



industriales
etsii

Escuela Técnica
Superior
de Ingeniería
Industrial

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería
Industrial

**DISEÑO DE UN SISTEMA DE COMUNICACIONES
INDUSTRIALES PARA EL CONTROL Y MONITORIZACIÓN DE
UN PROCESO INDUSTRIAL BÁSICO.**

TRABAJO FIN DE ESTUDIO

**MÁSTER UNIVERSITARIO EN SISTEMAS ELECTRÓNICOS
E INSTRUMENTACIÓN**

Autor: Eddy Emil Darley Peña
Director: Ginés Doménech Asensi
Codirector: Juan de la Cruz Martínez Alajarín

Cartagena, 17 de Septiembre de 2018



Universidad
Politécnica
de Cartagena

RESUMEN

Este trabajo está orientado a la parte de automatización y comunicación Industrial del Máster Sistemas Electrónicos e Instrumentación, donde se ha diseñado un sistema que monitoriza y controla el pH y temperatura de los efluentes de una planta de tratamiento de agua residual industrial. Para lograr este proyecto de control se ha tenido que utilizar cuatro PLC SIMATIC S7 serie 1200 y el software TIA PORTAL, software utilizado para la programación de los PLCs. Las variables medidas serán transmitidas por los buses de comunicación PROFIBUS DP y MODBUS. Estas variables serán simuladas por medio de botoneras ON-OFF y fuente de voltaje de 0 a 10 voltios conectadas a la entrada digital y analógica de cada PLC. El resultado obtenido de estas variables se visualizará en una página Web.

ABSTRACT

This work is oriented to the Industrial automation and communication part of the Master's Degree in Electronic Systems and Instrumentation, where a system has been designed that monitors and controls the pH and temperature of the effluents of an industrial wastewater treatment plant. To achieve this control project, four SIMATIC S7 1200 series PLCs and the TIA PORTAL software, used for programming the PLCs, had to be used. The measured variables will be transmitted by the PROFIBUS DP and MODBUS communication buses. These variables will be simulated by means of ON-OFF pushbutton panels and a voltage source of 0 to 10 volts connected to the digital and analog inputs of each PLC. The result obtained from these variables will be displayed on a Web page.

CONTENIDO

CAPÍTULO 1 DESCRIPCIÓN DEL SISTEMA.....	8
1.1 INTRODUCCIÓN	9
1.2 OBJETIVOS.....	10
1.3 FUNCIONAMIENTO Y COMPONENTES DEL SISTEMA	11
1.3.1 DEPOSITO 1	11
1.3.2 DEPOSITO 2	12
1.3.3 DEPOSITO 3	12
1.3.4 PLC MASTER	13
CAPÍTULO 2 PROTOCOLOS Y BUSES DE COMUNICACIONES	14
2 HISTORIA DE LAS REDES DE COMUNICACIÓN INDUSTRIAL	15
2.1 RED DE COMUNICACIÓN INDUSTRIAL.....	15
2.2 PROFIBUS	16
2.2.1 QUE ES PROFIBUS?.....	16
2.2.2 TIPOS DE PROFIBUS	16
2.2.3 MÓDULOS DE COMUNICACIÓN CM 1242-5 y CM143-5	17
2.3 CONEXIÓN DE PROFIBUS DP	19
2.3.1 TERMINALES	19
2.3.2 CABLE	19
2.4 FUNCIONAMIENTO	20
2.5 MODBUS	21
2.5.1 TIPO DE TRANSMISIÓN EN MODBUS	21
2.5.2 PROTOCOLO MODBUS-RTU	21
2.5.3 MODBUS ASCII.....	22
2.5.4 ESTRUCTURA DE LOS MENSAJES MODBUS RTU	22
2.5.5 MODULO DE TERMINALES MODBUS RTU.....	23
CAPÍTULO 3 CONTROLADOR PROGRAMABLE S7-1200 Y SOFTWARE UTILIZADO EN LA PROGRAMACIÓN.....	25
3 INTRODUCCIÓN AL PLC S7-1200.....	26
3.1 TIPOS DE LENGUAJE	27
3.2 PROGRAMACIÓN	28
3.3 SOFTWARE DE PROGRAMACIÓN	29
3.4 SIMBOLOGÍA DEL SOFTWARE TIA PORTAL	29
3.5 PROGRAMACIÓN DE PAGINA WEB.	31
CAPÍTULO 4 DESCRIPCIÓN DEL SISTEMA.....	35
4 BREVE DESCRIPCIÓN TRATAMIENTO DE AGUAS RESIDUALES	36

4.1 SENSORES DE PROXIMIDAD CAPACITIVO PARA DETECTAR EL NIVEL MODELO CA30CLN12M.	36
4.2 SENSOR PARA MEDICION DE pH	38
4.3 SENSOR PARA MEDICION DE TEMPERATURA	39
4.4 DESCRIPCIÓN DEL PROCESO POR MÉTODO GRAFCET	40
4.4.1 DIAGRAMA GRAFCET DEPOSITO 1	41
4.4.2 DIAGRAMA GRAFCET DEPOSITO 2	41
4.4.3 DIAGRAMA GRAFCET DEPOSITO 3	43
CAPÍTULO 5 CODIGO DE PROGRAMACION EN TIA PORTAL	44
5- CONFIGURACIÓN EN TIA PORTAL	45
5.1 SEGMENTOS DE PROGRAMACION AUTÓMATA DEPOSITO 1	47
5.2 SEGMENTOS DE PROGRAMACION PLC DEL DEPOSITO 2	49
5.3 SEGMENTOS DE PROGRAMACION AUTÓMATA DEPOSITO 3	58
5.4 SEGMENTOS DE PROGRAMACION DEL PLC CENTRAL	60
5.5 INTERFAZ GRAFICA	65
CONCLUSIONES	72
BIBLIOGRAFÍA	73

LISTA DE FIGURAS

Figura 1. Esquema de la colocación de los PLCs a la instalación a controlar.....	11
Figura 2. Modulo Maestro y esclavo en comunicación PROFIBUS. [5].....	17
Figura 3. Terminadores. [7].....	19
Figura 4. Cable utilizado en la comunicación PROFIBUS. [7]	20
Figura 5. Estructura de los mensajes [6].....	23
Figura 6. Módulo CB 1241-85 instalado en la parte frontal de un PLC SIMATIC S7-1200.	23
Figura 7. Conexiones del módulo de comunicaciones CB 1241-85. [6].....	24
Figura 8. Esquema de la conexión de buses de cada PLC.	24
Figura 9. Módulo Maestro y esclavo en comunicación PROFIBUS. [6].....	26
Figura 10. Esquema comparativo entre SCL, FUP y KOP.....	28
Figura 11. Simbología utilizada TIA portal.	29
Figura 12. Bloques de contadores CTU, CTD, CTUD. [6]	29
Figura 13. Representación de Instrucción In_Range. [6]	31
Figura 14. <i>Detector nivel capacitivo Carlo Gavazzi.</i>	36
Figura 15. <i>Conexionado eléctrico del detector nivel capacitivo Carlo Gavazzi.</i>	37
Figura 16. Diagrama GRAFCET del depósito 1.	41
Figura 17. Diagrama GRAFCET del depósito 2.	42
Figura 18. Diagrama GRAFCET del depósito 3.	43
Figura 19. Ventana de inicio software TIA PORTAL	45
Figura 20. Ventana de selección del autómeta.....	45
Figura 21. Ventana principal del para programación del autómeta.....	46
Figura 22. Ventada de redes de los dispositivos.....	46
Figura 23. Segmento 1 de programación 1 del autómeta.	47
Figura 24. Segmento 2 de programación 1 del autómeta.	47
Figura 25. Segmento 3 de programación 1 del autómeta.	48
Figura 26. Segmento 4 de programación 1 del autómeta.	48
Figura 27. Segmento 5 de programación 1 del autómeta.	49
Figura 28. Segmento 5 de programación 1 del autómeta.	49
Figura 29. Segmento de programación 1 del autómeta 2.	49
Figura 30. Segmento de programación 2 del autómeta 2.	50
Figura 31. Segmento de programación 3 del autómeta 2.	50
Figura 32. Segmento de programación 4 del autómeta 2.	51
Figura 33. Segmento de programación 4.1 del autómeta 2.	51
Figura 34. Segmento de programación 4.2 del autómeta 2	51
Figura 35. Segmento de programación 5 del autómeta 2	51
Figura 36. Segmento de programación 5.1 del autómeta 2.	52
Figura 37. Segmento de programación 5.2 del autómeta 2	52
Figura 38. Segmento de programación 5.3 del autómeta 2	52
Figura 39. Segmento de programación 5.4 del autómeta 2	53
Figura 40. Segmento de programación 5.5 del autómeta 2.	53
Figura 41. Segmento de programación 6 del autómeta 2.	53
Figura 42. Segmento de programación 6.1 del autómeta 2.	54
Figura 43. Segmento de programación 6.2 del autómeta 2.	54
Figura 44. Segmento de programación 6.2 del autómeta 2.	54

Figura 45. Segmento de programación 7 del autómata 2.	55
Figura 46. Segmento de programación 7.1 del autómata 2.	55
Figura 47. Segmento de programación 7.2 del autómata 2.	55
Figura 48. Segmento de programación 7.3 del autómata 2.	56
Figura 49. Segmento de programación 8 del autómata 2.	56
Figura 50. Segmento de programación 9 del autómata 2.	56
Figura 51. Segmento de programación 10 del autómata 2.	57
Figura 52. Segmento de programación 11 del autómata 2.	57
Figura 53. Segmento de programación 12 del autómata 2.	57
Figura 54. Segmento de programación 13 del autómata 2.	57
Figura 55. Segmento de programación 15 del autómata 2.	58
Figura 56. Segmento de programación 1 del autómata 3.	58
Figura 57. Segmento de programación 1.1 del autómata 3.	58
Figura 58. Segmento de programación 1.2 del autómata 3.	59
Figura 59. Segmento de programación 2 del autómata 3.	59
Figura 60. Segmento de programación 3 del autómata 3.	59
Figura 61. Segmento de programación 4 del autómata 3.	60
Figura 62. Segmento de programación 1 del autómata 4.	60
Figura 63. Segmento de programación 2 del autómata 4.	61
Figura 64. Segmento de programación 3 del autómata 4.	61
Figura 65. Segmento de programación 4 del autómata 4.	61
Figura 66. Segmento de programación 5 del autómata 4.	62
Figura 67. Segmento de programación 6 del autómata 4.	62
Figura 68. Segmento de programación 7 del autómata 4.	63
Figura 69. Segmento de programación 8 del autómata 4.	63
Figura 70. Segmento de programación 9 del autómata 4.	63
Figura 71. Segmento de programación 10 del autómata 4.	64
Figura 72. Segmento de programación 11 del autómata 4.	64
Figura 73. Segmento de programación 12 del autómata 4.	64
Figura 74. Segmento de programación 13 del autómata 4.	64
Figura 75. Segmento de programación 14 del autómata 4.	65
Figura 76. Interfaz gráfica proyecto, mostrando el pH neutro.	70
Figura 77. Interfaz gráfica proyecto, mostrando el pH ácido.	70
Figura 78. Interfaz gráfica proyecto, mostrando el pH alcalino.	71
Figura 79. Conexión de PLC's utilizados en el proyecto.	71

LISTA DE TABLAS

Tabla 1. Descripción de la distribución de los pines sobre RS 485. [6]	18
Tabla 2. Tabla característica Estándar PROFIBUS. [6]	18
Tabla 3. Formato de trama MODBUS RTU.	21
Tabla 4. Formato de trama MODBUS ASCII.	22
Tabla 5. Característica de CPU 1211. [6].....	27
Tabla 6 Tipo de datos para parámetros CTU, CTD, CTUD. [6].....	30
Tabla 7. funciones de comparación en diferentes tipos de lenguaje. [6].....	30
Tabla 8. Operadores lógicos TIA Portal. [6]	30
Tabla 9. Especificaciones detector de nivel.	37
Tabla 10. Especificaciones sensor pH.	38
Tabla 11. Especificaciones sensor pH.	39



industriales
etsii

**Escuela Técnica
Superior
de Ingeniería
Industrial**

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

CAPÍTULO 1

DESCRIPCIÓN DEL SISTEMA



**Universidad
Politécnica
de Cartagena**

1.1 INTRODUCCIÓN

Las empresas hoy en día tienen la responsabilidad de enfrentarse a un mundo globalizado que tiende a proteger cada día más el medioambiente debido a que el bienestar de nosotros los humanos se hace dependiente de las condiciones de nuestro mundo y su sistema.

En la actualidad es importante mantener una alta calidad en las aguas residuales que salen de las industrias, uno de los factores importante que se debe de corregir en estas aguas residuales es su pH, dicho pH puede ser alterado cuando estas aguas se han utilizado en el proceso de producción de algún producto, tanto como parte de la materia prima, así como refrigerante de máquina o limpieza de la misma. Cabe mencionar también que la temperatura del agua ya tratada para ser vertida al medioambiente no debe ser muy elevada, por eso se hace preciso un control para el enfriamiento del agua residual.

El presente proyecto responde las necesidades específica de las industrias, proponiendo alternativas de soluciones basadas en la automatización industrial de manera sencilla, con un fin para mejorar la comunicación en el proceso de tratamiento de aguas residuales industriales de una manera eficiente utilizando equipos y buses de comunicaciones adecuados.

La metodología utilizada en el desarrollo de este proyecto parte de la problemática a la hora de decidir que equipos y protocolos de comunicación utilizar valiéndose en análisis de situaciones actuales y proponiendo esta alternativa como solución sencilla a esta problemática.

Vale mencionar que este escrito no pretende ser un manual sobre como diseñar un sistema de planta de tratamiento de aguas residuales completo, sino más bien se enfoca en los diferentes protocolos de comunicación que podemos utilizar en un sistema de automatización industrial.

Sabiendo que la automatización se define como:

La Real Academia de las Ciencias Físicas y Exactas menciona a la “automatización como un conjunto de métodos y procedimientos para la sustitución del operador humano en las tareas físicas y mentales, desde otro punto de vista y manifestando una idea más clara se dice que la automatización es una herramienta de control que mediante señales de entrada permite influir directamente en un proceso que se está desarrollando con normalidad dentro de una línea productiva ,la automatización se encuentra inmersa en casi todas las áreas de una industria, por cuanto el objetivo fundamental es lograr la autonomía de cualquier tipo proceso para establecer una independencia total de sus funciones.

1.2 OBJETIVOS

El principal objetivo de este trabajo es el de diseñar un sistema de monitorización y control de pH, donde también se medirán nivel y temperatura, utilizando diferentes tipos de redes de comunicación o buses de campo que existen hoy en día. Estos datos se podrán visualizar mediante una web que servirá como interfaz hombre-máquina (HMI) utilizando el servidor web del PLC Siemens empleado. Dentro del objetivo general tenemos los siguientes objetivos particulares:

- Diseño e implementación de un sistema de control automático utilizando PLCs.
- Programación, diseño y realización del sistema simulando algunos componentes tales como válvulas, sensores, bombas, depósitos.
- Selección y utilización de diferentes buses de campo para para la comunicación entre PLCs, computador e interfaz gráfica.
- Diseño de una página web donde se podrá visualizar las variables medidas en tiempo real.
- Descripción general del PLC S7 1200, conexionado, módulos de comunicaciones, así como su correcta programación para un buen funcionamiento.

1.3 FUNCIONAMIENTO Y COMPONENTES DEL SISTEMA

Este proyecto está destinado para su aplicación en el control de pH, nivel, temperatura y caudal de un sistema de control de una planta de tratamiento de aguas residuales en la industria, compuesto por cuatro PLCs SIMATIC S7-1200 donde cada uno de estos autómatas tendrá a su cargo una función, que consistirá en monitorizar, controlar y enviar datos. Estos serán programados en el software TIA PORTAL. Habrá 5 depósitos, hay dos de los cuales estarán monitorizado con un controlador, y habrá otro PLC que monitorizara 3 depósitos, el último PLC servirá como nodo central donde llegaran los datos de las variables. En la Figura 1 se muestra un esquema de la instalación.

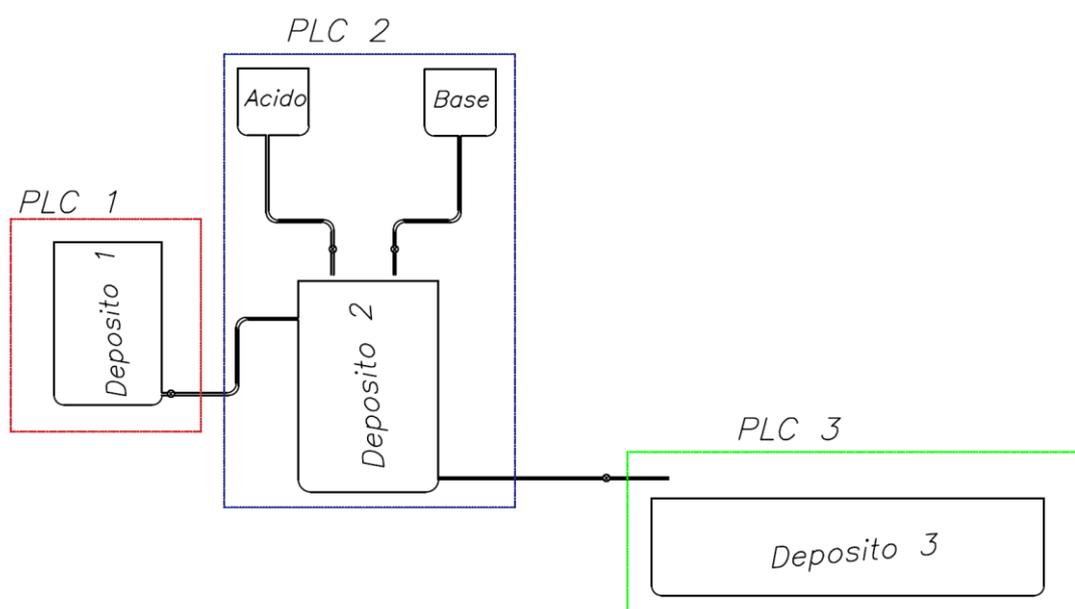


Figura 1. Esquema de la colocación de los PLCs a la instalación a controlar.

1.3.1 DEPOSITO 1

Este depósito se alimenta con el agua residual sin tratar, es decir, del agua usada en la industria para diferentes funciones en sus procesos. Su función es la de retener ésta agua hasta que pueda ser trasferida al depósito 2, en el que se realiza el tratamiento de pH.

El depósito 1 se ha modelado tal que los detectores de nivel se pueden sustituir por detectores físicos ON-OFF, capacitivo, etc. También se ha simulado el nivel de pH utilizando una de las entradas analógicas y escalándola con bloques de instrucciones que proporciona el mismo software TIA portal, este nos servirá para que de manera virtual podamos simular el nivel de este depósito. Este depósito tendrá dentro líquido con un pH diferente de 7 que es su neutralización. Hasta ahora se ha simulado con un depósito con una capacidad de almacenamiento de líquido de 100 litros.

Este depósito a su salida tiene una válvula para limitar el flujo al depósito número 2. Esta válvula se abrirá cuando en el depósito dos se necesiten más fluidos para ser neutralizado, es decir, cuando al detectarse que el nivel de líquido en el DEPOSITO dos está dentro del rango de la

lógica de programación ajustada se abrirá la válvula, estando esta válvula abierta hasta que el depósito 1 llegue al nivel mínimo detectado por el detector de nivel bajo ON-OFF o hasta que salga del rango permitido en la programación del DEPÓSITO 2.

El controlador se ha programado para que pueda dar una alarma cuando éste no esté funcionando correctamente dentro de los límites que se han establecidos mediante la programación.

Este controlador estará comunicando por medio del protocolo de comunicación PROFIBUS DP y será un esclavo dentro de esta red.

Nota: El caudal de salida de este depósito se supone constante e independiente de todo lo que pueda influir a no ser constante.

1.3.2 DEPÓSITO 2

El depósito número 2 es donde se hace la mezcla Ácido-Base para neutralizar el pH del líquido que ha llegado. Aquí llega el fluido del depósito 1, este fluido es posible que se encuentre con un líquido de igual pH o diferente y con un volumen diferente.

En este controlador se han hechos los cálculos mediante una programación para obtener el valor promedio del pH luego de que el flujo haya sido transferido desde el depósito número 1 y a la vez se determina el volumen final que tendría el depósito número 2. Para neutralizar el líquido de este depósito se han programado segmentos para controlar los depósitos de sustancias neutralizadoras ácido o base. Esta parte del programa se encargará de controlar el pH por encima de 7(70 indica valor escalado de esta variable) y otra parte el pH por debajo de dicho valor.

Los depósitos de sustancias ácido y de base se simularán mediante sendos bloques de funciones los cuales harán la función de añadirle ácido o base dependiendo de lo que necesite el líquido del depósito para ser neutralizado. En caso de que este proyecto sea hecho físicamente, estos depósitos estarían controlados de una manera más fácil, es decir, al tener el sensor de pH conectado en el depósito número 2. Este indicará el nivel de pH que tiene el líquido y mediante un arreglo de comparadores programados en TIA Portal este abrirá la válvula del neutralizador que le hace falta, es decir abrirá la válvula del depósito de ácido y del depósito de base.

Este depósito cuenta con un PLC que se encarga de monitorizar todo y a la vez hace los cálculos correspondiente, recibe variables de otros PLCs conectado a los depósitos a través del protocolo de comunicación de campo PROFIBUS DP. Éste actúa como el Maestro dentro de los PLCs conectados con el protocolo de comunicación PROFIBUS a los depósitos.

1.3.3 DEPÓSITO 3

Este depósito es donde se envía el fluido ya neutralizado y tiene como función dejar enfriar el agua con pH neutro antes de verterla al medioambiente. Utiliza la entrada analógica para simular la temperatura, de forma tal que esta entrada ya está adecuada, es decir, cableada y calibrada para conectarle algún sensor de temperatura tales como termopar o sensor TRD como PT100.

Esta variable con el valor de temperatura almacenada es enviada al PLC del depósito 2. Como se había mencionado antes el PLC está conectado mediante el protocolo de comunicación PROFIBUS DP y está identificado como un esclavo dentro de este modo de comunicación.

1.3.4 PLC MASTER

Este es el PLC central, encargado de almacenar todas las variables que le llegan del PLC 2 con el cual está conectado mediante el protocolo MODBUS RTU, que se explica en el capítulo 2. La función de este es hacerle una petición a su PLC esclavo ya establecido por el tipo de comunicación, cuando este PLC esclavo le envíe los datos solicitados este estará programado de forma tal, para subir estos datos para ser visualizados en una página web utilizada mediante el servidor web de Siemens. Esta página web servirá como interfaz gráfica.



industriales
etsii

**Escuela Técnica
Superior
de Ingeniería
Industrial**

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

CAPÍTULO 2

**PROTOCOLOS Y BUSES DE
COMUNICACIONES**



**Universidad
Politécnica
de Cartagena**

2 HISTORIA DE LAS REDES DE COMUNICACIÓN INDUSTRIAL

A finales de los años 60's los controles de proceso industriales productivos se efectuaban por medio de una red de control para cada una de las variables del sistema. En esa temporada existían los enormes paneles de control, los cuales utilizaban como indicador de algún suceso ocurrido en el proceso de producción luces de diferentes colores que equivaldrían al evento acontecido. Cabe mencionar también que en esa fecha estaban apareciendo los dispositivos basados en microprocesadores.

Luego en la década de los 70's se comenzó a implantar los ordenadores para controlar los procesos, principalmente para realizar labores de atención. Esto trajo consigo la sustitución de los enormes paneles de control. A finales de esta misma década, se desarrolló los microcontroladores, microprocesadores y los Controladores Lógicos Programables (PLC's) dando lugar a los controles distribuidos capaces de controlar uno o varios lazos del sistema y de comunicarse con los demás niveles de jerarquía.

La llegada de dispositivos programables y sensores inteligentes que benefician la automatización exigen su programación y control de forma remota, solo que para que esto suceda es necesario estar integrado en una red de comunicación.

Cabe mencionar que las primeras Redes Industriales entre controladores PLC's fueron de Modbus (MODICON). En el año 1982, Se creó un grupo de trabajo en Francia para conseguir que se trabajara con un bus industrial único y se crea la especificación FIP (Factory Instrumentation Protocol). En 1983 se inicia P-NET proveniente de Dinamarca. A mitad de los años 80's, surge el protocolo de comunicación Controller Area Network (CAN) de Bosch. En el año 1985 en Alemania Se forma el grupo Profibus

En la década de los 90's se tienen diversos protocolos de comunicación no compatibles. Los fundamentados en productos existentes o prototipos como: Bitbus (Intel), Hart (Rousemount), MIL1553B y otros como FIP y Profibus.

Hoy en día debido al surgimiento de conceptos de comunicación en el campo del control y supervisión de procesos, se está trabajado para la estandarización de la comunicación entre los dispositivos y se ha iniciado con el desarrollo de software de supervisión y gestión bajo el modelo de sistemas multiplataforma y softwares libres.

2.1 RED DE COMUNICACIÓN INDUSTRIAL

Las Comunicaciones Industriales se pueden definir como el espacio de la tecnología que estudia la transmisión de datos que llevan una información entre circuitos y sistemas electrónicos utilizados para llevar a cabo tareas de gestión y control del ciclo de vida de los productos industriales. En la década de 1980, las comunicaciones industriales comenzaron a realizarse mediante comunicaciones digitales punto a punto para, posteriormente, evolucionar hacia la aplicación de redes multipunto. La revolución de las comunicaciones industriales ha demostrado gran potencial en el perfeccionamiento de los sistemas de control de procesos y su aplicación contribución a la mejoría de sus servicios permitiendo realizar tareas como el control de maquinarias y la vigilancia de líneas del área de producción. [1]

Una red de comunicaciones industriales se puede definir como una red de tiempo real utilizada en un sistema de producción para conectar distintos procesos de aplicación con el propósito de asegurar la explotación de la instalación (comando, supervisión, mantenimiento y gestión). [1]

Otra definición corresponde a un sistema de comunicación que provee servicios bajo restricciones temporales y está constituido por protocolos capaces de gestionar estas restricciones [1]

Desde finales de los años ochenta se han propuesto muchos sistemas de comunicación industrial, para conectar los captadores y actuadores a los controladores, que sustituyeran al tradicional cableado hilo a hilo, estos sistemas se denominan buses de campo. [2]

En este proyecto se utilizaran diferentes redes de comunicaciones tales como: MODBUS RTU, PROFIBUS Y ETHERNET.

2.2 PROFIBUS

2.2.1 QUE ES PROFIBUS?

PROFIBUS es un estándar internacional de comunicaciones de bus de campo para la conexión de módulos de control de procesos y automatización de plantas que en lugar de tender cables individuales desde un controlador principal a cada sensor y actuador, se utiliza un único cable multipunto (u otro enlace de comunicaciones, por ejemplo, de fibra o inalámbrico) para conectar todos los dispositivos, con mensajería alta velocidad, bidireccional y en serie utilizada para las transferencias de información. [3]

PROFIBUS es un bus de campo accesible que cumple con todos los requisitos en un rango muy extenso de aplicaciones. Es además la norma de redes de comunicaciones preferida en el continente europeo y supone tener el mayor número de instalaciones operando bajo este protocolo en todo el mundo. Además de ser libre, no pertenece a ningún fabricante en exclusivo, está certificado y es un producto destinado a satisfacer todas las necesidades de la automatización y control de procesos en los próximos años. Se dice que esta red es libre, porque permite que los dispositivos de un gran número de fabricantes que estén certificados en este bus se comuniquen entre ellos sin la obligación de utilizar interfaces. Las principales normalizaciones provienen de los estándares europeos DIN 1924 y EN 50170.

2.2.2 TIPOS DE PROFIBUS

Existen tres tipos o versiones de PROFIBUS comúnmente conocidos como:

1. PROFIBUS DP
2. PROFIBUS PA
3. PROFIBUS FMS

He aquí una breve introducción a estos tres tipos.

- PROFIBUS DP

Corre sobre un cable blindado de dos núcleos con cubierta de color violeta y su velocidad varía de 9.6Kbps a 12Mbps. Se puede elegir una velocidad particular para que una red tenga tiempo suficiente para comunicarse con todos los dispositivos presentes en la red. Si los sistemas cambian lentamente, entonces una velocidad de comunicación más baja es adecuada y si los sistemas cambian rápidamente, entonces la comunicación efectiva se producirá a través de una velocidad más rápida. La transmisión balanceada RS485 que se utiliza en PROFIBUS DP sólo permite conectar 32 dispositivos a la vez, pero se pueden conectar más dispositivos y la red se puede ampliar con el uso de concentradores o repetidores.

- PROFIBUS PA

Es más lento que PROFIBUS DP y funciona a una velocidad fija de 31,2 Kbps a través de un cable blindado de dos hilos con cubierta azul. La comunicación puede iniciarse para minimizar el riesgo de explosión o para los sistemas que intrínsecamente necesitan equipos seguros. Los formatos de mensaje en PROFIBUS PA son idénticos a los de PROFIBUS DP.

- PROFIBUS FMS

Es el perfil de comunicación capaz de maniobrar todas las tareas intensivas de transmisión de datos muy comunes en las comunicaciones industriales, por lo que se le considera la solución universal para la transmisión de información en el nivel superior y de campo del modelo jerárquico de automatización.

Los potentes servicios FMS abren un extenso rango de aplicaciones y suministran de gran flexibilidad. Este puede ser también situado para tarea de comunicación extensa y compleja. [4]

2.2.3 MÓDULOS DE COMUNICACIÓN CM 1242-5 y CM143-5

En el proyecto para la comunicación entre PLCs se utilizan módulos de comunicación CM 1243-5(Master) y CM 1242-5(esclavo).

El Modulo CM 1243-5 es un módulo de comunicación utilizado en PROFIBUS para conectar el PLC Simatic S7 como maestro DP. Este tiene la característica de que debe de conectársele la alimentación desde el PLC al cual estará conectado este módulo.

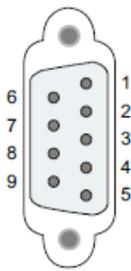
El Modulo CM 1242-5 presenta un encapsulado similar al de CM 1243-5, y es utilizado para conectar el PLC Simatic S7 como esclavo. A diferencia del módulo CM 1243-5 este no lleva alimentación externa.

La figura 2 muestra los módulos compactos CM 1243-5 y CM 1242-5 correspondientes al Maestro y Esclavo respectivamente.



Figura 2. Modulo Maestro y esclavo en comunicación PROFIBUS. [5]

Los módulos PROFIBUS disponen de conectores con el estándar RS485, cuyo esquema de conexiones se muestra en la tabla 1.



RS485	
Pin	Descripcion
1	No usado
2	No usado
3	RxD/TxD- línea datos B
4	RTS
5	DGND
6	Alimentación 5 V
7	No usado
8	RxD/TxD- línea de datos A
9	No usado

Tabla 1. Descripción de la distribución de los pines sobre RS 485. [6]

La siguiente tabla muestras algunos datos técnicos más importantes

DATOS TÉCNICOS DE PROFIBUS	
Estándar	PROFIBUS
Método de acceso	Paso por testigo con maestro-esclavo
Velocidad de transmisión	9.6kbit/s - 12 Mbit/s
Medio de transmisión	eléctrico: cable de dos hilos apantallado óptico: cables de FO(cristal y plástico) sin hilos: infrarrojos
Max No De nodos	127
tamaño de la red	Eléctrica: Max 9.6 km (dependiente de velocidad) óptica: 150 km(depends de Velocidad)
Topologías	Bus, anillo redundante
Aplicación	Comunicación de proceso, campo o datos

Tabla 2. Tabla característica Estándar PROFIBUS. [6]

PROFIBUS RS 485

- Transmisión asíncrona NRZ según RS485.
- Velocidad de 9.6 kBit/s hasta 12 Mbit/s.
- 32 estaciones por Segmento
- Distancia:
 - 12Mbit/s = 100m
 - 1.5 Mbit/s = 400m
 - para velocidad mayor a 187.5 kBit/s y distancia igual a 1000m.
- Posibilidad de incrementar la distancia hasta 10 km con repetidores.

2.3 CONEXIÓN DE PROFIBUS DP

2.3.1 TERMINALES

Estas son resistencias colocadas al final de la línea, una de sus funciones es atenuar la señal.

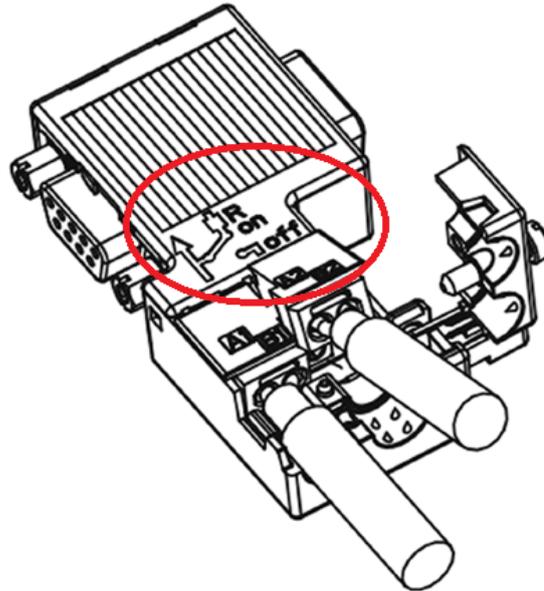


Figura 3. Terminadores. [7]

2.3.2 CABLE

La longitud máxima longitud del cable (apantallado y trenzado) depende de la velocidad a la que se transmiten los datos. Como en la mayoría de los proyectos de red, al inicio y al final de la línea de cada Segmento se debe conectar una resistencia terminadora. No está de más mencionar que el cableado del bus posee tanta importancia que alrededor del 90% de los fallos que se registran en una instalación que utiliza el protocolo de comunicación Profibus es ocasionado por la instalación defectuosa de terminales y cables.

Existen también conductores de fibra óptica, pero a diferencia de los otros estos son utilizados en ambientes donde se encuentre una gran interferencia electromagnética, asimismo en aplicaciones donde se necesita proporcionar un aislamiento eléctrico perfecto y en otros casos en que se necesita incrementar la distancia máxima del bus sin disminuir la velocidad de transmisión.

Cable par trenzado apantallado (STP):

- o Impedancia... 135 – 165 ohm
- o Capacitancia... < 30 pF/m
- o Resistencia... 110 ohm/km
- o Diámetro del núcleo... 0.64 mm
- o Sección... 0.34 mm²

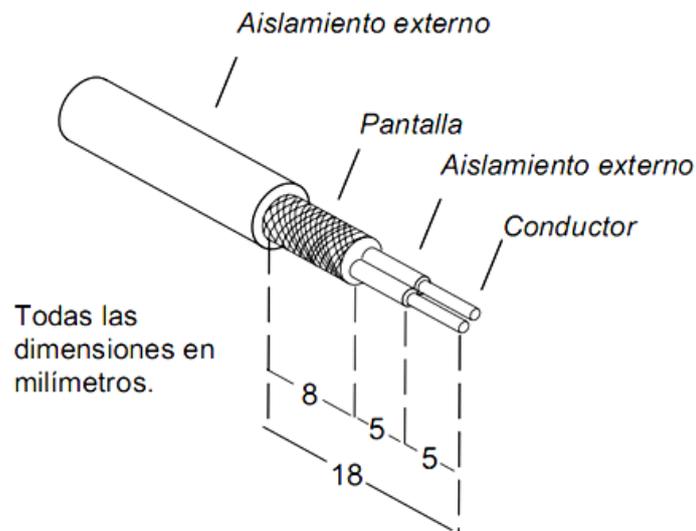


Figura 4. Cable utilizado en la comunicación PROFIBUS. [7]

2.4 FUNCIONAMIENTO

En el protocolo Profibus se establecen las características de comunicación de un sistema de bus de campo serie. Puede ser un sistema multimaestro que permite la operación conjunta de varios sistemas de automatización. Hay dos tipos de dispositivos que caracterizan a Profibus: Dispositivo Maestro y Dispositivo Esclavo, también llamados dispositivos activos y pasivos. Los dispositivos maestros, pueden enviar y solicitar datos a otras estaciones, siempre que mantengan el derecho de acceso (token) al bus. Los dispositivos esclavos sólo pueden enviar datos cuando un participante maestro se los ha solicitado.

Los dispositivos esclavos son periféricos, tales como dispositivos entrada/salida, islas de válvula, transductores de medida y en general equipos simples de campo. Por el contrario los dispositivos maestros suelen ser equipos inteligentes, como por ejemplo autómatas programables. Podemos tener así mismo integrados en estos sistemas elementos que se pueden programar para funcionar como maestros o como esclavos.

Profibus utiliza un método mixto para ordenar la comunicación entre estaciones. El método que utiliza para comunicarse entre una estación maestra y otra es del tipo token bus, mientras que la comunicación entre una estación maestra y una esclava es del tipo maestro-esclavo. [8]

2.5 MODBUS

Modbus es un protocolo de comunicación, fundamentado en la arquitectura maestro-esclavo o cliente-servidor, diseñado a finales de los años 70's por Modicon para sus productos de controladores lógicos programables (PLC's).

Debido a que este fue protocolo público, de fácil manejo y que requiere poco tratamiento, es decir, manipular bloques de datos sin muchas restricciones, se convirtió en uno de los protocolos de comunicaciones estándar más utilizados en las industrias. Es el protocolo de mayor disponibilidad para el conexionado de dispositivos electrónicos industriales. [9]

2.5.1 TIPO DE TRANSMISIÓN EN MODBUS

En una interface serial, los bits de datos son enviados de forma secuencial a través de un canal de comunicación o bus. Numerosos dispositivos utilizan comunicación serial para la transferir sus datos, incluyendo también las interfaces RS232 y RS485.

Las normas que definen los registros RS232 y RS485 no especifican el formato ni la secuencia de caracteres para la transmisión y recepción de datos. En consecuencia, para interface, es obligatorio identificar de igual forma el protocolo utilizado para la comunicación. Entre los distintos protocolos existentes, dos protocolos muy utilizados en la industria son Modbus ASCII y Modbus –RTU.

2.5.2 PROTOCOLO MODBUS-RTU

El protocolo Modbus fue desarrollado por Modicon parte de la empresa Schneider Automation. En el éste protocolo está especificado el formato de los mensajes utilizado por los elementos que forman parte de la red Modbus, los servicios o funciones que pueden ser ofrecidos a través de la red y, También cómo estos elementos intercambian datos en la red.

Éste caracteriza porque los bytes se remiten en su codificación binaria plana, sin ninguna conversión. Está originalmente diseñado para comunicaciones en bus serie. Como ventaja se tiene la mejora de velocidad de la transmisión de los datos, la buena disposición del canal de comunicación. Entre sus inconvenientes está que requiere una gestión de tiempos entre bytes recibidos para saber cuándo se inician y terminan las tramas.

TRAMA RTU

START	ADDRESS	FUNCTION	DATA	CRC CHECK	END
T1-T2-T3-T4	8 Bits	8 Bits	n*8 Bits	16 Bits	T1-T2-T3-T4

Tabla 3. Formato de trama MODBUS RTU.

Con la trama Modbus RTU, la especificación de la misma se realiza por intervalos de tiempo de caracteres de silencio. Un carácter de silencio tiene una duración de un byte de datos enviado por el medio, pero no traslada datos. Su duración (T) depende de la velocidad (Vt) y del número bits utilizados para su codificación (N):

$$T = \frac{N}{V_t}$$

La tecnología Modbus para velocidades que puedan alcanzar 19.200bps, el tiempo entre sus tramas debe estar alrededor de 3,5 veces la duración de un carácter, y para velocidades superiores es recomendable un tiempo fijo de 1,75ms. Por ejemplo, para una cantidad de 23 configuración del puerto serie con velocidad de 19.200bps, con un bit de parada y un bit de paridad (un total de 11 bits, sumando el de inicio y 8 de datos) se tiene que: $3,5 \cdot 11 / 19.200 = 2\text{ms}$.

La trama Modbus RTU lleva incorporado un código Cyclical Redundancy Check (CRC) de 16 bits cuya finalidad es detectar errores, que es calculado por el emisor a partir de todos los bytes de la trama enviados antes del CRC, a excepción de los delimitadores. Para ello se usa un algoritmo específico, definido en la especificación de Modbus serie. Luego el receptor vuelve a calcular el código de la misma forma que lo ha hecho el emisor para comprobar que el valor obtenido del cálculo es igual al valor actual en la trama y poder validar los datos. [2]

2.5.3 MODBUS ASCII

En ModBus ASCII Los datos se deben codificar con caracteres ASCII entre el "0" correspondiente a 16#30 y el "9" correspondiente a 16#39 y entre "A" correspondiente a 16#41 y "F" que corresponde a 16#46.

Ejemplo: Se quiere enviar un byte con un valor 16#FF, se debe enviar la el valor "FF", por lo que entonces se enviarán 16#46 y 16#46 que equivalen a dos bytes:. Asimismo se han utilizado tres caracteres especiales. El uno de ellos es el carácter ":" que corresponde a 16#3a, empleado para el inicio de la trama. Los protocolos Modbus RTU y ASCII están pensado para ser empleados principalmente sobre un medio físico asíncrono, como se puede ver en estos ejemplos EIA/TIARS-232, EIA/TIARS-485, o EIA-RS-422. [10]

TRAMA ASCII

START	ADDRESS	FUNCTION	DATA	LRC CHECK	END
1 CHAR					2 CHARS
:	2 CHARS	2 CHARS	N CHARS	2 CHARS	CRLS

Tabla 4. Formato de trama MODBUS ASCII.

2.5.4 ESTRUCTURA DE LOS MENSAJES MODBUS RTU

Debido a que en el proyecto se implementa una red con el protocolo MODBUS-RTU nos enfocaremos en este protocolo.

La comunicación Modbus-RTU aplica el sistema maestro-esclavo(Master-Slave) para el intercambiar de mensajes entre dispositivo conectado a la red que permite conectar hasta 247 esclavos y un único maestro. Esta comunicación comienza con el maestro solicitándole al esclavo, y luego este contesta al maestro lo que le fue pedido. En ambos los mensajes (pregunta y respuesta), utilizan la misma estructura que es: Dirección, Código de la Función y Datos. Solo el contenido de los datos tiene tamaño variable. A continuación se muestra en figura 5 un esquema de la estructura de datos utilizada en este protocolo. [11]

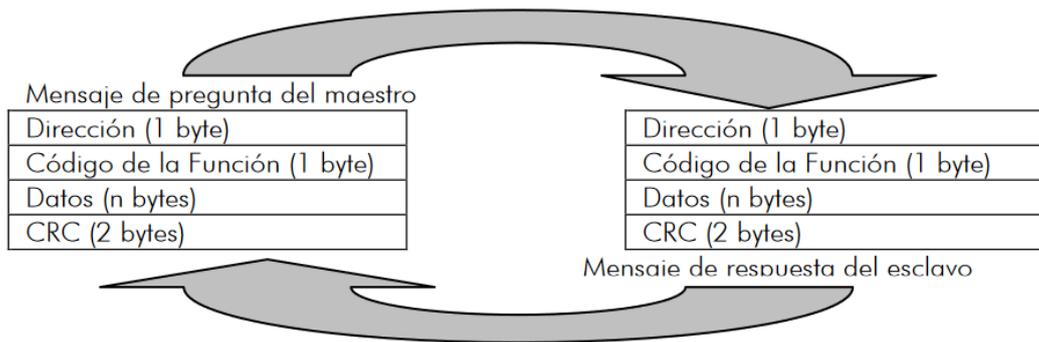


Figura 5. Estructura de los mensajes [6]

2.5.5 MODULO DE TERMINALES MODBUS RTU

En el proyecto se está utilizando el módulo de comunicación CB-1241-85. El PLC SIMATIC s7-1200 permite la comunicación punto a punto (PtP) mediante los módulos de Comunicación RS232 y RS485. Para esta última, en este proyecto se emplearán el módulo CB-1241 RS-485. Estos módulos pueden ser empleados para actuar tanto como Maestro como Esclavo, sin más que configurar el bloque de comunicaciones existente en la biblioteca MODBUS del software TIA Portal. Un Maestro MODBUS puede comunicarse con hasta 32 esclavos utilizando el protocolo RS485.

El módulo de comunicaciones es CB 1241-85. Va instalado la parte frontal del PLC SIMATIC S7-1200 y como se mencionó anteriormente es útil para actuar como maestro y como esclavo.

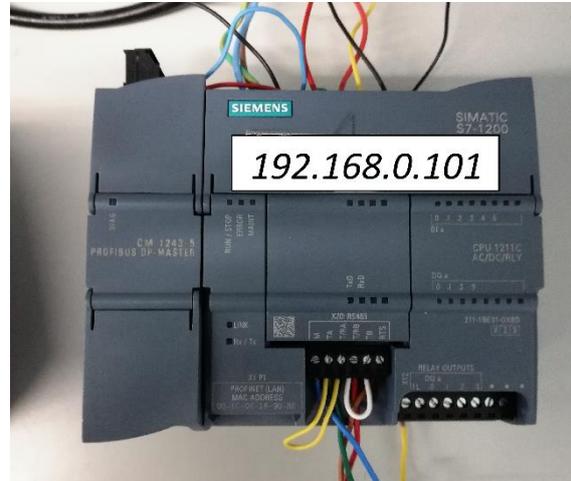


Figura 6. Módulo CB 1241-85 instalado en la parte frontal de un PLC SIMATIC S7-1200.

A continuación en la figura 7 se muestra como es el equivalente de los pines del módulo CB-1241-85 al conector RS485.

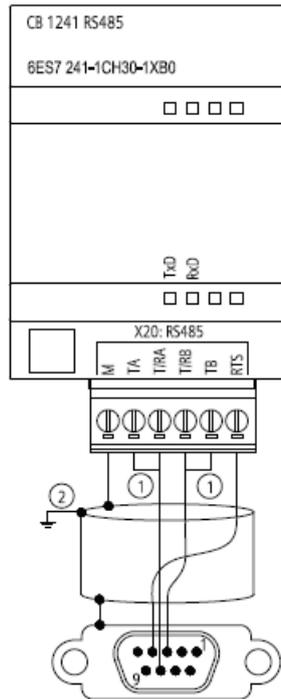


Figura 7. Conexiones del módulo de comunicaciones CB 1241-85. [6]

En la figura 8 se muestran los cómo van distribuidos los PLC y la redes de comunicación utilizadas por ellos.

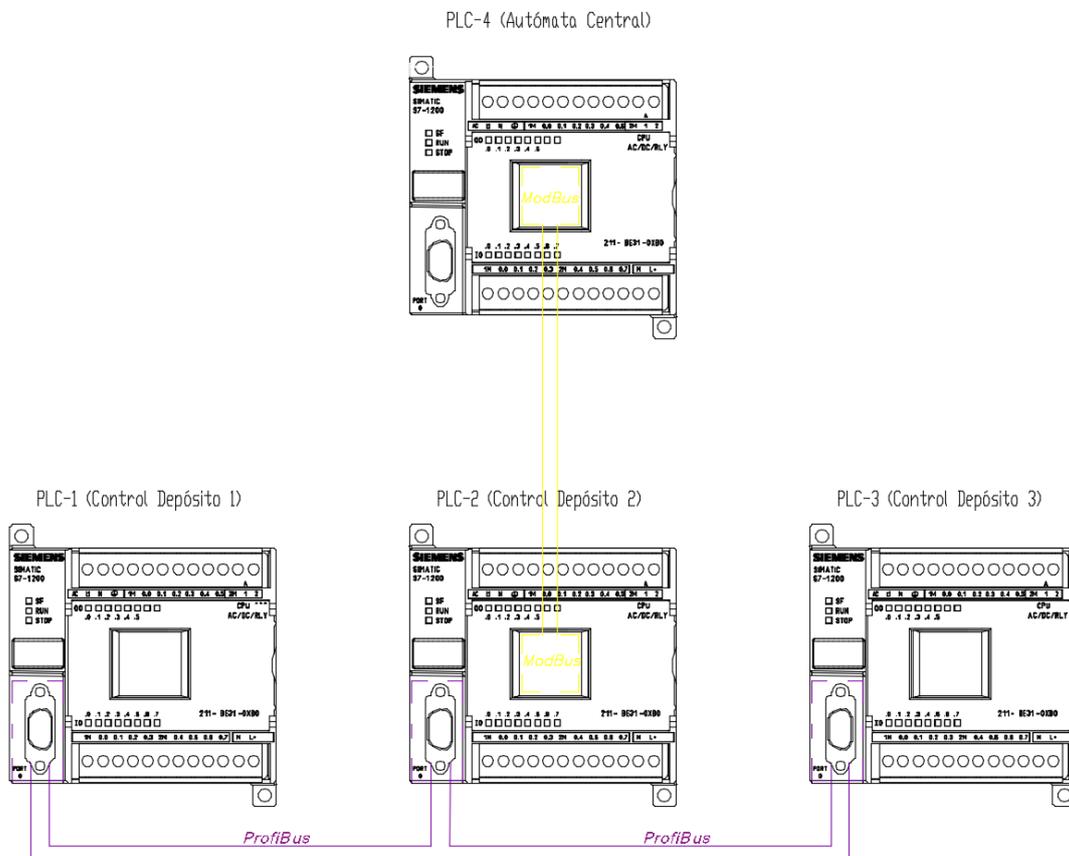


Figura 8. Esquema de la conexión de buses de cada PLC.



industriales
etsii

**Escuela Técnica
Superior
de Ingeniería
Industrial**

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

CAPÍTULO 3

**CONTROLADOR PROGRAMABLE S7-
1200 Y SOFTWARE UTILIZADO EN LA
PROGRAMACIÓN.**



**Universidad
Politécnica
de Cartagena**

3 INTRODUCCIÓN AL PLC S7-1200

El PLC S7-1200 proporciona potencia y fácil manejo imprescindible para controlar una gran diversidad de dispositivos para las diferentes necesidades existentes de automatización. Esto es debido a su diseño compacto, flexibilidad en la configuración y una amplia variedad de instrucciones, el S7-1200 es ideal para controlar una gran cantidad de aplicaciones. [6]

La CPU combina un micro-procesador, una fuente de alimentación de 24V integrada, circuitos de entrada y salida, puerto PROFINET integrado, entrada y salida de control de movimiento de velocidad muy alta y un sistema de control de movimiento integrado, así como entradas analógicas en una carcasa compacta para crear un controlador potente. [6]

Después de descargar el programa, la CPU incluye la lógica necesaria para supervisar y controlar los dispositivos de la aplicación. La CPU monitoriza las entradas y cambia las salidas de acuerdo con la lógica de nuestro programa, que puede incluir lógica booleana, conteo temporización, operaciones matemática complejas, y las comunicaciones con otros dispositivos inteligentes. [6]

La CPU introduce un puerto PROFINET para la comunicación establecer comunicaciones con otros dispositivos conectados mediante esta red. Tiene disponibles módulos complementarios para la comunicación en redes PROFIBUS, GPRS, RS485, RS232, IEC, DNP3 y WDC. [6]

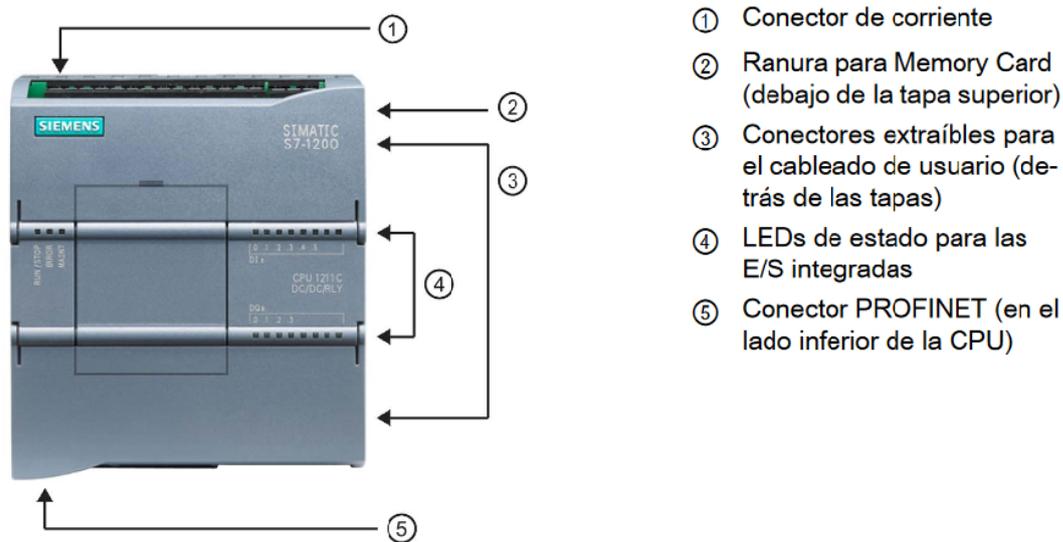


Figura 9. Módulo Maestro y esclavo en comunicación PROFIBUS. [6]

Función	CPU 1211C	
Dimensiones físicas (mm)	90x100x75	
Memoria	Trabajo	50kB
	Carga	1MB
	Remanente	10KB
E/S integradas locales	Digital	6 entradas/4 salidas
	Analógico	2 entradas
Tamaño de la Memoria Imagen de proceso	Entrada(I)	1024 bytes
	Salidas(Q)	1024 bytes
Área de marcas (M)	4096 bytes	
Ampliación con módulos de señales(SM)	Ninguna	
Signal board (SB), Battery Board(BB) o communication Board(CB)	1	
Módulo de comunicación(CM)(ampliación en el lado izquierdo)	3	
		6 para C/E
Contadores rápidos	Total	0
	1MHz	de la 0-5
	100/80 kHz	0
	30/20 kHz	0
	200 kHz	4
Salidas de Impulso	Total	0
	1 MHz	de salida 0-3
	100 kHz	0
	20 kHz	SIMATIC MC
Memory Card	20 días	
Tiempo de del reloj RealTime	1	
Puerto de comunicación Ethernet PROFINET		
Velocidad de funciones matemáticas con números reales	2,3 us/Instrucción	
	0,08 us/Instrucción	

Tabla 5. Característica de CPU 1211. [6]

3.1 TIPOS DE LENGUAJE

Cuando se quiere programar para implementar un sistema automático en un PLC, se puede realizar de formas diferentes en función del tipo de programa y del autómatas a utilizar. Los autómatas Siemens pueden ofrecer hasta 5 formas de programar sus autómatas, pero esto es dependiendo del modelo ya que puede incluir solo algunos de estos. [6]

FUP: es el lenguaje gráfico que se basa en la utilización de puertas lógicas o bloques lógicos, tales como "OR" o "AND" del álgebra booleana. Además, permite representar las funciones matemáticas con un grado de complejidad muy alto, esto mediante cuadros lógicos. Debido a esto resultaría muy sencillo ver las agrupaciones por bloques de las distintas secuencias lógicas, tanto desde la de lógica más compleja hasta la booleana. [6]

KOP: también conocido como diagrama de contacto o ladder, es un lenguaje gráfico que está basado en esquemas eléctricos de control básico, como son bobinas y condensadores para representar una secuencia lógica. De los lenguajes de programación este es el más extendido

debido a su parecido con el esquema de un automatismo realizado mediante la lógica del cableado, pero cuando se trabaja con sistemas complejos este puede resultar bastante difícil tanto para su comprensión o como seguimiento. [6]

SCL: lenguaje estructurado de alto nivel que se basa en texto. Resulta un lenguaje de control optimizado con el que se puede trabajar con datos para ejecutar algoritmos de alta complejidad. Una ventaja es que simplifica el tamaño del código, pero a la vez resulta ciertamente complicado y no se utiliza en programación centrada en lógica booleana. [6]

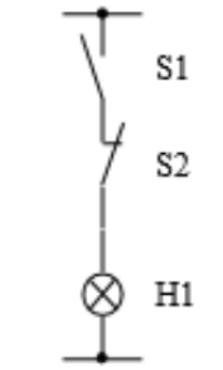
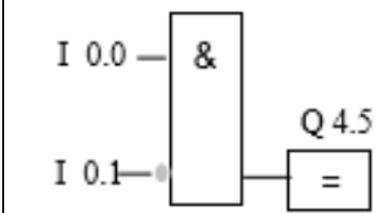
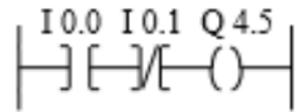
Esquema Electrico	SCL	FUP	KOP
	<pre>A I 0.0 ANI 0.1 = Q 4.5</pre>		

Figura 10. Esquema comparativo entre SCL, FUP y KOP

3.2 PROGRAMACIÓN

Elegido ya el tipo de lenguaje, se procederá a explicar cómo funciona la programación en la marca Siemens, por lo que resulta de una gran importancia saber sobre los módulos en los que están organizados todos los elementos.

OB: Módulo o bloque de organización que sirve para gestionar los demás bloques y funciones del PLC. Este es llamado según su correspondencia en funciones de sus tipos. A continuación se enlistan los diferentes tipos.

OB 1: bloque principal donde se realizan las llamadas de las demás funciones (FB y FC), es ejecutada de forma continua, queriendo decir que cada vez que finalice, volverá a empezar.

OB 100: este módulo se ejecuta solo al iniciar el autómata. Mayormente conocido como OB de arranque.

OB cíclica: es un módulo que se ejecuta periódicamente.

OB de fallo: bloque asíncrono ejecutado cuando ocurre algún fallo. Evita que el autómata entre en parada o ejecuta un listado de instrucciones al ocurrir el evento.

Los OB usan 20 bytes de memoria local con variables específicas.

FC: esta es una función que puede ser llamada desde un OB, otra FC o un FB. Se utiliza para estructurar un código para que quede organizado y ordenado de forma ramificada. Este también permite reutilizar el código sin necesidad de volver a escribirlo, utilizando entradas y salidas de dichas funciones. Los bloques FC Utilizan habitualmente la memoria global o compartida del autómata.

FB: es un Bloque de función parecido a los bloques FC, pero su diferencia es que utiliza bloques de memoria dedicada asociada a cada FB, denominada DB de instancia. Se puede hacer su

llamada desde cualquier otra FB, OB o FC. Cada vez que se realiza una llamada es necesario asignarle una DB de instancia.

3.3 SOFTWARE DE PROGRAMACIÓN

TIA Portal es una plataforma que tiene como fin la unificación de todos los programas utilizados en Siemens en sus distintos dispositivos, de forma tal que resulte toda la programación de una aplicación lo más compacta posible. El software cuenta con una interfaz que se encuentra enfocada a los nuevos autómatas como los modelos de las series S-1200 y S-1500 de Siemens.

3.4 SIMBOLOGÍA DEL SOFTWARE TIA PORTAL

En TIA portal los códigos de programación utilizan simbologías, en la siguiente imagen se muestra las simbologías más utilizadas en el lenguaje KOP, dicho lenguaje de programación utiliza bits en sus operaciones.

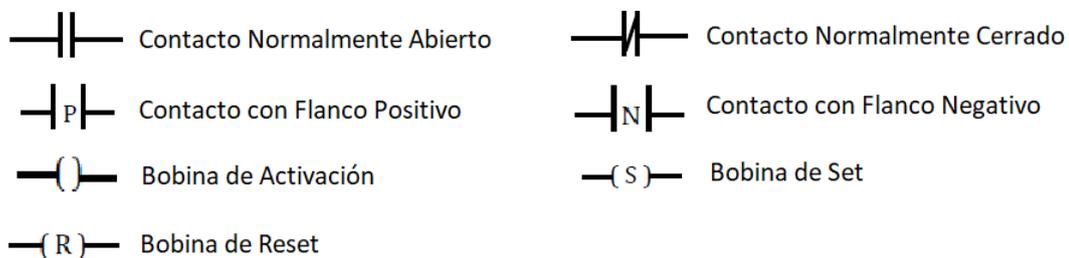


Figura 11. Simbología utilizada TIA portal.

Contadores: las instrucciones de contador se utilizan para hacer una cuenta de los eventos de programa internos y los eventos de proceso externos. Cada contador emplea una estructura almacenada en un bloque de datos para actualizar los datos del contador. El bloque de datos se asigna cuando se coloca la instrucción de contador en el editor.

- CTU es un contador ascendente.
- CTD es un contador descendente.
- CTUD es un conteo ascendente y descendente.

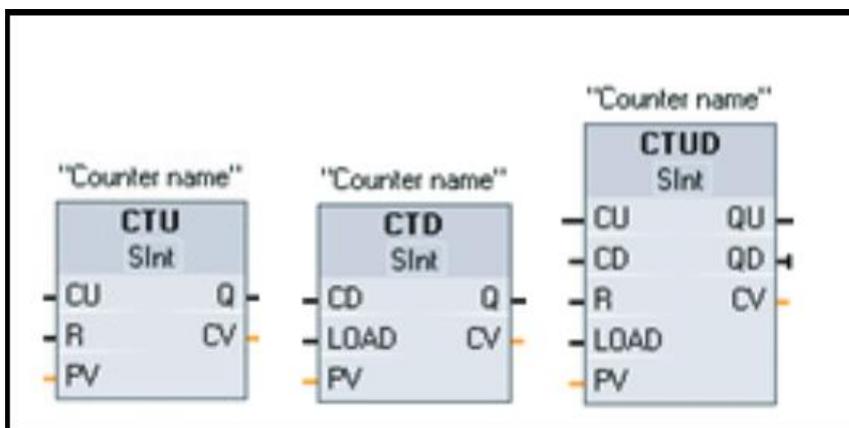


Figura 12. Bloques de contadores CTU, CTD, CTUD. [6]

PARÁMETRO	TIPO DE DATOS	DESCRIPCIÓN
CU,CD	Boolean	Conteo ascend-descend, incrementando 1
R(CTU, CTUD)	Boolean	Lleva el valor del contador a cero
LD(CTD, CTUD)	Boolean	Control el valor cargado
PV	Sint,Int,USInt,Uint,UDInt	Valor de conteo predeterminado
Q,QU	Boolean	si CV>=PV es verdadero
QD	Boolean	si CV<=PV es verdadero
CV	Sint,Int,Dint,USInt,Uint,UDInt	Valor del conteo actual

Tabla 6 Tipo de datos para parámetros CTU, CTD, CTUD. [6]

La cantidad de contadores que se pueden utilizar en un programa está limitado solo por el espacio de memoria disponible en la CPU. Los contadores utilizan los siguientes espacios de memoria:

- En los tipos de datos USInt o SInt, la instrucción de contaje utiliza 3 bytes.
- En los tipos de datos nt o UInt, la instrucción de contaje utiliza 6 bytes.
- En los tipos de datos DInt o UDInt, la instrucción de contaje utiliza 12 byte.

Comparadores: Compara los elementos introducidos con el mismo tipo de datos. Si la comparación en el lenguaje de contactos KOP es TRUE (verdadera), se activa el contacto o se pone a 1. Si la comparación en el lenguaje de bloques o cuadros FUP es TRUE (verdadera), la salida del bloque se pone en TRUE.

KOP	FUP	SCL
		<pre> out := in1 = in2; or IF in1 = in2 THEN out := 1; ELSE out := 0; END IF; </pre>

Tabla 7. funciones de comparación en diferentes tipos de lenguaje. [6]

A continuación se muestra una tabla con los diferentes tipos de bloques para comparación que existe en TIA portal.

TIPO	DESCRIPCIÓN
==	es igual a
<>	es diferente de
>=	es mayor o igual a
<=	es menor o igual a
>	es mayor que
<	es menor que

Tabla 8. Operadores lógicos TIA Portal. [6]

En la programación del proyecto también se ha utilizado instrucciones tales como In_Range y Out_Range. Estas instrucciones comprueban si el valor de entrada está dentro o fuera de un

rango de valores específicos. Si la comparación es TRUE (verdadera), la salida entonces será verdadera. A continuación se muestra una imagen de como éste se representa en el lenguaje utilizado en este proyecto y en los demás lenguajes.

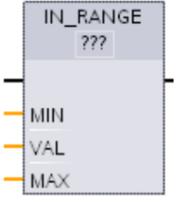
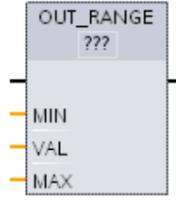
KOP / FUP	SCL
	<pre>out := IN_RANGE(min, val, max);</pre>
	<pre>out := OUT_RANGE(min, val, max);</pre>

Figura 13. Representación de Instrucción In_Range. [6]

Estas dos instrucciones tienen como parámetros MIN, VAL, MAX que son entradas utilizadas para establecer el máximo y mínimo del rango en el que el valor oscilará. Estos datos de entrada deberán ser del mismo tipo y los tipos de datos soportados son: Sint, Int, DInt, USInt, UInt, Real, LReal.

3.5 PROGRAMACIÓN DE PAGINA WEB.

En este proyecto los valores de las variables que están siendo monitoreados por los PLCs se subirán a un servidor web proporcionado por Siemens para ser visualizadas en tiempo real.

Esta programación está hecha en HTML que es un lenguaje de programación que se utiliza para el desarrollo de páginas de web. La sigla HTML corresponde a HyperText Markup Language.

A continuación se muestran algunos de las instrucciones más utilizadas para la creación de esta página web.

- **Como estructura la paginas tenemos las siguientes instrucciones.**

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

Es utilizado para la configuración de fuente de la página a formato UTF-8.

```
<meta http-equiv="Refresh" content="10">
```

Debido a que solo utilizaremos la página para visualizar los datos, hemos puesto este comando en la línea de encabezado para que cada 10 segundos se actualice la página web automáticamente.

```
<HTML> ... </HTML>
```

Abarca y señala toda la página web, que tiene de cabecera y cuerpo.

:=<Varname>:

Comando utilizado para la lectura de variable del PLC. Se puede incluir en cualquier lugar del texto HTML.

```
<!-- AWP_In_Variable Name=""<Varname1>"" Name=""<Varname2>""  
Name=""<Varname3>"" -->
```

Comando utilizado para la escritura en variables del PLC.

En la escritura existen dos sintaxis, una es la de campo de entradas y la otra es lista de selección.

Ejemplo:

- Sintaxis utilizando campos de entradas:

```
<!--AWP_In_Variable Name=""Objetivo"" -->  
<form method="post">  
<p>Input Target Level: <input name=""Objetivo""  
type="text"><input type="submit" value="Escribir en PLC"> </p>  
</form>
```

- Sintaxis utilizando lista de selección:

```
<!-- AWP_In_Variable Name=""Bloque_de_datos".ManualOverrideEnable' -->  
< form method="post">  
< select name="" Bloque_de_datos".ManualOverrideEnable'>  
< option value=1> Si </option>  
< option value=0> No </option>  
</select> <input type="submit" value="submit setting"> </form>
```

<HEAD>... </HEAD>

Abarca y señala el encabezado de una página web, que comprende un conjunto de datos e informaciones que no se mostraran en la en la página en tiempo real, entre ellas está el título de la página, pero que ayuda al navegador y al buscador para descifrar o a descubrir correctamente la página web.

Los comandos AWP son utilizados en la página web para el intercambio de datos entre la página de usuario y la CPU. Estos comandos son introducidos en forma de comentarios y nos ofrecen las siguientes posibilidades.

Lectura y escritura de las variables del PLC.

Definición y asignación de tipo Enum.

Definición e importación de fragmentos de bloques de datos.

<TITLE>... </TITLE>

Está dentro de la instrucción cabecera (HEAD), lo que se pone dentro de aquí se muestra en la barra del título de la página web.

<BODY> ... </BODY>

Engloba todo el cuerpo de la página, y constituye el conjunto de informaciones (texto e imágenes) que se muestran en una página web, así como las indicaciones de cómo deben mostrarse. Esta instrucción admite atributos.

- **Para formatos de párrafos tenemos:**

Con los formatos de párrafos el texto de la página web a construir se puede estructurar en encabezado de las diferentes secciones de la página, que pueden tener distintos niveles desde el 1 a hasta el 6, siendo 1 el más importante. A continuación se muestra la instrucción que debe ir en los extremos de cada texto.

<H1> ... </H1> o <H2> ... </H2> (hasta 6)

<P>... </P>

Párrafos normales.

<P align="center">... </P>

El atributo align nos permite alinear el texto del párrafo. Este se puede aplicar a otras etiquetas tales como la etiqueta <H1>, <H2>.

**
**

Permite dividir un párrafo comenzando en una línea nueva pero sin dejar espacio.

<HR>

Coloca una línea horizontal de separación.

- **Para formato de Texto tenemos:**

Formatos Físicos:

** **-Texto en Negrita.

<I> </I> -Texto en Cursiva.

<U> </U>-Texto Subrayado.

<BIG> </BIG>-Texto Grande.

<SMALL> </SMALL>-Texto Pequeño.

Permite combinar etiquetas anidándola, es decir una dentro de otra:

< B >...<I> ... </I>...

** ... **

Esta Indicación nos sirve para elegir el color a usar en las letras.

** ... **

Indicación para elegir el tamaño de letra a usar.

La etiqueta FONT permite combinaciones de los atributos COLOR, SIZE y FACE

Para definir la apariencia de la página web se ha utilizado un programa llamado CSS que es un lenguaje de programación que determina la forma de un documento elaborado en un lenguaje de marcado como HTML. Así que los elementos de la página web hechos en HTML se les proporcionará la apariencia física de su preferencia utilizando CSS: colores, dimisión entre

elementos, diferentes tipos de letras, etc. desligando de este modo la estructura de la presentación.

Esta separación entre la estructura y la presentación es muy significativo, porque permite que sólo cambiando los códigos CSS se modifique por completo la apariencia de una página web. Esto facilita también, que el usuario pueda emplear hojas de estilo personalizadas.



industriales
etsii

**Escuela Técnica
Superior
de Ingeniería
Industrial**

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

CAPÍTULO 4

DESCRIPCIÓN DEL SISTEMA



**Universidad
Politécnica
de Cartagena**

4 BREVE DESCRIPCIÓN TRATAMIENTO DE AGUAS RESIDUALES

Las aguas residuales industriales son generadas por las industrias como consecuencia de sus actividades. La variedad de las aguas residuales en la industria puede ser lo bastante grande debido a algunas actividades como por ejemplo, refrigeración, aguas de proceso, limpieza y estas pueden contener contaminantes de diferente naturaleza.

En la mayor parte de los procesos industriales se utiliza agua, una vez esta agua se ha utilizado, debe ser tratada antes de ser vertida, para luego de su tratamiento ser devuelta al medio natural a ser usada nuevamente por la industria.

Se debe tomar como medida que el tratamiento del agua deber ser suficiente para que al ser vertida esta no cause ningún impacto ambiental en su destino. En caso de que esta sea vertida a una red pública se debe tomar en cuenta que composición de aguas residuales se debe cumplir para estar acorde con los parámetros físicos y químicos de la norma vigente en ese país o estado.

Las aguas residuales industriales si no son conveniente para su vertido, antes de ser tratadas, contienen compuestos tóxicos en ocasiones compuestos difíciles de degradación y/o tener un pH alejado o cercano a su neutralización.

El sistema desarrollado en este proyecto trata de una manera automática de monitorizar y controlar el pH desde que el líquido llega al depósito 1 para ser tratado en el depósito 2, donde en esta parte del depósito 2 se le añadan los que es Ácido si el agua tiene el pH mayor de 7 o sustancia Base si dicho pH es menor a 7.

Los ácidos que se pueden utilizar para reducir el pH del agua son los precursores de las sales más habituales que se hallan disueltas de manera natural en el agua, como son los cloruros (ácido clorhídrico), los fosfatos (ácido orto fosfórico), los carbonatos (ácido carbonico-CO₂), los sulfatos (ácido sulfúrico).

En el caso contrario cuando el pH está por debajo de 7 se utilizara carbonato de sodio una sustancia natural procedente de un mineral llamado trona y que tiene la capacidad de aumentar este pH hasta alcanzar su neutralidad.

A continuación se detallara los materiales para la construcción de este proyecto.

4.1 SENSORES DE PROXIMIDAD CAPACITIVO PARA DETECTAR EL NIVEL MODELO CA30CLN12M.



Figura 14. Detector nivel capacitivo Carlo Gavazzi.

En la tabla 9 se muestra las especificaciones más importantes de este sensor.

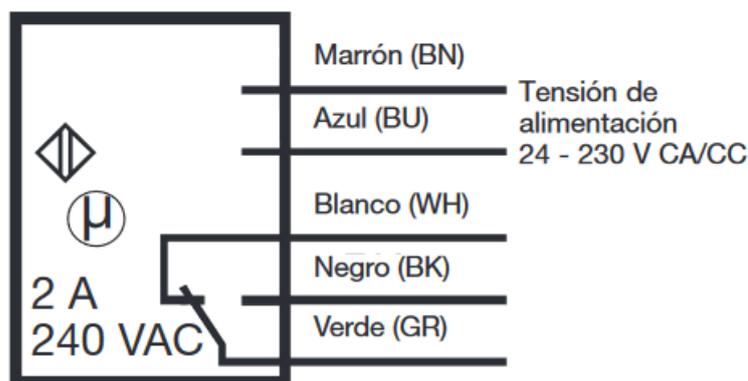


Figura 15. Conexión eléctrica del detector nivel capacitivo Carlo Gavazzi.

ESPECIFICACIÓN SENSOR CA30CLN12M			
Distancia de funcionamiento		Entorno	
Distancia de detección	4-12 mm ajustable	Categoría de Instalación	III(IEC 60664/60664A;60947-1)
Histeresis(H)	de 3 a 20%	Grado de contaminación	3(IEC 60664/60664A; 60947-1)
Tension De Operación (U _B)	de 20 a 255 VCA / V CC	Grado de protección	(IEC 60529;60947-1) NEMA 1, 2, 3, 4x, 5,6,6P, 12
Frecuencia de alimentación	de 47 a 63 Hz	Temperatura Ambiente	
Potencia operativa nominal	de 0.5 a 2.5 VA	Temperatura de funcionamiento	de -20° a + 70°C
Salida	Rele 2 A, SPDT@240 VCA	Temperatura de Almacenamiento	de -40° a + 85°C
AC12	2A	Tensión de aislamiento nominal	> 250 VCA (rms)
AC140	2A	Material de la caja	
DC12	2A	Cuerpo Parte Posterior Trimmer	PBTP Arnitel LCP Vectra
DC13	2A	Conexión	
Pretección	Contra inversión de polaridad y transitorios	Cable	PVC, gris, 2 m 5 x 0.75 mm ² , Ø = 7.5mm
Frecuencia Operativa (f)	<1 Hz	Peso	< 320g
Tiempo de Respuesta			
OFF-ON (T _{ON})	< 500 ms		
ON-OFF (T _{OFF})	< 500 ms		

Tabla 9. Especificaciones detector de nivel.

4.2 SENSOR PARA MEDICION DE pH

Clasificación	Medición de pH	
Fabricante	ENDRESS+HAUSER	
Principio físico	<p>El valor del pH se utiliza como unidad de medida de la acidez o alcalinidad de un medio líquido. El vidrio de la membrana del electrodo suministra un potencial electroquímico que depende del valor del pH del medio. Este potencial es generado por la penetración selectiva de iones H⁺ a través de la capa externa de la membrana. Se forma una capa límite electroquímica con un potencial eléctrico en este punto. Un sistema de referencia Ag/AgCl integrado sirve como electrodo de referencia necesario. El transmisor convierte el voltaje medido en el correspondiente valor de pH.</p>	
Variables medidas	pH	
<p>Usos principales:-su gran seguridad y precisión para certificar que las mediciones no se vean perjudicadas por los cambios de temperatura, presión, mezclas de gas vapor, densidad, turbulencia, burbujas/ebullición, nivel bajo, fluidos de constante dieléctrica variante, pH y viscosidad.</p>		
<p>Ventajas:-Electrodo resistentes con trampa de iones para expansión de contaminantes, de bajo mantenimiento debido a su gran diafragma anular de politetrafluoetileno (PTFE) que desecha la suciedad.</p> <p>-Útil en aplicaciones con grado exigencia: tales como vidrio para productos muy alcalinos y presión estable de alrededor de 16 bar (232 psi).</p> <p>-cuenta con un Anillo opcional de sal para aplicaciones de conductividad baja como agua para alimentación de calderas..</p> <p>-Gran seguridad debido a la transmisión de señal inductiva y sin contacto (no se ve afectado por el polvo en el proceso de llenado o vaciado, lo que ayuda a que el mantenimiento predictivo sea fácil.</p> <p>-Costes de operación reducidos debido a un tiempo de parada del proceso minimizado y una vida útil del sensor muy extensa.</p>		
<ul style="list-style-type: none"> • Información complementaria: Son adecuados para el monitoreo a tiempo completo y de niveles límites de procesos con condiciones estables:- como las Industrias químicas - Industrias papeleras - Centrales energéticas – Plantas para incinerar. <p>Tratamiento de aguas:- Agua potable - Aguas utilizadas como refrigerantes- Aguas de pozo -Con certificados ATEX, FM y CSA para zonas explosivas.</p>		
Web	https://www.es.endress.com/es/instrumentacion-campo/medicion-nivel/radar-guiado-medicion-nivel-fmp51	
Datasheet	https://portal.endress.com/wa001/dla/5000557-/6431/000/04-/TI00191CEN_1314.pdf	
Posibles variables de entrada	pH	
Tipo de salida	4-20 mA, 0- 1500mv	

Tabla 10. Especificaciones sensor pH.

4.3 SENSOR PARA MEDICION DE TEMPERATURA

Clasificación	SONDA DE TEMPERATURA – TIPO INDUSTRIAL MODULAR	
Fabricante	ENDRESS+HAUSER	
Principio físico	<p>Basado en termopar o en RTD</p> <p>Los termopares son semiconductores que ante cambios de temperatura sufren variaciones en su resistencia interna, pudiendo ser de tipo positivo (PTC) o negativo (NTC). Mientras que las RTD's son metales que ante aumentos de Tª sufren un aumento de su resistencia.</p>	
Variables medidas	Temperatura	
<p>Usos principales:</p> <ul style="list-style-type: none"> -Amplio rango de uso -Estilo métrico y conexión roscada a proceso -Termómetro modular -Apto para áreas peligrosas 		
<p>Ventajas:</p> <ul style="list-style-type: none"> -Alto nivel de flexibilidad -Alto nivel de compatibilidad -Cuello de extensión para proteger ante sobrecalentamientos -Protección “Ex ia” y “Ex nA” 		
<p>Información complementaria: Industrias de aguas, de proceso o alimentarias.</p>		
Web	<p>https://www.es.endress.com/es/instrumentacion-campo/medicion-temperatura/termometro-modular-prueba-explosiones</p> <p>https://www.es.endress.com/es/instrumentacion-campo/medicion-temperatura/termoresistencia-zonas-explosivas-pt100</p>	
Datasheet	<p>https://portal.endress.com/wa001/dla/5000000/5738/000/01/TI274ten_0111.pdf</p> <p>https://portal.endress.com/wa001/dla/5000499/1557/000/03/TI00256ten_0111.pdf</p>	
Posibles variables de entrada	-	
Tipo de salida	4-20 mA	
Comunicación digital:	HART, PROFIBUS PA o FIELDBUS	

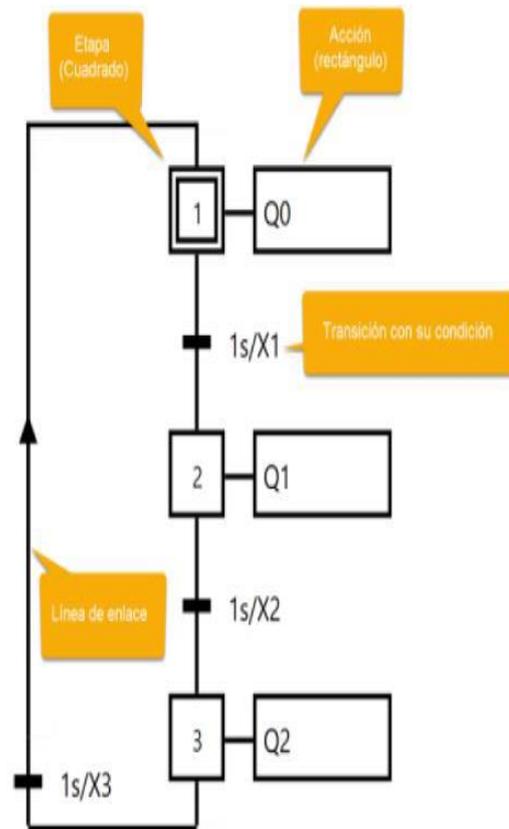
Tabla 11. Especificaciones sensor pH.

4.4 DESCRIPCIÓN DEL PROCESO POR MÉTODO GRAFCET

Para representar procesos existen diversos métodos que permiten la automatización de procesos sin ser personas especialistas en esta área de diseño. En este caso se utilizara el método GRAFCET, que es un diagrama práctico que representa los procesos tomando en cuenta las acciones a realizarse en este proceso, además tomando las condiciones que se deben de cumplir para llegar hasta esa acción. Este método está homologado en Europa por varias normas, una de esas normas es la DIN EN 60848.

El método GRAFCET se compone del siguiente diagrama:

- Etapas (Cuadros): pueden estar activas o inactivas, donde en cada una de ellas se activan o desactivan acciones según la secuencia del proceso. En cada GRAFCET solo Puede haber una etapa activa de forma simultánea.
- Líneas de enlace: Estas permiten conectar los elementos individuales.
- Transiciones (línea perpendicular que cruza la línea de enlace): La transición tiene una condición, debido a que si esta se cumple se activara la etapa siguiente inactivando la anterior.
- Acciones: Las acciones siempre están relacionada con una etapa. Se puede conectar diferentes acciones a una etapa. en una acción se activara por ejemplo una electroválvula y una bomba.



NOTA: En las transiciones solo aparece las condiciones más principales, es decir la más relevantes que son las necesarias para pasar a la siguiente etapa, evitando con esto hacer este diagrama poco legibles y muy difíciles de representar.

4.4.1 DIAGRAMA GRAFCET DEPOSITO 1

Este diagrama representa el proceso de automatización de un depósito de almacenamiento de aguas residuales proveniente de una industria. Este proceso tiene una etapa de inicio que comenzará al iniciar el PLC. Cada etapa tiene asociada una o varias acciones que se distinguen por el título. También las etapas tienen una transición que la inactiva, y activa la siguiente, donde cabe la posibilidad de que el proceso se interrumpa volviendo al inicio por algún tipo de Fallo o Alarma. En las etapas y casi en todas las transiciones estamos escribiendo o leyendo datos de los sensores, que están siendo guardadas en una variable para ser transmitida hacia el PLC central.

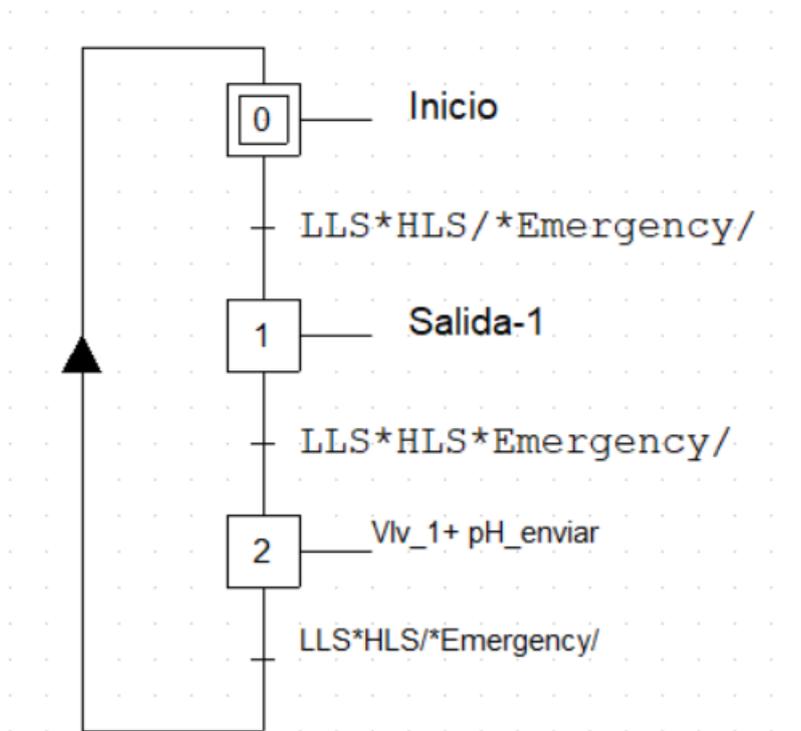


Figura 16. Diagrama GRAFCET del depósito 1.

4.4.2 DIAGRAMA GRAFCET DEPOSITO 2

Este diagrama representa la parte principal del proceso de automatización del DEPOSITO 2. Donde aquí es donde se centra la parte importante de la programación debido que aquí es donde se controla el pH. Se recogen los datos de los PLCs conectado en el bus de campo PROFIBUS.

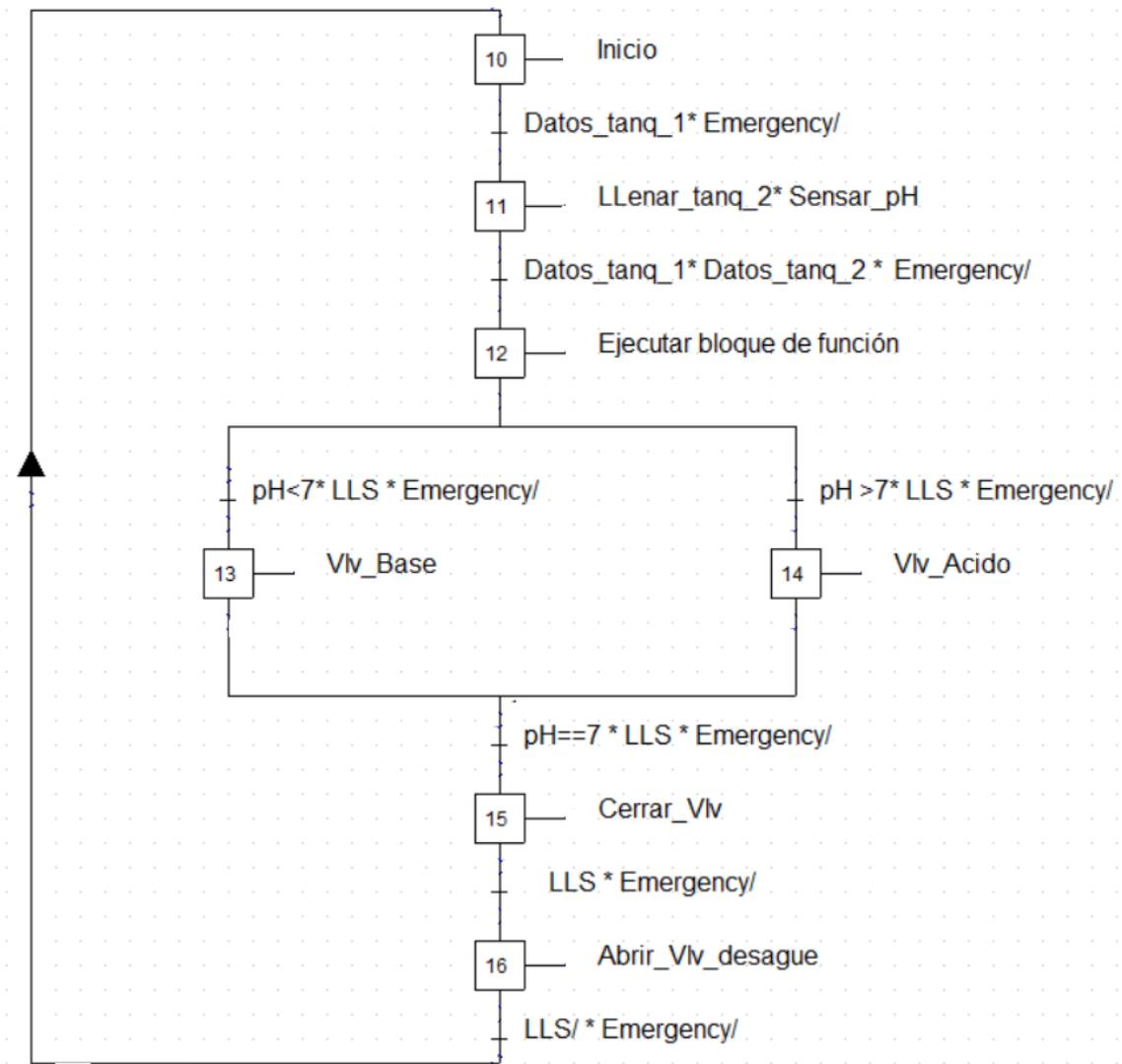


Figura 17. Diagrama GRAFCET del depósito 2.

4.4.3 DIAGRAMA GRAFCET DEPOSITO 3

Este diagrama representa la función principal del autómata destinado para esta área que es el DEPOSITO donde estará el agua ya neutra esperando que la temperatura sea igual al valor de consigna dado para así ser liberada a naturaleza.

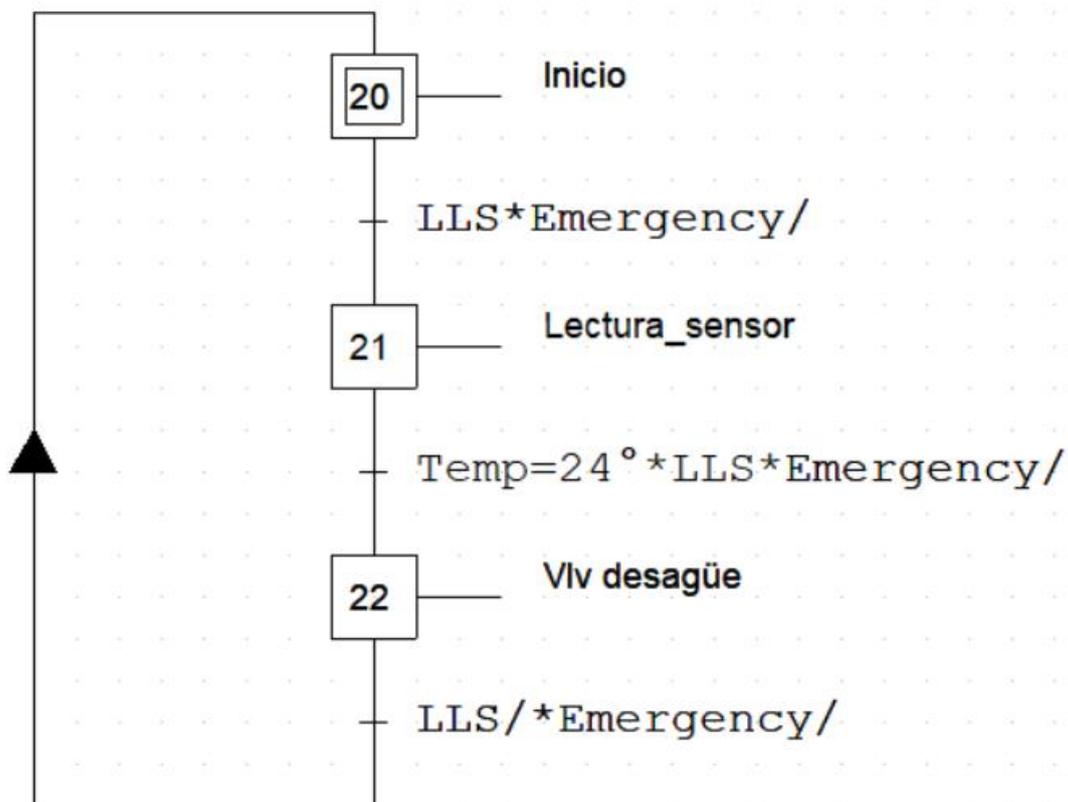


Figura 18. Diagrama GRAFCET del depósito 3.



industriales
etsii

**Escuela Técnica
Superior
de Ingeniería
Industrial**

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial

CAPÍTULO 5

**CODIGO DE PROGRAMACION EN TIA
PORTAL**



**Universidad
Politécnica
de Cartagena**

5- CONFIGURACIÓN EN TIA PORTAL

Esta herramienta de Siemens se utiliza para programar diferentes tipos de autómatas y configurar otros productos. Entre ellos está la programación de los autómatas y programar sus HMI's.

A continuación se muestra los pasos más importantes para la configuración de los autómatas que utilizaremos en esta instalación.

- 1) Creamos un proyecto rellenando el nombre del autor y del proyecto:

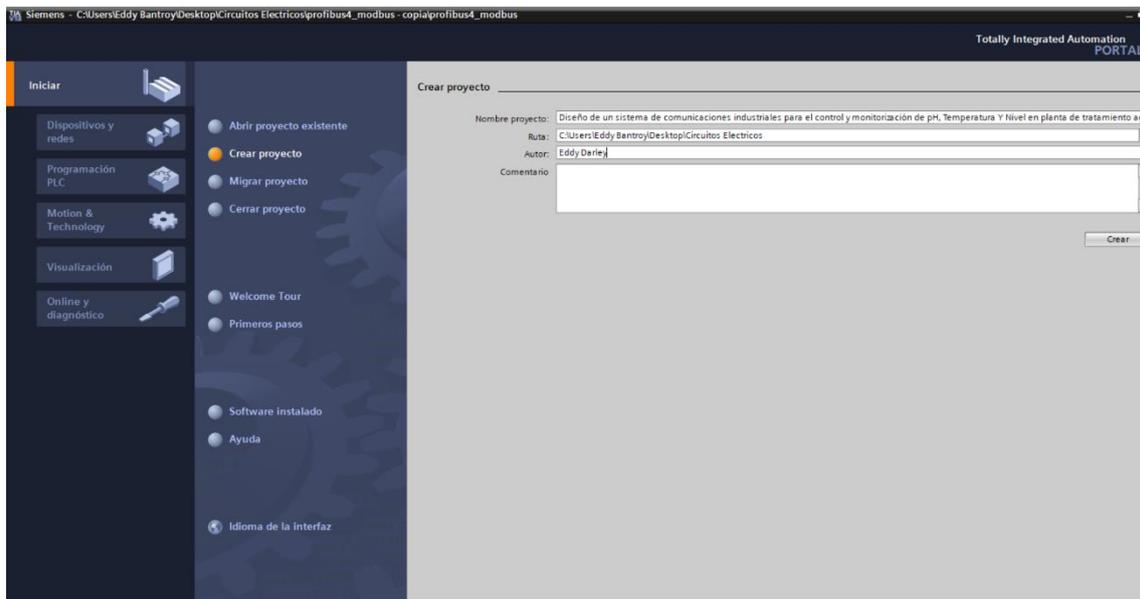


Figura 19. Ventana de inicio software TIA PORTAL

- 2) Tomamos referencia de los hardware a utilizar en esta instalación:

4 controladores CPU 1211 AC/DC/Rly.
Referencia: 6ES7 211-1BE31-0XB0

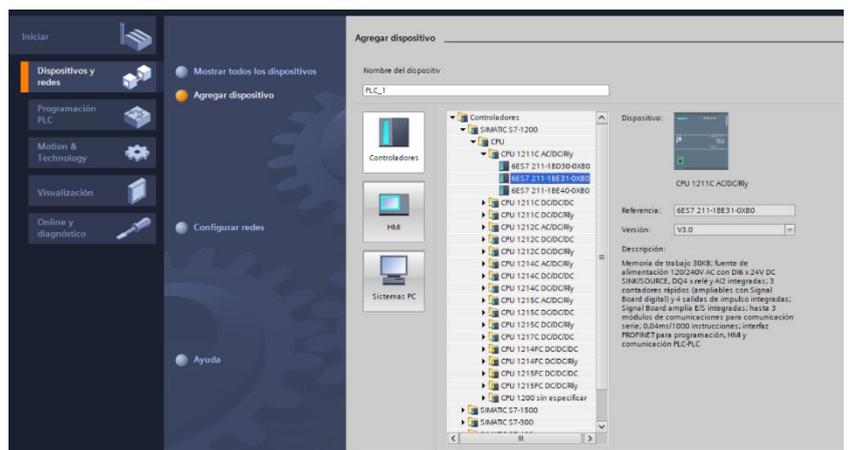


Figura 20. Ventana de selección del autómata.

- 3) Todos los dispositivos agregados los tendremos en una vista general del proyecto como se muestra en la imagen, donde al hacer clic en cada dispositivo, nos aparecerán sus configuraciones:

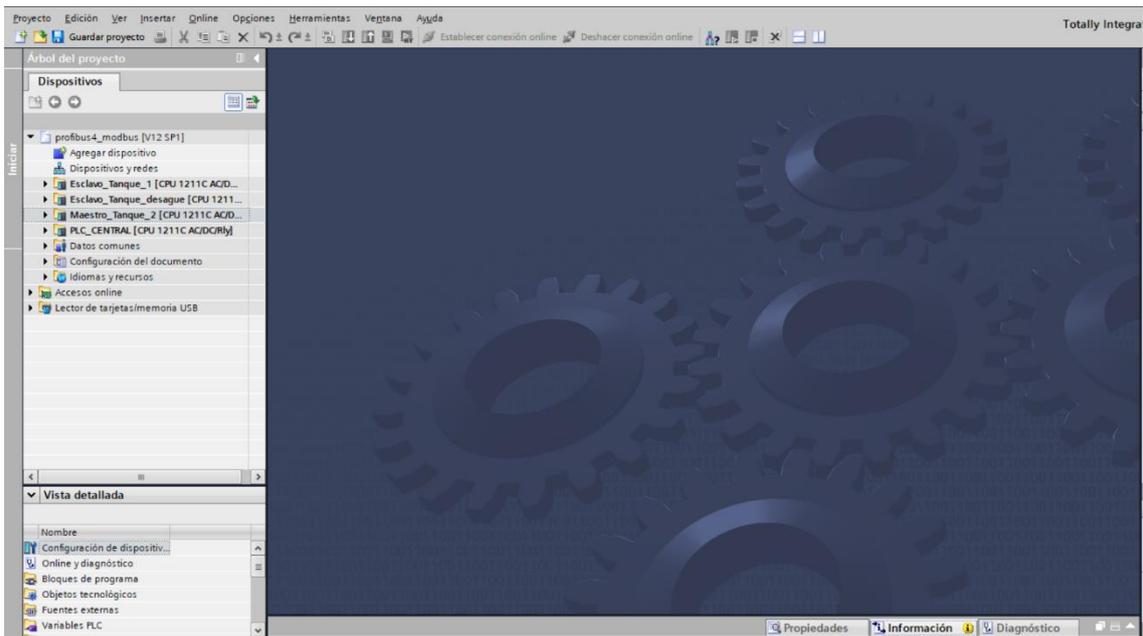


Figura 21. Ventana principal del para programación del autómeta

- 4) Haremos clic en dispositivos y redes donde nos permitirá crear una subred de todos los dispositivos que podemos comunicar y reconocer en el proyecto.

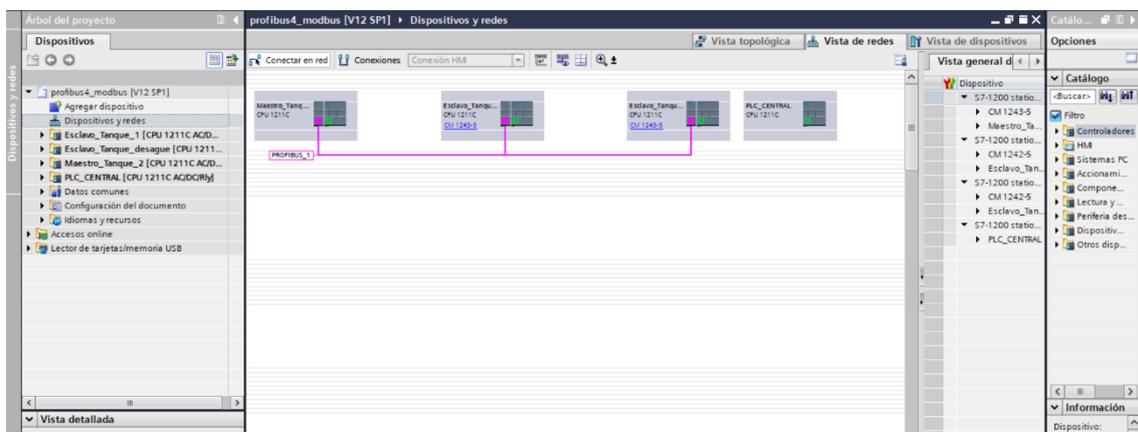


Figura 22. Ventada de redes de los dispositivos

En este capítulo también se desarrolla el código de programación perteneciente a cada uno de los de los autómatas. Esta programación está divididas por segmentos como se muestra a continuación.

5.1 SEGMENTOS DE PROGRAMACION AUTÓMATA DEPOSITO 1

Segmento 1: este Segmento se encarga de simular el llenado del depósito 1 que es el depósito donde llega el agua residual. Para eso se ha utilizado el contador CTUD %DB2 que cuenta desde 0 hasta 100 siendo esto el número de litros almacenado. Utiliza un paro que es el contacto NC que lleva relación con la salida Q0.0 que es la que cerraría la electroválvula de paso de la tubería de llenado del depósito.

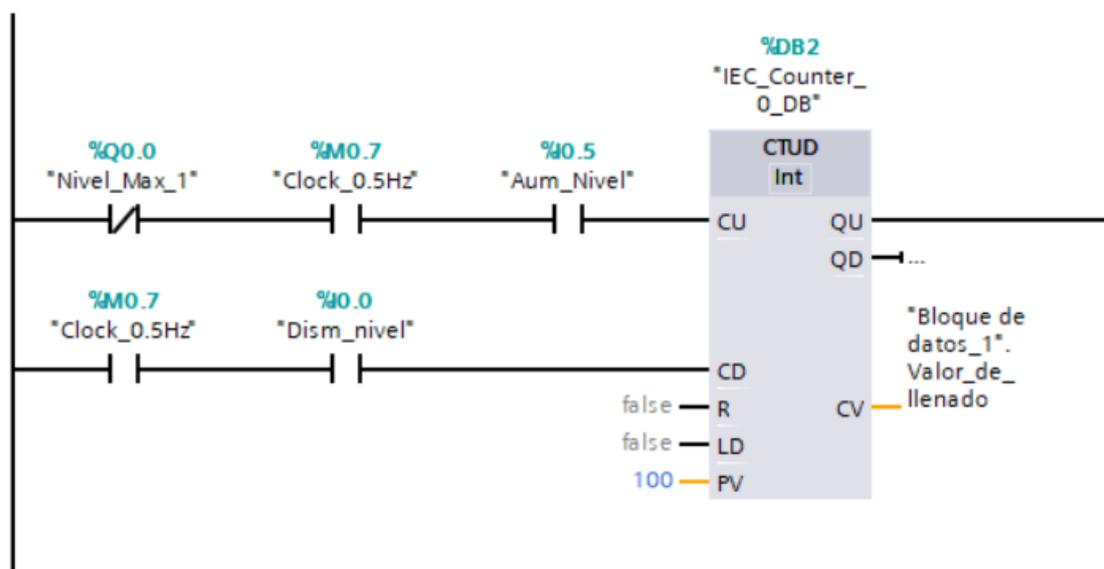


Figura 23. Segmento 1 de programación 1 del autómata.

Segmentos 2 y 3: los segmentos 2 y 3 son los detectores de nivel del depósito están encargados de establecer los límites (inferior y superior) de llenado del depósito 1 y prender y pagar válvula de la tubería de entrada.



Figura 24. Segmento 2 de programación 1 del autómata.



Figura 25. Segmento 3 de programación 1 del autómata.

Segmento 4: éste es un bloque de función que contiene esta la programación de la función que se describe a continuación. Éste recibe la entrada analógica del pH la pasa por el bloque NORM_X que es el bloque que se utiliza para normalizar el valor de entrada en una escala lineal. Luego este valor pasa al SCALE_X que es el encargado de escalar el valor de entrada mapeándolo en el rango establecido.

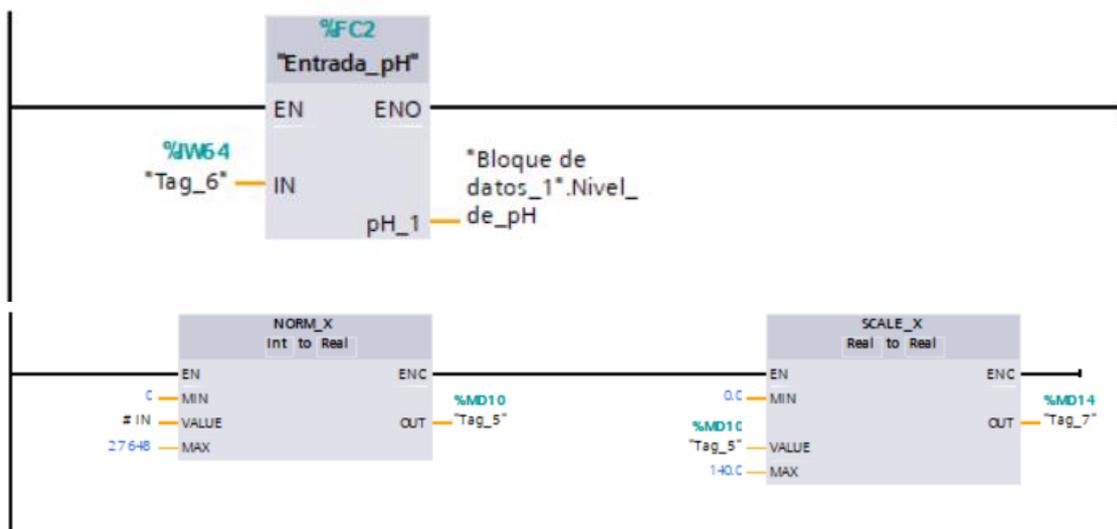


Figura 26. Segmento 4 de programación 1 del autómata.

Segmento 5: aquí ocurre la transferencia del dato almacenado en la variable Nivel_de_pH de bloque de datos a través del bloque de instrucción MOVE y es recibido en el otro PLC. Este dato sale por la salida %QW1 salida asignada para el intercambio de datos entre los PLC contados a través de PROFIBUS



Figura 27. Segmento 5 de programación 1 del autómeta.

Segmento 6: este Segmento funciona de la misma manera que el Segmento 5, solo que este moverá otra variable a través del bloque MOVE y se recibirá en el otro PLC en otra entrada diferente a la que llegara el dato del Segmento 5. Aquí se está enviando el valor de pH censado en el instante.

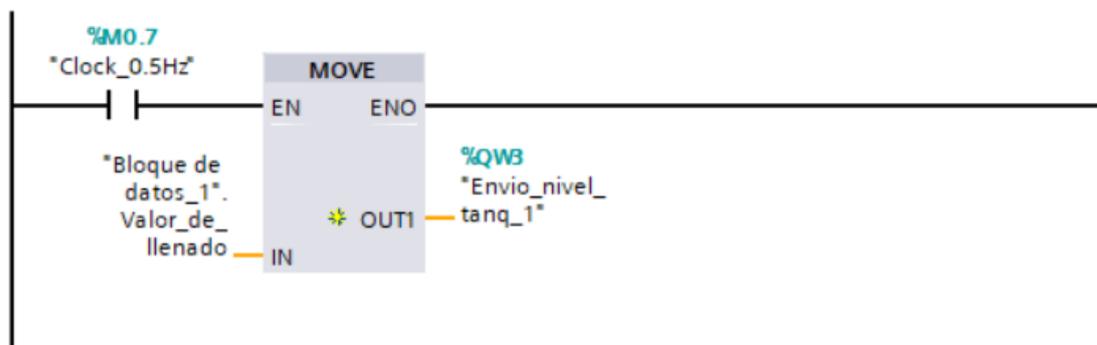


Figura 28. Segmento 5 de programación 1 del autómeta.

5.2 SEGMENTOS DE PROGRAMACION PLC DEL DEPOSITO 2

Este es el autómeta encargado de recibir las variables de los otros autómetas conectado mediante el protocolo de comunicación PROFIBUS y también es el encargado de manejar el la sustancia de Ácido y de Base para la neutralización del agua proveniente del depósito 1. A continuación se explicara los Segmentos que conforman la programación de este autómeta.

Segmento 1: en este Segmento se recibe los datos enviados por el autómeta que está monitorizando el depósito 1. Este dato recibido por la entrada %W1 asignada a intercambio de datos red ProfiBus, es almacenada en una variable dentro de un bloque llamada Maestro4.

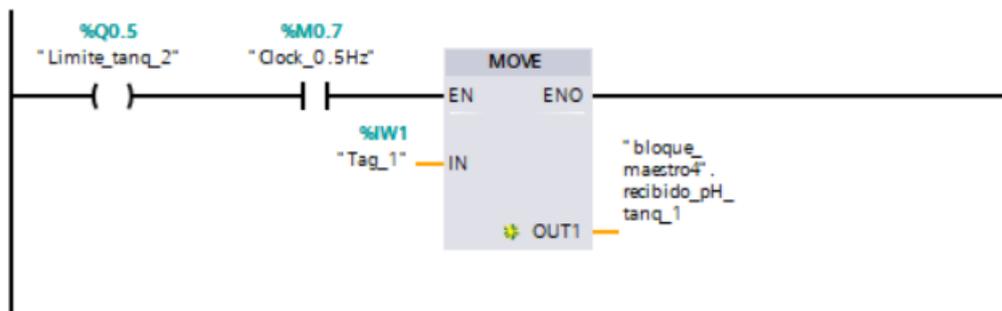


Figura 29. Segmento de programación 1 del autómeta 2.

Segmento 2: Aquí pasa lo mismo que el Segmento 1 solo que se está recibiendo el fluido total acumulado en litros que han pasado desde el depósito 1 a depósito 2.

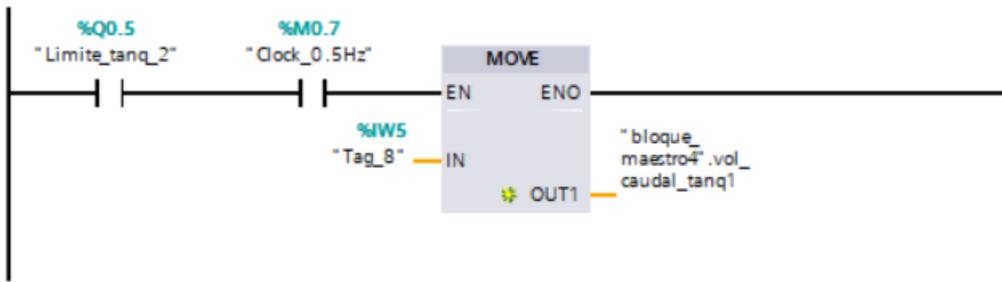


Figura 30. Segmento de programación 2 del autómata 2.

Segmenta 3: En este segmento se simula de forma manual el nivel de líquido que se encuentra en el depósito antes de recibir el volumen de líquido del DEPOSITO 1, siempre y cuando se cumpla con la condicionar puesta en este Segmento.

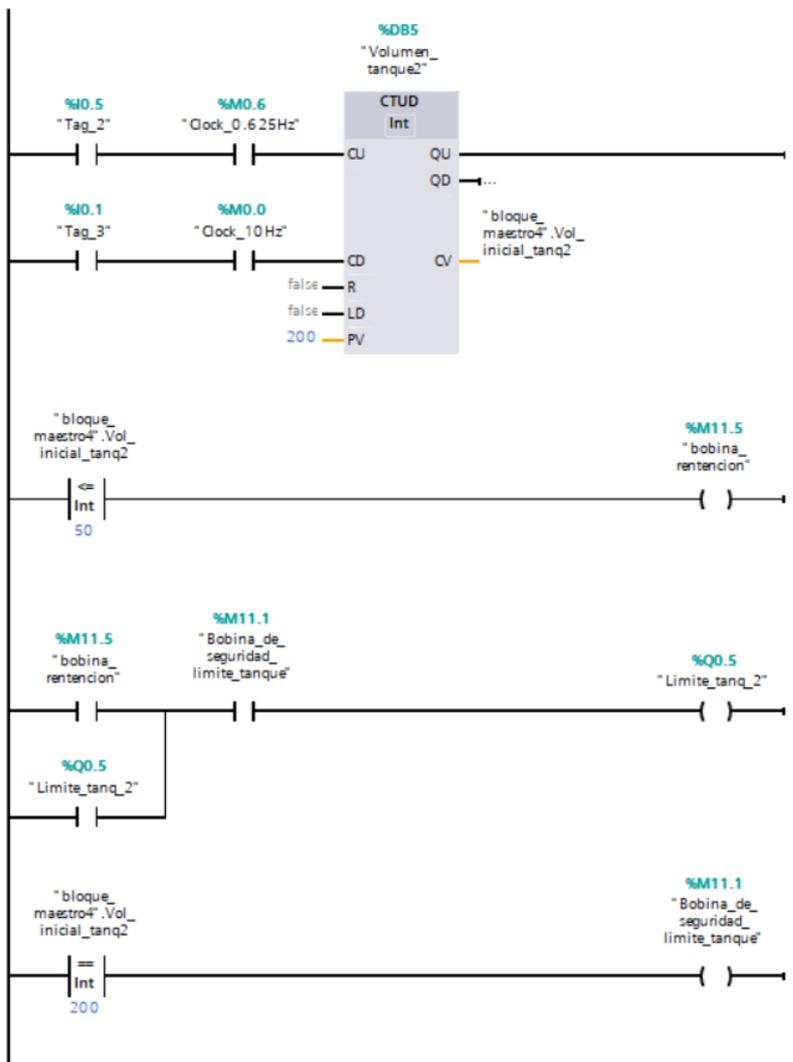


Figura 31. Segmento de programación 3 del autómata 2.

Segmento 4: este segmento se encarga de censar el pH que tenemos en el depósito 2 antes de que llegue el fluido del depósito 1. Este sistema está listo para ser utilizado con un sensor físico el cual nos evitaría utilizar el módulo de cálculo hecho más adelante.

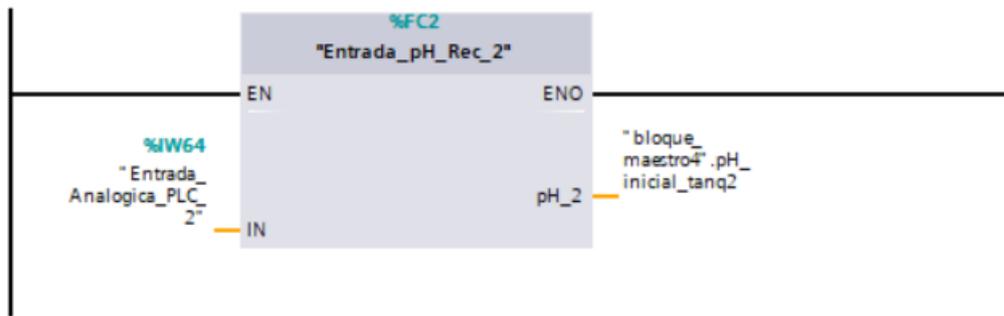


Figura 32. Segmento de programación 4 del autómata 2.

Dentro de este bloque tenemos lo que es la habilitación de la entrada analógica para recibir la señal del sensor analógico que se le estaría conectado al sistema. En la siguiente imagen se puede apreciar el contenido del bloque.

Segmento 4.1



Figura 33. Segmento de programación 4.1 del autómata 2.

Segmento 4.2: Al final se utiliza un convertor para cambiar de tipo de datos de Real a Entero.

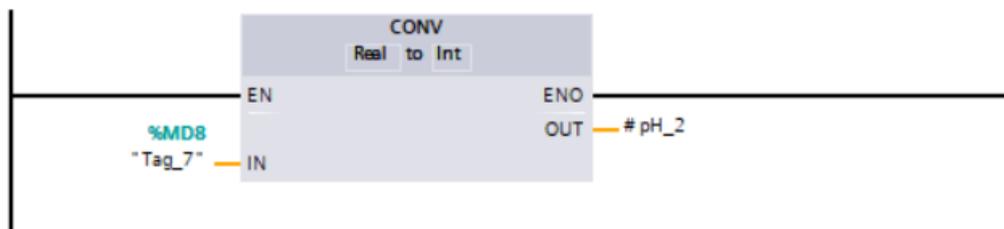


Figura 34. Segmento de programación 4.2 del autómata 2

Segmento 5: en este segmento es donde se efectúa el cálculo donde se relaciona el volumen existente en el depósito 2 con el volumen que llega del depósito 1, al igual pasa con los pH, al final este bloque nos dará el pH final antes de ser neutralizado.

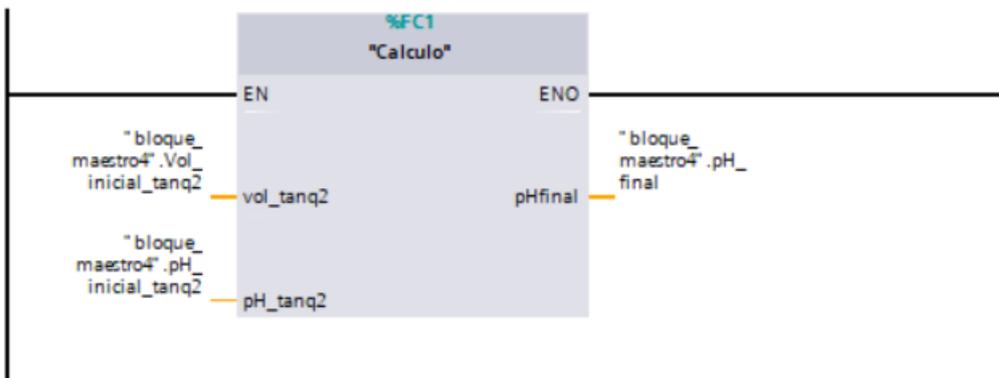


Figura 35. Segmento de programación 5 del autómata 2

A continuación se mostrara la programación que compone este bloque del Segmento 5.

En la figura de abajo se muestra un bloque donde se multiplican los datos de las variables recibidas del depósito 1.

Segmento 5.1

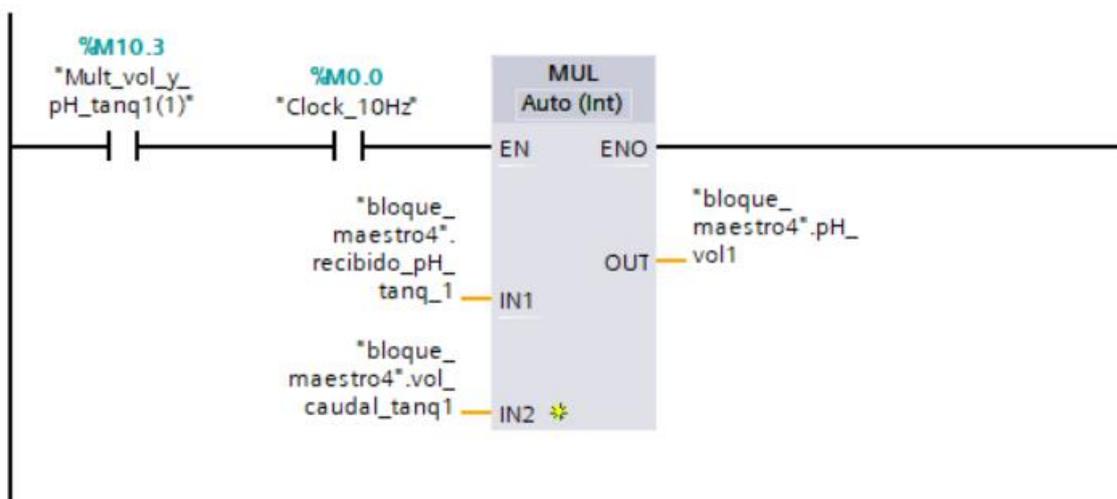


Figura 36. Segmento de programación 5.1 del autómata 2.

El siguiente Segmento es el 5.2 es donde se multiplican las variables de DEPOSITO 2.

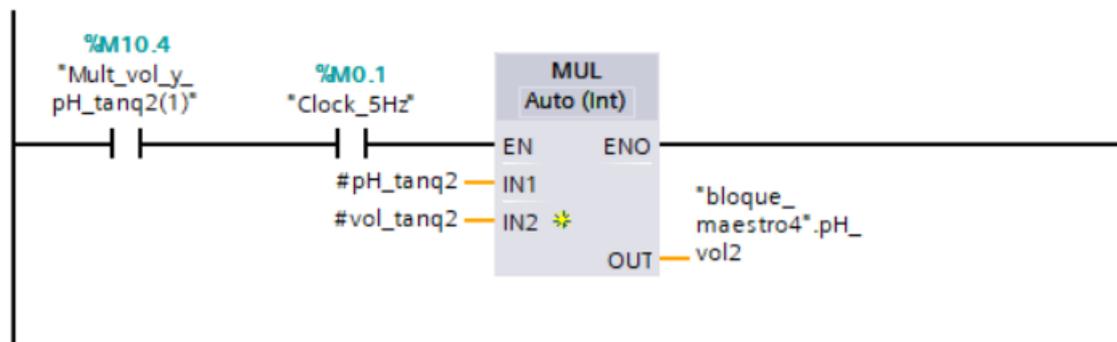


Figura 37. Segmento de programación 5.2 del autómata 2

Segmento 5.3 hace la suma del pH llegado del depósito 1 y el pH depósito 2

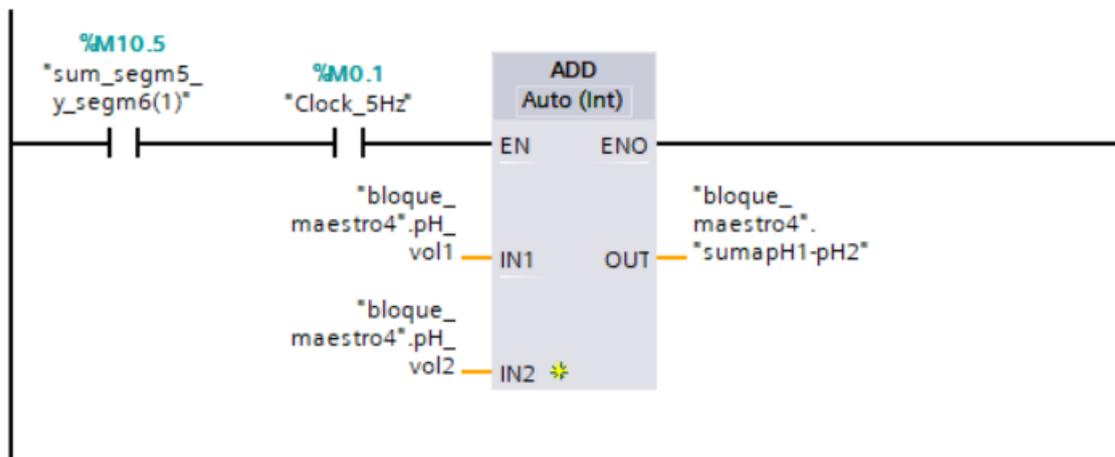


Figura 38. Segmento de programación 5.3 del autómata 2

El **Segmento 5.4** hace la suma del volumen llegado del depósito 1 y el volumen depósito 2 dando como resultado el volumen total que existe en el depósito 2.

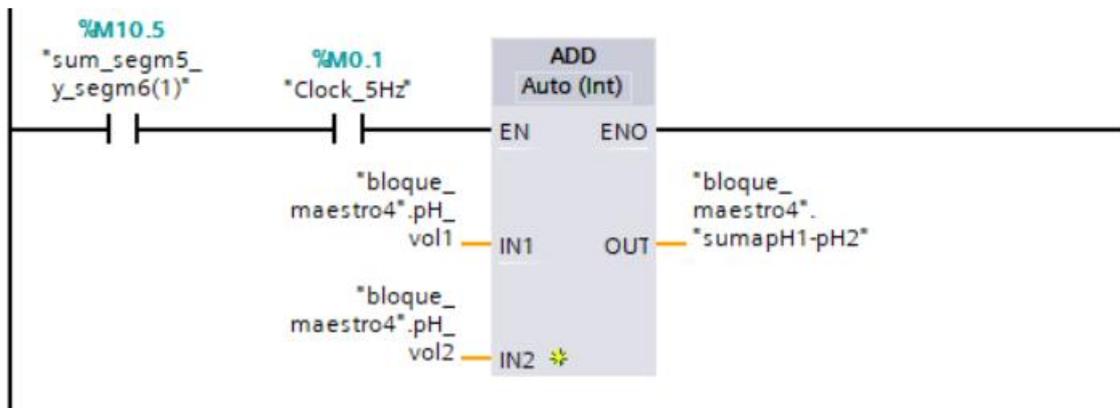


Figura 39. Segmento de programación 5.4 del autómata 2

Este es el último Segmento es el 5.5 del bloque de cálculo, este da como resultado el pH final ya con la mezcla los dos líquidos con pH diferente, es decir, a la salida dará el pH resultante producto de la mezcla el líquido que viene del depósito 1 y el líquido que estaba en el depósito 2.

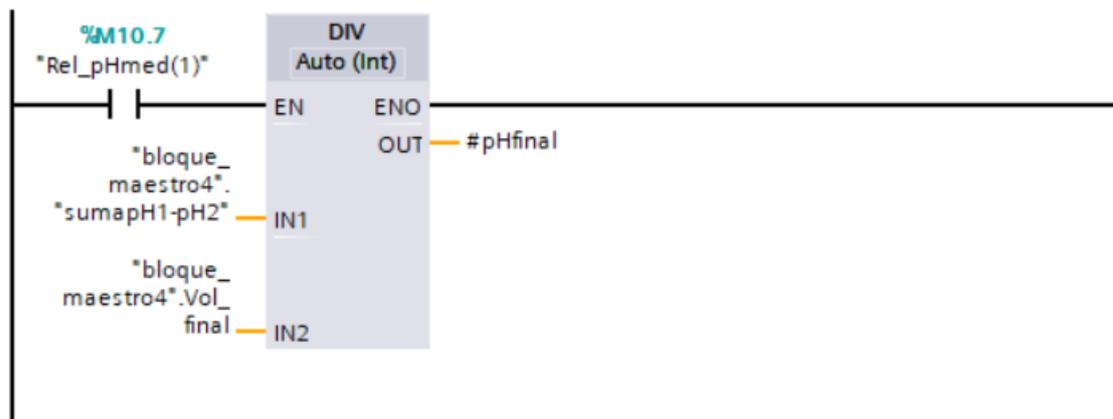


Figura 40. Segmento de programación 5.5 del autómata 2.

Segmento 6: en este Segmento hay un bloque de funciones que dentro lleva una lógica que al detectar que el pH_final este en un rango de 0-6.9 escalada a unidades enteras de 0-69 para simular la abertura de la válvula asignada al depósito de sustancia Base cuando la condición dentro del bloque se cumplan.

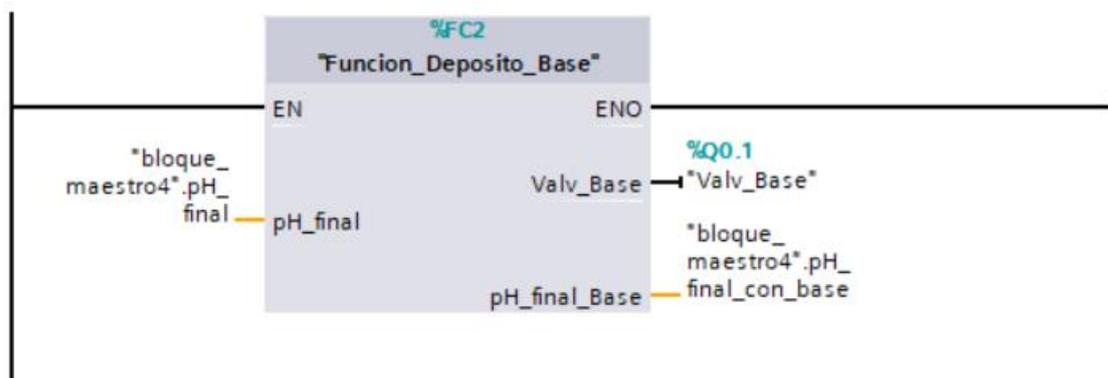


Figura 41. Segmento de programación 6 del autómata 2.

A continuación se muestran los Segmentos de programación que componen este bloque de funciones.

- 1) **Segmento 6.1:** En este Segmento utiliza instrucciones de comparación que en este caso quedaría como a condición IF-THEN. Al principio compara si el valor de pH_final está dentro del rango asignado en la lógica de comparación que va desde, va desde 0 hasta menor que 70. Al cumplir esto, se activara la válvula del depósito de base.

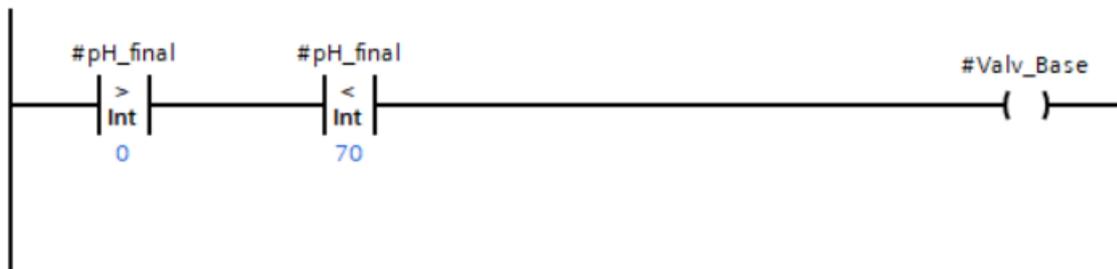


Figura 42. Segmento de programación 6.1 del autómata 2.

- 2) **Segmento 6.2:** aquí se usa un bloque MOVE que es utilizado para simular el estado "OFF" de la válvula del depósito que contiene la sustancia Base.

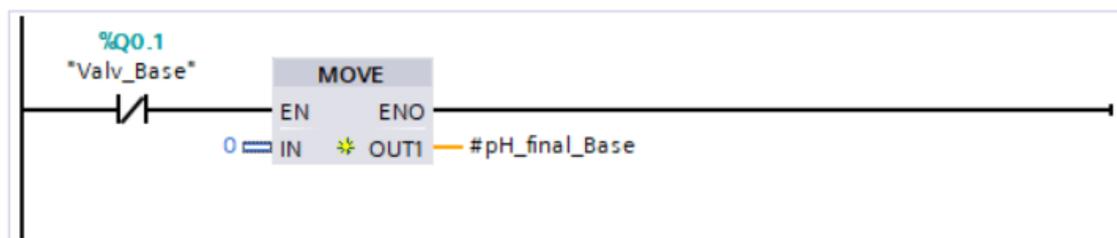


Figura 43. Segmento de programación 6.2 del autómata 2.

- 3) Segmentos 6.3 aquí se usa un bloque MOVE que es utilizado para simular el estado "ON" de la válvula del depósito que contiene la sustancia Base.

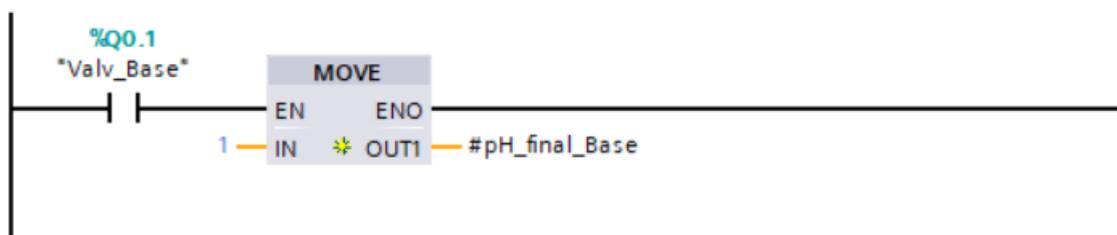


Figura 44. Segmento de programación 6.2 del autómata 2.

Segmento 7: en este Segmento hay un bloque de funciones que dentro lleva una lógica que al detectar que el pH_final este en un rango mayor a 7.4 hasta 14 escalada a unidades enteras de 75 -140 para simulará la abertura de la válvula asignada al depósito de sustancia Acida cuando la condición dentro del bloque se cumplan.

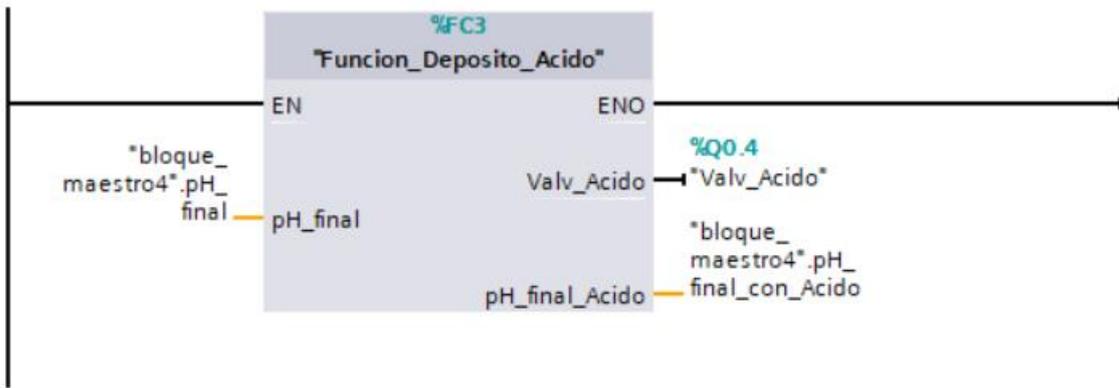


Figura 45. Segmento de programación 7 del autómata 2.

Este al igual que el segmento 6 es un bloque de Funciones. Dentro está compuesto por los siguientes bloques.

1) Segmento 7.1

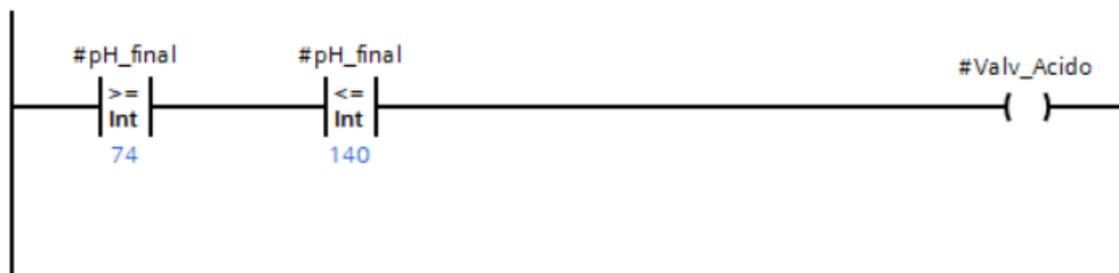


Figura 46. Segmento de programación 7.1 del autómata 2.

2) Segmento 7.2

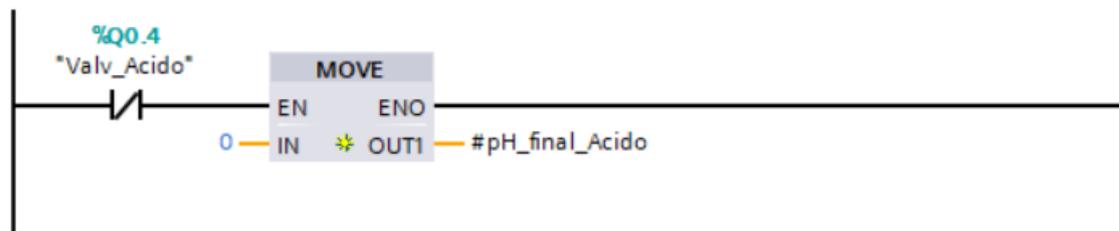


Figura 47. Segmento de programación 7.2 del autómata 2.

3) Segmento 7.3

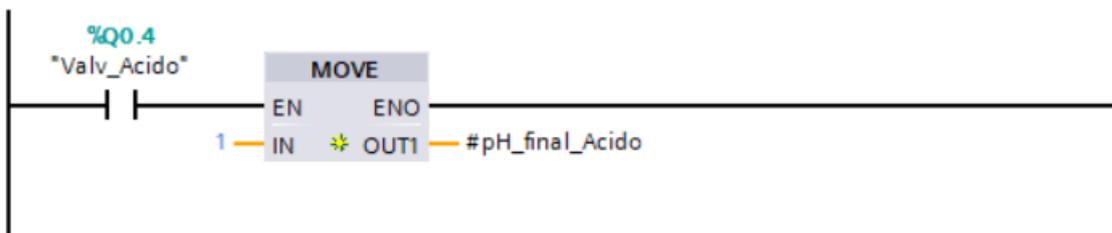


Figura 48. Segmento de programación 7.3 del autómata 2.

Segmento 8: en esta sección está el módulo de comunicación Modbus_Comm_Load utilizado para establecer la comunicación y parametrizar la conexión con los parámetros entre el maestro y el esclavo.

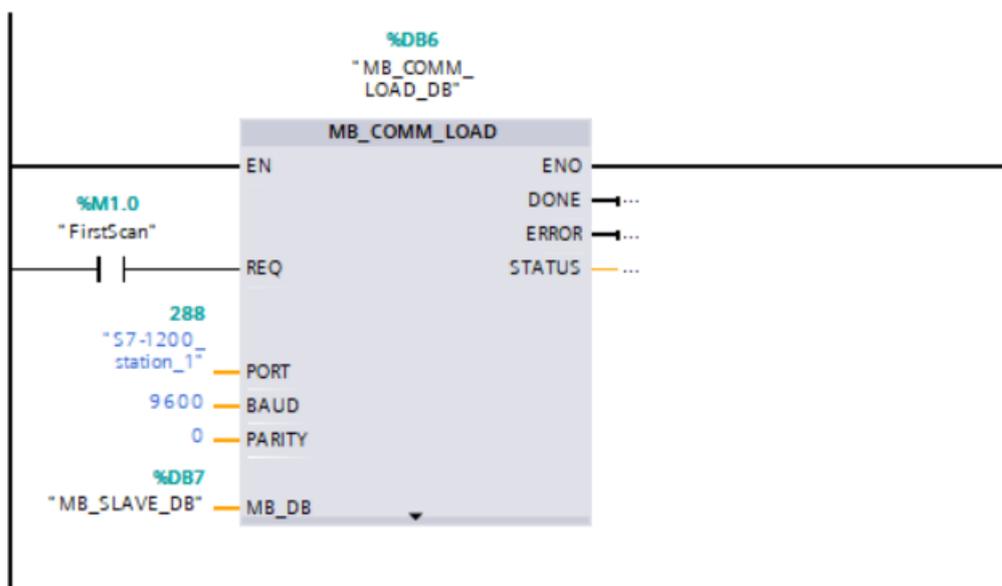


Figura 49. Segmento de programación 8 del autómata 2.

Segmento 9: en este segmento está el bloque MODBUS ESCLAVO utilizado para definir la dirección de estación MODBUS-RTU y para especificar el área de almacenamiento de datos local para la transferencia de datos en el registro de retención.

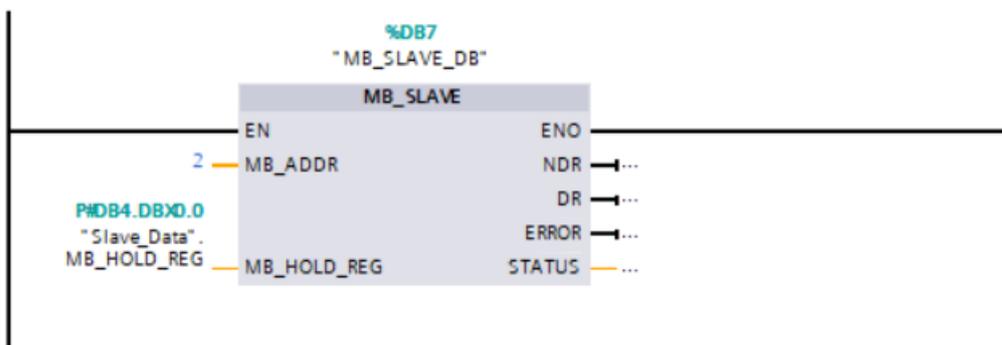


Figura 50. Segmento de programación 9 del autómata 2

Segmento 10: esta parte es utilizada para enviar la variable pH antes de ser neutralizada al autómata central. Estos datos son transferidos a través de la conexión MODBUS.

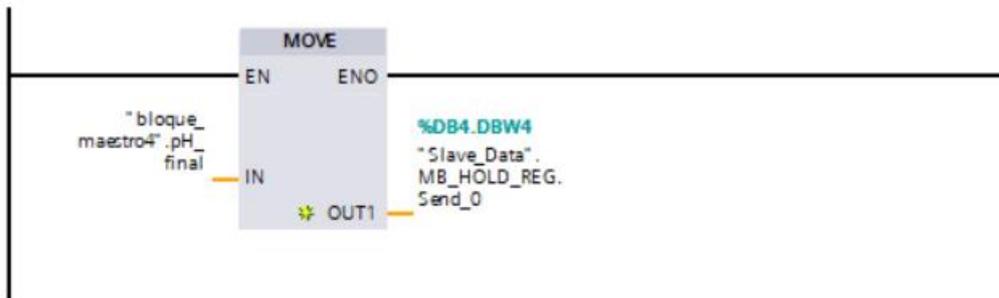


Figura 51. Segmento de programación 10 del autómata 2

Segmento 11: Envío de la variable pH cuando es detectado por el bloque del Segmento que se encarga de abrir la válvula de sustancias Base.

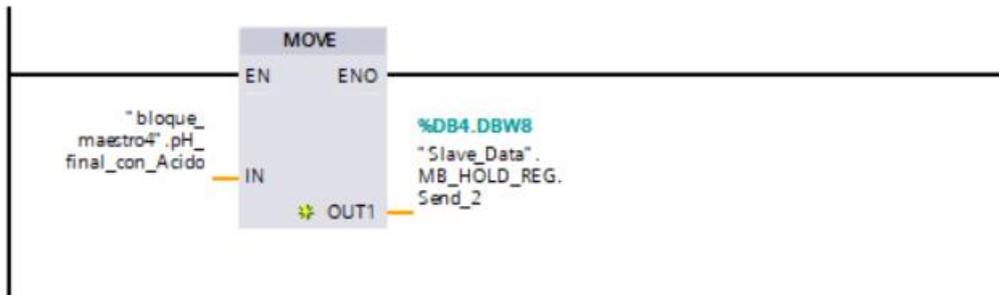


Figura 52. Segmento de programación 11 del autómata 2

Segmento 12: Envío de la variable pH cuando es detectada por el bloque del Segmento que se encarga de abrir la válvula de sustancias Base.

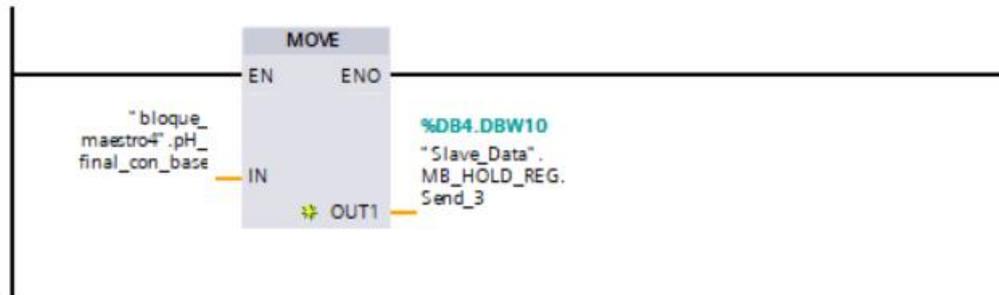


Figura 53. Segmento de programación 12 del autómata 2

Segmento 13: Envío de la variable que almacena la cantidad de litros existente en el depósito 2 al autómata central.

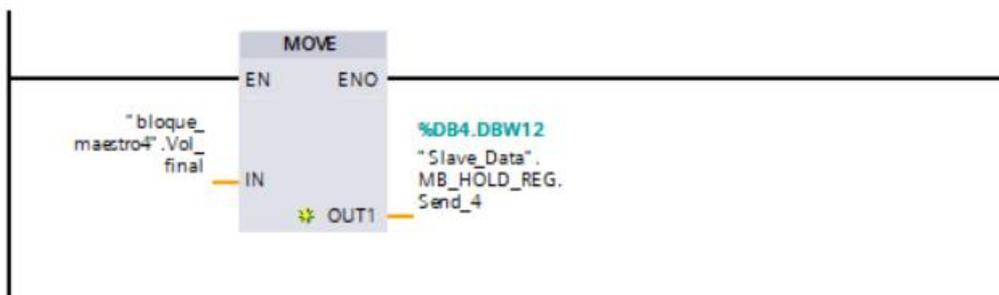


Figura 54. Segmento de programación 13 del autómata 2

Segmento 14: envía la variable temperatura correspondiente al depósito 3, variable enviada desde el autómata 3 y recibida en el autómata 2 a través de comunicación PROFIBUS, para luego ser enviado desde este autómata al autómata central a través de MODBUS.

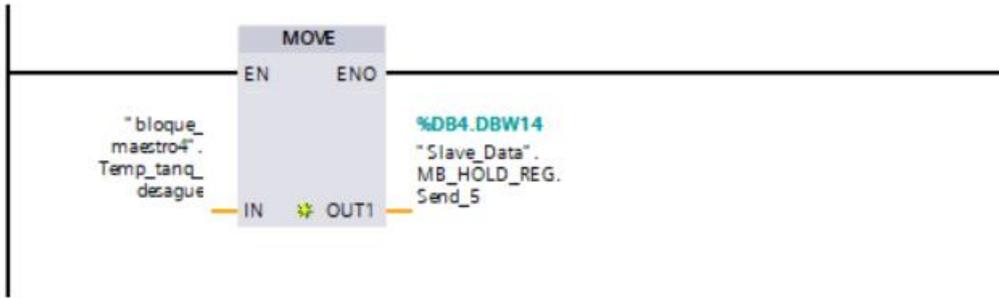


Figura 55. Segmento de programación 15 del autómata 2.

5.3 SEGMENTOS DE PROGRAMACION AUTÓMATA DEPOSITO 3

Este autómata está asignado para el depósito 3, depósito donde se deposita el agua ya neutra para luego ser enviada a su lugar de destino, no antes cumpliendo que su temperatura de este líquido monitorizada sea la indicada en el autómata. Este cuenta con 3 Segmentos que se detallan a continuación.

Segmento 1: este Segmento cuenta con un bloque de función empleado para detectar la señal enviada por el sensor de temperatura.

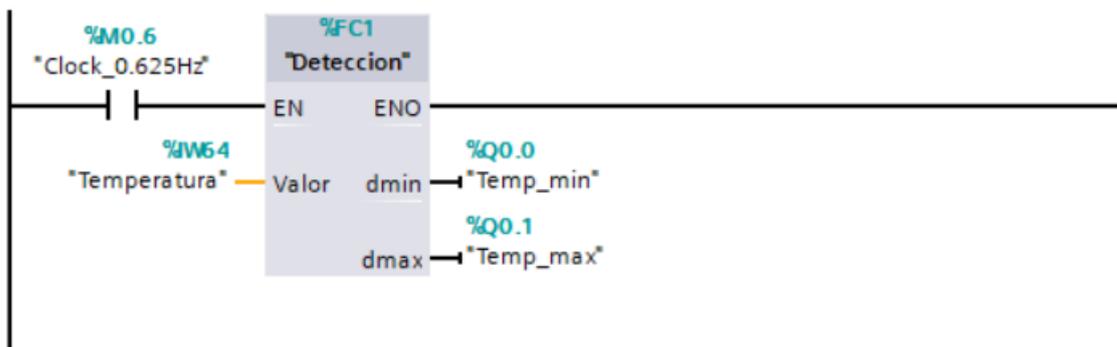


Figura 56. Segmento de programación 1 del autómata 3

Este bloque de funciones está conformado por la siguiente programación.

Segmento 1.1: Bloque NORM_X recibe el valor de la señal en la entrada analógica del PLC y lo normaliza entre un numero de 0 a 27648, luego lo transfiere a la variable que se le ha asignado para para su almacenamiento.

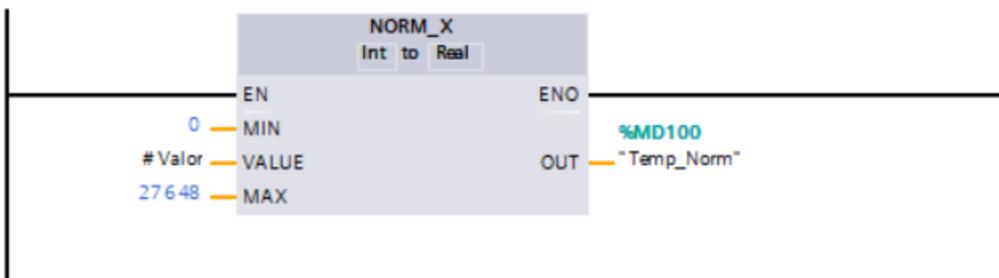


Figura 57. Segmento de programación 1.1 del autómata 3

Segmento 1.2: Escalando el valor de la temperatura de 0 a 200 °C.

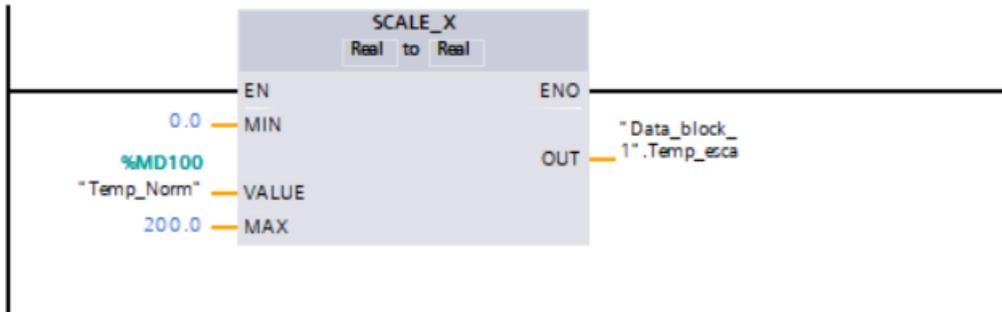


Figura 58. Segmento de programación 1.2 del autómata 3

Segmento 2: Utilizado para convertir el tipo de datos de Word a Entero.

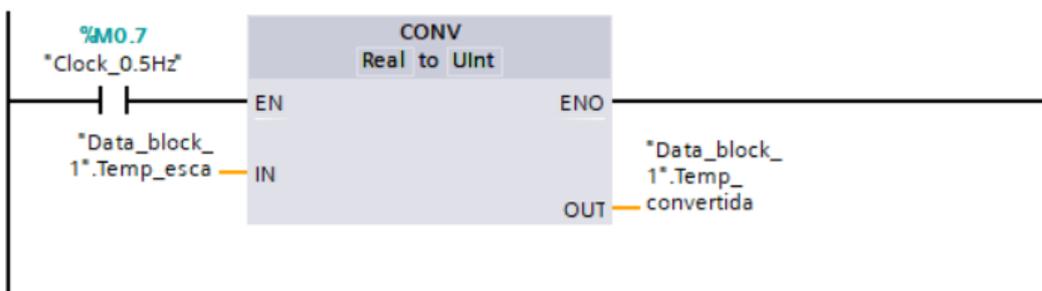


Figura 59. Segmento de programación 2 del autómata 3

Segmento 3: en este Segmento se está pasando la variable que almacena el dato de temperatura adquirido por los sensores a una salida especificada para la transferencia de datos via comunicación PROFIBUS.

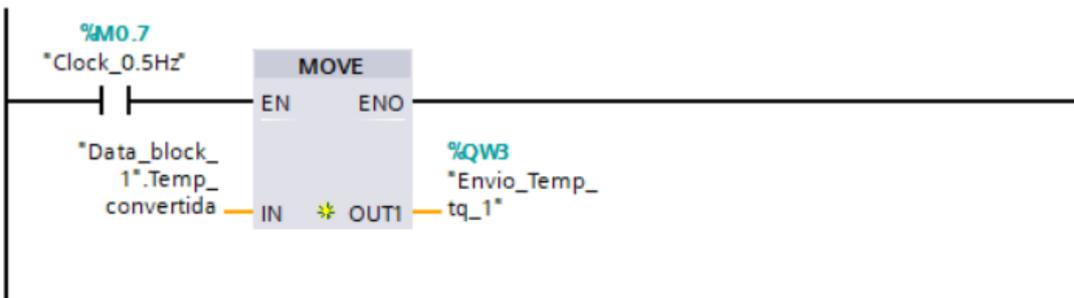


Figura 60. Segmento de programación 3 del autómata 3.

Segmento 4: es el Segmento encargado de abrir la válvula de salida cuando la temperatura está en el rango de valor puesto en la lógica programada.

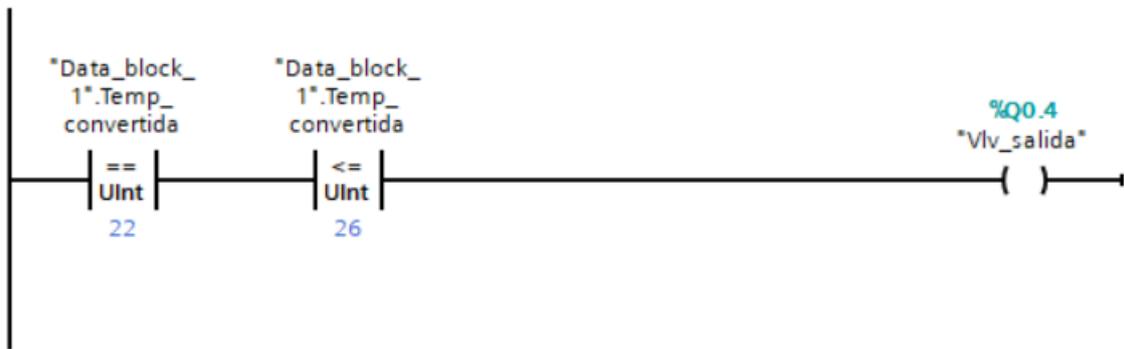


Figura 61. Segmento de programación 4 del autómata 3

5.4 SEGMENTOS DE PROGRAMACION DEL PLC CENTRAL

Segmento 1: en esta sección está el módulo de comunicación Modbus_Comm_Load utilizado para establecer la comunicación y parametrizar la conexión con los parámetros entre el maestro y el esclavo.

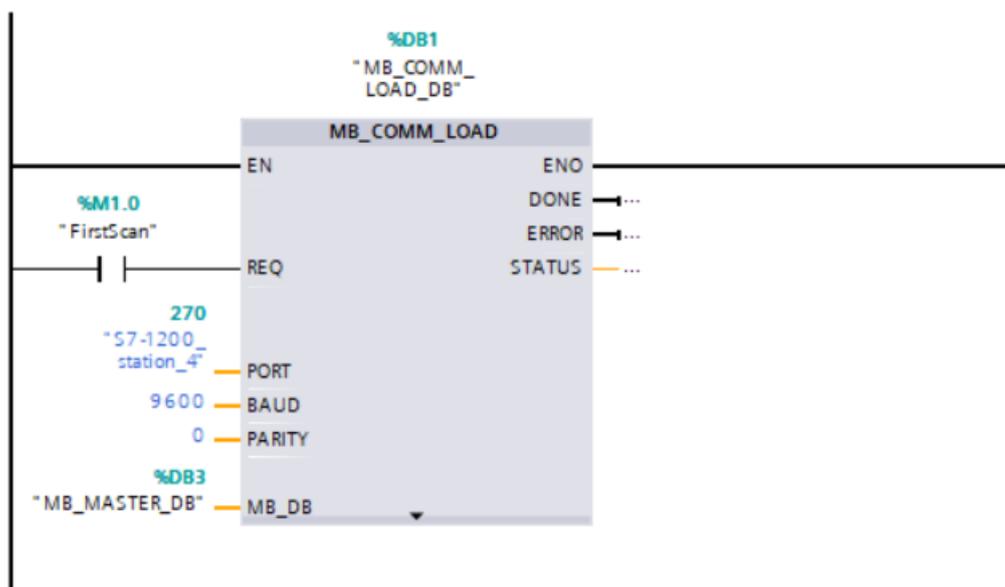


Figura 62. Segmento de programación 1 del autómata 4

Segmento 2: aquí se ha se utiliza el bloque MODBUS_MASTER para definir como MODBUS dominante el módulo de comunicación seleccionado con el bloque de configuración MODBUS_COMM_LOAD. Aquí el bloque está en modo escritura, es decir escribir el valor para ser visualizado en el autómata esclavo.

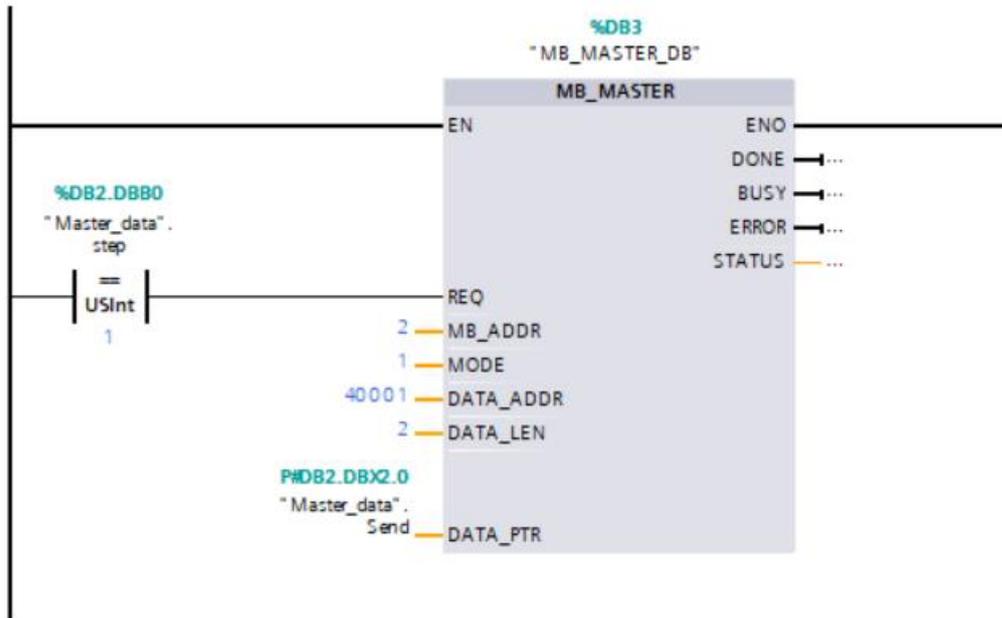


Figura 63. Segmento de programación 2 del autómata 4

Segmento 3: en este Segmento se envía un valor al autómata esclavo, donde los se podrá visualizar en la tabla de variables.

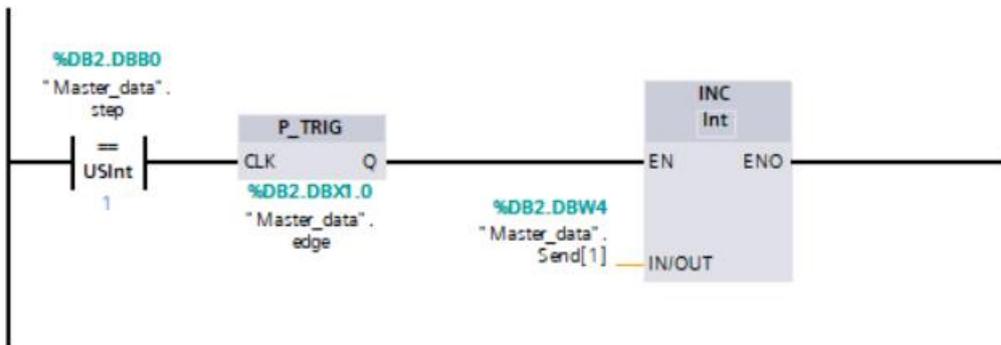


Figura 64. Segmento de programación 3 del autómata 4.

Segmento 4: al cumplirse la condición pondrá en set la bobina signada por lo que activara el contacto de REQ del bloque MB_MASTER.

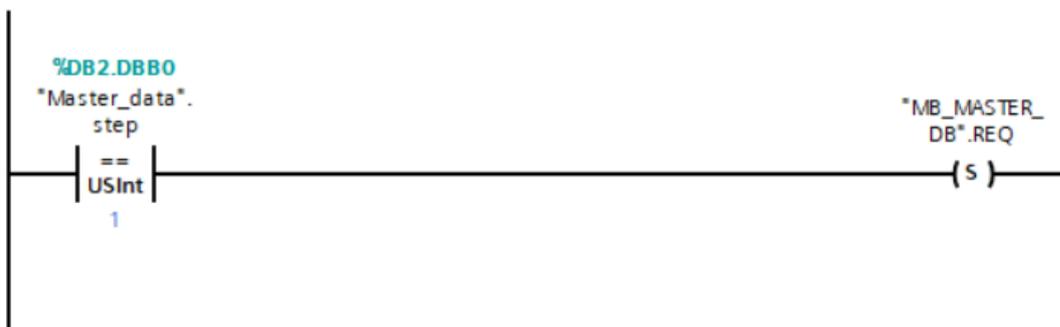


Figura 65. Segmento de programación 4 del autómata 4.

Segmento 5: en el Segmento 5 se comprueba si la comunicación se ha realizado (bit DONE) o se ha producido un error (bit ERROR). Si se cumple alguna de estas dos condiciones se actualiza la variable (step) de 1 a 2.

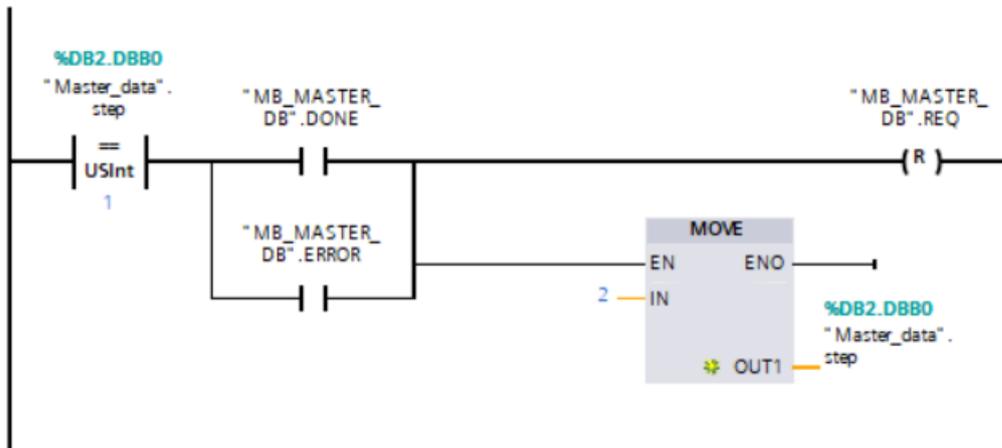


Figura 66. Segmento de programación 5 del autómeta 4

Segmento 6: Aquí está el bloque MODBUS_MASTER, en este Segmento se ha utilizado en modo lectura, para leer las variables enviadas desde el autómeta esclavo.

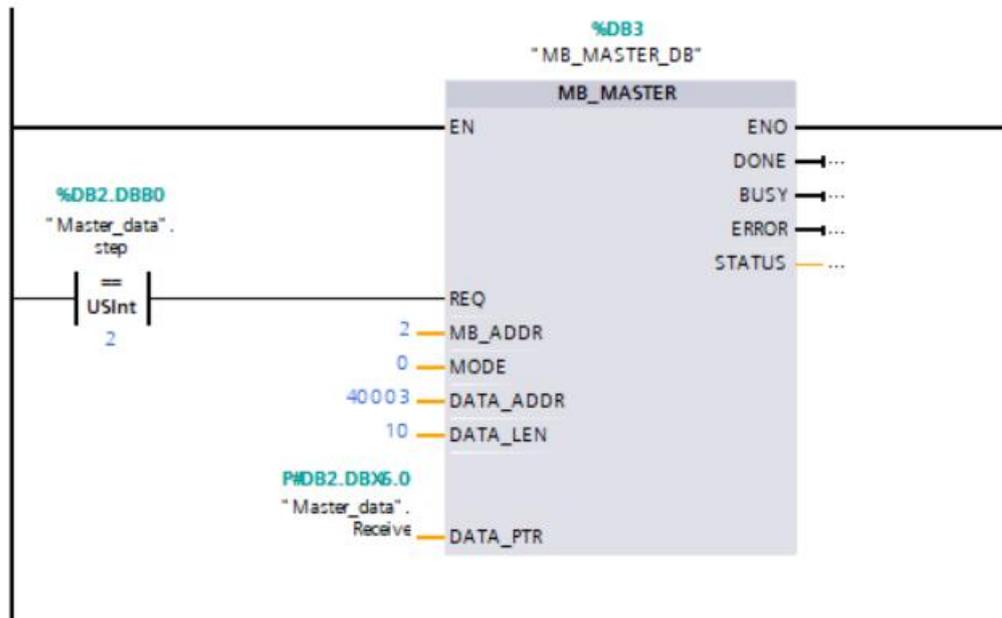


Figura 67. Segmento de programación 6 del autómeta 4

Segmento 7: al cumplirse la condición pondrá en set la bobina signada por lo que activara el contacto de REQ del bloque MB_MASTER.



Figura 68. Segmento de programación 7 del autómeta 4

Segmento 8: en este segmento se comprueba que la comunicación se ha realizado (bit DONE) o se ha producido un error (bit ERROR). Al cumplirse una de estas dos condiciones se actualiza la variable (step), que pasará de 2 a 1.

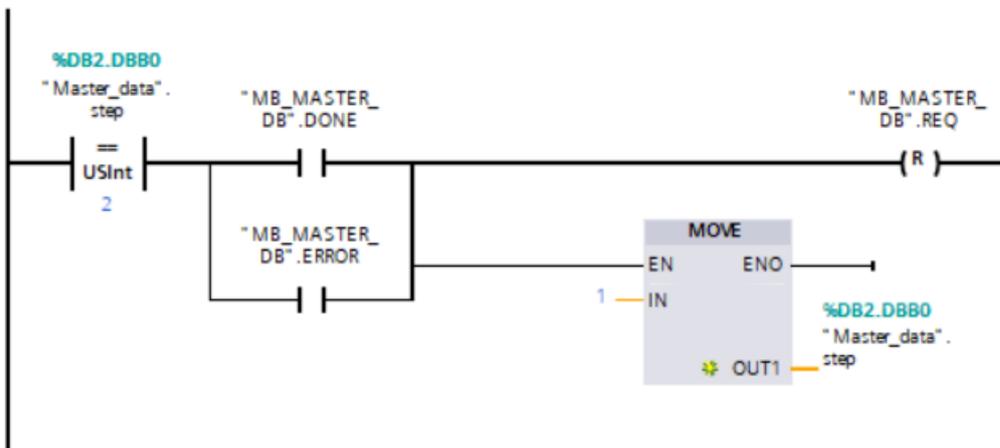


Figura 69. Segmento de programación 8 del autómeta 4

Segmento 9: en este Segmento se está moviendo la variable recibida en el autómeta central que ha llegado desde el autómeta esclavo. Esta variable luego de ser movida se almacenara en una variable para uso en la visualización en tiempo real en una página web.

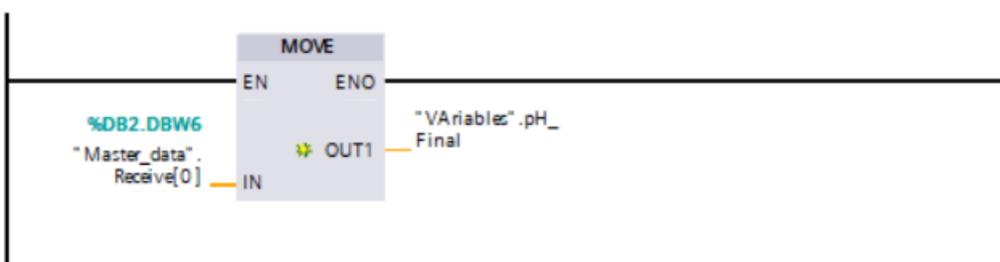


Figura 70. Segmento de programación 9 del autómeta 4

Segmento 10: aquí se hace el movimiento de variable recibida a variable propia para ser visualizada en la página web.

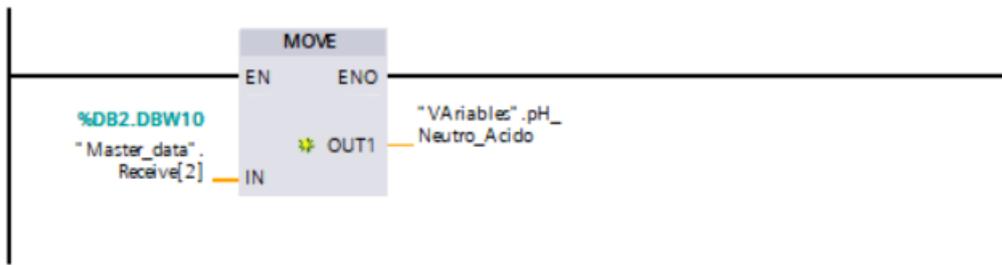


Figura 71. Segmento de programación 10 del autómeta 4

Segmento 11: aquí se hace el movimiento de variable recibida a variable propia para ser visualizada en el HMI.

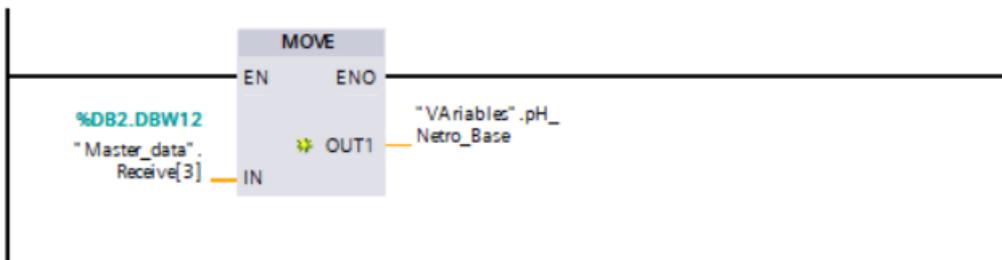


Figura 72. Segmento de programación 11 del autómeta 4

Segmento 12: aquí se hace el movimiento de variable recibida a variable propia para ser visualizada en el HMI.

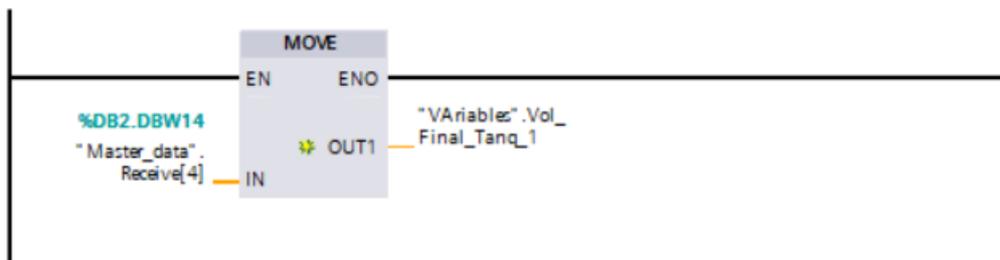


Figura 73. Segmento de programación 12 del autómeta 4

Segmento 13: aquí se hace el movimiento de variable recibida a variable propia para ser visualizada en el HMI.

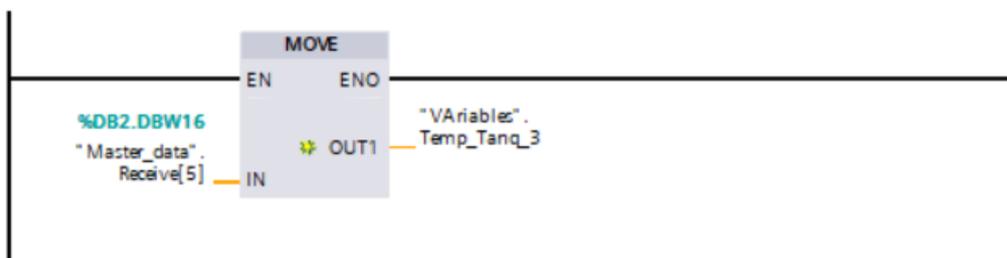


Figura 74. Segmento de programación 13 del autómeta 4

Segmento 14: aquí se ha añadido el bloque de instrucciones WWW para arrancar el servidor web de la CPU, o mejor dicho, sincroniza las páginas web declaradas por el usuario con el programa de usuario en la CPU. Las páginas web definidas, junto con el servidor web de la CPU,

ofrecen la posibilidad de acceder con un navegador web a páginas web de libre configuración de la CPU.

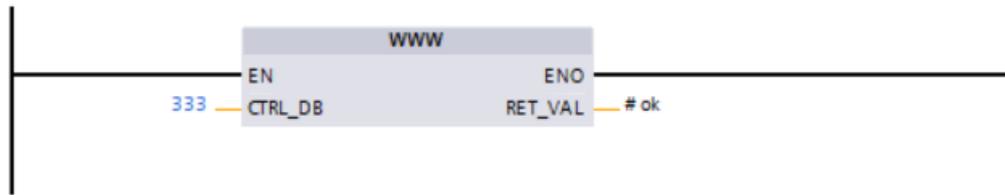


Figura 75. Segmento de programación 14 del autómata 4

5.5 INTERFAZ GRAFICA

Podemos ver la interfaz gráfica de este proyecto como una ventana que permite comunicarse con la máquina. Esta interfaz puede estar contenida en dispositivo de pantalla táctil, página web o en el computador (esta dos últimas utilizadas en este proyecto). Las señales son introducidas a nuestra página web.

El servidor web del autómata S7-1200 ofrece acceso a través página web a datos de la CPU y a los datos del proceso llevado a cabo. Para su configuración, se debe activar la opción de servidor web, seleccionando la CPU en la **Vista de dispositivos** y luego hacemos clic en **Propiedades --> General --> Servidor web --> General** y en la casilla **Activar servidor web en el módulo**.

En este punto, si compilamos y cargamos cualquier programa en el PLC, seremos ya capaces de acceder a la CPU desde un navegador web (tecleando la dirección IP de la CPU en la barra de direcciones) y acceder a propiedades de la CPU y monitorizar el valor de las variables definidas.

El código de programación cargado en la CPU del autómata para poder visualizar las variables en una página web creado en el lenguaje HTML y CCS es el siguiente:

5.5.1 CODIGO EN HTML

```
<html>
:= "VAriables".pH_Final
:= "VAriables".pH_Netro_Base
:= "VAriables".pH_Neutro_Ácido
:= "VAriables".Vol_Final_Tanq_1
:= "VAriables".Temp_Tanq_3

<head>
<meta http-equiv="refresh" content="10">
<title>S7-1200 Web Server</title>
<link href="estilo.css" rel="stylesheet" type="text/css">

<style type="Text/css">
body {
margin-left: 0px;
margin-top: 0px;
margin-right: 0px;
margin-bottom: 0px;
```

```

        text-align: center;
        font-weight: bold;
    }
</style>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>
<body>
<div class='space'></div>
<header><hgroup>
<h1 class="lx"><strong> PANEL DE VISUALIZACION PARA EL CONTROL DE pH</strong>
</h1>
<table width="300" height="239" align="center" cellpadding="0" cellspacing="2">
    <td>

        <tr style="height:2%;">
            <td bgcolor="#999999">
                <p class="negrit"> pH:</p>
            </td>
            <td bgcolor="#999999">
                <p class="valores" style="margin-bottom: 5px; font-weight: bold;">
:= "VAriables".pH_Netro_Base: </p>
            </td>
        </tr>

        <tr style="height:2%;">
            <td bgcolor="#999999">
                <p class="negrit">pH con Base:</p>
            </td>
            <td bgcolor="#999999">
                <p class="valores" style="margin-bottom: 5px; font-weight: bold;">
:= "VAriables".pH_Netro_Base: </p>
            </td>
        </tr>

        <tr style="height:2%;">
            <td bgcolor="#999999">
                <p class="negrit">pH con Ácido:</p>
            </td>
            <td bgcolor="#999999">
                <p class="valores" style="margin-bottom: 5px; font-weight: bold;">
:= "VAriables".pH_Neutro_Ácido: </p>
            </td>
        </tr>

        <tr style="height:2%;">
            <td bgcolor="#999999">
                <p class="negrit">Volumen Recipiente 1:</p>
            </td>
            <td bgcolor="#999999">

```

```
<p class="valores" style="margin-bottom: 5px; font-weight: bold;">
:= "Variables".Vol_Final_Tanq_1: </p>
</td>
</tr>
```

```

<tr style ="height:2%;">
<td bgcolor ="#999999">
  <p class=" negrit "> Temperatura DEPOSITO 3:</p>
  </td>
  <td bgcolor="#999999">
    <p class=" valores" style="margin - bottom: 5px ; font-weight: bold;">
:= "VARIABLES".Temp_Tanq_3: </p>
  </td>
</tr>

<form method="POST" action="">
  <tr style="height: 2%;" >
  <td style=" width=25%; border-
top-
style: Solid ; border-
top-
width:
2px; border-
top-
color:#ffffff;">
  </td>

</form>
</td>
</tr>
</table>

</ body>
</ html>

```

5.5.2 CODIGO DE ESTILO CCS

```

harset "utf-8";
/* CSS Document */

body{background -image:url(inds.jpg);
background - repeat:no - repeat;

background-size:100%, 100%;
margin-bottom: 0px;
margin-top: 0px;
margin-left: 0px;
margin-right: 0px;

}
.space
{
  height::500px;
  border:solid;

```

```
}
```

```
h1 {  
  color :#903;  
  text-decoration:underline;  
}
```

```
p  
{  
  Color :#00C;  
}
```

```
.valores  
{  
  color:#FFF;  
  font-weight:500;  
}
```

```
.negrit  
{  
  font-weight: bold;  
  font-size: 24px;  
  color: #00F;  
}
```

```
.lx {  
  Text -align: center;  
}  
.lx strong {  
  Fon t-weight: bold;  
}
```

Las siguientes figuras muestran la página web diseñada para este proyecto, donde se podrá visualizar las variables leídas en la CPU, a su vez dentro de la misma figura se puede ver su equivalencia en Voltios, que es la señal.

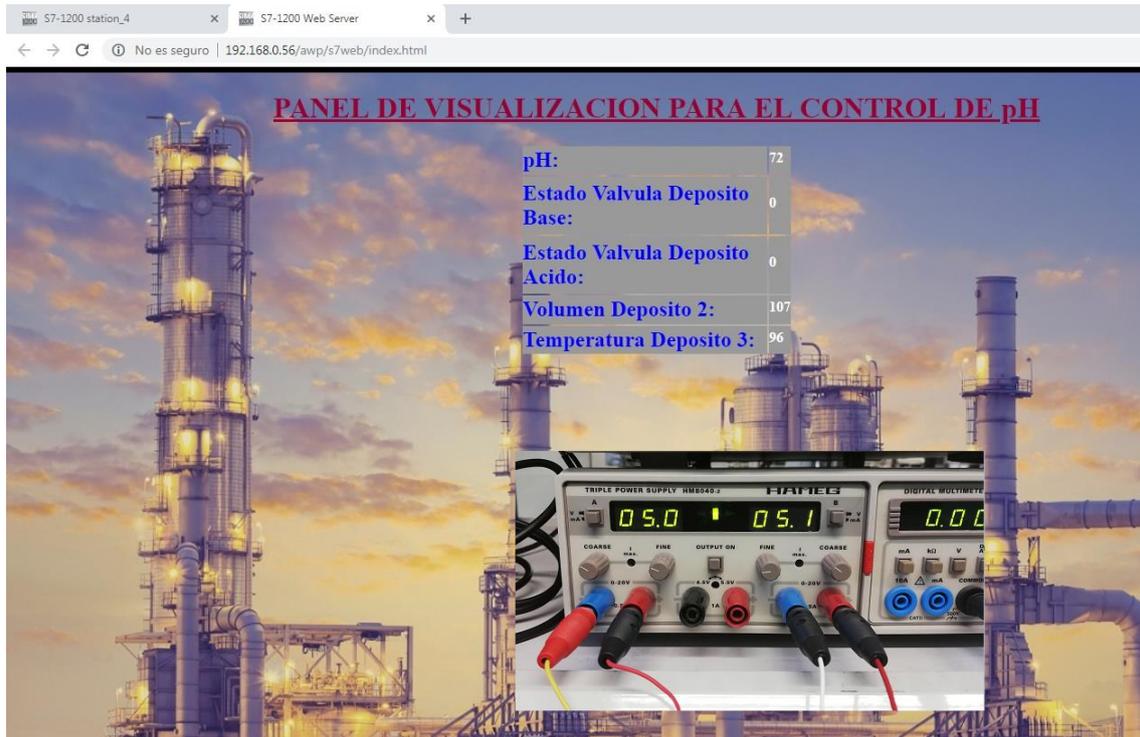


Figura 76. Interfaz gráfica proyecto, mostrando el pH neutro.



Figura 77. Interfaz gráfica proyecto, mostrando el pH ácido.

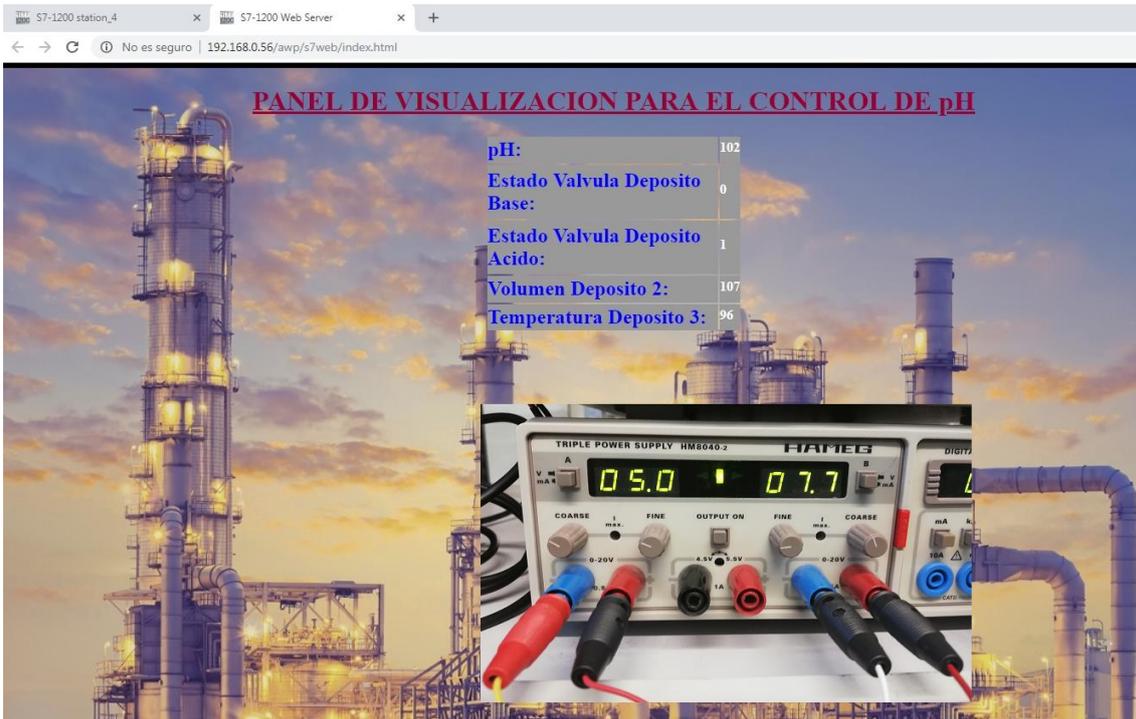


Figura 78. Interfaz gráfica proyecto, mostrando el pH alcalino.

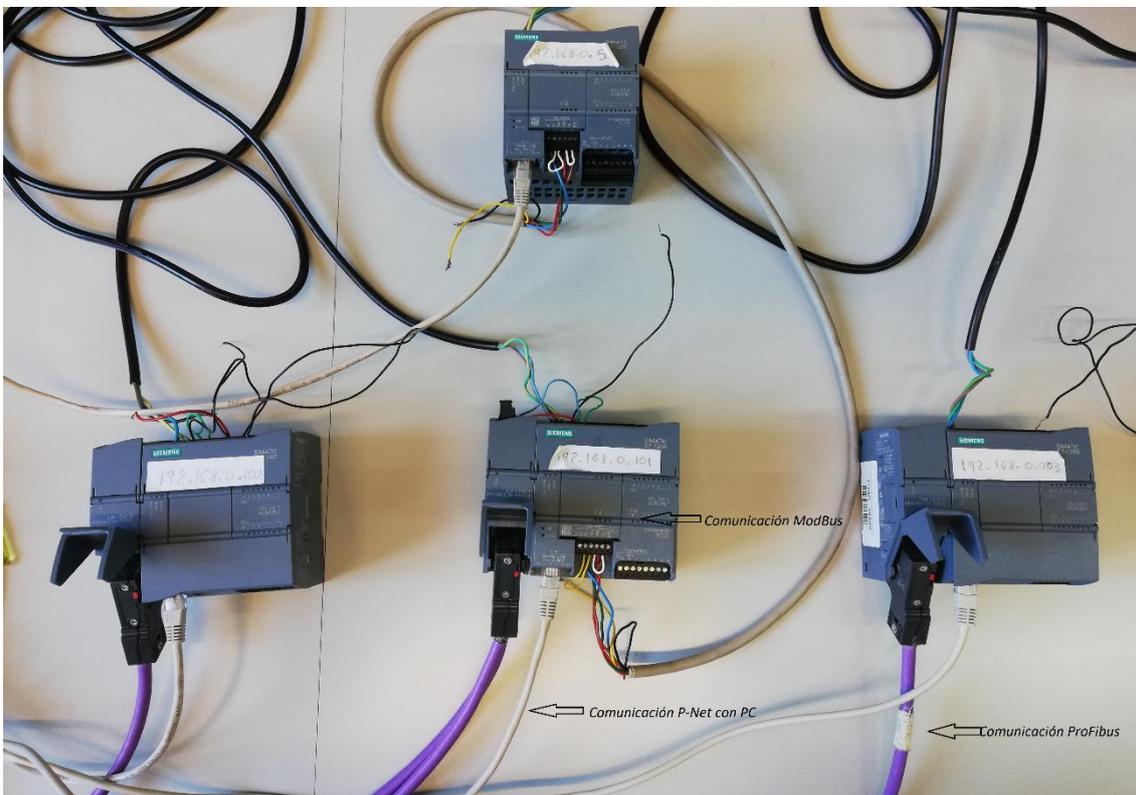


Figura 79. Conexión de PLC's utilizados en el proyecto.

CONCLUSIONES

- 1) Al determinar las características del hardware de automatización llamado Siemens s7 1200, hemos podido conocer sobre sus grandes funciones en el control y monitorización de variables gracias a sus entradas y salidas programables.
- 2) Antes de implementar con el autómeta Siemens 1200, fue necesario estudiar los manuales del fabricante donde nos explican todo lo relacionado al hardware y parte del software.
- 3) Con el estudio del programa TIA PORTAL, he notado que es una poderosa herramienta de programación debido a que permite de una forma muy intuitiva configurar los equipos necesarios, en este caso lo de este proyecto, implementar visualizaciones y también el control de todo el sistema mediante una interfaz gráfica de tu preferencia como pantalla táctil o a través de un servidor web. Este programa incluye también depuradores de fallos muy eficaces para detectar cualquier posible fallo que haga que el sistema no funcione, descubriéndolo de la manera más sencilla que se mencionan a continuación: Haciendo clic en cualquier elemento sea entradas o salidas, marcas o clic en herramientas, referencias cruzadas donde vamos a poder visualizar los Segmentos donde aparece el error y su acceso.
- 4) Se ha desarrollado un sistema de fácil implementación que permite el monitoreo, registro y control de las variables como son la Temperatura, nivel y pH) de una planta de tratamiento de aguas residuales.
- 5) Se puede concluir que este proyecto es una herramienta versátil y muy funcional para de forma rápida saber los comportamientos de estas variables físicas del sistema y mantener un nivel óptimo del valor de pH dentro de la planta de tratamiento de aguas residuales.
- 6) Con su funcionamiento y buena comunicación entre el sistema y el servidor web nos permitirá ver correctamente los valores de las variables registradas censadas por los distintos sensores.
- 7) Debido al modo en que se ha desarrollado esta programación es posible elegir diferentes sensores y actuadores e instalarlos una manera muy sencilla.

BIBLIOGRAFÍA

- [1] «INTRODUCCION A LAS REDES DE COMUNICACIÓN,» [En línea]. Available: <ftp://ftp.unicauca.edu.co/Facultades/FIET/DEIC/Materias/Redes%20Industriales/Redes/Conferencias/Capitulo%201.pdf>.
- [2] «Buses de Campo,» [En línea]. Available: http://automata.cps.unizar.es/Historia/Webs/buses_de_campo.htm.
- [3] «What is Profibus?,» [En línea]. Available: [http://www.hiport.co.uk/Images/What%20is%20Profibus\(1\).htm](http://www.hiport.co.uk/Images/What%20is%20Profibus(1).htm).
- [4] J. Andrade, «SlideShare,» [En línea]. Available: <https://es.slideshare.net/panchoraf/exposicion-profibus-fms-final>.
- [5] Siemens, «Automation24,» [En línea]. Available: <https://www.automation24.es>.
- [6] Siemens, Manual del sistema SIMATIC CONTROLADOR PROGRAMABLE S7 1200, 2012.
- [7] I. Gutiez, «Normas básicas para realizar un cableado Profibus.,» [En línea]. Available: <https://programacion Siemens.com>.
- [8] «PROFIBUS,» [En línea]. Available: <http://www.etitudela.com/celula/downloads/2profibus.pdf>.
- [9] «LogicBus,» [En línea]. Available: <http://www.logicbus.com.mx/Modbus.php>.
- [10] A. Candelas, «Automatización Avanzada-Comunicación con RS-485 y MODBUS,» 11 10 2011. [En línea]. Available: <https://rua.ua.es/dspace/bitstream/10045/18990/1/AA-p3.pdf>.
- [11] WEG, «Manual de la Comunicación Modbus,» WEG, 12 2012. [En línea]. Available: <http://ecatalog.weg.net/files/wegnet/WEG-srw01-manual-de-la-comunicacion-modbus-rtu-10000521680-4.0x-manual-espanol.pdf>.
- [12] Johannes, «GRAF CET Workbook,» [En línea]. Available: <https://www.tia-expert.com/grafcet-studio/>.