

**IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y SUPERVISIÓN  
MULTIPLATAFORMA INDUSTRIAL PARA LA CALDERA DE PROCESO DE  
PASTEURIZACIÓN DIDÁCTICA DEL LABORATORIO DE CONTROL Y  
AUTOMATIZACIÓN DE LA UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE  
CALDAS, FACULTAD TECNOLÓGICA**

**CESAR FABIÁN BOLÍVAR GUERRERO  
CODIGO. 20162383043**

**ADRIANA NIYIRETH MARTINEZ CASTILLO  
CODIGO. 20152383002**



**UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS  
FACULTAD TECNOLÓGICA  
INGENIERÍA EN CONTROL  
BOGOTÁ D.C  
2019**

**IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y SUPERVISIÓN  
MULTIPLATAFORMA INDUSTRIAL PARA LA CALDERA DE PROCESO DE  
PASTEURIZACIÓN DIDÁCTICA DEL LABORATORIO DE CONTROL Y  
AUTOMATIZACIÓN DE LA UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE  
CALDAS, FACULTAD TECNOLÓGICA**

**CESAR FABIÁN BOLÍVAR GUERRERO  
ADRIANA NIYIRETH MARTINEZ CASTILLO**

**TESIS PRESENTADA AL PROGRAMA DE INGENIERÍA EN CONTROL PARA  
OPTAR AL TÍTULO DE:  
INGENIERO EN CONTROL**

**DIRECTOR DEL PROYECTO  
ING. JORGE EDUARDO PORRAS BOHADA**

**EVALUADOR DEL PROYECTO  
ING. FRANK NIXON GIRALDO**

**UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS  
FACULTAD TECNOLÓGICA  
INGENIERÍA EN CONTROL  
BOGOTÁ D.C  
FEBRERO DE 2019**



*Dedicatoria*

*A mis padres Luciano Martinez y Martha Castillo por ser las personas que me han acompañado durante todo mi trayecto estudiantil y me han demostrado su cariño, paciencia y apoyo incondicional, siempre serán pilares fundamentales en mi vida.*

*A mis hermanos y sobrinos porque llenan de alegría cada día de mi vida.*

*Adriana Niyireth Martinez Castillo*

*A mi madre Elsy Guerrero quien con su paciencia y perseverancia me da la fortaleza de seguir adelante y a quien le debo lo que soy.*

*A mi padre Roberto Bolívar quien con sus grandes esfuerzos, ha sido ejemplo de superación para mí, ustedes han sido la base de mi formación profesional y apoyo moral incondicional.*

*Cesar Fabian Bolívar Guerrero*

*Agradecimientos*  
*A la Universidad Distrital Francisco José de Caldas, al grupo de profesores,*  
*estudiantes y amigos con los que compartimos y nos preparamos*  
*profesionalmente;*  
*Agradecemos a Giovanni Bolívar por aportarnos sus conocimientos en la parte*  
*mecánica de este proyecto.*

*Por último y no menos importante a nuestros padres por su amor, trabajo y*  
*sacrificio en todos estos años, gracias a ustedes hemos logrado llegar hasta aquí*  
*y convertimos en lo que somos.*

## RESUMEN

La automatización industrial se basa en el control y supervisión de variables que se obtienen de un proceso repetitivo, el cual logra un rendimiento superior a la operatividad manual, la administración y control de esto se obtiene gracias a las diferentes plataformas de programación industrial como RsLogix5000, CODESYS V3.5, TIA Portal, entre otros; Debido a esto los estudiantes de ingeniería en control de la Universidad Distrital Francisco José de Caldas, Facultad Tecnológica, han construido como parte de su formación académica y profesional plantas a escala de procesos industriales, este es el caso de la planta pasteurizadora didáctica modular, la cual usa RsLogix5000 para el control y supervisión de variables como temperatura, nivel, presión, pH entre otras; uno de los módulos que la integran es la caldera, encargada de generar vapor a 40 PSI-140°C, en este proyecto se realizó el acondicionamiento de los sensores y de los equipos actuadores: bomba, resistencia y válvula manual tipo aguja que permite la salida de vapor, para ella se implementaron mecanismos de acople para convertirla en una válvula proporcional de apertura controlada, igualmente el tablero de control tuvo una reestructuración para integrar el SoftPLC<sup>1</sup> V1.0 y la pantalla HMI (Human machine interface), por último se cambió el tanque de reserva de agua a uno de mayor capacidad y robustez.

Se integró un nuevo controlador SoftPLC V1.0 con la plataforma de programación CODESYS V3.5, teniendo en cuenta que la caldera ya posee un controlador Allen Bradley de Rockwell Automation y su plataforma de programación es RsLogix5000, estos dos controladores se convierten en un sistema multiplataforma que controlan el funcionamiento de la caldera de manera independiente. El SOFTPLC V1.0 emula las funciones de un PLC con I/O análogas y digitales con ayuda del programa CODESYS V3.5 SP14. Esta plataforma obtiene los datos de las señales análogas y digitales mediante un microcontrolador PSoC 5LP por medio del protocolo SPI (Serial peripheral interface). Con respecto al diseño del controlador se utilizaron Toolbox del software MATLAB para el proceso de identificación del sistema, complementado por métodos experimentales (Identificación de procesos). Los controlares obtenidos fueron transformados en un “pseudocódigo” para ser implementados en el SoftPLC V1.0, Una pantalla HMI de 7” complementa el proceso de monitoreo y supervisión del funcionamiento de la caldera.

Se espera que este proyecto beneficie a los estudiantes de la Universidad Distrital mediante el desarrollo de prácticas de laboratorio e implementen el control y la supervisión de la caldera con las plataformas de programación y a su vez desarrollen las habilidades de la puesta en marcha del proceso.

**Palabras Clave:** Caldera, Controlador, HMI, Multiplataforma, SoftPLC, Válvula proporcional.

---

<sup>1</sup> SoftPLC es un software que se instala en una SBC (Computador de placa reducida) que emula las funciones de un PLC (Controlador lógico programable).

## ABSTRACT

Industrial automation is based on the control and supervision of variables that occur in a repetitive process, which achieves superior performance and manual operation, administration and control of this refers to the different platforms of industrial programming as RsLogix5000, CODESYS V3.5, TIA Portal, among others; As a result of this, the students of engineering and control of the Francisco José de Caldas District University, Technological Faculty, have built as part of their academic and professional training in a scale of industrial processes, this is the case of the modular didactic pasteurization plant , the RsLogix5000 for the control and monitoring of variables such as temperature, level, pressure, pH among others; one of the modules that integrates the boiler, is responsible for generating steam at 40 PSI-140 ° C, in this project is made the conditioning of sensors and actuator equipment: pump, resistance and manual valve type needle that allows the output of steam, for which conversion mechanisms were implemented for the proportional opening conversion, also included in the control panel a restructuring to integrate the SoftPLC V1.0 and the HMI (Human Machine Interface) screen, finally see The reservoir of water to one of greater capacity and robustness.

A new SoftPLC V1.0 controller was integrated with the CODESYS V3.5 programming platform, taking into account that the boiler already has an Allen Bradley controller from Rockwell Automation and its programming platform is RsLogix5000, these two controllers are converted into a system multiplatform that control the operation of the boiler independently. The SOFTPLC V1.0 emulates the functions of a PLC with I / O and analog and digital with the help of the program CODESYS V3.5 SP14. This platform obtains the data of the analog and digital signals through a PSoC 5LP microcontroller for the SPI protocol medium (serial peripheral interface). With regard to the design of the controller, Toolbox of the MATLAB software is used for the system identification process, complemented by experimental methods (process identification). The controls became a "pseudocode" to be implemented in the SoftPLC V1.0. A 7 "HMI screen complements the process of monitoring and supervising the operation of the boiler.

It is expected that this project will benefit the students of the District University through the development of laboratory practices and the implementation of the control and supervision of the boiler with the programming platforms and the activities of the start-up in the process.

**Keywords:** Boiler, PID Controller, HMI, Multiplatform, SoftPLC, Proportional valve.

## TABLA DE CONTENIDO

	<b>pág.</b>
1. INTRODUCCIÓN	14
2. PLANTEAMIENTO Y DESCRIPCIÓN DEL PROBLEMA	15
3. OBJETIVOS	16
3.1 OBJETIVO GENERAL	16
3.2 OBJETIVOS ESPECÍFICOS	16
4. ESTADO DEL ARTE	17
5. MARCO DE REFERENCIA	28
5.1 Vapor	28
5.1.1 Generadores de vapor	28
5.2 Caldera de Vapor	28
5.3 Componentes de la Caldera de generación de vapor.	29
5.3.1 Componentes de observación	29
5.3.2 Componentes de seguridad	30
5.3.3 Componentes de Alimentación de Agua	31
5.3.4 Componentes de Limpieza	31
5.3.5 Componentes de Control	32
5.3.6 Procedimiento de calderas para la producción de vapor	33
5.4 Válvulas de Control	33
5.5 Control de una Caldera	39
5.6 Sistemas de control	41
5.7 Modelamiento de los sistemas	43
5.7.1 Método de obtención de modelos	44
5.8 Identificación de los sistemas	44
5.8.1 Métodos de identificación de sistemas	44
5.8.2 Procesos a modelar	47
5.9 Controladores PID	48
5.9.1 PID: Acción de control Proporcional – Integral - Derivativa	49
5.10 SoftPLC	51
5.10.1 SoftPLC para SBC.	52
5.10.2 Módulo expansor I/O para SoftPLC	53
5.10.3 Plataformas de programación	54
6. METODOLOGÍA	55
6.1 DIAGNÓSTICO DE LA CALDERA	55
6.1.1 Equipos de control y supervisión	55
6.1.2 Sensores	56
6.1.3 Actuadores	56
6.1.4 Infraestructura	57
6.2 ACONDICIONAMIENTO DE LA CALDERA	59
6.2.1 Etapa de control	59
6.2.2 Planta	60
6.2.3 Etapa final	61
6.2.3.1 Diseño y construcción del mecanismo de CV-105	63
6.2.3.2 Instalación de CV-105 en la caldera	66

6.2.3.3	Diseño y construcción del control electrónico de CV-105	68
6.3	INTEGRACIÓN DE LA NUEVA PLATAFORMA DE PROGRAMACIÓN	72
6.3.1	Comunicación entre Raspberry PI 2B y PSoC 5LP	72
6.3.2	Diseño y construcción del hardware del SoftPLC V1.0	73
6.3.3	Diseño y construcción del software del SoftPLC V1.0	79
6.3.4	Implementación de la interfaz HMI	81
6.4	ARQUITECTURA DE COMUNICACIONES - SoftPLC V1.0 y Allen Bradley	83
6.5	DISEÑO DEL ALGORITMO DE CONTROL	84
6.5.1	Caracterización de los sensores TT-100 y PT-100	85
6.5.2	Caracterización del sensor del actuador CV-105	93
6.5.3	Proceso de identificación de la planta	94
6.5.4	Diseño del controlador	99
7.	PRUEBAS Y RESULTADOS	103
7.1	VALVULA PROPORCIONAL CV-105	103
7.2	SOFTPLC V1.0	103
7.3	CALDERA	104
7.4	CONTROLADOR	106
8.	CONCLUSIONES	108
9.	RECOMENDACIONES	109
10.	BIBLIOGRAFIA	110
11.	ANEXOS	113
11.1	ANEXO 1 - DISEÑO MECANICO DE CV-105	113
11.2	ANEXO 2 - ALGORITMO DE CONTROL DE CV-105	117
11.3	ANEXO 3 - ALGORITMO DE PROGRAMACIÓN DE LA RASPBERRI PI 2B	121
11.4	ANEXO 4 - ALGORITMO DE COMUNICACIÓN DEL PSoC 5LP	123
11.5	ANEXO 5 - DIAGRAMA ESQUEMATICO DEL SoftPLC V1.	126
11.6	ANEXO 6 - DIAGRAMA DE TABLERO DE BORNES Y CONTROL X1 Y X2	127
11.7	ANEXO 7 - DIAGRAMA P&ID FINAL DE LA CALDERA	129
11.8	ANEXO 8 - PRÁCTICA DE CONTROL BÁSICO DE LA CALDERA CON EL SoftPLC v1.0	130
11.9	ANEXO 9 - PRÁCTICA DE IDENTIFICACIÓN Y DISEÑO DEL CONTROLADOR.	149

## LISTA DE FIGURAS

	<b>pág.</b>
Fig 1. Caldera del proceso de pasteurización de la Universidad Distrital. [1]	17
Fig 2. Caldera vertical pirotubular de la Universidad Tecnológica de Pereira. [4]	19
Fig 3. Conexión de OPC server para relés electromecánicos. [14]	21
Fig 4. Esquema de interconexión entre equipos.[18]	22
Fig 5. Modelo simplificado de un ASS.[18]	23
Fig 6. Bloques de modulo software SoftPLC. [21]	24
Fig 7. Diagrama de flujo del sistema con Raspberry Pi.[24]	25
Fig 8. Bloques de la metodología propuesta.[26]	26
Fig 9. Esquema representativo funcionamiento de un generador de vapor.[28]	28
Fig 10. Indicadores de nivel de agua.[30]	29
Fig 11. Manómetro.[31]	29
Fig 12. Analizador de Gases.[32]	30
Fig 13. Ejemplo de Indicador de Temperatura.[33]	30
Fig 14. Partes de una Válvula de seguridad.[34]	30
Fig 15. Bomba Monofásica. [35]	31
Fig 16. Partes Principales en una Caldera.[35]	32
Fig 17. Válvulas de movimiento lineal. [29]	36
Fig 18. Válvulas de movimiento rotativo.[29]	38
Fig 19. Curvas características.[29]	38
Fig 20. Control Feedforward.[37]	39
Fig 21. Control Feedback.[37]	39
Fig 22. Control en Combinación feedback + feedforward.[37]	40
Fig 23. Diagrama de bloques de un sistema de lazo cerrado.[11]	42
Fig 24. Sistema general en tiempo discreto.[11]	43
Fig 25. Diagrama de bloques modelos paramétricos.[11]	46
Fig 26. Diagrama en bloques de un sistema controlado con un PID.[11]	49
Fig 27. Especificación de un sistema de control en el dominio temporal.[11]	50
Fig 28. Paquete de Instalación para Raspberry con CODESYS.[39]	52
Fig 29. Raspberry Pi 2B.[40]	53
Fig 30. CODESYS. [40]	54
Fig 31. Caldera recibida - Junio de 2018 para diagnóstico. [Autor]	55
Fig 32. Tableros recibidos - Junio de 2018 para diagnóstico. [Autor]	57
Fig 33. Canalización recibida - Junio de 2018 para diagnóstico. [Autor]	58
Fig 34. Acondicionamiento-Cambios etapa de control, planta, etapa final. [Autor]	59
Fig 35. Acondicionamiento del tablero de control X1-X2. [Autor]	59
Fig 36. (1)TK-001, (2) TK-002, (3) R-001, (4) P-001. [Autor]	60
Fig 37. Diseño y vista 3D de TK-002: Tanque reserva. [Autor]	60
Fig 38. TK-002: Tanque reserva final. [Autor]	61
Fig 39. Válvula tipo aguja, B. Sensor de posición, C. Motor DC [Autor]	62
Fig 40. Mecanismos de acople para actuador proporcional.[42]	63

Fig 41. Dimensiones base y engranajes E1 y E2. [Autor]	64
Fig 42. Dimensiones de ejes de acople a E1 y E2. [Autor]	64
Fig 43. Resultado final de mecanismo de engranajes. [Autor]	65
Fig 44. Mecanismo de discos acoplado al mecanismo de engranajes. [Autor]	65
Fig 45. Mecanismo final de CV-105. [Autor]	66
Fig 46. Tubería para conexión de CV-105. [Autor]	67
Fig 47. Tubería para mantenimiento de CV-105. [Autor]	67
Fig 48. Diseño esquemático de PCB para CV-105. [Autor]	68
Fig 49. Construcción de PCB para CV-105. [Autor]	69
Fig 50. Esquemático de bloques del programa en PSoC Creator. [Autor]	70
Fig 51. Diseño, 3D y resultado final de la tarjeta de CV-105 con carcasa. [Autor]	70
Fig 52. Configuración externa e interna del bloque SPIS_1. [Autor]	73
Fig 53. Configuración de periféricos de E/S. [Autor]	73
Fig 54. Ubicación de dispositivos electrónicos en la tarjeta del SoftPLC. [Autor]	74
Fig 55. Esquemático y ubicación de dispositivos de Target. [Autor]	75
Fig 56. Diseño y vista 3D de carcasa tarjeta de SoftPLC. [Autor]	76
Fig 57. Resultado final – SoftPLC V1.0. [Autor]	77
Fig 58. SPI Master Configuración. [Autor]	79
Fig 59. Escala lineal de la apertura de la válvula. [Autor]	80
Fig 60. Ignition EDGE.[43]	81
Fig 61. Pantalla HMI. [Autor]	82
Fig 62. Diseño y vista 3D de carcasa pantalla HMI. [Autor]	82
Fig 63. SCADA Final - Caldera. [Autor]	83
Fig 64. Arquitectura de conexión de los dispositivos. [Autor]	84
Fig 65. Diagrama de bloques sistema de control. [Autor]	85
Fig 66. Grafica de Temperatura de subida TT-100 con SoftPLC V1.0. [Autor]	87
Fig 67. Grafica de Temperatura de bajada TT-100 con SoftPLC V1.0. [Autor]	87
Fig 68. Grafica Temperatura subida TT-100 con CompactLogix L23E. [Autor]	89
Fig 69. Grafica Temperatura bajada TT-100 con CompactLogix L23E. [Autor]	90
Fig 70. Grafica de presión de subida PT-100 con SoftPLC V1.0. [Autor]	91
Fig 71. Grafica de presión de bajada PT-100 con SoftPLC V1.0. [Autor]	91
Fig 72. Grafica de presión de subida PT-100 con CompactLogix L23E. [Autor]	93
Fig 73. Grafica de presión de bajada PT-100 con CompactLogix L23E. [Autor]	93
Fig 74. Grafica del sensor de posición vs ángulo de apertura CV-105. [Autor]	94
Fig 75. Señal temperatura salida (y1) señal paso apertura CV-105 (u1). [Autor]	95
Fig 76. Señal salida temp en tanque y señal paso modificada CV-105. [Autor]	96
Fig 77. Señal salida temperatura tanque y señal paso cuadrada CV-105. [Autor]	96
Fig 78. Señal salida temp en tanque y señal pseudoaleatoria CV-105. [Autor]	97
Fig 79. Respuesta a la señal paso de $G(z)_1$ . [Autor]	97
Fig 80. Respuesta a la señal paso de $G(z)_2$ . [Autor]	98
Fig 81. Respuesta a la señal paso de $G(z)_3$ . [Autor]	98
Fig 82. Respuesta a la señal paso de $G(z)_4$ . [Autor]	99
Fig 83. Arquitectura de control Diseñada e Implementada. [Autor]	100

Fig 84. Tiempo de estabilización de la planta lazo abierto igual a 450s. [Autor]	100
Fig 85. Tiempo de estabilización de planta en lazo cerrado = 233s. [Autor]	101
Fig 86. CV-105 Final. [Autor]	103
Fig 87. Tablero de control X1 Final. [Autor]	103
Fig 88. Etiquetas finales del tablero de control X1. [Autor]	104
Fig 89. Instalación final de SoftPLC V1.0. [Autor]	104
Fig 90. Diagrama P&ID final de la caldera. [Autor]	105
Fig 91. Estructura Física final de la Caldera. [Autor]	105
Fig 92. Resultados identificación de procesos. [Autor]	106
Fig 93. Respuesta del controlador diseñado. [Autor]	107
Fig 94. Respuesta final del controlador en HMI. [Autor]	107

## LISTA DE TABLAS

	<b>pág.</b>
Tabla 1. Comparación de características de SBC de producción nacional.[27]	27
Tabla 2. Diferentes estructuras de modelos paramétricos.[11]	46
Tabla 3. Comparación de características de PSoC.[40]	54
Tabla 4. Equipos de control y supervisión de la caldera [Autor]	56
Tabla 5. Sensores de la caldera. [Autor]	56
Tabla 6. Actuadores de la caldera [Autor]	56
Tabla 7. Equipos de almacenamiento y proceso [Autor]	57
Tabla 8. Configuración de estado de las válvulas para salida de vapor [Autor]	67
Tabla 9. Componentes para Target [Autor]	68
Tabla 10. Componentes para la tarjeta del SoftPLC V1.0. [Autor]	74
Tabla 11. Componentes para Target [Autor]	76
Tabla 12. Datos de subida y bajada TT-100 con SoftPLC V1.0. [Autor]	85
Tabla 13. Datos de subida y bajada TT-100 con CompactLogix L23E. [Autor]	87
Tabla 14. Datos de subida y bajada PT-100 con SoftPLC V1.0. [Autor]	90
Tabla 15. Datos de subida y bajada PT-100 con CompactLogix L23E. [Autor]	92
Tabla 16. Datos de linealización del sensor de posición para CV-105. [Autor]	93
Tabla 17. Características de FT según señal de identificación. [Autor]	99
Tabla 18. Asignación de variables para el código. [Autor]	102

## 1. INTRODUCCIÓN

Hoy en día el uso de calderas en procesos industriales es de gran importancia, debido a que se encargan de generar vapor para el calentamiento de otro proceso; han venido en aumento con la evolución de la instrumentación y el empleo de nuevas tecnologías en las plataformas de programación de los controladores para realizar las lecturas precisas de las variables físicas (nivel, temperatura, presión, flujo entre otras) y elementos como actuadores que permitan llevarlas a niveles deseados; por esta razón es indispensable por la alta demanda en la industria la puesta en marcha de procesos industriales haciendo uso de diferentes plataformas de programación para PLC's,

Entonces surge la necesidad de integrar el manejo de sistemas multiplataforma con diferentes protocolos de comunicación para el control y supervisión de un proceso; contar con instrumentación, equipos e infraestructura que permitan ejecutar estrategias de control, teniendo como finalidad una supervisión y control óptimo.

A diferencia de otros proyectos realizados Universidad Distrital Francisco José de Caldas como: Aplicación de un sistema de control en la caldera de proceso de pasteurización didáctica para el laboratorio de industrial, Diseño e implementación de un control de temperatura para la plataforma didáctica de procesos térmicos del grupo de investigación integra y Desarrollo e implementación de un control y supervisión para la planta térmica AMATROL, Este proyecto enfoca su funcionamiento al diseño e implementación de un sistema multiplataforma para controlar el proceso de generación de vapor, de tal modo que pueda apuntar a ser una herramienta didáctica para el programa de ingeniería en control, gracias a la interacción con dos diferentes plataformas de programación (Rockwell y CODESYS), Adicional el proceso involucra áreas del conocimiento como termodinámica, control e instrumentación entre otras, por lo que podría brindar a la comunidad académica una oportunidad para aplicar o demostrar diferentes conceptos implementados en ambientes industriales.

Como parte de este documento, se realizará una descripción de la planta, en el que se identificarán sensores, actuadores, equipos y software (ampliamente utilizados en grandes empresas que implementan sistemas de control). Se mostrarán los pasos ejecutados para cumplir con el desarrollo de este proyecto y la realización de las prácticas que permiten al evidenciar el funcionamiento del sistema multiplataforma, por último se espera que mediante estas prácticas otros estudiantes puedan complementar sus estudios, abordar diferentes tipos de algoritmos, habilidades con diferentes plataformas de programación y estrategias de control.

## **2. PLANTEAMIENTO Y DESCRIPCIÓN DEL PROBLEMA**

El laboratorio de control y automatización de la Facultad Tecnológica de la Universidad Distrital Francisco José de Caldas, cuenta con una planta Pasteurizadora didáctica de Procesos, conformada por un módulo de dosificación y corrección de PH, un módulo de calentamiento por medio de una caldera, y un módulo de enfriamiento con el fin del desarrollar actividades académicas teórico – prácticas por parte de los estudiantes.

La caldera tiene limitaciones en cuanto a la falta de interconexión con plataformas de programación industrial, ya que solo cuenta con una conexión al controlador de Rockwell, además el sensor de presión y de nivel están en mal estado y no cumplen su correcto funcionamiento para el proceso, al mismo tiempo la válvula proporcional que controla la salida de vapor tiene fallas en el sistema de engranajes, como consecuencia se reducen las prácticas de laboratorio en la parte de instrumentación y control de este módulo de generación de vapor.

### **3. OBJETIVOS**

#### **3.1 OBJETIVO GENERAL**

Implementar un sistema de control y supervisión multiplataforma industrial para la caldera de proceso de pasteurización didáctica del laboratorio de control y automatización de la Universidad Distrital Francisco José de Caldas, Facultad tecnológica.

#### **3.2 OBJETIVOS ESPECÍFICOS**

- Realizar un diagnóstico de la caldera y acondicionar los sensores y la válvula proporcional para el correcto funcionamiento del proceso.
- Implementar un sistema de acondicionamiento de I/O para el SoftPLC y un sistema de conmutación que permita la selección de la plataforma de programación.
- Diseñar e implementar un sistema de control para la apertura de la válvula proporcional utilizando el método de identificación de procesos.
- Implementar un sistema de supervisión para la interacción del usuario y la caldera mediante una HMI.
- Elaborar dos prácticas de laboratorio en las cuales se demuestre el funcionamiento de la caldera.

#### 4. ESTADO DEL ARTE

Teniendo en cuenta que la caldera es un módulo del proceso de pasteurización didáctico del laboratorio de control y automatización de la Universidad Distrital, facultad tecnológica, la cual lleva varias etapas de construcción y reestructuración por parte de los estudiantes de la asignatura Instrumentación de procesos II desde el segundo semestre del 2015 hasta la fecha.

El único antecedente de tesis de grado que existe de la caldera hasta el momento es la *“Aplicación de un sistema de control en la caldera de proceso de pasteurización didáctica para el laboratorio de industrial de la Universidad Distrital francisco José de caldas facultad tecnológica”* el cual se basa en la instrumentación y control de la caldera para la generación de vapor en un proceso de pasteurización tipo Batch, realizado en el 2016, ver figura 2, donde se pudo identificar experimentalmente algunas características termodinámicas, así como un comportamiento propio del sistema y gracias a ello se evidencio que es un sistema cuya reacción ante perturbaciones o una señal de referencia es bastante prolongado, por lo que la aplicación de un método de control PID (proporcional, integral, derivativo) es suficiente para controlarlo.



Fig 1. Caldera del proceso de pasteurización de la Universidad Distrital. [1]

Además para esa misma época se realiza la adecuación del módulo de enfriamiento del anterior proceso de pasteurización; Por medio de análisis matemáticos y modelamientos físicos se ha logrado diseñar diferentes controladores que se ajustan a la tarea principal de la planta. La tarea principal que debe realizar la etapa de refrigeración es llevar un producto determinado de una temperatura inicial a una temperatura final mucho más baja que la anterior. Para lograrlo se ha diseñado un sistema hidráulico que permite desplazar el producto a través de dispositivos de disipación térmica dispuestos en una unidad de refrigeración. Sumado a esto electrónicamente se modifica el caudal del sistema hidráulico para lograr un control sobre la temperatura del producto. Por último, los sistemas de control y de supervisión fueron implementados de tal manera que cumplieran con estándares

industriales, haciendo que los estudiantes que realicen prácticas sobre la planta identifiquen los diferentes sistemas que podrían encontrar en una industria. La estación cuenta con relés industriales, un sistema embebido micro PLC, un sistema de comunicación industrial sobre TCP/IP y puede ser programado en los diferentes rangos de la norma IEC-61131-3. [1]

En la revisión de artículos, conferencias y tesis de grado se ha encontrado tanto a nivel nacional como internacional aplicaciones de sistemas multiplataforma en distintas áreas de la electrónica, además diferentes tipos de controladores que se han diseñado e implementado para procesos de generación de vapor por medio de una caldera. La siguiente revisión se ha realizado de forma cronológica en cuanto a las aplicaciones que se han dado a los diferentes controladores en una caldera, los sistemas multiplataforma y la implementación de SoftPLC a distintos proyectos.

Una caldera de vapor es una unidad de proceso de gran importancia presente en todo tipo de industrias. Las necesidades de vapor de agua en una industria son esenciales y pueden resumirse en: Vapor para turbinas para un ahorro eléctrico, vapor para el intercambio de calor, vapor como materia prima y vapor como medio de obtención de energía eléctrica.

Controlar de forma efectiva las condiciones de operación de una caldera es una necesidad obvia, si se tiene en cuenta que, las elevadas presiones y temperaturas de trabajo son las principales responsables de los problemas de peligrosidad, por riesgo de explosiones. Tampoco debe olvidarse el aspecto económico, considerando no sólo los costes de construcción, sino también los elevados costes de operación y de mantenimiento, relacionados con las condiciones de operación mencionadas. En esta tesis desarrollada en el 2004, se obtuvo el modelo matemático de una caldera de vapor convencional y posteriormente se usó el modelo para simular el comportamiento de las condiciones anteriormente mencionadas y observar la dinámica del sistema de generación de vapor. [2]

La tesis Modelamiento y control de una planta pasteurizadora, realizada en la Pontificia Universidad Javeriana, Facultad de Ingeniería, hace referencia a la generación de vapor por medio de una caldera que pertenece a un proceso de pasteurización, la planta piloto con la cual se analiza dicho proceso y se obtienen diferentes datos. Esta planta está diseñada para desarrollar la pasteurización de productos a temperaturas en el rango 70-80°C. La planta piloto tiene un control OPTOMUX que abre una válvula de aire triple que conmuta cuando el producto ha alcanzado la temperatura de pasteurización. Este actúa además sobre una válvula de dos vías que controla el flujo de vapor que conmuta según un bloque PID disponga. La disposición de la planta hace que con el control del vapor se controle también la temperatura del producto a pasteurizar. [3]

La Universidad Tecnológica de Pereira, realiza el diseño para la instrumentación y control de una variable para la caldera del laboratorio de mecatrónica, la intención

de hacer una instrumentación digital para la caldera del laboratorio, ver fig.1, es convertir este sistema en uno capaz de tener control sobre la variable de presión y poder mostrar en una interfaz hombre-máquina (HMI) las condiciones en las que se encuentra, lo anterior gracias al uso de sensores adecuados a la necesidad de cada variable. La finalidad de este proyecto es diseñar un sistema controlado y supervisado, que permita a los usuarios del laboratorio hacer procesos y realizar diferentes aplicaciones, de tal manera que estas sean más fáciles gracias a las investigaciones realizadas en este proyecto. [4]



Fig 2. Caldera vertical pirotubular de la Universidad Tecnológica de Pereira. [4]

Con el desarrollo de controladores clásicos a través de modelos experimentales de la caldera, se realiza controles con metodología moderna como: *Direct Fuzzy Backstepping control for turbine main steam valve of multi-machine power system* (Control de retroceso directo difuso para la válvula de vapor principal de la turbina del sistema de potencia de varias máquinas), el problema de estabilidad del control de la válvula de vapor en un sistema de potencia de varias máquinas, se aborda mediante un retroceso con método directo difuso. Un controlador se diseña utilizando un método de retroceso difuso para parámetros inciertos y perturbaciones externas en el sistema. En el proceso de diseño, las características no lineales se mantienen sin ninguna linealización. La estabilidad limitada de todo el sistema está garantizada por la teoría de Lyapunov. Finalmente, los resultados de la simulación demuestran la efectividad del controlador. [5]

Con los datos adquiridos en una caldera se puede evaluar el rendimiento de ciertos parámetros como en: *Performance evaluation of the boiler combustion control system based on data driven* (Evaluación del rendimiento del sistema de control de la combustión de la caldera basado en datos). Este documento presenta un tipo de método de evaluación de desempeño basado en datos. El funcionamiento normal del sistema no produce perturbaciones adicionales y el cálculo es simple de implementar. Este método puede ser una implementación confiable y precisa para otro sistema de control multivariable. [6]

En lo que más se ha trabajado en las calderas para la generación de vapor es sobre la metodología de control como la Investigación sobre la estrategia de control de la caldera de almacenamiento de calor eléctrico basada en agentes múltiples [7], están también los desarrollados en MPC (modelo control predictivo) en cascada de supervisión para calderas industriales [8], otros como controles continuos como el feed-forward y feedback para el sistema de control de coordinación de la caldera-turbina [9] y por último en las válvulas de vapor en los sistemas de potencias no lineales se ha desarrollado un control de modo deslizante de retroceso adaptable. [10]

El primer trabajo significativo en control automático fue el regulador de velocidad centrífugo de James Watt para el control de la velocidad de una máquina de vapor, en el siglo XVII. Minorsky Hazen y Nyquist, entre muchos otros, aportaron trabajos importantes en las etapas iniciales del desarrollo de la teoría de control. En 1922, Minorsky trabajó en controladores automáticos para el guiado de embarcaciones, y mostró que la estabilidad puede determinarse a partir de las ecuaciones diferenciales que describen el sistema. En 1932, Nyquist diseñó un procedimiento relativamente simple para determinar la estabilidad de sistemas en lazo cerrado, a partir de la respuesta en lazo abierto a entradas sinusoidales en estado estacionario. En 1934, Hazen, quien introdujo el término servomecanismos para los sistemas de control de posición, analizó el diseño de los servomecanismos con relé, capaces de seguir con precisión una entrada cambiante. [11]

El siguiente tema son los sistemas multiplataforma que ejercen funciones para el aprendizaje de software de programación, teniendo en cuenta la realización de proyectos, en el año 2005 se desarrolla un sistema P2P (punto a punto) multiplataforma, ofrece muchas funciones atractivas, como colaboración, auto organización, equilibrio de carga, disponibilidad, tolerancia a fallas y anonimato. Sin embargo, también enfrenta serios desafíos. En trabajos previos, se implementa una plataforma de colaboración P2P sincrónica llamada TOMSCOP. Basado en los servicios de grupos de pares elementales ofrecidos por el marco general de JXTA, TOMSCOP proporciona los cuatro tipos de servicios adicionales: transporte de mensajes síncronos, administración de salas de pares, soporte de comunicación entre pares y administración de espacios de aplicaciones. [12]

Los autores presentan una arquitectura orientada a servicios de tres capas que acomoda diferentes plataformas de sensores y expone su funcionalidad de manera uniforme a la aplicación comercial de red de sensores inalámbricos multiplataforma, trabajo que fue motivado por casos comerciales reales de la industria del petróleo y el gas. Se implementa tres plataformas de sensores (Particle,  $\mu$ Node y Sindrion) integradas a través del estándar universal plug and play (UPnP) e incorporadas en un sistema de software empresarial, revelando una serie de desafíos que deben tenerse en cuenta cuando se implementan redes inalámbricas de sensores y actuadores en sitios industriales, como guía. [13]

A nivel industrial se implementa sistema multiplataforma y multitecnología para interpretación y análisis de disparo de protección en subestaciones de distribución: donde se describe la interpretación y el análisis del disparo de las protecciones en las subestaciones de distribución. El sistema fue diseñado para ser adecuado a diferentes tecnologías de relevadores (desde electromecánica a digital), a diferentes fabricantes y también a diferentes protocolos de comunicación. El sistema obtiene información local de la subestación y los datos de los relés de protección, utilizando un sistema experto basado en reglas (CLIPS). Además, para simular fallas, se utilizó un equipo de prueba de relés. Los relés de protección se organizaron para simular un compartimiento de alimentación de una subestación de distribución, lo que permite la configuración de diferentes fallas del sistema de energía para diferentes tipos de relés. [14]

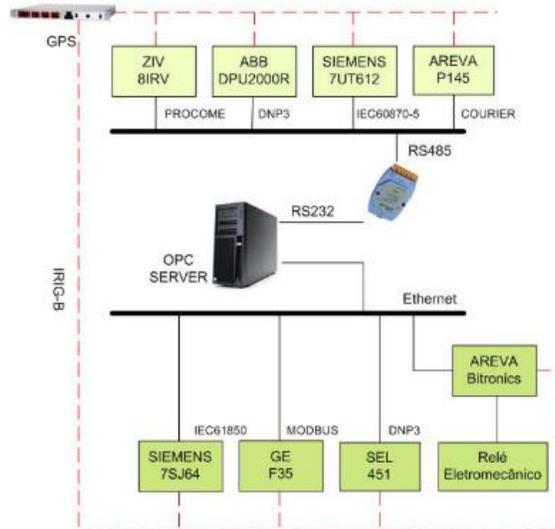


Fig 3. Conexión de OPC server para relés electromecánicos. [14]

A nivel Agropecuario se realiza un sistema automatizado multiplataforma para el monitoreo y control de riego por aspersión con el control web y móvil es un facilitador de los diversos procesos en varias áreas, entre ellas el sector agrícola. Específicamente en la gestión del riego, la tecnología de menor costo no puede satisfacer las necesidades del agricultor, que es el suministro de agua correcto para las plantas y el control remoto del riego. El sistema permite el monitoreo remoto desde cualquier lugar por medio de una computadora y/o dispositivo móvil por Internet. [15]

Para los sistemas de recolección y envío de paquetes existen sistemas Transitic, que son típicamente sistemas de transporte utilizados para transportar paquetes/productos. El diseño de estos sistemas es cada vez más complejo debido al aumento del flujo y la velocidad, presentando problemas de hardware y software. Los ingenieros de control, que dominan la lógica de los sistemas de control, también tienen que adaptarse al tipo de hardware ya presente en el cliente (PLC) sin necesariamente dominar su programación. En el desarrollo se presenta las especificidades de las generaciones de Straton y Unity y analizan el costo de agregar una nueva plataforma. [16]

Debido a la incursión de sistemas operativos de licencia libre y su aceptación en los diferentes sectores industriales, surge la necesidad de integrar todos los sistemas de supervisión y control de esta nueva tendencia de una forma eficiente y totalmente interoperable. Esta es la razón por la cual un estudio enfocado a crear una herramienta que permite controlar procesos en cualquier tipo de sistema operativo o plataforma, válido en sistemas Linux, al usar conceptos de comunicaciones industriales y sistemas telemáticos, como OPC (OLE para Control de procesos) y Servicios Web. Estas herramientas se implementaron en un sistema de supervisión y control de procesos industriales con capacidad de ejecución en cualquier tipo de plataforma y sistema operativo, con funciones de almacenamiento y acceso remoto para los usuarios del sistema por medio de Internet y dispositivos móviles, reduciendo así los costos de implementación y operación al usar herramientas de software libre desarrollado en el año 2012. [17]

La versatilidad en cuanto al ingenio del diseñador para la Integración de plataformas de bajo costo en sistemas de energía eléctrica para control y monitoreo; Para mejorar los sistemas de control, es necesario proporcionar a estas instalaciones tecnologías que ofrezcan interoperabilidad y control de tiempo crítico de los activos. Los protocolos de comunicación Industrial Ethernet corresponden a una alternativa dentro de las herramientas de automatización y mantenimiento implementadas en las redes de distribución eléctrica; este tipo de protocolos se destacan de manera positiva en el mercado debido a sus comunicaciones en tiempo real, altas tasas de transmisión de datos entre activos físicos, uso de topologías de red flexibles para la comunicación entre activos y una gran cantidad de nodos conectados en la red. Este artículo presenta una forma simple y confiable de integrar protocolos de comunicación Industrial Ethernet a través del uso de plataformas electrónicas de bajo costo como, por ejemplo, Arduino, Raspberry PI, BeagleBone, entre otras existentes en el mercado.

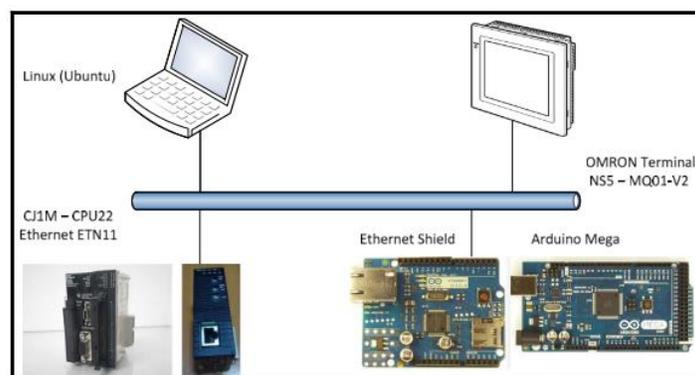


Fig. 4. Esquema de interconexión entre equipos.[18]

Estos protocolos integran herramientas amigables de HMI (Interfaz Hombre-Máquina) y SCADA (Control de Supervisión y Adquisición de Datos). Este enfoque también permite simular el comportamiento de un PLC (Controlador Lógico Programable) desde el punto de vista de la interacción con el equipo industrial

OMRON, reasignando su memoria y permitiendo la adquisición de datos desde varios dispositivos a través de protocolos Industrial Ethernet, como Ethernet/IP, Modbus/TCP, POWERLINK, entre otros. En la Fig. 4, se ve un modelo simplificado, estos sistemas están cubiertos por IEC 61850, un estándar para sistemas de automatización en subestaciones donde la comunicación entre diferentes niveles se lleva a cabo a través de Ethernet de alta velocidad, proporcionando respuestas por debajo de 4 ms para los relés de protección. [18]

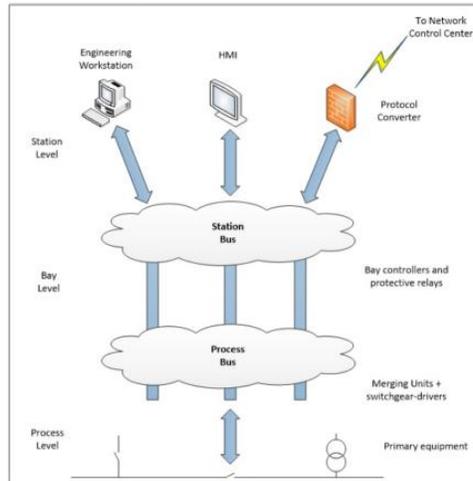


Fig 5. Modelo simplificado de un ASS.[18]

En el entorno académico y parte del Industrial ha llegado herramientas de bajo costo que se pueden implementar para emular las funcionalidades de un PLC, este tiene el nombre de SoftPLC que utiliza un PC en combinación con el software para poder realizar las funciones de un PLC, existen SBC (Single Board Computer) como Raspberry pi, BeagleBone, Arduino, que pueden implementar para un SoftPLC, también empresas como Siemens han sacado su propia versión de SoftPLC.

Se encontraron dos artículos de implementación con WinPLC y Matlab donde se muestra la conexión directa y la implementación de esquemas de control diseñados de aplicaciones preparadas para un WinPLC industrial. El documento trata de la realización industrial del control de retroalimentación de aplicaciones exigentes en su control, algoritmos de control y visualización junto con el diseño, la implementación y el control en tiempo real de MATLAB/Simulink. Este último beneficio produce datos interesantes del proceso de control real para dar una razón más al uso de la solución WinCon. El enfoque de los métodos de desarrollo que se describen en el documento permite la extensión de la posibilidad de minimizar los costos y la utilización de sistemas de control industrial de hardware ampliamente externo. Se describe el método de desarrollo de las aplicaciones de control por el compilador de control WinPLC/WinPAC y REX industrial en comparación con los métodos de desarrollo clásicos que requieren costosas tarjetas de medición RT y producen solo una solución de desarrollo de propósito. [19]

Este artículo, se basa en la especificación del proceso de desarrollo de esquemas de control diseñados de aplicaciones preparadas para una utilización industrial

mediante conexión directa e implementación al sistema de control elegido WinPLC/WinPAC. Se trata de una manera de cómo producir una solución para aplicaciones de control que se preparan, diseñan e implementan en MATLAB-Simulink para la plataforma objetivo. El sistema de control multiplataforma REX se utiliza como un puente entre MATLAB y la plataforma objetivo. [20]

La depuración de la repetición determinística es un enfoque para encontrar errores en el software implementado. Registra una aplicación que se ejecuta en el campo para que se pueda reproducir de forma determinista sin conexión en un sistema de desarrollo con fines de depuración. Para habilitar la depuración determinista de la repetición, es necesario registrar todas las influencias externas y las fuentes de no determinismo en la ejecución original del programa. Desde ese registro de seguimiento y desde un estado inicial conocido, el programa puede reproducirse de manera determinista sin requerir ninguna conexión con el entorno original. El sistema de control automático basado en PC es de bajo costo, bien abierto, fácil de usar, etc., por lo que se convierte en una nueva dirección en la tecnología de automatización. La formulación de las normas internacionales IEC61131-3 no solo normaliza el campo del lenguaje de programación de control industrial, sino que también proporciona las condiciones para el desarrollo de la tecnología PLC. Finalmente, se verificaron el rendimiento y la confiabilidad en tiempo real del SoftPLC. Los resultados muestran que el sistema es confiable y el efecto de control es bueno. [21] [22]

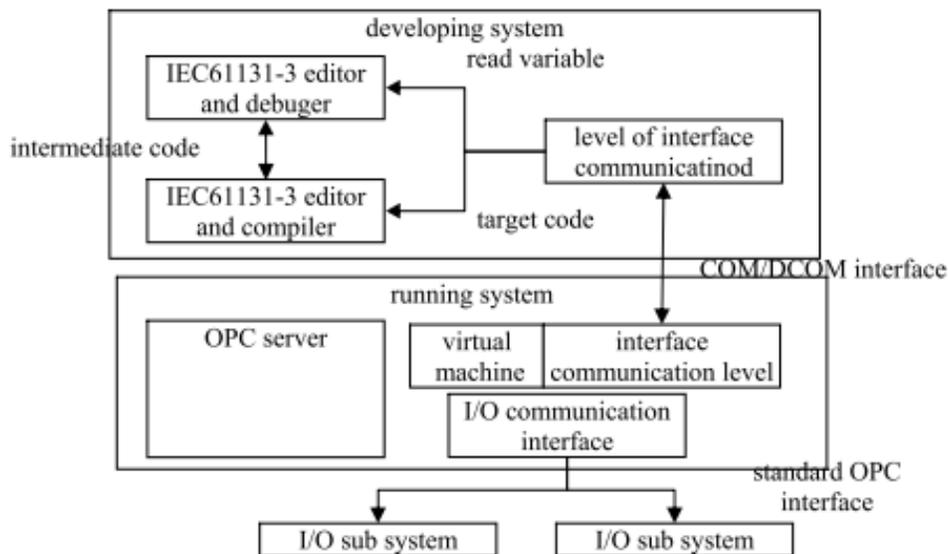


Fig 6. Bloques de módulo software SoftPLC. [21]

Aplicaciones de alguno SoftPLC basados en SBC (Single Board Computer) *El diseño e implementación del sistema de máquina virtual en el sistema embebido SoftPLC:*

- *Embedded SoftPLC* es una nueva tecnología de alta tecnología en el área de control industrial, que supera los defectos del PLC tradicional y ha captado la

atención general en esta área. El sistema de máquina virtual es el núcleo del sistema SoftPLC integrado. Este artículo muestra cuidadosamente el diseño y la implementación del sistema de máquina virtual con el ejemplo de un producto de sistema SoftPLC integrado: ESDDC. La máquina virtual diseñada en este documento presentará un nuevo plan de programación de SoftPLC y, al mismo tiempo, brinda una nueva forma de pensar el diseño de software para los desarrolladores de software. [23]

El hecho de creer que la Raspberry sólo está con un fin académico, lo contraria los siguientes trabajos donde se puede apreciar el alcance que puede tener este dispositivo, siguiendo unos recomendaciones en su uso.

- *Raspberry Pi para la automatización de la planta de tratamiento de agua:* La automatización de la planta de tratamiento de agua ya se ha desarrollado y utilizado ampliamente en muchos países. Pero la mayoría usa controladores lógicos programables: PLC. Este documento se centra en un sistema de control y monitoreo innovador e inteligente para la planta de tratamiento de agua mediante el uso de "Raspberry Pi" como una alternativa efectiva a los PLC para la automatización de pequeñas plantas de tratamiento de agua. Raspberry Pi es una minicomputadora que tiene la capacidad de controlar el sistema viene con ventajas como el bajo costo y el tamaño compacto. [24]

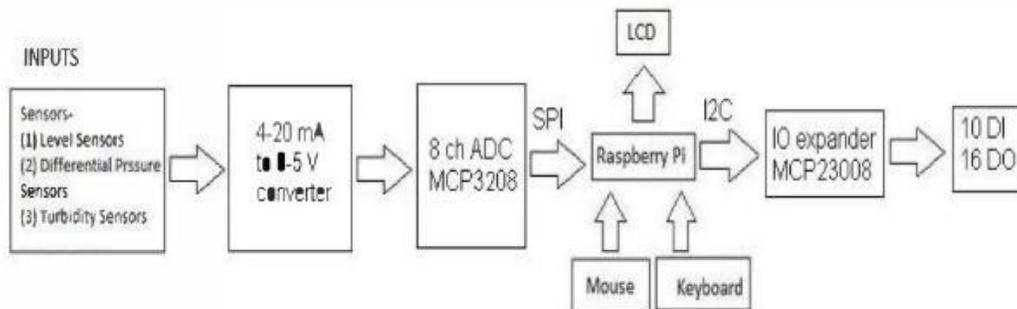


Fig 7. Diagrama de flujo del sistema con Raspberry Pi.[24]

- *Desarrollo del modelo tridimensional de experimento PLC virtual basado en Unity3D:* El experimento PLC es un curso importante de educación técnica y capacitación en todo tipo de escuelas. El modelo físico del PLC y el modelo de control real son indispensables en el experimento PLC tradicional. No es conductivo para el autoestudio de los estudiantes porque el costo de mantenimiento y actualización de este experimento es mayor y el lugar experimental es fijo. Para resolver estos problemas, este documento diseña un esquema de experimento de PLC virtual de software puro utilizando PLC virtual SIEMENS que proporciona señales de control y un modelo tridimensional de experimento virtual basado en Unity3D. El nuevo esquema de experimento puede mostrar intuitivamente la autenticidad del experimento y superar las desventajas del PLC virtual SIEMENS que solo puede mostrar el estado de las señales de control pero no puede mostrar los estados de funcionamiento de los equipos periféricos. Al final de este documento, se proporciona un ejemplo, un

modelo experimental tridimensional de semáforo virtual basado en Unity3D para PLC virtual SIEMENS, para ilustrar la viabilidad del esquema. [25]

- Automatización de la subestación de 11 kV usando frambuesa pi desarrollado en la India, en la metodología convencional, la automatización de la subestación se realiza mediante el uso de PLC y sistemas SCADA, pero para una pequeña automatización de subestación, este sistema es considerablemente costoso. Este proyecto adopta soluciones de automatización de bajo costo basadas en la tecnología IOT reciente que utiliza procesadores integrados como raspberry pi para la automatización de una pequeña subestación de 11 kV. En el proyecto el software Codesys IEC 61131-3 se utiliza para convertir un Raspberry pi como un SoftPLC. Los parámetros analógicos y digitales del campo se supervisan y controlan usando el GPIO, relés de interfaz y puertos de comunicación disponibles en raspberry pi. La interfaz de usuario y las pantallas HMI se desarrollan utilizando el segmento de visualización web Codesys, al que se puede acceder tanto a nivel local como a través de una red cableada o inalámbrica. [26]

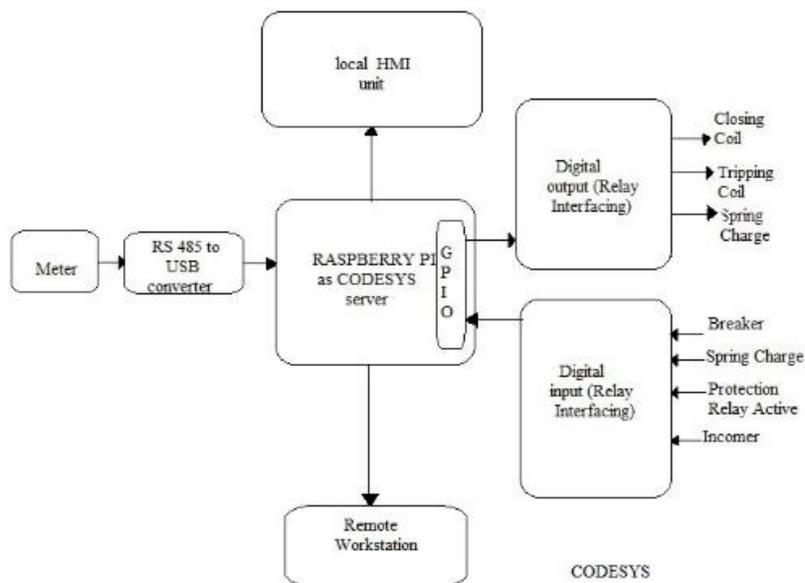


Fig 8. Bloques de la metodología propuesta.[26]

- *Implementación de control para equipos periféricos de máquina basados en el SoftPLC externo integrado con CNC:* Este artículo analiza la estructura general del sistema de control CNC con el módulo SoftPLC que se implementó como una unidad externa. Se investigaron las ventajas de reducir el ciclo de EtherCAT, aumentar la confiabilidad del sistema de control y las desventajas asociadas con el costo adicional de la solución propuesta. La solución propuesta también utiliza el SoftPLC externo en lugar del controlador de seguridad y reduce los requisitos de hardware para CNC. Esto finalmente conduce a una reducción general en el costo del sistema de control distribuido. Se realizó un análisis de soluciones de hardware para seleccionar una microcomputadora de placa única para SoftPLC. [27]

Tabla 1. Comparación de características de SBC de producción nacional.[27]

System	Characteristics				
	Manufacturer	Processor	RAM	Periph. interfaces	OS
Raspberry Pi B+	Raspberry Pi Foundation	ARM1176 JZ-F, (700 MHz)	512 MB	GPIO, USB, Ethernet	Linux
Tion-Pro 28	"Electronic Equipment Factory" Ltd.	Freescall IMX287 (454 MHz ARM9)	128 MB	GPIO, USB, Ethernet, UART	Linux
Fastwel CPB905	"Fastwel Group" LLC	AMD Geode LX800 (500 MHz)	256 MB	Ethernet, UART, USB, FBUS, PCI, ISA	Linux, QNX, Windows XP Embedded/CE
Fastwel CPB909	"Fastwel Group" LLC	Intel Atom E38xx (1.33 -1.91 GHz)	Up to 4 GB	GPIO, Ethernet, PCI, PCI-e	Linux, QNX, Windows XP Embedded/CE
HolaDuo-PC	STARTERK IT.RU	Dual ARM Cortex-A9 (1 GHz)	1 GB	SPI, USB, Ethernet, UART, mini PCI-e	Linux
Baikal T1	"T-Platforms" Ltd.	2 core P5600 MIPS 32 r5 (1,2 GHz)	At least 1 GB	GPIO, USB, UART, SPI, Ethernet, PCI-e	Linux

## 5. MARCO DE REFERENCIA

### 5.1 Vapor

Cuando se habla de vapor, casi siempre se hace referencia al vapor de agua, ya que esta ha sido una sustancia muy importante en la historia de la humanidad. El vapor de agua es básicamente agua con un nivel energético tal, que es capaz de vencer las fuerzas de cohesión que mantienen unidas a las moléculas, causando que estas se separen unas de otras. En termodinámica se denomina al estado de vapor como al estado gaseoso de una sustancia que está por debajo del punto crítico de dichas sustancia. En la actualidad el vapor de agua se sigue utilizando en una gran variedad de procesos, debido a que la materia prima para producirlo (el agua), es abundante, de fácil obtención y por lo tanto de costos bajos; es una sustancia con la capacidad de transportar grandes cantidades de energía por unidad de masa, además de ser de fácil generación, distribución, y control.

Hoy en día los principales usos del vapor en la industria son la generación de trabajo, principalmente en centrales termoeléctricas, nucleares y geo eléctricas, y en procesos de transferencia del calor, como por ejemplo en la industria Petroquímica, Química, Farmacéutica, Metalúrgica, Naval, Textil, Papelera, Cervecera, Tabacalera, Alimentación, Bebidas, Caucho, Servicios, entre otras.

#### 5.1.1 Generadores de vapor

Son dispositivos diseñados para suministrar calor a una masa de agua que puede estar o no en reposo, con el fin de provocar en ella el cambio de estado. Estos dispositivos son conocidos como calderas y funcionan poniendo en contacto el agua a evaporar con superficies a temperaturas elevadas conocidas como superficies de calefacción, logrando así que el calor viaje de las superficies con temperaturas altas hacia el agua que se encuentra a una temperatura más baja. La Figura 9 muestra de forma esquemática el funcionamiento de un generador de vapor. [28]



Fig 9. Esquema representativo funcionamiento de un generador de vapor.[28]

### 5.2 Caldera de Vapor

La caldera de vapor es una unidad de proceso de gran importancia presente en todo tipo de industrias, se utilizan debido a que muchos procesos emplean grandes cantidades de vapor. La caldera se caracteriza por una capacidad nominal de producción de vapor, en toneladas/hora, a una presión especificada y con una capacidad adicional de caudal en puntas del consumo de la fábrica.

En definición la caldera es un recipiente metálico, cerrado, destinado a producir vapor o calentar agua, mediante la acción del calor a una temperatura superior a la del ambiente y presión mayor que la atmosférica. [29]

### 5.3 Componentes de la Caldera de generación de vapor.

A continuación se dan a conocer los componentes que integran la caldera para la generación de vapor.

#### 5.3.1 Componentes de observación

##### - *Indicadores de nivel de agua*

Toda caldera deberá estar provista, a lo menos, de dos indicadores de nivel de agua, independientes entre sí. Uno de ellos deberá ser de observación directa del nivel de agua, del tipo tubo de vidrio. [30]

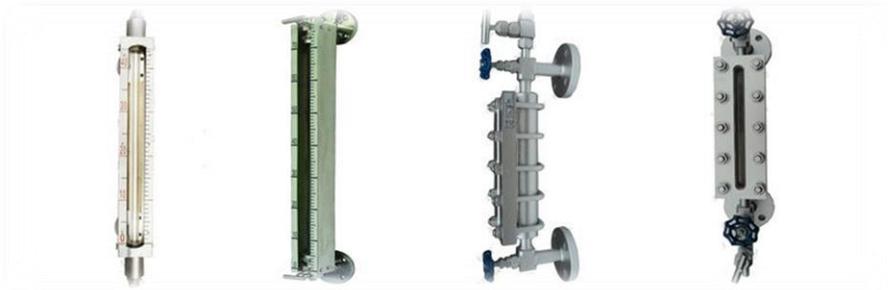


Fig 10. Indicadores de nivel de agua.[30]

##### - *Indicadores de presión*

Toda caldera deberá estar provista de uno o más manómetros (ver Fig. 11), que se conectarán a la cámara de vapor de la caldera Mediante un tubo que forme un sello de agua. [31]



Fig 11. Manómetro.[31]

##### - *Analizadores de gases*

Son aparatos que sirven para controlar la calidad de la combustión, a través del análisis de los gases que salen por la chimenea (Fig. 12) [32]



Fig 12. Analizador de Gases.[32]

- *Indicadores de temperatura*

Son instrumentos destinados a medir la temperatura, ya sea del agua de alimentación, del vapor, de los gases de la combustión del petróleo, etc. [33]



Fig 13. Ejemplo de Indicador de Temperatura.[33]

**5.3.2 Componentes de seguridad**

- *Válvulas de seguridad*

Tiene por objeto dar salida al vapor de la caldera cuando ésta sobrepasa a la presión máxima de trabajo ver Fig. 14.

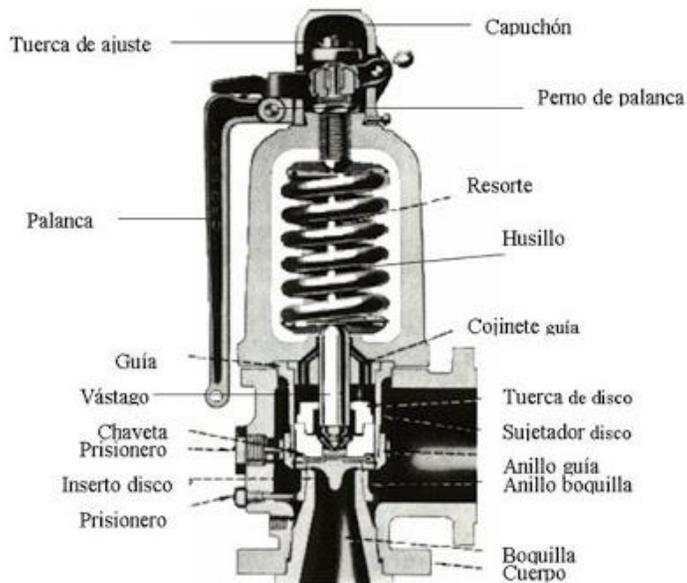


Fig 14. Partes de una Válvula de seguridad.[34]

- *Tapón fusible*

El tapón fusible, es un elemento que permite el paso de vapor y agua hacia el hogar, cuando el nivel de agua en la caldera baja más allá del mínimo permitido.

- *Alarmas*

Toda caldera dispondrá de un sistema de alarma, acústica o visual, que funcione cuando el nivel de agua alcance el mínimo o el máximo, deteniendo a la vez, el funcionamiento del sistema de combustión, cuando se alcance el nivel mínimo de agua. [34]

### 5.3.3 Componentes de Alimentación de Agua

- *Bombas*

Este accesorio al igual que el inyector, nos permite reponer el agua que se ha vaporizado en el interior de la caldera. Entre éstas, tenemos las Bombas Centrífugas y las de émbolo.



Fig 15. Bomba Monofásica. [35]

- *Inyectores*

Los inyectores, son dispositivos que funcionan con el mismo vapor que produce la caldera y son capaces de descargar agua a una presión mayor que la presión interna de la caldera.

### 5.3.4 Componentes de Limpieza

- *Puertas de inspección*

Según sus dimensiones se llaman puertas de hombre o tapas de registro. Éstas últimas sólo permiten el paso de un brazo. Ambas puertas sirven para efectuar limpiezas o inspecciones en el interior de los colectores principales o de los tubos según sea su ubicación.

- *Llaves de purga*

Entre las llaves de purga, se pueden distinguir las válvulas de extracción de fondo y las de extracción de superficie. La primera de ellas va ubicada en las partes más bajas de la caldera y sirven para extraer los lodos o barros provenientes de la vaporización de las aguas duras y acción del uso de los desincrustantes.

### 5.3.5 Componentes de Control

#### - Retardadores

Consisten en una plancha lisa, del mismo ancho que el diámetro interior del tubo, torcida en forma de hélice, la que se mete en el tubo de caldera. Los gases calientes tienen ahora que recorrer un camino mayor, siendo más lento el paso de ellos por el interior de los tubos y entregando mayor cantidad de calor al agua. La eficiencia de la caldera se aumenta entre un 2 % y 8 % con el uso de retardadores.

#### - Presostatos

Son accesorios que funcionan sobre la base de la máxima y mínima presión de trabajo de la caldera. Actúan sobre el quemador, apagándolo al llegar a la máxima presión para lo cual fue regulado y encendiéndolo al alcanzar la mínima presión deseada.

#### - Termostato

Son accesorios que funcionan de acuerdo a la temperatura del agua. Apagan el quemador cuando se obtiene la máxima temperatura para la cual fue regulada.

#### - Control de nivel de agua

Los controles de nivel de agua, tienen por objeto controlar que el agua, dentro de la caldera, se mantenga en un valor o en un rango pre-determinado. [35]

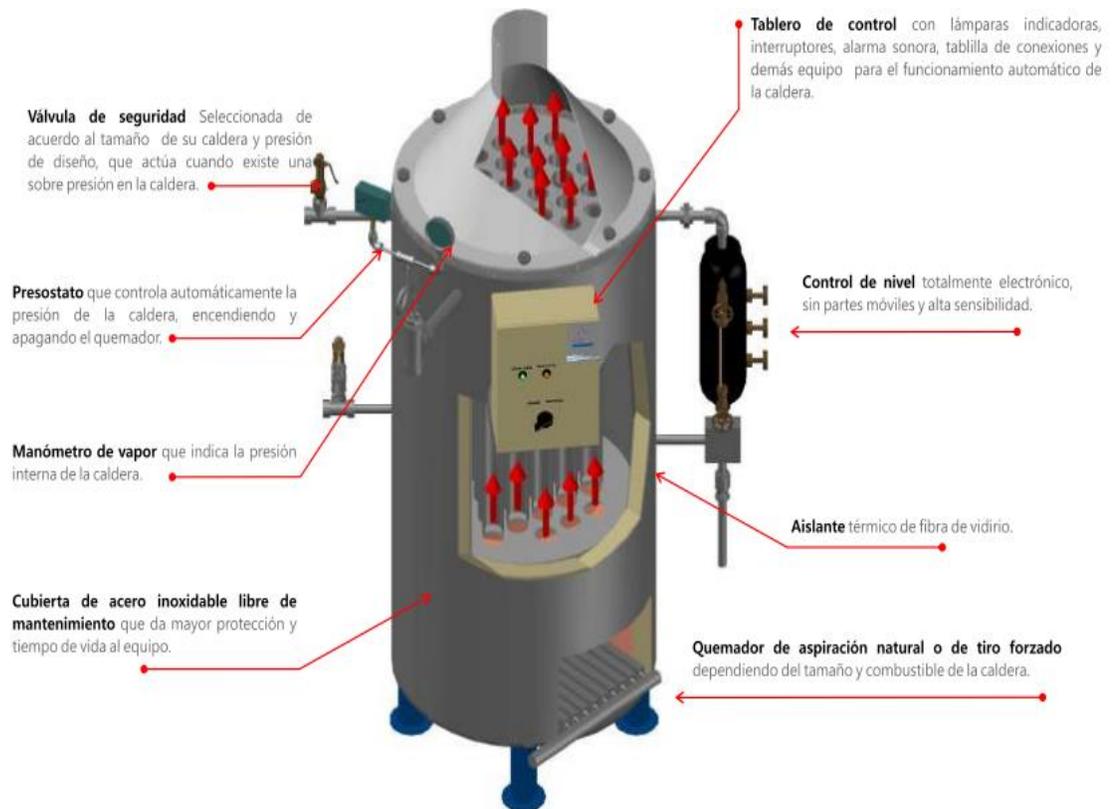


Fig 16. Partes Principales en una Caldera.[35]

### **5.3.6 Procedimiento de calderas para la producción de vapor**

Para pasar el agua desde fase líquida a vapor, es necesario añadir calor con el fin de aumentar la temperatura hasta su punto de ebullición. Este calor que eleva la temperatura del agua se conoce como calor sensible. La temperatura de ebullición del agua es de 100°C a presión atmosférica, aumentando cuando la presión aumenta y viceversa. La temperatura de ebullición del agua se conoce también como temperatura de saturación del vapor producido. Las relaciones entre temperatura de saturación y presión son propiedades termodinámicas fijas del vapor.

Cuando empieza la conversión de agua a vapor, la temperatura se mantiene constante, aunque se siga añadiendo calor. El fluido se encuentra a las condiciones de saturación presión/temperatura durante toda la conversión del agua a vapor. El calor que se añade durante el periodo de conversión del agua a vapor se conoce como calor latente de vaporización.

La cantidad de calor suministrado al vapor incluye el calor sensible<sup>2</sup> y el calor latente<sup>3</sup> de vaporización. Generalmente, cuando la presión del vapor saturado aumenta, la cantidad necesaria de calor sensible aumenta y la cantidad de calor latente disminuye.

Añadiendo calor sensible adicional al vapor saturado, aumenta la temperatura por encima de la correspondiente a la de saturación. El vapor que se obtiene por encima de la temperatura de saturación se conoce con el nombre de vapor recalentado. El recalentamiento aumenta la entalpía del vapor, o lo que es igual, su calor total. El recalentamiento también produce la expansión del vapor, aumentando su volumen específico. [36]

## **5.4 Válvulas de Control**

La válvula de control es el elemento final más utilizado en los sistemas de regulación de los procesos industriales, jugando un papel fundamental en el lazo de control. La selección, por tanto, del tipo adecuado de válvula para una tarea específica, basada en la naturaleza del control requerido, así como en las condiciones bajo las cuales tiene que funcionar, es una tarea que exige un análisis y un conocimiento detallado de la tecnología existente.

En el control automático de los procesos industriales, la válvula de control juega un papel muy importante en el bucle de regulación. Realiza la función de variar el

---

<sup>2</sup> Calor sensible es aquel que recibe un cuerpo o un objeto y hace que aumente su temperatura sin afectar su estructura molecular y por lo tanto su estado.

<sup>3</sup> El calor latente es la energía requerida por una cantidad de sustancia para cambiar de fase, de sólido a líquido (calor de fusión) o de líquido a gaseoso (calor de vaporización). Se debe tener en cuenta que esta energía en forma de calor se invierte para el cambio de fase y no para un aumento de la temperatura.

caudal de fluido de control que modifica, a su vez, el valor de la variable medida, comportándose como un orificio de área continuamente variable. [29]

Las válvulas se clasifican básicamente según sea el diseño del cuerpo y el movimiento del obturador. Siendo estas: válvulas con obturador de movimiento lineal y válvulas con obturador de movimiento rotativo.

#### **5.4.1 Válvulas de movimiento lineal.**

Las válvulas de movimiento lineal, en las que el obturador se mueve en la dirección de su propio eje, se clasifican en válvula de globo, válvula en ángulo, válvula de tres vías mezcladora o diversora, válvula de jaula, válvula de compuerta, válvula en Y, válvula de cuerpo partido, válvula Saunders y válvula de compresión.

##### **- Válvula de globo**

Llamada así por disponer de un obturador en forma de globo, se caracteriza porque el flujo de entrada o salida es perpendicular al eje del obturador. Pueden verse en las Fig. 17a, 17b y 17c y son de *simple asiento*, de *doble asiento* y de *obturador equilibrado*.

*Válvulas de simple asiento*, que cierran en contra de la presión del proceso, precisan de un actuador de mayor tamaño. Por lo tanto, se emplean cuando la presión diferencial del fluido es baja y se precisa que las fugas, a través de la válvula con el obturador en posición de cierre, sean mínimas. El cierre estanco se logra con asientos provistos de una arandela de teflón o de otros materiales blandos.

*Válvulas de doble asiento*, llamadas también de simple asiento con obturador equilibrado, la fuerza de desequilibrio desarrollada por el fluido a través del obturador es menor que en la válvula de simple asiento. Esto es debido a que, en la válvula de doble asiento, el fluido actúa en sentidos contrarios sobre los Obturadores.

*Válvulas con obturador equilibrado*, lo hace por encima y por debajo del único obturador. Por este motivo se emplean en válvulas de gran tamaño o bien cuando deba trabajarse con una alta presión diferencial. En posición de cierre, las fugas a través de la válvula son mayores que en una válvula de simple asiento, debido a que es mecánicamente imposible que el doble obturador asiente perfectamente sobre los dos asientos.

##### **- Válvula en ángulo**

En la Fig. 17d presenta un flujo de salida perpendicular al flujo de entrada con un recorrido menos curvilíneo que en una válvula de globo, por lo que permite obtener un flujo de caudal regular sin excesivas turbulencias y es, además, adecuada para sustituir a una válvula de globo cuando el fluido circula con sólidos en suspensión o a excesiva velocidad provocada por una alta presión diferencial de trabajo.

- **Válvula de tres vías**

La *válvula de tres vías* se emplea generalmente para mezclar fluidos –válvulas *mezcladoras* Fig. 17e – o bien para derivar, de un flujo de entrada, dos de salida – válvulas *diversoras* Fig. 17f. Las válvulas de tres vías intervienen típicamente en el control de temperatura de intercambiadores de calor, facilitando un control muy rápido de la temperatura, gracias a que el fluido de calefacción (vapor o fluido térmico) puede derivar, a través de la válvula, sin pasar por el intercambiador.

- **Válvula de jaula**

La *válvula de jaula* Fig. 17g recibe esta denominación por la forma de jaula que tiene, bien con los orificios dispuestos en una jaula fija en cuyo interior desliza el obturador, en cuyo caso se denomina *válvula de jaula fija* o bien con orificios en el obturador, en cuyo caso se denomina *válvula de jaula móvil*.

- **Válvula de compuerta**

Esta válvula Fig. 17h, denominada también válvula de tajadera, efectúa su cierre con un disco vertical plano, o de forma especial, y que se mueve verticalmente al flujo del fluido. Por su disposición es adecuada generalmente para control toda-nada, ya que en posiciones intermedias tiende a bloquearse.

- **Válvula en Y**

La *válvula en Y* Fig. 17i tiene el asiento y el obturador inclinados 45° respecto al flujo del fluido. Si el cuerpo está instalado en horizontal, es difícil desmontar las partes internas inclinadas 45° con relación a la horizontal. Es adecuada como válvula de cierre y de control. Como válvula todo-nada se caracteriza por su baja pérdida de carga y como válvula de control presenta una gran capacidad de caudal.

- **Válvula de cuerpo partido**

Esta válvula Fig. 17j es una modificación de la válvula de globo de simple asiento teniendo el cuerpo partido en dos partes entre las cuales está presionado el asiento. Esta disposición permite una fácil sustitución del asiento y facilita un flujo suave del fluido sin espacios muertos en el cuerpo. Se emplea principalmente para fluidos viscosos y en la industria alimentarla.

- **Válvula Saunders**

En la válvula *Saunders* o de *diafragma* Fig. 17k el obturador es una membrana flexible que, a través de un vástago unido a un servomotor, es forzada contra un resalte del cuerpo, que actúa de asiento, cerrando así el paso del fluido. La válvula se caracteriza porque el cuerpo puede revestirse fácilmente de goma o de plástico para trabajar con fluidos agresivos. Tiene la desventaja de que el servomotor de accionamiento debe ser muy potente. Se utiliza principalmente en procesos químicos difíciles, en particular en el manejo de fluidos negros o agresivos o bien en el control de fluidos conteniendo sólidos en suspensión.

- **Válvula de manguito**

La *válvula de manguito*, que puede verse en la Fig. 17l, funciona mediante el pinzamiento de dos o más elementos flexibles, por ejemplo un tubo de goma. Igual

que las válvulas Saunders o de diafragma, se caracteriza porque proporciona un óptimo control en posición de cierre parcial y se aplica, fundamentalmente, en el manejo de fluidos negros corrosivos, viscosos o conteniendo partículas sólidas en suspensión. [29]

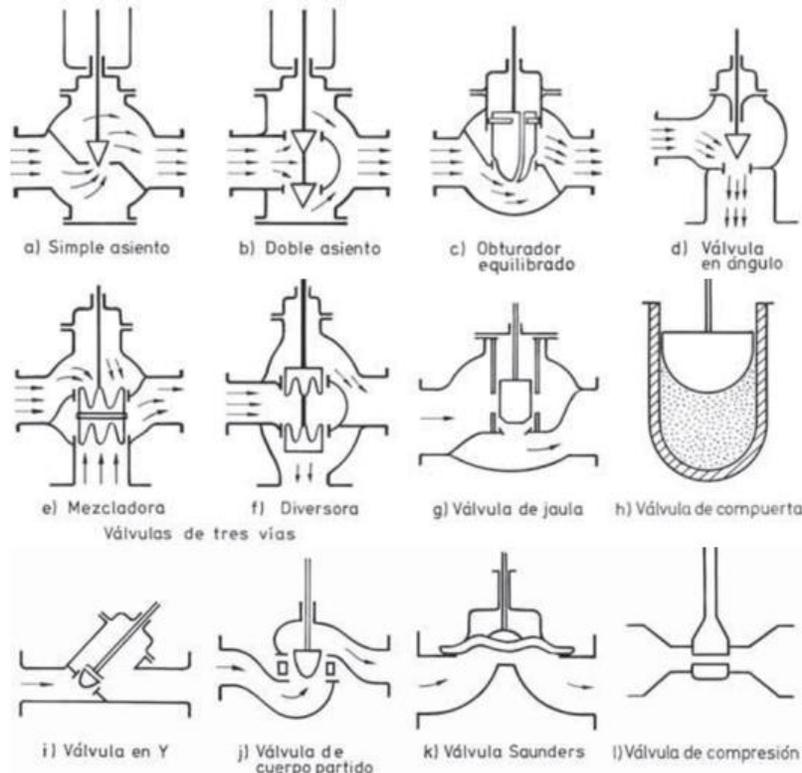


Fig 17. Válvulas de movimiento lineal. [29]

#### 5.4.2 Válvulas de movimiento circular.

Las válvulas en las que el obturador tiene un movimiento rotativo se clasifican en: válvula de disco excéntrico rotativo (Camflex), válvula de obturador cilíndrico excéntrico, válvula de mariposa, válvula de bola, válvula de macho, válvula de orificio ajustable y válvula de flujo axial.

##### - Válvula de disco excéntrico (Camflex)

Esta válvula Fig. 18m consiste en un obturador de superficie segmentada esférica que se auto alinea durante el movimiento de giro excéntrico, proporcionando un cierre estanco y unas bajas fuerzas dinámicas del fluido. Esto permite utilizar un servomotor de par reducido. Por otra parte, la tapa de extensión permite la operación para un margen amplio de temperaturas.

##### - Válvula de obturador cilíndrico excéntrico

Esta válvula (Fig. 18n) tiene un obturador cilíndrico excéntrico que asienta contra un cuerpo cilíndrico. El cierre hermético se consigue con un revestimiento de goma o de teflón en la cara del cuerpo donde asienta el obturador. La válvula es de bajo coste y es adecuada para fluidos corrosivos o conteniendo sólidos en suspensión.

- **Válvula de mariposa**

En la *válvula de mariposa* (Fig. 18o) el cuerpo está formado por un anillo cilíndrico dentro del cual gira transversalmente un disco circular. La válvula puede cerrar herméticamente mediante un anillo de goma encastrado en el cuerpo. Un servomotor exterior acciona el eje de giro del disco y ejerce su par máximo cuando la válvula está casi cerrada (en control todo-nada se consideran  $90^\circ$  y en control continuo  $60^\circ$ , a partir de la posición de cierre, ya que la última parte del giro es bastante inestable), siempre que la presión diferencial permanezca constante. En la selección de la válvula es importante considerar las presiones diferenciales correspondientes a las posiciones de completa apertura y de cierre; se necesita una fuerza grande del actuador para accionar la válvula en caso de una caída de presión elevada.

- **Válvula de bola**

En estas válvulas, el cuerpo tiene una cavidad interna esférica que alberga un obturador en forma de esfera o de bola (de ahí su nombre). La bola tiene un corte adecuado (usualmente en V) que baja la curva característica de la válvula, y gira transversalmente accionada por un servomotor exterior. El cierre estanco se logra con un aro de teflón incorporado al cuerpo contra el cual asienta la bola cuando la válvula está cerrada. En posición de apertura total, la válvula equivale aproximadamente al 75% del tamaño de la tubería. La válvula de bola se emplea principalmente en el control de caudal de fluidos negros, o bien en fluidos con gran porcentaje de sólidos en suspensión.

- **Válvula de macho**

Es una válvula de bola típica (Fig. 18q) que consiste en un macho u obturador de forma cilíndrica o troncocónica con un orificio transversal igual al diámetro interior de la tubería. El macho ajusta en el cuerpo de la válvula y tiene un movimiento de giro de  $90^\circ$ . Se utiliza generalmente en el control manual todo-nada de líquidos o gases y en regulación de caudal.

- **Válvula de orificio ajustable**

En la *válvula de orificio ajustable* el obturador consiste en una camisa de forma cilíndrica que está perforada con dos orificios, uno de entrada y otro de salida, y que gira mediante una palanca exterior accionada manualmente o por medio de un servomotor. El giro del obturador tapa, parcial o totalmente, las entradas y salidas de la válvula, controlando así el caudal. La válvula incorpora, además, una tajadera cilíndrica que puede deslizarse dentro de la camisa gracias a un macho roscado de accionamiento exterior (Fig. 18r).

- **Válvula de flujo axial**

Las *válvulas de flujo axial* consisten en un diafragma accionado neumáticamente que mueve un pistón, el cual, a su vez, comprime un fluido hidráulico contra un obturador formado por un material elastómero. De este modo, el obturador se expansiona para cerrar el flujo anular del fluido. Este tipo de válvulas se emplea

para gases y es especialmente silencioso. Otra variedad de la válvula de flujo axial es la válvula de manguito, que es accionada por compresión exterior del manguito a través de un fluido auxiliar a una presión superior a la del propio fluido. Se utiliza también para gases (ver Fig. 18s). [29]

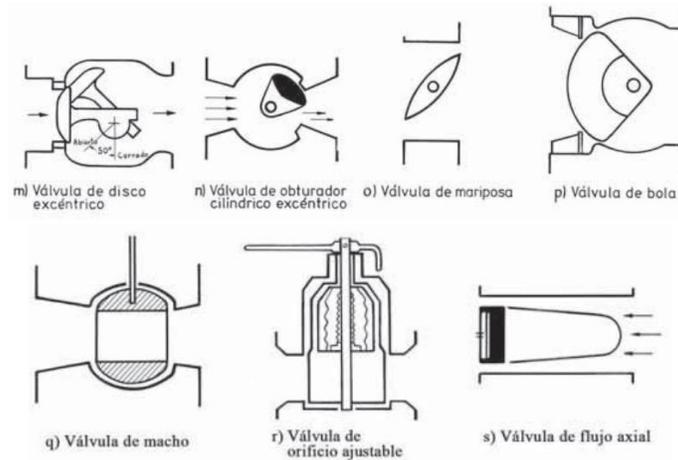


Fig 18. Válvulas de movimiento rotativo.[29]

### 5.4.3 Características Inherentes de la Válvula.

La característica de un fluido incompresible fluyendo en condiciones de presión diferencial constante a través de la válvula se denomina característica de caudal inherente y se representa, usualmente, considerando como abscisas la carrera del obturador de la válvula y, como ordenadas, el porcentaje de caudal máximo bajo una presión diferencial constante. Las curvas características más significativas son: apertura rápida, lineal e isoporcentual.

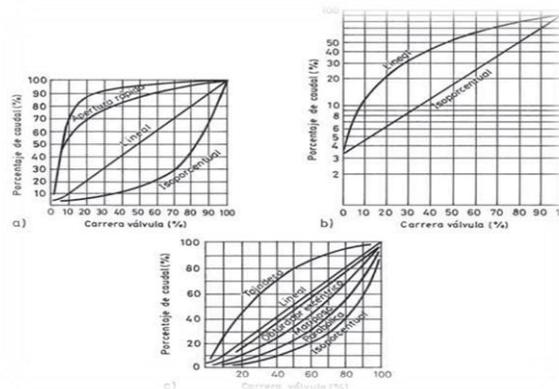


Fig 19. Curvas características.[29]

Las curvas características se obtienen mecanizando el obturador para que, al variar la carrera, el orificio de paso variable existente entre el contorno del obturador y el asiento configure la característica de la válvula. En la fig. 19 pueden verse varios tipos de obturadores cuya forma y mecanización determinan esta característica.

Según las curvas características nuestra válvula es de tipo lineal, En el obturador con característica lineal, En el obturador con característica lineal (Fig. 19b, 19f, 19g, 19i), el caudal es directamente proporcional a la carrera según la ecuación:

$$q = K * l \tag{1}$$

En la que:

q = caudal a pérdida de carga constante

K = constante

l = carrera de la válvula. [29]

## 5.5 Control de una Caldera

### 5.5.1 Control en lazo abierto

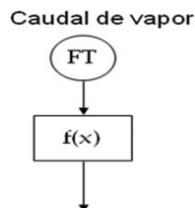


Fig 20. Control Feedforward.[37]

Este control, también llamado “feedforward”, consiste en asociar directamente una demanda determinada para cada caudal de vapor medido mediante la programación de una curva que relaciona ambas. La respuesta de este lazo de control, si bien es rápida, depende de la precisión de la curva programada. El gran inconveniente es que las condiciones de la caldera no siempre son las mismas que cuando se configuró la curva, y esto produce un error en el comportamiento de este control.

### 5.5.2 Control de demanda en lazo cerrado a un elemento

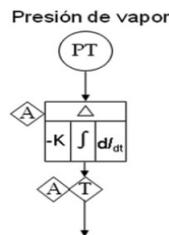


Fig 21. Control Feedback.[37]

Este control, también llamado “feedback” representación gráfica en la Fig. 16, utiliza la medición de presión de vapor para relacionar el suministro de vapor a la demanda o necesidad de vapor del proceso.

Este control es sencillo e intuitivo. A medida que el proceso demanda más vapor, la presión de vapor en el colector de salida disminuye, demandando más producción de vapor a la caldera.

De la misma forma, si la presión de vapor en el colector de salida aumenta, el controlador demanda menos carga a la caldera. El control de demanda a un elemento mejora en mucho las prestaciones del control en lazo abierto. Responde bien en aquellos casos en los que el consumo de vapor del proceso es estable, es decir, no requiere variaciones de caudal rápidas. Por el contrario, en procesos donde el consumo de vapor varía significativamente de forma rápida, este control es lento. Esto es debido a que el controlador, para tomar una acción correctiva, requiere que exista una desviación entre el punto de consigna y la variable controlada, en este caso la presión de vapor. Esto sucede un tiempo después de que el caudal de vapor haya sufrido una variación.

Un caso de procesos donde este control no responde lo suficientemente bien es en procesos por lotes (tipo Batch), donde la apertura de una válvula puede suponer un aumento de hasta el 10% en la demanda de vapor.

Este control es válido en procesos de una caldera que alimenta a una turbina de vapor, siempre y cuando la configuración sea “turbina sigue a caldera”, es decir, es la turbina de vapor la que adapta su carga en función de la producción de vapor de la caldera. En casos de configuración “caldera sigue a turbina”, es decir, la caldera regula su producción en función de la carga que se ha seleccionado para la turbina, este control tampoco es suficiente.

### 5.5.3 Control de demanda en lazo cerrado a dos elementos

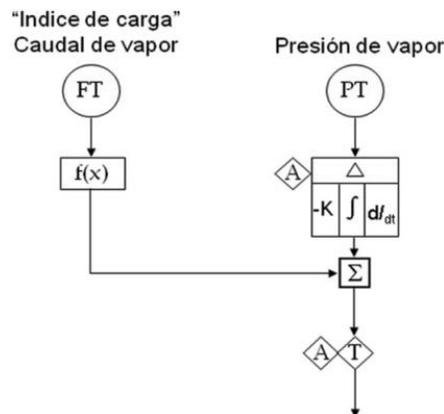


Fig. 22. Control en Combinación feedback + feedforward.[37]

En los dos casos indicados anteriormente, en los que el control a un elemento no es suficiente, la solución es aplicar este control, también llamado “feedback + feedforward”. Este control utiliza un llamado “índice de carga”, que se suma a la señal de salida del controlador de presión para entregar la señal de salida de demanda. Este índice puede ser el caudal de vapor.

Este control añade la ventaja de rapidez del control en lazo abierto sin estar penalizado el hecho de tener que depender de una curva caudal de vapor/demanda que puede no cumplirse. La señal de demanda se posiciona de acuerdo a la curva programada, a la que posteriormente el controlador realiza ajustes, incrementando o disminuyendo la señal. En el caso ideal de que el proceso respondiera exactamente igual que la curva programada, el factor de corrección del controlador sería nulo. [37]

## **5.6 Sistemas de control**

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; por lo que un sistema de control se puede definir como un sistema que reciben una entrada y con base en esta refleja una salida. Este se caracteriza por tener elementos de control que permiten realizar un control sobre un sistema, es decir conseguir dominio sobre las variables de salida dependiendo de los datos obtenidos por sus elementos de medición.

Antes de analizar los sistemas de control, deben definirse ciertos términos básicos.

*Variable controlada y señal de control o variable manipulada.* La variable controlada es la cantidad o condición que se mide y controla. La señal de control o variable manipulada es la cantidad o condición que el controlador modifica para afectar el valor de la variable controlada. Normalmente, la variable controlada es la salida del sistema. Controlar significa medir el valor de la variable controlada del sistema y aplicar la variable manipulada al sistema para corregir o limitar la desviación del valor medido respecto del valor deseado.

*Plantas.* Una planta puede ser una parte de un equipo, cuyo objetivo es efectuar una operación particular. Se podría llamar entonces a la planta a cualquier objeto físico que se va a controlar (como un dispositivo mecánico, un horno de calefacción, un reactor químico o una nave espacial).

*Procesos.* El Diccionario Merriam-Webster define un proceso como una operación o un desarrollo natural progresivamente continuo, marcado por una serie de cambios graduales que se suceden unos a otros de una forma relativamente fija y que conducen a un resultado o propósito determinados; o una operación artificial o voluntaria que se hace de forma progresiva y que consta de una serie de acciones o movimientos controlados, sistemáticamente dirigidos hacia un resultado o propósito determinado. En este libro se llamará proceso a cualquier operación que se va a controlar. Algunos ejemplos son los procesos químicos, económicos y biológicos.

*Sistemas.* Un sistema es una combinación de componentes que actúan juntos y realizan un objetivo determinado. Un sistema no está necesariamente limitado a los sistemas físicos. El concepto de sistema se puede aplicar a fenómenos abstractos y dinámicos, como los que se encuentran en la economía. Por tanto, la palabra

sistema debe interpretarse en un sentido amplio que comprenda sistemas físicos, biológicos, económicos y similares.

*Perturbaciones.* Una perturbación es una señal que tiende a afectar negativamente el valor de la salida de un sistema. Si la perturbación se genera dentro del sistema se denomina interna, mientras que una perturbación externa se genera fuera del sistema y es una entrada.

*Control realimentado.* El control realimentado se refiere a una operación que, en presencia de perturbaciones, tiende a reducir la diferencia entre la salida de un sistema y alguna entrada de referencia, y lo realiza tomando en cuenta esta diferencia. Aquí sólo se especifican con este término las perturbaciones impredecibles, ya que las perturbaciones predecibles o conocidas siempre pueden compensarse dentro del sistema.

- *Sistemas de control realimentados*

Un sistema que mantiene una relación determinada entre la salida y la entrada de referencia, comparándolas y usando la diferencia como medio de control, se denomina sistema de control realimentado. Un ejemplo sería el sistema de control de temperatura de una habitación. Midiendo la temperatura real y comparándola con la temperatura de referencia (temperatura deseada), el termostato activa o desactiva el equipo de calefacción o de enfriamiento para asegurar que la temperatura de la habitación se mantiene en un nivel confortable independientemente de las condiciones externas.

- *Sistemas de control en lazo cerrado*

Los sistemas de control realimentados se denominan también sistemas de control en lazo cerrado. En la práctica, los términos control realimentado y control en lazo cerrado se usan indistintamente. En un sistema de control en lazo cerrado, se alimenta al controlador la señal de error de actuación, que es la diferencia entre la señal de entrada y la señal de realimentación (que puede ser la propia señal de salida o una función de la señal de salida y sus derivadas y/o integrales), con el fin de reducir el error y llevar la salida del sistema a un valor deseado ver Figura 4. El término control en lazo cerrado siempre implica el uso de una acción de control realimentado para reducir el error del sistema.

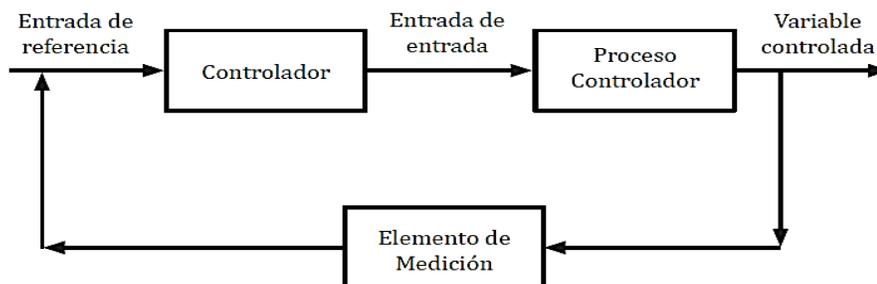


Fig 23. Diagrama de bloques de un sistema de lazo cerrado.[11]

- *Sistemas no lineales*

Un sistema es no lineal si no se aplica el principio de superposición. Por tanto, para un sistema no lineal la respuesta a dos entradas no puede calcularse tratando cada entrada a la vez y sumando los resultados. Aunque muchas relaciones físicas se representan a menudo mediante ecuaciones lineales, en la mayor parte de los casos las relaciones reales no son verdaderamente lineales. De hecho, un estudio cuidadoso de los sistemas físicos revela que incluso los llamados «sistemas lineales» sólo lo son en rangos de operación limitados. En la práctica, muchos sistemas electromecánicos, hidráulicos, neumáticos, etc., involucran relaciones no lineales entre las variables. Por ejemplo, la salida de un componente puede saturarse para señales de entrada grandes. Puede haber una zona muerta que afecte a las señales pequeñas. (La zona muerta de un componente es un rango pequeño de variaciones de entrada a las cuales el componente es insensible.) Puede ocurrir una no linealidad de la ley cuadrática en algunos componentes. Por ejemplo, los amortiguadores que se utilizan en los sistemas físicos pueden ser lineales para operaciones a baja velocidad, pero pueden volverse no lineales a altas velocidades, y la fuerza de amortiguamiento puede hacerse proporcional al cuadrado de la velocidad de operación.

- *Sistemas de control Tiempo discreto*

Un sistema en tiempo discreto es un operador matemático que transforma una señal en otra por medio de un grupo fijo de reglas y funciones. La notación  $T[.]$ , es usada para representar un sistema general, tal como se muestra en la Figura 5. En el cual, una señal de entrada  $x(n)$  es transformada en una señal de salida  $y(n)$  a través de la transformación  $T[.]$ . Las propiedades de entrada-salida de cada sistema puede ser especificado en algún número de formas diferentes.



Fig 24. Sistema general en tiempo discreto.[11]

Los sistemas de control en tiempo discreto toman muestras de los datos en determinado tiempo, fuera de estos tiempos el sistema no tiene información alguna del proceso controlado, estos se encuentran en forma de pulsos digitales, cuando se habla de un sistema de control digital se hace la referencia a la utilización de un sistema de procesamiento, es decir, PLC, DCS, Computador etc. [11]

### 5.7 Modelamiento de los sistemas

Un sistema representa una unidad donde se hacen tratamientos físicos o químicos de materiales que puede ser contrastada con un modelo que representa una descripción matemática del sistema real. La disposición de varios sistemas unidos entre sí por flujos comunes de materiales y/o información constituye un proceso. El estudio de un proceso, mediante la manipulación de su representación matemática o de su modelo físico, constituye una simulación. Los estudios clásicos de un proceso en estado estacionario se complementan con un análisis dinámico, lo que

exige un conocimiento de los criterios de estabilidad y de los métodos de operación para evaluar exitosamente el funcionamiento del proceso.

### **5.7.1 Método de obtención de modelos**

Existen dos métodos principales para obtener el modelo de un sistema:

- **Modelado Teórico:** Se trata de un método analítico, en el que se recurre a leyes básicas de la física para describir el comportamiento dinámico de un fenómeno o proceso.
- **Identificación del sistema:** se trata de un método experimental que permite obtener el modelo de un sistema a partir de datos reales recogidos de la planta bajo estudio.

El modelado teórico tiene un campo de aplicación restringido a procesos muy sencillos de modelar, o a aplicaciones en que no se requiera gran exactitud en el modelo obtenido. En muchos casos, además, la estructura del modelo obtenido a partir del conocimiento físico de la planta posee un conjunto de parámetros desconocidos y que sólo se pueden determinar experimentando sobre el sistema real. De ahí la necesidad de recurrir a los métodos de identificación de sistemas.

## **5.8 Identificación de los sistemas**

Se entiende por identificación de sistemas a la obtención de forma experimental de un modelo que reproduzca con suficiente exactitud, para los fines deseados, las características dinámicas del proceso objeto de estudio.

En términos generales, el proceso de identificación comprende los siguientes pasos:

- Obtención de datos de entrada - salida.
- Tratamiento previo de los datos registrados.
- Elección de la estructura del modelo.
- Obtención de los parámetros del modelo.
- Validación del modelo.

Dependiendo de la causa estimada, deberá repetirse el proceso de identificación desde el punto correspondiente. Por tanto, el proceso de identificación es un proceso iterativo.

### **5.8.1 Métodos de identificación de sistemas**

Para la identificación de sistemas existen diferentes métodos, para identificar el comportamiento de sistemas algunos de los diferentes criterios son:

#### **Método Paramétrico**

En este método se hace una elección o aproximación a una posible estructura y un número finito de parámetros que relacionan las señales del sistema (entrada, salida y perturbaciones), debido a un amplio estándar de sistemas dinámicos que permiten

describir el comportamiento de cualquier sistema lineal. Los modelos paramétricos se describen en el dominio discreto, estos modelos se basan en la obtención de datos por muestreo que sirven de base para la identificación. En el caso que se requiere un modelo continuo, siempre es posible realizar una transformación del dominio discreto al continuo.

La expresión general de un modelo discreto es (2)

$$st = nt + w(t) \quad (2)$$

Dónde:

- $w(t)$  = modela la salida debido a perturbaciones.
- $n(t)$  = salida debido a la señal de entrada.
- $s(t)$  = salida medible del sistema.

En las siguientes ecuaciones se puede observar otra expresión para cada una de las variables anteriores (3), (4) y (5).

$$n(t) = G(q^{-1}, \theta) \cdot u(t) \quad (3)$$

$$w(t) = H(q^{-1}, \theta) \cdot e(t) \quad (4)$$

$$s(t) = A(q^{-1}, \theta) \cdot y(t) \quad (5)$$

Dónde:

- $(q^{-1})$ ; Operador de retardo
- $\theta$ ; Vector de parámetros
- $u(t)$ ; Entradas del sistema
- $e(t)$ ; Ruido de la entrada
- $y(t)$ ; Salida de interés del sistema

Tanto,  $G(q^{-1}, \theta)$  como  $H(q^{-1}, \theta)$  son cocientes de polinomios del tipo:

$$G(q^{-1}, \theta) = \frac{B(q^{-1})}{F(q^{-1})} = \frac{b_1 \cdot q^{-nk} + b_2 \cdot q^{-nk-1} + \dots + b_{nb} \cdot q^{-nk-nb+1}}{1 + f_1 \cdot q^{-1} + \dots + f_{nf} \cdot q^{-nf}} \quad (6)$$

$$H(q^{-1}, \theta) = \frac{c(q^{-1})}{D(q^{-1})} = \frac{1+c_1 \cdot q^{-1} + \dots + c_{nc} \cdot q^{-nc}}{1+d_1 \cdot q^{-1} + \dots + d_{nd} \cdot q^{-nd}} \quad (7)$$

Y  $A(q^{-1}, \theta)$  un polinomio del tipo:

$$A(q^{-1}, \theta) = 1 + a_1 \cdot q^{-1} + \dots + b_{na} \cdot q^{-na} \quad (8)$$

El vector de parámetros  $\theta$  contiene los coeficientes  $\theta_0, \theta_1, \theta_2, \theta_3$  y  $\theta_4$  de las funciones de transferencia anteriores. La estructura genérica de estos modelos es por tanto:

$$\begin{aligned}
 A(q^{-1}) \cdot y(t) &= G(q^{-1}, \theta) \cdot u(t) + H(q^{-1}, \theta) \cdot e(t) \\
 &= \frac{B(q^{-1})}{F(q^{-1})} \cdot u(t) + \frac{C(q^{-1})}{D(q^{-1})} \cdot e(t)
 \end{aligned} \tag{9}$$

Para elegir la estructura de este tipo de modelos hay que determinar el orden de cada uno de los polinomios anteriores, es decir  $na, nb, nc, nd, nf$  y el retardo entre la entrada y la salida  $nk$ . Una vez elegidos estos valores, sólo queda determinar el vector de coeficientes  $\theta(a_i, b_i, c_i, d_i, f_i)$  que hacen que el modelo se ajuste a los datos de entrada - salida del sistema real.

En muchos casos, algunos de los polinomios descritos anteriormente no se incluyen en la descripción del modelo, dando lugar a los siguientes casos particulares

Tabla 2. Diferentes estructuras de modelos paramétricos.[11]

Tipo de modelo	Condición	Estructura resultante
Modelo ARX	$F(q^{-1})=D(q^{-1})=C(q^{-1})=1$	$A(q^{-1})y(t) = B(q^{-1}) \cdot u(t) + e(t)$
Modelo Output Error (OE)	$C(q^{-1})=D(q^{-1})=A(q^{-1})=1$	$y(t) = \frac{B(q^{-1})}{F(q^{-1})} \cdot u(t) + e(t)$
Modelo ARMAX	$F(q^{-1})=D(q^{-1})=1$	$A(q^{-1}) \cdot y(t) = B(q^{-1}) \cdot u(t) + C(q^{-1}) \cdot e(t)$
Modelo Box Jenkins (BJ)	$A(q^{-1})=1$	$y(t) = \frac{B(q^{-1})}{F(q^{-1})} \cdot u(t) + \frac{C(q^{-1})}{D(q^{-1})} \cdot e(t)$

En la Figura 25 se muestra el diagrama de bloques equivalente para cada uno de los modelos anteriores.

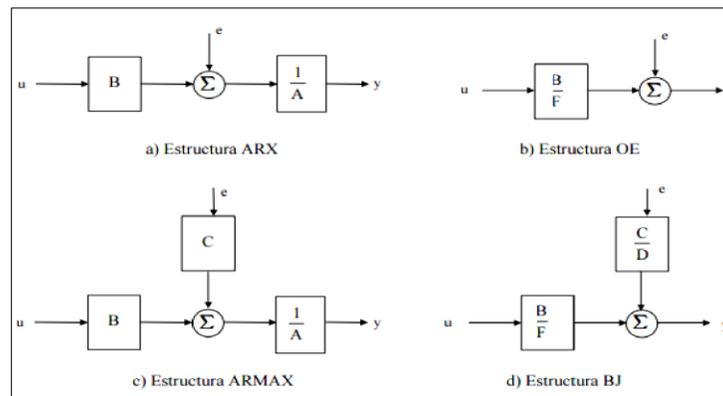


Fig 25. Diagrama de bloques modelos paramétricos.[11]

La anulación de alguno de los polinomios, resultando estructuras simplificadas, facilita el proceso de ajuste de parámetros. Cada una de las estructuras (ARX,

ARMAX, OE o BJ) tiene sus propias características y debe ser elegida fundamentalmente en función del punto en el que se prevé que se añade el ruido en el sistema. En cualquier caso, puede ser necesario ensayar con varias estructuras y con varios órdenes dentro de una misma estructura hasta encontrar un modelo satisfactorio.

### 5.8.2 Procesos a modelar

El control de nivel de un líquido, requiere de una serie de subsistemas, los cuales reciben y entregan señales que representan las variables del proceso, dichas señales son suministradas por sensores, ubicados en la entrada y la salida, con el fin de retroalimentar el sistema y conocer su evolución para tomar decisiones sobre como manipular las variables de control del sistema.

Modelar un proceso tiene como objetivo el obtener su función de transferencia, requiere de un total entendimiento del mismo, no es aconsejable abordar el asunto globalmente, lo cual lo hace muy complejo, en cambio es muy útil seguir una técnica que facilite el trabajo para obtener la función de transferencia, primero vamos a entender cómo funciona el proceso a modelar, para ello usaremos el siguiente esquema que representa un proceso clásico de control de nivel de líquido.[11]

#### - *Sistemas térmicos*

Los sistemas térmicos son aquellos que involucran la transferencia de calor de una sustancia a otra. Estos sistemas se analizan en términos de resistencia y capacitancia

- *R = Valor de la resistencia depende del modo en que se transfiere el calor.*

El flujo de calor neto depende si existe una diferencia de temperatura entre ellos. Si  $q$  es la razón de flujo de calor y  $T_2 - T_1$ , la diferencia de temperatura, en (10) se muestra razón de flujo de calor  $q$ .

$$q = \frac{T_2 - T_1}{R} \quad (10)$$

- *La capacitancia térmica es el almacenamiento de la energía interna en un sistema, si la razón de flujo de calor en el interior de un sistema es  $q_1$  y la razón de flujo de calor que sale es  $q_2$ , entonces:*

$$\text{La Tasa de calor de energía interna} = q_1 - q_2$$

Un incremento de la energía interna significa un incremento de la temperatura, es decir que

- *Cambio de la energía interna = masa\*capacidad calorífica\*cambio de temperatura en las (11) y (12) se muestra la tasa de cambio de energía.*
- *m = masa*

- $c = \text{capacidad calorífica específica}$

$$q_1 - q_2 = \frac{mcdT}{dt} \quad (11)$$

- $mc = \text{Capacitancia térmica } C \text{ entonces:}$

$$q_1 - q_2 = \frac{CdT}{dt} \quad (12)$$

### *Resistencia Calefactora*

Las resistencias caloríficas permiten que exista transferencia de calor sobre un líquido, como el líquido de la caldera es agua, se opta por las resistencias de tipo tubular y sumergible. Se debe elegir la resistencia para calentar cierto volumen de agua, basados en principios de termodinámica para transferencia de calor. (13) ecuación energía calorífica.

$$\Delta Q = mc\Delta T \quad (13)$$

Dónde:

- $\Delta Q = \text{Variación energía calorífica}$
- $m = \text{masa del agua}$
- $c = \text{calor específico del agua}$
- $\Delta T = \text{Variación de temperatura}$

Ver (14), densidad del agua.

$$\rho = \frac{m}{v} \quad (14)$$

Dónde:

- $\rho = \text{Densidad del agua}$
- $m = \text{masa del agua}$
- $v = \text{volumen del agua [11]}$

## **5.9 Controladores PID**

### **Estructura**

En primera instancia considerar un lazo de control con una entrada y una salida (SISO) de un grado de libertad:

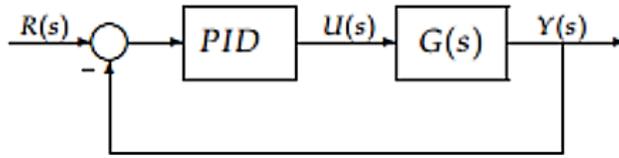


Fig 26. Diagrama en bloques de un sistema controlado con un PID.[11]

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

### 5.9.1 PID: Acción de control Proporcional – Integral - Derivativa

Esta acción combinada reúne las ventajas de cada una de las tres acciones de control individuales. La ecuación de un controlador con esta acción combinada se obtiene mediante:

$$u(t) = k_p e(t) \int_0^t e(\tau) d\tau + k_p T_d \frac{de(t)}{dt} \quad (15)$$

Y su función transferencia resulta:

$$C_{PID} = k_p \left( 1 + \frac{1}{T_i} + T_d s \right) \quad (16)$$

La estructura del controlador PID es simple y se basa en diferentes parámetros que se describen a continuación:

- Constante Proporcional ( $K_p$ ): da una salida del controlador que es proporcional al error, es decir:  $u(t) = K_p e(t)$ , que descrita desde su función de transferencia, (17)

$$C_p(s) = k_p (1) \quad (17)$$

Donde  $k_p$  es una ganancia proporcional ajustable. Un controlador proporcional puede controlar cualquier planta estable, pero posee desempeño limitado y error en régimen permanente (off-set).

- Constante Integral ( $K_i$ ): da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlar lento.

$$u(t) = k_i \int_0^t e(\tau) d\tau + C_i(s) \quad (18)$$

La señal de control ( $u$ ) tiene un valor diferente de cero cuando la señal de error ( $e$ ) es cero. Por lo que se concluye que dada una referencia constante, o perturbaciones, el error en régimen permanente es cero.

- Control Proporcional Integral (CPI): se define mediante (19)

$$su(t) = k_p e(t) + \frac{k_p}{T_i} \int_0^t e(\tau) d\tau \quad (19)$$

Donde  $T_t$  se denomina tiempo integral y es quien ajusta la acción integral. La función de transferencia resulta:

$$C_{PI}(s) = k_p \left( 1 + \frac{1}{T_i s} \right) \quad (20)$$

Para poder analizar el desempeño de los controladores PID, es oportuno citar las especificaciones de diseño que se requieren de un sistema de control. Las mismas se definen en relación a la respuesta  $y(t)$  de un sistema con una referencia tipo escalón unitario y son las siguientes:

**Exactitud.** Impuesto un cierto valor de referencia  $r(t)$  al cual queremos que el sistema controlado llegue - en este caso este valor es 1 - puede ocurrir que no lo alcance, permaneciendo un error de estado estacionario  $e_{ss} = r_{ss} - y_{ss}$ , entre la referencia o valor deseado y el valor de estado estacionario, que nos dice cuan exacto es el sistema de control.

**Velocidad de respuesta o tiempo de crecimiento.**  $t_r$ . Normalmente definida por medio del tiempo que el sistema tarda en llegar del 10% al 90% del valor de estado estacionario  $y_{ss}$ . Algunos autores prefieren el tiempo de retardo  $t_d$ , ya que este valor incorpora la información del retardo puro  $T_u$ . El mismo se define como el tiempo necesario para llegar al 50% del valor del estado estacionario  $y_{ss}$ .

**Sobreelongación o máximo sobreimpulso  $M_p$ .** Diferencia entre el valor máximo alcanzado y el valor de estado estacionario, la cual se pretende que no sobrepase un cierto porcentaje del valor estacionario.

**Tiempo de establecimiento  $T_s$ .** Tiempo que tarda la salida en entrar en una banda ubicada alrededor del valor de estado estacionario  $y_{ss}$  y que por lo general se define como un 2% o 5% del mismo.

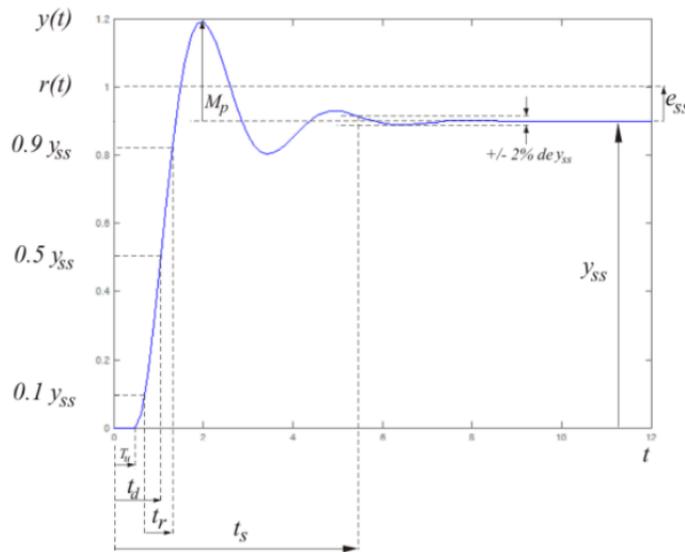


Fig 27. Especificación de un sistema de control en el dominio temporal.[11]

## 5.10 SoftPLC

SoftPLC o Software para PLC se define como un programa que se ejecuta en tiempo real en un computador ordinario e imita la operación de un controlador lógico programable Software que emula el funcionamiento de un PLC pero que se ejecuta en un hardware de PC. Se puede decir que esta reciente tecnología puede estar, o en el futuro próximo, compitiendo en ciertas aplicaciones con los PLC. Estos programas presentan ciertas características positivas entre las cuales se pueden mencionar:

- Menor coste que los grandes Sistemas de Control Distribuido (DCS).
- Facilidad en el reemplazo de componentes y repuestos para su puesta en función.
- Presentan una potencia elevada debido a los procesadores con los que son ejecutados (Pentium IV, etc.).
- Según sea la opinión es posible que la confiabilidad de un PC sea mayor que la de un PLC debido a que estos son hardware industrial con componentes pre-envejecidos, uso excesivo en tiempo, etc.

Las aplicaciones basadas en software de PLC bajo plataforma de PC adquieren muchas ventajas como incrementos de prestaciones, memoria de programa y estabilidad. En estos casos el control lo realiza un PC.

- En general hacen uso de hardware estándar.
- Presentan un entorno de programación al usuario.
- Son desarrollados bajo lenguajes específicos.
- La programación es análoga a un PLC.
- Poseen librerías especiales de tiempo real y determinístico.
- Existen aplicaciones donde un autómatas programable puede estar más limitado y un PC las puede realizar fácilmente.
- Las condiciones críticas de funcionamiento de un programa para PLC no son muy estrictas, presentan facilidad para adaptarse y escalar.

Existen trabajos previos en los que se han realizado sistemas de control bajo software libre y propietario y en donde la implementación de un autómatas programable es su objetivo. Entre estos se pueden mencionar los siguientes:

### 1. Bajo software propietario (Windows):

- VLC: Visual Logic Controller.
- PC WORX.
- TriLOG.
- RSLogix5000, etc.

### 2. Bajo software libre (Linux):

- Proyecto MatPLC.
- ISaGRAF para Linux.
- Classic Ladder.
- SoftPLC Company etc.

Desde hace tiempo han irrumpido en el mercado distintas soluciones de placas PC "Single Board Computer" (SBC), como **Arduino**, **Nanode**, **Wasmote**, **Raspberry Pi**, **Beaglebone Black**, **LinkIt ONE**, **Edison** que, lentamente, se están incorporando al mundo de la automatización.

Estos mini PC o placas son open-source hardware y muchas de ellas poseen lenguajes de programación de alto nivel (C/C++/C#, Java, Python, Ruby, PHP, Javascript, Node.js, Octave, etc.). En un momento como el actual, en que la **informática y la automatización industrial se dan la mano** con más fuerza que nunca, éstos son dispositivos que están presentes en la comunidad de profesionales y aficionados a la electrónica.

Estas placas PC disponen de un Hardware que no está pensado para entornos industriales, entornos duros tanto por las condiciones ambientales como por el ruido, que puede llegar a afectar a los equipos. A pesar de ello, sí que pueden ser utilizados por los profesionales de la automatización como una solución económica y de prestaciones limitadas, sobre todo en el campo de la formación. [38]

#### 5.10.1 SoftPLC para SBC.

Conocidas empresas en el mundo industrial del SoftPLC, como son **Codesys**, **KW Software (Phoenix Contact)**, **ISaGRAF** o **Copalp (Copadata)**, han visto el creciente impulso de las SBC y han lanzado productos adaptados a ellas.

**Codesys** ofrece "**CODESYS Control for Raspberry Pi SL**" que permite **convertir Raspberry Pi en un SoftPLC**. Incluye una versión DEMO operativa de 2 horas, lo cual supone para el mundo de la formación una solución muy económica y real de SoftPLC con el que adentrarse en el mundo de la programación en uno de los entornos más utilizados por los controladores industriales.



Fig 28. Paquete de Instalación para Raspberry con CODESYS.[39]

Por su parte, **KW Software** convertido ahora en **Phoenix Contact Software** presenta el **Starter Kit IEC 61131 sobre la base de la Raspberry Pi**. Un completo paquete de software para la puesta en marcha y pruebas de esta plataforma Raspberry Pi como "mini autómatas". El sistema de programación Multiprog se utiliza para crear el programa de PLC en el PC. El sistema runtime **Proconos embebido CLR** asume el procesamiento del programa creado en el Raspberry Pi.

Otro ejemplo del acercamiento de estas tecnologías a la industria es la solución **EMBEDDED PLC** que convierte la **SBC Rabbit Semiconductor BL2500**

en un **embedded Programmable Logic Controller (PLC)**, capaz de cumplir con los 5 lenguajes de programación IEC61131-3 con **ISaGRAF V3.50 Workbench**.

**Copalp con su Softlogic straton** basado en IEC-61131-3 para soluciones embedded dispone de straton runtime para Raspberry Pi, que permite programarlo según IEC 61131-3 en los 5 lenguajes de programación (ST, IL, LD, SFC, FBD). [39]

### 5.10.2 Módulo expansor I/O para SoftPLC

#### - *Raspberry Pi:*

Según sus fabricantes la Raspberry Pi es un ordenador de bajo costo del tamaño de una tarjeta de crédito, que se conecta a un monitor de ordenador o un televisor, y utiliza un teclado y un ratón estándar. Es un dispositivo pequeño que les permite a personas de todas las edades explorar la computación, y aprender a programar en lenguajes como lo son Python y Scratch. Es capaz de hacer todo lo que esperaríamos hacer con una computadora de escritorio, desde navegar por Internet y reproducción de vídeo de alta definición, hacer hojas de cálculo, usar procesadores de texto, y hasta divertirse con videojuegos.

#### - *Raspberry Pi 2 Modelo B*

Cuenta con 1Gb de RAM, usa un Quad-Core a 900MHz que a los que estamos acostumbrados a ella, nos va a resultar muy extraño tanta agilidad. Por otro lado, al cambiar su arquitectura, vamos a poder disfrutar de otros sistemas operativos más complejos, como el nuevo Windows 10, que tendrá una versión para desarrolladores gratuita en Raspberry Pi 2.



Fig 29. Raspberry Pi 2B.[40]

#### - *PSoC:*

Son microprocesadores que en la última década han tomado gran esfuerzo debido a sus grandes ventajas frente a los tradicionales microprocesadores, estos ofrecen una arquitectura que agiliza el análisis de información, estos procesan a una gran velocidad pueden llegar hasta 74 MHz, estas plataforma de diseño por sus prestaciones se han ido afianzando en aplicaciones como redes de sensores y transmisores inteligentes, puesto que estos dispositivos son de bajo costo comparado con las funcionalidades que tienen, puesto que estos tiene bloques digitales como temporizadores, contadores, protocolos de comunicación PWM, Además de eso cuentan con bloques analógicos como ADC, DAC, entre otros con los cuales se pueden ayudar al usuario a reducir circuitos y bajar costos.

Tabla 3. Comparación de características de PSoC.[40]

PSoC	CPU	CLOCK	MEMORIA FLASH
PSoC 1	8 bit	Hasta 24 MHz, 4 MIPS	8 KB-32 KB
PSoC 3	8-bit 8051	Hasta 67 MHz, 33 MIPS	8KB-64KB
PSoC 4	32-bit	Hasta 48 MHz,	32KB con lectura acelerada
PSoC LP 5	32-bit	Hasta 74 MHz	256 kB

### 5.10.3 Plataformas de programación

- *CODESYS*:

Es una misma plataforma con un lenguaje unificado para la programación de autómatas y controladores. Esto ha permitido a muchos fabricantes ofrecer soluciones sin tener que partir de cero en el desarrollo de estas. Es un software muy utilizado, son decenas de empresas del mundo de la Automatización Industrial que utilizan Codesys en sus equipos de automatización: Beckhoff, Festo, EATON, KEBA, IFM, LENZE, Schneider Electric, ABB, ESA, el software de programación de estas marcas está basado en Codesys, aunque con ciertas personalizaciones. Codesys es un software que utiliza el estándar de programación de PLC IEC 61131-3 con los 5 lenguajes de programación lo que te asegura que tu programación cumple la norma.

- Lenguaje escalera (LD - Ladder Diagram)
- Diagrama de bloque de funciones (FBD - Function Block Diagram)
- Texto estructurado (ST - Structured Text)
- Lista de instrucciones (IL - Instruction List)
- Bloques de función secuenciales (SFC - Sequential Function Chart)

Además dispone una versión DEMO 100% operativa que permitir aprender a programar autómatas basados en el lenguaje IEC 61131-3 y simular los programas.[40]



## 6. METODOLOGÍA

### 6.1 DIAGNÓSTICO DE LA CALDERA

Teniendo en cuenta la implementación del nuevo sistema multiplataforma se realiza un diagnóstico a la caldera, ver figura 31, en la cual se definen los elementos principales que la componen, el estado de funcionamiento, con el fin de realizar las mejoras pertinentes para el acondicionamiento del nuevo sistema. A continuación se muestra el resultado del diagnóstico:

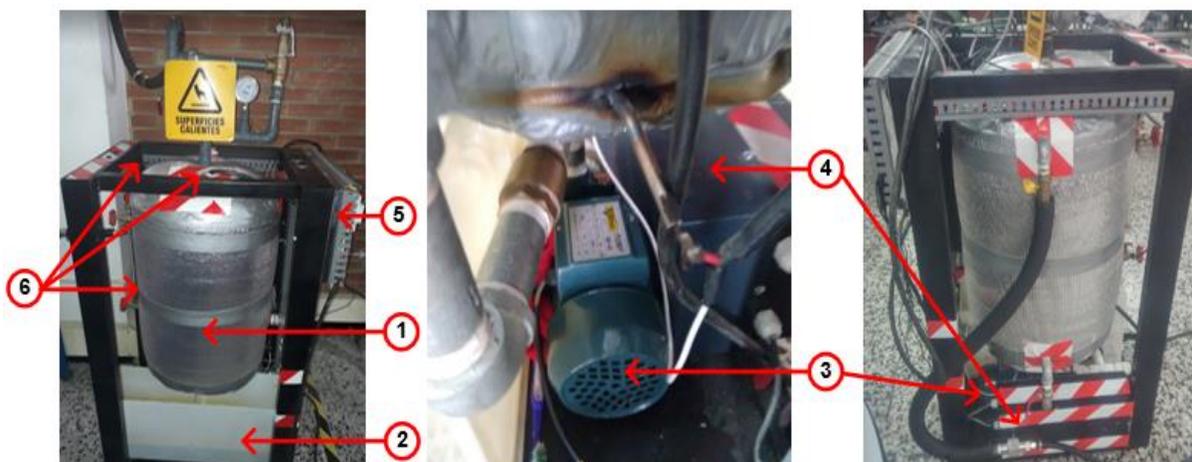


Fig 31. Caldera recibida - Junio de 2018 para diagnóstico. [Autor]

1. Un tanque de acero al carbón (TK-001) cubierto de frescasa y termo-espuma que funciona como un aislante térmico.
2. Un tanque de reserva (TK-002) de agua en acrílico en mal estado, poco almacenamiento y difícil acceso del líquido
3. Una bomba (P-001) de  $\frac{1}{2}$ " HP en buen estado
4. Una válvula servo-accionada (CV-105) defectuosa y por ende sin servicio
5. Un tablero de control sin identificación en los cables de los equipos con pérdidas de espacio por mal ubicación de algunos elementos del tablero
6. Tres sensores (nivel, presión, temperatura) en buen estado, esto se debe a que los estudiantes del semestre 2018-I de la asignatura Instrumentación de procesos II ajustaron el sensor de nivel y de presión, el sensor de temperatura fue cambiado por los estudiantes 2017-III.

La caldera está integrada por los equipos y elementos indicados a continuación:

#### 6.1.1 Equipos de control y supervisión

Los equipos de control y supervisión que se encuentran instalados en la Caldera, se muestran en la Tabla 4; con sus respectivas especificaciones de funcionamiento.

Tabla 4. Equipos de control y supervisión de la caldera [Autor]

TABLA DE EQUIPOS DE CONTROL Y SUPERVISIÓN DE LA CALDERA							
ITEM	TAG	EQUIPO	FABRICANTE	REFERENCIA	ENTRADAS/SALIDAS	COMUNICACIÓN	POWER
1	YIC-01	PLC (Controlador Lógico Programable)	Allen Bradley	1769-L23E	16 entrada binarias 16 salida binarias 1 entrada análoga	Ethernet/IP Serial - RS232 DeviceNet	24VDC
2	VFD-01	Variador de frecuencia	Allen Bradley	PowerFLEX40	4 entrada binarias 3 salida de relé 1 entrada análoga (4-20mA) 1 salida análoga (0-10V)	Ethernet/IP Serial - RS485 DeviceNet	24VDC
3	HMI-01	Pantalla HMI	Allen Bradley	PanelView PLUS 600	N/A	Ethernet/IP Serial - RS232 DeviceNet	24VDC
4	SW-01	Switch Ethernet	Allen Bradley	Stratix 2000	5 Puertos RJ45	Ethernet/IP	24VDC

### 6.1.2 Sensores

Como elementos de instrumentación se encuentran instalados en la Caldera, los equipos indicados en la Tabla 5.

Tabla 5. Sensores de la caldera. [Autor]

TABLA DE SENSORES DE LA CALDERA							
ITEM	TAG	EQUIPO	FABRICANTE	REFERENCIA	UBICACIÓN	TIPO SEÑAL	POWER
1	LT-102	Sensor de nivel *	N/A	N/A	Visor de nivel - Caldera	Binaria	24 VDC 5 VDC
2	PT-100	Sensor de presión	NPX Semiconductor	MPX5700	Caldera	0.2 - 4.7V	5 VDC
3	TT-100	Transmisor de temperatura	N/A	RTD PT100	Caldera	4-20mA	24VDC

\*El sensor de nivel fue ajustado por los estudiantes de la materia de instrumentación de procesos II en el semestre 2018-I, utilizaron led infrarrojo (receptor-emisor), junto con un LM358.

### 6.1.3 Actuadores

Como elementos actuadores se encuentran instalados en la Caldera, los equipos indicados en la Tabla 6.

Tabla 6. Actuadores de la caldera [Autor]

TABLA DE EQUIPOS DE ACTUADORES DE LA CALDERA					
ITEM	TAG	EQUIPO	TIPO SEÑAL	POWER	SERVICIO
1	HV-101	Válvula manual	N/A	N/A	Nivel bajo del visor de nivel de la caldera.
2	HV-102	Válvula manual	N/A	N/A	Nivel alto del visor de nivel de la caldera.
3	HV-103	Válvula manual	N/A	N/A	Para el desagüe de la caldera.
4	HV-104	Válvula Check (antiretorno)	N/A	N/A	Limitar el paso del agua en un sentido y dejar paso libre en el contrario.
5	HV-105	Válvula manual	N/A	N/A	Limitar el flujo de vapor que sale de la caldera hacia CV-105

6	PV-106	Válvula manual (seguridad)	N/A	N/A	Limitar la presión de la caldera no mayor a 60PSI.
7	CV-105	Válvula servoaccionada	Serial	12VDC	Controlar el flujo de vapor que sale de la caldera. NO FUNCIONA
8	R-001	Resistencia 2500W	Binaria (ON/OFF)	220VAC	Calentar agua que está dentro de la caldera.
9	P-001	Bomba de agua	Binaria (ON/OFF)	110VAC	Succión y descarga de agua hacia la caldera.
10	PI-106	Indicador de presión	N/A	N/A	Indica la medición de presión en PSI de la caldera.

### 6.1.4 Infraestructura

La caldera cuenta con dos tableros de control, ver figura 32, el tablero X2 donde se encuentran los equipos mencionados en la tabla 4, y un tablero X1 que permite la distribución de energía y organización de las señales de los sensores y actuadores.



Fig 32. Tableros recibidos - Junio de 2018 para diagnóstico. [Autor]

Para el almacenamiento y proceso de la planta, ver figura 31, se encuentran instalados los equipos indicados en la Tabla 7.

Tabla 7. Equipos de almacenamiento y proceso [Autor]

TABLA DE EQUIPOS DE ALMACENAMIENTO Y PROCESO					
ITEM	TAG	EQUIPO	TIPO SEÑAL	POWER	SERVICIO
1	TK-001	Tanque tipo Vessel	N/A	N/A	Tanque en acero de generación de vapor
2	TK-002	Tanque de techo abierto	N/A	N/A	Tanque en acrílico para reserva de liquido

El cableado de los equipos corresponde a cable de instrumentación 18AWG y es conducido mediante canaletas plásticas al interior del tablero y hacia los dispositivos de la caldera (sensores y actuadores). Las tuberías y accesorios para la canalización del tanque de reserva hacia la bomba (1) es PVC  $\frac{3}{4}$ " y de la bomba hacia la caldera (2) es de acero galvanizado de  $\frac{3}{4}$ ". Ver figura 33.

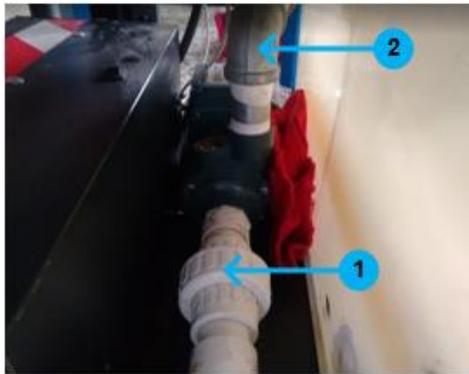


Fig 33. Canalización recibida - Junio de 2018 para diagnóstico. [Autor]

En términos generales los sensores, actuadores y equipos de control-supervisión, se encontraron en buenas condiciones, excepto la válvula servoaccionada que no se encontraba en funcionamiento y era de gran tamaño; En cuanto a la infraestructura el cableado se encontró desorganizado y en algunos casos sin marcación, los recorridos de tuberías eran muy largos y desperdiciaban espacio en la estructura, a la canaleta plástica le faltaban dos tapas y por último, el tanque de reserva estaba defectuoso y no era idóneo para acoger la cantidad mínima de llenado de la caldera y esto hacía que el proceso fuera muy molesto y lento.

A continuación se muestra el acondicionamiento para optimizar el proceso de generación de vapor y enlazar el proceso de la caldera con el sistema multiplataforma entre Rockwell y Codesys.

## 6.2 ACONDICIONAMIENTO DE LA CALDERA

Para un proceso más funcional y cómodo se realizaron cambios en diferentes etapas de la caldera, ver figura 34, en color rojo se muestra las reformas en canalización, válvulas, redistribución de tableros y aprovechamiento de espacio de la estructura.

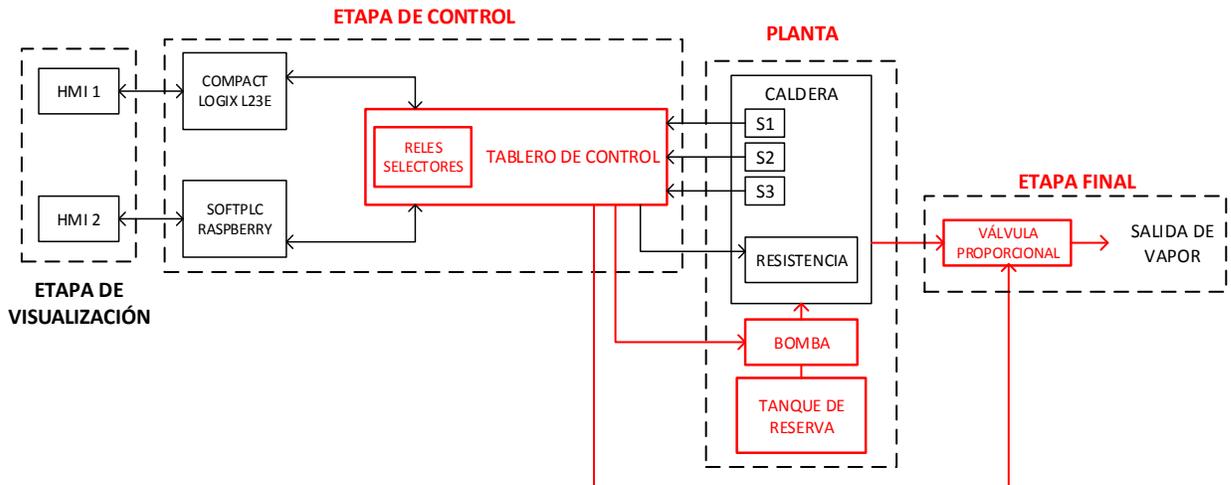


Fig 34. Acondicionamiento-Cambios etapa de control, planta, etapa final. [Autor]

### 6.2.1 Etapa de control

Es la encargada de suministrar y distribuir energía eléctrica y corrientes débiles a todos los dispositivos de la caldera. Consta de dos tableros con elementos internos como borneras de conexión, frenos para riel, breakers, contactores y relés, para que el sistema multiplataforma funcione en la caldera fue necesario realizar los siguientes cambios, ver figura 35:

- Redistribución, organización y aprovechamiento de espacio del tablero
- Etiqueta del cableado y marcación de elementos del tablero
- Nuevas conexiones y elementos como relés y borneras



Fig 35. Acondicionamiento del tablero de control X1-X2. [Autor]

### 6.2.2 Planta

Es la responsable de transportar, almacenar y generar vapor de agua con la ayuda de tanques y actuadores, en la figura 36, se observa los elementos que la componen, los cambios más significativos que se le hicieron a esta etapa fue la adecuación de un nuevo tanque de reserva y su debida re-distribución de tubería.



Fig 36. (1)TK-001, (2) TK-002, (3) R-001, (4) P-001. [Autor]

#### - DISEÑO Y CONSTRUCCIÓN DEL TANQUE DE RESERVA (TK-002)

Fue necesario realizar redistribución de tubería PVC y de acero galvanizado, ver figura 36, que transporta el agua desde el tanque de reserva (TK-002) al tanque de principal (TK-001) por medio de una bomba, este recorrido se optimizo para aprovechar el espacio que de la parte inferior de la estructura con el fin de dimensionar el siguiente tanque de reserva:

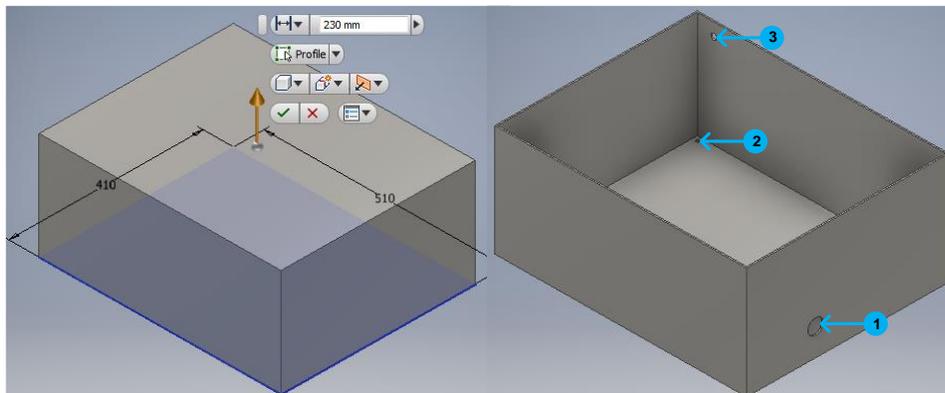


Fig 37. Diseño y vista 3D de TK-002: Tanque reserva. [Autor]

En la figura 37, se muestra el diseño en Autodesk Inventor Professional 2016, la vista 3D del nuevo tanque de reserva, TK-002, las flechas azules de la figura, muestran los 3 orificios que tiene el tanque para acoplarse a la caldera:

1. Orificio para conexión de tubería de  $\frac{3}{4}$ "PVC con bomba.
2. Orificio para conexión de válvula manual de  $\frac{1}{4}$ " para desagüe y limpieza.
3. Orificio para conexión de retorno de agua de la dosificadora a la caldera, con acople de  $\frac{1}{2}$ " en acero inoxidable.

Adicional, en la figura 37, se observan sus medidas:

- Ancho: 410mm
- Largo: 510mm
- Profundo: 230mm

TK-002, está elaborado de acero inoxidable calibre 18 y tiene una capacidad de llenado de aproximadamente 45L. En la figura x, se muestra el tanque instalado en la estructura, el cual nos permite hacer el llenado de la caldera sin necesidad de recircular líquido en otro recipiente y no se desperdicia ni se dejan residuos de agua en el laboratorio.



Fig 38. TK-002: Tanque reserva final. [Autor]

### 6.2.3 Etapa final

Teniendo en cuenta que la válvula de control CV-105 para la salida de vapor que posee la caldera es tipo aguja y el mecanismo automático de apertura que la convierte en válvula proporcional estaba defectuoso y ocupaba mucho espacio, y estas válvulas nuevas son generalmente muy costosas (alrededor de USD\$1200) y no están al alcance de la academia; Por este motivo se optó por el desarme y desacople de sus partes para diseñar un nuevo sistema mecánico funcional y de menor tamaño, empleando dispositivos del mercado local, obteniendo así un sistema de bajo costo comparado con algunos fabricantes de válvulas de control con prestaciones similares.

Para el diseño de los mecanismos de acople con la válvula tipo aguja, fue necesario la compra de un motor DC y un sensor de posición, con las siguientes especificaciones técnicas:

#### A. VÁLVULA TIPO AGUJA

La referencia de la válvula tipo aguja que tiene la caldera es VA150M IT ½ NPT 6000PSI, ver figura 39, de marca ABAC cuenta con:

- Material acero inoxidable
- Empaquetadura PTFE (politetrafluoroetileno)
- Rosca laminada, sin arandela de sello
- Temperatura máxima 260°C
- Presión máxima de servicio 420 bar = 6000 PSI
- Capacidad de pasaje CV<sup>4</sup> 1.20
- Conexión de entrada ½ NPT M
- Conexión de salida ½ NPT H
- Diseñada, fabricada e inspeccionada bajo el sistema de calidad de ISO 9001:2000. [41]
- Valor aproximado COP\$200.000

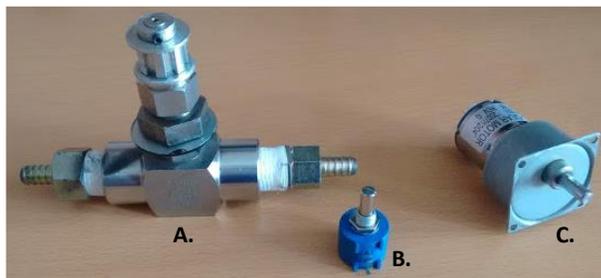


Fig 39. Válvula tipo aguja, B. Sensor de posición, C. Motor DC [Autor]

#### B. MOTOR DC

El motor DC adquirido es el encargado de desplazar el vástago de la válvula, ver figura 39, con velocidad de 60 RPM (Revoluciones por minuto) y alimentación de 24VDC, con valor de COP\$35.000

#### C. SENSOR DE POSICIÓN

Para medir la posición de la válvula se adquirió un potenciómetro multivuelta (10 vueltas), ver figura 39, tiene un valor de resistencia de 10KΩ y alimentación de 5VDC, su valor es de COP\$20.000.

---

<sup>4</sup> El factor CV de una válvula es un coeficiente que expresa el caudal, en galones por minuto de agua a 15°C, que pasa por dicha válvula cuando a través de ella existe una caída de presión de 1 psi.

### 6.2.3.1 Diseño y construcción del mecanismo de CV-105

Para realizar este diseño se investigo acerca de diferentes mecanismos de acople para válvulas de control tipo aguja o globo, en la figura 40 se muestran mecanismos con engranajes, vástago y arandelas.

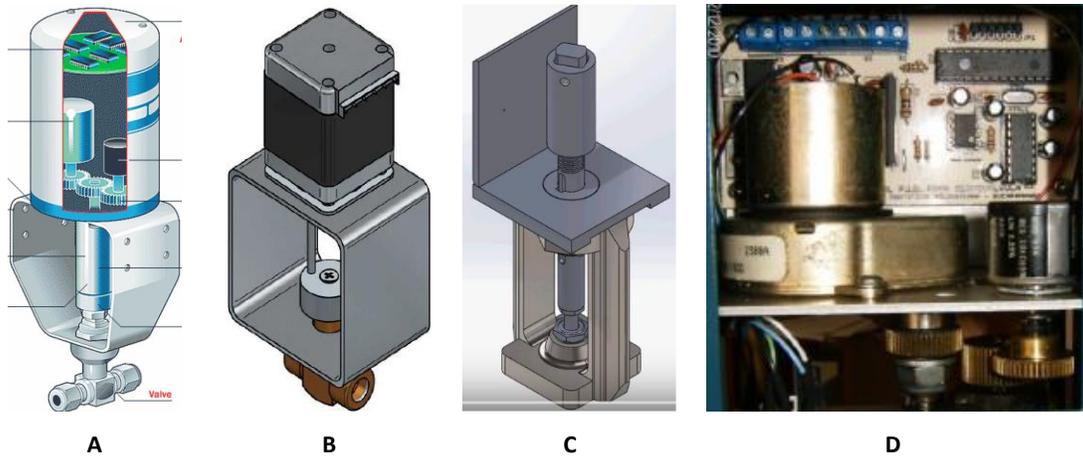


Fig 40. Mecanismos de acople para actuador proporcional.[42]

Teniendo en cuenta la investigación de los anteriores mecanismos, se obtuvo que las imágenes A y B de la figura 40, era lo que más se acercaba al mecanismo que necesitaba CV-105, de igual forma se logra una relación de transmisión de 4:1, lo anterior se obtiene de la siguiente formula:

$$\tau = \frac{Z_2}{Z_1} = \frac{44}{11} = 4$$

Donde:

$Z_2$  = Numero de dientes de E2

$Z_1$  = Numero de dientes de E1

La válvula tipo aguja de la caldera realiza dos tipos de movimientos, lineal y rotacional, esto se debe al empuje que hace el vástago al girar sobre una rosca y este desplazamiento hace que la aguja rote para cerrar el paso de fluido, para realizar la apertura al 100% el vástago debe girar  $1710^\circ = 4.75$  vueltas; el diseño planteado se basa en mecanismos con engranajes para el desplazamiento rotacional y de tipo corredera para el desplazamiento lineal, esto con el fin de ser empalmados con los acoples del motor DC y el sensor de posición, elementos fundamentales que componen el control de la válvula para que sea de apertura proporcional; Debido a lo anterior se inició el diseño con una caja de engranajes y desde esta seguir con los mecanismos de acople para la válvula.

La caja de engranajes tiene las dimensiones que se muestran en la figura 41, la base es de aluminio y los engranajes (E1 y E2) que se usaron en el mecanismo son de acero, E1 tiene 11 dientes y E2 tiene 44 dientes.

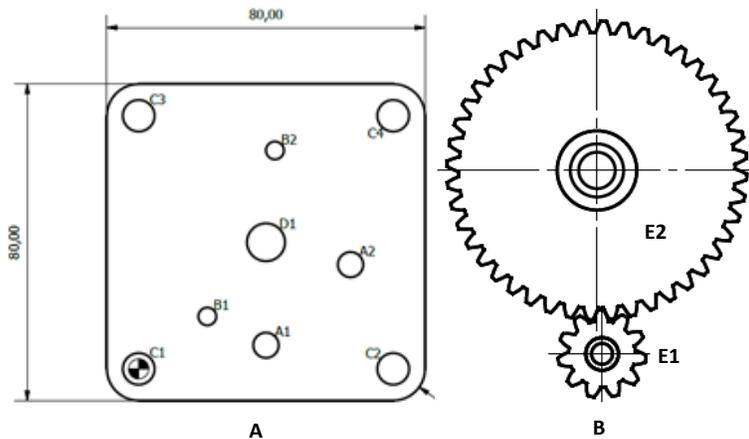


Fig 41. Dimensiones base y engranajes E1 y E2. [Autor]

Para el acople del motor DC y el sensor de posición, a E1 y E2, se diseñaron dos ejes de bronce, las dimensiones se pueden ver en la figura 42.

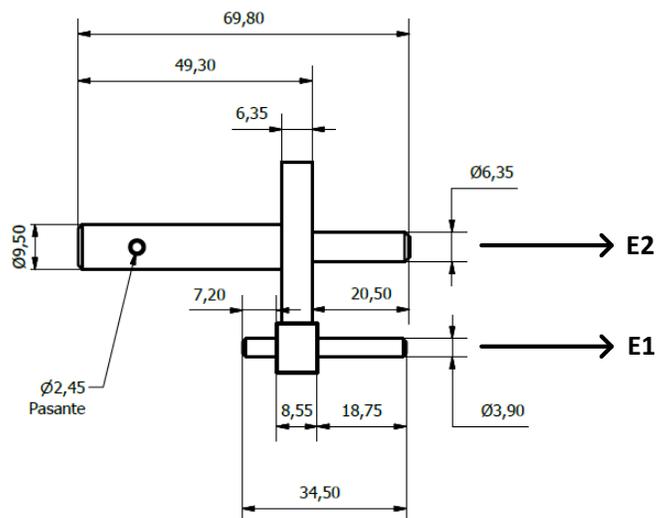


Fig 42. Dimensiones de ejes de acople a E1 y E2. [Autor]

El resultado del mecanismo de engranajes acoplado al sensor y al motor se muestra en la figura 43, en vista lateral izquierda (A) y vista de frente (B) con su respectiva descripción de componentes.

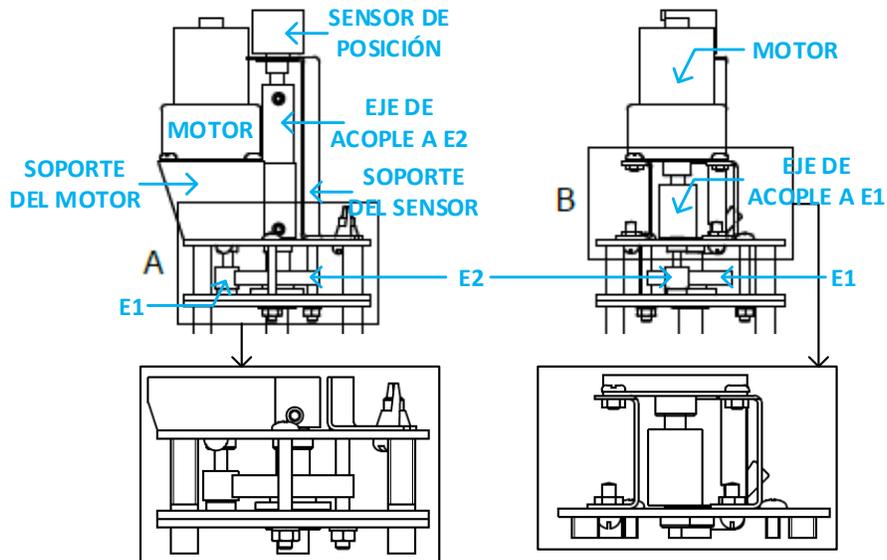


Fig. 43. Resultado final de mecanismo de engranajes. [Autor]

Con lo anterior se tiene la imitación del movimiento rotacional que hace la aguja al sensor y al motor para permitir el paso controlado de vapor; paso siguiente se realizó un segundo mecanismo para acoplar el movimiento rotacional con el lineal y el mecanismo mostrado en la figura 43 no se mueva en el eje Y, por último se planteó un segundo mecanismo basado en discos insertados en ejes para que el movimiento lineal que ejerce el vástago sea realizado por los discos, en la figura 44, se observa el mecanismo de discos acoplado con el mecanismo de engranajes.

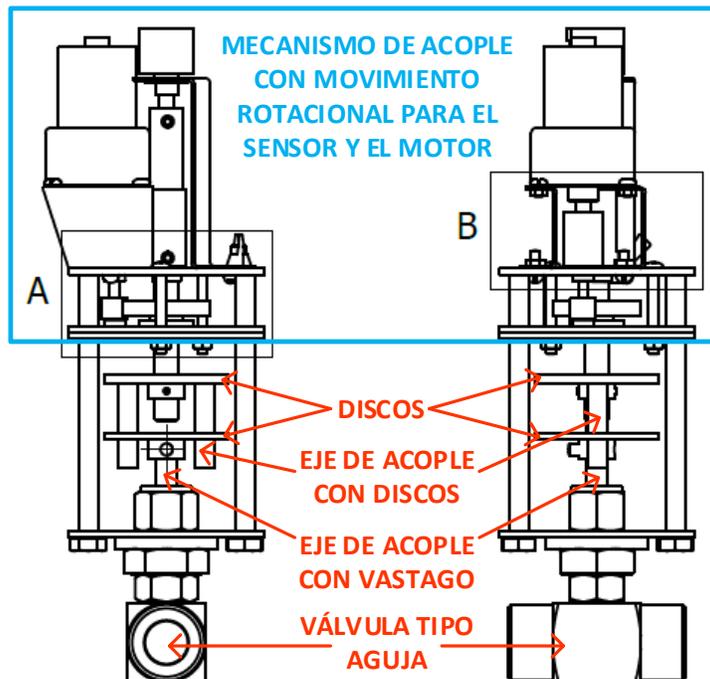


Fig. 44. Mecanismo de discos acoplado al mecanismo de engranajes. [Autor]

El anexo 1, muestra todos los datos del diseño de los mecanismos para controlar la válvula CV-105, (dimensiones, materiales, características mecánicas, planos, etc), Finalmente se obtuvo el diseño de la figura 45.

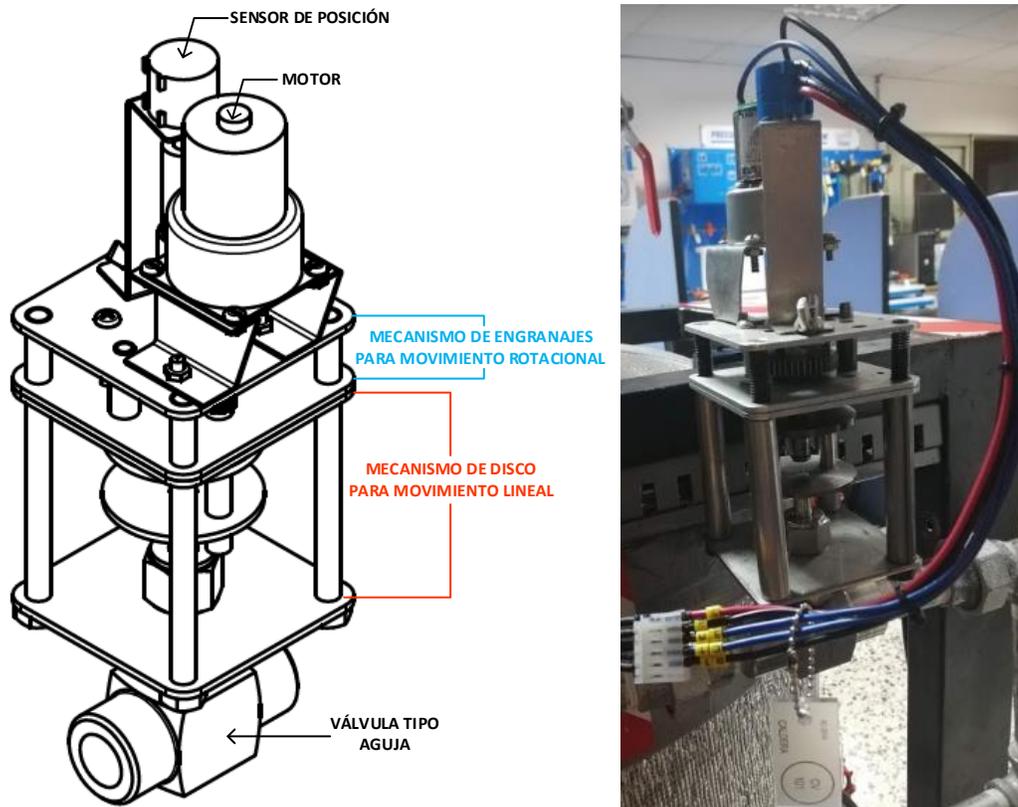


Fig 45. Mecanismo final de CV-105. [Autor]

### 6.2.3.2 Instalación de CV-105 en la caldera

La figura 46, muestra la implementación de nueva tubería de ½" de acero galvanizado que se instaló para acoplar CV-105, permitiendo la salida de vapor controlada, para esta sección de tubería se usó:

- 9 Niple de diferentes tamaños (entre 3cm a 15cm),
- 4 Codos
- 2 Unión tipo T
- 2 Tuerca unión universal
- 2 Válvulas manuales (HV-105 y HV-105A)



Fig 46. Tubería para conexión de CV-105. [Autor]

En caso de mantenimiento para CV-105 se deben seguir los siguientes pasos:

1. Verificar que la caldera no se encuentre en funcionamiento.
2. Aflojar las tuercas unión universal que se encuentran a cada extremo de CV-105.
3. Desmontar la válvula.
4. Colocar niple de conexión de 15cm, en la ubicación de CV-105.
5. Apretar fuerte las tuercas unión universal
6. La configuración de la tubería debe quedar como observa en la figura 47.

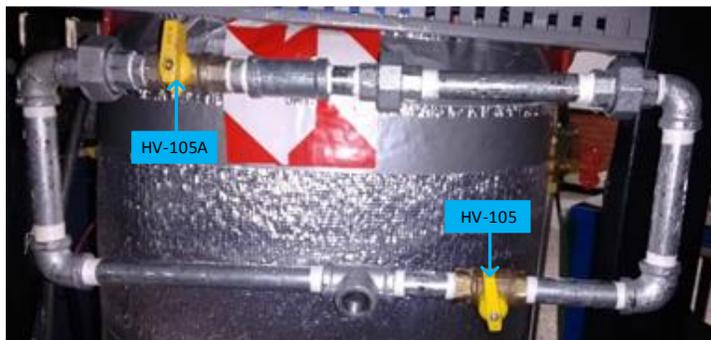


Fig 47. Tubería para mantenimiento de CV-105. [Autor]

La tabla 8, muestra la configuración normalmente abierto (NA) o normalmente cerrado (NC) de las válvulas CV-105, HV-105 y HV-105A, a tener en cuenta en caso de un estado normal o en mantenimiento de la caldera.

Tabla 8. Configuración de estado de las válvulas para salida de vapor [Autor]

TABLA DE CONFIGURACIÓN DE ESTADO DE LAS VÁLVULA PARA SALIDA DE VAPOR							
ITEM	TAG	EQUIPO	ESTADO DE LA CALDERA				OBSERVACIONES
			NORMAL	MANTENIMIENTO			
1	CV-105	Válvula proporcional	NA	N / A	N / A	N / A	En mantenimiento la válvula se debe retirar y colocar el niple.
2	HV-105	Válvula manual	NA	NA	NA	NC	En mantenimiento de CV-105 hay tres configuraciones de estado de HV-105 y HV-105A
3	HV-105A	Válvula manual	NC	NA	NC	NA	

### 6.2.3.3 Diseño y construcción del control electrónico de CV-105

Para que los mecanismos anteriormente descritos sean controlados de forma electrónica, se diseñó un circuito impreso para el control de la posición de apertura proporcional de la válvula. En la figura 48, se muestra los componentes del diseño esquemático de la PCB realizada en Eagle 9.1.1, Donde el componente fundamental es el microcontrolador PSoC 4 CY8CKIT-049 4245, encargado de leer las señales de: sensor de posición, señal de control (4-20mA, 0-10V, 0-5V o comunicación por RS-232) y sentido de giro del motor.

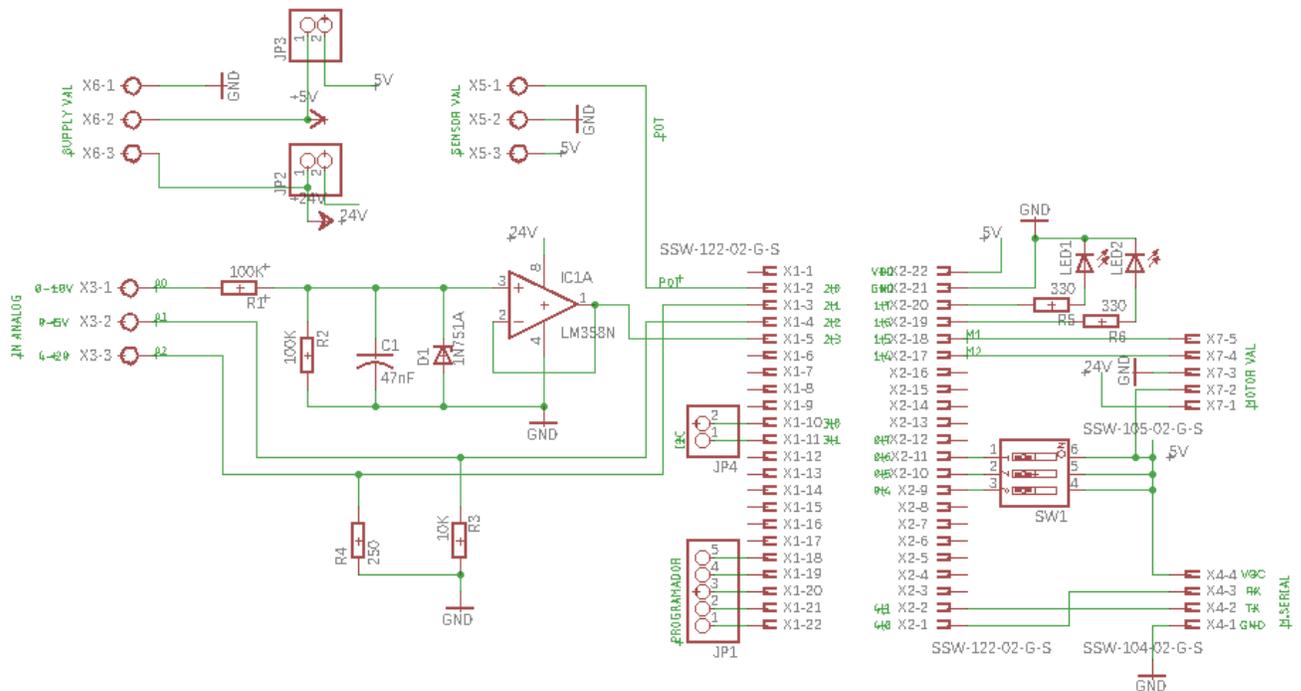


Fig 48. Diseño esquemático de PCB para CV-105. [Autor]

Los componentes usados en el diseño se describen en la siguiente tabla:

Tabla 9. Componentes para Target [Autor]

TABLA COMPONENTES DE PCB DE CONTROL CV-105		
CANTIDAD	REFERENCIA	DISPOSITIVO
2	100K	Resistencia
1	10K	Resistencia
2	330	Resistencia
1	250K	Resistencia
1	47nF	Condensador poliéster
1	1N751A	Diodo Zener
1	LM358N	Amplificador Operacional
1	SW1	Dip switch
1	LED1	Led blanco

1	LED2	Led azul
1	Conector	Regleta 90°
3	Conector	Bornera azul
1	Modulo	Modulo puente H
1	Modulo	Modulo RS-232 to TTL
1	CY8CKIT-049 4245	PSoC 4

La PCB del control de la válvula es de fibra de vidrio, sus dimensiones son 77mm de ancho por 62 mm de largo, en la figura 49, se observa como fue el procedimiento de elaboración e integración de los dispositivos que la componen.

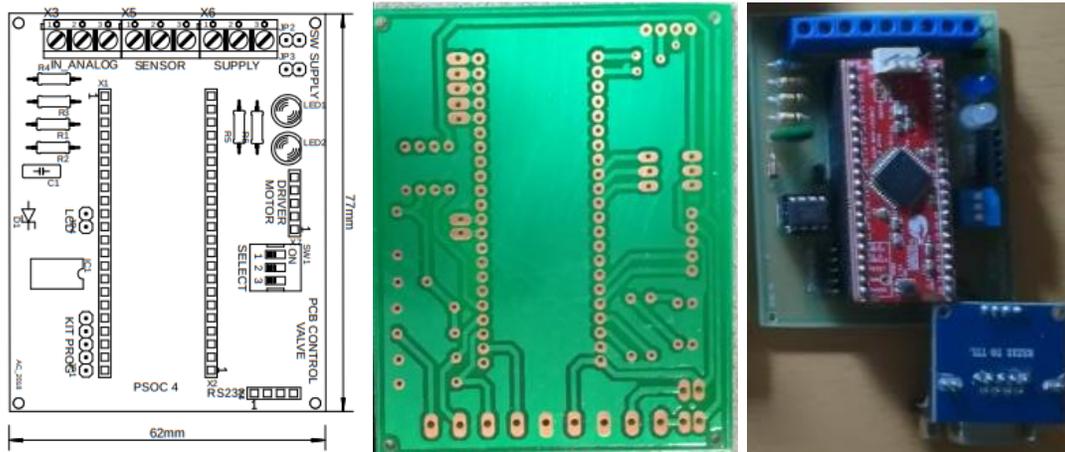


Fig 49. Construcción de PCB para CV-105. [Autor]

Para la programación del microcontrolador PSoc 4 CY8CKIT-049 4245, realizado en lenguaje C, en el software de programación PSoc Creator 4.2, en la figura 50, se observa el esquema de bloques para la configuración de lectura y ejecución de las señales de control de la válvula, donde se usa un bloque ADC de 12 bit para la lectura del sensor de posición y la señales de control, Un bloque UART para la comunicación por RS-232, un bloque TIMER para un tiempo de ejecución del control proporcional de 10 ms, Un bloque PWM para accionar el motor, también cuenta con entradas digitales IN1\_1, IN2\_2, IN3\_3 para conexión del dip switch y elegir el tipo de señal de control ya sea de 4-20mA, 0-10V, 0-5V o RS-232, por último los dos led indicadores de apertura o cierre total de la válvula se muestran como LED\_BL y LED\_WH. El código de programación se encuentra en el anexo 2.

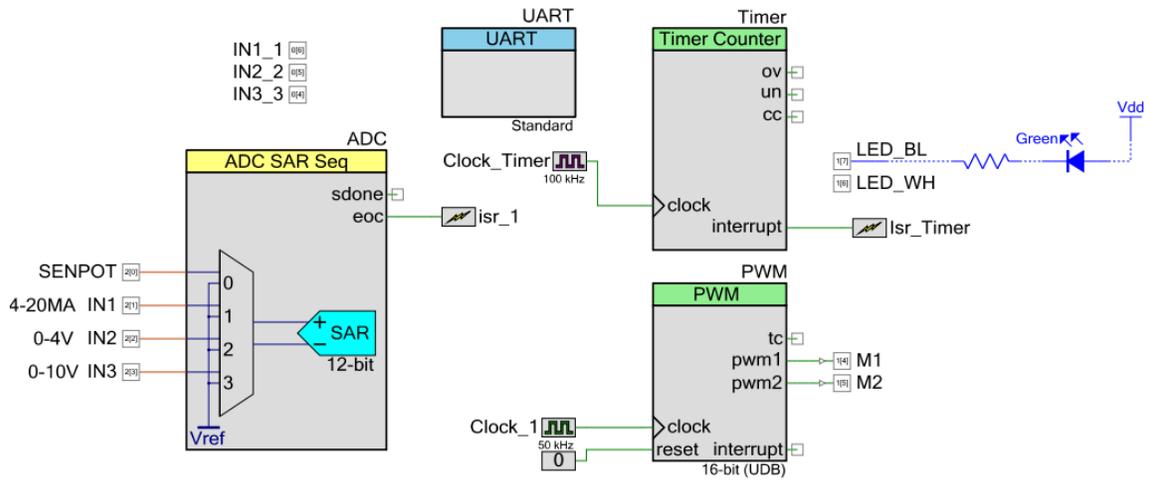


Fig 50. Esquemático de bloques del programa en PSoC Creator. [Autor]

Finalmente se diseña en Autodesk Revit 2017, una carcasa en acrílico de 3mm de grosor, de dimensiones 11.6cm x 10.6cm x 4.5cm (ancho x alto x profundo), para protección y ubicación de la tarjeta de control de CV-105 en el riel din del tablero de control X1, en figura 51, se observa el plano de dimensiones, la vista 3D del diseño y el resultado final de la instalación de la tarjeta de control de CV-105 junto con la carcasa.

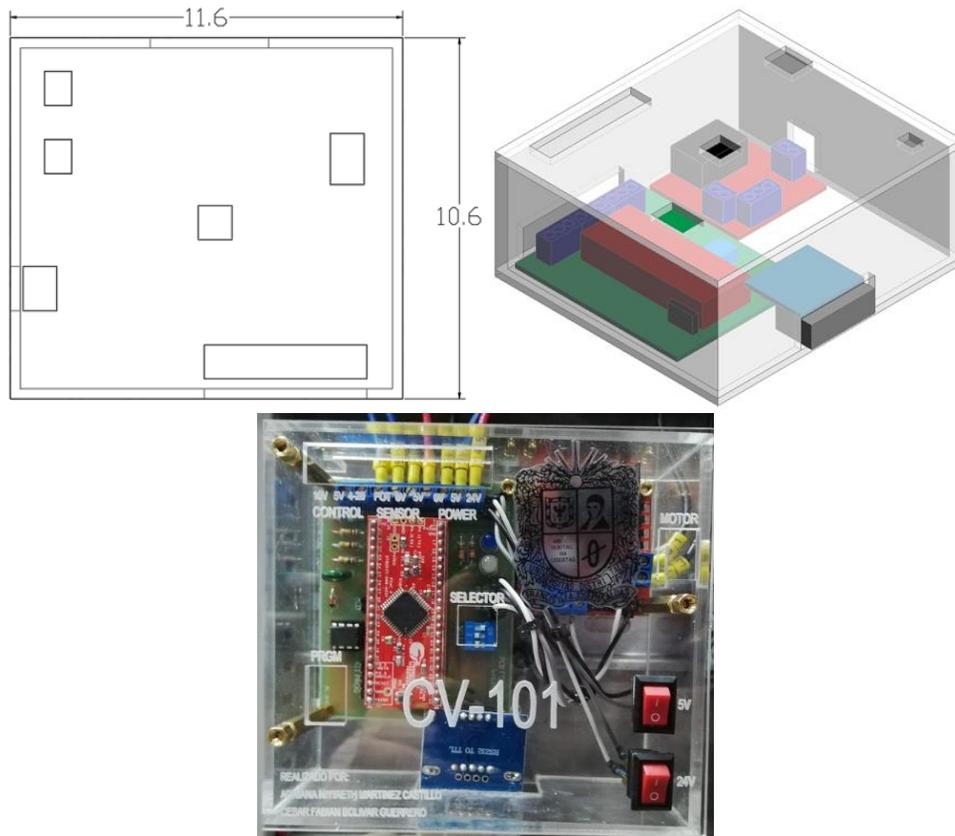


Fig 51. Diseño, 3D y resultado final de la tarjeta de CV-105 con carcasa. [Autor]

## **Características Generales del Controlador para CV-105**

### *Alimentación*

- Tensión de entrada: 24V DC (Driver motor)
- Tensión de entrada: 5V DC (Tarjeta)

### *Entradas digitales*

- 3 entradas con elección de Dip switch según sea necesario:
  - 100: 4 a 20mA
  - 010: 0 – 5V
  - 001: 0 – 10V
  - 111: RS – 232

### *Entradas análogas*

- 4 entradas
  - 1 para lectura sensor de posición
  - 3 por selección del Dip switch:
    - 4 – 20 mA
    - 0 – 10 V
    - 0 – 5 V
- Resolución 12 bits

### *Indicadores de posición para:*

- Posición 0° válvula – Led blanco
- Posición 1710° válvula – Led azul

### *Puertos*

- 1 puerto RS-232.

## **Dimensiones y montaje**

### *Dimensiones*

- 11.6cm x 10.6cm x 4.5cm (ancho x alto x profundo)

### *Montaje*

- Riel din

### *Tipo de borneras*

- Conexión con tornillo

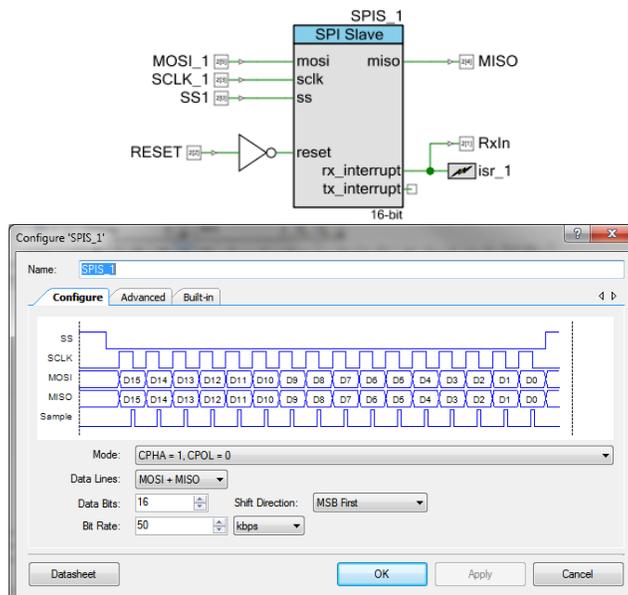
### 6.3 INTEGRACIÓN DE LA NUEVA PLATAFORMA DE PROGRAMACIÓN

Para implementar un sistema multiplataforma en el proceso de generación de vapor a través de la caldera, es necesario adicionar un PLC que cuente con otra plataforma de programación distinta a la de Rockwell, único controlador con el que cuenta la caldera. Teniendo en cuenta que más de 250 fabricantes de dispositivos de diferentes sectores industriales ofrecen sus dispositivos de automatización programable con la plataforma de programación CoDeSys basado en el estándar IEC 61131-3. Se diseñó un hardware que emula el funcionamiento de un PLC y así convertir una Raspberry Pi en un SoftPLC.

#### 6.3.1 Comunicación entre Raspberry PI 2B y PSoC 5LP

Antes del diseño del hardware del SoftPLC es necesario realizar pruebas de transferencia de datos entre la Raspberry Pi 2B y el PSoC 5LP, se toma como referencia la monografía “Diseño de un SOFT-PLC basado en un computador de placa reducida (SBC) raspberry”[40], de allí se implementa la comunicación SPI (Serial peripheral interface) pero esta no funciona, ya que en el documento no se muestran todas las configuraciones, entonces se decide modificar el código base de esta comunicación realizado en CoDeSys como bloque de función (FB), que se encuentra en el anexo 3.

Al mismo tiempo se programa el PSoC 5LP como esclavo, a continuación se muestra la figura 52, donde se visualiza la configuración interna y externa del bloque SPIS\_1 para la comunicación SPI, con una trama de datos de 16 bits y velocidad de tx (transmisión) de 50Kbps, mayor a la comunicación I2C la cual tiene una trama de datos de 8 bits y velocidad de tx baja.



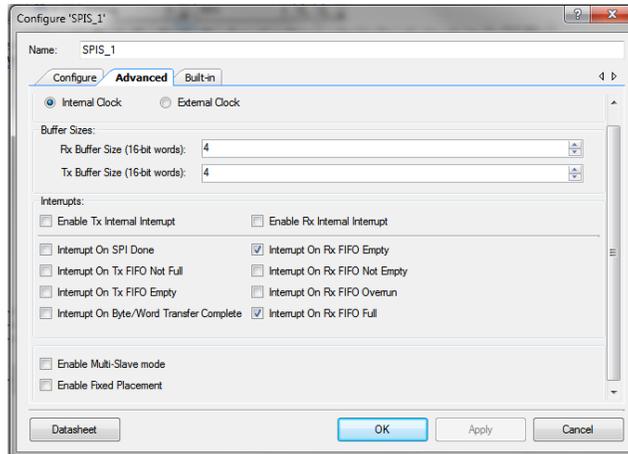


Fig 52. Configuración externa e interna del bloque SPIS\_1. [Autor]

Al tener una comunicación SPI satisfactoria, entre los dos dispositivos se procede a configurar los periféricos del PSoC 5LP como E/S análogas y digitales, en la figura 53, se muestra los bloques usados en la configuración. El bloque ADC\_DeISig es de 16 bits y los cuatro bloques DVDAC son de 12 bits y su respectivo seguidor de voltaje PGA, el resto de bloques hacen referencia a los pines habilitados como E/S digitales. En el anexo 4, se muestra el código de programación implementado.

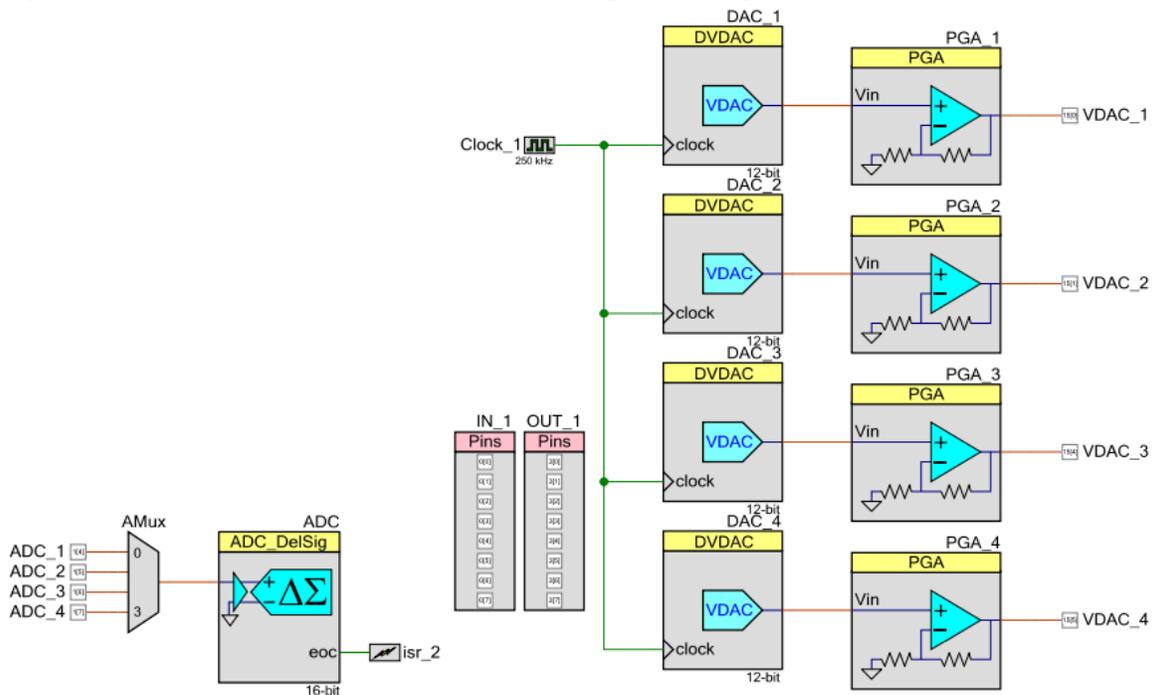


Fig 53. Configuración de periféricos de E/S. [Autor]

### 6.3.2 Diseño y construcción del hardware del SoftPLC V1.0

El diseño físico del SoftPLC tiene como referencia las dimensiones de la Raspberry Pi 2B, el módulo de 8 relés y el PSoC 5LP, con estos tres dispositivos se inicia el diseño teniendo en cuenta la posición de sus conexiones al usuario final; en el anexo 5 se muestra el esquemático del diseño y en la tabla 10, se describen todos los

componentes electrónicos utilizados en la board final de impresión de la tarjeta del SoftPLC con dimensiones de 240mm de ancho por 80mm de largo, ver figura 54, los recuadros rojos muestran la conexión modular por detrás de la tarjeta del módulo de 8 relés y la Raspberry Pi, la conexión del PSoC 5LP se realiza por delante de la tarjeta, todo eso se logra con la ayuda de regletas tipo hembra.

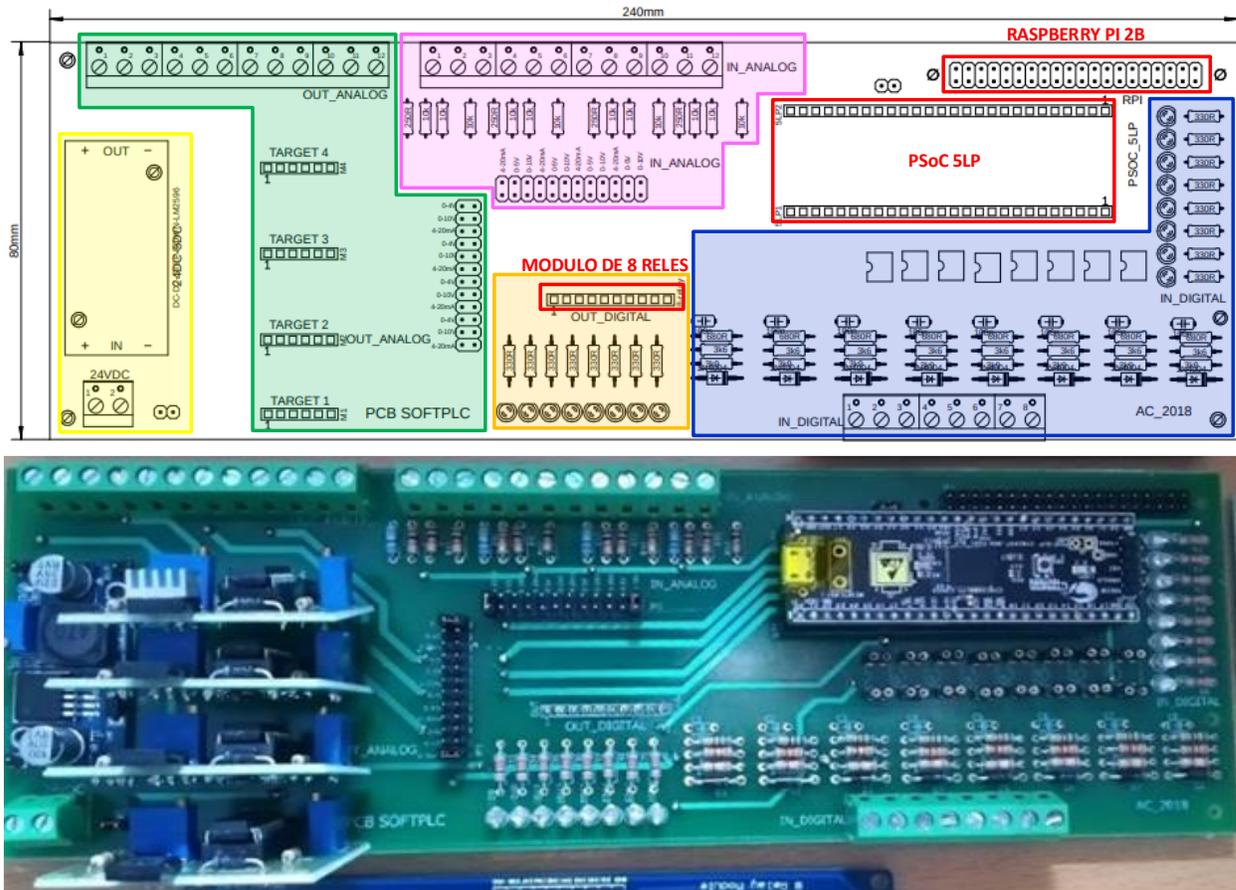


Fig 54. Ubicación de dispositivos electrónicos en la tarjeta del SoftPLC. [Autor]

Tabla 10. Componentes para la tarjeta del SoftPLC V1.0. [Autor]

TABLA COMPONENTES DE PCB DEL SoftPLC V1.0		
CANTIDAD	REFERENCIA	DISPOSITIVO
8	pc817	Optoacoplador
16	Led	LED3MM
2	Conector	PINHD-1X2 pin
2	Conector	PINHD-2X12 pin
1	Conector	PINHD-2X20 pin
4	Conector	conector 2 pines verde
1	Conector	conector 2 pines verde
8	Conector	conectores 3 pines verdes
8	100n	capacitor poliéster

12	10k	resistencia de 1/4W
6	1N4004	1N4004
16	330R	resistencia de 1/4W
8	3k6	resistencia de 1/4W
8	3k9	resistencia de 1/4W
4	5LP1	regleta hembra
8	680R	resistencia de 1/4W

El recuadro **amarillo** de la figura 54, muestra la ubicación de la entrada de 24V (alimentación principal del SoftPLC) y la fuente reductora a 5V.

El recuadro **naranja** representa los indicadores de ocho salidas digitales, debido a que el módulo de 8 relés está conectado por la parte de abajo de la tarjeta, las conexiones a los relés se deben de hacer por ese mismo lugar.

El recuadro de color **rosado** indica la conexión de las cuatro entradas análogas disponibles en el SoftPLC, la selección del tipo de señal a trabajar se debe hacer por medio de jumper de conexión ubicado en la parte inferior de las borneras (4-20 mA, 0-5V o 0-10V), se debe tener en cuenta que solo se puede usar un configuración de señal a la vez.

El recuadro **azul** muestra las ocho entradas digitales utilizables en el SoftPLC con sus respectivos indicadores, están protegidas con el Optoacoplador PC817 y su rango de entrada de 15 a 24V

El recuadro **verde** indica la conexión modular de cuatro target, encargadas de convertir la señal de voltaje de 0-4V a una señal de corriente de 4-20 mA o una señal de voltaje de 0-10V, para ser usadas como salidas análogas, La selección del tipo de señal a trabajar se debe hacer por medio de jumper de conexión, ubicado a la derecha de las targets, se debe tener en cuenta que solo se puede usar un configuración de señal a la vez, y el SoftPLC cuenta con cuatro salidas análogas, en la figura 55, se muestra el esquemático del diseño de la target y en la tabla 8 se describen todos los componentes electrónicos utilizados en la board de impresión final con dimensiones de 38mm de ancho por 42mm de largo.

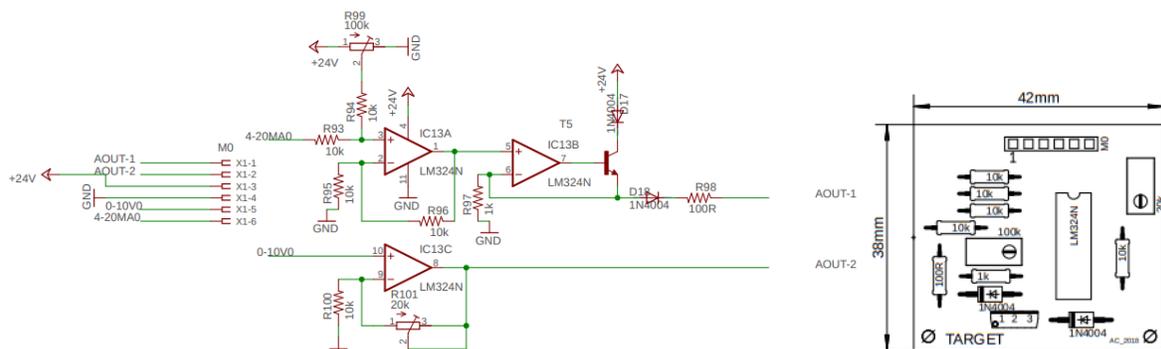


Fig 55. Esquemático y ubicación de dispositivos de Target. [Autor]

Tabla 11. Componentes para Target [Autor]

TABLA COMPONENTES DE PCB DE TARGET		
CANTIDAD	REFERENCIA	DISPOSITIVO
4	Transistor	c1162 transistor
4	100R	resistencia de 1/4W
4	100k	trimmer
20	10k	resistencia de 1/4W
4	1k	resistencia de 1/4W
4	20k	trimmer
4	LM324N	LM324N
1	Conector	Regleta 90°

Posteriormente se diseña en Autodesk Revit 2017, una carcasa en acrílico de 3mm de grosor, de dimensiones 27cm x 10.5cm x 9.4cm (ancho x alto x profundo), para protección y ubicación de la tarjeta de SoftPLC en el riel din del tablero de control X1, en figura 56, se observa el plano de dimensiones y la vista 3D del diseño final de la carcasa.

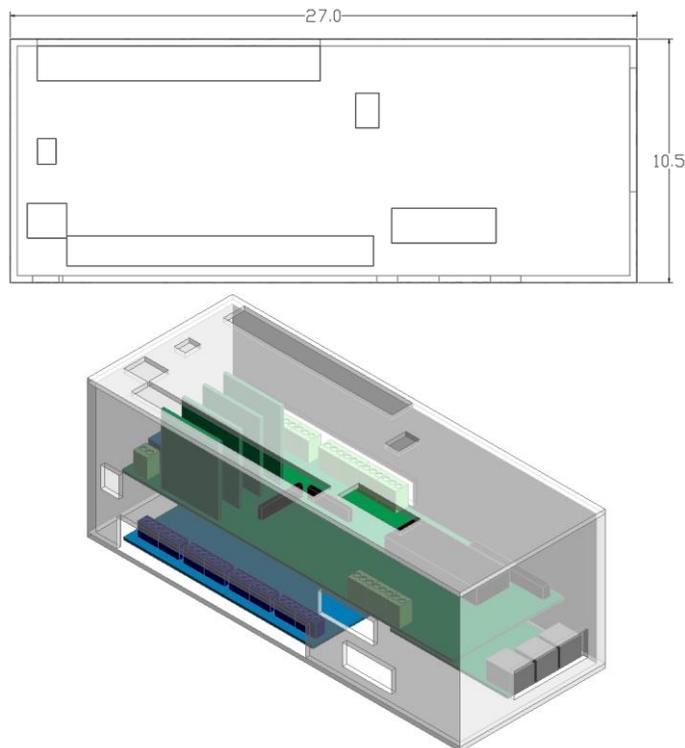


Fig 56. Diseño y vista 3D de carcasa tarjeta de SoftPLC. [Autor]

Como resultado final se obtiene un SoftPLC V1.0, ver figura 57, Su función principal es adquirir señales análogas y digitales a través de PSoC 5LP, provenientes de todo tipo de sensores y ejecutar la lógica que el estudiante requiera para controlar la caldera, La programación de la lógica se realiza mediante la plataforma CoDeSys

V3.5 SP14 que es proporcionado gratuitamente, por 3S-Smart, junto con el paquete de instalación 'CODESYS Control for RaspberryPi SL'.

Debido a que la plataforma se basa en el estándar IEC 61131-3, se facilita la programación al estudiante mediante lenguajes como Ladder (LD), Diagrama de bloque de funciones (FBD), Texto estructurado (ST), y bloque de función secuencial (SFC).

La comunicación y programación del dispositivo se realiza por medio de una interfaz de red (LAN o WLAN) y el equipo cuenta con un puerto RJ-45 compatible con gran número de instrumentos industriales y expansiones.



Fig 57. Resultado final – SoftPLC V1.0. [Autor]

### **Características Generales del SoftPLC V1.0**

#### **Alimentación**

- Tensión de entrada: 24V DC

#### **Entradas digitales**

- 8 entradas
- Nivel lógico bajo: 0 a 5V
- Nivel lógico alto: 14 a 25V

#### **Entradas análogas**

- 4 entradas
- Resolución 16 bits
- Por selección con jumper (Solo se puede usar una de estas señales por canal):
  - 4 – 20 mA
  - 0 – 10 V
  - 0 – 5 V

#### **Salidas digitales**

- 8 a relé NA o NC

#### **Salidas análogas**

- 4 salidas

- Resolución 12 bits
- Por selección con jumper (Solo se puede usar una de estas señales por canal):
  - 4 – 20 mA
  - 0 – 10 V
  - 0 – 4 V

#### *Puertos*

- 1 puerto RJ-45.
- 4 puertos USB
- 1 puerto HDMI

#### *Buses de campo*

- Modbus TCP Maestro / Esclavo
- Modbus RTU Maestro / Esclavo
- Ethernet/IP Scanner / Adapter
- PROFINET Controlador / Dispositivo

### **Características CPU del SoftPLC V1.0**

#### *Procesador*

- BCM2836 ARMv7 Multinúcleo (4 núcleos de 32 bits)
- Velocidad del reloj: 900MHz
- RAM: 1Gb

#### *Almacenamiento interno*

- MicroSD: 32Gb

#### *Sistema operativo*

- Raspbian

#### *Licencia*

- Tiempo de ejecución: 2 Horas sin licencia (se destina a fines de prueba y aprendizaje)

**Restricciones** *‘El sistema de tiempo de ejecución no tiene un comportamiento en tiempo real.’*

### **Dimensiones y montaje**

#### *Dimensiones*

- 27cm x 10.5cm x 9.4cm (ancho x alto x profundo)

#### *Montaje*

- Riel din

#### *Tipo de borneras*

- Conexión con tornillo

### 6.3.3 Diseño y construcción del software del SoftPLC V1.0

La plataforma de programación del SoftPLC V1.0 es CODESYS, y se centra en la configuración de la comunicación SPI entre la Raspberry Pi 2B y el PSoC 5LP, la cual permite que el SoftPLC V1.0 opere como un controlador lógico programable, debido a que se habilitan las I/O análogas y digitales descritas en el capítulo anterior.

A continuación se muestra la configuración de parámetros que se debe realizar en CODESYS para la Raspberry Pi 2B:

Parámetro	Tipo	Valor	Valor predeterminado	Unidad	Descripción
SPI port	STRING	'/dev/spidev0.0'	'/dev/spidev0.0'		file path of the SPI port
_diMode	DINT	1	0		SPI mode
_diBitsPerWord	DINT	8	0		SPI bits per word
_diMaxSpeed	DINT	100000	0		SPI max speed setting [Hz]

Fig 58. SPI Master Configuración. [Autor]

El código de programación usado para habilitar las I/O análogas y digitales del SoftPLC V1.0, es el que se muestra a continuación:

```

FUNCTION_BLOCK SPI_16_RW
VAR_INPUT
    DAC_1:UINT;
    DAC_2:UINT;
    DAC_3:UINT;
    DAC_4:UINT;
    GPO:UINT;           //SALIDAS DIGITALES PSOC
END_VAR
VAR_OUTPUT
    ADC_1:UINT;
    ADC_2:UINT;
    ADC_3:UINT;
    ADC_4:UINT;
    GPI:UINT;          //ENTRADAS DIGITALES PSOC
END_VAR
VAR
    W_Register:UINT:=43690; //ABRIR LA COMUNICACION CON EL PSOC
    i:INT:=0;
    RxdataHi:UINT;
    RxdataLo:UINT;
    dX:ARRAY[0..5] OF UINT;
    TxDataBuffer:ARRAY[0..1] OF BYTE;
    RxDataBuffer:ARRAY[0..1] OF BYTE;
    TxBuffer:POINTER TO ARRAY[0..1] OF BYTE;
    RxBuffer:POINTER TO ARRAY[0..1] OF BYTE;
    RxState:POINTER TO ARRAY[0..1] OF BYTE;
    SPI_State:BOOL;
    f1:UINT;
    A1: ARRAY[0..5] OF UINT;
END_VAR

PROGRAM PLC_PRG
VAR
    SPI_SLAVE:SPI_16_RW;
    OUT_AN0:UINT; //SALIDA ANÁLOGA CANAL 0
    OUT_AN1:UINT; //SALIDA ANÁLOGA CANAL 1
    OUT_AN2:UINT; //SALIDA ANÁLOGA CANAL 2
    OUT_AN3:UINT; //SALIDA ANÁLOGA CANAL 3
    OUT_DIG:UINT; //SALIDAS DIGITALES [0.....7]
    INP_AN0:UINT; //ENTRADA ANÁLOGA CANAL 0

```

```

        INP_AN1:UINT; //ENTRADA ANÁLOGA CANAL 1
        INP_AN2:UINT; //ENTRADA ANÁLOGA CANAL 2
        INP_AN3:UINT; //ENTRADA ANÁLOGA CANAL 3
        INP_DIG:UINT; //ENTRADAS DIGITALES [0.....7]
    END_VAR

```

### - Configuración de las I/O análoga y digital.

La lectura del valor digital de cada canal análogo es de gran velocidad por lo que los valores varían bastante; Además la resolución es de 16 bits en la lectura análoga, por lo tanto se recomienda crear una tarea para leer las entradas análogas con un tiempo de ejecución cíclico de 300 ms o realizar un promedio de la lectura realizada.

En cuanto a la salida análoga del canal 0 que controla la posición de la válvula proporcional CV-105, el valor a enviar debe estar entre 0-4070, un valor más grande a 4070 bloquea la comunicación del SoftPLC V1.0.

Sin embargo lo mejor es usar la escala lineal de la apertura de la válvula, donde el ángulo puede ser un valor entre 0° y 1710° para obtener un valor entre 0 y 4070, que se muestra a continuación:

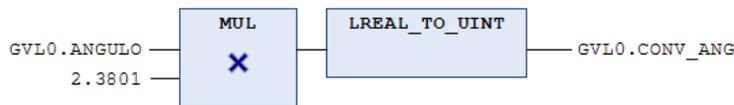


Fig 59. Escala lineal de la apertura de la válvula. [Autor]

Para la lectura de cada entrada digital se puede leer el bit que representa la variable total de la entrada digital de la siguiente manera:

```

PROGRAM PLC_PRG
VAR
    INP0: BOOL; //PARO DE EMERGENCIA FISICO
    INP1: BOOL; //PARO, BOTON ROJO FISICO
    INP2: BOOL; //INICIO, BOTON VERDE FISICO
    INP3: BOOL; //SENSOR DE NIVEL
    INP4: BOOL; //LIBRE
    INP5: BOOL; //LIBRE
    INP6: BOOL; //LIBRE
    INP7: BOOL; //LIBRE
END_VAR

```

```

INP0:=INP_DIG.0; //PARO DE EMERGENCIA FISICO
INP1:=INP_DIG.1; //PARO, BOTON ROJO FISICO
INP2:=INP_DIG.2; //INICIO, BOTON VERDE FISICO
INP3:=INP_DIG.3; //SENSOR DE NIVEL
INP4:=INP_DIG.4;
INP5:=INP_DIG.5;
INP6:=INP_DIG.6;
INP7:=INP_DIG.7;

```

Las salidas Digitales se ejecutan de igual forma que en las entradas digitales:

```

PROGRAM PLC_PRG
VAR
    OUT0: BOOL := TRUE; //LUZ ROJA
    OUT1: BOOL := TRUE; //LUZ VERDE
    OUT2: BOOL := TRUE; //LUZ AMARILLA
    OUT3: BOOL := TRUE;

```

```

OUT4: BOOL := TRUE;
OUT5: BOOL := TRUE;
OUT6: BOOL := TRUE;           //RESISTENCIA
OUT7: BOOL := TRUE;           //BOMBA
END_VAR

OUT_DIG.0:=OUT0; //LUZ ROJA
OUT_DIG.1:=OUT1; //LUZ VERDE
OUT_DIG.2:=OUT2; //LUZ AMARILLA
OUT_DIG.3:=OUT3;
OUT_DIG.4:=OUT4;
OUT_DIG.5:=OUT5;
OUT_DIG.6:=OUT6; //RESISTENCIA
OUT_DIG.7:=OUT7; //BOMBA

```

En el anexo 8, “*Practica 1. Control básico de la caldera con SoftPLC V1.0*” Se explica el paso a paso de la configuración y programación del SoftPLC V1.0, se debe seguir esta guía al pie de la letra para generar y controlar la salida de vapor.

#### 6.3.4 Implementación de la interfaz HMI

Ignition EDGE es una aplicación de bajo coste de la plataforma de software integrada para sistemas SCADA Ignition de Inductive Automation para diseñar aplicaciones en pantallas HMI; basado en web multiplataforma (Linux, Windows, Raspbian). [43]



Fig 60. Ignition EDGE.[43]

La visualización se hace por medio de una pantalla táctil de 7” conectada al puerto HDMI, esta aplicación está instalada en el SoftPLC V1.0, Las características técnicas de la pantalla son:

- Referencia: 7inch HDMI Display-B
- Resolución: 800 x 480 puntos
- Alimentación: 5V DC
- Dimensiones: 164.9mm x 124.27mm (ancho x alto)
- Touch: Capacitivo 5 puntos
- Puertos: Un HDMI y dos mini USB para conexión de alimentación
- Incluye: CD de instalación, cable USB a micro USB, cable HDMI y postes metálicos para instalación.
- Soporta: Raspberry Pi, Windows 7 - 8 - 10, Driver libre. [44]

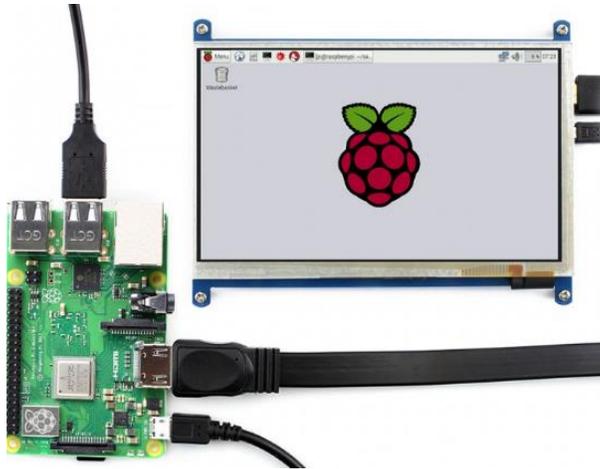


Fig 61. Pantalla HMI. [Autor]

En la figura 61, se muestra la configuración de conexión de la pantalla con la Raspberry Pi, La comunicación entre el SoftPLC V1.0 con el ignition edge se realiza por medio del protocolo OPC-UA, el anexo 8 “Práctica de control básico de la caldera con el SoftPLC V1.0” muestra todos los pasos a seguir para realizar la configuración de esta comunicación.

Debido a que la pantalla no tiene carcasa de protección se realiza el diseño en Autodesk Revit 2017, sus dimensiones de 176mm x 131mm x 15mm (ancho x largo x profundo) y vista 3D se a continuación:

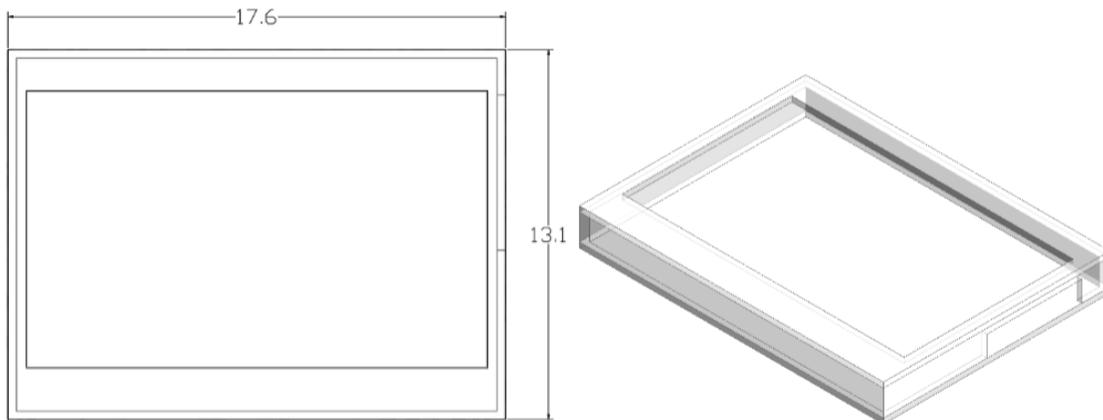


Fig 62. Diseño y vista 3D de carcasa pantalla HMI. [Autor]

Como parte del sistema SCADA, para la caldera se generan 3 despliegues en la HMI ver figura 63, en los cuales además de facilitar la visualización del comportamiento de las variables de control mediante diferentes objetos, se tiene la posibilidad de ingresar valores de referencia, el arranque y parada de las diferentes rutinas según lo indicado en el numeral anterior. Gran parte de los gráficos mostrados en la interfaz, corresponden a un modelo 3D generado mediante el software Revit de Autodesk.

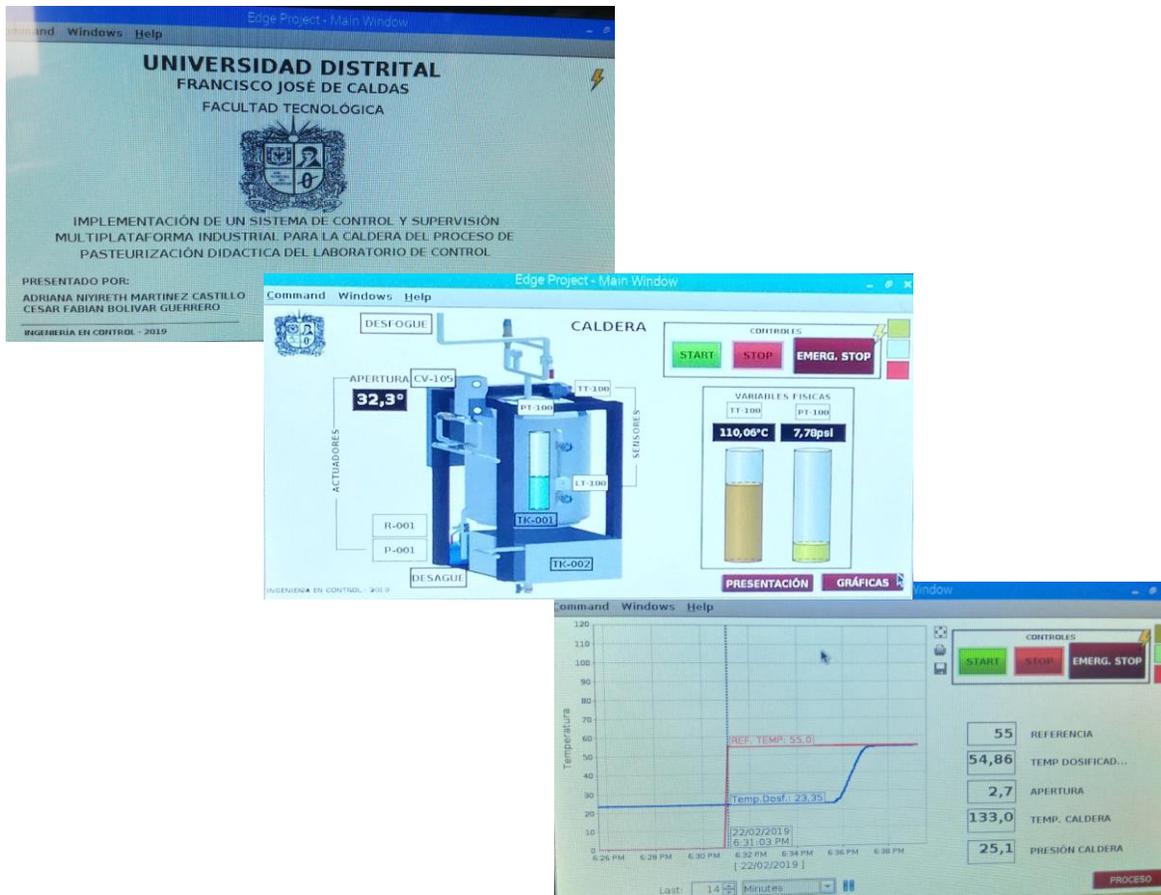


Fig 63. SCADA Final - Caldera. [Autor]

Todos los esquemas de visualización y los despliegues generados para la HMI, se ejecutan de manera similar en una estación de trabajo, de tal forma que se cuenta con la supervisión local y remota de los procesos que se llevan a cabo en la caldera.

#### 6.4 ARQUITECTURA DE COMUNICACIONES - SoftPLC V1.0 y Allen Bradley

Los equipos mencionados en el numeral 6.1.1, y el SoftPLC V1.0 conforman una red dentro de la planta de la caldera, cuya comunicación se centra en un switch Ethernet. Todos los equipos cuentan con direcciones IP fijas asignadas con anterioridad por los directores de grupo y estudiantes con el propósito de tener claridad sobre la disponibilidad de estas. El switch se encuentra conectado a un switch de borde en el cual se centralizan las comunicaciones de las diferentes plantas que conforman el laboratorio.

En este caso se mencionaran las direcciones utilizadas por los equipos, se deben tener en cuenta a la hora de elegir una plataforma para programar y controlar la caldera:

- **PLC Allen Bradley:** 130.130.130.83

- La conexión del controlador de la válvula CV-105 se hace por RS-232
- HMI PANELVIEW PLUS 600: 130.130.130.63
- PowerFlex 40: 130.130.130.93
- **SoftPLC V1.0:** 130.130.130.111
  - La conexión del controlador de la válvula CV-105 se hace por medio de 4 – 20mA y la conexión de la pantalla HMI por cable HDMI.

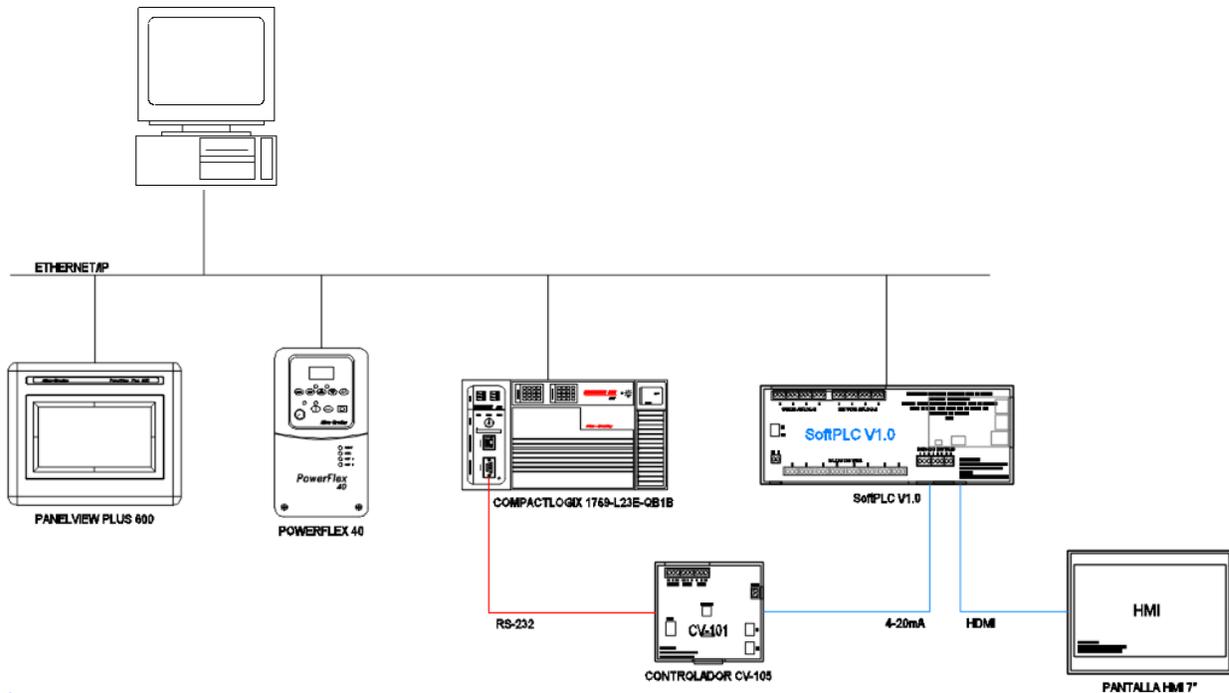


Fig 64. Arquitectura de conexión de los dispositivos. [Autor]

## 6.5 DISEÑO DEL ALGORITMO DE CONTROL

De acuerdo con el diagrama de la figura 67, el control de temperatura se realiza mediante la válvula proporcional CV-105, la cual regulará el flujo de vapor producido por la caldera, como consecuencia de la señal entregada por el transmisor de temperatura del módulo dosificador, en el tanque de calentamiento y el valor deseado o referencia. Estos equipos reportaran o recibirán algún tipo de señal del controlador (SoftPLC V1.0). El operador podrá interactuar con todo el sistema mediante el SCADA, no solo en actividades de monitoreo, sino en el ajuste de valores como la referencia.

Para el sistema de control se plantea un sistema realimentado, aplicado a la variable de temperatura, el cual mantiene una relación determinada entre la salida  $Y(z)$  y la

entrada de referencia ( $r(z)$ ), comparándolas y usando la diferencia  $e(z)$  como señal entrada para el controlador  $C(z)$ .

Estas operaciones son calculadas por el SoftPLC V1.0 programado con un algoritmo de control PID, el cual producirá una señal y la ejecutara sobre la planta  $G(z)$ .

Donde:

- A = Actuador
- P = Planta
- S = Sensor

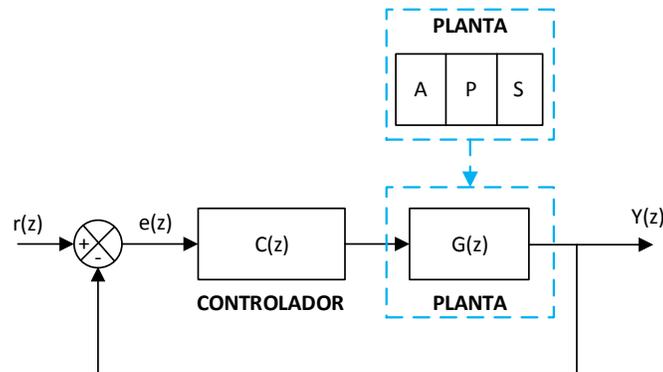


Fig 65. Diagrama de bloques sistema de control. [Autor]

### 6.5.1 Caracterización de los sensores TT-100 y PT-100

#### - Linealización sensor de temperatura TT-100 con SoftPLC V1.0

Se realiza el proceso de caracterización de TT-100, mediante la toma de datos y validación de la señal entregada por el controlador SoftPLC V1.0, de acuerdo al patrón de medida con la lectura de la termocupla tipo K conectada al dispositivo UT39C, se toman datos cada 3°C desde el inicio del proceso hasta 140°C, en la tabla 12 se muestran los valores obtenidos del aumento y disminución de temperatura dentro de la caldera, en los datos de bajada se observan menos datos debido a que la caldera no cuenta con un actuador que realice el trabajo de enfriamiento o disminución de la temperatura.

Tabla 12. Datos de subida y bajada TT-100 con SoftPLC V1.0. [Autor]

DATOS DE SUBIDA		DATOS DE BAJADA	
VOLTAJE	TEMP UT39C	VOLTAJE	TEMP UT39C
1.38359069824218	25	2.62451171875000	140
1.44439697265625	27	2.61703491210937	139
1.47308349609375	29	2.60513305664062	138
1.53877258300781	32	2.59704589843750	137
1.55181884765625	34	2.59658813476562	136
1.55632019042968	35	2.56393432617187	134
1.58775329589843	37	2.55088806152343	133
1.61521911621093	39	2.53051757812500	131

1.64237976074218	41
1.65390014648437	43
1.69143676757812	45
1.70150756835937	47
1.73805236816406	49
1.75033569335937	52
1.78688049316406	54
1.80572509765625	57
1.84013366699218	59
1.86195373535156	61
1.89872741699218	63
1.95632934570312	65
2.01034545898437	67
2.04269409179687	69
2.06398010253906	71
2.01698303222656	73
2.07038879394531	75
2.07336425781250	0.8
2.09579467773437	85
2.14080810546875	90
2.21191406250000	95
2.26257324218750	100
2.32467651367187	105
2.32826232910156	110
2.38098144531250	115
2.42248535156250	120
2.47200012207031	125
2.52372741699218	130
2.58956909179687	135
2.64450073242187	137
2.65968322753906	139
2.63389587402343	140

2.51914978027343	130
2.50900268554687	129
2.49435424804687	127
2.47230529785156	125
2.46559143066406	124
2.45811462402343	123
2.45315551757812	122
2.44796752929687	121
2.43576049804687	120
2.38800048828125	118
2.37297058105468	115
2.35466003417968	113
2.32757568359375	110
2.32269287109375	107
2.31170654296875	106
2.30827331542968	104
2.30606079101562	102
2.28706359863281	100
2.27905273437500	98
2.23915100097656	96
2.21427917480468	94
2.14721679687500	92
2.12348937988281	90
2.03254699707031	84

Con los datos de subida y bajada de la tabla 12, se procede a realizar una regresión polinómica de 2do orden, individualmente se obtiene como resultado la figura 68, donde se muestra la gráfica y la ecuación (21) del comportamiento de la temperatura de subida.

$$y = 38.683x^2 - 63.446x + 40.106 \quad (21)$$

Adicional también se muestra la figura 69, donde se puede ver la gráfica y la ecuación (22) del comportamiento de la temperatura de bajada.

$$y = 39.746x^2 - 83.347x + 86.465 \quad (22)$$

El coeficiente de correlación de la ecuación (21) es de 0.99 y de la ecuación (22) es de 0,98.

Este procedimiento es necesario para conocer la conversión del valor digital a la variable física de temperatura en grados Celsius [°C] del sensor TT-100 y calibrar la lectura ingresada por el canal 0 del SoftPLC V1.0.

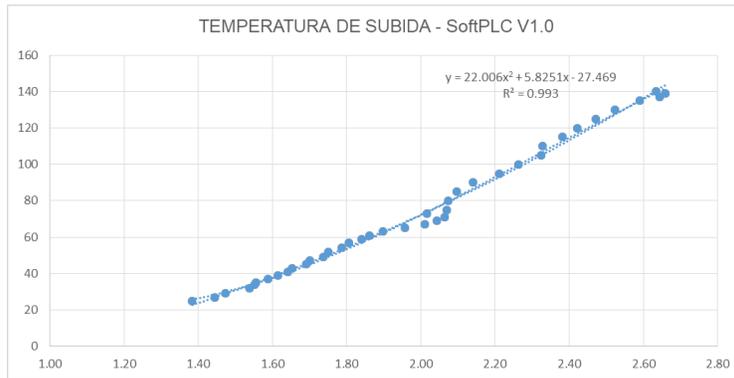


Fig 66. Grafica de Temperatura de subida TT-100 con SoftPLC V1.0. [Autor]

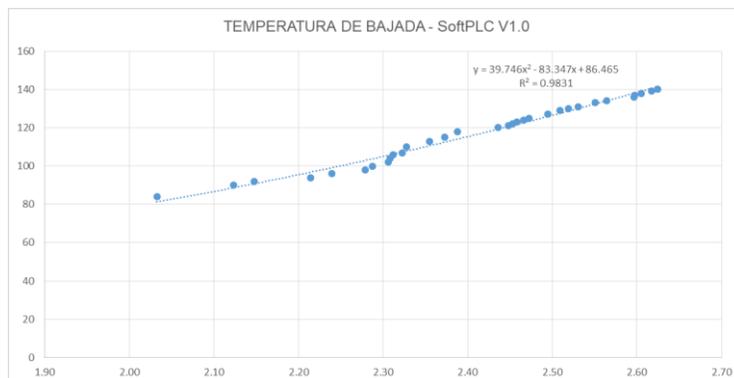


Fig 67. Grafica de Temperatura de bajada TT-100 con SoftPLC V1.0. [Autor]

**- Linealización sensor de temperatura TT-100 con CompactLogix L23E**

Se realiza el proceso de caracterización de TT-100, mediante la toma de datos y validación de la señal entregada por el controlador *CompactLogix L23E*, de acuerdo al patrón de medida con la lectura de la termocupla tipo K conectada al dispositivo UT39C, en la tabla 13 se muestran los valores obtenidos en EU (Unidades de ingeniera) del aumento y disminución de temperatura dentro de la caldera, en los datos de bajada se observan menos datos debido a que la caldera no cuenta con un actuador que realice el trabajo de enfriamiento o disminución de la temperatura.

Tabla 13. Datos de subida y bajada TT-100 con CompactLogix L23E. [Autor]

DATOS DE SUBIDA		DATOS DE BAJADA	
EU COMPACTLOGIX	TEMPERATURA UT39C	EU COMPACTLOGIX	TEMPERATURA UT39C
4822	18	10030	144
4942	20	9981	143

5091	22
5213	24
5319	26
5420	28
5519	30
5616	32
6013	41
6067	42
6183	45
6203	46
6268	48
6368	50
6435	52
6515	54
6596	56
6673	58
6756	60
6844	62
6923	64
6990	66
7065	68
7133	70
7208	72
7294	74
7384	76
7452	78
7532	80
7628	82
7690	84
7768	86
7849	88
7919	90
7993	92
8076	94
8157	96
8216	98
8289	100
8379	102
8455	104
8533	106
8610	108
8692	110
8768	112

9943	142
9890	141
9783	138
9715	137
9673	136
9651	135
9620	134
9563	133
9506	132
9499	131
9451	130
9374	128
9312	127
9285	126
9258	125
9191	124
9104	122
9029	120
8933	117
8843	115
8620	110
8539	108
8480	106
8307	102
8193	99
8138	98
8061	96
7990	94
7871	91
7844	90

8833	114
8932	116
8991	118
9055	120
9140	122
9224	124
9293	126
9373	128
9451	130
9530	132
9615	134
9702	136
9767	138
9855	140

Con los datos de subida y bajada de la tabla 13, se procede a realizar una regresión polinómica de 2do orden, individualmente se obtiene como resultado la figura 70, donde se muestra la gráfica y la ecuación (23) del comportamiento de la temperatura de subida.

$$y = 5E-07x^2 + 0.0175x - 81.22 \quad (23)$$

Adicional también se muestra la figura 71, donde se puede ver la gráfica y la ecuación (24) de la regresión lineal del comportamiento de la temperatura de bajada.

$$y = 0.0246x - 102.78 \quad (24)$$

El coeficiente de correlación de la ecuación (23) y (24) es de 0.99.

Este procedimiento es necesario para conocer la conversión del valor digital a la variable física de temperatura en grados Celsius [°C] del sensor TT-100 y calibrar la lectura ingresada por el canal 0 del CompactLogix L23E

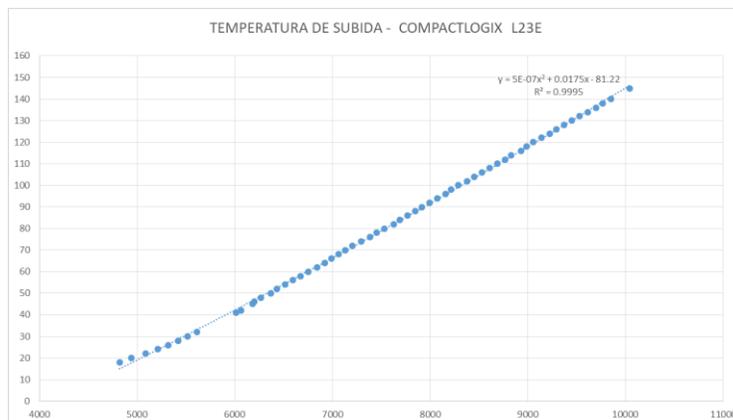


Fig 68. Grafica Temperatura subida TT-100 con CompactLogix L23E. [Autor]

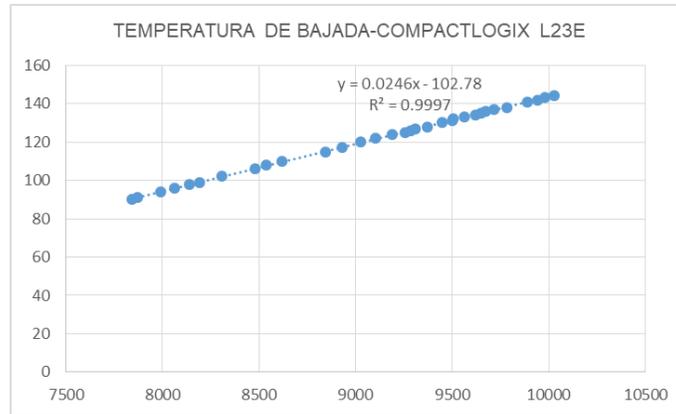


Fig 69. Grafica Temperatura bajada TT-100 con CompactLogix L23E. [Autor]

**- Linealización sensor de presión PT-100 con SoftPLC V1.0**

Se realiza el proceso de caracterización de PT-100, mediante la toma de datos y validación de la señal entregada por el controlador SoftPLC V1.0, de acuerdo al patrón de medida con la lectura del manómetro PI-106 instalado en la caldera, Se toman datos desde el inicio del proceso de 0 PSI hasta 40 PSI, en la tabla 14 se muestran los valores obtenidos del aumento y disminución de presión durante el proceso de salida de vapor hacia el tanque de calentamiento de la dosificadora.

Tabla 14. Datos de subida y bajada PT-100 con SoftPLC V1.0. [Autor]

DATOS DE SUBIDA		DATOS DE BAJADA	
VOLTAJE	PRESIÓN MANÓMETRO	VOLTAJE	PRESIÓN MANÓMETRO
2.10784912109375	42	2.13569641113281	42
1.99401855468750	40	2.03887939453125	40
1.89361572265625	38	1.94381713867187	38
1.79069519042968	36	1.85340881347656	36
1.70104980468750	34	1.76956176757812	34
1.59736633300781	32	1.70654296875000	32
1.48796081542968	30	1.59667968750000	30
1.36840820312500	28	1.51947021484375	28
1.29676818847656	26	1.42807006835937	26
1.19224548339843	24	1.34246826171875	24
1.08642578125000	22	1.26235961914062	22
0.98609924316406	20	1.17767333984375	20
0.87989807128906	18	1.08528137207031	18
0.76881408691406	16	1.00990295410156	16
0.67810058593750	14	0.91873168945313	14
0.55564880371094	12	0.84312438964844	12
0.46028137207031	10	0.74249267578125	10
0.35804748535156	8	0.65444946289063	8
0.24833679199219	6	0.56671142578125	6

0.12924194335938	4
0.03837585449219	0

0.46981811523438	4
0.36483764648438	0

Con los datos de subida y bajada de la tabla 14, se procede a realizar una regresión polinómica de 2do orden, individualmente se obtiene como resultado la figura 72, donde se muestra la gráfica y la ecuación (25) del comportamiento de presión de subida.

$$y = -0.0357x^2 + 19.778x + 0.6752 \quad (25)$$

Adicional también se muestra la figura 73, donde se puede ver la gráfica y la ecuación (26) del comportamiento de presión de bajada.

$$y = -0.4242x^2 + 24.414x - 8.0328 \quad (26)$$

El coeficiente de correlación de la ecuación (25) y (26) es de 0.99.

Este procedimiento es necesario para conocer la conversión del valor digital a la variable física de presión en libra de fuerza por pulgada cuadrada [PSI] del sensor PT-100 y calibrar la lectura ingresada por el canal 1 del SoftPLC V1.0.

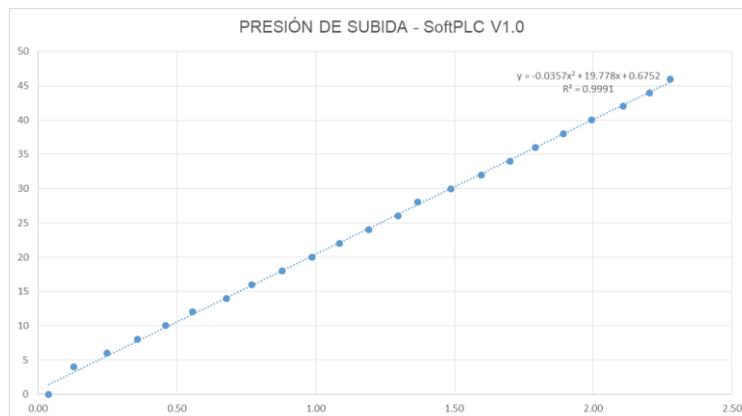


Fig 70. Grafica de presión de subida PT-100 con SoftPLC V1.0. [Autor]

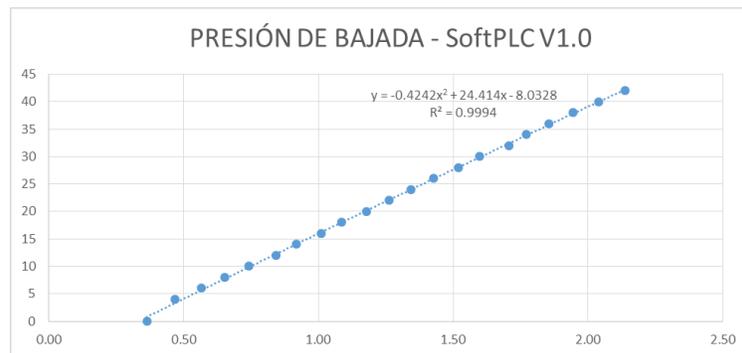


Fig 71. Grafica de presión de bajada PT-100 con SoftPLC V1.0. [Autor]

### - **Linealización sensor de presión PT-100 con CompactLogix L23E**

Se realiza el proceso de caracterización de PT-100, mediante la toma de datos y validación de la señal entregada por el controlador *CompactLogix L23E*, de acuerdo

al patrón de medida con la lectura del manómetro PI-106 instalado en la caldera, se toman datos desde el inicio del proceso de 0 PSI hasta 40 PSI, en la tabla 15 se muestran los valores obtenidos del aumento y disminución de presión durante el proceso de salida de vapor hacia el tanque de calentamiento de la dosificadora.

Tabla 15. Datos de subida y bajada PT-100 con CompactLogix L23E. [Autor]

DATOS DE SUBIDA		DATOS DE BAJADA	
EU COMPACTLOGIX	PRESION MANOMETRO ASTRO GAUGES	EU COMPACTLOGIX	PRESION MANOMETRO ASTRO GAUGES
255	0	2114	40
348	4	1982	38
446	6	1974	36
543	8	1888	34
648	10	1829	32
742	12	1763	30
850	14	1683	28
938	16	1598	26
1034	18	1517	24
1134	20	1442	22
1226	22	1356	20
1336	24	1270	18
1414	26	1206	16
1515	28	1121	14
1591	30	1030	12
1709	32	950	10
1793	34	858	8
1889	36	775	6
1982	38	686	4
2080	40	613	0

Con los datos de subida y bajada de la tabla 15, se procede a realizar una regresión polinómica de 2do orden, individualmente se obtiene como resultado la figura 74, donde se muestra la gráfica y la ecuación (27) del comportamiento de presión de subida.

$$y = 4E-07x^2 + 0.0199x - 3.0082 \quad (27)$$

Adicional también se muestra la figura 75, donde se puede ver la gráfica y la ecuación (28) del comportamiento de presión de bajada.

$$y = 6E-09x^2 + 0.0254x - 14.301 \quad (28)$$

El coeficiente de correlación de la ecuación (27) y (28) es de 0.99.

Este procedimiento es necesario para conocer la conversión del valor digital a la variable física de presión en libra de fuerza por pulgada cuadrada [PSI] del sensor PT-100 y calibrar la lectura ingresada por el canal 1 del CompactLogix L23E.

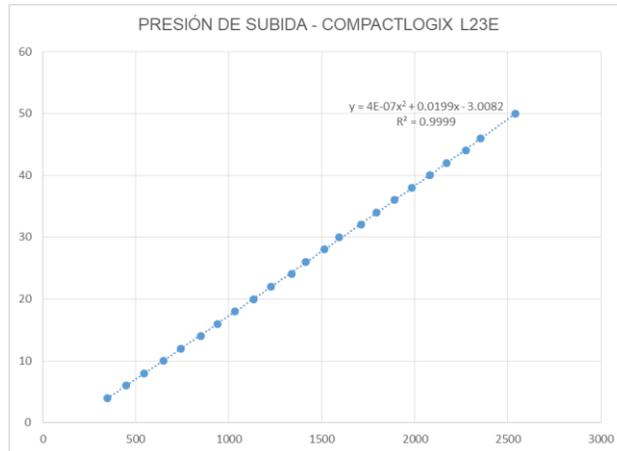


Fig 72. Grafica de presión de subida PT-100 con CompactLogix L23E. [Autor]

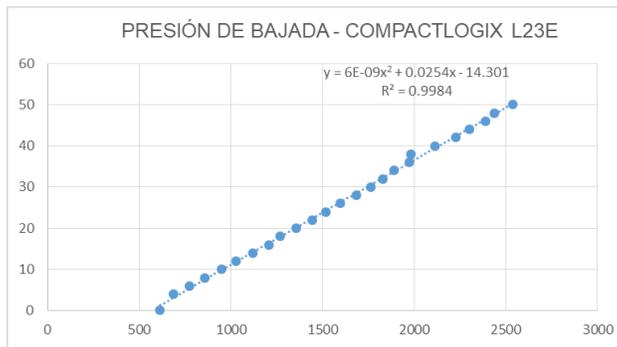


Fig 73. Grafica de presión de bajada PT-100 con CompactLogix L23E. [Autor]

### 6.5.2 Caracterización del sensor del actuador CV-105

Se realiza el proceso de caracterización del sensor de posición instalado en el mecanismo para controlar la apertura de la válvula CV-105, se toman datos desde la apertura total hasta el cierre total de la válvula, en la tabla 16 se muestran los valores obtenidos de la posición del sensor (valor digital) respecto al ángulo de apertura de la válvula durante el proceso de salida de vapor hacia el tanque de calentamiento de la dosificadora.

Tabla 16. Datos de linealización del sensor de posición para CV-105. [Autor]

SENSOR DE POSICIÓN	ANGULO
0	0
98	90
200	180
303	270
403	360
503	450
608	540

714	630
814	720
918	810
1018	900
1125	990
1225	1080
1330	1170
1430	1260
1530	1350
1634	1440
1743	1530
1843	1620
1943	1710

Con los datos de linealización del sensor de posición para CV-105 de la tabla 16, se procede a realizar una regresión polinómica de 2do orden y se obtiene como resultado la figura 76, donde se muestra la gráfica y la ecuación (29) del comportamiento del sensor.

$$y = -5E-06x^2 + 0.8889x \quad (29)$$

El coeficiente de correlación de la ecuación (29) es de 1.

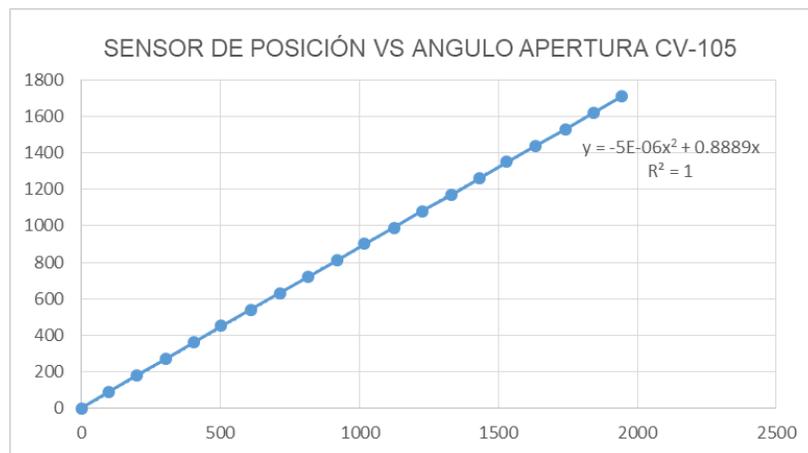


Fig 74. Grafica del sensor de posición vs ángulo de apertura CV-105. [Autor]

Este procedimiento es necesario para conocer los grados de apertura que realiza la válvula tipo aguja teniendo en cuenta las 4.75 vueltas que ejecuta para su apertura total, se obtiene como resultado, la ecuación (30):

$$4.75 \text{ vueltas} = 1710^\circ \quad (30)$$

### 6.5.3 Proceso de identificación de la planta

Al no obtener un modelo matemático que nos permita determinar el comportamiento del sistema a un estímulo, es necesario realizar experimentos a través de estímulos,

para ello se debe generar señales de diferentes formas y amplitudes como La señal paso, la señal paso modificada, la señal paso cuadrada, y una señal pseudoaleatoria. Para la generación de estas señales se ha utilizado la herramienta Stateflow Matlab, y de allí se exporto a código PLCopen XML como Texto estructurado y se implementa en el entorno de desarrollo de CODESYS.

Estas señales se inyectan al controlador de la válvula proporcional CV-105 y se toman los datos de temperatura en el tanque de calentamiento de la planta de dosificación tomando como punto de operación de 80 °C con una apertura de la válvula en un 20% aproximadamente.

La primer señal aplicada es la señal paso con amplitud de 360, esta amplitud es el valor de apertura que se le envía al controlador de la válvula para que este de la orden al motor para que abra la válvula, en la figura 77, se puede observar la señal, y la respuesta de temperatura en el tanque de área de calentamiento en la dosificadora el cual está con temperatura inicial de 27°C con un tiempo de muestreo de 600ms.

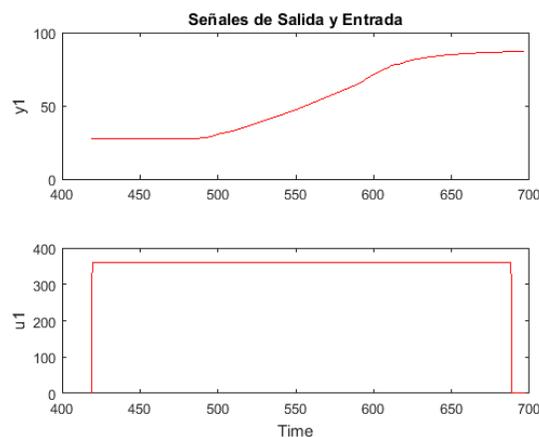


Fig 75. Señal temperatura salida ( $y1$ ) señal paso apertura CV-105 ( $u1$ ). [Autor]

Al ver la figura 77, la señal  $y1$  (señal de salida) tiene un retardo de 80 segundos mientras aumenta de temperatura en el tanque. Cabe mencionar que esta fue una de las pruebas de la señal paso donde se obtienen mayor cantidad de información del sistema con esta señal.

La segunda señal aplicada a la válvula es la señal paso modificada que abre la válvula a 270° y pasado 650 segundos cambia de apertura a 360°, en la figura 78, está la señal de salida ( $y1$ ) de temperatura en el tanque 2 de la dosificadora con valor inicial de 28°C y la señal de entrada ( $u1$ ) en la apertura de la válvula con tiempo de muestreo de 300ms

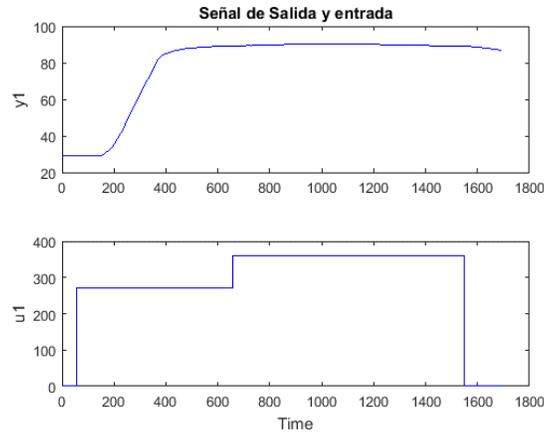


Fig 76. Señal salida temp en tanque y señal paso modificada CV-105. [Autor]

En la figura 78, se observa un retardo de aproximadamente de 100 segundos para el cambio de temperatura en el tanque y luego aumenta la temperatura hasta llegar a 80°C pasado los 650 segundos se abre la válvula 90° más y aumenta la temperatura a 90°C donde se mantiene.

La tercer señal es la paso cuadrada que se ajusta el periodo de apertura de la válvula a 120 segundos y el tiempo de muestra es de 900ms; el primer punto de apertura es de 360° donde la señal de salida no tiene cambio de temperatura, después la válvula abre a 450° en ese instante de la apertura no hay cambio significativo y 25 segundos después la temperatura empieza a aumentar llegando a 40° C, la válvula cierra el paso de flujo de vapor a 270° causando que la temperatura en el tanque aumente lentamente con el pasar del tiempo y la oscilación de la apertura de la válvula como se puede ver en la figura 79.

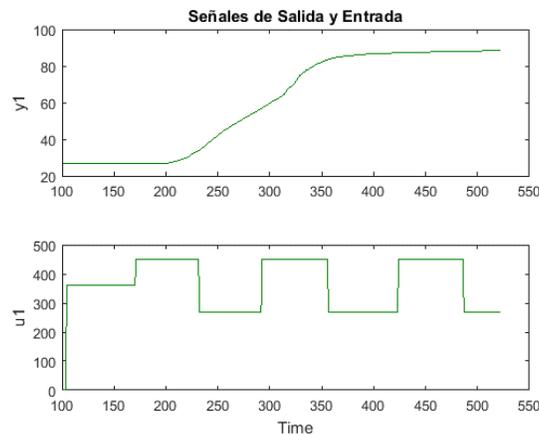


Fig 77. Señal salida temperatura tanque y señal paso cuadrada CV-105. [Autor]

Por último se aplica la señal pseudoaleatoria la cual tiene una variación entre 45° y 300°, el periodo entre 60 y 100 segundos; en la señal de salida la temperatura comienza aumentar luego de 300 segundos aproximadamente, es decir en las primeras variaciones no ocurre un cambio significativo en la temperatura, después

empieza la temperatura acorde a la apertura que se esté dando en la válvula, el tiempo de muestreo es de 900ms y se puede ver la señal de salida y de entrada en la figura 80.

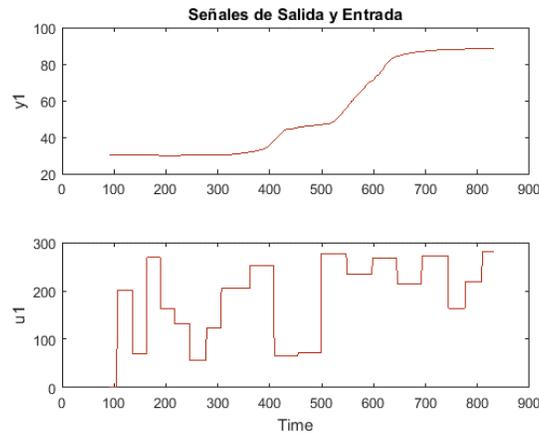


Fig 78. Señal salida temp en tanque y señal pseudoaleatoria CV-105. [Autor]

Con la información de cada una de las pruebas anteriores y por medio del 'Toolbox System Identification' de Matlab se procesaron los datos para encontrar la mejor aproximación al comportamiento del sistema para ello se realizaron varias estimaciones para cada señal del experimento encontrando la mejor para cada señal, a continuación se presentara los mejores ajustes encontrados para cada señal.

Para la señal paso el ajuste a la estimación es de 86.19%, la función de transferencia se presenta en la ecuación 31 y la respuesta al paso en la figura 81.

$$G(z)_1 = z^{-120} * \frac{0.001232}{1 - 0.9963z^{-1}} \quad (31)$$

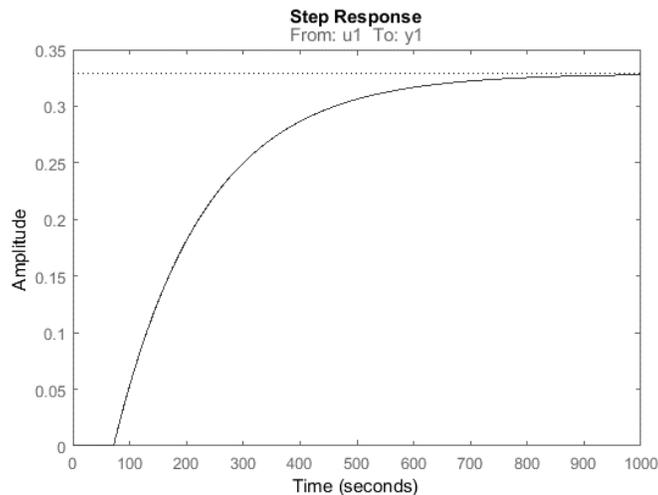


Fig 79. Respuesta a la señal paso de  $G(z)_1$ . [Autor]

En la señal paso modificada el ajuste a la estimación es de 83.73%, la función de transferencia se presenta en la ecuación (32) y la respuesta al paso en la figura 82.

$$G(z)_2 = z^{-300} * \frac{0.0005743z^{-1}}{1-0.9985z^{-1}} \quad (32)$$

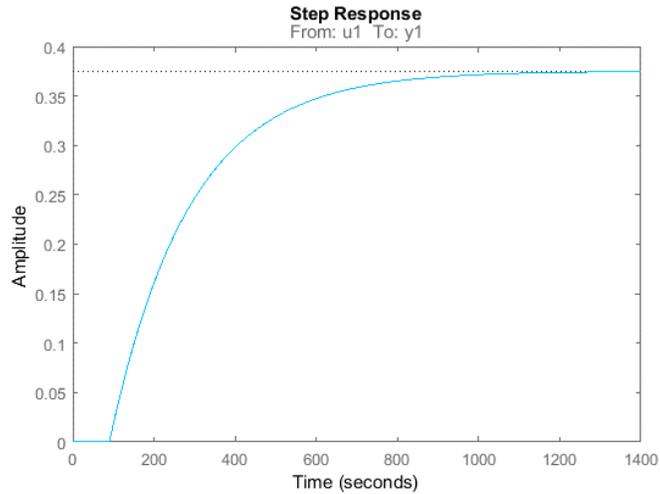


Fig 80. Respuesta a la señal paso de  $G(z)_2$ . [Autor]

En la señal paso cuadrada el ajuste a la estimación es de 81.76%, la función de transferencia se presenta en la ecuación (33) y la respuesta al paso en la figura 83, 83.39%

$$G(z)_3 = z^{-100} * \frac{0.001535z^{-1}}{1-0.9948z^{-1}} \quad (33)$$

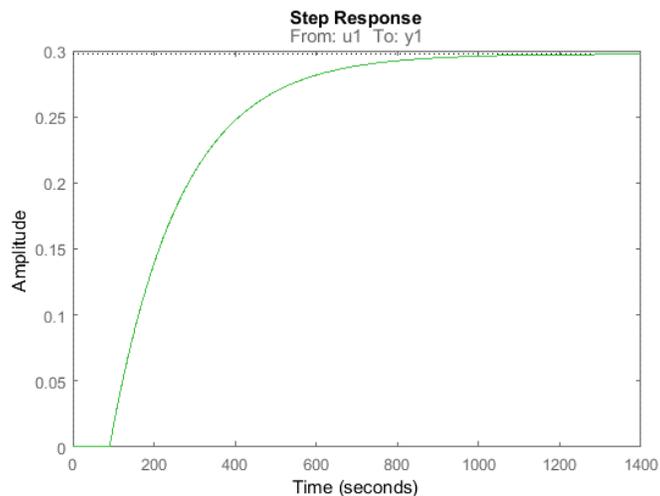


Fig 81. Respuesta a la señal paso de  $G(z)_3$ . [Autor]

En la señal pseudoaleatoria el ajuste a la estimación es de 83.39%, la función de transferencia se presenta en la ecuación (34) y la respuesta al paso en la figura 84.

$$G(z)_4 = z^{-210} * \frac{2.493 * 10^{-5} z^{-1}}{1 - 1.992z^{-1} + 0.9916z^{-2}} \quad (34)$$

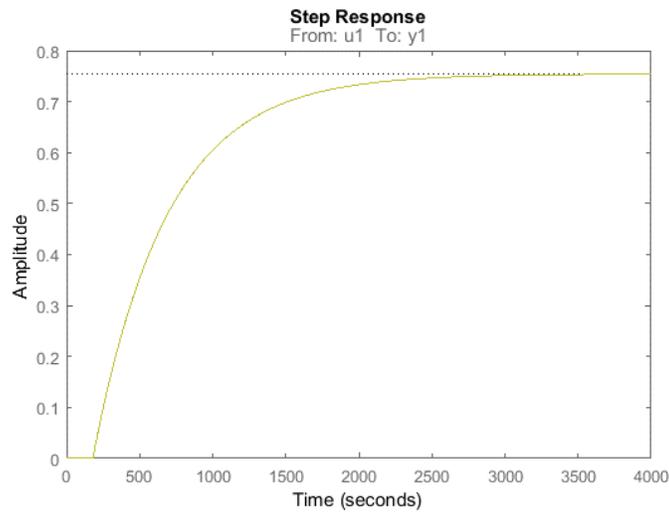


Fig 82. Respuesta a la señal paso de G4(z)4. [Autor]

A continuación se muestra en la tabla 17 las características de cada función de transferencia de la planta según la señal aplicada.

Tabla 17. Características de FT según señal de identificación. [Autor]

SEÑAL	FUNCIÓN DE TRANSFERENCIA	% DE ESTIMACIÓN
PASO	$G(z)_1 = z^{-120} * \frac{0.001232}{1 - 0.9963z^{-1}}$	86.19
PASO MODIFICADA	$G(z)_2 = z^{-300} * \frac{0.0005743z^{-1}}{1 - 0.9985z^{-1}}$	83.73
PASO CUADRADA	$G(z)_3 = z^{-100} * \frac{0.001535z^{-1}}{1 - 0.9948z^{-1}}$	81.76
PSEUDOALEATORIA	$G(z)_4 = z^{-210} * \frac{2.493 * 10^{-5} z^{-1}}{1 - 1.992z^{-1} + 0.9916z^{-2}}$	83.39

Con la información de la tabla 17, se decide escoger el sistema con la aproximación más cercana al 100% y con menos complejidad en la función de transferencia la cual es  $G(z)_1$

#### 6.5.4 Diseño del controlador

En el diseño del controlador es necesario tener el modelo matemático que muestra el comportamiento físico de la planta el cual es  $G(z)_1$  que se obtiene del proceso realizado anteriormente. Para el diseño se usa el 'Toolbox sisotool' de Matlab en

donde se ajustan los parámetros para encontrar la función de transferencia del controlador que debe tener como parámetros de diseño:

- $M_p$  (máximo sobre impulso)  $< 10\%$
- Tiempo de estabilización igual o inferior al tiempo de estabilización de la planta en lazo abierto
- $e_{ss}$  (Error de estado estacionario)  $< 3\%$ .

El controlador que brinda los anteriores parámetros de diseño es un controlador PID el cual se diseña para la arquitectura de señal de referencia, controlador y planta como se ve en la se ve en la figura 85.

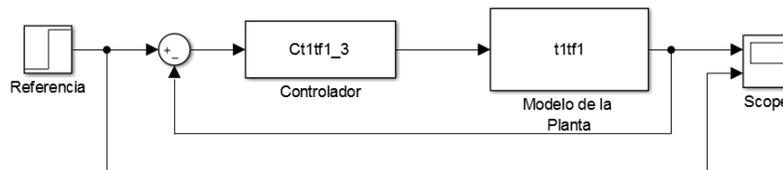


Fig 83. Arquitectura de control Diseñada e Implementada. [Autor]

La sintonización del PID se realiza mediante 'Automated Tuning' que da un ajuste inicial de control sobre la planta, luego se ajusta manualmente para llevar el control a los parámetros de diseño establecidos; en la figura 86, se ve la respuesta al paso de la planta con un tiempo de estabilización de 450 segundos, la función de transferencia obtenida para el controlador el tiempo de estabilización se reduce a 233 segundos y un  $M_p$  de 1.91%, ver figura 87.

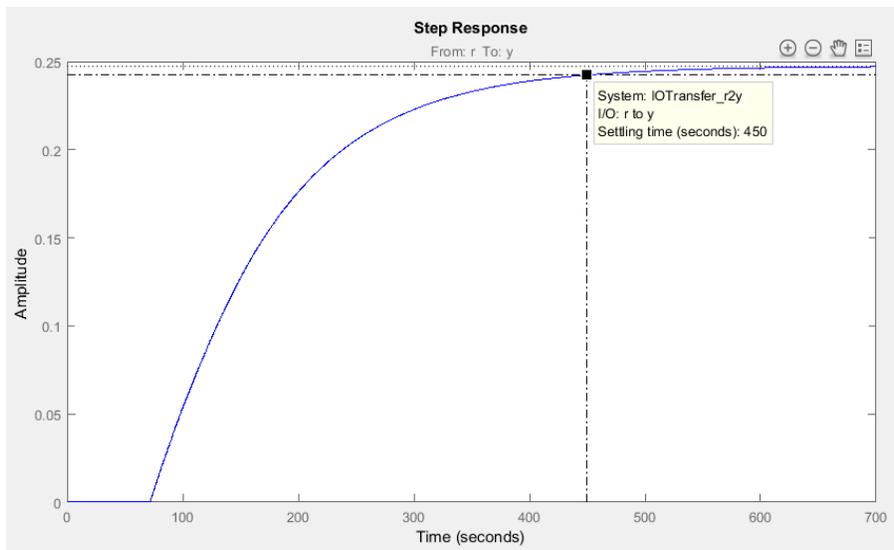


Fig 84. Tiempo de estabilización de la planta lazo abierto igual a 450s. [Autor]

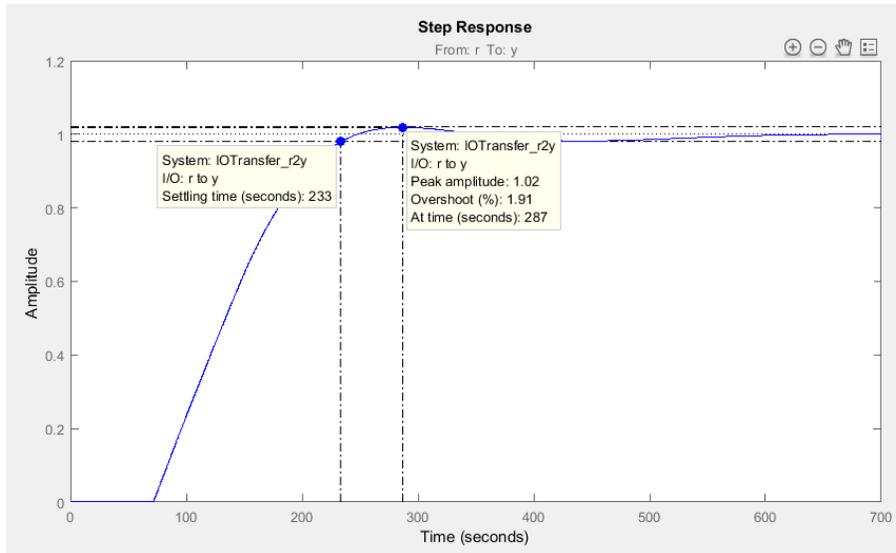


Fig 85. Tiempo de estabilización de planta en lazo cerrado = 233s. [Autor]

La función de transferencia del controlador PID está definida por la ecuación (35):

$$C(z) = \frac{6.6577 * (z - 0.9962) * (z - 0.9835)}{(z - 1) * (z - 0.9675)} \quad (35)$$

Para llevar el controlador a una forma que sea posible implementar dentro del SoftPLC V1.0, es necesario transformar la función de transferencia discreta en una ecuación en diferencias de tiempo discreto; a continuación se describe el proceso realizado.

Primero se establece el controlador en una ecuación de salida sobre entrada, ecuación (36)

$$C(z) = \frac{Y(z)}{E(z)} = \frac{6.6577 * (z - 0.9962) * (z - 0.9835)}{(z - 1) * (z - 0.9675)} \quad (36)$$

Luego se multiplican los factores de la ecuación (36), resultando la ecuación (37).

$$C(z) = \frac{Y(z)}{E(z)} = \frac{6.6577z^2 - 13.18024z + 6.2999}{z^2 - 1.9675z + 0.9675} \quad (37)$$

Como los polinomios tienen potencias positivas, se debe escribir la expresión en potencias negativas para luego realizar la transformada Z inversa, ecuación (38).

$$C(z) = \frac{Y(z)}{E(z)} = \frac{6.6577z^2 - 13.18024z + 6.2999}{z^2 - 1.9675z + 0.9675} * \left( \frac{z^{-2}}{z^{-2}} \right) \quad (38)$$

De la ecuación (38) al multiplicar los factores se obtiene ecuación (39).

$$C(z) = \frac{Y(z)}{E(z)} = \frac{6.6577 - 13.18024z^{-1} + 6.2999z^{-2}}{1 - 1.9675z^{-1} + 0.9675z^{-2}} \quad (39)$$

El siguiente paso es despejar  $Y(z)$  en la ecuación (39), se obtiene la ecuación (40).

$$Y(z)[1 - 1.9675z^{-1} + 0.9675z^{-2}] = E(z)[6.6577 - 13.18024z^{-1} + 6.2999z^{-2}] \quad (40)$$

Se operan los términos en la ecuación (40) para obtener la ecuación (41) donde se despeja  $Y(z)$ .

$$\begin{aligned} Y(z) - 1.9675z^{-1}Y(z) + 0.9675z^{-2}Y(z) &= 6.6577E(z) - 13.18024z^{-1}E(z) + 6.2999z^{-2}E(z) \\ Y(z) &= 1.9675z^{-1}Y(z) - 0.9675z^{-2}Y(z) + 6.6577E(z) - 13.18024z^{-1}E(z) + 6.2999z^{-2}E(z) \end{aligned} \quad (41)$$

Ya teniendo organizada la ecuación (41) se procede a realizar la transformada  $Z$  inversa, que se observa en la ecuación (42).

$$\begin{aligned} Z^{-1}\{Y(z) - 1.9675z^{-1}Y(z) + 0.9675z^{-2}Y(z) + 6.6577E(z) - 13.18024z^{-1}E(z) + 6.2999z^{-2}E(z)\} \\ Y[n] = 1.9675 * Y[n-1] - 0.9675 * Y[n-2] + 6.6577 * E[n] - 13.18024 * E[n-1] + 6.2999 * E[n-2] \end{aligned} \quad (42)$$

Con la ecuación en diferencias presentada en la ecuación (42), se asignan las variables para poder obtener el pseudocódigo que se implementa en el algoritmo de control para la planta como se ve en la tabla 18.

Tabla 18. Asignación de variables para el código. [Autor]

$Y[n]$	$\rightarrow$	$m1$
$Y[n-1]$	$\rightarrow$	$m2$
$Y[n-2]$	$\rightarrow$	$m3$
$E[n]$	$\rightarrow$	$m4$
$E[n-1]$	$\rightarrow$	$m5$
$E[n-2]$	$\rightarrow$	$m6$

Reemplazando las variables en la ecuación (42) como se indica en la tabla 18 el pseudocódigo es de la siguiente forma ecuación (43):

$$m1 = 1.9675 * m2 - 0.9675 * m3 + 6.6577 * m4 - 13.18024 * m5 + 6.2999 * m6 \quad (43)$$

Con las asignaciones equivalentes a ecuación (44), con tiempo de muestreo de 600ms.

$$\begin{aligned} m6 &= m5 \\ m5 &= m4 \\ m3 &= m2 \\ m2 &= m1 \end{aligned} \quad (44)$$

## 7. PRUEBAS Y RESULTADOS

### 7.1 VALVULA PROPORCIONAL CV-105

En el capítulo 6.2.3 Acondicionamiento de la etapa final de la caldera, se describe la construcción e implementación del mecanismo y de la tarjeta que controla la válvula proporcional CV-105, como resultado final se obtiene el actuador de la válvula tipo aguja que permite el paso controlado de vapor hacia la etapa de dosificación, en la figura 88 se muestra el diseño final de la válvula proporcional CV-105, con un valor no superior a los COP \$550.000.



Fig 86. CV-105 Final. [Autor]

### 7.2 SOFTPLC V1.0

Una de las condiciones detectadas en el tablero de control X1, es la imposibilidad de instalar el SoftPLC V1.0 por sus dimensiones; por tal motivo se realiza una expansión al tablero de control X1, la figura 89 muestra el resultado final de la expansión del tablero X1, donde también reside el controlador de la válvula CV-105 y en la parte superior la pantalla HMI.



Fig 87. Tablero de control X1 Final. [Autor]

En las instalaciones de cableado para tableros de control es absolutamente necesario contar con una buena documentación de todos los componentes instalados (Ver anexo 6). Esta documentación muestra el etiquetado de dichos componentes, (todos los componentes de los tableros fueron marcados como se observa en la figura 90) de tal manera que su localización sea rápida y precisa, facilitando al mismo tiempo las labores de conexión y mantenimiento.



Fig. 98. Etiquetas finales del tablero de control X1. [Autor]

El resultado final del SoftPLC v1.0 se observa en la figura 91, al lado izquierdo de él, se visualiza el controlador de la válvula CV-105.



Fig. 99. Instalación final de SoftPLC V1.0. [Autor]

### 7.3 CALDERA

La reestructuración y expansión del tablero de control X1, marquillaje y etiquetado de todos los dispositivos de la caldera con marquillas en acrílico para los tag descritos en el P&ID mostrado en la figura 92 y en el anexo 7. La redistribución e instalación de nueva tubería para el paso de agua o salida de vapor, el cambio del tanque de reserva TK-002 por uno completamente nuevo de acero inoxidable, la construcción e implementación de un actuador para la válvula tipo aguja y SoftPLC V1.0, que permiten el paso de salida de vapor a la etapa de dosificación, son los resultados más importantes que obtuvo la caldera en este proyecto, en la figura 93, se muestra la caldera final.

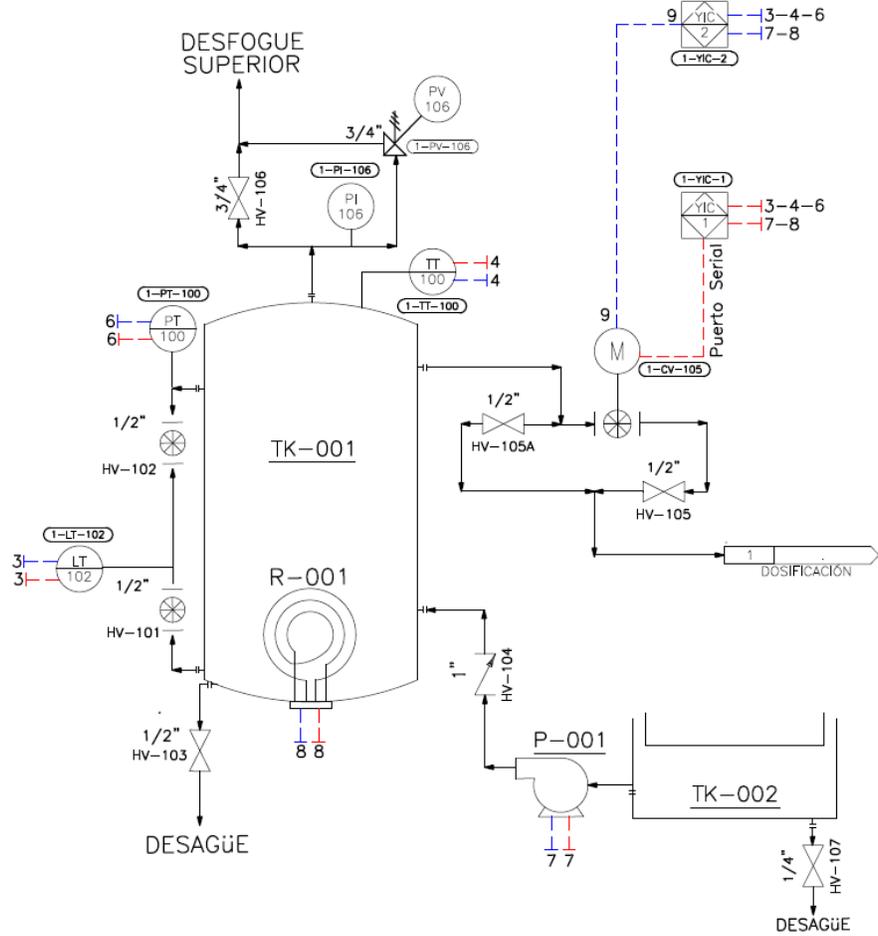


Fig 90. Diagrama P&ID final de la caldera. [Autor]



Fig 91. Estructura Física final de la Caldera. [Autor]

## 7.4 CONTROLADOR

El controlador obtenido con la implementación del capítulo “6.5.3 Proceso de identificación de la planta”, es un controlador PID, gracias a la señal paso inyectada para observar el comportamiento que esta tiene, una función de transferencia de primer orden se obtuvo de este procedimiento, paso siguiente fue el desarrollo planteado en el capítulo “6.5.4 Diseño del controlador”, donde se obtuvo como resultado la ecuación (43) y gracias a las memorias de programa descritas en la tabla 18, se proceden a programarse y ejecutarse en el SoftPLC V1.0.

A continuación se muestra la descripción de los resultados obtenidos del controlador, en la figura 92, se observa un diagrama resumen con el proceso de identificación realizado, teniendo en cuenta las 4 estructuras probadas, y como modelos resultantes se elige el de la señal paso con un % de estimación de 86.19

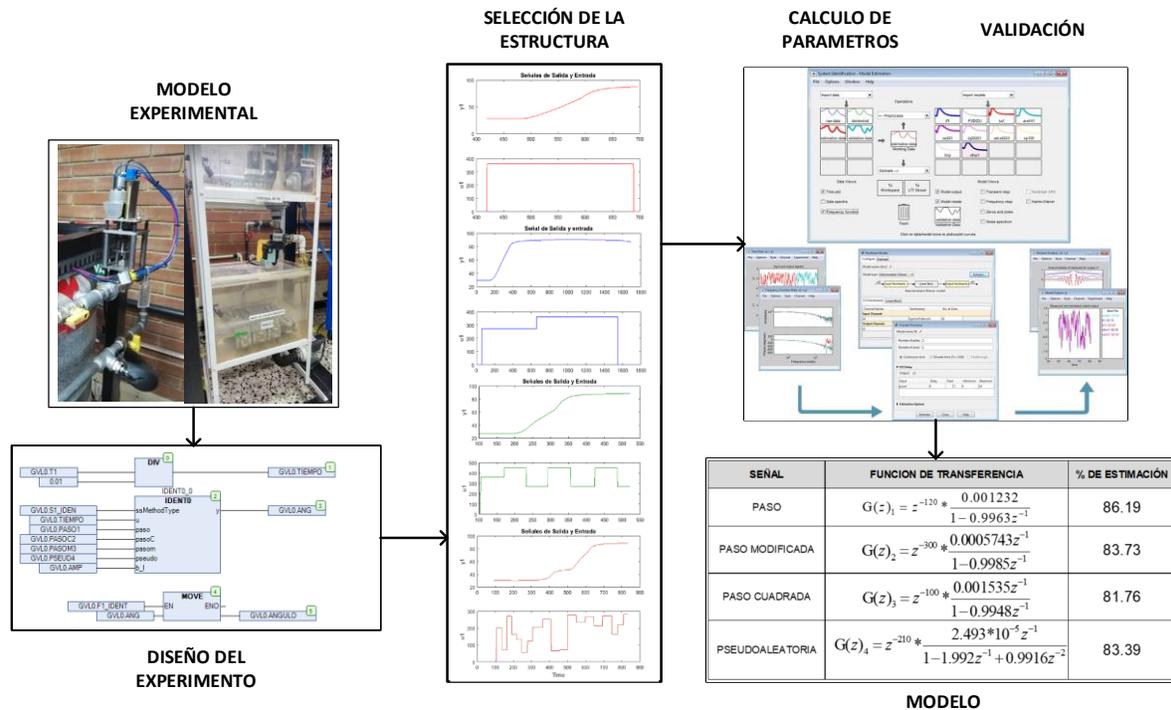


Fig 92. Resultados identificación de procesos. [Autor]

En la figura 93, se observa la función de transferencia obtenida del proceso anterior y la gráfica de la respuesta del sistema, donde se obtiene lo siguiente:

$$K_p = 3,77$$

$$K_i = 0,0214$$

$$K_d = 53,4$$

$$T_f = 18,4$$

$$T_s = 0,6 \text{ seg}$$

$$C(z) = \frac{6.6577*(z-0.9962)*(z-0.9835)}{(z-1)*(z-0.9675)}$$

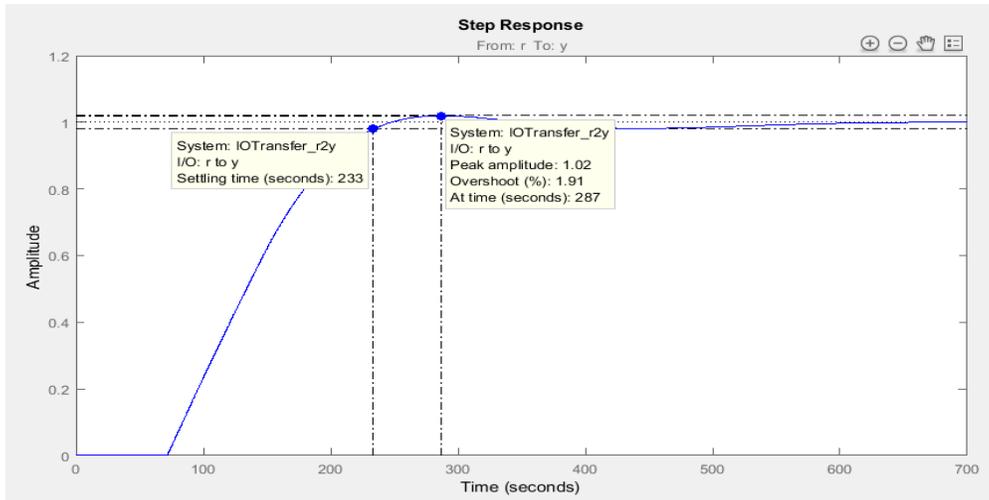


Fig 93. Respuesta del controlador diseñado. [Autor]

Teniendo la función de transferencia del proceso se procede a generar un pseudocódigo para programar el comportamiento del controlador, en la figura 94, se visualiza la respuesta del controlador vista desde la HMI del proyecto, en ella se inserta un valor de referencia a 60°C, la temperatura del tanque de la dosificadora está cerca al valor de la referencia, el ángulo de apertura la válvula CV-105 es pequeño ya que está cerca al valor de referencia y por último se visualiza la temperatura y presión de la caldera. Se obtiene un tiempo de estabilización en lazo cerrado de 233 segundos con un error de estado estacionario igual a cero y un sobre impulso cerca del 2% teniendo en cuenta el proceso de identificación y las condiciones establecidas para el diseño del controlador.

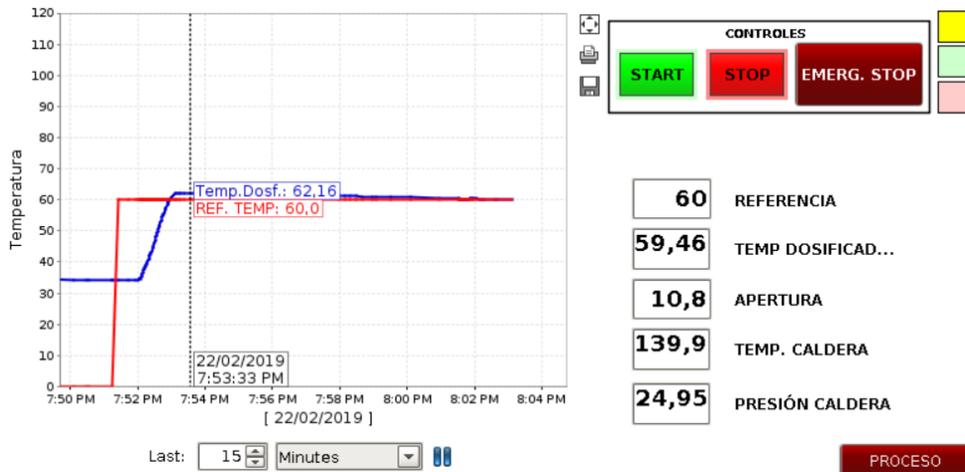


Fig 94. Respuesta final del controlador en HMI. [Autor]

## 8. CONCLUSIONES

El desarrollo mecánico implementado e instalado para la válvula tipo aguja que permite la salida controlada de vapor, se elaboró desde cero ya que adquirir un actuador proporcional para esta válvula costaba alrededor de USD\$1200, por este motivo se implementaron dos tipos de mecanismos con la ayuda de un motor DC y un sensor de posición, su comportamiento es muy similar a los actuadores industriales del mercado, el valor de la construcción del actuador para la válvula es de alrededor USD\$165, esto quiere decir que costo 7 veces menos que un actuador industrial comercializado hoy en día.

La automatización industrial, se basa en plataformas de programación abiertas o cerradas según fabricante, para controlar procesos industriales, esto conlleva a que el estudiante a la hora de ejercer su profesión como ingeniero, debe conocer y manejar la configuración y programación de más de una plataforma de programación, se implementó un SoftPLC V1.0, que cuenta con la mayoría de las características técnicas de un PLC del mercado de baja gama, Así la caldera se puede manejar de forma independiente y se convierte en un sistema multiplataforma funcional donde se pueden realizar diferentes prácticas de laboratorio, alternando de PLC's para controlar el proceso.

En comparación con proyectos anteriores implementados por otros estudiantes de la Universidad Distrital que de igual forma a este proyecto manejan controladores basados en computadores de placa reducida como la Raspberry PI con módulo de expansión de I/O por comunicación I2C de resolución de 8 bits; El SoftPLC V1.0 maneja comunicación SPI con resolución de 16 bits para lectura y escritura de I/O análogas y digitales, esto nos ayuda a tener una mayor precisión en el proceso y se comprueba que la comunicación SPI a 16 bits funciona y puede ser aplicable a futuros proyectos donde se requiera usar resoluciones mayores a 8 bits.

Comparando la respuesta de la simulación del diseño del controlador en lazo abierto la planta responde como un sistema de primer orden con tiempo de estabilización de 450seg, implementando el algoritmo de control su tiempo de estabilización se ve reducido a 233seg y su Mp es del 1,9% el cual se encuentra dentro de los parámetros de diseño.

El sistema SCADA permite la supervisión local y remota del proceso que se llevan a cabo en la caldera, mediante una interfaz HMI y una estación de trabajo respectivamente.

Este documento describe el paso a paso de la implementación, configuración y puesta en marcha del SoftPLC V1.0, el cual puede ayudar al crecimiento de las plantas que se encuentran en el laboratorio y convertirlas en plantas con sistemas multiplataforma, a través de la adición de equipos de bajo costo y gran funcionalidad.

## 9. RECOMENDACIONES

El proceso de recirculación de agua del tanque de la caldera (TK-001) al tanque de reserva (TK-002), se ve interrumpido y se debe hacer de forma manual, debido a que la válvula HV-103 no está conectada a TK-002 y esto se debe realizar por medio de un balde color rojo que se encuentra en el laboratorio, se recomienda conectar HV-103 a TK-002 por medio de una manguera y sus respectivos accesorios de esta manera el proceso de recirculación de agua es más cómodo para el estudiante, adicional al realizar esa conexión, a HV-103 se le puede implementar y acondicionar un actuador y así por medio de un control permita la apertura controlada de la válvula para la salida del líquido hacia TK-002.

El cambio del tanque principal de la caldera por uno de material en acero inoxidable, se recomienda para que el proceso de recirculación de agua a TK-002 sea más limpio, ya que el tanque actual es de acero al carbón y el uso al que se está sometiendo TK-001 de almacenar agua a temperaturas altas hace que este produzca óxido y cuando se está expulsando el agua, sale con pequeñas partículas de óxido y el agua se torne de un color naranja, esto no permite que el tanque de reserva almacene por mucho tiempo el líquido ya que produce un olor fuerte y las tuberías también sufren un proceso de oxidación y se pueden ver afectadas en un futuro, en caso de que permita el almacenamiento de agua con óxido es muy importante la limpieza y mantenimiento periódicamente de los tanques, también se recomienda que al finalizar el proceso de generación de vapor, se debe expulsar toda el agua con la ayuda de la presión de aire por medio de un compresor que dispone el laboratorio y no almacenar agua usada o en malas condiciones higiénicas en los tanques, solo se puede almacenar en TK-002 agua limpia o que no adquiera óxido.

Cuando se realiza la salida de vapor de forma manual por el tramo de tubería que expulsa el vapor hacia el techo del laboratorio (Desfogue superior), se recomienda instalar una válvula check después del escape de la válvula de seguridad PV-106, esto con el fin de que el vapor no retorne cuando abra la válvula HV-106.

Finalizadas las prácticas tener en cuenta las presiones residuales en las mangueras (precaución ya que puede quedar agua caliente con presión produciendo expulsión súbita de agua caliente), además se recomienda mantener distancias seguras con respecto a la caldera mientras este en uso debido a las superficies calientes y nunca situarse en frente de los ductos de purga.

## 10. BIBLIOGRAFIA

- [1] C. A. BELTRAN CRUZ and J. G. POVEDA, "Aplicación de un sistema de control en la caldera de proceso de pasteurización didáctica para el laboratorio de industrial de la universidad distrital francisco jose de caldas facultad tecnológica," UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS FACULTAD TECNOLÓGICA, 2016.
- [2] J. Peñalba Galán, "Modelado y Simulación de una Caldera Convencional," 2004.
- [3] J. MORENO, "Modelamiento y control de planta Pasteurizadora," Pontificia Universidad Javeriana Facultad de Ingeniería, 2013.
- [4] C. F. JIMENEZ ACOSTA and J. M. MARTINEZ PAREJA, "Diseño para la instrumentación y control de una variable para la caldera del laboratorio de mecatrónica," UNIVERSIDAD TECNOLÓGICA DE PEREIRA, 2015.
- [5] C. Jingwen and S. Liying, "Direct Fuzzy Backstepping Control for Turbine Main Steam Valve of Multi-machine Power System," in *2015 2nd International Conference on Information Science and Control Engineering*, 2015, pp. 702–706.
- [6] S. Li, Y. Wang, and Z. Zhao, "Performance evaluation of the boiler combustion control system based on data-driven," in *2015 IEEE International Conference on Information and Automation*, 2015, no. 9161714011, pp. 2547–2551.
- [7] Cheng Ling, Qi Ge, Jin Lu, Zhong Ming, and Song Deyu, "Research on control strategy of electric heat storage boiler based on multi-agent," in *2016 IEEE International Conference on Power and Renewable Energy (ICPRE)*, 2016, pp. 508–512.
- [8] P. Keadtipod and D. Banjerdpongchai, "Design of supervisory cascade model predictive control for industrial boilers," in *2016 International Automatic Control Conference (CACCS)*, 2016, pp. 122–125.
- [9] Y. Huang, Y. Cheng, and W. Zhang, "A feed-forward and feedback control scheme for boiler-turbine coordination control system," in *2017 Chinese Automation Congress (CAC)*, 2017, pp. 3860–3863.
- [10] Qingyu Su, Wanzhen Quan, Guowei Cai, and Jian Li, "Improved adaptive backstepping sliding mode control for generator steam valves of non-linear power systems," *IET Control Theory & Appl.*, vol. 11, no. 9, pp. 1414–1419, 2017.
- [11] E. SUÁREZ CHAPARRO and P. N. TORRES CASTILLO, "Diseño e Implementación de un Sistema Scada para el Control PID de Nivel y Temperatura Independientes en Dos Tanques de la Planta de Procesos Análogos (PPA)," Universidad Distrital Francisco Jose de Caldas, 2017.
- [12] N. Nakamura, S. Takahama, L. Barolli, Jianhua Ma, and K. Sugita, "A Multiplatform P2P System: Its Implementation and Applications," in *19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers)*, 2005, vol. 1, pp. 171–176.
- [13] Marin-Perianu *et al.*, "Decentralized enterprise systems: a multiplatform wireless sensor network approach," *IEEE Wirel. Commun.*, vol. 14, no. 6, pp. 57–66, Dec. 2007.

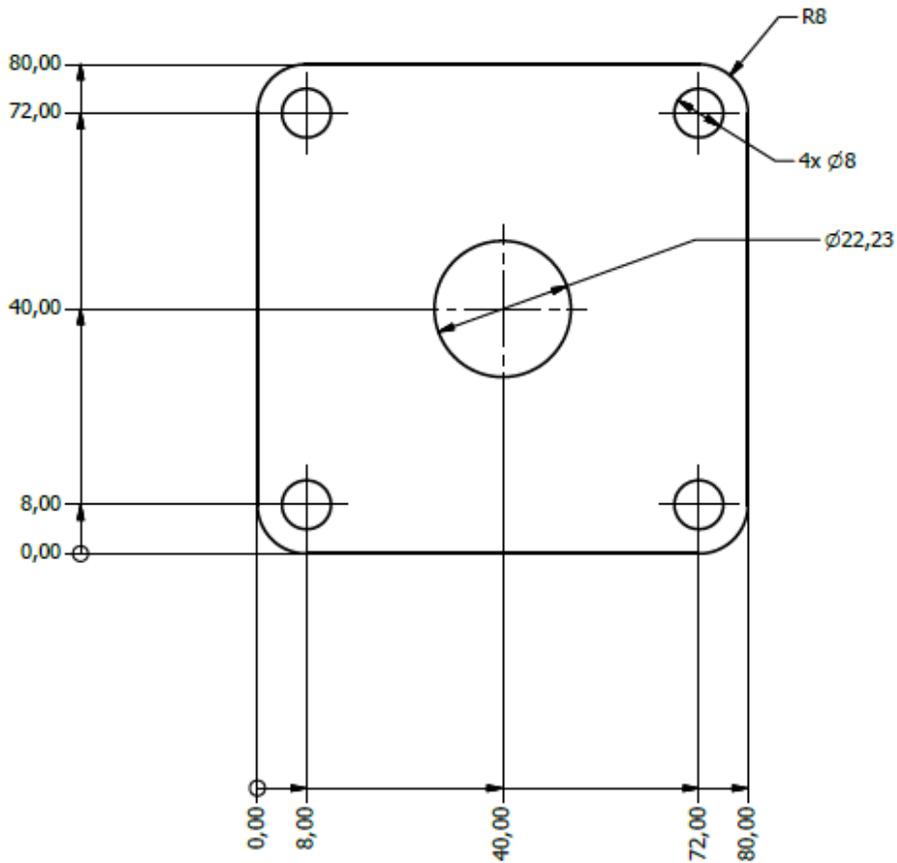
- [14] F. Crispino *et al.*, “Multiplatform and multi-technology system for interpretation and analysis of protection tripping in distribution substations,” in *2010 IEEE/PES Transmission and Distribution Conference and Exposition: Latin America (T&D-LA)*, 2010, pp. 694–699.
- [15] A. Henrique, D. Mendes, L. Cesar, and D. Drumond, “Multiplatform automated system for monitoring and sprinkler irrigation control,” *Rev. Sist. Inf. Da FSMA*, vol. 8, pp. 2–7, 2011.
- [16] R. Bevan, P. Berruet, F. de Lamotte, M. Adam, O. Cardin, and P. Castagna, “Generation of multiplatform control for transitic systems using a component-based approach,” in *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*, 2012, pp. 1–8.
- [17] G. M. Coral, O. A. Rojas, and F. A. Campos, “Sistema de control y supervisión industrial multiplataforma,” Universidad del Cauca, 2012.
- [18] F. Brito, I. Fonseca, J. T. Farinha, and F. M. Barbosa, “Integrating low cost platforms on electrical power systems for control and condition monitoring,” in *2016 51st International Universities Power Engineering Conference (UPEC)*, 2016, pp. 1–5.
- [19] O. Krejcar and P. Konarik, “Low Cost RT Process Control Using Windows PLC by MATLAB/Simulink Throw the REX Control System by Secured Wireless Network,” in *2009 Fourth International Conference on Systems*, 2009, pp. 38–42.
- [20] M. Tutsch, Z. Machacek, O. Krejcar, and P. Konarik, “Development Methods for Low Cost Industrial Control by WinPAC Controller and Measurement Cards in Matlab Simulink,” in *2010 Second International Conference on Computer Engineering and Applications*, 2010, vol. 2, pp. 444–448.
- [21] Q. Liang and L. Li, “The Study of Soft PLC Running System,” *Procedia Eng.*, vol. 15, pp. 1234–1238, 2011.
- [22] H. Prahofe, R. Schatz, C. Wirth, and H. Mossenbock, “A Comprehensive Solution for Deterministic Replay Debugging of SoftPLC Applications,” *IEEE Trans. Ind. Informatics*, vol. 7, no. 4, pp. 641–651, Nov. 2011.
- [23] M. Zhang, Y. Lu, and T. Xia, “The Design and Implementation of Virtual Machine System in Embedded SoftPLC System,” in *2013 International Conference on Computer Sciences and Applications*, 2013, pp. 775–778.
- [24] S. S. Lagu and S. B. Deshmukh, “Raspberry Pi for Automation of Water Treatment Plant,” in *2015 International Conference on Computing Communication Control and Automation*, 2015, vol. 1, pp. 532–536.
- [25] H. Wang, J. Lu, W. Li, and Z. Jiang, “Development of three dimensional virtual PLC experiment model based on Unity3D,” in *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)*, 2017, pp. 1–4.
- [26] A. John, R. Varghese, S. S. Krishnan, S. Thomas, T. A. Swayambu, and P. Thasneem, “Automation of 11 kv substation using raspberry pi,” in *2017 International Conference on Circuit ,Power and Computing Technologies (ICCPCT)*, 2017, pp. 1–5.
- [27] G. M. Martinov, N. V. Kozak, and R. A. Nezhmetdinov, “Implementation of control for peripheral machine equipment based on the external soft PLC

- integrated with CNC,” in *2017 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, 2017, pp. 1–4.
- [28] D. E. Osejo, “Diseño de una caldera de generacion de vapor pirotubular de 10BHP expandible a 25BHP mediante modulos,” *Programa Ing. Mec.*, vol. 91, pp. 399–404, 2017.
- [29] A. Creus Solé, *Intrumentacion Industrial*, 8th Ed. Mexico D.F, 2011.
- [30] Manner, “Agua para calderas,” 2018. [Online]. Available: <http://manner.biz/de-agua-para-calderas.html>. [Accessed: 19-Mar-2018].
- [31] Aki, “Manómetro en rosca para medir presión,” 2018. [Online]. Available: <http://www.aki.es/productos/manometro-en-rosca-para-medir-presion/idp24258>. [Accessed: 19-Mar-2018].
- [32] Combutec, “La tecnología al servicio de los gases de combustión - Combutec,” 2018. [Online]. Available: <http://combustion-tecnologica-del-gas-2010.com/la-tecnologia-al-servicio-los-gases-combustion/>. [Accessed: 19-Mar-2018].
- [33] Estructplan, “Aparatos Sometidos a Presion - Partes Principales que Componen una Caldera,” 2018. [Online]. Available: <http://www.estrucplan.com.ar/Producciones/entrega.asp?IdEntrega=2953>. [Accessed: 19-Mar-2018].
- [34] M. Cedron, “valvulas de seguridad para calderas,” 2018. [Online]. Available: <http://calderasseguridad.blogspot.com.co/>. [Accessed: 19-Mar-2018].
- [35] Pimmsa, “Calderaspimmsa.com.mx,” 2018. [Online]. Available: [http://www.calderaspimmsa.com.mx/c052\\_descargas/c052\\_componentes\\_cp\\_v.pdf](http://www.calderaspimmsa.com.mx/c052_descargas/c052_componentes_cp_v.pdf). [Accessed: 19-Mar-2018].
- [36] A. J. Sanchez, *Control avanzado de procesos*, 3era Ed. Madrid, 2001.
- [37] C. CALDERA, “Control de demanda de una caldera,” *Instrumentation and Control Systems Documentation*, 2018. [Online]. Available: <https://instrumentacionhoy.blogspot.com.co/2015/01/control-de-demanda-en-una-caldera.html>. [Accessed: 19-Mar-2018].
- [38] M. D. Ruiz Pernia and A. Ríos, “Diseño e Implementación de un SoftPLC,” Universidad de los Andes, 2007.
- [39] D. InfoPLC, “Los otros PLCs - Single Board Computer - infoPLC,” *InfoPLC*, 2018. [Online]. Available: <http://www.infoplcn.net/blogs-automatizacion/item/102505-plc-single-board-computer>. [Accessed: 19-Mar-2018].
- [40] D. E. P. Reducida, S. B. C. Raspberry, D. Felipe, G. Segura, M. Angel, and O. Vallejos, “Diseño de un soft-plc basado en un computador de placa reducida (sbc) raspberry,” 2018.
- [41] H. Valves, “Valvulas manuales.”
- [42] E. E. Henao-bravo *et al.*, “Low - Cost Electronic Opening Control Valve,” *Tecciencia*, vol. 12, no. 22, pp. 87–96, 2017.
- [43] I. Automation, “Industrial Automation Software Solutions by Inductive Automation,” 2019. [Online]. Available: <https://inductiveautomation.com/>. [Accessed: 01-Feb-2019].
- [44] U. Manual, “7inch HDMI Display-B,” pp. 5–8, 2018.

## 11. ANEXOS

### 11.1 ANEXO 1 - DISEÑO MECANICO DE CV-105

			39	4	TUERCA - M3,5
19	4	PRISIONERO M5 x 6	38	4	TORNILLO - M3,5 x 5
18	1	ACOPLE MOTOR	37	2	TUERCA- M4
17	1	ACOPLE POTENCIOMETRO	36	2	TORNILLO - M4 x 10
16	2	BARRAS ACERO INOX	35	2	TUERCA- M4 Tipo 5
15	1	DISCO 1	34	2	ARANDELA AS 1237 - 4
14	1	BLUJE ENGRANAJE 2	33	2	TORNILLO M4 x 35
13	1	BLUJE ESPACIADOR	32	1	TUERCA MARIPOSA M6
12	2	BLUJE ENGRANAJE 1	31	1	TORNILLO - M6 x 12
11	1	PLACA ALUMINIO 2	30	1	PASADOR - 2,5 x 16
10	1	BLUJE PLASTICO	29	1	PRISIONERO - M5 x 8
9	1	ARANDELA PLASTICA	28	1	ARANDELA IFI 535 - 20
8	2	EJE ENGRANAJE 2	27	1	DISCO 2
7	1	ENGRANAJE 2	26	2	TUERCA M20x2,5
6	1	ENGRANAJE 1	25	1	Valvula ABAC VA150M
5	1	BLUJE RODAMIENTO	24	1	ASME B18.21.1 - 3/8
4	1	RODAMIENTO HK 0908	23	1	BASE POTENCIOMETRO
3	2	PLACA ALUMINIO	22	1	POTENCIOMETRO
2	4	AS 1111 - M8 x 110	21	2	BASE MOTOR
1	1	PLACA ACERO INOX	20	1	MOTOR
ELEMENTO	CTDAD	Nº DE PIEZA	ELEMENTO	CTDAD	Nº DE PIEZA
Diseño de		Revisado por		Fecha	
GIOVANNY BOLIVAR GUERRERO		GIOVANNY BOLIVAR GUERRERO		30/07/2018	
UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS			MECANISMO DE CONTROL VALVULA DE VAPOR		
			Escala 1 : 2 Hoja 1 / 18		



PLACA DE ACERO INOX CALIBRE 12

Diseño de GIOVANNY BOLIVAR GUERRERO	Revisado por GIOVANNY BOLIVAR GUERRERO	Fecha 30/07/2018	
UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS		MECANISMO DE CONTROL VALVULA DE VAPOR	
PLACA ACERO INOX		Escala 1 : 1	Hoja 2 / 18

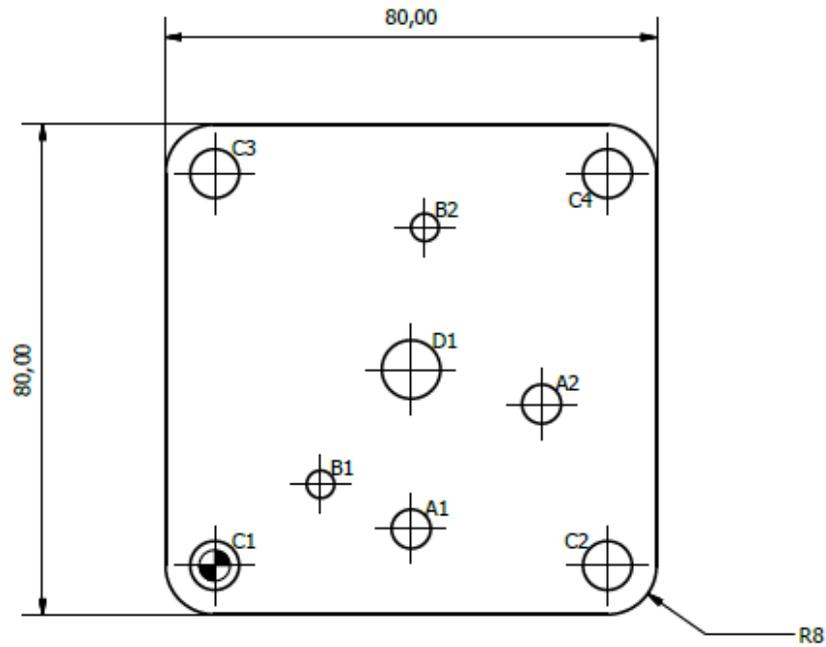
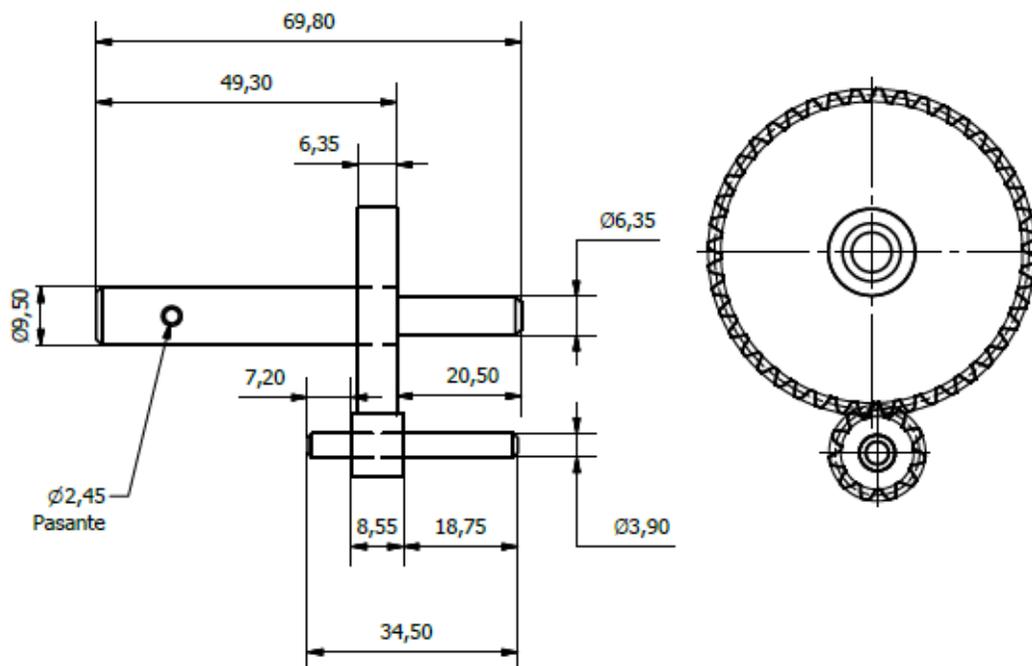


TABLA DE AGUJEROS			
AGUJERO	COTA EN X	COTA EN Y	DESCRIPCIÓN
A1	32,00	5,98	∅6,35 Pasante
A2	53,26	26,36	
B1	17,25	13,25	∅4,50 Pasante
B2	34,25	55,20	
C1	0,00	0,00	∅8,00 - Pasante
C2	64,00	0,00	
C3	0,00	64,00	
C4	64,00	64,00	
D1	32,00	32,00	∅9,55 - Pasante

ESPESOR DE LAMINA 2.5 mm

Diseño de GIOVANNY BOLIVAR GUERRERO	Revisado por GIOVANNY BOLIVAR GUERRERO	Fecha 30/07/2018	
UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS		MECANISMO DE CONTROL VALVULA DE VAPOR	
PLACA ALUMINIO		Escala 1: 1	Hoja 3 / 18



		Engranaje 1	Engranaje 2
Tipo de modelo		Componente	Componente
Número de dientes	z	11 su	44 su
Diámetro de separación	d	8,800 mm	35,200 mm
Diámetro exterior	da	10,528 mm	35,792 mm
Diámetro raíz	df	7,968 mm	32,992 mm
Diámetro de círculo base	db	8,269 mm	33,077 mm
Diámetro de separación de trabajo	dw	8,800 mm	35,200 mm
Anchura de cara	b	8,550 mm	6,350 mm
Altura de cabeza del diente	a*	0,7000 su	0,7500 su
Juego	c*	0,2000 su	0,2500 su
Empalme raíz	rf*	0,3000 su	0,3000 su
Grosor de diente	s	1,478 mm	1,035 mm
Distancia entre centros	D	22 mm	22 mm

Diseño de GIOVANNY BOLIVAR GUERRERO	Revisado por GIOVANNY BOLIVAR GUERRERO	Fecha 30/07/2018	
UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS		MECANISMO DE CONTROL VALVULA DE VAPOR	
ENGRANAJES		Escala 1 : 1	Hoja 18 / 18

## 11.2 ANEXO 2 - ALGORITMO DE CONTROL DE CV-105

```
Control_Valvula.c

#include "project.h"
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#ifndef bool
    #define bool int
    #define false ((bool)0)
    #define true ((bool)1)
#endif

bool ts;
char v[4],valor4[5];
char soft;
int SPOT=0;
int SIN[4];
int PIN[3];
int prom[10]={0,0,0,0,0,0,0,0,0,0};
int prom1[10]={0,0,0,0,0,0,0,0,0,0};
int prom2[10]={0,0,0,0,0,0,0,0,0,0};
int VPOT=0;
int VIN1=0;
int VIN2=0;
int VIN3=0;
int k=0;
int k1=0;
int k2=0;
int F1=0;
int conv=0;
int conv1=0,conv4=0,conv5=0,conv10=0;
int ANGULO=0;
int Ang_order;
int Ang_Act;
int error;
int map;
int vol_4v;
int vol_5v;
int vol_10v;
int KP=100;
int vl;
int entero;

CY_ISR(ADC)
{
    SPOT=ADC_GetResult16(0);//potenciometro motor
    conv=SPOT-1257;
    ANGULO=(int) (-0.000005*(conv*conv)+0.8889*(conv));
    LED_BL_Write(0);
    LED_WH_Write(0);
    if (ANGULO >= 1670){LED_BL_Write(1);}
    if (ANGULO <= 0){LED_WH_Write(1);}
    if(F1==1)
    {
```

```

Control_Valvula.c
SIN[F1]=ADC_GetResult16(F1);//4-20MA
if(k==10){k=0;ADC_StopConvert();}
prom[k]=SIN[F1];
k++;
VIN1=(prom[0] + prom[1]+ prom[2]+ prom[3] + prom[4] + prom[5] +
prom[6] + prom[7] + prom[8] + prom[9])/10;
conv4=(VIN1-830);
vol_4v=0.57*conv4;
if(vol_4v<=0){vol_4v=0;}
Ang_order=vol_4v;
}
if(F1==2)
{
SIN[F1]=ADC_GetResult16(F1);//0-5
if(k1==10){k1=0;ADC_StopConvert();}
prom1[k1]=SIN[F1];
k1++;
VIN2=(prom1[0] + prom1[1]+ prom1[2]+ prom1[3] + prom1[4] + prom1[5]
] + prom1[6] + prom1[7] + prom1[8] + prom1[9])/10;
conv5=0.4812*VIN2;
vol_5v=-0.000005*(conv5*conv5)+0.8889*(conv5);
if(vol_5v<=0){vol_5v=0;}
Ang_order=vol_5v;
}
if(F1==3)
{
SIN[F1]=ADC_GetResult16(F1);// 0-10V
if(k2==10){k2=0;ADC_StopConvert();}
prom2[k2]=SIN[F1];
k2++;
VIN3=(prom2[0] + prom2[1]+ prom2[2]+ prom2[3] + prom2[4] + prom2[5]
] + prom2[6] + prom2[7] + prom2[8] + prom2[9])/10;
conv10=0.4812*VIN3;
vol_10v=-0.000005*(conv10*conv10)+0.8889*(conv10);
if(vol_10v<=0){vol_10v=0;}
Ang_order=vol_10v;
}
if(F1==4)
{
Ang_order=entero;
}
}

CY_ISR(InterruptHandler)
{
Timer_ClearInterrupt(Timer_INTR_MASK_TC);/* Clear the TCPWM terminal
count interrupt */
ts=~ts;
Ang_Act=ANGULO;
error=abs(Ang_order-Ang_Act);
map=(uint16)(KP)*(error);//1.3743
if(map>4095){map=4096;}
}

```

Control\_Valvula.c

```

if(map<=250) {map=0;}
if(Ang_order>=Ang_Act)
{
    PWM_WriteCompare1(000);
    PWM_WriteCompare2(map);
    if(Ang_order==Ang_Act)
    {
        PWM_WriteCompare1(000);
        PWM_WriteCompare2(000);
    }
}
if(Ang_order<=Ang_Act)
{
    PWM_WriteCompare1(map);
    PWM_WriteCompare2(000);
    if(Ang_order==Ang_Act)
    {
        PWM_WriteCompare1(000);
        PWM_WriteCompare2(000);
    }
}
}
int main(void)
{
    CyGlobalIntEnable; // Enable global
interrupts. */
    ADC_Start(); //INICIALIZACION ADC
    ADC_StartConvert();
    ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
    Timer_Start(); //INICIALIZACION TIMER
    Isr_Timer_StartEx(InterruptHandler); //INTERRUPCION TIMER
    isr_1_StartEx(ADC); //INTERRUPCION ADC
    PWM_Start(); //INICIALIZACION PWM
    UART_Start();
    for(;;)
    {
        // PWM_WriteCompare1(000); //cerrar
        // PWM_WriteCompare2(500); //abrir
        PIN[0]=IN1_1_Read();
        PIN[1]=IN2_2_Read();
        PIN[2]=IN3_3_Read();
        /***** SERIAL_RS232 *****/
        if(PIN[0] && PIN[1] && PIN[2])
        {
            soft=UART_UartGetChar();

            if (0 != soft)
            {
                v[v1]=soft;
                v1=v1+1;
                if(v1>=4) {v1=0;}
            }
        }
    }
}

```

Control\_Valvula.c

```
        entero=atoi(v);
        Fl=4;
        sprintf(valor4,"Ser=%d ",entero);
    }
/***** Corriente 4-20 mA *****/
    else if(PIN[0]==1)
    {
        ADC_StartConvert();
        ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
        Fl=1;
    }
/***** Voltaje 0 - 5V *****/
    else if(PIN[1]==1)
    {
        Fl=2;
    }
/***** Voltaje 0 - 10V *****/
    else if(PIN[2]==1)
    {
        Fl=3;
    }
/***** NADA *****/
    else{Fl=0;}
}
}
```

## 11.3 ANEXO 3 - ALGORITMO DE PROGRAMACIÓN DE LA RASPBERRI PI 2B

POU: SPI\_16\_RW

```
1  FUNCTION_BLOCK SPI_16_RW
2  VAR_INPUT
3      DAC_1 : UINT ;
4      DAC_2 : UINT ;
5      DAC_3 : UINT ;
6      DAC_4 : UINT ;
7      GPO : UINT ; //SALIDAS DIGITALES PSOC
8  END_VAR
9  VAR_OUTPUT
10     ADC_1 : UINT ;
11     ADC_2 : UINT ;
12     ADC_3 : UINT ;
13     ADC_4 : UINT ;
14     GPI : UINT ; //ENTRADAS DIGITALES PSOC
15 END_VAR
16 VAR
17     W_Register : UINT := 43690 ; //ABRIR LA COMUNICACION CON EL PSOC
18     i : INT := 0 ;
19     RxdDataHi : UINT ;
20     RxdDataLo : UINT ;
21     dX : ARRAY [ 0 .. 5 ] OF UINT ;
22     TxDataBuffer : ARRAY [ 0 .. 1 ] OF BYTE ;
23     RxDataBuffer : ARRAY [ 0 .. 1 ] OF BYTE ;
24     TxBuffer : POINTER TO ARRAY [ 0 .. 1 ] OF BYTE ;
25     RxBuffer : POINTER TO ARRAY [ 0 .. 1 ] OF BYTE ;
26     RxState : POINTER TO ARRAY [ 0 .. 1 ] OF BYTE ;
27     SPI_State : BOOL ;
28     f1 : UINT ;
29     A1 : ARRAY [ 0 .. 5 ] OF UINT ;
30
31
32
33 END_VAR
34
```

---

```
1  A1 [ 0 ] := W_Register ;
2  A1 [ 1 ] := DAC_1 ;
3  A1 [ 2 ] := DAC_2 ;
4  A1 [ 3 ] := DAC_3 ;
5  A1 [ 4 ] := DAC_4 ;
6  A1 [ 5 ] := GPO ;
7
8  TxDataBuffer [ 0 ] := UINT_TO_BYTE ( SHR ( A1 [ i ] , 8 ) ) ;
9  TxDataBuffer [ 1 ] := UINT_TO_BYTE ( A1 [ i ] AND 255 ) ;
10 TxBuffer := ADR ( TxDataBuffer ) ;
11 SPI_State := SPI_master . transferExt ( pabyTxBuffer := TxBuffer , pabyRxBuffer :=
12 RxBuffer , udiLen := 2 , uiDelayus := 10 , udiSpeedHz := 0 ) ;
13 RxDataBuffer := RxBuffer ^ ;
14 RxdDataHi := BYTE_TO_UINT ( RxDataBuffer [ 0 ] ) ;
```

```
15   RxdDataLo := BYTE_TO_UINT ( RxDataBuffer [ 1 ] );
16   dX [ i ] := SHL ( RxdDataHi , 8 ) OR RxdDataLo ;
17
18   ADC_1 := dX [ 5 ] ;
19   ADC_2 := dX [ 0 ] ;
20   ADC_3 := dX [ 1 ] ;
21   ADC_4 := dX [ 2 ] ;
22   GPI   := dX [ 3 ] ;
23   f1    := dX [ 4 ] ;
24
25   i := i + 1 ;
26   IF ( i = 6 ) THEN
27     i := 0 ;
28   END_IF
29
30
31
32
```

## 11.4 ANEXO 4 – ALGORITMO DE COMUNICACIÓN DEL PSoC 5LP

```

                                                                    SPI_Slave_P5LP.c
#include "project.h"

int32 RxBuffer[5],TxBuffer[5];
int Rxdata,Txdata,F1=0,F2=0,i=0,k=0;
int32 ana[4];
int32 vl=0;
#define ADR_PSOC 43690u

CY_ISR(Rx)
{
    Rxdata=SPIS_1_ReadRxData();
    if(F1==1)
    {
        RxBuffer[i]=Rxdata;
        SPIS_1_WriteTxData(TxBuffer[i]);
        i++;
        if(i==5)
        {
            i=0;
            F1=0;
        }
    }else {SPIS_1_WriteTxData(0);}

    if(Rxdata==ADR_PSOC)
    {
        F1=1;
        F2=1;
    }
    else
    {
        F1=F1;
        F2=F2;
    }
}

CY_ISR(ADC)
{
    /*-----Envio de Datos-----*/
    TxBuffer[0]=ana[0];
    TxBuffer[1]=ana[1];
    TxBuffer[2]=ana[2];
    TxBuffer[3]=ana[3];
    TxBuffer[4]=IN_1_Read();

    /*-----Recepcion de Datos-----*/
    DAC_1_SetValue(RxBuffer[0]);
    DAC_2_SetValue(RxBuffer[1]);
    DAC_3_SetValue(RxBuffer[2]);
    DAC_4_SetValue(RxBuffer[3]);
    OUT_1_Write(RxBuffer[4]);
}
int main(void)
```

SPI\_Slave\_P5LP.c

```

{
    CyGlobalIntEnable;
    ADC_Start();
    AMux_Start();
    DAC_1_Start();
    DAC_2_Start();
    DAC_3_Start();
    DAC_4_Start();
    isr_1_StartEx(Rx);
    isr_2_StartEx(ADC);
    PGA_1_Start();
    PGA_2_Start();
    PGA_3_Start();
    PGA_4_Start();
    SPIS_1_Start();
    RxBuffer[4]=255;

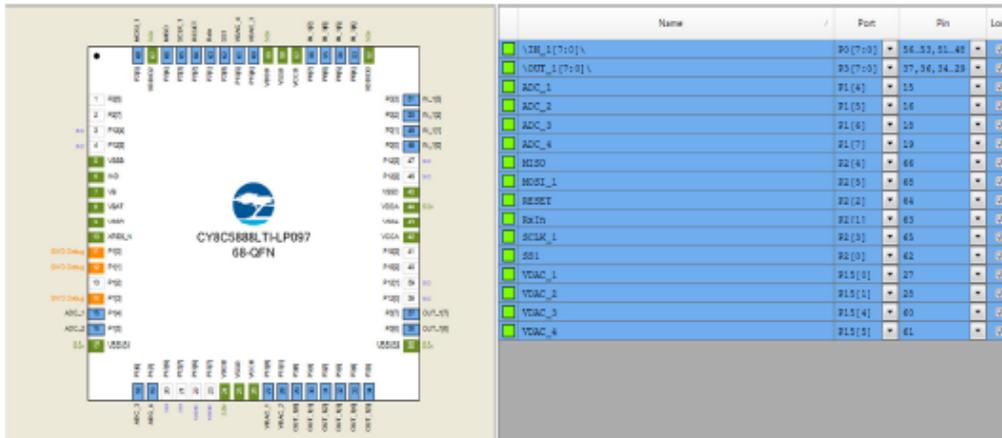
    for(;;)
    {
        AMux_FastSelect(0);
        ADC_StartConvert();
        ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
        ana[0]=ADC_GetResult32();
        ADC_StopConvert();
        if(ana[0]<=0){ana[0]=0;}
        if(ana[0]>=65535){ana[0]=65535;}

        AMux_FastSelect(1);
        ADC_StartConvert();
        ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
        ana[1]=(ADC_GetResult32());
        ADC_StopConvert();
        if(ana[1]<=0){ana[1]=0;}
        if(ana[1]>=65535){ana[1]=65535;}

        AMux_FastSelect(2);
        ADC_StartConvert();
        ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
        ana[2]=(ADC_GetResult32());
        ADC_StopConvert();
        if(ana[2]<=0){ana[2]=0;}
        if(ana[2]>=65535){ana[2]=65535;}

        AMux_FastSelect(3);
        ADC_StartConvert();
        ADC_IsEndConversion(ADC_WAIT_FOR_RESULT);
        ana[3]=ADC_GetResult32();
        ADC_StopConvert();
        if(ana[3]<=0){ana[3]=0;}
        if(ana[3]>=65535){ana[3]=65535;}
    }
}

```



### ANEXO 4.1: asignación de pines:

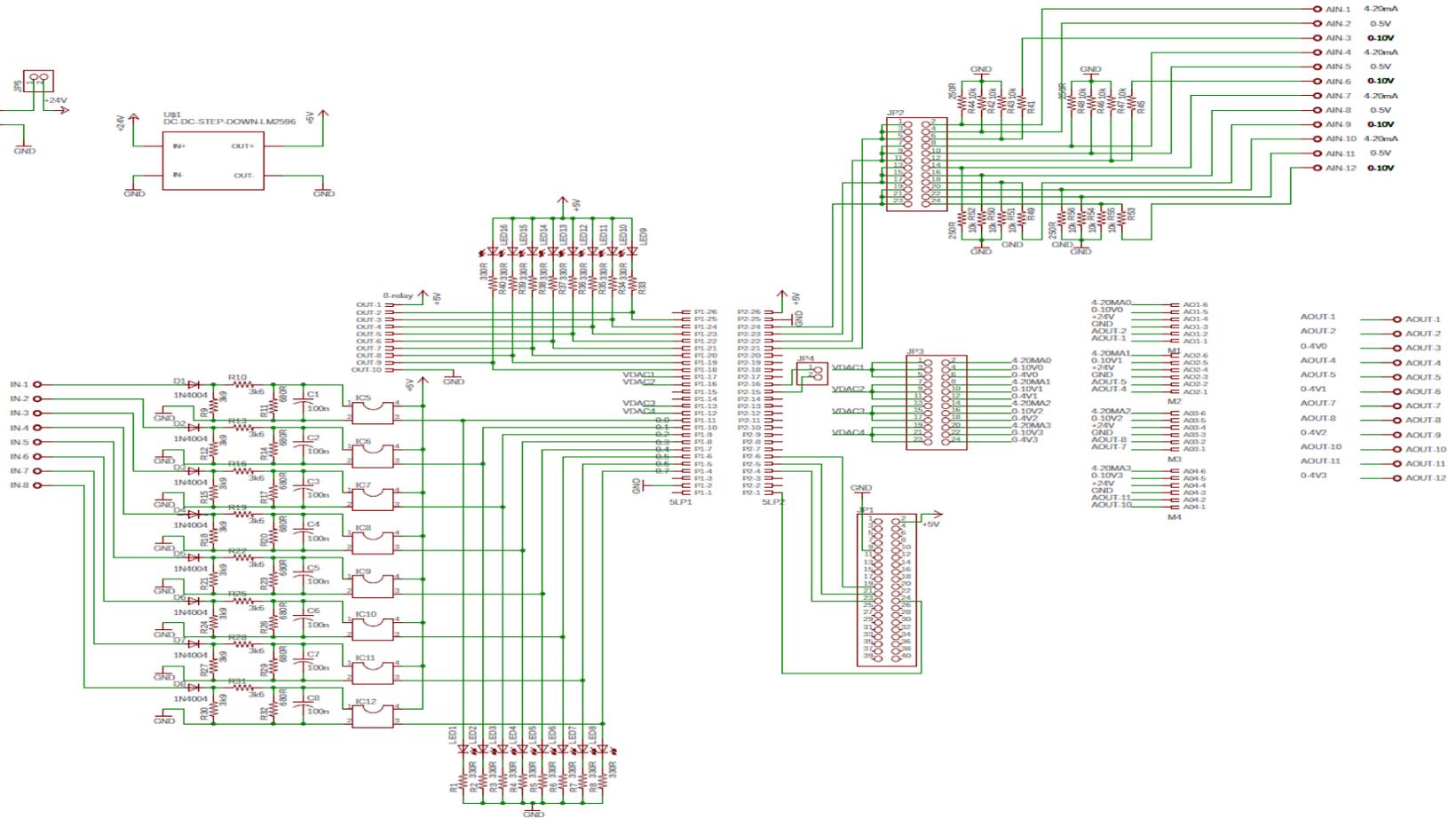
Type /	Name	Domain	Desired Frequency	Nominal Frequency	Accuracy (%)	Tolerance (%)	Divider	Start on Reset	Source Clock
System	XTAL	DIGITAL	24 MHz	7 MHz	±0	-	0	<input type="checkbox"/>	
System	XTAL 32kHz	DIGITAL	32,768 kHz	7 MHz	±0	-	0	<input type="checkbox"/>	
System	Digital Signal	DIGITAL	7 MHz	7 MHz	±0	-	0	<input type="checkbox"/>	
System	USB_CLK	DIGITAL	48 MHz	7 MHz	±0	-	1	<input type="checkbox"/>	IM2x2
System	I/O	DIGITAL	7 MHz	1 MHz	-50, +100	-	0	<input type="checkbox"/>	
System	DMO	DIGITAL	3 MHz	3 MHz	±1	-	0	<input type="checkbox"/>	
System	PULL_OUT	DIGITAL	24 MHz	24 MHz	±1	-	0	<input type="checkbox"/>	IMO
System	MASTER_CLK	DIGITAL	7 MHz	24 MHz	±1	-	1	<input type="checkbox"/>	PULL_OUT
System	BUS_CLK (CPU)	DIGITAL	7 MHz	24 MHz	±1	-	1	<input type="checkbox"/>	MASTER_CLK
Local	SPB_1_HzClock	DIGITAL	1.00 kHz	1.00 kHz	±1	±5	240	<input checked="" type="checkbox"/>	Aster MASTER_CLK
Local	Clock_1	DIGITAL	250 kHz	250 kHz	±1	-	96	<input checked="" type="checkbox"/>	Aster MASTER_CLK
Local	ADC_4heCLK	ANALOG	960 kHz	960 kHz	±1	-	25	<input checked="" type="checkbox"/>	MASTER_CLK
Local	ADC_Ext_CP_Clk	DIGITAL	7 MHz	24 MHz	±1	-	1	<input checked="" type="checkbox"/>	MASTER_CLK
Local	DAC_1_BUS_CLK	DIGITAL	7 MHz	24 MHz	±1	-	0	<input checked="" type="checkbox"/>	BUS_CLK
Local	DAC_4_BUS_CLK	DIGITAL	7 MHz	24 MHz	±1	-	0	<input checked="" type="checkbox"/>	BUS_CLK
Local	DAC_3_BUS_CLK	DIGITAL	7 MHz	24 MHz	±1	-	0	<input checked="" type="checkbox"/>	BUS_CLK
Local	DAC_2_BUS_CLK	DIGITAL	7 MHz	24 MHz	±1	-	0	<input checked="" type="checkbox"/>	BUS_CLK

### ANEXO 4.2: Relojes del sistema.

Option	Value
Configuration	
Device Configuration Mode	Compressed
Enable Error Correcting Code (ECC)	<input type="checkbox"/>
Enable Fast DMO During Startup	<input checked="" type="checkbox"/>
Heap Size (bytes)	5x80
Include CMSIS Core Peripheral Library Files	<input checked="" type="checkbox"/>
Instruction Cache Enabled	<input checked="" type="checkbox"/>
Stack Size (bytes)	5x800
Store Configuration Data in ECC Memory	<input checked="" type="checkbox"/>
Unused Bonded ID	Allow but warn
Operating Conditions	
Temperature Range	-40C - 85/125C
Variable VDDA	<input type="checkbox"/>
VDDA (V)	5.0
VDDD (V)	5.0
VDDIO0 (V)	5.0
VDDIO1 (V)	5.0
VDDIO2 (V)	5.0
VDDIO3 (V)	5.0

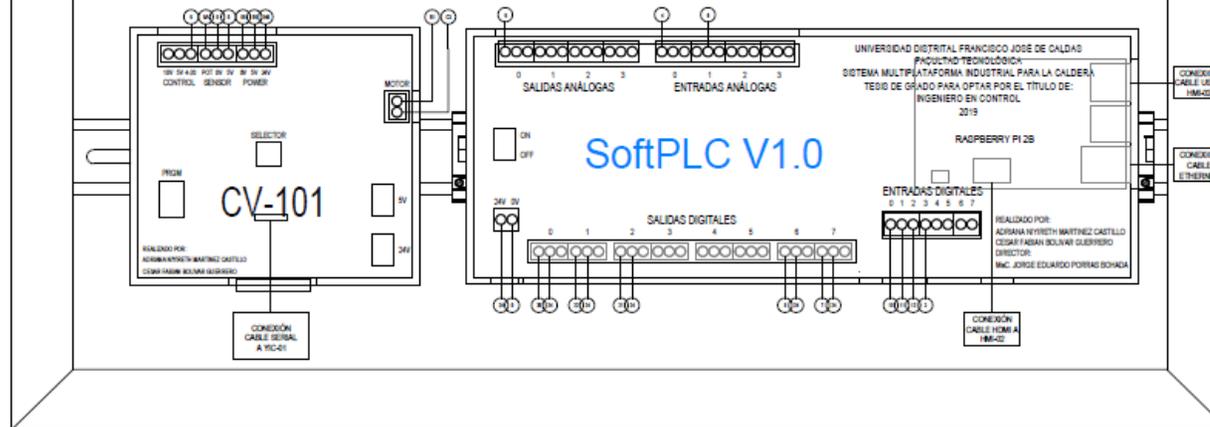
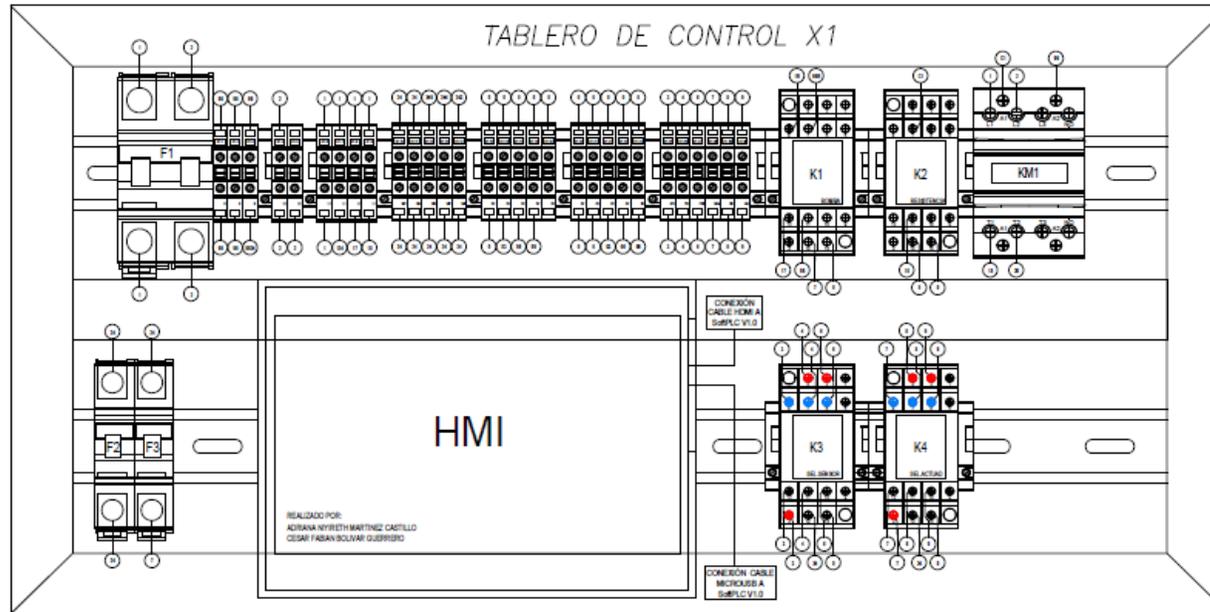
### ANEXO 4.3: Configuración del sistema.

# 11.5 ANEXO 5 – DIAGRAMA ESQUEMATICO DEL SoftPLC V1.

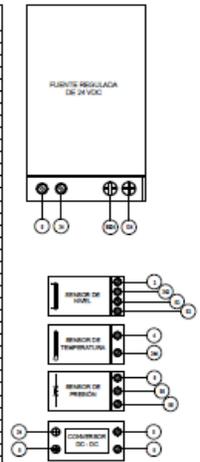


# 11.6 ANEXO 6 – DIAGRAMA DE TABLERO DE BORNES Y CONTROL X1 Y X2

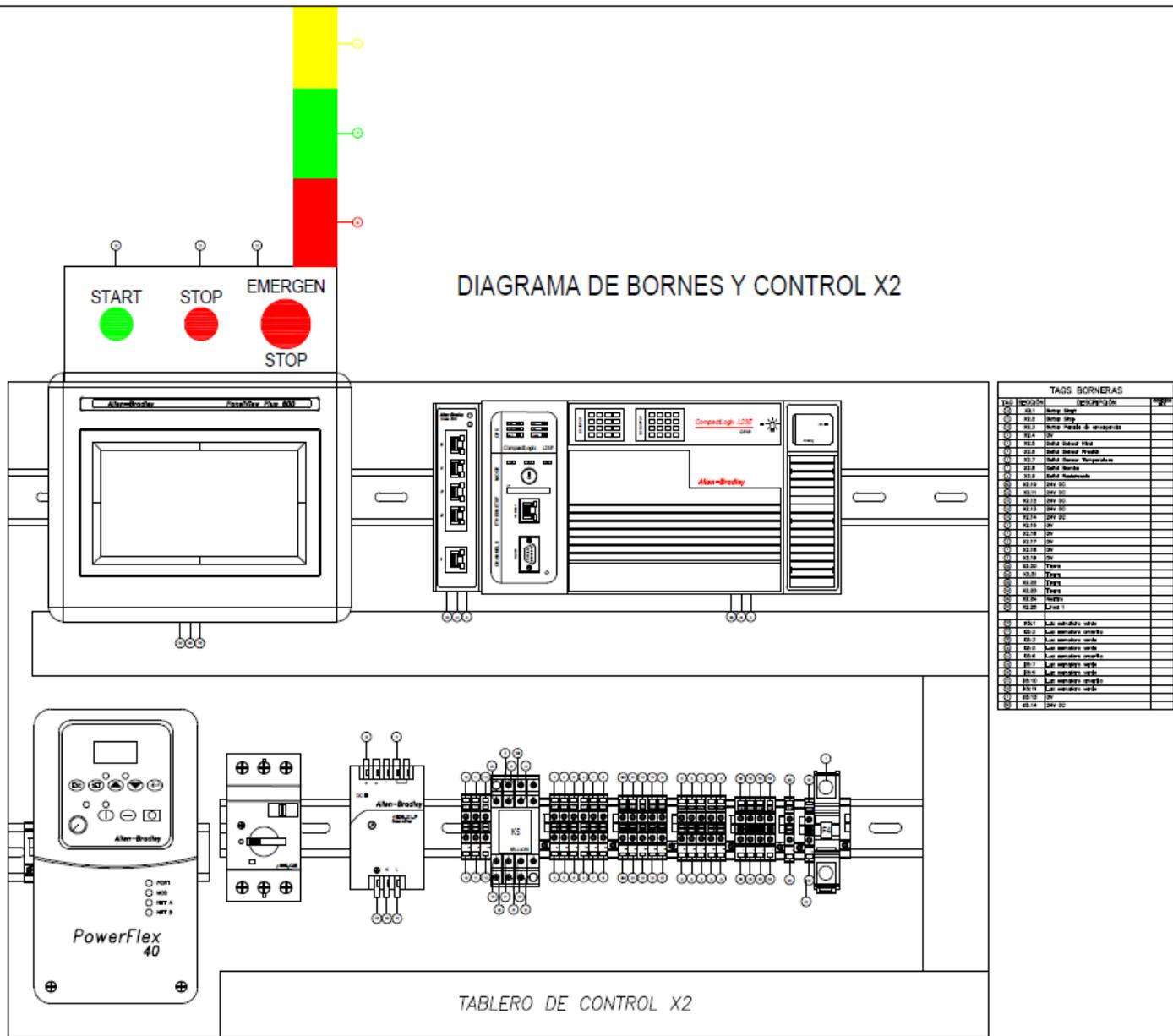
## DIAGRAMA DE BORNES Y POTENCIA X1



TAGS BORNERAS		COMENTARIO
TAG	DESCRIPCIÓN	
K1.1	Motor-Bomba K1 Bomba	
K1.2	Motor-Controlador K1	
K1.3	Motor-Fuente 24V	
K1.4	Línea 2-Abre/cierra Bombas 2500W	
K1.5	Línea 2-Reserva Línea	
K1.6	Línea 1-Reserva P1	
K1.7	Línea 1-Abre/cierra Fuente 24V	
K1.8	Línea 1-Abre/cierra Bomba	
K1.9	Línea 1-Abre/cierra Resistencia 2500W	
K1.10	24V DC-Índice de Fuente 24V	
K1.11	24V DC-Abre/cierra Raspberry	
K1.12	24V DC-Abre/cierra Válvula	
K1.13	24V DC-Abre/cierra Sensor Temp	
K1.14	24V DC-Abre/cierra Sensor Nivel	
K1.15	24V DC-Índice de Fuente 24V	
K1.16	24V DC-Abre/cierra Sensor Nivel	
K1.17	24V DC-Abre/cierra Sensor Presión	
K1.18	24V DC-Índice de Línea	
K1.19	24V-Índice de Fuente 24V	
K1.20	24V-Índice de Fuente 24V	
K1.21	24V-Raspberry	
K1.22	24V-Sensor Nivel	
K1.23	24V-Sensor Presión	
K1.24	24V-Índice	
K1.25	Saldador Sensor Nivel	
K1.26	Saldador Sensor Temperatura	
K1.27	Saldador Sensor Presión	
K1.28	Saldador Sensor Nivel	
K1.29	Saldador Resistencia	
K1.30	Saldador Válvula	
K1.7	Motor Bomba	
K1.8	Línea 1 Bomba	
K1.11	Motor	
K1.12	Línea 1-Abre/cierra Bomba	
K1.13	24V	
K1.14	Saldador Bomba	
K1.7	Abre/cierra Indica Al de K1	
K1.11	Línea 1-Abre/cierra Resistencia	
K1.13	24V	
K1.14	Saldador Resistencia	
K1.15	Línea 1	
K1.16	Línea 2	
K1.17	Abre/cierra Bomba Al de K1	
K1.20	Motor	
K1.21	Línea 1-Abre/cierra Resistencia 2500W	
K1.22	Línea 2-Abre/cierra Resistencia 2500W	
K1.23	Saldador Sensor Presión-CompactLogix L33E V1.0	
K1.24	Saldador Sensor Temperatura-CompactLogix L33E V1.0	
K1.25	Saldador Sensor Nivel-CompactLogix L33E V1.0	
K1.26	Saldador Sensor Presión-SurPLC	40
K1.27	Saldador Sensor Temp-SurPLC	41
K1.28	Saldador Sensor Nivel-SurPLC	42
K1.29	Saldador Sensor Presión	43
K1.30	Saldador Sensor Temperatura	
K1.31	Saldador Sensor Nivel	
K1.32	24V DC	
K1.33	24V DC	
K1.34	24V DC	
K1.2	Saldador Válvula-CompactLogix L33E V1.0	44
K1.3	Saldador Resistencia-CompactLogix L33E V1.0	45
K1.4	Saldador Bomba-CompactLogix L33E V1.0	46
K1.6	Saldador Válvula-CompactLogix L33E V1.0	47
K1.7	Saldador Resistencia-SurPLC	48
K1.8	Saldador Bomba-SurPLC	49
K1.10	Saldador Válvula	
K1.11	Saldador Resistencia	
K1.12	Saldador Bomba	
K1.13	24V	
K1.14	24V DC	
K1	24V-Controlador válvula	
K1	24V-Controlador válvula	
K1	24V-Controlador válvula	
K1	Saldador 1. alto del motor DC	
K1	Saldador 2. alto del motor DC	
K1	Luz sensorio verde	
K1	Luz sensorio amarillo	
K1	Luz sensorio verde	
K1	Boton Stop	
K1	Boton Shop	
K1	Boton Parada de emergencia	

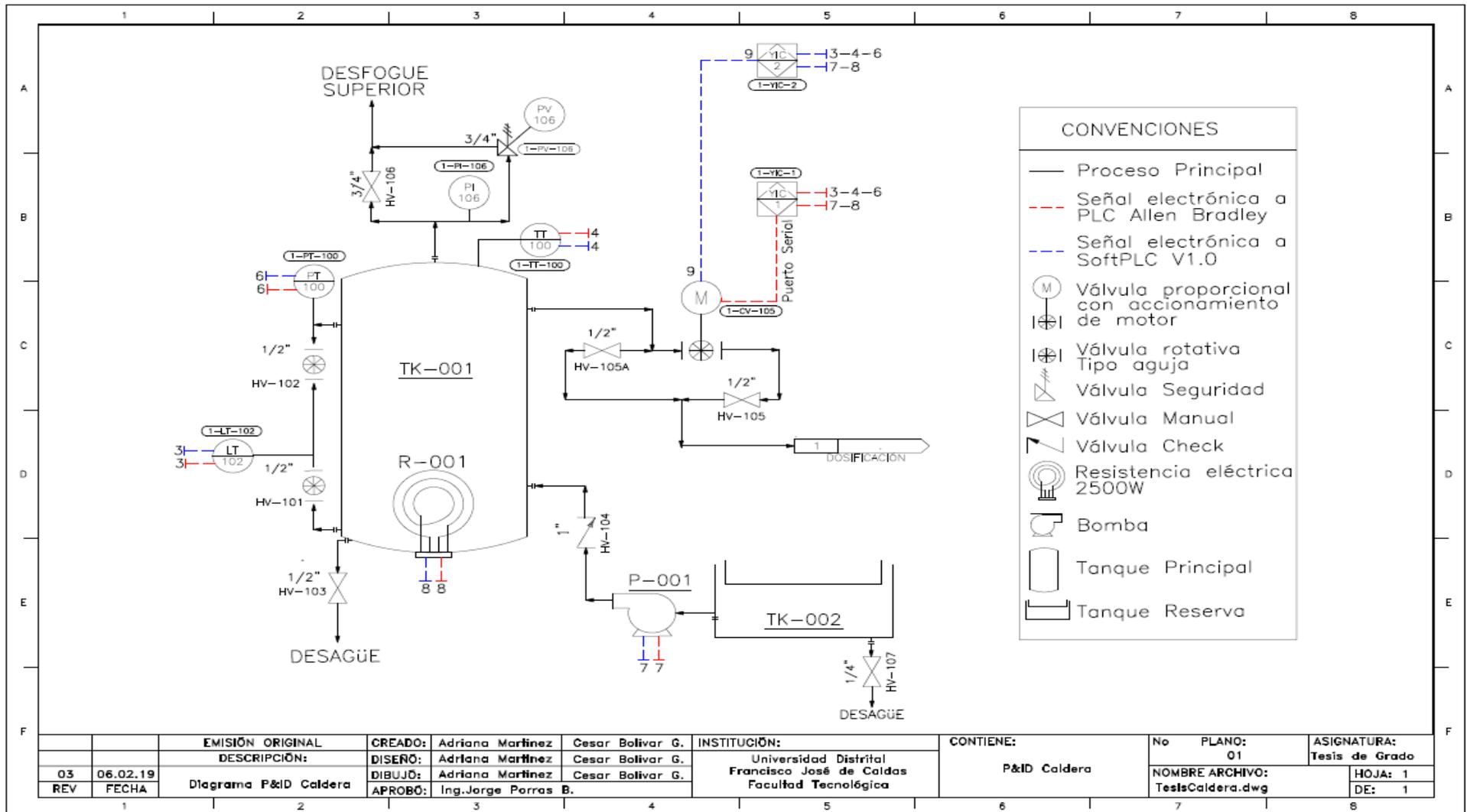


<p>INSTITUCIÓN: <b>UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS</b> FACULTAD DE INGENIERÍA CL 684 De Alcazar #45F - 70 Bogotá D.C. - Colombia</p>	DISEÑO: Adriana Nyreth Martinez Castillo	PROYECTO CURRICULAR: Ingeniería en Control	DESCRIPCIÓN: DIAGRAMA DE CONEXIÓN DE BORNERAS Y POTENCIA TABLERO DE CONTROL.	CONVENCIONES: BORNERA BORNAMIENTO RIELE 14 PINES CONTACTOR	DISEÑADOR DOBLE MARQUILLAS	ASIGNATURA: TESIS DE GRADO	
	DIBUJO: Adriana Nyreth Martinez Castillo	COLABORADORES: Cesar Fabian Bolivar Guerrero	FECHA: Febrero 28 de 2019	REVISIÓN: 	ESCALA: 1:50	PLANO N°: 1	HOJA: 1
	APROBADO: Ing. Jorge Eduardo Fornes Bohica	OBSERVACIONES: 	ARCHIVO: DIAGRAMA ELECTRO TABLERO CALDERA.dwg				



 <p>INSTITUCIÓN: <b>UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS</b> FACULTAD TECNOLÓGICA Cl. 684 de Alier #49F - 70 Bogotá D.C. - Colombia</p>	<p>DESIGNO: Adriana Nyarah Martínez Castillo</p>	<p>PROYECTO CURRICULAR: Ingeniería en Control</p>	<p>DESCRIPCIÓN: DIAGRAMA DE CONEXIÓN DE BORNERAS Y POTENCIA TABLERO DE CONTROL.</p>	<p>CONVENCIONES:   BORNERA   BORNERA DOBLE   BORNERA PIN   RELE 14 PINES   CONTACTOR   BORNERA DOBLE   BORNERA PIN         </p>	<p>ASIGNATURA: TESES DE GRADO</p>
	<p>DEBUCO: Adriana Nyarah Martínez Castillo</p>	<p>COLABORADORES: César Fabian Solís Guerrero</p>	<p>FECHA: Febrero 06 de 2019</p>	<p>REVISIÓN:</p>	<p>ESCALA: 1:80    PLANO Nº: 2    HOJA: 2 DE: 3</p>
<p>APROBADO: Ing. Jorge Eduardo Torres Bohade</p>	<p>OBSERVACIONES:</p>	<p>ARCHIVO: DIAGRAMA ELECTRICO TABLERO CALDERA.dwg</p>			

### 11.7 ANEXO 7- DIAGRAMA P&ID FINAL DE LA CALDERA



11.8 ANEXO 8- PRÁCTICA DE CONTROL BÁSICO DE LA CALDERA CON EL SOFTPLC V1.

	<p><b>UNIDAD TEMÁTICA:</b> CONTROL BÁSICO DE LA CALDERA CON EL SOFTPLC V 1.0</p>	
	<p><b>ACTIVIDAD:</b> MANEJO DE E/S ANÁLOGAS Y DIGITALES.</p>	
		
<p>Código: PC001-A</p>	<p><input checked="" type="checkbox"/> ONLINE <input type="checkbox"/> OFFLINE</p>	<p>Duración: 2 h</p>

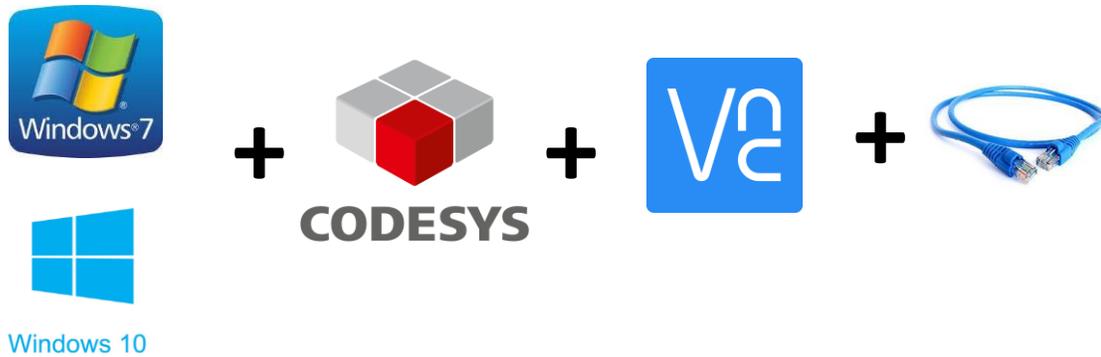
**Objetivo de la Práctica:**

Realizar los pasos descritos para el control y supervisión de la caldera mediante el SoftPLC V1.0.

**Material necesario y requisitos para el desarrollo de la Actividad Práctica:**

- PC con Windows 7/10
- Software CODESYS V3.5 SP14.
- VNC Viewer 6.17.
- Patch Cord (Cable de Red).

**Esquema Grafico de la Actividad Práctica.**



**Figura 1.1:** Grafico de los requerimientos para la práctica.

**Requerimientos previos.**

Para la realización de esta práctica es necesario haber identificado los componentes de la caldera en el diagrama P&ID y las características técnicas de los mismos; A demás de instalar en un PC el software de programación CODESYS V3.5 SP14 y el VNC Viewer.

## I. COMPILACIÓN DEL CÓDIGO BASE EN EL SoftPLC V1.0

- A. Conectar un extremo del Patch Cord al Face Plate del puesto de trabajo y el otro al puerto Ethernet del PC de trabajo.
- B. Configurar la dirección IP en el PC de la siguiente, ver figura 1.2:
  1. Ir a la ruta: panel de control>>Todos los elementos de panel de control>> Centro de redes y recursos compartidos
  2. Clic en conexión de área local.
  3. Clic en propiedades.
  4. Doble clic en Protocolo de internet versión 4 (TCP/IP).
  5. Por ultimo colocar la dirección IP del puesto de trabajo con su respectiva máscara.

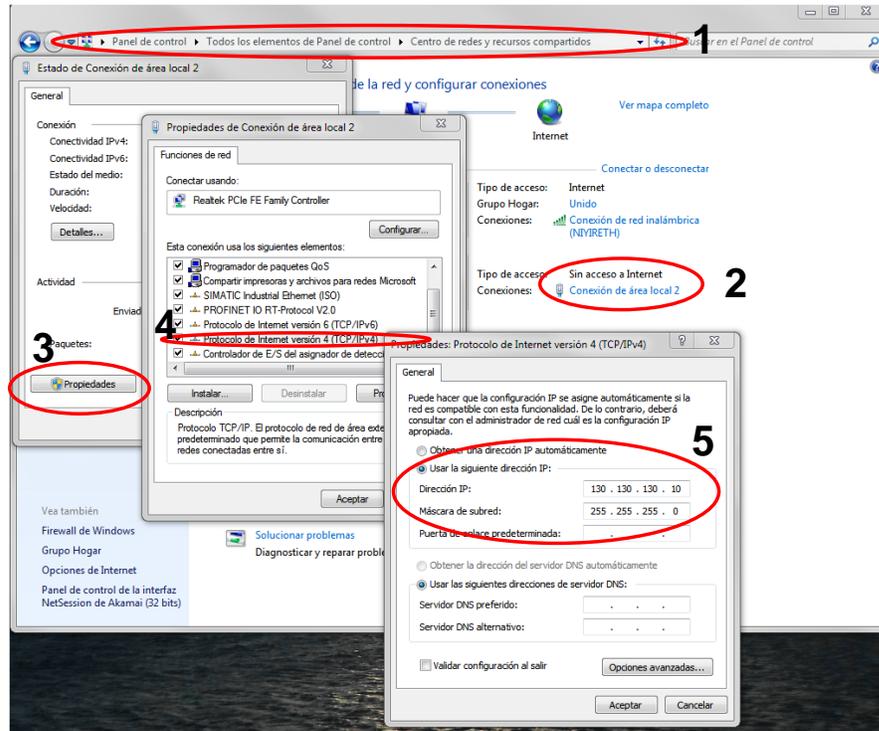


Figura 1.2: Configuración de la IP del PC.

- C. Abrir CODESYS V3.5 SP14, luego dar nuevo proyecto  y aparece la figura 1.3:
  1. Escoger Proyecto estándar
  2. Dar un nombre y ubicación para ser localizado.

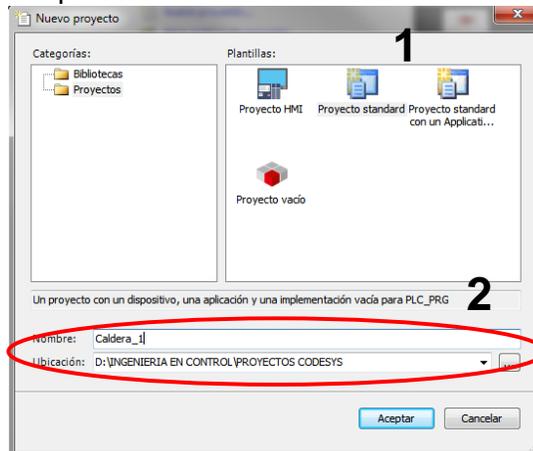
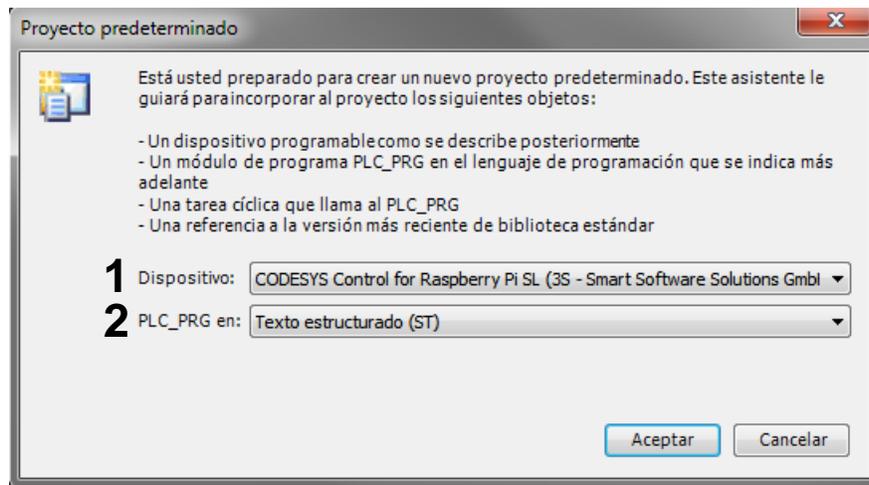


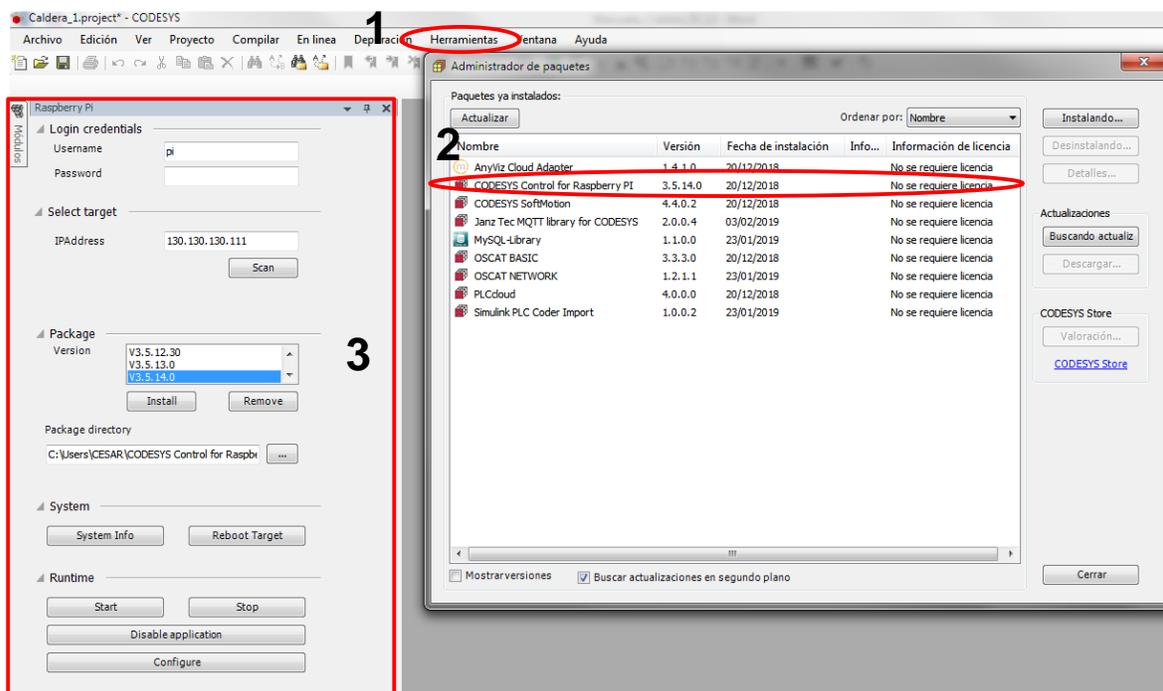
Figura 1.3: Configuración del nuevo proyecto.

- D. La siguiente ventana en aparecer se muestra en la figura 1.4.
1. Se debe escoger dispositivo que aparece en la figura 1.4.
  2. Elegir el lenguaje del programa principal que se desee.



**Figura 1.4:** Dispositivo y lenguaje de programación.

- E. Al estar abierta la plataforma de programación se debe tener instalado el paquete para controlar la SoftPLC V1.0, ver figura 1.5:
1. Hacer Clic en herramientas.
  2. Abrir Administrador de paquetes, y tener instalado o instalar CODESYS Control for Raspberry Pi versión 3.5.14.0
  3. Aparecerá la ventana de configuración para la SoftPLC V1.0, donde el **Password es caldera**, se escanea la IP: 130.130.130.111 y por último se da clic en Start para iniciar el Runtime.



**Figura 1.5:** Paquete de Control For Raspberry PI 3.5.14.0.

- F. En el árbol de procesos ubicado a la parte izquierda de la plataforma de programación, ver figura 1.6 y figura 1.7:
1. El ítem de GPIO, dar clic derecho
  2. Abrir la Opción de 'Actualizar dispositivo', escoger GPIO Pi2.
  3. Dar clic en actualizar y luego cerrar.

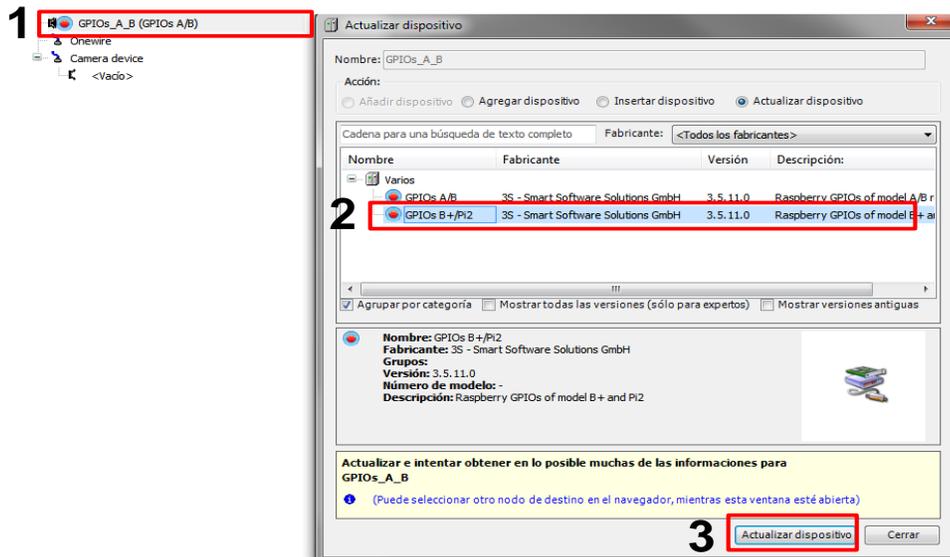


Figura 1.6: Actualización de GPIO SoftPLC V1.0.

1. El ítem SPI dar clic derecho.
2. Abrir la opción de 'Agregar dispositivo', escoger SPI\_Master.
3. Clic en Agregar Dispositivo.

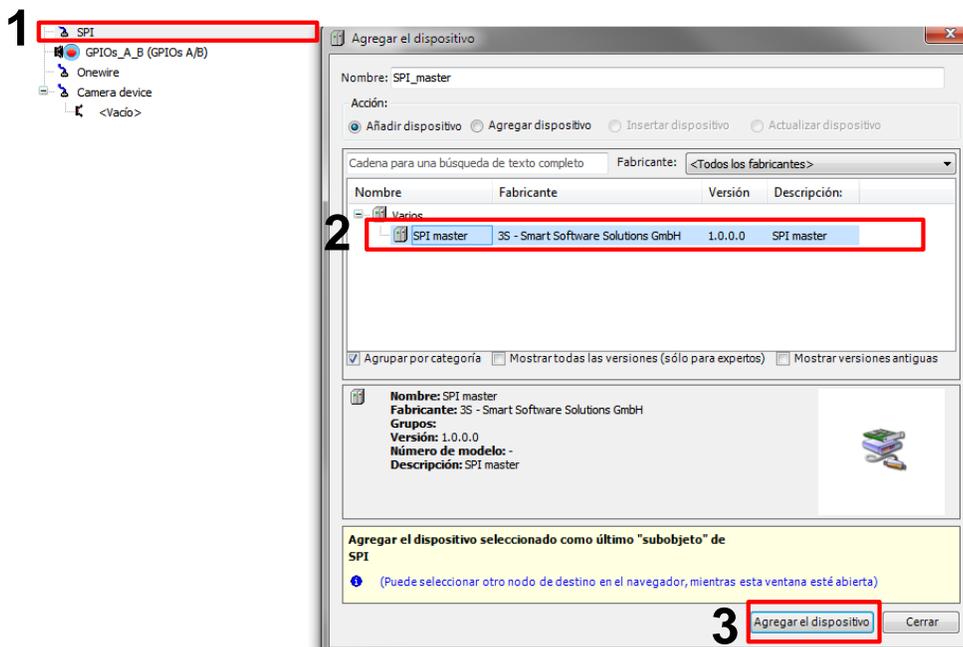


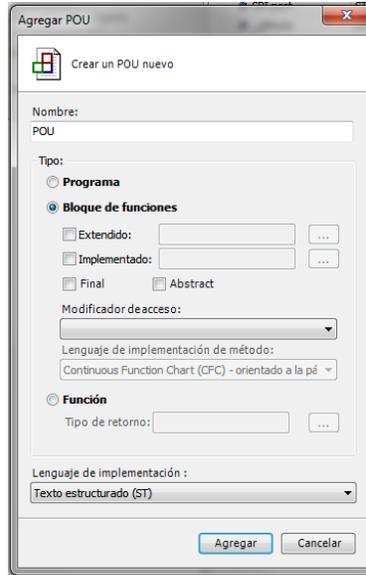
Figura 1.7: SPI Master Adición.

G. Establecer los siguientes parámetros en SPI\_Master al dar doble clic encima, ver figura 1.8:

Parámetro	Tipo	Valor	Valor predeterminado	Unidad	Descripción
SPI port	STRING	'/dev/spidev0.0'	'/dev/spidev0.0'		file path of the SPI port
_diMode	DINT	1	0		SPI mode
_diBitsPerWord	DINT	8	0		SPI bits per word
_diMaxSpeed	DINT	100000	0		SPI max speed setting [Hz]

Figura 1.8: SPI Master Configuración.

- H. Crear un bloque de funciones (FB) llamado SPI\_16\_RW, con clic derecho sobre Application>>Agregar Objeto>>POU y aparece la figura 1.9:



**Figura 1.9:** Configuración del POU como bloque de función.

- I. Copiar y pegar el programa del Bloque de Funciones SPI\_16\_RW
1. Declaración de las variables en el Bloque de Función SPI\_16\_RW

```

FUNCTION_BLOCK SPI_16_RW
VAR_INPUT
    DAC_1:UINT;
    DAC_2:UINT;
    DAC_3:UINT;
    DAC_4:UINT;
    GPO:UINT;          //SALIDAS DIGITALES PSOC
END_VAR
VAR_OUTPUT
    ADC_1:UINT;
    ADC_2:UINT;
    ADC_3:UINT;
    ADC_4:UINT;
    GPI:UINT;          //ENTRADAS DIGITALES PSOC
END_VAR
VAR
    W_Register:UINT:=43690; //ABRIR LA COMUNICACION CON EL PSOC
    i:INT:=0;
    RxdataHi:UINT;
    RxdataLo:UINT;
    dX:ARRAY[0..5] OF UINT;
    TxDataBuffer:ARRAY[0..1] OF BYTE;
    RxDataBuffer:ARRAY[0..1] OF BYTE;
    TxBuffer:POINTER TO ARRAY[0..1] OF BYTE;
    RxBuffer:POINTER TO ARRAY[0..1] OF BYTE;
    RxState:POINTER TO ARRAY[0..1] OF BYTE;
    SPI_State:BOOL;
    f1:UINT;
    A1: ARRAY[0..5] OF UINT;
END_VAR

```

## 2. Programa para el funcionamiento de la comunicación SPI.

```
A1[0]:=W_Register;
A1[1]:=DAC_1;
A1[2]:=DAC_2;
A1[3]:=DAC_3;
A1[4]:=DAC_4;
A1[5]:=GPO;

TxDataBuffer[0]:=UINT_TO_BYTE(SHR(A1[i],8));
TxDataBuffer[1]:=UINT_TO_BYTE(A1[i] AND 255);
TxBuffer:=ADR(TxDataBuffer);
SPI_State:=SPI_master.transferExt(pabyTxBuffer:=TxBuffer,pabyRxBuffer:=RxBuffer,udiLen:=2,uiDelayus:=10,udiSpeedHz:=0);
RxBuffer:=ADR(RxState);
RxDataBuffer:=RxBuffer^;
RxdataHi:=BYTE_TO_UINT(RxDataBuffer[0]);
RxdataLo:=BYTE_TO_UINT(RxDataBuffer[1]);
dX[i]:=SHL(RxdataHi,8) OR RxdataLo;

ADC_1:=dX[5];
ADC_2:=dX[0];
ADC_3:=dX[1];
ADC_4:=dX[2];
GPI :=dX[3];
f1 :=dX[4];

i:=i+1;
IF(i=6)THEN
    i:=0;
END_IF
```

J. En el programa principal o en otro programa, hacer el llamado del Bloque de Funciones SPI\_16\_RW declararlo con un nombre y declarar las variables del programa con el nombre que desee como se ve en

1. Declarar el bloque de funciones y las variables del bloque en el programa.

```
PROGRAM PLC_PRG
VAR
SPI_SLAVE:SPI_16_RW;
OUT_AN0:UINT; //SALIDA ANÁLOGA CANAL 0
OUT_AN1:UINT; //SALIDA ANÁLOGA CANAL 1
OUT_AN2:UINT; //SALIDA ANÁLOGA CANAL 2
OUT_AN3:UINT; //SALIDA ANÁLOGA CANAL 3
OUT_DIG:UINT; //SALIDAS DIGITALES [0.....7]
INP_AN0:UINT; //ENTRADA ANÁLOGA CANAL 0
INP_AN1:UINT; //ENTRADA ANÁLOGA CANAL 1
INP_AN2:UINT; //ENTRADA ANÁLOGA CANAL 2
INP_AN3:UINT; //ENTRADA ANÁLOGA CANAL 3
INP_DIG:UINT; //ENTRADAS DIGITALES [0.....7]
END_VAR
```

2. Llamar el bloque y asignar las variables para su lectura y escritura (ver Comentarios).

```
SPI_SLAVE(
DAC_1:= OUT_AN0, //SEÑAL DE CONTROL VÁLVULA 4-20MA
DAC_2:= OUT_AN1, //SIN USO
DAC_3:= OUT_AN2, //SIN USO
DAC_4:= OUT_AN3, //SIN USO
GPO:= OUT_DIG, //SALIDAS DIGITALES
ADC_1=> INP_AN0, //SEÑAL DEL TRANSMISOR DE TEMPERATURA
ADC_2=> INP_AN1, //SEÑAL DEL SENSOR DE PRESIÓN
ADC_3=> INP_AN2, //SIN USO
ADC_4=> INP_AN3, //SIN USO
GPI=> INP_DIG); //ENTRADAS DIGITALES.
```

K. El siguiente paso es crear una variable llamada RST(reset) como se ve a continuación  
**RST : BOOL := TRUE; //RESET COMUNICACIÓN**

L. para el uso de la variable RST se debe configurar el GPIO de la SoftPLC V1.0, como se ve en la figura 1.10 y 1.11.

1. En el GPIO buscar el GPIO 27 y habilitar como salida.

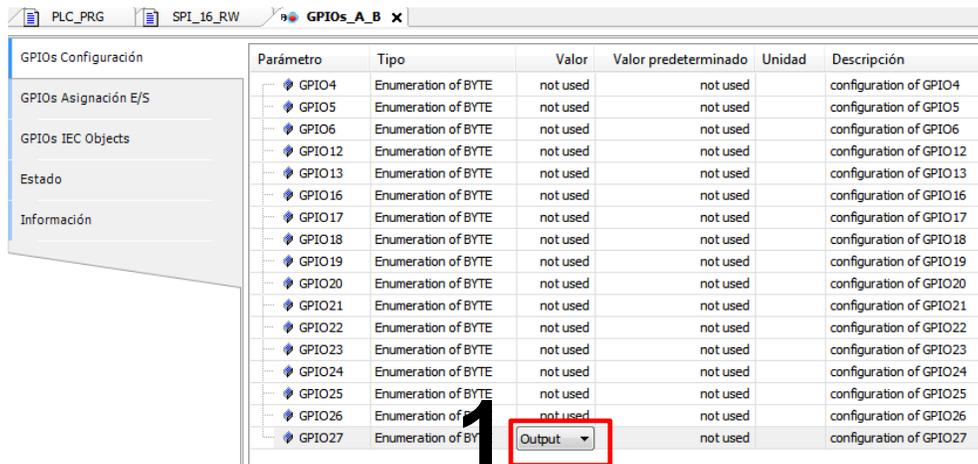


Figura 1.10: Configuración GPIO 27 como salida.

2. Anotar la dirección de salida del GPIO 27 que se muestra en la figura 1.11.

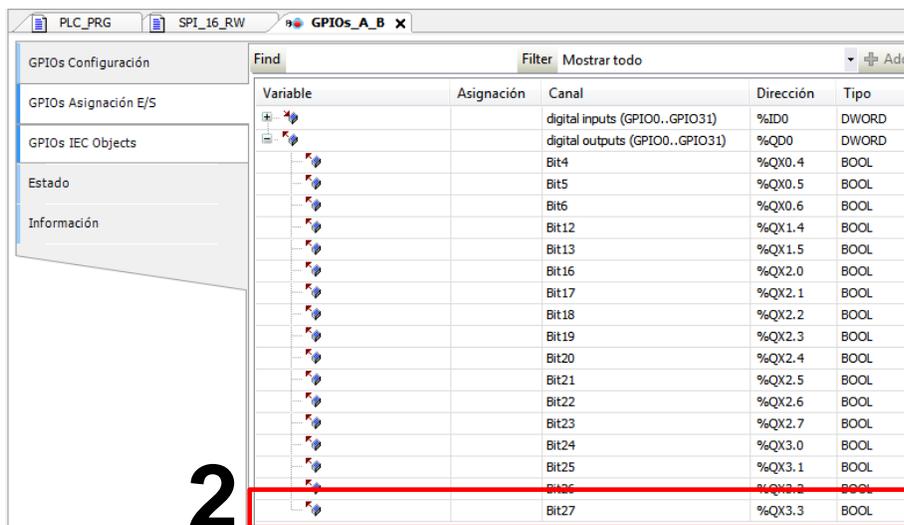


Figura 1.11: Dirección del GPIO 27.

- M. Se debe asignar a la dirección del GPIO 27 la variable RST creada anteriormente, de la siguiente manera:  
**%QX3.3 := RST;**

**NOTA 1:** la asignación de esta variable en esta dirección es necesaria ya que el Control de CODESYS para la SoftPLC V1.0 no está licenciado y la ejecución de la licencia solo dura 2 horas y luego de eso se debe apagar la SoftPLC V1.0 y volverla a encender, la comunicación SPI con el PSoC 5LP puede caer desincronización por lo tanto RST debe funcionar como pulsador para que establezca de nuevo la comunicación con el módulo expansor E/S análogas y digitales.

**NOTA 2:** la dirección del GPIO 27 puede cambiar de acuerdo al uso de los buses de campo que se puedan implementar en el SoftPLC V1.0.

## II. CONFIGURACIÓN DE LAS ENTRADAS/ SALIDAS ANALÓGICAS Y DIGITALES.

- A. La lectura del valor digital de cada canal análogo es de gran velocidad por lo que los valores varían bastante; Además de que la resolución es de 16 bits en la lectura análoga, por lo tanto se recomienda crear una tarea para leer las entradas análogas con un tiempo de ejecución cíclico de 300 ms o realizar un promedio de la lectura realizada.
- B. En cuanto a la salida análoga del canal 0 que controla la posición de la válvula proporcional CV-105, el valor a enviar debe estar entre 0-4070, un valor más grande a 4070 bloquea la comunicación del SoftPLC V1.  
Sin embargo lo mejor es usar la escala lineal de la apertura de la válvula, donde el ángulo puede ser un valor entre 0° y 1710° para obtener un valor entre 0 y 4070, que se muestra a continuación:

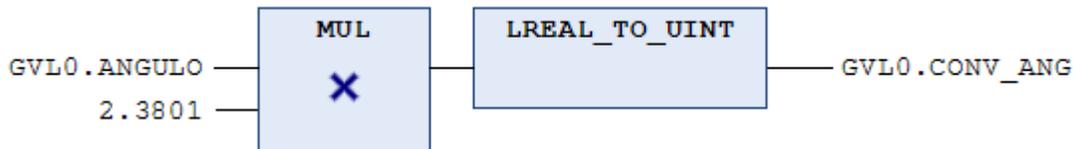


Figura 1.12: Escala lineal de la apertura de la válvula.

- C. Para la lectura de cada Entrada Digital se puede leer el bit que representa la variable total de la entrada digital de la siguiente manera:

1. Crear las variables donde se guardan el bit de cada entrada digital (ver Comentarios):

```
PROGRAM PLC_PRG
VAR
  INP0: BOOL; //PARO DE EMERGENCIA FISICO
  INP1: BOOL; //PARO, BOTON ROJO FISICO
  INP2: BOOL; //INICIO, BOTON VERDE FISICO
  INP3: BOOL; //SENSOR DE NIVEL
  INP4: BOOL; //LIBRE
  INP5: BOOL; //LIBRE
  INP6: BOOL; //LIBRE
  INP7: BOOL; //LIBRE
END_VAR
```

2. Asignar el valor del bit de la variable INP\_DIG en la ejecución del programa de la siguiente manera:

```
INP0:=INP_DIG.0; //PARO DE EMERGENCIA FISICO
INP1:=INP_DIG.1; //PARO, BOTON ROJO FISICO
INP2:=INP_DIG.2; //INICIO, BOTON VERDE FISICO
INP3:=INP_DIG.3; //SENSOR DE NIVEL
INP4:=INP_DIG.4;
INP5:=INP_DIG.5;
INP6:=INP_DIG.6;
INP7:=INP_DIG.7;
```

- D. Las salidas Digitales se realiza de igual forma que en las entradas digitales:

1. Crear las variables que se envían para la salidas digitales (ver Comentarios):

```
PROGRAM PLC_PRG
VAR
  OUT0: BOOL := TRUE; //LUZ ROJA
  OUT1: BOOL := TRUE; //LUZ VERDE
  OUT2: BOOL := TRUE; //LUZ AMARILLA
  OUT3: BOOL := TRUE;
  OUT4: BOOL := TRUE;
  OUT5: BOOL := TRUE;
  OUT6: BOOL := TRUE; //RESISTENCIA
  OUT7: BOOL := TRUE; //BOMBA
END_VAR
```

2. Asignar la variable que se envía a las salidas digitales OUT\_DIG en la ejecución del programa de la siguiente manera:

```
OUT_DIG.0:=OUT0; //LUZ ROJA
OUT_DIG.1:=OUT1; //LUZ VERDE
OUT_DIG.2:=OUT2; //LUZ AMARILLA
OUT_DIG.3:=OUT3;
OUT_DIG.4:=OUT4;
OUT_DIG.5:=OUT5;
OUT_DIG.6:=OUT6; //RESISTENCIA
OUT_DIG.7:=OUT7; //BOMBA
```

Así de esta manera ya es posible crear las rutinas necesarias para la generación de vapor en la caldera creando programas o bloque de funciones en el lenguaje de programación en el que mejor se desempeñe, además existen bloques de funciones PID en la **librería Util** que pueden ser implementados para el control de la válvula proporcional para aumentar la temperatura en algún punto de la pasteurizadora. Para mayor información buscar en la ayuda de CODESYS.

### III. CARGAR EL PROGRAMA A EL SoftPLC V1.0

- A. Con la realización de los pasos anteriores, lo único que falta es cargar el programa al SoftPLC V1.0 y ejecutarlo en línea. Para ello se deben seguir los siguientes pasos:

1. Primero se debe de energizar el SoftPLC V1.0 con el Interruptor que se ve en la figura 1.13:



Figura 1.13: Encendido del SoftPLC V1.0.

2. Esperar unos minutos mientras se inicia el sistema operativo de la SoftPLC V1.0

- B. En la plataforma de programación se debe realizar los siguientes pasos para cargar el programa ver figura 1.14:

1. Escribir la contraseña de la SoftPLC V1.0 la cual es **caldera**.
2. Dar clic en el botón de Scan, donde aparecerá una ventana.
3. Se escoge la dirección IP de la SoftPLC V1.0 (130.130.130.111)
4. Y luego clic en OK.
5. Clic en examinar, donde aparecerá la ventana de seleccionar dispositivo,
6. Se da clic en la IP del dispositivo.
7. Y por último clic en aceptar.

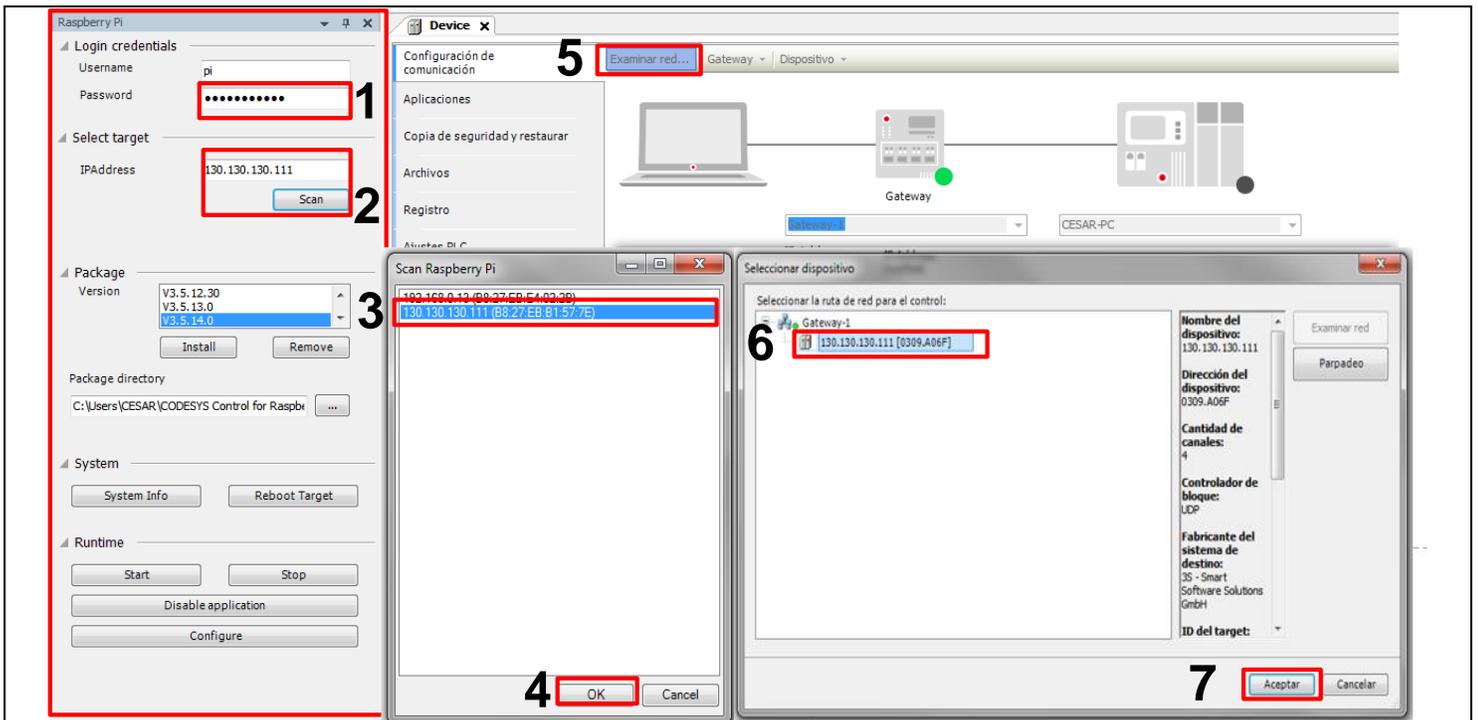


Figura 1.14: Configuración de reconocimiento SoftPLC V1.0 en CODESYS.

C. En la ventana del dispositivo aparecerá activo la configuración del Gateway-1 y del dispositivo la SoftPLC V1.0, ver figura 1.15.

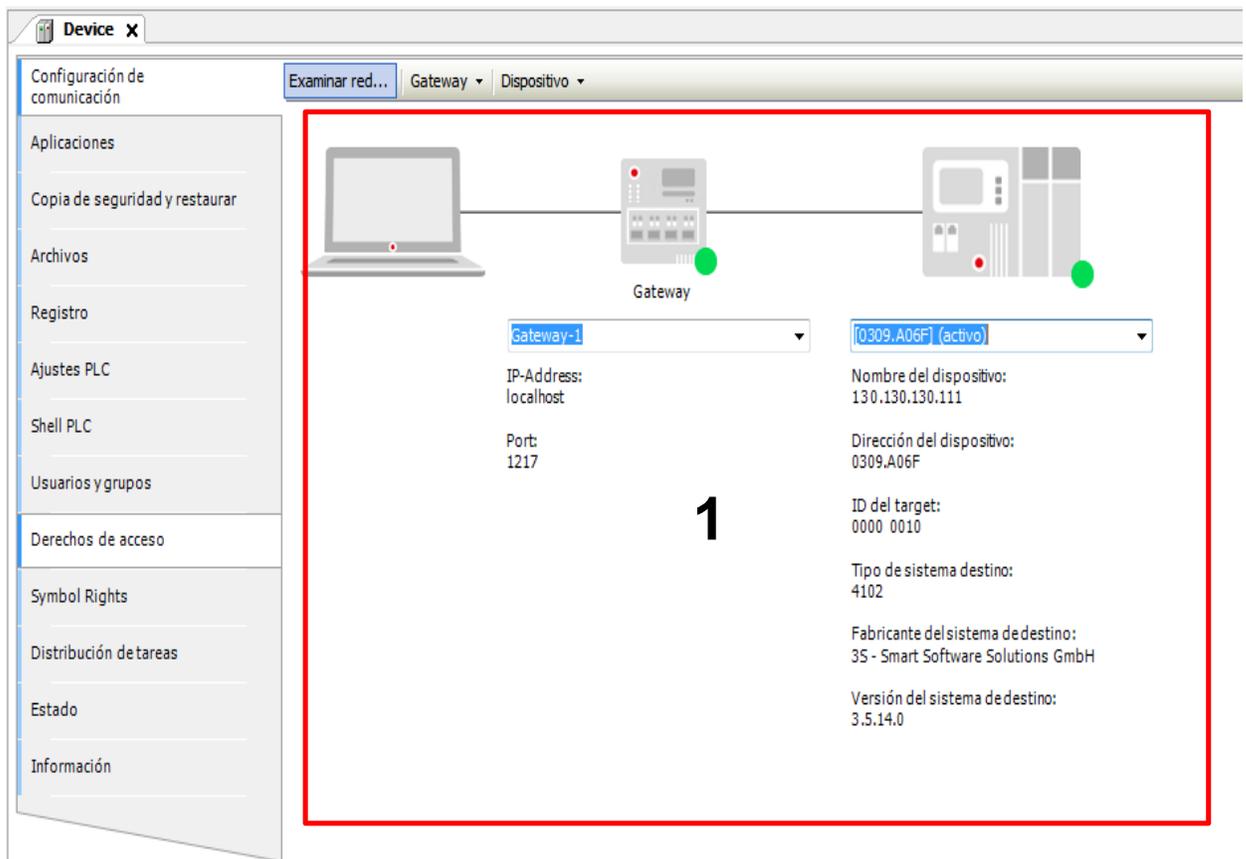


Figura 1.15: Dispositivo Activo SoftPLC V1.0 en CODESYS.

- D. En la parte superior de del entorno de programación se debe de buscar la herramienta de En línea, dar clic y luego dar clic en Iniciar Sesión ALT+F8, ver figura 1.16.

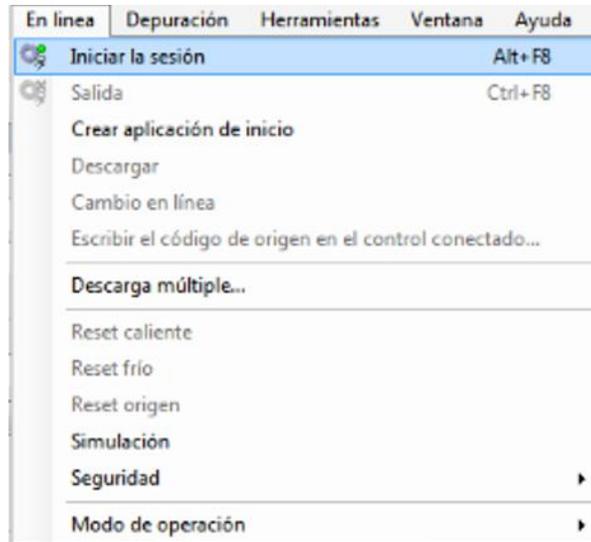


Figura 1.16: Carga del programa al SoftPLC V1.0.

- E. Aparece la ventana para aceptar si se quiere o no cargar la Aplicación en el SoftPLC V1.0

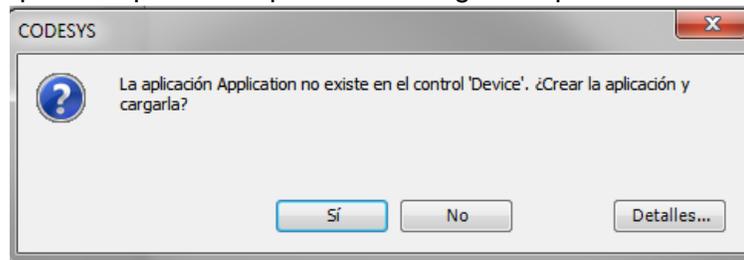


Figura 1.17: Carga del programa al SoftPLC V1.0.

- F. Por ultimo aparece en el árbol de procesos ubicado a la izquierda de la pantalla el dispositivo resaltado en verde ver figura 1.18.

1. Ubicarse en el cuadro rojo y dar clic en Inicio de la aplicación.

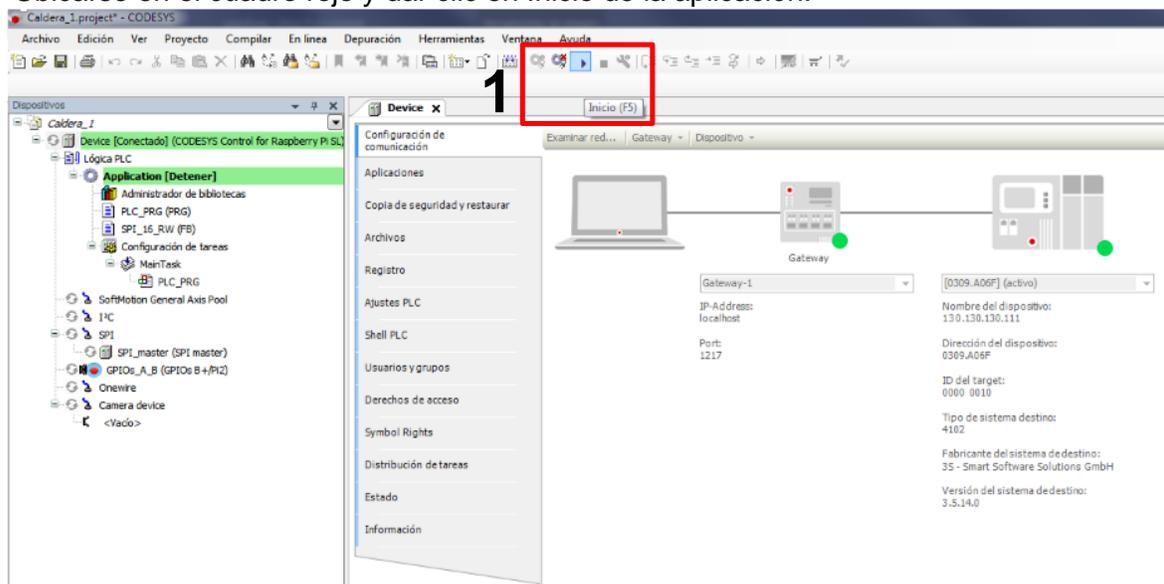


Figura 1.18: Inicio Online del programa en el SoftPLC V1.0.

#### IV. ENCENDIDO DEL CONTROLADOR DE LA VÁLVULA PROPORCIONAL CV-105.

Para el manejo de la válvula proporcional CV-105 se debe tener en cuenta lo siguiente:

1. En la parte del SELECTOR en la figura 1.19, se debe configurar el DIP SWITCH [1 2 3] según el tipo de señal de entrada que va a controlar a CV-105:
  - 100 >>>>Señal de control 4-20mA.
  - 010 >>>>Señal de control 0-5V.
  - 001 >>>>Señal de control 0-10V.
  - 111 >>>>Comunicación Serial RS232.
  - a. La señal de control que ejerce el SoftPLC V1.0 sobre CV-105 es en un lazo de corriente de 4-20mA.
  - b. La señal de control que ejerce el PLC Compactlogix L23E QB1B sobre CV-105 es por comunicación serial.
2. Encender el controlador de CV-105 con los dos interruptores (5V y 24V) ubicados a la parte derecha inferior, ver figura 1.19.

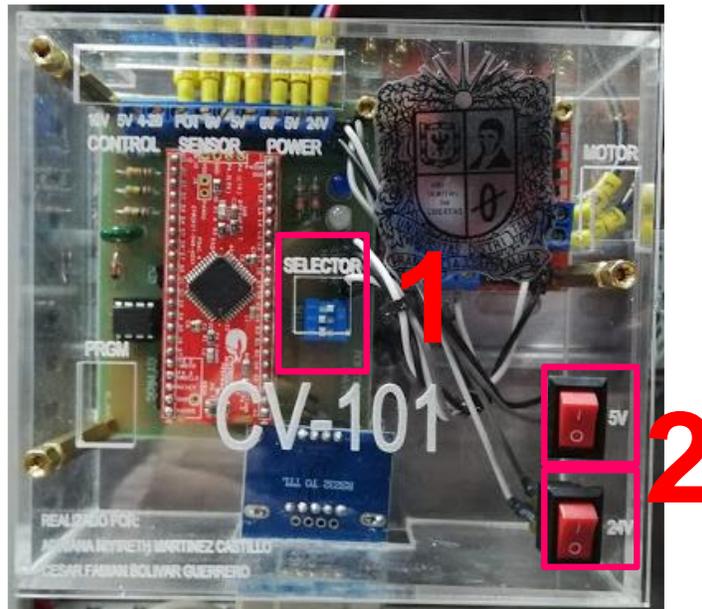


Figura 19: Configuración y encendido de CV-105.

Para que los sensores y actuadores de la caldera funcionen con Rockwell o CODESYS, se debe tener en cuenta la activación de F2, ver figura 1.20.

- I >>>> PLC Compactlogix L23E QB1B.
- O >>>> SoftPLC V1.0.



Figura 1.20: Selección de Dispositivo de Control.

## V. CONFIGURACIÓN Y PUESTA EN MARCHA DE LA HMI DE 7” 800X480

Para la visualización en la HMI 7” se puede realizar de dos maneras:

1. A través de la plataforma de programación CODESYS se puede realizar una HMI por medio de la plataforma donde se puede interactuar directamente con ella cuando el SoftPLC V1.0 está en línea. También se puede interactuar con la HMI por una visualización web.
2. La otra forma es con el SCADA de Inductive Automation llamado ‘Ignition EDGE’ el cual tiene soporte en procesadores ARM para Integrarse en dispositivos de campo donde se requiera visualización de la HMI, servidor remoto que se sincroniza a un servidor central o como transmisor de datos por MQTT Publisher remoto a MQTT Cliente.

### HMI a través de la plataforma CODESYS:

En la figura 1.21 se puede ver la plataforma de programación CODESYS, donde se muestran los pasos a seguir para crear la HMI.

1. Se debe dar clic derecho sobre Application.
2. Buscar Agregar objeto y dar clic.
3. Buscar Visualización y dar clic.

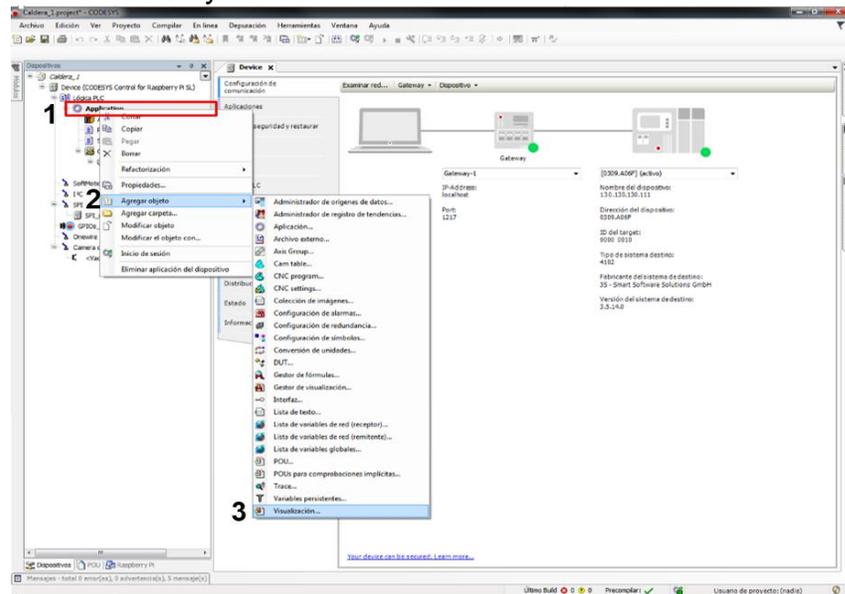
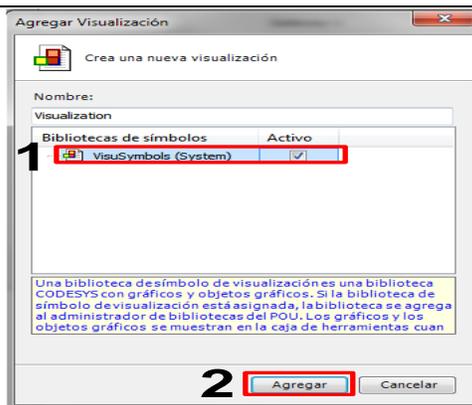


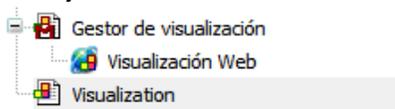
Figura 1.21: Creación del Objeto de Visualización.

- A. Aparece la ventana de agregar visualización, ver figura 1.22.
1. Seleccionar
  2. Dar clic en Agregar.



**Figura 1.22:** Creación de una nueva Visualización.

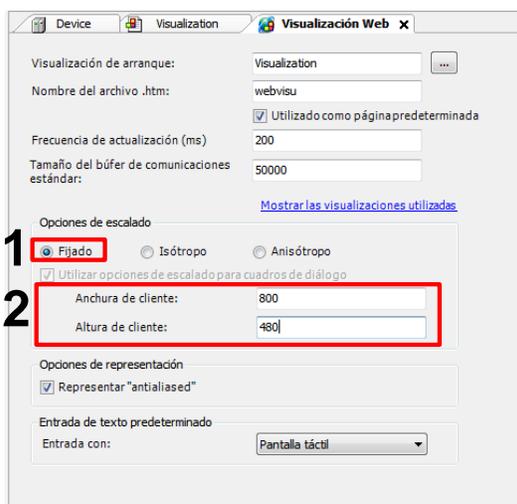
**B.** En el árbol de procesos aparece el objeto creado



**Figura 1.23:** visualización en el árbol de procesos.

**C.** Doble clic sobre Visualización Web se despliega sobre el área de trabajo la ventana de configuraciones, ver figura 1.23.

1. En opciones de escalado escoger Fijado.
2. Escribir los valores del tamaño de la pantalla HMI 7"



**Figura 1.24:** Configuración de la visualización Web.

**D.** Doble clic sobre el objeto de visualización creado en el árbol de procesos abre la ventana donde se configuran los elementos de control y supervisión, ver figura 1.25.

1. Área de trabajo donde se pueden insertar los elementos de control como los que se pueden ver en la figura 1.25 e interactuar con estos cuando este en Línea el SoftPLC V1.0-
2. El toolbox de visualización donde se pueden agregar los elementos de control y supervisión, para colocarlos en el Área de trabajo y luego modificar sus propiedades.

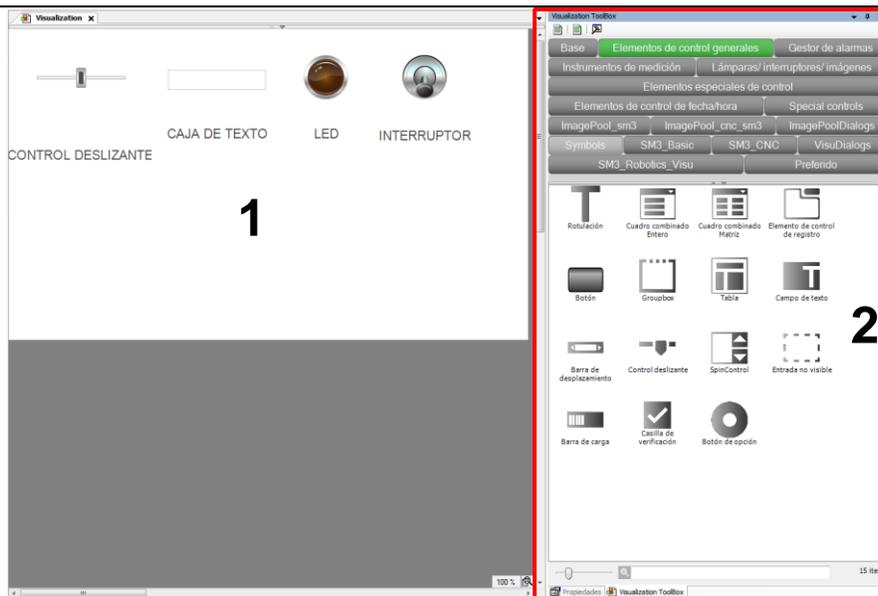


Figura 1.25: Configuración de la visualización Web.

- E. Para interactuar con la HMI por medio de una visualización web se debe tener la siguiente dirección web: **130.130.130.111:8080/webvisu.htm**

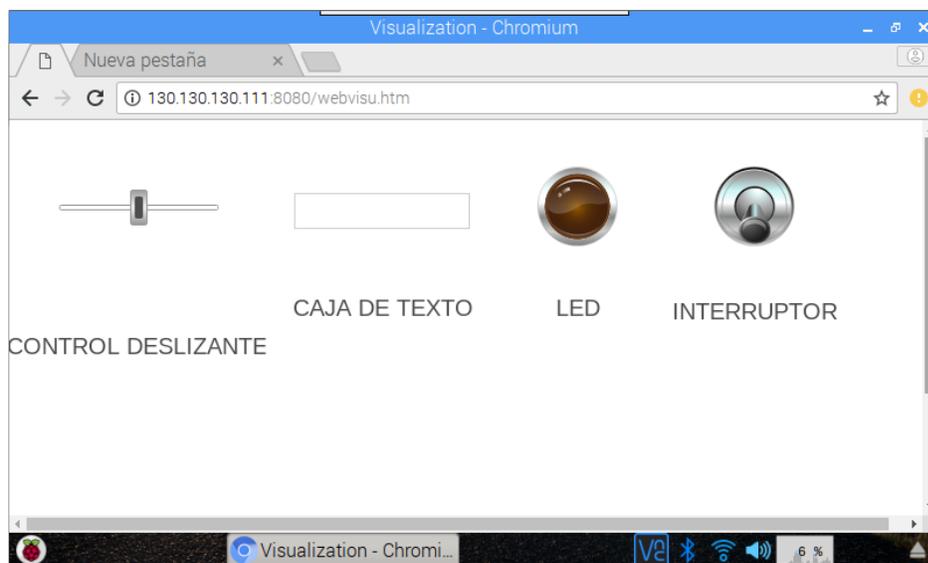


Figura 1.26: Imagen tomada desde la Pantalla HMI 7”.

### HMI a través de la plataforma Ignition EDGE.

- A. Para usar la pantalla con el sistema SCADA multiplataforma Ignition EDGE, se debe instalar VNC Viewer o algún software que le permita crear un acceso remoto con el SoftPLC V1.0 para una mejor manipulación desde el PC, además de tener configurado el programa de CODESYS para la conexión OPC-UA y que esté en Línea.
- B. Al colocar la IP del SoftPLC V1.0 en el programa de acceso remoto, este pide:
  - login: pi.
  - Password: caldera
- C. Aparece la ventana de comandos por Terminal del SoftPLC V1.0, como se ve en la figura 1.27:
  1. Ingresar el comando `cd /usr/local/ignition`
  2. Luego en el directorio de ignition ingresar el siguiente comando `sudo ./ignition.sh start`

3. por último esperar a que se inicie el servicio.

```
pi@130: /usr/local/ignition
login as: pi
pi@130.130.130.111's password:
Linux 130.130.130.111 4.14.34-v7+ #1110 SMP Mon Apr 16 15:18:51 BST 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Feb 11 02:35:43 2018 from 130.130.130.1
pi@130:~ $ cd /usr/local/ignition
pi@130:/usr/local/ignition $ sudo ./ignition.sh start
Starting Ignition-Gateway...
Waiting for Ignition-Gateway.....
running: PID:1553
pi@130:/usr/local/ignition $
```

Figura 1.27: inicio del servicio de Ignition EDGE.

D. Se debe abrir el Web Page del Gateway para realizar las configuraciones pertinentes en cuanto a lo que se necesite; tener en cuenta que es un Software con licencia de prueba, esta prueba dura solamente 2 horas y se debe reiniciar la versión Trial (1), ver figura 1.28.

- La página es 130.130.130.111:8088/main/web/home?1

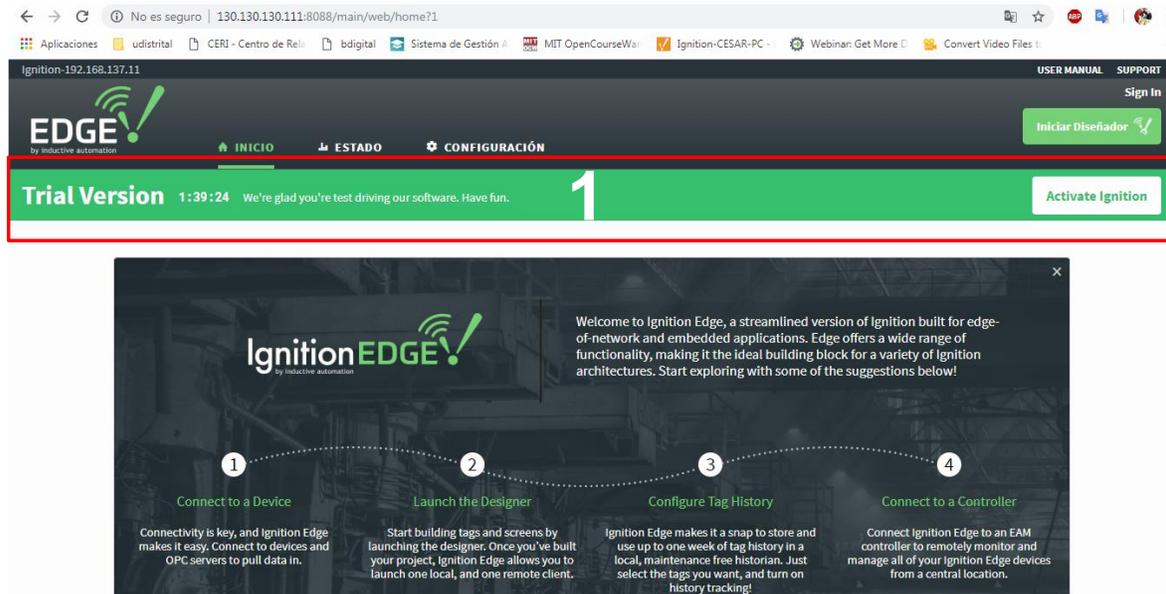


Figura 1.28: Gateway Web Page.

E. En este SCADA multiplataforma se puede configurar diferentes tipos de comunicación con el SoftPLC V1.0, en esta práctica se muestra la conexión a través de OPC-UA, para ello se debe configurar en el Gateway Web page el servidor para la conexión.

1. Clic en configuración.
2. Buscar OPC CONNECTIONS, dar clic en server.

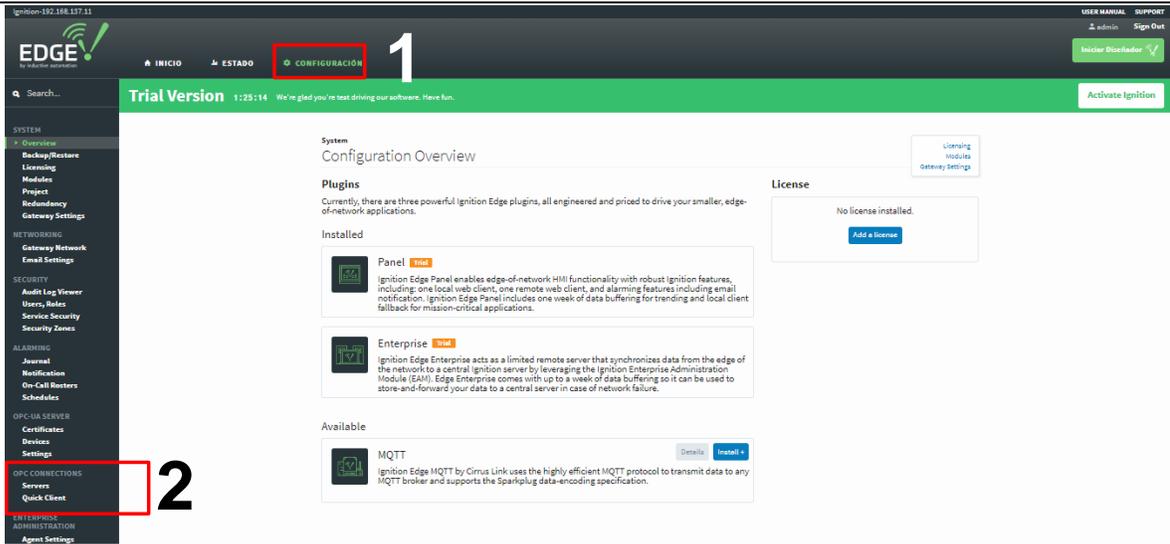


Figura 1.29: Conexión OPC-UA Ignition Edge con SoftPLC V1.0.

3. En la ventana que aparece, dar clic en >> Create new OPC Server Connection, ver figura 1.30.

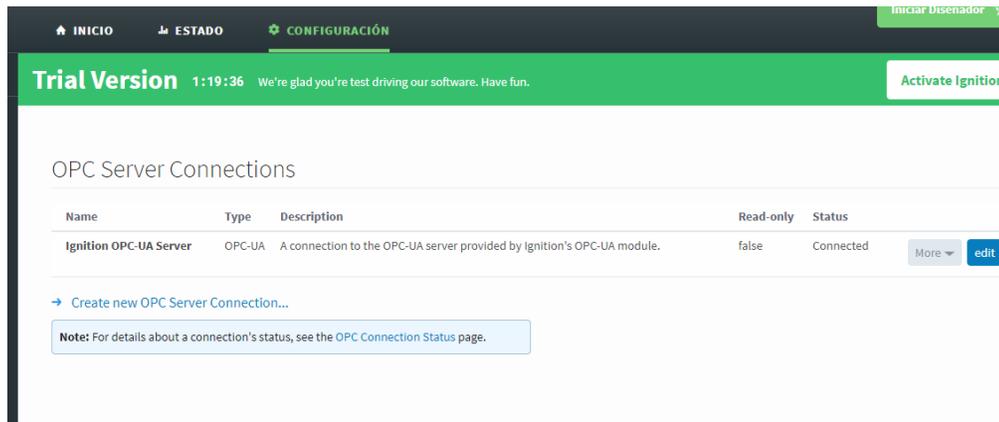


Figura 1.29: Creación de nueva conexión OPC server.

4. Se debe tener el OPC-UA Endpoints que configura CODESYS al SoftPLC, que en este caso es:
  - opc.tcp://130.130.130.111:4840
5. dar clic en Next.

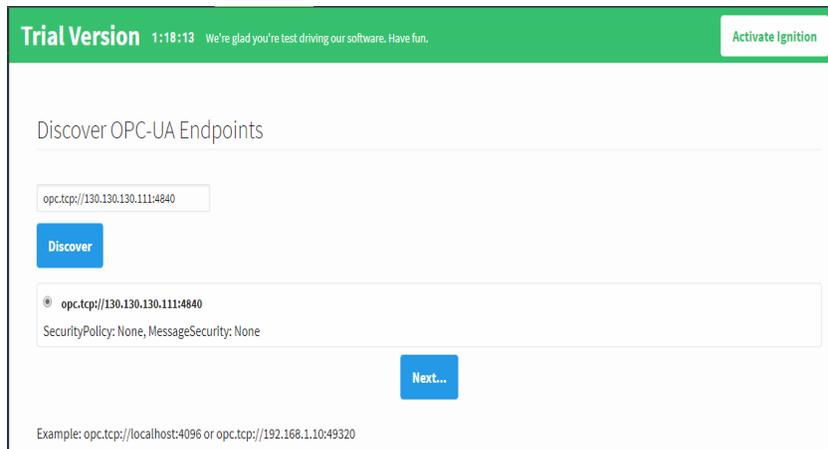


Figura 1.30: Punto de conexión OPC-UA.

- F. En la siguiente ventana se debe:
1. Escribir nombre que identifique la conexión.
  2. Clic en la creación del nuevo OPC Server.

New OpcUaConnectionSettings

**Main**

Name  **1**

Description

Read-only  If selected, the opc server will be read-only, and calls to write will fail. (default: false)

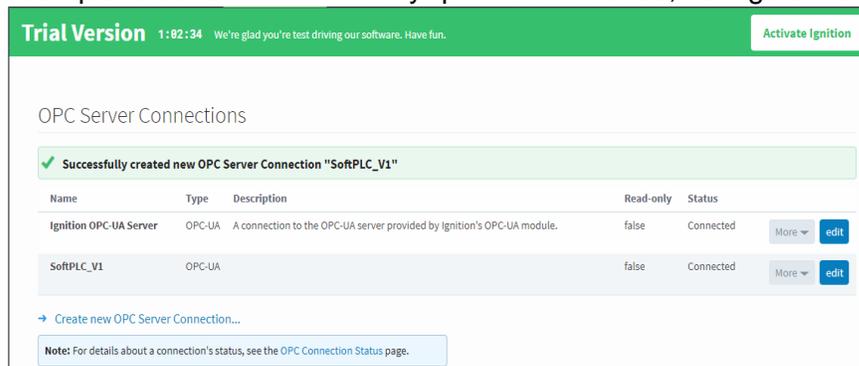
Enabled  (default: true)

**2**

Create New OPC Server Connection

**Figura 1.31:** Nombre y finalización de la configuración.

- G. Al final debe aparecer que la conexión es buena y que no tiene fallas, ver figura 1.32.



**Figura 1.32:** estado de conexión con el SoftPLC V1.0.

- H. Luego de tener la conexión del OPC Server, se debe realizar el diseño de la HMI, dando clic en:



Donde se descarga un archivo al PC, ya que estamos en modo remoto y el Gateway está funcionando y se puede ver en el mismo dominio de red; el archivo que se descarga tiene el nombre de designer.jnlp Este archivo se debe de abrir, el cual tomara un tiempo en abrirse.

Luego aparece la siguiente ventana:

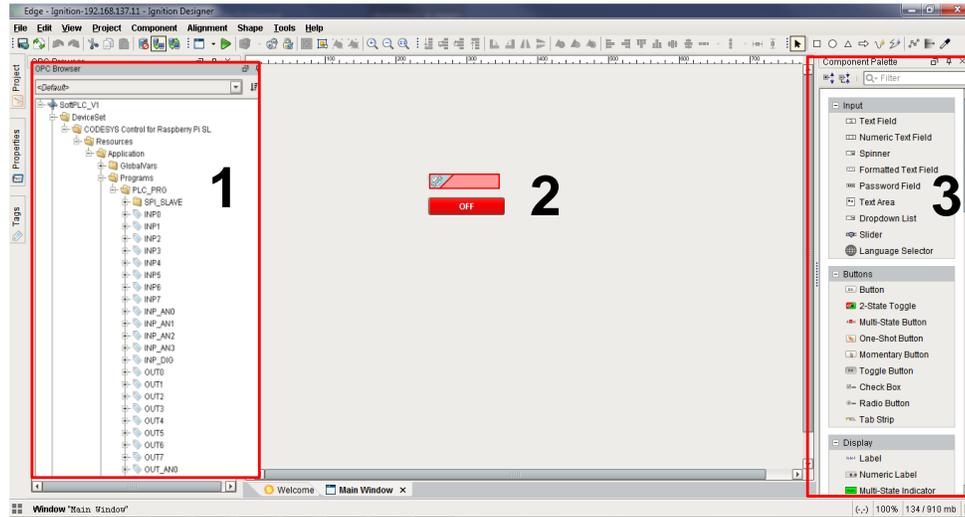
- Username: admin
- Password: password



**Figura 1.33:** Inicio de Diseñador HMI.

- I. Se abre el entorno para el desarrollo del control y supervisión de variables que están en el proceso de generación de vapor, ver figura 1.34.

1. Se visualiza la conexión por OPC-UA donde aparece en el árbol de procesos las variables para la generación de vapor.
2. Área de trabajo donde se desarrolla la HMI.
3. Paleta de componentes para el control y supervisión del proceso que deben ser añadidas al área de trabajo.



**Figura 1.34:** Inicio de Diseñador HMI.

## 11.9 ANEXO 9- PRÁCTICA DE IDENTIFICACIÓN Y DISEÑO DEL CONTROLADOR.

	<p><b>UNIDAD TEMÁTICA:</b> IDENTIFICACIÓN DE PROCESOS Y DISEÑO DE UN CONTROLADOR PARA EL CALENTAMIENTO DEL TANQUE DE LA DOSIFICADORA A TRAVÉS DE CV-105.</p>	
	<p><b>ACTIVIDAD:</b> Control en lazo cerrado de CV-105</p>	
		
<p><b>Código:</b> PC002-B</p>	<p><input checked="" type="checkbox"/> <b>ONLINE</b>   <input type="checkbox"/> <b>OFFLINE</b></p>	<p><b>Duración:</b> 3 h</p>

### Objetivo de la Práctica:

Hallar el modelo matemático de planta del tanque de calentamiento de la dosificadora mediante la identificación de procesos en lazo abierto.

### Material necesario y requisitos para el desarrollo de la Actividad Práctica:

- Software CODESYS V3.5 SP14. O RSLogix 5000 V19
- MATLAB 2016 0 superiores

### Esquema Grafico de la Actividad Práctica.



**Figura 2.1:** Grafico de los requerimientos para la práctica.

### Requerimientos previos.

Para la elaboración de esta práctica se debe haber realizado el control y supervisión de la caldera tanto con el SoftPLC V1.0 en CODESYS o en el CompactLogix L23E en RSLogix 5000.

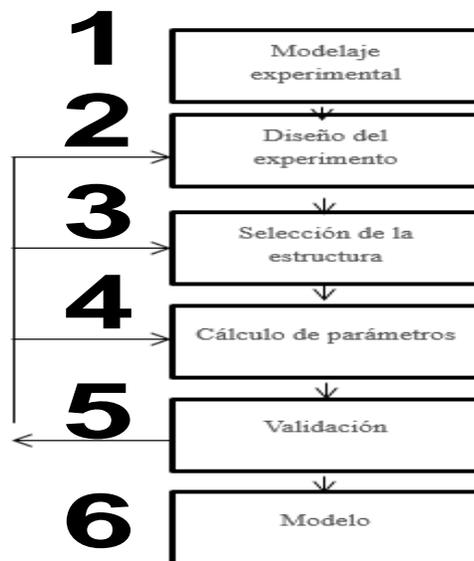
### Funcionamiento de la estación:

El funcionamiento del sistema de la caldera se basa en generar vapor, este se logra mediante el calentamiento del agua hasta el punto de ebullición, el agua ingresa al sistema a partir del tanque el cual debe tener un nivel mínimo de agua para que la bomba no entre en cavitación, el nivel del agua que se está suministrando a la caldera se puede verificar mediante el sensor de nivel, y al alcanzar el nivel necesario, apaga la bomba y se enciende la resistencia para comenzar el calentamiento, este proceso dura aproximadamente 1 hora y se debe tener presente que la presión medida por el manómetro no supere los 40 psi. Una vez realizado el proceso se puede suministrar vapor a la estación de dosificación a través de la válvula proporcional.

## Desarrollo de la Actividad Práctica:

### 1. Experimentación y Almacenamiento de Datos.

La figura 2.2 muestra la metodología que siempre se debe implementar para identificación de procesos, donde tenemos en primer lugar; un diseño del experimento, después se escoge una estructura, se realiza cálculo de parámetros y por ultimo validación, como se ve a continuación:

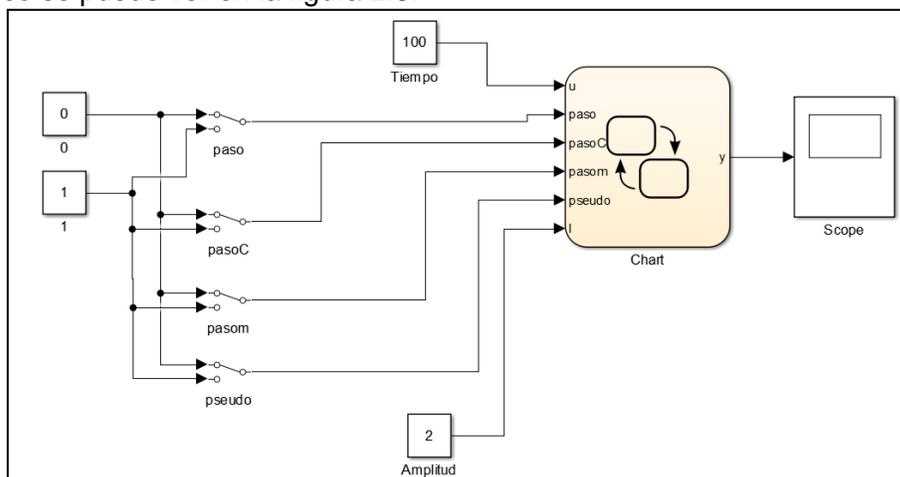


**Figura 2.2:** Metodología general para la identificación de Sistemas

Par la realización de la metodología de identificación de sistemas, del paso 1 al 6 se realiza mediante una serie de pasos a seguir, el paso 1 se tiene ya instaurado de acuerdo a las 4 señales que se deben de aplicar a la planta, el paso 2 son las señales del experimento ya escogidas en el paso anterior seleccionando el punto de operación al que se debe de aplicar los experimentos.

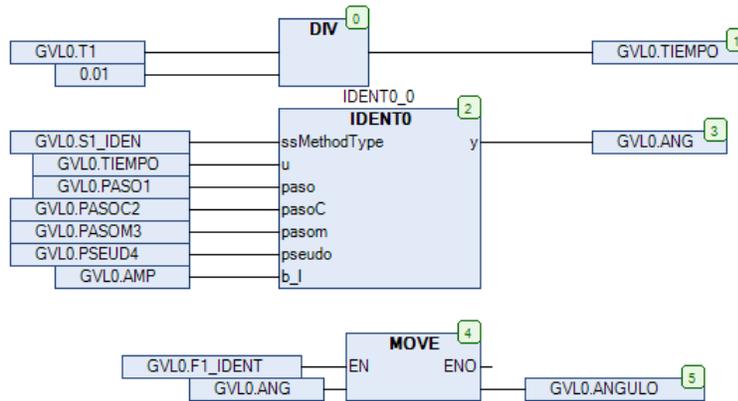
Del paso 3 al 6 se realizan mediante el uso de System Identification de MATLAB, la teoría de la metodología general para la Identificación de sistemas se puede encontrar en el documento '**SEÑALES PARA PROCESOS DE IDENTIFICACIÓN EN LAZO ABIERTO**'. Para esta práctica se realizó los siguientes pasos.

- A. Se diseña el experimento en StateFlow de MATLAB, donde se puede generar las señales de identificación como la Señal Paso, la Señal Paso modificada, la Señal Paso Cuadrada y la Señal Pseudoaleatoria; donde se puede ingresar el tiempo de ejecución de la señal y la amplitud de la misma, el esquema básico se puede ver en la figura 2.3.



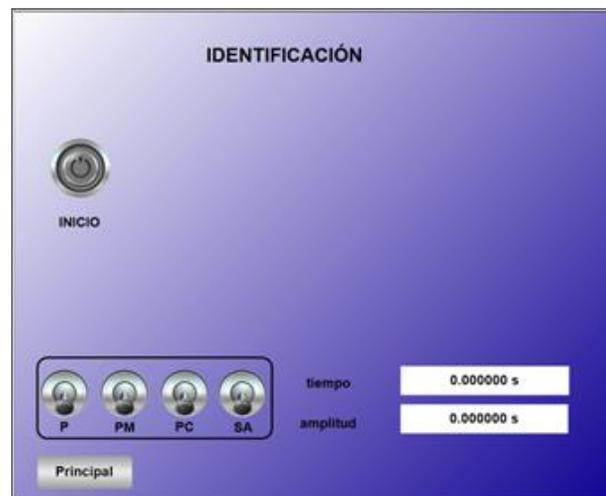
**Figura 2.3:** Diseño del experimento

B. Luego de tener el diseño del experimento generado en StateFlow de MATLAB se debe de generar el código en este caso de PLCopen XML que se exporta como texto estructurado en un Bloque de Funciones que se implementa en CODESYS, ver figura 2.4, el tiempo depende de la tarea en la que este configurada el programa.



**Figura 2.4:** Bloque de funciones de las señales de identificación.

C. Se le asignan los parametros escogiendo la señal que se inyecta a la valvula proporcional como la amplitud y el tiempo de duracion o repeticion de la misma, a continuacion se ve la interfaz.



**Figura 2.5:** Interfaz realizada en CODESYS.

Con la generación de las señales, la información de estas y de la respuesta de la planta se deben de guardar en alguna tipo de archivo, o en una base de datos para que puedan ser procesadas y así obtener el modelo matemático representativo de la planta, para ello en este caso se usa la traza y se guardan los datos en un archivo de texto, CSV, o en trace.

A continuación de muestra el paso a paso desde la creación de la traza hasta el almacenamiento de los datos.

D. Se debe crear el objeto de la traza para graficar y guardar los datos de la siguiente forma, ver figura 2.6.

1. Clic derecho sobre Application.
2. Clic sobre crear Objeto.
3. Buscar Trace y dar clic encima.

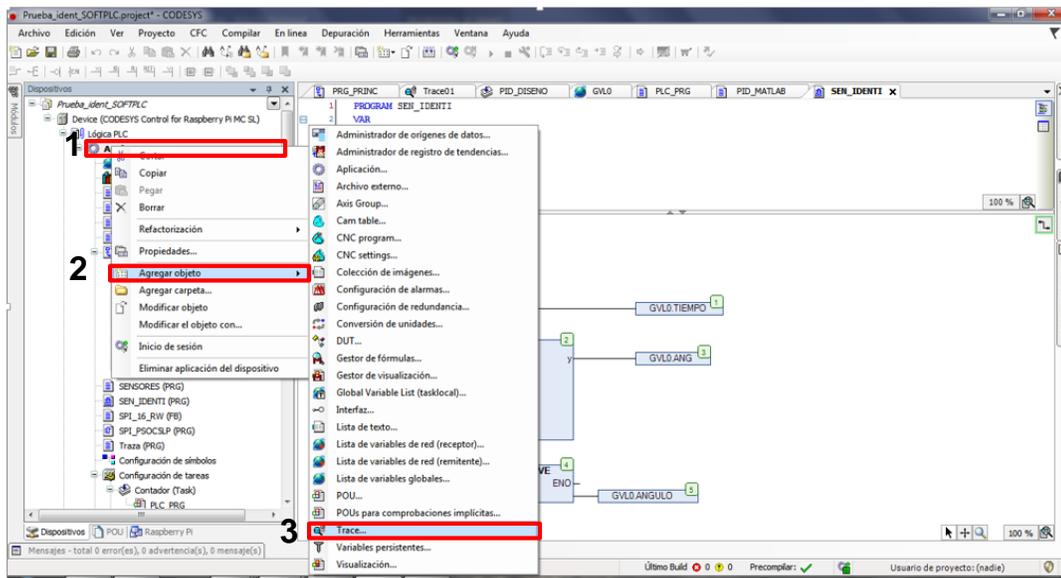


Figura 2.6: Creación de la Traza.

- E. Se crea una tarea nueva para la ejecución de la traza y un programa donde estén las variables que se quieren graficar y los datos a Guardar, ver figura 2.7.
1. Crear la Tarea en Configuración de tareas.
  2. Aparecerá la Tarea con el nombre dado, y se adiciona el programa que se ejecuta en esa Tarea, en este caso el programa se llama Traza.
  3. Por último se debe colocar que sea cíclica, y darle el tiempo en que se quiera guardar y graficar los datos.

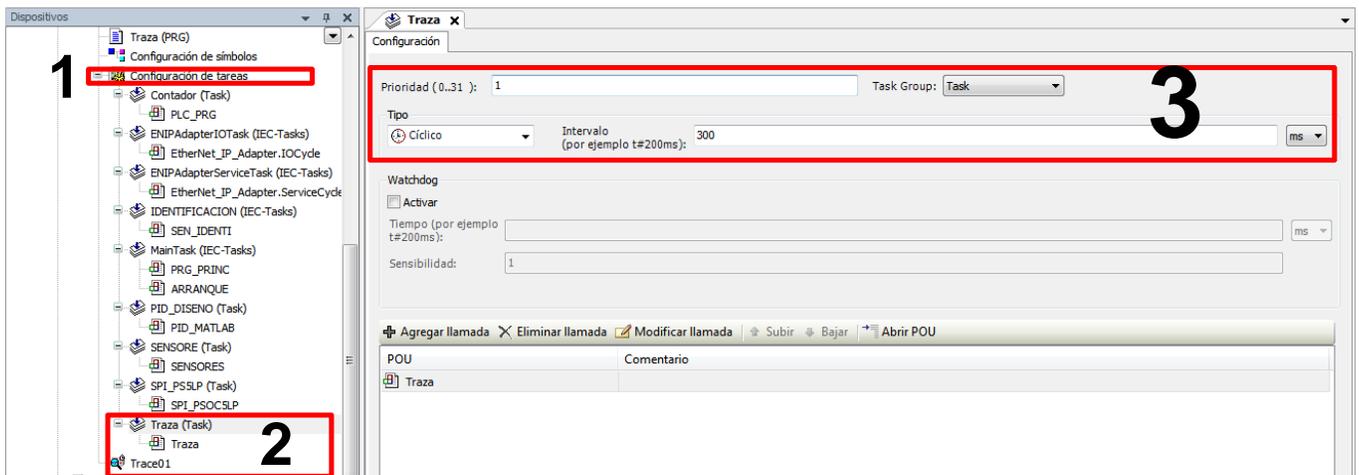
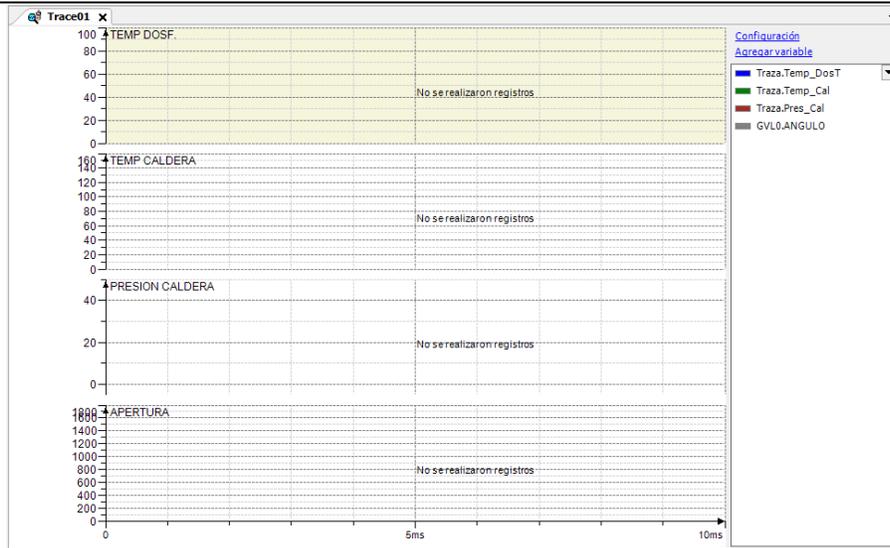


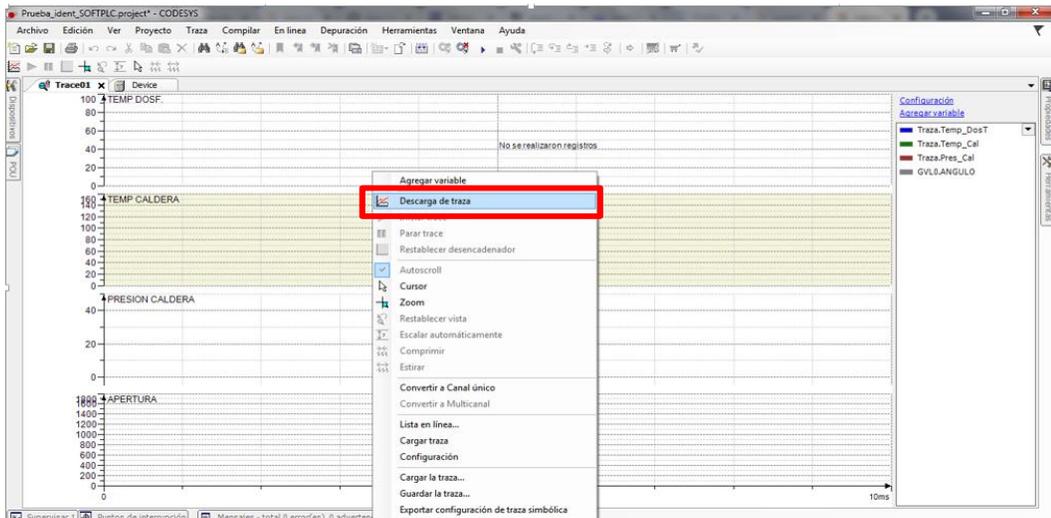
Figura 2.7: Configuración de la tarea de la traza.

- F. Se abre el Trace creado y aparece la siguiente ventana, ver figura 2.8. En la traza se agregan las variables a Graficar y se configura el tamaño del buffer para guardar los datos.

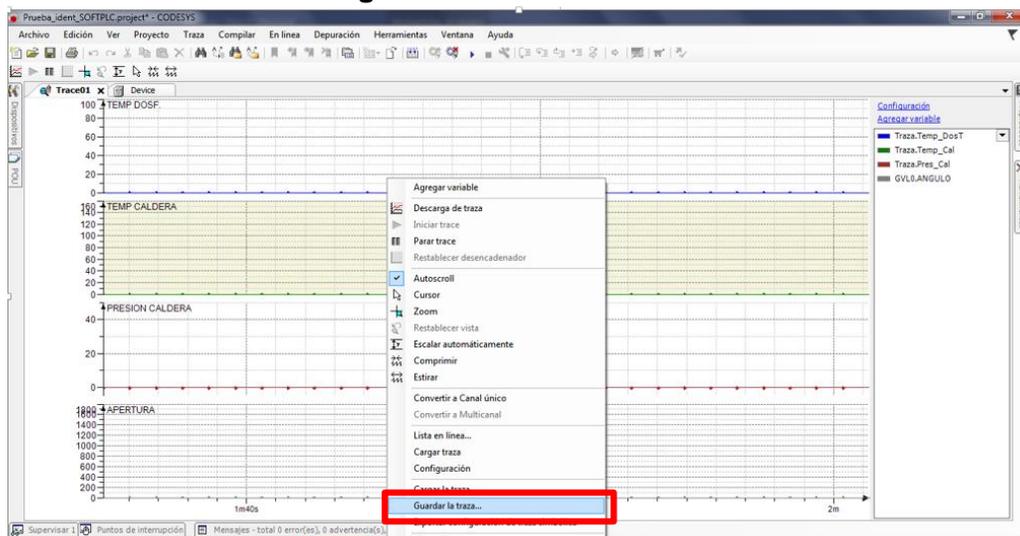


**Figura 2.8:** Traza para los datos a graficar y guardar.

- G.** Ya cuando se esté ejecutando el programa en el SoftPLC V1.0 se debe de descargar la traza, que es prácticamente el inicio de la gráfica de los datos y el almacenamiento de los mismos, ver figura 2.9 y 2.10.



**Figura 2.9:** Inicio de la traza.



**Figura 2.10:** Almacenamiento de los datos de la Traza.

**NOTA:** Cabe indicar que el momento de guardar los datos es al final del experimento de las señales de identificación, y el archivo que almacena todas las variables es el .txt, en la figura 2.11 se ve la ventana que sigue después de guardar la traza.

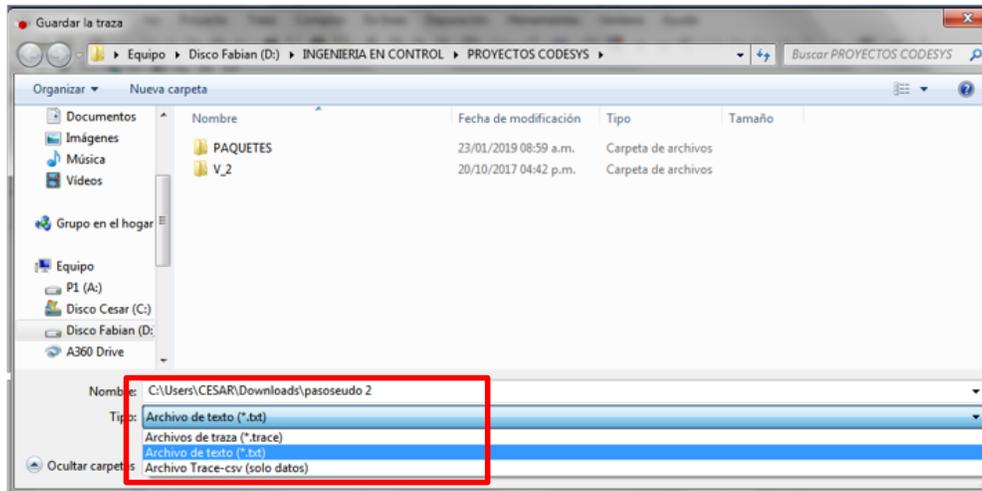


Figura 2.11: tipos de almacenamiento de la traza.

## Parte 2: Selección de la estructura y Modelo.

### 2. SELECCIÓN DE LA ESTRUCTURA (CONTINUO DE PRIMER ORDEN)

#### a. Captura de Datos

En este paso vamos a utilizar IDENT de Matlab, Donde se hace el proceso de importación de los datos de entrada y de salida, así como se muestra en la figura 2.12.

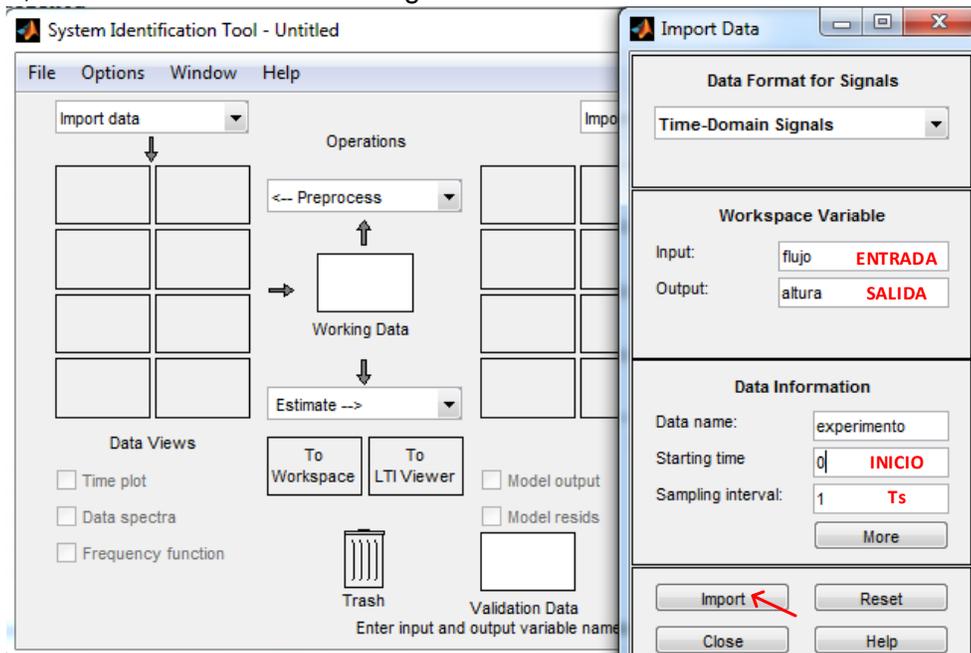
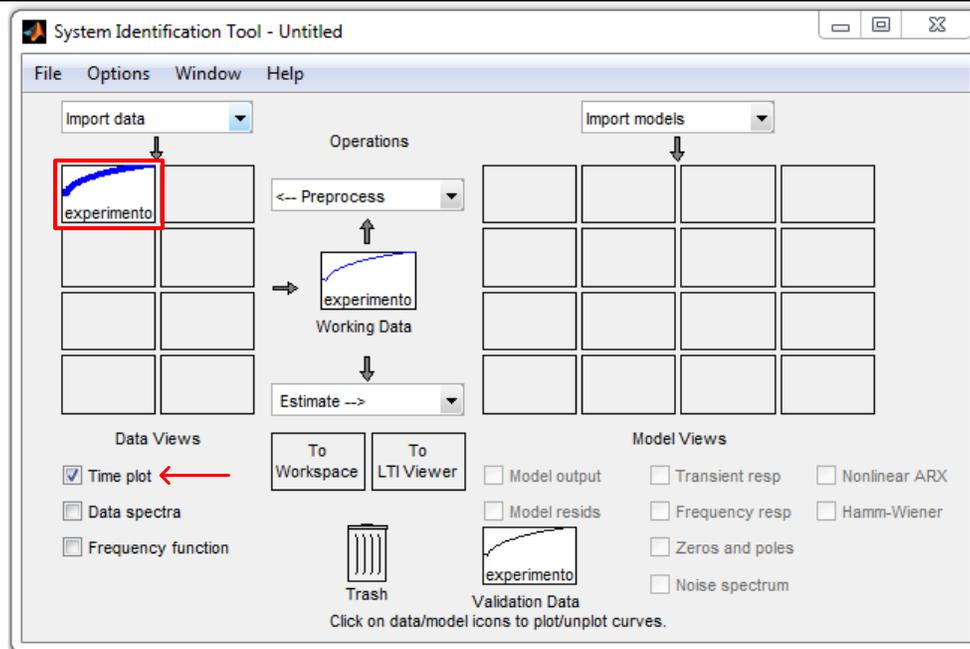
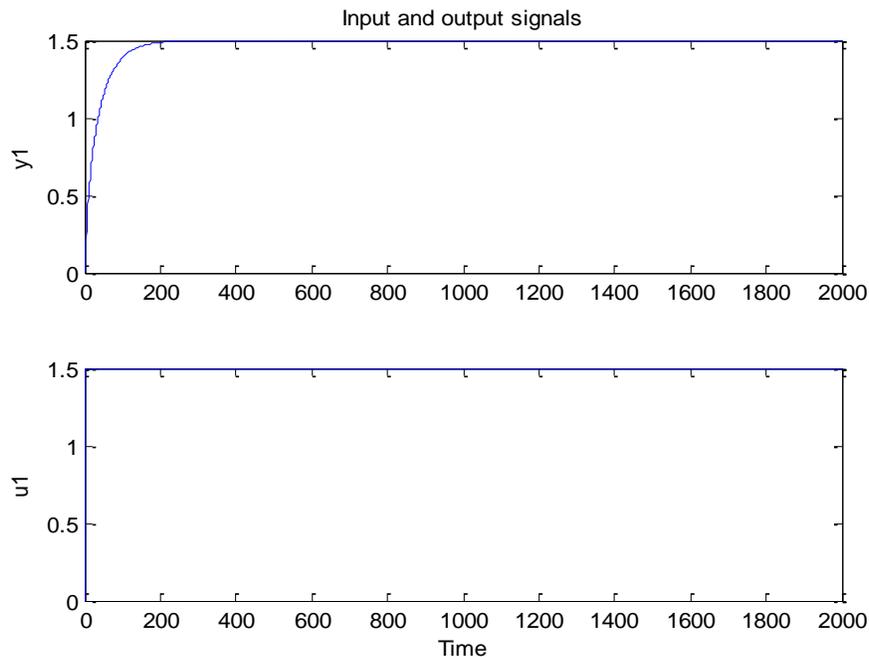


Figura 2.12: Configuración parámetros - Import Data.

Ya teniendo la configuración de todos los parámetros de la importación de datos, damos clic en **Import** para graficar los datos. Nos debe aparecer una gráfica guardada en color azul en el primer recuadro de las ocho posibles gráficas, como se ve en la figura 2.13, en seguida se selecciona la opción **Time plot**, para poder ver la gráfica que se genera automáticamente al escoger este parámetro, como se muestra en la figura 2.14.



**Figura 2.13:** Configuración parámetros – Experimento.



**Figura 2.14:** Señales estímulo y respuesta del sistema.

En la figura 2.13 se muestra la verificación de los datos ingresados en Import Data, donde podemos visualizar el estímulo – la entrada ( $U_1$ ) y la salida del sistema ( $Y_1$ ).

Se debe tener en cuenta que se pueden importar hasta 8 ejemplos de experimentos de un solo sistema o de diferentes sistemas, para escoger la señal con la que se desee trabajar se debe seleccionar y luego arrastrar hasta **Working Data** (recuadro verde), después se debe arrastrar los datos hasta **Validation Data** (recuadro magenta) para así asegurarnos que con el experimento calculo el modelo y realizo la validación del modelo, ver figura 2.15.

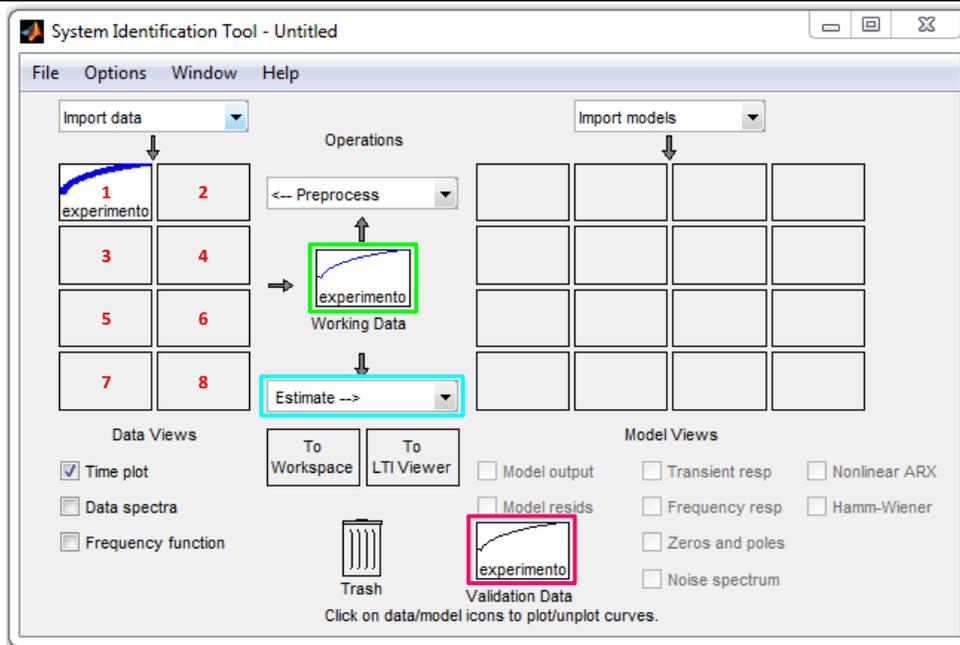


Figura 2.15: Configuración Working Data, Validation Data.

En el recuadro azul –Estimate - de la anterior figura se selecciona una estructura y elegimos la estructura para trabajar con sistemas continuos **Process Models**, como se ve en la figura 2.16.

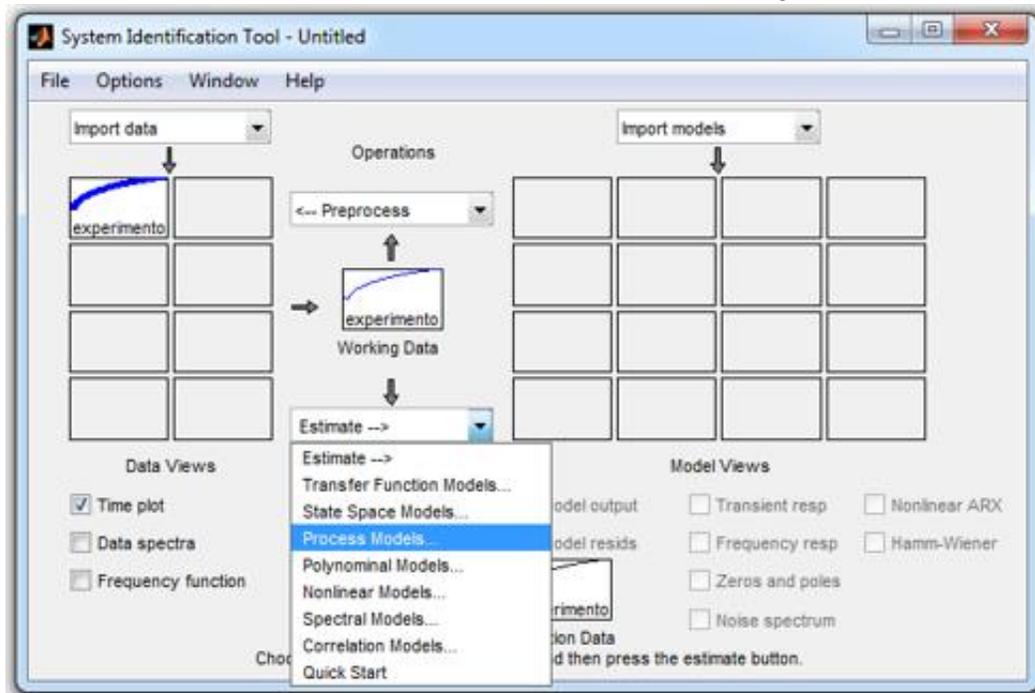


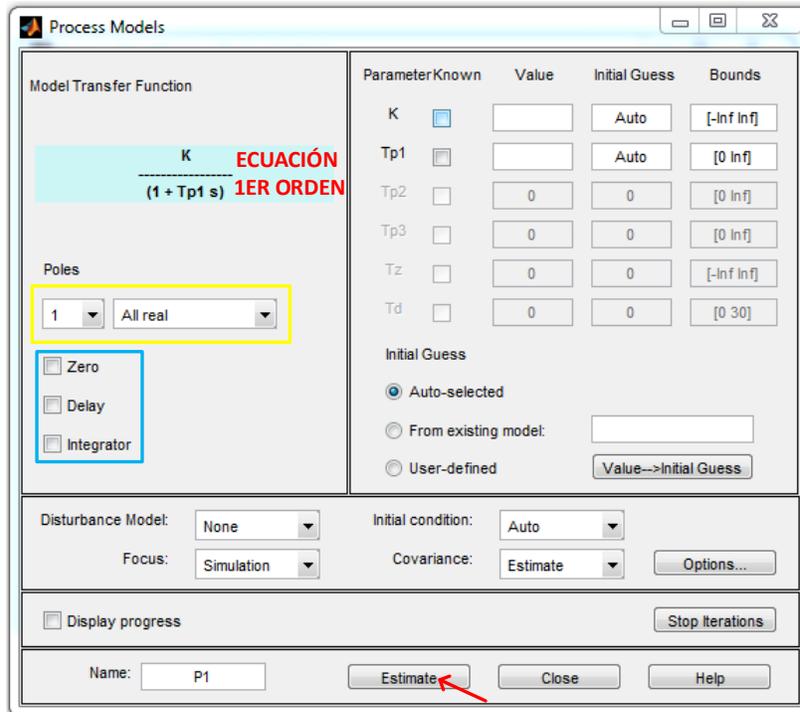
Figura 2.16. SELECCIÓN ESTRUCTURA – Sistema continuo.

### 3. CÁLCULO DE PARÁMETROS

#### a. Explicación General de Toolbox

Automáticamente se genera una nueva ventana como se ve en la figura 2.17, en donde permite a través de los selectores (recuadro amarillo) escoger el número de polos y elegir si el polo es real o imaginario, además se puede escoger un cero, un retardo o un integrador (recuadro azul).

Para este ejemplo configuramos un sistema de primer orden, al haber terminado de configurar los parámetros dentro de **Process Models**, escogemos **Estimate** para que realice el cálculo de la constante K (Ganancia) y TP1 (constante de tiempo).



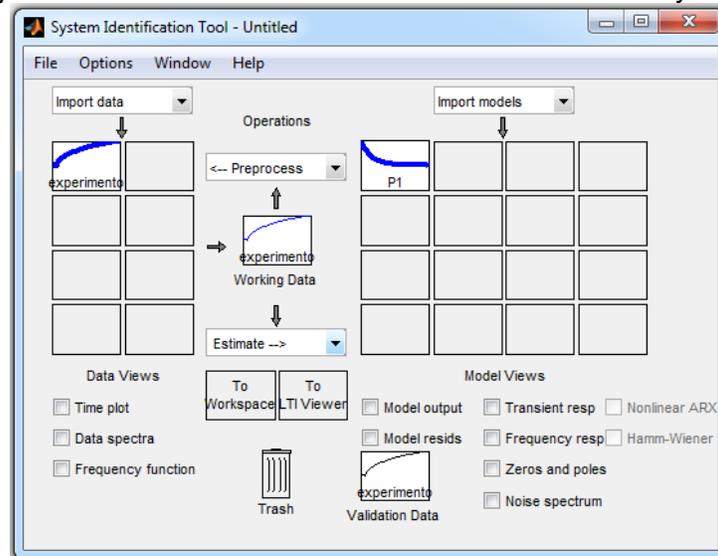
**Figura 2.17:** Configuración Process Models. (Configuración de diferentes funciones de transferencia)

En la figura 2.18 se observa cómo se genera una nueva señal en la parte derecha de mi ventana principal de System Identification (P1) y me genera los valores de K y TP1 que son:

**$K = 0.9992$**

**$TP1 = 35.8103$**

Matlab genera los resultados de las constantes, por medio de mínimos cuadrados y regresión lineal, en la figura 2.19 se muestran los datos arrojados por **process Models**. También se puede observar un coeficiente de correlación (**92.31%**) y el ajuste cuando más cercano es a 100% la correlación y la regresión es buena.



**Figura 2.18:** Generación de P1.

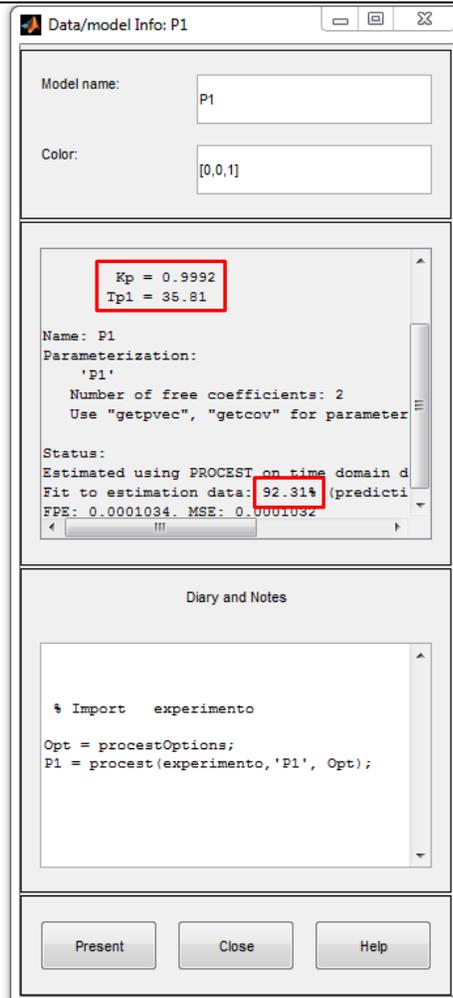


Figura 2.19: Resultado de K, TP1 y el coeficiente de correlación.

## 4. VALIDACIÓN

### a. Resultados

Sin embargo se recomienda volver al menú principal de System Identification, seleccionar P1 y seleccionar Model output, hacemos la validación y visualmente Matlab arroja la comparación donde se ven dos señales, color negro es la señal del experimento y la señal azul es la señal del sistema de primer orden y nos dice que el ajuste es de **92.31**, como se observa en la figura 2.20.

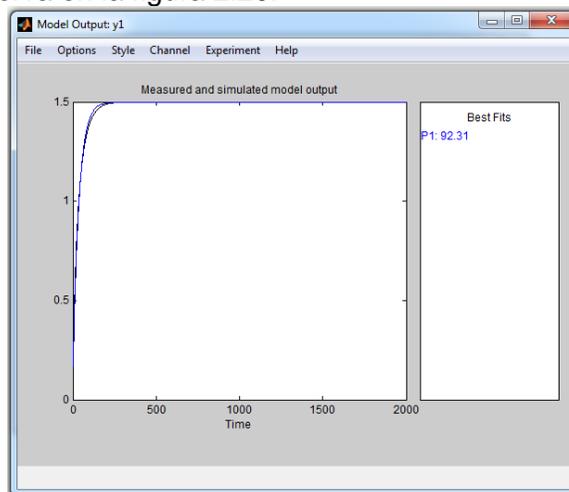


Figura 2.20: Validación del experimento.

Lo anterior se realiza con una estructura de sistema continuo, a continuación se muestra con una estructura en tiempo discreto, se debe tener en cuenta que se debe dejar todo tal como está, ya que las modificaciones se empiezan a hacer en la SELECCIÓN DE LA ESTRUCTURA.

## 5. SELECCIÓN DE LA ESTRUCTURA (DISCRETA DE PRIMER ORDEN)

### a. Explicación General de Toolbox

Para tiempo Discreto se selecciona la opción **Transfer Functions Models** como se ve en la figura 2.21.

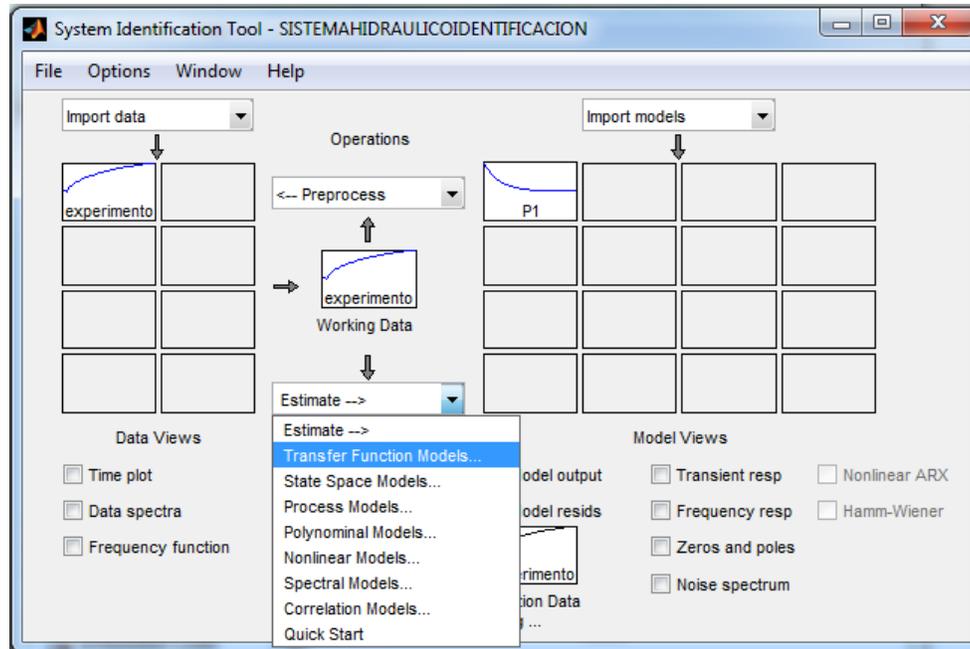
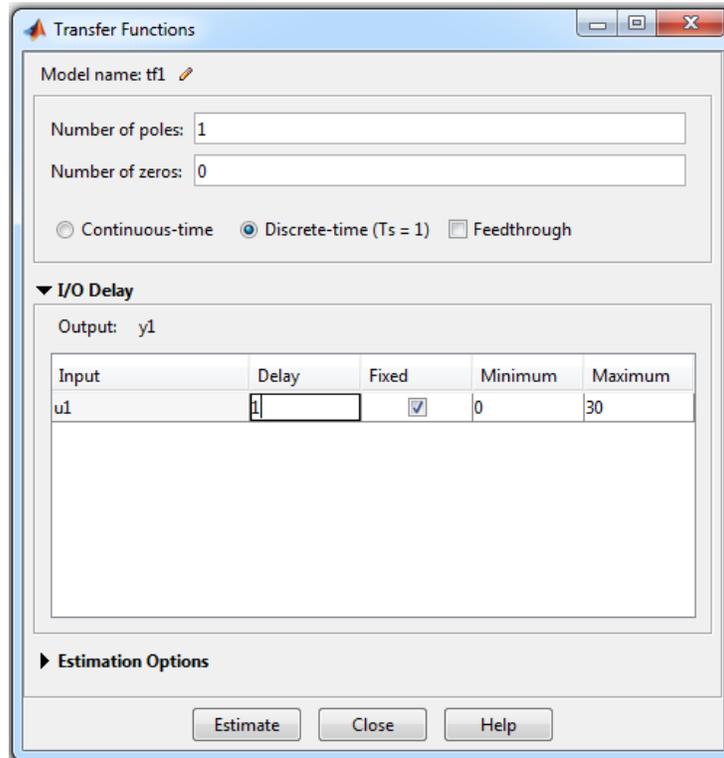


Figura 2.21: SELECCIÓN ESTRUCTURA – Sistema Discreto.

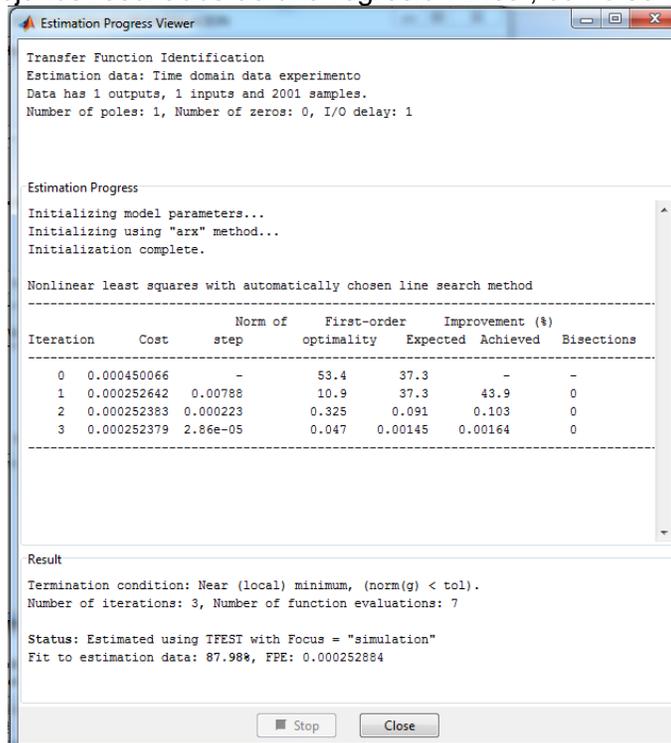
De inmediato se despliega una nueva ventana como se ve en la figura 2.22, donde podemos configurar el sistema, agregar el número de polos y el número de ceros que debe tener la función de transferencia, se seleccionar **Discrete-time ( $T_s = 1$ ) (Esto depende del tiempo de muestro)**, se despliega **I/O DELAY** Para poder escoger el retardo del sistema.

El retardo del sistema se escoge teniendo en cuenta la figura 2.14 donde se ven las diferentes respuestas del sistema al inyectarle energía, se debe de verificar al inyectarle una señal paso al sistema que este no tenga retardo o si lo tiene. Se debe tener cuidado a la hora de hacer identificación este no reconoce retardos, se debe de agregar el retardo en I/O DELAY en caso de tener retardo en el sistema.

Como en este sistema no hay retardo, quiere decir que el sistema continuo no hay retardo pero en el sistema discreto si, por lo menos hay un retardo, todo sistema discreto solamente responde en la siguiente muestra, no responde inmediatamente. Entonces en Delay colocamos 1, como se ve en la figura 2.22, para configurar un sistema de primer orden discreto debemos colocar en número de polos = 1 y en número de zeros = 0, y le damos Estimate.

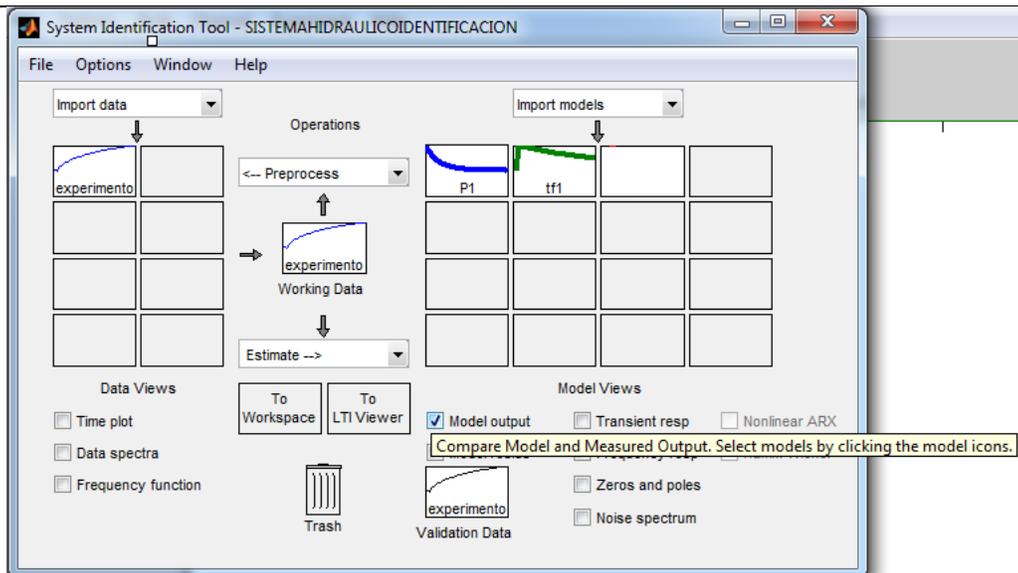


**Figura 2.22:** Configuración Transfer Functions – I/O DELAY (Retardo del sistema)  
Al hacer clic en Estimate, arroja los resultados de una regresión lineal, como se ve en la figura 2.23.

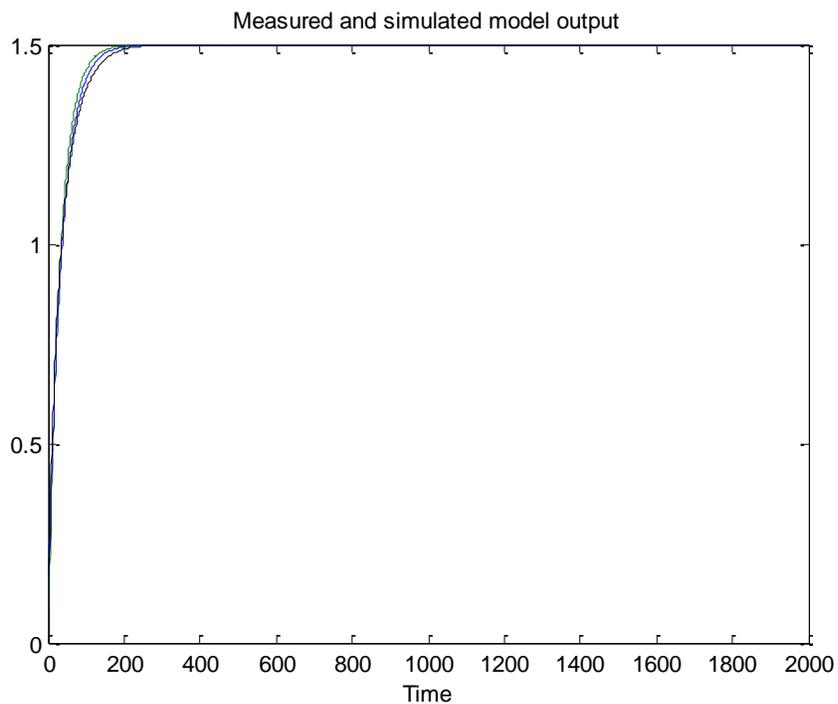


**Figura 2.23.** Regresión lineal generada por Transfer Functions.

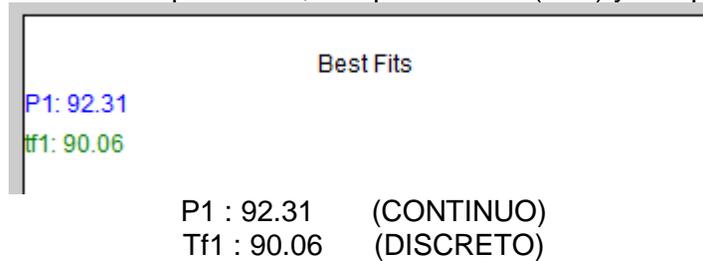
Seleccionar las dos señales P1 y tf1 y la opción de **Model Output**, como se muestra en la figura 2.24 y se obtiene la validación del experimento (señal negra), el continuo (señal azul), y el discreto (señal verde), se puede observar que en este caso el ajuste da mejor en el continuo, Ver figura 2.25 y 2.26.



**Figura 2.24:** Selección de señales y de Model Output para realizar la comparación de los sistemas y el experimento.



**Figura 2.25.** Resultado del experimento, tiempo continuo (azul) y tiempo discreto (verde).



**Figura 2.26:** Resultados del tiempo continuo y discreto en el experimento

## 6. SELECCIÓN DEL MEJOR MODELO.

Al aumentar la complejidad del sistema, se aumenta el orden de los polinomios en la función de transferencia, teniendo en cuenta la forma del sistema continuo y la del sistema discreto se realiza el mismo procedimiento que se muestra en la figura 2.22, pero se realiza el siguiente cambio para aumentar el orden del polinomio:

Número de polos: 1

Número de ceros: 1

Delay: 1

Siempre el polinomio del denominador debe ser mayor o igual al del denominador de la función de transferencia. La configuración del sistema discreto con las anteriores características se ven en la figura 2.27.

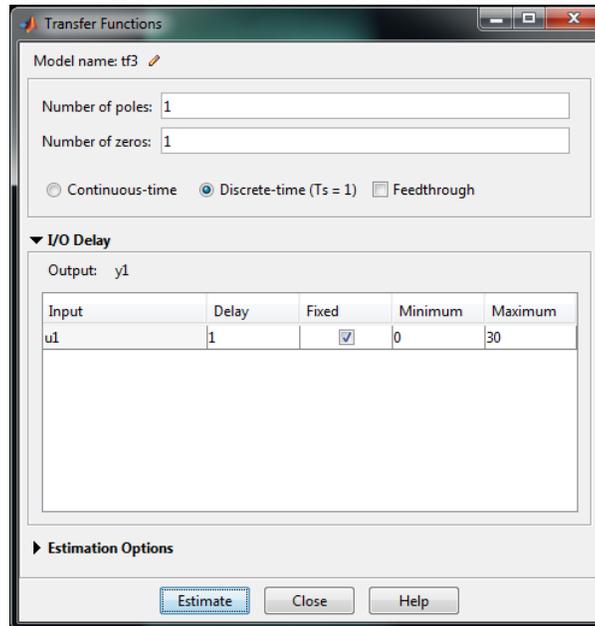


Figura 2.27: Configuración de Polos, Ceros y Retraso de FT.

Al volver a la ventana principal de System Identification, se puede observar que genera una nueva señal con el nombre de tf2 de color rojo, como se ve en la figura 2.28, de igual modo procedemos a ejecutar la comparación de los tres modelos que nos arroja los datos y la gráfica mostradas en la figura 2.29.

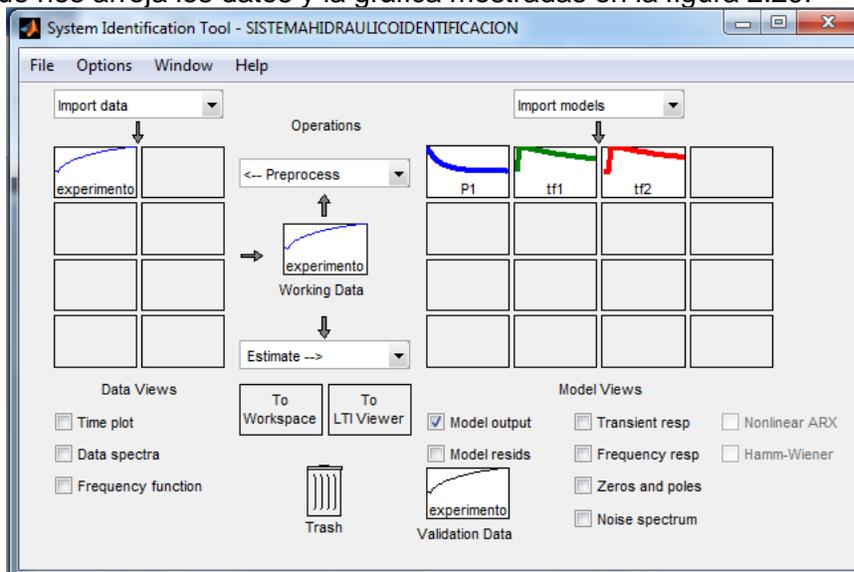
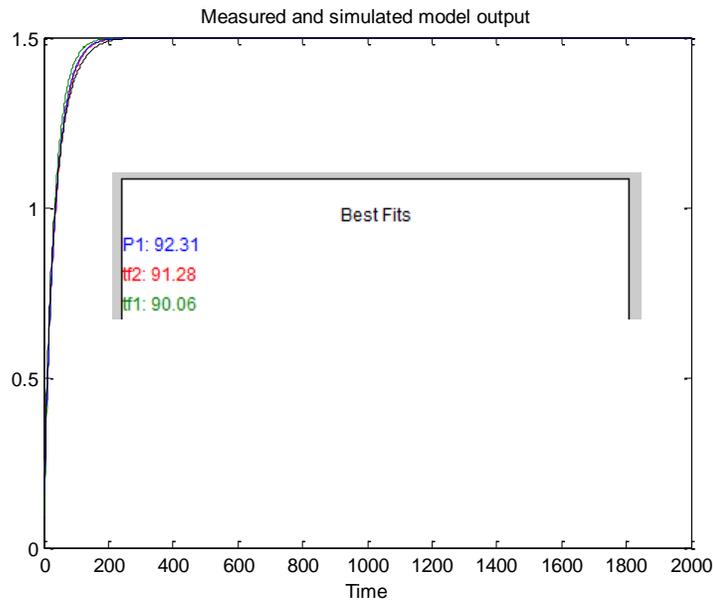


Figura 2.28: Selección de señales y de Model Output para realizar la comparación de P1, tf1, tf2 y el experimento.



**Figura 2.29:** Resultado del experimento, tiempo continuo (azul) y tiempo discreto (verde) y tiempo discreto con un polo y un cero (roja).

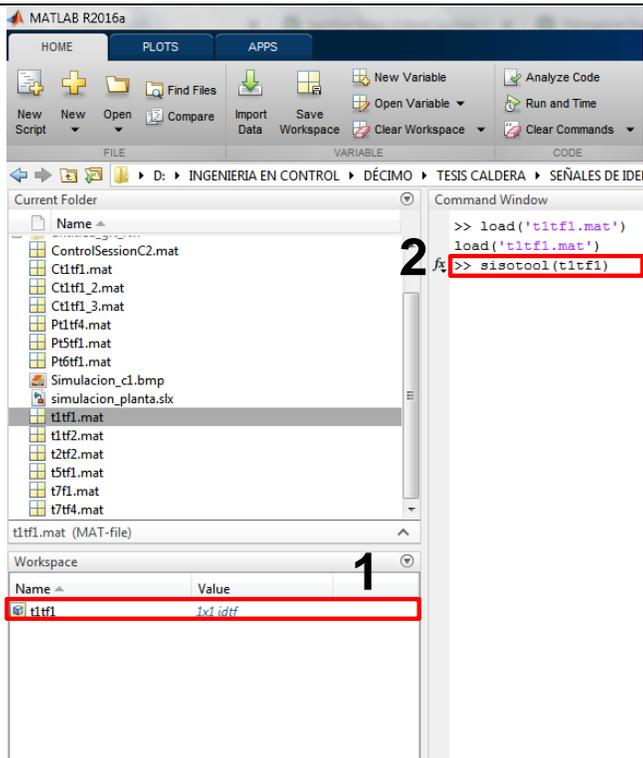
Donde se observa un nuevo sistema tf2, entre 90.06 y 91.28, no hay mucha distancia matemáticamente, entonces la forma de elegir el modelo, es escoger el sistema con complejidad más pequeña pero que tenga una aproximación más alta.

Como conclusión con estos tres sistemas, si se debe elegir alguno de los tres, la mejor elección es tf1, aunque no es la más alta aproximación es más sencilla de realizar ya que no tiene ceros en la función de transferencia.

## 7. DISEÑO DEL CONTROLADOR.

Para el diseño del controlador se utiliza la herramienta de SISOTOOL de Matlab, en donde se debe de cargar la función de transferencia del modelo aproximado que resultado de la planta generado por System Identification, para ello se debe de exportar el mejor ajuste desde System Identification a el Workspace de Matlab, como se ve en la figura 2.30.

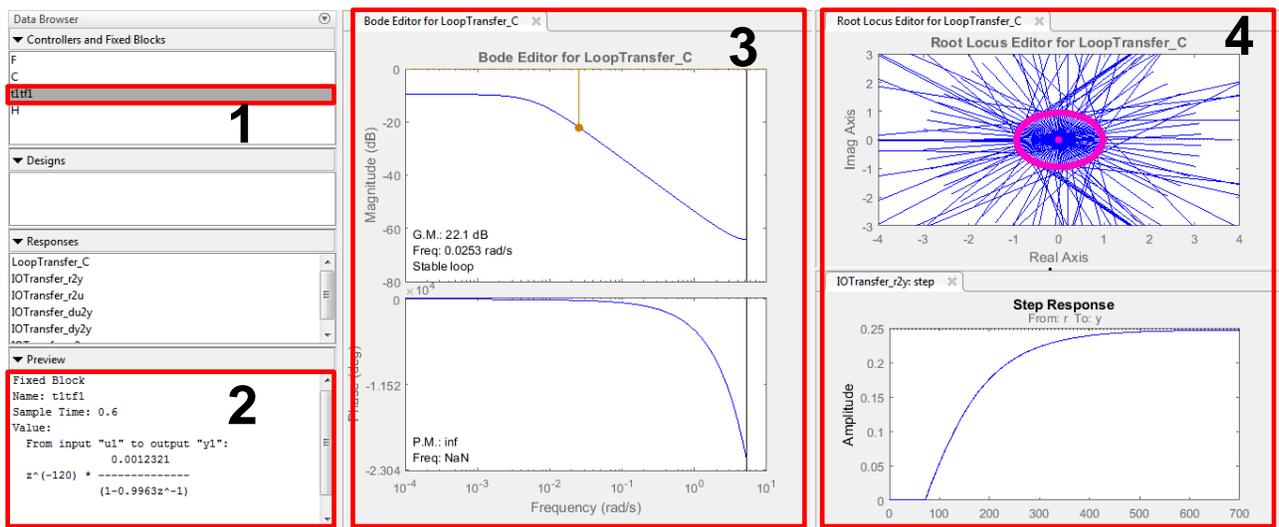
1. Así aparece la función de transferencia con mejor ajuste al modelo del comportamiento de la planta.
2. En el Command Window de Matlab, se invoca la herramienta de Sisotool con la función de transferencia de la planta.



**Figura 2.30:** Función de Transferencia En el Workspace y herramienta de Sisotool.

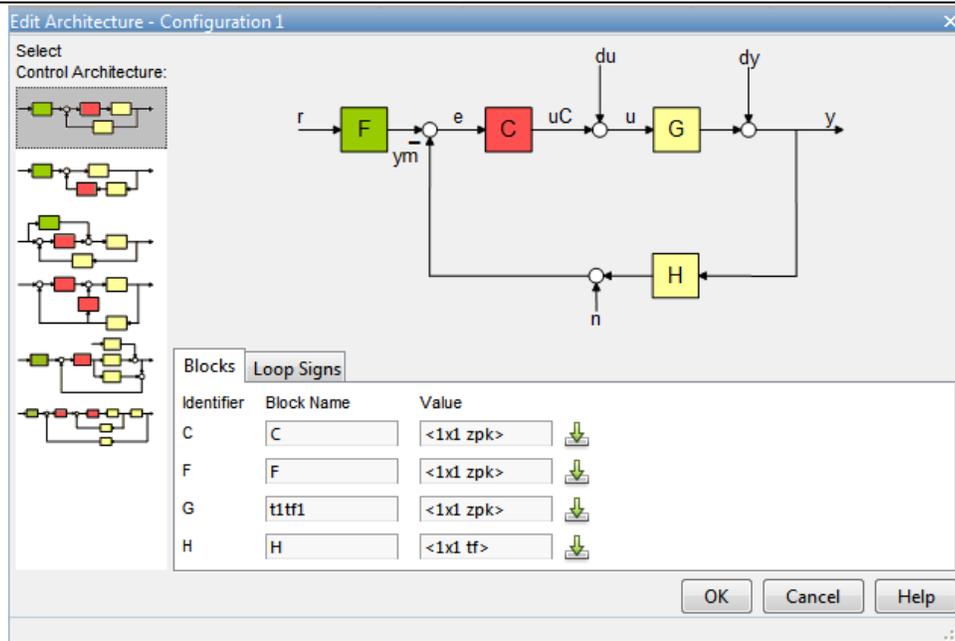
Se debe de esperar un tiempo mientras carga la herramienta con la función de transferencia, luego aparece la ventana donde se puede ver lo siguiente, ver figura 2.31.

1. Los bloques de la Arquitectura del controlador en lazo cerrado.
2. La vista previa de la función, en este caso de la función de transferencia de la planta.
3. El diagrama de bode de la planta con el controlador igual a 1.
4. El lugar de las raíces, y la respuesta al paso de la planta.



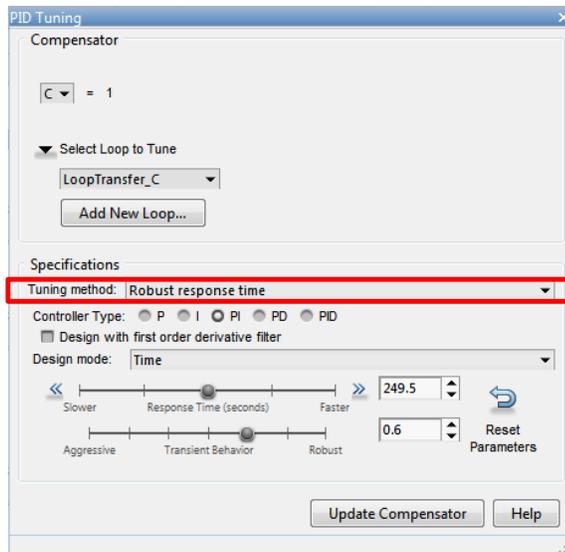
**Figura 2.31:** Ventana principal de Sisotool.

Cuando ya se tiene cargada la función de transferencia se debe escoger la arquitectura del lazo cerrado que aparece en los iconos de la parte superior de la herramienta de Sisotool; Aparece la siguiente ventana, donde se escoge la Arquitectura, ver figura 2.32.

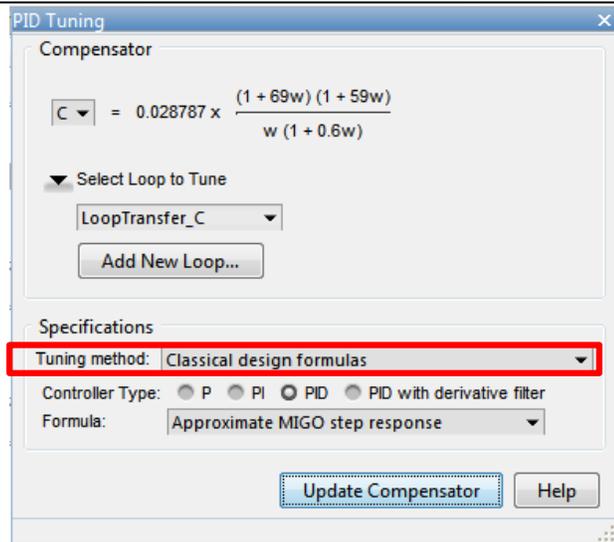


**Figura 2.32:** Configuración de la Arquitectura en lazo cerrado.

Se tienen varias maneras para diseñar el controlador, se debe escoger la que mejor se desempeñe el estudiante, para este caso se realiza por PID Tuning, donde se ajusta manualmente la velocidad del controlador y la robustez del mismo, ver figura 2.33 y 2.34.



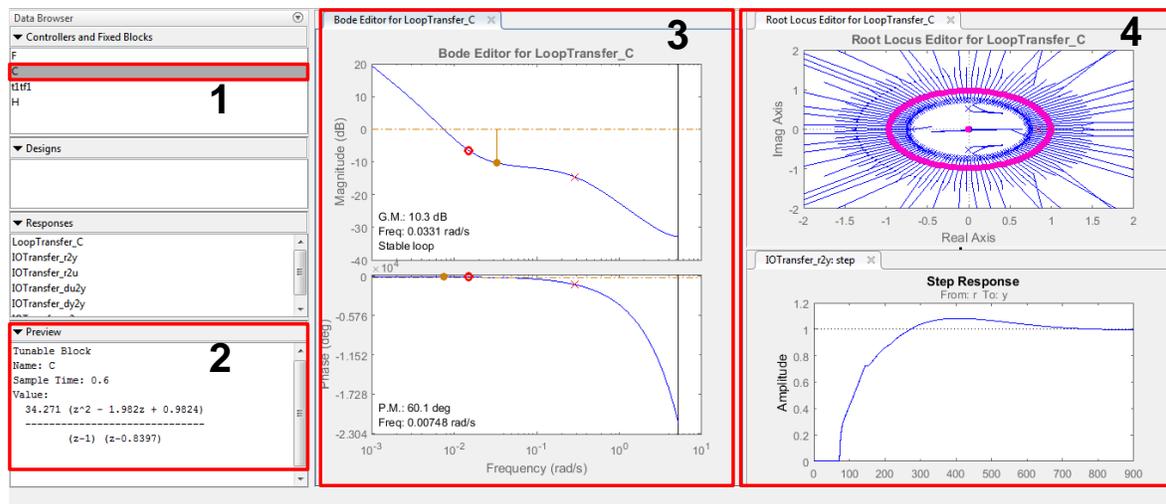
**Figura 2.33:** PID Tuning- Método Sintonización- Respuesta al tiempo robusta.



**Figura 2.34:** PID Tuning- Método Sintonización- Diseño con fórmulas clásicas.

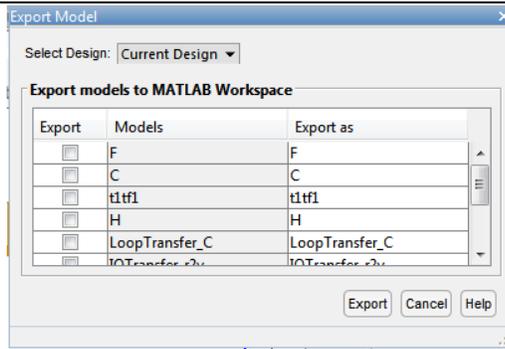
Al sintonizar el controlador con los parámetros establecidos como el Máximo sobre impulso menor al 10%, o para este caso sin sobre impulso por que no se ha implementado un sistema para bajar la temperatura, que no exista error de estado estacionario, y que el tiempo de estabilización sea menor o igual al de la planta en lazo abierto. Al final de toda la sintonización en la ventana principal de la herramienta de Sisotool aparece lo siguiente, ver figura 2.35.

1. Los bloques de la Arquitectura del controlador en lazo cerrado.
2. La vista previa de la función, en este caso de la función del controlador en tiempo discreto
3. El diagrama de bode de la planta con el controlador sintonizado.
4. El lugar de las raíces, y la respuesta al paso de la planta en lazo cerrado



**Figura 2.35:** Sisotool con la sintonización del controlador.

Luego de ello se debe exportar la función del controlador al Workspace de Matlab, ver figura 2.36, guardar la información en alguna parte, para luego realizar el procedimiento de transformación de la función de tiempo discreto a Ecuaciones en diferencia e implementar el pseudocódigo en la plataforma de CODESYS.



**Figura 2.35:** Exportar datos del controlador a Workspace de Matlab.

Los pasos anteriores se pueden realizar de igual forma para el diseño del controlador e implementación en RSLogix 5000 de Rockwell.

Al final de realizar las operaciones matemáticas para llegar a las ecuaciones en diferencias se implementan en un programa de CODESYS, con tiempo de muestreo para este caso de 600ms.

```

Q0:=1.9675      ; //KP*(1+(TS/TI)+ (TD/(N+TS)));
Q1:=0.9675     ; //KP*(1+(N/(N+TS))+(TS*N/(TI*(N+TS))+(2*TD/(N+TS))));
Q2:=6.6577    ; //KP*(N*TD/(N+TS));
Q3:=13.18024;  //-(1+(N/(N+TS)));
Q4:=6.2999    ; //(N/(N+TS));
M1:=0.0;      //U[n]
M2:=0.0;      //U[n-1]
M3:=0.0;      //U[n-2]
M4:=0.0;      //e[n]
M5:=0.0;      //e[n-1]
M6:=0.0;      //e[n-2]

```

```

IF GVL0.PID_FLAG1 = 1 THEN
  GVL0.PID_FLAG2:=0;
  GVL0.ERROR:= GVL0.REF_TEMP-GVL0.Temp_Dos;
  M4:=GVL0.ERROR;
// /*CONTROLADOR DISCRETO*/
M1:= Q0*M2-Q1*M3+Q2*M4-Q3*M5+Q4*M6; //Ecuación en diferencias asignada en memorias.
M5:= M4;
M6:= M5;
M2:= M1;
M3:= M2;
GVL0.ANGULO:= LREAL_TO_REAL(M1);
IF GVL0.ANGULO<= 0.00 THEN
  GVL0.ANGULO:=0.00;
END_IF
IF GVL0.ANGULO >= 1690.00 THEN
  GVL0.ANGULO:= 1600.00;
END_IF
END_IF

```