, 5, true);  
  
  Indio.analogWrite(2, 5, true);  
  
  channel = 2;  
  
  channelUpLimit = 7;  
  
  channelLowLimit = 2;  
  
  MenuID = 11;  
  
  MenuLevel = 2;  
  
  enterPressed = 0;  
  
  lcd.clear();  
  
  ScrollCursor();  
  
}
```

```
lcd.setCursor(6, 0);

lcd.print(" Control de Nivel");

lcd.setCursor(6, 2);

lcd.print("Set Point ");

lcd.setCursor(6, 3);

lcd.print("Sensor Nivel");

lcd.setCursor(6, 4);

lcd.print("Valvulas");

lcd.setCursor(6, 5);

lcd.print("Control Auto");

lcd.setCursor(6, 6);

lcd.print("LCD backlight");

lcd.setCursor(6, 7);

lcd.print("Back");

}
```

Se define los parámetros del sub-menú “MenuDigitalOutIndBomba”, para realizar un control ON-OFF de la bomba. El usuario verifica esto a través del mensaje encendido o apagado en el LCD

```
void MenuDigitalOutIndBomba() {

  for (int i = 1; i <= 8; i++) {

    Indio.digitalMode(i, OUTPUT);

    Indio.digitalWrite(i, LOW);
```

```
}  
  
channel = 2;  
  
channelUpLimit = 5;  
  
channelLowLimit = 2;  
  
MenuID = 21;  
  
MenuLevel = 3;  
  
enterPressed = 0;  
  
lcd.clear();  
  
ScrollCursor();  
  
lcd.setCursor(6, 2);  
  
lcd.print("Encendido");  
  
lcd.setCursor(6, 3);  
  
lcd.print("Apagado");  
  
lcd.setCursor(6, 5);  
  
lcd.print("Back");  
  
}
```

Se define los parámetros del sub-menú “MenuAnaloginValvulas”, para realizar la toma de datos de voltaje a través de los puertos análogos los cuales controlan las válvulas 1 y 2

```
void MenuAnaloginValvulas() {  
  
    Indio.setADCResolution(14);  
  
    Indio.analogReadMode(3, V10);
```

```

Indio.analogReadMode(4, V10);

channel = 4;

channelUpLimit = 4;

channelLowLimit = 4;

MenuID = 17;

MenuLevel = 3;

enterPressed = 0;

lcd.clear();

ScrollCursor();

MenuAnaloginValvulasLive();

}

```

Se define los parámetros del sub-menú “MenuAnaloginValvulasLive”, este ocupa los datos tomados por el menú del mismo nombre más la palabra “Live”

```

void MenuAnaloginValvulasLive() {

Indio.digitalWrite(1,LOW);

lcd.setCursor(6, 0);

lcd.print(" Valvula 1 ");

lcd.setCursor(6, 1);

lcd.print("CH3      ");

lcd.setCursor(35, 1);

lcd.print(Indio.analogRead(3));

```

```

lcd.setCursor(67, 1);

lcd.print("V");

lcd.setCursor(6, 2);

lcd.print("  Valvula 2 ");

lcd.setCursor(6, 3);

lcd.print("CH4      ");

lcd.setCursor(35, 3);

lcd.print(Indio.analogRead(4));

lcd.setCursor(67, 3);

lcd.print("V");

lcd.setCursor(6, 4);

lcd.print("Back  ");

}

```

Se define los parámetros del sub-menú “MenuAnalogIn20mAInd”, para realizar la toma de datos de corriente a través de los puertos análogos

```

void MenuAnalogIn20mAInd() {

  Indio.setADCResolution(14);

  Indio.analogReadMode(1, mA);

  channel = 4;

  channelUpLimit = 4;

  channelLowLimit = 4;

```

```

MenuID = 18;

MenuLevel = 3;

enterPressed = 0;

lcd.clear();

ScrollCursor();

MenuAnalogIn20mALiveInd();

}

```

Se define los parámetros del sub-menú “MenuAnalogInValvulasLive”, este ocupa los datos tomados por el menú del mismo nombre más la palabra “Live”

```

void MenuAnalogIn20mALiveInd() {

    lcd.setCursor(6, 0);

    lcd.print("Sensor Ultrasonico");

    lcd.setCursor(6, 1);

    lcd.print(" TEACH 4 - 20 mA ");

    lcd.setCursor(6, 3);

    lcd.print("CH1      ");

    lcd.setCursor(35, 3);

    lcd.print(Indio.analogRead(1));

    lcd.setCursor(67, 3);

    lcd.print("mA");

    lcd.setCursor(6, 4);

```

```

    lcd.print("Back ");
}

```

Se define los parámetros del sub-menú “MenuAnalogIn20mASensor”, para realizar la toma de datos de corriente del sensor ultrasónico TEACH a través de los puertos análogos

```

void MenuAnalogIn20mASensor() {

    Indio.setADCResolution(14);

    Indio.analogReadMode(1, mA);

    channel = 4;

    channelUpLimit = 4;

    channelLowLimit = 4;

    MenuID = 22;

    MenuLevel = 3;

    enterPressed = 0;

    lcd.clear();

    ScrollCursor();

    MenuAnalogIn20mALiveSensor();

}

```

Se define los parámetros del sub-menú “MenuAnalogIn20mALiveSensor”, este ocupa los datos tomados por el menú del mismo nombre más la palabra “Live”

```

void MenuAnalogIn20mALiveSensor() {

    lcd.setCursor(6, 0);

```

```

lcd.print("Sensor Ultrasonico");

lcd.setCursor(6, 1);

lcd.print(" TEACH 4 - 20 mA ");

lcd.setCursor(6, 3);

lcd.print("CH1      ");

lcd.setCursor(35, 3);

lcd.print(Indio.analogRead(1));

lcd.setCursor(67, 3);

lcd.print("mA");

lcd.setCursor(6, 4);

lcd.print("Back ");

}

```

Se define los parámetros del sub-menú “MenuAnalogIn10VInd”, para realizar la toma de datos de voltaje a través de los puertos análogos

```

void MenuAnalogIn10VInd() {

  Indio.setADCResolution(14);

  Indio.analogReadMode(2, V10);

  channel = 4;

  channelUpLimit = 4;

  channelLowLimit = 4;

  MenuID = 19;

```

```

MenuLevel = 3;

enterPressed = 0;

lcd.clear();

ScrollCursor();

MenuAnalogIn10VLiveInd();

}

```

Se define los parámetros del sub-menú “MenuAnalogIn10VLiveInd”, este ocupa los datos tomados por el menú del mismo nombre más la palabra “Live”

```

void MenuAnalogIn10VLiveInd() {

    lcd.setCursor(6, 0);

    lcd.print("Valor de Set Point");

    lcd.setCursor(6, 1);

    lcd.print("    0 - 5 ");

    lcd.setCursor(6, 3);

    lcd.print("CH2      ");

    lcd.setCursor(35, 3);

    lcd.print(Indio.analogRead(2));

    lcd.setCursor(67, 3);

    lcd.print("V");

    lcd.setCursor(6, 4);

    lcd.print("Back ");
}

```

```
}

```

Se define los parámetros del segundo sub-menú “MenuControlAuto”, para activar la función de control automático del módulo permitiendo visualizar las señales más relevantes que recibe.

Esta función es vista por el usuario como el ítem “CONTROL AUTO” en el LCD del Industruino,

```
void MenuControlAuto() {

    Indio.setADCResolution(14);

    Indio.digitalWrite(1,LOW);

    Indio.analogReadMode(1, mA);

    Indio.analogReadMode(2, V10);

    anOutUpLimit = 5.5;

    Indio.analogWriteMode(1, V10);

    Indio.analogWriteMode(2, V10);

    Indio.analogWrite(1, anOutCh1, false);

    Indio.analogWrite(2, anOutCh2, false);

    nivl=0;

    channel = 7;

    channelUpLimit = 7;

    channelLowLimit = 7;

    MenuID = 20;

    MenuLevel = 2;
}
```

```

enterPressed = 0;

lcd.clear();

ScrollCursor();

MenuControlAutoLiveInd();

}

```

Se define los parámetros del sub-menú “MenuControlAutoLiveInd”, este ocupa los datos tomados por el menú del mismo nombre más la palabra “Live”.

En esta sección específica se detallan los parámetros del control automático para utilizarse en el control fuzzy.

```

void MenuControlAutoLiveInd() {

Indio.digitalWrite(1,HIGH);

pot1 = Indio.analogRead(2);

valpot1= pot1*21.5;

sensor = Indio.analogRead(1);

sensor1= sensor*6;

sensor0= sensor*5;

valpot1 = map (valpot1, 0, 110, 0, 180);

sensor1 = map (sensor1, 25, 120, 0, 180);

valpot0 = pot1*19.50;

nivl = valpot1 - sensor1 ;

lcd.setCursor(6, 0);

lcd.print("SP      ");

```

```
lcd.setCursor(36, 0);  
  
lcd.print(valpot0);  
  
lcd.setCursor(6, 1);  
  
lcd.print("sensor ");  
  
lcd.setCursor(36, 1);  
  
lcd.print(sensor0);  
  
lcd.setCursor(6, 2);  
  
lcd.print("val2 ");  
  
lcd.setCursor(36, 2);  
  
lcd.print(valpot3);  
  
lcd.setCursor(6, 3);  
  
lcd.print("sen ");  
  
lcd.setCursor(36, 3);  
  
lcd.print(sensor1);  
  
lcd.setCursor(6, 4);  
  
lcd.print("NV: ");  
  
lcd.setCursor(36, 4);  
  
lcd.print(niv1);  
  
lcd.setCursor(6, 6);  
  
lcd.print("LCD ");  
  
lcd.setCursor(6, 7);
```

```
lcd.print("Back");
```

```
}
```

Submenú del menú principal - pantalla de configuración para Industruino

Esta sección detalla las acciones que realiza Industruino cuando se ingresa a un menú o sub-menú

```
void Navigate()
```

```
{
```

```
  if (valueEditing != 1) {
```

 if (MenuLevel == 0) comprueba si el menú activado actual es la "pantalla de salpicadura" (primer nivel)

```
    {
```

```
      {
```

 if (enterPressed == 1) MenuSelect(); Si se pulsa "enter", cargar el "Menú principal".

```
      }
```

```
    }
```

 if (MenuLevel == 1) { comprueba si el menú activado actual es el "Menú principal" (primer nivel)

 if (channel == 3 && enterPressed == 1) MenuDemoInd(); Si el cursor está en la fila 3 y se pulsa "enter", cargar el menú "Setup".

 if (channel == 4 && enterPressed == 1) MenuDemoProto(); Si el cursor está en la fila 4 y se presiona Enter, carga el menú "Demo".

 if (channel == 5 && enterPressed == 1) MenuWelcome(); Si el cursor está en la fila 5 y se pulsa "enter" carga la "pantalla de salpicadura"

```
}
```

`if (MenuLevel == 2) {` { comprueba si el menú activado actual es el "DEMO" (segundo nivel)

```
if (MenuID == 3) {
```

`if (channel == 0 && enterPressed == 1) MenuDigitalOutIndBomba();` Si el cursor está en la fila 0 y se pulsa "enter", cargar el " MenuDigitalOutIndBomba ".

`if (channel == 1 && enterPressed == 1) MenuInpot1();` Si el cursor está en la fila 1 y se pulsa "enter", cargar el " MenuInpot1", control de la electro-válvula 1

`if (channel == 2 && enterPressed == 1) MenuInpot2();` Si el cursor está en la fila 2 y se pulsa "enter", cargar el " MenuInpot2", control de electro-válvula 2

`if (channel == 3 && enterPressed == 1) MenuAnalogIn20mASensor();` Si el cursor está en la fila 3 y se pulsa "enter", cargar el " MenuAnalogIn20mASensor ", muestra la lectura del sensor TEACH

`if (channel == 4 && enterPressed == 1) MenuSelect();` Si el cursor está en la fila 4 y se pulsa "enter", cargar el " MenuSelect "

```
}
```

```
if (MenuID == 11) {
```

 si se encuentra en el menú "MenuDemoIndra" entonces:

`if (channel == 2 && enterPressed == 1) MenuAnalogIn10VInd();` Si el cursor está en la fila 2 y se pulsa "enter", cargar el " MenuAnalogIn10VInd " valor de set point

`if (channel == 3 && enterPressed == 1) MenuAnalogIn20mAInd();` Si el cursor está en la fila 3 y se pulsa "enter", cargar el " MenuAnalogIn20mAInd" valor de sensor

`if (channel == 4 && enterPressed == 1) MenuAnaloginValvulas();` Si el cursor está en la fila 4 y se pulsa "enter", cargar el " MenuAnaloginValvulas " (valvulas)

`if (channel == 5 && enterPressed == 1) MenuControlAuto();` Si el cursor está en la fila 5 y se pulsa "enter", cargar el "MenuControlAuto" control Automático FUZZY

`if (channel == 6 && enterPressed == 1) MenuSetup();` Si el cursor está en la fila 6 y se pulsa "enter", cargar el " MenuSetup "

`if (channel == 7 && enterPressed == 1) MenuSelect();` Si el cursor está en la fila 7 y se pulsa "enter", cargar el "MenuSelect"

}

Implementación de conjuntos difusos y funciones de pertenecía de control FUZZY

`if (MenuID == 20) {`

`if (nivl>=100 && nivl<200){ Indio.analogWrite(1, 2, true);`

`Indio.analogWrite(2, 5, true);`

`delay(5);`

`MenuControlAutoLiveInd();`

}

`if (nivl>=50 && nivl<99){ Indio.analogWrite(1, 3.5, true);`

`Indio.analogWrite(2, 3.5, true);`

`delay(5);`

`MenuControlAutoLiveInd();`

}

`if (nivl>=25 && nivl<49){ Indio.analogWrite(1, 3.75, true);`

`Indio.analogWrite(2, 3.25, true);`

`delay(5);`

```
MenuControlAutoLiveInd();  
  
}  
  
if (nivl>=10 && nivl<24){ Indio.analogWrite(1, 4, true);  
  
    Indio.analogWrite(2, 2.75, true);  
  
    delay(5);  
  
    MenuControlAutoLiveInd();  
  
}  
  
if (nivl>=3 && nivl<9){ Indio.analogWrite(1, 5, true);  
  
    Indio.analogWrite(2, 3.5, true);  
  
    delay(5);  
  
    MenuControlAutoLiveInd();  
  
}  
  
if (nivl>=-3 && nivl<2){ Indio.analogWrite(1, 5, true);  
  
    Indio.analogWrite(2, 2.25, true);  
  
    delay(5);  
  
    MenuControlAutoLiveInd();  
  
}  
  
if (nivl>=-10 && nivl<-4){ Indio.analogWrite(1, 5, true);  
  
    Indio.analogWrite(2, 2, true);  
  
    delay(5);  
  
    MenuControlAutoLiveInd();
```

```
    }  
  
    if (nivl>=-24 && nivl<-11){ Indio.analogWrite(1, 4, true);  
  
        Indio.analogWrite(2, 1.75, true);  
  
        delay(5);  
  
        MenuControlAutoLiveInd();  
  
    }  
  
    if (nivl>=-50 && nivl<-25){ Indio.analogWrite(1, 4, true);  
  
        Indio.analogWrite(2, 1.25, true);  
  
        delay(5);  
  
        MenuControlAutoLiveInd();  
  
    }  
  
    if (nivl>=-100 && nivl<-50){ Indio.analogWrite(1, 4, true);  
  
        Indio.analogWrite(2, 0.75, true);  
  
        delay(5);  
  
        MenuControlAutoLiveInd();  
  
    }  
  
    if (nivl>=-101 && nivl<-200){ Indio.analogWrite(1, 3.5, true);  
  
        Indio.analogWrite(2, 0.5, true);  
  
        delay(5);  
  
        MenuControlAutoLiveInd();  
  
    }
```

```
if ((millis() - lastLCDredraw) > 268) {
```

```
MenuControlAutoLiveInd();
```

```
lastLCDredraw = millis();
```

```
}
```

if (channel == 7 && enterPressed == 1) MenuDemoInd(); Si el cursor está en la fila 7 y se pulsa "enter", cargar el " MenuDemoInd "

```
}
```

```
}
```

if (MenuLevel == 3) { comprueba si el menú activado actual es el correcto (tercer nivel)

Menú de Control Manual de la Bomba

```
if (MenuID == 21) { MenuDigitalOutIndBomba
```

```
if ( buttonEnterState == LOW ) Bomba apagada
```

```
{
```

```
lcd.setCursor(0, channel);
```

```
lcd.print("*");
```

```
}
```

```
if ( buttonEnterState == HIGH ) Bomba encendida
```

```
{
```

```
lcd.setCursor(0, channel);
```

```
lcd.print(">");
```

```
}
```

`if (channel == 2 && buttonEnterState == LOW) Indio.digitalWrite(1, HIGH);` Si el cursor está en la fila 2 y se pulsa "enter", se activa el puerto digital 1 de potencia en el Industruino

`if (channel == 3 && buttonEnterState == LOW) Indio.digitalWrite(1, LOW);` Si el cursor está en la fila 3 y se pulsa "enter", se desactiva el puerto digital 1 de potencia en el Industruino

`if (channel == 5 && enterPressed == 1) MenuDemoProto();` Si el cursor está en la fila 5 y se pulsa "enter", carga el menú "MenuDemoProto"

}

`if (MenuID == 18) {`

`if ((millis() - lastLCDredraw) > 268) {`

`MenuAnalogIn20mALiveInd();`

`lastLCDredraw = millis();`

}

`if (channel == 4 && enterPressed == 1) MenuDemoInd();` Si el cursor está en la fila 4 y se pulsa "enter", carga el menú "MenuDemoInd"

}

`if (MenuID == 22) {`

`if ((millis() - lastLCDredraw) > 268) {`

`MenuAnalogIn20mALiveSensor();`

`lastLCDredraw = millis();`

}

`if (channel == 4 && enterPressed == 1) MenuDemoProto();` Si el cursor está en la fila 4 y se pulsa "enter", carga el menú "MenuDemoProto"

```
}

```

```
if (MenuID == 23) {

```

```
    if ((millis() - lastLCDredraw) > 268) {

```

```
        MenuInpot1Live();

```

```
        lastLCDredraw = millis();

```

```
    }

```

if (channel == 4 && enterPressed == 1) MenuDemoProto(); Si el cursor está en la fila 4 y se pulsa "enter", carga el menú "MenuDemoProto"

```
}

```

```
if (MenuID == 24) {

```

```
    if ((millis() - lastLCDredraw) > 268) {

```

```
        MenuInpot2Live();

```

```
        lastLCDredraw = millis();

```

```
    }

```

if (channel == 4 && enterPressed == 1) MenuDemoProto(); Si el cursor está en la fila 4 y se pulsa "enter", carga el menú "MenuDemoProto"

```
}

```

```
if (MenuID == 19) {

```

```
    if ((millis() - lastLCDredraw) > 268) {

```

```
        MenuAnalogIn10VLiveInd();

```

```
        lastLCDredraw = millis();

```

```
    }

```

`if (channel == 4 && enterPressed == 1) MenuDemoInd();` Si el cursor está en la fila 4 y se pulsa "enter", carga el menú "MenuDemoProto"

```
}
```

En esta sección carga cada menú ID y lo redirige a un sub-menú para realizar la función establecida por el mismo.

```
if (MenuID == 9) { Controla la Intensidad de luz del LCD
```

`if (channel == 0 && enterPressed == 1)` Si el cursor está en la fila 0 y se pulsa "enter", carga el menú "modo de edición de valores"

```
{
```

```
    TargetValue = backlightIntensity;
```

```
    backlightIntensity = EditValue();
```

```
    analogWrite(backlightPin, (map(backlightIntensity, 5, 0, 255, 0)));
```

```
}
```

```
if (channel == 1 && enterPressed == 1) MenuParametersReset();
```

```
if (channel == 2 && enterPressed == 1) MenuDemoInd();
```

```
}
```

```
if (MenuID == 10) { Parametro reset
```

```
    if (channel == 4 && enterPressed == 1) ResetParameters();
```

```
    if (channel == 5 && enterPressed == 1) MenuDemoInd();
```

```
}
```

```
if (MenuID == 17) { Menú de control de válvulas
```

```
    if ((millis() - lastLCDredraw) > 268) {
```

```
MenuAnaloginValvulasLive();
```

```
lastLCDredraw = millis();
```

```
}
```

```
if (channel == 4 && enterPressed == 1) MenuDemoInd();
```

```
}
```

```
}
```

if (channel != lastChannel && valueEditing != 1 && MenuID != 0) { actualiza la posición del cursor si el contador de botones ha cambiado y el 'modo de edición de valores' no está funcionando

```
ScrollCursor();
```

```
}
```

```
}
```

```
}
```

float EditValue(): Una función para editar una variable usando la UI - la función es llamada por la función principal de control de la UI "Navegar" y se carga con una variable para ser editada{

row = channel; Guarda la posición actual del cursor para que después de usar los botones para el 'modo de edición de valores' la posición del cursor pueda ser restablecida.
channel = 0; Resetea el contador de botones para evitar arrastrar un valor desde el cursor.

constrainEnc = 0; deshabilitar la limitación del alcance del contador de botones

valueEditingInc = 1; incremento por cada pulsación de botón

valueEditing = 1;

enterPressed = 0; Borra cualquier posible pulsación accidental de "Enter" que podría haber sido cargado sobre

`while (enterPressed != 1) {` permanece en el "modo de edición de valores" hasta que se pulsa la tecla intro

`ReadButtons();` comprueba los botones para cualquier cambio

`lcd.setCursor(0, row);`

`lcd.print("*");`

`if (channel != lastChannel) {` cambia el estado del canal cuando se presiona un botón

`if (channel < lastChannel && TargetValue <= 4) {` si se presiona el botón 'Arriba', y está dentro del rango de restricción.

`TargetValue += valueEditingInc;` variable de incremento con valor de incremento predefinido

`}`

`if (channel > lastChannel && TargetValue > 0) {` si se pulsa el botón "Abajo", y está dentro del rango de restricción.

`TargetValue -= valueEditingInc ;` variable objetivo de decrecimiento con valor de incremento predefinido

`}`

Despeja una sección de una fila para hacer espacio para el valor actualizado

`for (int i = 60; i <= 70; i++) {`

`lcd.setCursor(i, row);`

`lcd.print(" ");`

`}`

Imprimir el valor actualizado

```

    lcd.setCursor(66, row);

    Serial.println(TargetValue);

    lcd.print(TargetValue, 0);

    lastChannel = channel;

}

//delay(50);

}

```

`channel = row;` cargar de nuevo la posición de la fila anterior al contador de botones para que el cursor permanezca en la misma posición que se dejó antes de cambiar al 'modo de edición de valores'.

`constrainEnc = 1;` habilitar la restricción del rango del contador de botones para mantenerse dentro del rango del menú

`channelUpLimit = 2;` límite superior de la fila

`valueEditing = 0;` para indicar que estamos dejando el "modo de edición de valores"

`enterPressed = 0;` borra cualquier posible pulsación accidental de "Enter" que podría haber sido cargado sobre

`return TargetValue;` devuelve el valor editado a la función principal de control de la interfaz de usuario 'Navegar' para su posterior procesamiento

```

}

```

`float EditFloatValue()` Una función para editar una variable usando la UI - la función es llamada por la función principal de control de la UI "Navegar" y se carga con una variable para ser editada

```

{

```

`row = channel;` guardar la posición actual del cursor para que después de usar los botones para el "modo de edición de valores" la posición del cursor pueda ser restablecida.

`channel = 0;` resetea el contador de botones para evitar arrastrar un valor desde el cursor.

`constrainEnc = 0;` limitación del alcance del contador de botones

`valueEditingInc = 0.5;` Incremento por cada pulsación de botón

`valueEditing = 1;` para indicar que vamos a entrar en el "modo de edición de valores."

`enterPressed = 0;` borra cualquier posible pulsación accidental de "Enter" que podría haber sido cargado sobre

`while (enterPressed != 1) {` permanece en el "modo de edición de valores" hasta que se pulsa la tecla intro

`ReadButtons();` comprueba los botones para cualquier cambio

`lcd.setCursor(0, row);`

`lcd.print("*");`

`if (channel != lastChannel) {` si se presiona el botón de arriba o abajo entonces

`if (channel < lastChannel && TargetValue <= anOutUpLimit) {` si se presiona el botón 'Arriba', y está dentro del rango de restricción.

`TargetValue += valueEditingInc;` variable de incremento con valor de incremento predefinido

`}`

`if (channel > lastChannel && TargetValue > 0) {` si se pulsa el botón "Abajo", y está dentro del rango de restricción.

`TargetValue -= valueEditingInc ;` variable objetivo de decrecimiento con valor de incremento predefinido

```
}
```

Borrando una sección de una fila para hacer espacio para el valor actualizado

```
for (int i = 35; i <= 50; i++) {
```

```
  lcd.setCursor(i, row);
```

```
  lcd.print(" ");
```

```
}
```

```
//print updated value
```

```
lcd.setCursor(35, row);
```

```
Serial.println(TargetValue);
```

```
lcd.print(TargetValue, 2);
```

```
lastChannel = channel;
```

```
}
```

```
//delay(50);
```

```
}
```

`channel = row;` cargar de nuevo la posición de la fila anterior al contador de botones para que el cursor permanezca en la misma posición que se dejó antes de cambiar al 'modo de edición de valores'.

`constrainEnc = 1;` habilitar la restricción del rango del contador de botones para mantenerse dentro del rango del menú

`channelUpLimit = 2;` límite de la fila superior

`valueEditing = 0;` para indicar que estamos dejando el "modo de edición de valores

`enterPressed = 0;` borra cualquier posible pulsación accidental de "Enter" que podría haber sido cargado sobre

`return TargetValue;` devuelve el valor editado a la función principal de control de la interfaz de usuario 'Navegar' para su posterior procesamiento

```
}
```

Funciones periféricas

```
void ReadButtons() {

    buttonEnterState = digitalRead(buttonEnterPin);

    buttonUpState = digitalRead(buttonUpPin);

    buttonDownState = digitalRead(buttonDownPin);

    if (buttonEnterState == HIGH && prevBtnEnt == LOW)

    {

        if ((millis() - lastBtnEnt) > transEntInt)

        {

            enterPressed = 1;

        }

        lastBtnEnt = millis();

        lastAdminActionTime = millis();

        Serial.println(enterPressed);

    }

}
```

```
prevBtnEnt = buttonEnterState;

if (buttonUpState == HIGH && prevBtnUp == LOW)

{

    if ((millis() - lastBtnUp) > transInt)

    {

        channel--;

    }

    lastBtnUp = millis();

    lastAdminActionTime = millis();

    //Serial.println("UpPressed");

}

prevBtnUp = buttonUpState;

if (buttonDownState == HIGH && prevBtnDown == LOW)

{

    if ((millis() - lastBtnDown) > transInt)

    {

        channel++;

    }

    lastBtnDown = millis();

    lastAdminActionTime = millis();

    //Serial.println("DownPressed");
```

```

}

prevBtnDown = buttonDownState;

if (constrainEnc == 1) {

    channel = constrain(channel, channelLowLimit, channelUpLimit);

}

}

```

`void ResetParameters()` { restablece los parámetros de configuración de Industruino y guarda los ajustes en la EEPROM

`backlightIntensity = backlightIntensityDef;` cargar el valor predeterminado de la intensidad de la luz de fondo

`analogWrite(backlightPin, (map(backlightIntensity, 5, 0, 255, 0)));` mapear el valor (de 0-5) a un valor PWM correspondiente (0-255) y actualizar la salida

`MenuSetup();` volver al menú de configuración

```

}

```

Funciones básicas de la interfaz de usuario

`void ScrollCursor();` hace que el cursor se mueva

```

{

```

`lastChannel = channel;` mantiene el seguimiento de los cambios en el contador del botón

`for (int i = 0; i <= 6; i++) {` borrar toda la columna al redibujar un nuevo cursor

```
lcd.setCursor(coll, i);  
  
lcd.print(" ");  
  
}  
  
lcd.setCursor(coll, channel); establecer una nueva posición del cursor  
  
lcd.print(">"); // dibujar el cursor  
  
}
```

3.7 Montaje final de módulo de nivel PCT-3

Para colocar el panel frontal en el módulo PCT-3, de acuerdo a las dimensiones se verifica el espacio para las conexiones. Una vez revisado se procede a la instalación, fijamos el panel a la base del módulo con tornillos auto perforantes de manera cuidadosa para no dañar el conexionado, seguidamente se conecta la bomba, fuente de la bomba, sensor Teach y electro-válvulas al panel. (Véase en el anexo C)

Figura. 33

Módulo completo con el panel abierto



Nota: El gráfico muestra el módulo PCT 3 completamente instalado con el panel frontal abierto.

Figura. 34

Módulo completo con el panel cerrado



Nota: El gráfico muestra el módulo PCT 3 completo con el panel frontal cerrado.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

Mediante el controlador Industruino se logró Automatizar el módulo didáctico de nivel PCT 3 con la adquisición y transmisión de diferentes tipos de datos digitales y analógicos como por ejemplo se utilizó la señal analógica 4-20 mA para el sensor Teach, la señal 0-5 Vcc para el Set-Point, obteniendo así una estación de nivel mucho más pequeña con grandes prestaciones.

Al fusificar las variables de entrada se recibe datos de diferentes magnitudes (corriente, voltaje) las cuales no pueden relacionarse directamente, mediante Industruino podemos mapear estas entradas obteniendo así un valor cuantitativo proporcional para que al establecer los grados de pertenencia exista una clara relación entre los valores independientemente de su magnitud y de esta manera facilitar la implementación del control de nivel.

Al realizar las electro-válvulas didácticas se fabricó un acople mediante una impresora 3D para mover una válvula manual tipo bola mediante un servo-motor de 25 kg de torque, este acople se colocó verticalmente entre el eje del servomotor y el eje de la válvula manual. El servomotor se controla mediante PWM con lo cual se obtiene motricidad fina al momento de abrir la electro-válvula.

Debemos considerar que cada variable de entrada (sensor, Set-Point), no presentan las mismas magnitudes ni escalas por lo cual mapear los datos cuantitativos es de suma importancia, solo esta manera podemos realizar operaciones algebraicas dentro del software arduino para el control de los actuadores (electro-válvulas, bomba).

Existe en la actualidad una un sin número de aplicaciones mediante control fuzzy poco exploradas como la inteligencia artificial, aplicaciones automotrices, plantas químicas, etc., por lo cual debemos mantenernos abiertos a fallos de cálculos, hay que recordar que el control fuzzy se basa en la experiencia de un operario más no en cálculos pre programados como en el resto de sistemas de control.

REFERENCIAS BIBLIOGRÁFICAS

- Alegsa , L. (2016, 10 16). *Alegsa*. Retrieved 04 16, 2020, from Sistema de control:
http://www.alegsa.com.ar/Dic/sistema_de_control.php
- Andira. (2019, 06 19). *Asociación Nacional de Distribuidores de la Industria de la Refrigeración y Aire Acondicionado A.C.* Retrieved 02 01, 2020, from Rotámetro: <https://www.andira.org.mx/2017/03/22/como-funciona-el-rotometro/>
- Arduino. (2005). *Arduino.cc*. Retrieved 01 10, 2020, from Arduino-Home:
<https://www.arduino.cc/>
- Avnet Company. (2019). Retrieved Enero 6, 2020, from mexico.newark.com
- Brio, B. M., & Sanz Molina, A. (2002). *Redes neuronales y sistemas difusos*. (2. Edición, Ed.) © RA-MA Editorial. Retrieved 01 28, 2020
- Cajamarca Velasco, J. A., & Velasco Herrera, C. E. (2018). *dspace.ups.edu.ec*. Retrieved 12 20, 2019, from dspace.ups.edu.ec:
<http://dspace.ups.edu.ec/handle/123456789/16358>
- Cerven, D., & Naranjo , E. (2017). *Instrumentación de procesos industriales*. Retrieved 03 01, 2020, from Rotámetro: <https://instrumentacionuc.wixsite.com/facultad-ingenieria/copia-de-2>
- Distritec. (2013, 09 23). *Electroválvula*. Retrieved 02 05, 2020, from <http://www.distritec.com.ar/que-es-una-electrovalvula-y-para-que-sirve/>
- Gómez, C. T. (2005). *academia.edu*. Retrieved 01 07, 2020, from [academia.edu](http://www.academia.edu/download/34038513/Carlos_Tutosaus-introduccion_automatizacion.pdf):
[//www.academia.edu/download/34038513/Carlos_Tutosaus-introduccion_automatizacion.pdf](http://www.academia.edu/download/34038513/Carlos_Tutosaus-introduccion_automatizacion.pdf).
- Hernández , R. (2017, 08 15). Introducción a los sistemas de control. In L. M. Castillo (Ed.), *Introducción a los sistemas de control* (p. 530). Naucalpan de Juárez: Pearson. Retrieved 01 15, 2020, from <http://lcr.uns.edu.ar/fcr/images/Introduccion%20a%20Los%20Sistemas%20de%20Control.pdf>

- IFP . (2019, 02 25). *Sistema de control*. Retrieved 04 22, 2020, from <https://www.ingmecafenix.com/automatizacion/sistema-de-control/>
- Industruino Home*. (2019). Retrieved 01 06, 2020, from <https://industruino.com/>
- John Moncrieff Ltd*. (2019). Retrieved Enero 6, 2019, from www.johnmoncrieff.co.uk
- Martín, V. A. (2019, 07 29). *Carril DIN*. Retrieved 01 05, 2020, from <http://vamos-comunicaciones-ferroviarias.blogspot.com/2012/12/que-es-carril-din-what-is-din-lane.html>
- Mecafenix. (2018, 06 12). *Electrónica*. Retrieved 02 05, 2020, from Fuente de alimentación: <https://www.ingmecafenix.com/electronica/fuente-de-alimentacion/>
- Mecafenix, F. (2019, 02 25). *Sistema de control*. Retrieved 01 01, 2020, from <https://www.ingmecafenix.com/automatizacion/sistema-de-control/>
- Raffino, M. E. (2019, 02 24). *Fuente de alimentación*. Retrieved 01 10, 2020, from <https://concepto.de/fuente-de-alimentacion/>
- Robologs. (2015, 09 29). *Industruino*. Retrieved 11 03, 2019, from El arduino para automatización industrial: <https://robologs.net/2015/09/29/industruino-el-arduino-para-automatizacion-industrial/>
- Veto y Compañía Ltda. (2019, 01 10). *Medición de nivel de líquidos*. Retrieved 01 10, 2020, from <https://www.veto.cl/blog/medicion-de-nivel-de-liquidos.html>
- Xi CHEN, Yanbo Che, KWE Cheng. (2009). *ieeexplore*. Retrieved 01 01, 2020, from ieeexplore: <https://ieeexplore.ieee.org/document/5228628/authors>

ANEXOS