



**UNIVERSIDAD POLITÉCNICA SALESIANA  
SEDE GUAYAQUIL**

**CARRERA DE INGENIERÍA ELECTRÓNICA**

Proyecto técnico previo a la obtención del Título de:

**INGENIERO ELECTRÓNICO**

TEMA:

“DISEÑO Y DESARROLLO DE UN MODULO DIDACTICO DE INSTRUMENTACIÓN  
CON CONEXIÓN AL SERVIDOR WEB NODE-RED PARA EL MONITOREO Y CONTROL  
DEL PROCESO CON MINI PLC LOGO V 8.3 Y RASPBERRY PI 4.”

**AUTORES:**

Quispillo Farez Luis

Quispillo Fares Alex

**TUTOR:**

PhD. Víctor Huilcapi Subia

Guayaquil – Ecuador

2023

## **CERTIFICADO DE RESPONSABILIDAD Y AUDITORÍA DEL TRABAJO DE TITULACIÓN**

Nosotros, Alex Mauricio Quispillo Farez con documento de identificación No 0927072652 y Luis Geovanny Quispillo Farez con documento de identificación No 0921926226; manifestamos que:

Somos los autores y responsable del presente trabajo; y, autorizamos a que sin fines de lucro la **Universidad Politécnica Salesiana** pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Guayaquil, 26 de septiembre del 2023

Atentamente,



Alex Quispillo Fares

C.I. 0927072652



Luis Quispillo Farez

C.I. 0921926226

**CERTIFICADO DE CESIÓN DE DERECHO DE AUTOR DEL TRABAJO DE TITULACIÓN  
A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Alex Mauricio Quispillo Fares con documento de identificación No. 0927072652 y Luis Geovanny Quispillo Farez con documento de identificación No. 0921926226, Manifestamos nuestra voluntad y por medio del presente documento cedemos a la **Universidad Politécnica Salesiana** la titularidad sobre los derechos patrimoniales en virtud de que somos autores de trabajo de grado titulado: **“DISEÑO Y DESARROLLO DE UN MÓDULO DIDÁCTICO DE INSTRUMENTACIÓN CON CONEXIÓN AL SERVIDOR WEB NODE-RED PARA EL MONITOREO Y CONTROL DEL PROCESO CON MINI PLC LOGO V 8.3 Y RASPBERRY PI 4”**, cual ha sido desarrollado para optar por el título de: **INGENIERO ELECTRÓNICO**, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En la aplicación a lo determinado en la Ley de Propiedad Intelectual, en condición de autor me reservo los derechos morales de la obra antes citadas. En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.



Alex Quispillo Fares

C.I. 0927072652



Luis Quispillo Farez

C.I. 0921926226

## CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Víctor Manuel Huilcapi Subia con documento de identificación No 0917206294, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **“DISEÑO Y DESARROLLO DE UN MÓDULO DIDÁCTICO DE INSTRUMENTACIÓN CON CONEXIÓN AL SERVIDOR WEB NODE-RED PARA EL MONITOREO Y CONTROL DEL PROCESO CON MINI PLC LOGO V 8.3 Y RASPBERRY PI 4”**, realizado por Alex Mauricio Quispillo Fares con documento de identificación No 0927072652 y por Luis Geovanny Quispillo Farez con documento de identificación No 0921926226, obteniendo un producto que cumple con los objetivos de diseño de aprobación, informe final y demás requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerado como trabajo de titulación.

Guayaquil, 26 de septiembre del 2023



PhD. Víctor Huilcapi Subia

TUTOR DEL PROYECTO

## **DEDICATORIA**

Este presente trabajo es dedicado a JESÚS mi salvador, en él encontré mi refugio, que en medio de la circunstancia ha sido mi fortaleza, que en medio de las heridas de mi pasado ha sido mi sanador, por amor y fe sigo su camino y su verdad, a pesar de mis faltas Él perdona y me alienta a seguir.

A mis padres, son una fuente de inspiración viva, las cosas que alcanzaron fueron a base de esfuerzo y dedicación, sin cada uno de ellos como apoyo no hubiese podido tener la fortaleza y ánimo de seguir desde el inicio de esta carrera. Después de todo ellos tuvieron conmigo y a pesar de las dificultades que haya existido, a ellos más que nunca quiero dedicarles y honrarlos en estos párrafos y con el resultado del trabajo de titulación.

A mis hermanos Luis, Danny y mis hermanas Nelly, Shaden que de una u otra manera me dieron su apoyo emocional e incondicional.

A cada una de las increíbles personas que fueron parte de mi proceso de formación en lo profesional y personal.

Y a mi querida novia Génesis que ha estado alentado y apoyando en todo este proceso.

**Alex Quispillo Fares**

## **AGRADECIMIENTO**

Una vez más agradezco a Dios por acompañarme en transcurso de mi vida, por la familia que me ha dado, y por las personas que ha puesto en mi vida para formación en lo personal y profesional ya que nada es casualidad sino propósito.

A mi tutor PhD. Víctor Huilcapi por el apoyo en este trabajo de titulación, paciente y correctivo para mejora de este proyecto fuese posible.

A mi hermano y futuro Ingeniero Luis Quispillo Farez y su esposa Ing. Ana Vega sin su apoyo no fuese posible el desarrollo de este proyecto desde el inicio.

A mi colega, hermano y amigo Ing. Ricardo Carrera Brown por sus asesorías a lo largo del tiempo de este proyecto.

Al Ing. Danny Ricaurte por la oportunidad de continuar con mis estudios de formación en esta carrera en la empresa donde elaboraba como auxiliar de planta tiempo atrás, desde la distancia le agradezco por todo su apoyo.

Y por último a cada uno de los docentes que fueron partes de mi formación de carrera ingeniería en la Universidad Politécnica Salesiana, agradecidos con cada uno de ustedes.

**Alex Quispillo Fares**

## **DEDICATORIA**

A Dios por sus incontables bendiciones a lo largo de mi vida y la oportunidad de terminar esta carrera de ingeniería.

A mi amada esposa Ing. Ana Vega Tutiven por su gran amor y apoyo incondicional que he recibido a lo largo de nuestra unión, siempre será mi complemento especial que Dios me ha dado para cuidado y protección.

A mis padres que son una inspiración real en mi vida por todo el esfuerzo de guiarme y sustentarme a lo largo de mi carrera en mi juventud, ahora en mi etapa como hombre de hogar los honro día a día con las bendiciones que Dios me ha dado.

**Luis Quispillo Farez**

## RESUMEN

Año	Título	Alumnos	Director de proyecto técnico	Tema de proyecto técnico
2023	Ingeniero Electrónico	Alex Mauricio Quispillo Fares Luis Geovanny Quispillo Farez	PhD. Víctor Huilcapi	Diseño y desarrollo de un módulo didáctico de instrumentación con conexión al servidor web Node-RED para el monitoreo y control del proceso con mini PLC LOGO V 8.3 y Raspberry Pi 4.

La universidad politécnica salesiana, con sede en Guayaquil, es una institución académica que brinda servicio de educación superior, que consta con una infraestructura adecuadas para las prácticas en laboratorios, apropiadas para que los estudiantes puedan expandir sus conocimientos, de lo teórico a lo práctico.

El objetivo del proyecto técnico va dirigido a los estudiantes de la Carrera de Ingeniería en Electrónica con Mención en Sistemas Industriales, con la finalidad de conocer, aprender y simular las diferentes practicas proyectado a la industria 4.0 e internet de las cosas (IoT), sobre un proceso de llenado y recirculación de líquido de generalmente utilizado para explicación de sistema automatizados.

La planta didáctica consta en el control de nivel y temperatura, por lo cual, se implementó un sensor ultrasonido y un relé de nivel en los dos tanques de acero inoxidable, además una PT-100 y la resistencia calefactora para el calentamiento del líquido de este caso el agua, el control de temperatura será por PWM. por último, el Mini PLC LOGO 8 y el Miniordenador Raspberry Pi 4 son partes esenciales del módulo para mostrar el control del proceso y el accionamiento de esta por medio del servidor web Node-RED. El correcto funcionamiento del proyecto se basó con experimentos y pruebas que se ejecutaron con los equipos presentados anteriormente, los cuales se encuentran en aplicaciones de industrias, y su respectiva comunicación en tiempo real del



servidor Node-RED que es una herramienta basada en editor de flujo en el cual se puede agregar o suprimir nodos y haya vinculación unos a otros.

El proyecto técnico presentado brevemente se denomina **DISEÑO Y DESARROLLO DE UN MÓDULO DIDÁCTICO DE INSTRUMENTACIÓN CON CONEXIÓN AL SERVIDOR WEB NODE-RED PARA EL MONITOREO Y CONTROL DEL PROCESO CON MINI PLC LOGO V 8.3 Y RASPBERRY PI 4**. Con el fin de, complementar y dar un enfoque a los estudiantes para el desarrollo en el conocimiento de nuevas tecnologías.

### **PALABRAS CLAVES**

Node RED, LOGO V8.3, Comunicación, Servidor web, Raspberry Pi 4, Programación gráfica, Monitoreo, Dashboard.

## ABSTRACT

Year	Degree	Students	Supervisor	Thesis Topic
2023	Electronic Engineering	Alex Mauricio Quispillo Fares Luis Geovanny Quispillo Farez	Ph.D. Víctor Huilcapi	Design and development of a didactic module of instrumentation with connection to the web server node-red for process monitoring and control with mini plc logo v 8.3 and raspberry pi 4

The Salesian Polytechnic University, based in Guayaquil, is an academic institution that provides higher education services, which has adequate infrastructure for laboratory practices, appropriate for students to expand their knowledge, from the theoretical to the practical.

The objective of the technical project is aimed at students of the Degree in Electronics Engineering with a specialization in Industrial Systems, with the aim of knowing, learning and simulating the different practices projected to Industry 4.0 and the Internet of Things (IoT), on a process of filling and recirculation of liquid generally used to explain automated systems.

The didactic plant consists in the control of level and temperature, for which, an ultrasonic sensor and a level relay was implemented in the two stainless steel tanks, in addition to a PT-100 and the heating resistance for heating the liquid in this case water, the temperature control will be by PWM. Finally, the Mini PLC LOGO 8 and the Mini-computer Raspberry Pi 4 are essential parts of the module to show the control of the process and the drive of the same by means of the web server Node-RED. The correct functioning of the project was based on experiments and tests that were executed with the equipment presented above, which are in industrial applications, and their respective real-time communication of the Node-RED server which is a tool based on flow editor in which you can add or delete nodes and link them to each other.

The technical project briefly presented is called **DESIGN AND DEVELOPMENT OF A DIDACTIC INSTRUMENTATION MODULE CONNECTED TO THE WEB SERVER**

**NODE-RED FOR PROCESS MONITORING AND CONTROL WITH MINI PLC LOGO V 8.3 AND RASPBERRY PI 4.** In order to complement and give a focus to the students for the development in the knowledge of new technologies.

**KEYWORDS**

Node-RED, LOGO V8.3, Communication, Web Server, Raspberry Pi 4, Graphical Programming, Monitoring, Dashboard.

# ÍNDICE

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>1. ANÁLISIS DEL PROBLEMA .....</b>	<b>2</b>
1.1 PROBLEMA DE ESTUDIO .....	2
1.2 DELIMITACIÓN.....	2
1.2.1 Delimitación temporal.....	2
1.2.2 Delimitación espacial.....	2
1.2.3 Delimitación académica.....	2
1.3 JUSTIFICACIÓN .....	3
1.4 OBJETIVOS .....	3
1.4.1 Objetivo general.....	3
1.4.2 Objetivos específicos .....	3
1.5 Metodología.....	4
1.5.1 Metodología Investigativa.....	4
1.5.2 Metodología Experimental.....	4
<b>2. FUNDAMENTOS TEÓRICOS.....</b>	<b>5</b>
2.1 Procesos automatizados en industria 4.0 .....	5
2.2 Controles Automáticos.....	5
2.2.1 Introducción .....	5
2.2.2 Plantas .....	5
2.2.3 Procesos .....	6
2.2.4 Sistemas .....	6
2.3 Comunicación industrial.....	6
2.3.1 Ventajas de las comunicaciones en la industria .....	7
2.3.2 Capa Física.....	8
2.3.3 Topología de redes.....	8
2.4 PROTOCOLOS DE COMUNICACIÓN .....	9
2.4.1 Protocolo HTTP .....	9
2.4.2 Conexión S7.....	9
2.4.3 TSAP.....	9
2.5 Control Lógico Programable (PLC) .....	10
2.5.1 Estructura interna del PLC.....	10

2.6 Mini PLC LOGO .....	12
2.7 Sensores industriales.....	12
2.8 Bomba De Agua.....	13
2.8.1 Tipos y partes de una bomba.....	13
2.9 Raspberry Pi.....	14
2.9.1 Especificaciones técnicas de la Raspberry Pi 4 .....	15
2.10 Internet de las cosas .....	16
2.11 Motor de flujo IoT Node-RED .....	16
2.11.1 Programación visual y comunicaciones.....	16
2.11.2 Dashboard .....	17
<b>INTEGRACIÓN DE HERRAMIENTA INFORMÁTICA Y PROGRAMACIÓN.....</b>	<b>18</b>
2.12 Iniciando Raspberry Pi OS .....	18
2.12.1 Formateo de la Micro SD Card.....	18
2.12.2 Descarga de herramienta Raspberry Pi Imager.....	18
2.12.3 Inicio e instalación del Raspberry Pi OS en la microSD .....	19
2.12.4 Configuración de idioma en Raspberry Pi 4 .....	23
2.13 Node-RED .....	25
2.13.1 Ejecución del editor de flujos.....	25
2.13.2 Editor de flujos Node-RED.....	28
2.13.3 Instalación de complementos en el editor de flujo.....	30
2.13.4 Node (Nodos).....	32
2.13.5 Flow (Flujo) .....	32
2.13.6 Mensaje .....	33
2.13.7 Wire (Cable).....	33
2.13.8 Widgets .....	33
2.13.9 node-red-contrib-s7.....	34
<b>3. MARCO METODOLÓGICO .....</b>	<b>37</b>
3.1 Diseño del módulo didáctico .....	37
3.1.1 Desarrollo del Tablero de control .....	37
3.1.2 Desarrollo de la Planta industrial.....	37
3.2 Elementos que conforma el módulo didáctico.....	39
3.2.1 LOGO! 8 versión 8.3 .....	39
3.2.2 Módulo de expansión LOGO 8 AM2 RTD 2AI SIEMENS .....	39

3.2.3 Raspberry Pi 4.....	40
3.2.4 ARGON ONE PI 4.....	41
3.2.5 Interruptor Termomagnético tipo C .....	42
3.2.6 El Sensor Ultrasonido UK1A/G7-0ESY .....	42
3.2.7 Sensor PT-100 Ø 6 DIN / Cabezal – PIXSYS .....	43
3.2.8 Router inalámbrico TP-LINK TL-WR840N.....	44
3.2.9 Fuente de alimentación AUTONIC 50 W 100/240 VAC /24VDC .....	45
3.2.10 UPS Serie NT-500 - FORZA.....	46
3.2.11 Relé de estado sólido 480 VAC/ 32 VDC – XURUI.....	47
3.2.12 Resistencia calefactora 2000W-220V .....	49
3.2.13 Relé de bobina.....	49
3.2.14 Relé de nivel CAMSCO.....	50
3.2.15 Válvula Solenoide UNI-D UW-15.....	51
3.2.16 Pantalla Touch 7” LCD.....	51
<b>4. PLANTEAMIENTO Y DESARROLLO DE LAS PRÁCTICAS .....</b>	<b>52</b>
4.1 Práctica #1 .....	52
4.2 Práctica #2 .....	53
4.3 Práctica #3 .....	54
4.4 Práctica # 4 .....	55
4.5 Práctica #5 .....	62
4.6 Práctica #6 .....	70
4.7 Práctica #7 .....	75
4.8 Práctica #8 .....	84
4.9 Práctica #9 .....	92
4.10 Práctica #10 .....	100
<b>5. ANEXOS.....</b>	<b>115</b>
5.1 Resolución teórico y práctico del planteamiento #1:.....	115
5.2 Resolución teórico y práctico del planteamiento # 2:.....	124
5.3 Resolución teórica del planteamiento # 3:.....	127
5.4 Simulación y desarrollo del módulo didáctico .....	129
<b>CONCLUSIÓN .....</b>	<b>133</b>
<b>RECOMENDACIÓN .....</b>	<b>134</b>
<b>BIBLIOGRAFÍA .....</b>	<b>136</b>



## ÍNDICE DE FIGURAS

Figura 1. Pirámide de automatización .....	7
Figura 2. Bloque de comunicación S7 .....	9
Figura 3. Estructura interna de un PLC .....	11
Figura 4. Mini PLC LOGO! y LOGO! Soft Comfort.....	12
Figura 5. Modelos de bombas de agua en distintas aplicaciones .....	13
Figura 6. Parte de una Bomba de Agua .....	14
Figura 7. Editor de programación visual de Node Red .....	17
Figura 8. Dashboard de Node-red de control y lectura de proceso .....	17
Figura 9. Se elige la opción Download for Windows.....	18
Figura 10. Raspberry Pi Imager V1.7.1 .....	20
Figura 11. Operating System .....	20
Figura 12. Clic en Raspberry Pi OS full (32-bit).....	21
Figura 13. Selección de microSD para su instalación .....	22
Figura 14. Escritura en la Micro SD.....	22
Figura 15. Escritorio de Raspberry Pi OS .....	23
Figura 16. Configuración del idioma en Raspberry Pi OS .....	24
Figura 17. Ventana de configuración Raspberry Pi.....	24
Figura 18. Ventana de reinicio para establecer la configuración.....	25
Figura 19. Inicio de Node-RED.....	26
Figura 20. Terminal de Node-RED .....	26
Figura 21. Ventana LXTerminal.....	27
Figura 22. Ventana LXTerminal.....	27
Figura 23. Campo de Node-RED.....	29
Figura 24. Sidebar (barra lateral).....	30
Figura 25. Arranque para la visualización del Dashboard.....	30
Figura 26. Pasos para instalación de complementos 1/2 .....	31
Figura 27. Pasos para instalación de complementos 2/2 .....	31
Figura 28. Nodos .....	32
Figura 29. Diferencia de termino flujo .....	33
Figura 30. Complemento PLC S7 para node-RED.....	34
Figura 31. Configuración IP, puerto y TSAP dentro del nodo entrada S7 .....	35
Figura 32. Tabla para el uso de variables en Node-RED por S7.....	35



Figura 33. Lista de variables de entradas de LOGO!8 .....	36
Figura 34. Vista del módulo didáctico.....	37
Figura 35. Tanques del módulo didáctico.....	38
Figura 36. Diseño en 3D del módulo didáctico .....	38
Figura 37. Mini PLC LOGO 8 a la versión 8.3 .....	39
Figura 38. Módulo de expansión 6ED1055-1MA00-0BA2 .....	40
Figura 39. Placa interna Raspberry Pi 4 Modelo B .....	41
Figura 40. Argon one pi 4.....	42
Figura 41. Interruptor termomagnético.....	42
Figura 42. Sensor ultrasonido UK1A/G7-0ESY .....	43
Figura 43. Termorresistencia PT-100.....	44
Figura 44. Router TP-LINK TL-WR840N.....	45
Figura 45. Fuente de alimentación AUTONIC.....	46
Figura 46. Forza Serie NT-500.....	47
Figura 47. Relé XURUI.....	48
Figura 48. Resistencia Calefactora para inmersión .....	49
Figura 49. Relé de bobina.....	49
Figura 50. Relé de nivel y Sondas marca CAMSCO.....	50
Figura 51. Válvula solenoide UNI-D UW-15.....	51
Figura 52. Pantalla touch 7” .....	51
Figura 53. Comunicación entre LOGO 8 y Raspberry Pi 4.....	56
Figura 54. Práctica 4 LOGO! Soft Comfort .....	56
Figura 55. Creación de los grupos para el dashboard.....	58
Figura 56. Configuración en Layout práctica 4 .....	60
Figura 57. Programación visual de la práctica 4 completa.....	61
Figura 58. Dashboard de la práctica 4 .....	61
Figura 59. Comunicación entre LOGO! 8 y Raspberry Pi 4 .....	64
Figura 60. Programación en LOGO Soft Comfort práctica 5.....	65
Figura 61. grupo práctica 5.....	66
Figura 62. programación visual en Node-RED sobre la práctica 5 parte 1 .....	68
Figura 63. programación visual en Node-RED sobre la práctica 5 parte 2.....	69
Figura 64. Dashboard práctica 5.....	69
Figura 65. Ajuste de valor analógico.....	71
Figura 66. Texto de aviso en pantalla LOGO! 8 .....	72

Figura 67. Medición del tanque vacío .....	72
Figura 68. Ajuste de instrucción aritmética.....	73
Figura 69. Pantalla LOGO! 8.....	73
Figura 70. Programación en LOGO1 Soft Comfort práctica 6.....	74
Figura 71. Comunicación entre LOGO 8 y Raspberry Pi 4.....	77
Figura 72. Programación de la práctica 7 en LOGO Soft Comfort V8.3 .....	78
Figura 73. Grupo práctica 7 .....	79
Figura 74. Entorno de trabajo de Node-RED de la práctica 7 .....	81
Figura 75. Entorno de trabajo de Node-RED de la práctica 7 .....	82
Figura 76. Entorno de trabajo de Node-RED de la práctica 7 .....	82
Figura 77. Dashboard de la práctica 7 .....	83
Figura 78. Widgets personalizados práctica 7 .....	83
Figura 79. Comunicación entre LOGO! 8 y Raspberry Pi 4 práctica 8.....	86
Figura 80. Programación en FUP de la práctica 8 - 1/2.....	87
Figura 81. Programación en FUP práctica 8- 2/2 .....	88
Figura 82. Texto de aviso práctica 8.....	89
Figura 83. Grupo en dashboard práctica 8.....	90
Figura 84. Programación grafica de la práctica 8.....	91
Figura 85. Dashboard de la práctica 8 .....	91
Figura 86. Ventana para la comunicación entre LOGO 8 y Raspberry Pi 4 .....	93
Figura 87. Programación en LOGO! Soft Comfort de la práctica 9.....	94
Figura 88. Texto de aviso nivel bajo práctica 9.....	95
Figura 89. Marca de retroiluminación en color rojo.....	95
Figura 90. Creación de grupo práctica 9.....	96
Figura 91. Práctica 9 primera parte de la programación visual .....	97
Figura 92. Práctica 9 segunda parte de la programación visual .....	98
Figura 93. Dashboard de la práctica 9 .....	98
Figura 94. Dashboard de la práctica 9 .....	99
Figura 95. Entorno de trabajo de LOGO Soft Comfort .....	102
Figura 96. Programación de la práctica 10 .....	103
Figura 97. Creación de los grupos para el dashboard práctica 10 .....	104
Figura 98. Ajuste de cero y spam en Node-RED.....	107
Figura 99. Práctica 10 primera parte.....	107
Figura 100. Práctica 10 segunda parte.....	108

Figura 101. Práctica 10 tercera parte .....	109
Figura 102. menú en la ventana de dashboard práctica 10 .....	109
Figura 103. Tabla principal de la práctica 10 .....	110
Figura 104. Tabla 2 de nivel de la práctica 10.....	110
Figura 105. Tabla 3 de nivel de la práctica 10.....	111
Figura 106. Descarga de Remote-RED desde la Play Store.....	111
Figura 107. Dentro del nodo remote-access .....	112
Figura 108. Crear nueva configuración .....	112
Figura 109. QR de remote-access para visualización del dashboard de la práctica 10 .....	113
Figura 110. Apikey del Bot de nodo WhatsApp.....	114
Figura 111. Configuración API-KEY de WhatsApp.....	114
Figura 112. Estándar T568B.....	117
Figura 113. Diagrama de control .....	119
Figura 114. Diagrama de potencia.....	120
Figura 115. Disipador y Relé de estado solido .....	121
Figura 116. Distribución en el gabinete del proyecto.....	122
Figura 117. Tipos de terminales .....	123
Figura 118. Diagrama P&ID del módulo didáctico.....	126
Figura 119. Control de lazo abierto .....	127
Figura 120. Control de lazo cerrado .....	127
Figura 121. Montaje del tablero de control para desarrollo de las practicas .....	129
Figura 122. Tablero eléctrico 3D.....	129
Figura 123. Diseño demodulo 3D.....	130
Figura 124. Modulo didáctico completo.....	132

## ÍNDICE DE TABLAS

Tabla 1. Descripción del Raspberry Pi 4 .....	15
Tabla 2. Complementos de práctica 4.....	57
Tabla 3. Configuración de los nodos utilizados en la práctica 4 .....	59
Tabla 4. Complementos de práctica 5.....	65
Tabla 5. Configuración de los nodos utilizados en la práctica 5 .....	67
Tabla 6. Complementos de la práctica 7.....	78
Tabla 7. Configuración de los nodos utilizados en la práctica 7 .....	80
Tabla 8. Complementos de práctica 8.....	89
Tabla 9. Complementos de práctica 9.....	96
Tabla 10. Complementos de práctica 10.....	103
Tabla 11. Configuración nodos de práctica 10 .....	105
Tabla 12. Tabla de medidas de cables AWG.....	116
Tabla 13. Cableado estándar T568B.....	117
Tabla 14. identificación de letras y números .....	123
Tabla 15. Tabla ISA-S5.1 .....	124
Tabla 16. Nomenclatura del módulo didáctico.....	125
Tabla 17. análisis de lazo de control del módulo didáctico.....	128

## INTRODUCCIÓN

En primera instancia la Universidad Politécnica Salesiana con sede en Guayaquil tienen a expandir sus módulos didácticos a facilidad de enseñanza teóricas a prácticas de diferentes materias que se realizan en la carrera de ingeniería en electrónica y así mismo complementar con tecnologías que día a día se observa sus cambios.

Por esta razón principal surgió la idea de proponer un módulo didáctico basado la aplicación del internet de las cosas conocido como IoT y enfocado a lo que es Industria 4.0 en la cual, los estudiantes de la carrera de ingeniería electrónica puedan desarrollar prácticas descritas en este libro y además puedan surgir nuevas prácticas ya que el módulo tiende a expandir sus planteamientos.

Este proyecto de titulación se basó de un proceso común conocido en las materias de instrumentación, sensores, automatización, para mayor facilidad de comprensión del proceso y programación basado en lógica y programación visual. Todos los problemas planteados y resueltos están basados en teórico para sus respectivos análisis y lo práctico para su desenvolvimiento, descritas paso a paso en imágenes y configuraciones detallados en tablas, tomando en cuentas las recomendaciones antes de proceder el uso del módulo didáctico para el desarrollo de las prácticas.

# **1. ANÁLISIS DEL PROBLEMA**

## **1.1 PROBLEMA DE ESTUDIO**

La Universidad Politécnica Salesiana con sede en Guayaquil que expanden sus módulos didácticos en la carrera de ingeniería en electrónica, se ha observado que hace falta cubrir la necesidad para estudiantes y docente de un módulo para demostrar el monitoreo y control por medio de un servidor en la nube llamado Node-RED desde una Raspberry Pi y como elemento base el autómatas programable LOGO! 8 para un proceso comúnmente conocido en la materia de comunicaciones (SI), instrumentación, sensores y transductores, redes y automatización.

La aplicación del internet de las cosas (IoT) en empresas y centro comerciales es poco, pero su importancia ira creciendo a medida que, usuarios conozcan y se instruyan en modelos de control, y realicen la operación desde cualquier parte que estén, así mismo, los desarrolladores implementen nuevos modelos para la aplicación de procesos. Sobre todo, en el entorno industrial y comercial la aplicación IoT es muy útil en el ahorro de recursos, tiempo de espera, recopilación de datos por ejemplo para evitar o un evento en cuestión por una falla en la máquina dentro de una producción.

## **1.2 DELIMITACIÓN**

### **1.2.1 Delimitación temporal**

La implementación de este proyecto se realizará en el año 2022-2023

### **1.2.2 Delimitación espacial**

El proyecto será dirigido para el laboratorio de la universidad politécnica salesiana sede centenario.

### **1.2.3 Delimitación académica**

El proyecto consiste en la implementación de un módulo de prácticas, simulando el proceso de llenado y vaciado de tanque que recircula el proceso, por medio de un PLC LOGO V8.3 que será

el instrumento de programación para luego enlazare con la nube NODE RED por medio de Raspberry, que hoy en día se a lo que se llama industria 4.0

### **1.3 JUSTIFICACIÓN**

Si bien es cierto, es importante que el estudiante conozca de esta herramienta didáctica con esta versión actualizada para el estudio de la instrumentación industrial más la conexión con la nube, lugar en el que se hará el monitoreo y control del proceso, posteriormente, lo innovador, es el estudio del internet de las cosas, que desde antes se dio a conocer, pero, ahora con mayor fuerza su aplicación se está dando en el área de negocios locales, centros comerciales e industrias. La finalidad del presente tiene, por objeto ayudar en el desarrollo de conocimientos, habilidades y destrezas profesional. Este diseño amigable que muestra la simulación de un proceso industrial que se estudia en las materias de instrumentación, comunicaciones (SI), sensores y transductores, redes y automatización.

### **1.4 OBJETIVOS**

#### **1.4.1 Objetivo general**

Diseñar y equipar un módulo didáctico con tecnología IoT basado en LOGO! V 8.3 y editor de flujo Node-Red desde una Raspberry Pi 4 para monitoreo y control de nivel y temperatura de un proceso, a la Universidad Politécnica Salesiana sede Guayaquil.

#### **1.4.2 Objetivos específicos**

- Desarrollar el módulo de entrenamiento didáctico donde se encuentren los elementos necesarios para la comunicación y visualización del motor de flujo Node-Red.
- Detallar los protocolos necesarios para la comunicación y la programación, entre el servidor web Node-Red previamente instalado en Raspberry Pi 4 y autómeta programable LOGO! V8.3.

- Elaborar un manual de 10 prácticas para la utilización del módulo.

## **1.5 Metodología**

### **1.5.1 Metodología Investigativa**

Para el desarrollo del proyecto se realizará una recopilación de datos de libros, artículos, sitios web como referencias para el estudio sobre la programación visual mediante nodos en el uso del servidor web Node-red, como cloud computing (computadora de nube), parte importante del desarrollo en la transferencia y direccionamiento de datos en tiempo real con la comunicación con el PLC LOGO 8 y así mismo el comportamiento de los elementos que integran en la planta de nivel y temperatura en el control y monitoreo, para disponer los recursos necesarios para la implementación.

### **1.5.2 Metodología Experimental**

Con esta metodología se realizará las diferentes prácticas que van relacionada con el módulo didáctico que fueron expuestas en el documento.



## **2. FUNDAMENTOS TEÓRICOS**

### **2.1 Procesos automatizados en industria 4.0**

En base a las investigaciones en estos últimos años, se han desarrollado varios nuevos protocolos de comunicación, es por ello que nacieron modelo de conectividad de cuales se dio la proyección a industria 4.0 que, a finales de 2011 se ha producido una fascinación por el estudio que atrae a empresas y los institutos de investigación para trabajar en conjunto por el amplio impacto positivo por el recurso en la eficiencia operativa, así como el desarrollo de tipo comercial, producto y servicios totalmente inéditos (Hermann, Pentek, & Otto, 2015).

Se presenta una planta industrial dará la facilidad a los receptores de poder presenciar el empleo y su vez el control de los instrumentos y parte del proceso que están presente en diferentes industrias, por lo tanto, fortalecería la parte teórica por el resultado práctico. (Tumbaco & Villa, 2018)

### **2.2 Controles Automáticos**

#### **2.2.1 Introducción**

El control automático ha cumplido un rol importante en el avance de la ingeniería y ciencia. Hoy en día hay el privilegio de poder observar alrededor todo lo automático, sea en la parte domótica, sistema de alarma y vigilancia automáticas, en lo industrial se observa la importancia de un control de temperatura, presión, nivel, etc. Por lo tanto, esto a medida que se aplique al sistema un mejorado control automático, sus procesos van a ser mucho más eficientes y su rendimiento en optima condiciones (Ogata, 2010).

#### **2.2.2 Plantas**

La planta es una forma de describir a un sistema físico o la división de este sistema físico en partes que tiene como finalidad realizar tarea específica o una secuencia de tareas. Que puede ser una línea de producción hasta un sistema de refrigeración. (Ogata, 2010).

### **2.2.3 Procesos**

Se denomina a la operación o desarrollo natural que va de forma progresiva ininterrumpida, en el desarrollo puede haber una serie de cambio graduales de uno o más equipos que están enlazados para un fin o propósito determinado de manera controlada. (Ogata, 2010) .

### **2.2.4 Sistemas**

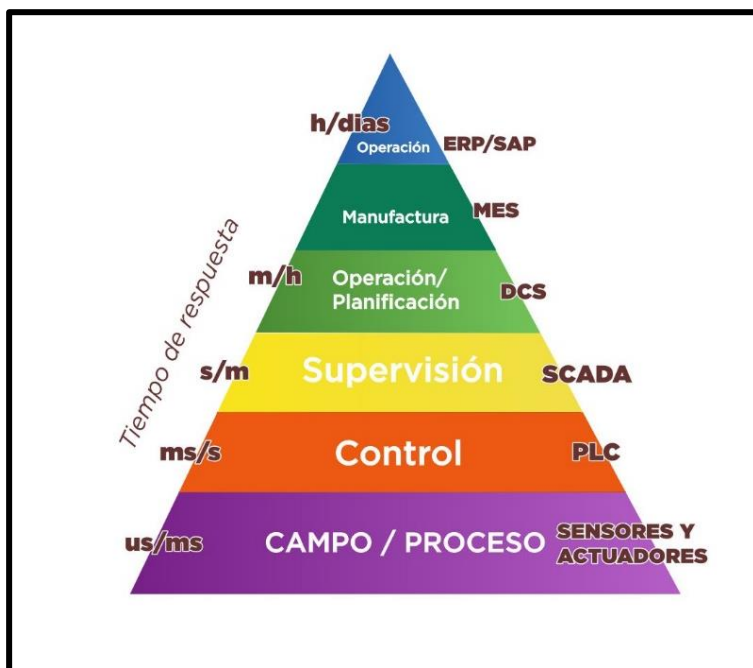
El concepto de sistema es una combinación de elementos o componentes que actúan unido y realizan un objetivo establecido. no está limitado sistemas físicos, por lo tanto, se aplica su concepto a sistemas abstracto o dinámicos (Ogata, 2010).

## **2.3 Comunicación industrial**

A principios, los sistemas industriales a medidas que sus procesos iban creciendo en complejidad, su cableado aumentaba en proporción o mucho más, por lo tanto, el objetivo de las industrias es reducir el cableado, de modo que se logró la solución, fue por ello por lo que se implementaron las comunicaciones en la industria abriendo una brecha a nuevas tecnologías que hoy en día se utilizan para sus procesos sean más efectivos y mejores resultados económicos (Belmonte, 2018).

La comunicación en la industria ha sido de muchos beneficios; simplifica el cableado de larga distancia, optimización de procesos, modelos de operación en tiempo real, todo dependiendo de las necesidades por la que se requiera implementar en algún nivel de automatización, como se muestra en la figura 1.

Figura 1. Pirámide de automatización



Fuente: (Sole, 2010)

### 2.3.1 Ventajas de las comunicaciones en la industria

El objetivo de cada industria es la búsqueda de mejorar su producción en calidad, en lo económico, y que sea competitivo, por lo tanto, la comunicación en la industria aplicándolo de acuerdo con las necesidades, se obtendría:

- Aumento de la eficacia de los diferentes procesos que interviene en la producción (Belmonte, 2018).
- Reducción de los stocks (Belmonte, 2018).
- Comunicación automática con los pedidos de los clientes (Belmonte, 2018) .
- Reducción del coste de mantenimiento, instalación y puesta a punto (Belmonte, 2018).
- Mantenimientos preventivos basados en hechos históricos (Belmonte, 2018).

Cabe enfatizar que se encuentran varias normas y estándares por parte de grupos de fabricantes. En la actualidad, estos sistemas son abiertos (no pertenecen a algún fabricante) por ende, el modelo OSI (Open System Interconnection) se define como el estándar de la comunicación entre dos interlocutores, este cuenta con siete niveles; físico, enlace, red, transporte, sesión, presentación (Belmonte, 2018).

### **2.3.2 Capa Física**

Cabe señalar que esta capa es la más sencilla de entender, está relacionado con todos los componentes que está unido en la conexión y característica con la que fueron aplicados: el cableado, los dispositivos transmisión, los niveles eléctricos de transmisión. Además, determina la forma que en la que se trasmite la información a través del medio (Belmonte, 2018).

### **2.3.3 Topología de redes**

Al referirse a topología, se define a la forma en la que está diseñada la red, relacionada tanto en parte física (el uso de componentes que están conectados los dispositivos) y en la parte lógica (medio de transmisión de datos en la red y subred). Cabe señalar que, los dispositivos que están conectados y enlazados a una red se los conoce como *nodos*.

Por lo tanto, se define la topología de red a la forma geométrica en conjunto con nodos y enlaces o conexión de varios elementos. Los tipos de topología existentes son: malla, estrella, árbol, bus, anillo. La más usada en sistemas industriales son la de bus y anillo. (Belmonte, 2018)

## 2.4 PROTOCOLOS DE COMUNICACIÓN

### 2.4.1 Protocolo HTTP

Su significado en español es protocolo de transferencia de hipertexto de comunicación, es un protocolo de comunicación utilizado en la Word Wide Web (WWW). Este permite el enlace y transferencia de datos como imágenes, videos, textos y demás, entre un cliente (navegador web) y un servidor.

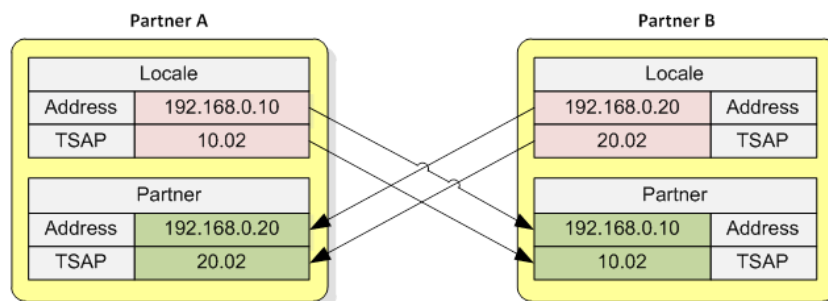
### 2.4.2 Conexión S7

Una conexión S7 es una conexión cliente-servidor, donde el cliente puede acceder directamente al servidor, escribiendo y leyendo datos, sin necesidad de ninguna función de lectura y escritura en el servidor, ciertamente esta conexión es propiamente de SIEMENS para la comunicación entre sus controladores (SIEMENS, 2020).

### 2.4.3 TSAP

Transport Service Access Point (significado en inglés), como se ilustra en la figura 2 es un medio que permite el intercambio de datos en una conexión S7. Por lo general utilizado en comunicación entre controladores y también con interfaces graficas que soporten esta conexión S7 de tipo cliente – servidor.

Figura 2. Bloque de comunicación S7



Fuente: (Snap7 , s.f.)

## 2.5 Control Lógico Programable (PLC)

El trabajo del PLC es la de ejecutar, controlar y monitorear un proceso, a partir de, secuencia de acciones repetitivas y con precisión de forma automática previamente definido desde un software especializado de programación lógica. Pueden realizar tarea de control de motores, monitorear sensores, ajustar la temperatura y además realizar calculo matemáticos.

### 2.5.1 Estructura interna del PLC

Internamente, un PLC componen de elementos claves:

- **Unidad central de proceso (CPU)**

La CPU es el cerebro del PLC constituida por *el procesador* que se encarga de ejecutar el proceso en el que fue programado por el usuario, además, se comunica con el exterior mediante sus puertos de comunicación y realizar funciones de revisión, análisis y ejecución del proceso. *La memoria* es la que almacena la información del PLC y puede estar compuesta de diferentes tipos de memoria física de diversas tecnologías, según la tipo y capacidad del controlador programable, ciertos modelos de PLC utilizan memorias tipo EPROM, EEPROM o Flash (memoria no volátil), RAM, esta última tiene la capacidad de escritura y lectura mucho mayor que las anteriores descritas, con la desventaja de pérdida de información al quitar su alimentación o conocida memoria volátil.

- **Interfaz de entradas**

Toman las señales derivado de elementos captadores tales como límites de carrera, botoneras, sensores, presostatos, trasmisores de nivel, presión, temperatura en el caso de entradas analógica o digital.

- **Interfaz de salida**

Mientras las salidas, estas son las que comandan las señales para su respectiva conmutación de bombas, motores, válvulas, control de variador de velocidad, dependiendo de la salida que se esté utilizando, tanto analógica o digital.

- **Buses de comunicación**

Conexiones que permite la comunicación entre las unidades de memoria, CPU, las interfaces de entradas y salidas, por lo tanto, se constituye por 3 buses:

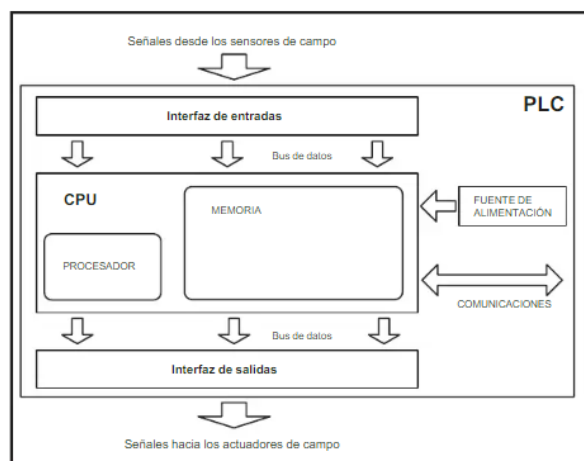
- **Bus de control:** evita el agolpamiento de datos en el intercambio de información.
- **Bus de Datos:** Transfiere datos del sistema.
- **Bus de Dirección:** es la parte que direcciona los datos tanto de la memoria como los demás periféricos.

- **Fuente de Alimentación**

Es la que proporciona la energía eléctrica necesaria para la alimentación de uno o varios dispositivos, en este caso el PLC.

La estructura interna de un PLC se describe por medio de un diagrama de bloque, como se ilustra en la figura 3.

Figura 3. Estructura interna de un PLC



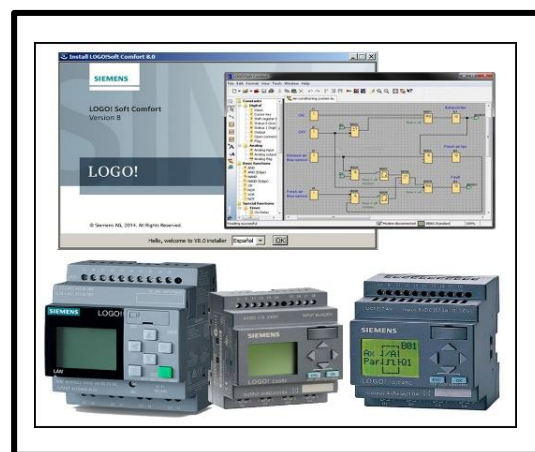
Fuente: (Daneri, 2008)

## 2.6 Mini PLC LOGO

Conocidos como autómatas programables o controlador lógico programable (PLC) su finalidad es la de ejecutar y controlar procesos definidos dentro la automatización industrial, realiza tareas lógicas y secuenciales, se tienen varios modelos como se ilustra en la figura 4.

La programación se desarrolla mediante funciones lógicas, secuenciales y combinatorias. El software que utiliza es propiamente de la marca SIEMENS conocida como LOGO! Soft Comfort, lo más práctico es programar dentro del software que realizar desde la pantalla del autómata programable. (Vera & Humberto, 2015)

Figura 4. Mini PLC LOGO! y LOGO! Soft Comfort



## 2.7 Sensores industriales

Ante la necesidad de llevar a cabo la automatización de un proceso es aquí donde intervienen en un papel importante dentro de la automatización llamados sensores industriales. Cabe resaltar la importancia de estos dispositivos ideales para la medición y detección de variables físicas o químicas de estos sirven para el control de procesos y monitorización de equipos o plantas.



## 2.8 Bomba De Agua

La función de la bomba de agua es transformar la energía mecánica en fuerza de impulso para el fluido o liquido en una dirección determinada.

El trabajo que realiza puede variar la velocidad, presión y altura, dependiendo del uso del que se quiera dar, respetando el correcto funcionamiento; fuente de alimentación, resistencia al ambiente, y además respetar su norma en el área de aplicación. Como se muestra en la figura 5 (Baéz & Hernandez, 2020).

Figura 5. Modelos de bombas de agua en distintas aplicaciones



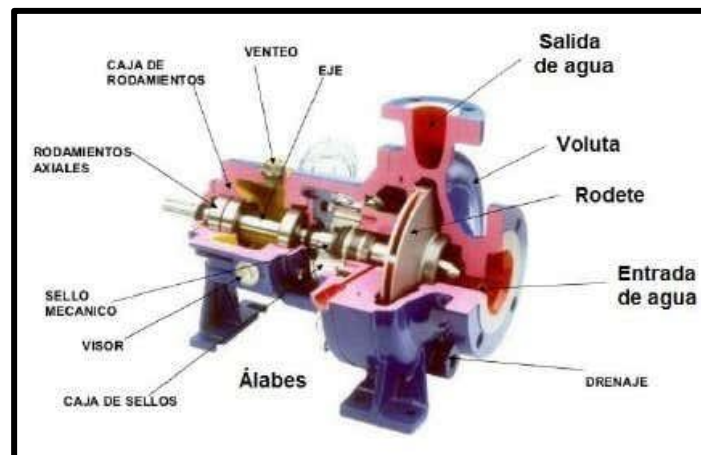
### 2.8.1 Tipos y partes de una bomba

Las principales que se nombran son: Electrobombas, motobombas, neumáticas, hidráulicas.

- **Carcaza o armazón:** su función es recubrir a la bomba, es decir, es el cuerpo que protege y resiste de daños exteriores.
- **Entrada y salida:** son los orificios de la bomba que entran y sale los líquidos o fluidos.
- **Impulsor o rotor:** Es una de la parte fundamental para el impulso del líquido que está contenido en el cuerpo de la bomba, esta puede estar constituida por alabes, aspas, engranajes, entre otras.

- **Sellos retenedores y anillos:** Da estabilidad, compactación y seguridad a las partes internas de la bomba para el funcionamiento correcto de la misma.
- **Eje impulsor o balero:** Es el eje que mantiene estable el elemento motor con el impulsor.
- **Cojinetes:** Es un rodamiento que reduce la fricción entre el eje de la bomba y los rodetes.
- **Panel de control:** función es el arranque de la bomba, internamente tiene botones o switches.
- **Motor:** Es la parte importante de la bomba que realiza el movimiento del eje y el impulsor para el paso del líquido de un lado a otro (Baéz & Hernandez, 2020).

Figura 6. Parte de una Bomba de Agua



Fuente: (Baéz & Hernandez, 2020)

## 2.9 Raspberry Pi

Es una placa electrónica inteligente, a diferencia de otras, este dispositivo es conocido como una motherboard de pequeño tamaño, además, se asocia al concepto de open source (librería abierta) lo que nos asegura la opción de trabajar con software legal y contar con información útil gratuita del internet. Hoy en día se puede encontrar series de Raspberry que han ido mejorando con el pasar de los años en cuanto a su procesamiento de datos de lectura y escritura.

### 2.9.1 Especificaciones técnicas de la Raspberry Pi 4

En el mercado se comercializa dos modelos que son; A y B. siendo la B la recomendable, por lo que integra un SoC (system on chip) que contiene varios módulos: Broadcom BCM2711, con 4 núcleos y de tipo ARM Cortex A72, que opera hasta una velocidad de 1,5GHz, además, contiene un chip grafico que proporciona video en 4K. Ciertamente, la SoC es un modelo de diseño de la que optimiza recursos, puesto que el rendimiento es mucho mejor en espacios reducido de las versiones anteriores. Como se muestra en la tabla 1 (Gombau, 2020).

Tabla 1. Descripción del Raspberry Pi 4

Características	Especificaciones
CPU	Quad core Cortex-A72 (64-bit) @ 1.5GHz
Gpu	H264 (1080p60 decode, 1080p30 encode) OpenGL ES 3.0 graphics, H.265 (4kp60 decode)
RAM	1GB, 2GB, 4GB.
Operación Voltaje Rango	5V - 3A mínima
Gpio Ports	28 I/O Pins
HDMI	2- Port with 4k Display (Mini-HDMI)
Bluetooth	5.0
WIFI	Disponibile
SD-Card	Disponibile
Temperatura de operación	0-50 grados
Display	DSI
USB	2×2.0, 2×3.0
LAN	Disponibile

Fuente: (Richard, 2022)

## **2.10 Internet de las cosas**

El IoT o “internet de las cosas” por sus siglas en inglés, es uno de los pilares de la transformación digital hoy en día. Al actuar en conjunto con otras tecnologías como la automatización, el cloud computing, movilidad, Big Data, analítica, ciberseguridad y entre otras.

Su aplicación reduce costo y genera nuevos ingresos al ser aplicado de manera eficiente en sus procesos como en la cadena de suministros y logística. El mayor logro es automatizar tareas que requiere de mucha intervención en mano de obra. (Kranz, 2017).

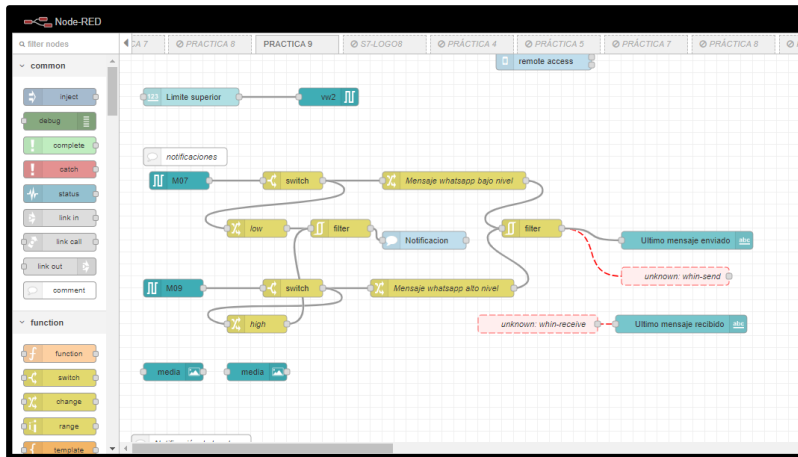
## **2.11 Motor de flujo IoT Node-RED**

Node-RED se ha establecido como cloud computing (computadora de nube), que se basa en una herramienta de programación visual muy ligera, sencilla y potente que permite tener conexiones con dispositivos de hardware, API, y servicios en línea (Ortiz, 2019).

### **2.11.1 Programación visual y comunicaciones**

Node-RED es un editor de flujo como se ilustra en la figura 7, basado en un navegador donde se puede agregar o suprimir nodos y allí vinculación uno a otro, con la finalidad de, hacer que se comuniquen entre ellos. Si bien es cierto, Node-red esta creado a partir de NodeJS que es un framework para implementar operaciones I/O, escrito con lenguaje JavaScript V8, es por esto por lo que NodeJS tienes procesos que ocupan niveles de memoria bastante menor a diferencia de otros lenguajes, además, soporta protocolos TCP, DNS, HTTP, MQTT, entre otros (Morales, Alvarez, & Mircha, 2016).

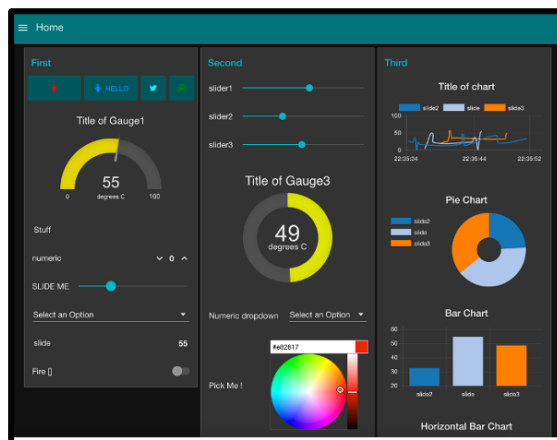
Figura 7. Editor de programación visual de Node Red



### 2.11.2 Dashboard

Mas allá de una programación visual, es posible basarse una gestión de datos en tiempo real en la que pueda ser visualizados por medio del dashboard que es creado dentro del editor de flujo como se ilustra en la figura 8, de la misma manera puede estar sujeta en la maniobra del control del proceso o ajustes limitados por el usuario, de manera manual por medio de pantalla touch. Sin duda, su enfoque está relacionado con IoT, industria 4.0, sistemas de inteligencia artificial (Ortiz, 2019).

Figura 8. Dashboard de Node-red de control y lectura de proceso



Fuente: (Node-RED, 2022)

## INTEGRACIÓN DE HERRAMIENTA INFORMÁTICA Y PROGRAMACIÓN

Para realizar las prácticas de este proyecto se utilizó el software de programación LOGO! Soft Confort V8.3 para lo que es el Mini PLC LOGO! 8 y para el uso de Node-RED en Raspberry Pi 4 se instaló el sistema operativo propio de la Raspberry basado en Debian (es un sistema operativo de código abierto y gratuito basado en LINUX, conocido por su estabilidad, seguridad y amplia comunidad de desarrollo) conocido como Raspberry Pi OS anteriormente conocido como Raspbian.

A continuación, se detalla cada uno de estos.

### 2.12 Iniciando Raspberry Pi OS

Primero, tener listo la microSD Card de clase 10 ideales para dispositivos que requieren una alta velocidad de lectura y escritura, con una capacidad mínima de 32 GB (recomendable) para la instalación del sistema operativo Raspberry Pi OS.

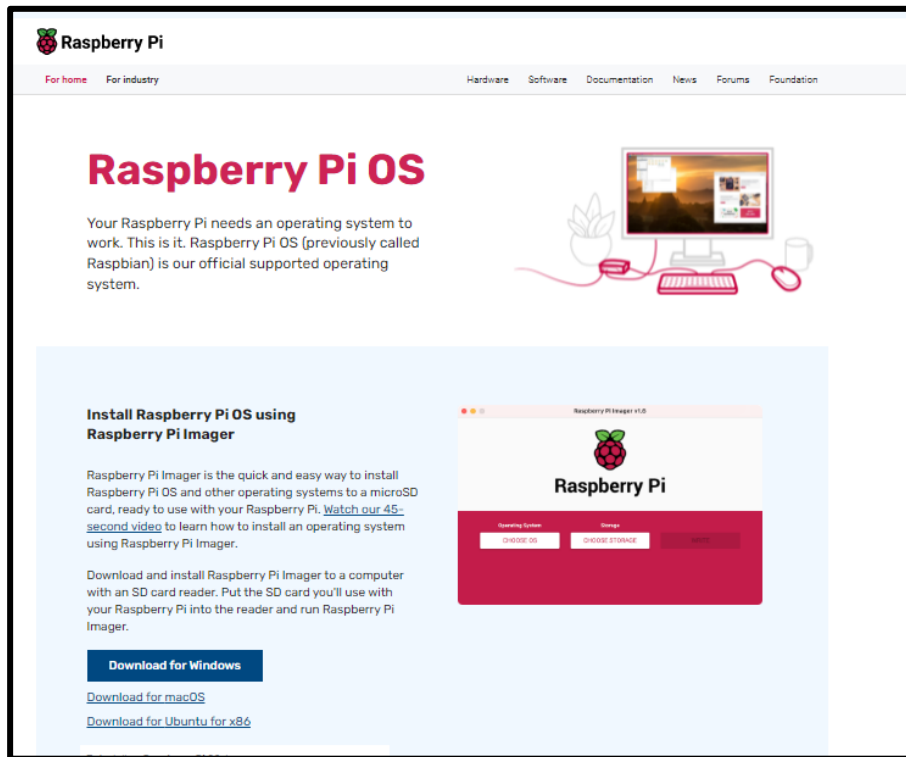
#### 2.12.1 Formateo de la Micro SD Card

Se descarga el software la cual se recomienda *SD Formatter 4.0* para Windows o Mac, luego formatea el microSD Card si es que es nueva o usada para evitar inconvenientes al instalar el Raspberry Pi OS.

#### 2.12.2 Descarga de herramienta Raspberry Pi Imager

Comienza descargando desde la página oficial [raspberrypi.com/software/](https://raspberrypi.com/software/), al usar *Raspberry Pi Imager* es una forma más rápida y simple para instalar el sistema operativo Raspberry Pi en la microSD Card, descarga para el sistema operativo de la computadora que usaras para la instalación en MicroSD, como se muestra en la figura 9.

Figura 9. Se elige la opción Download for Windows

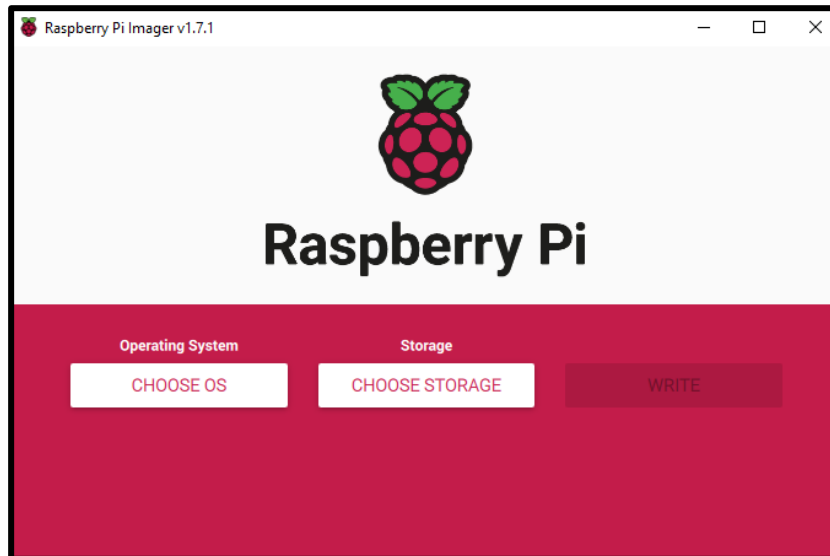


Fuente: (raspberrypi.com/software, 2022)

### 2.12.3 Inicio e instalación del Raspberry Pi OS en la microSD

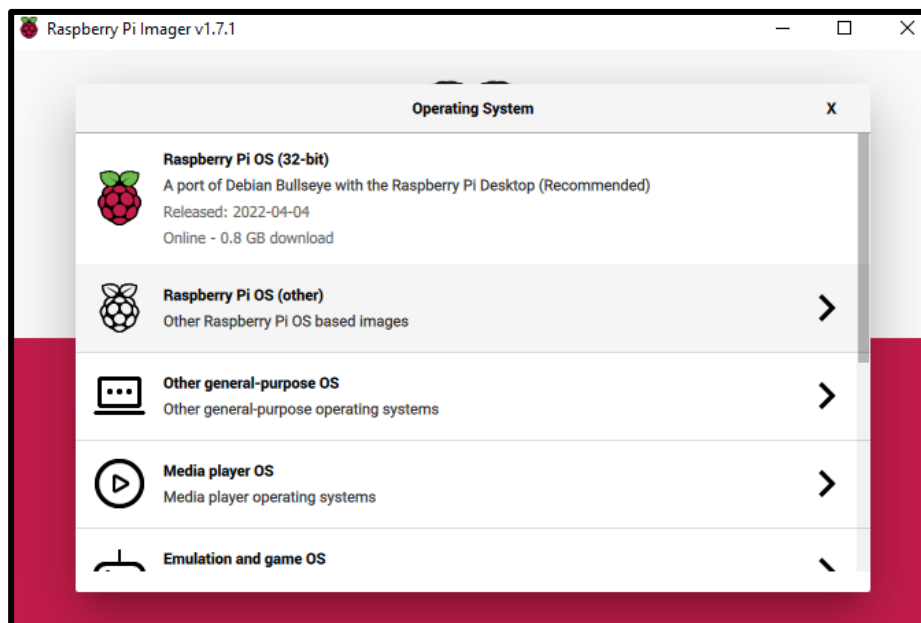
Una vez descargado e instalado, inicias el programa Raspberry Pi Imager para luego dirigirte al cuadro de opciones y das clic en **CHOOSE OS**, como se muestra en la figura 10.

Figura 10. Raspberry Pi Imager V1.7.1



Se da clic a la opción **Raspberry Pi OS (other)**, como se ilustra en la figura 11.

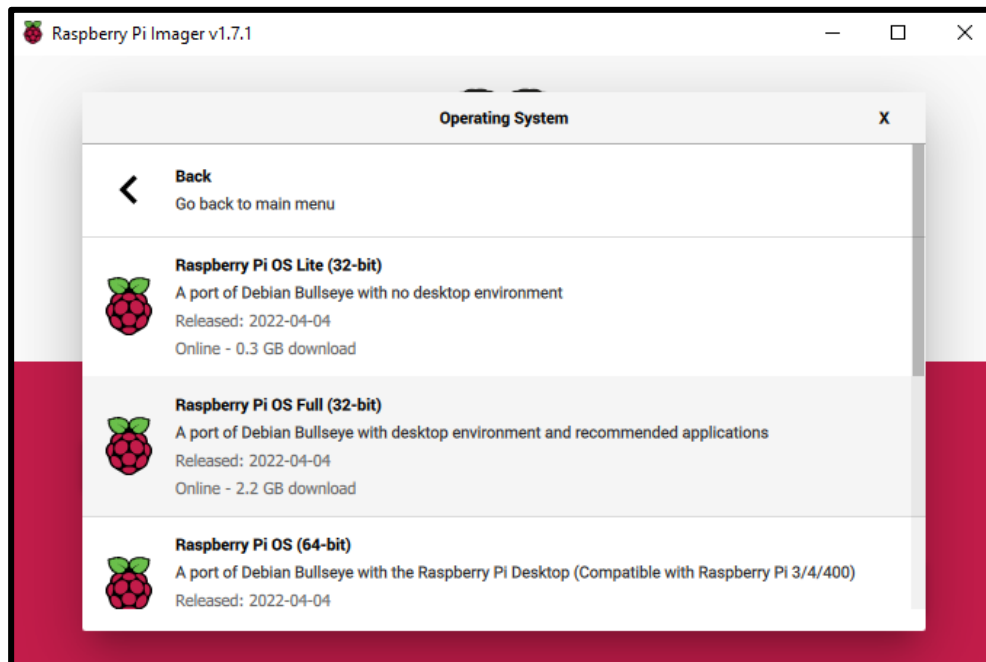
Figura 11. Operating System





Luego, dentro de la opción nos aparecerá otras varias opciones, como se muestra en la figura 12 de las cuales selecciona **Raspberry Pi OS full (32-bit)**, ya que en la lista nos proporciona la instalación de Node-RED.

Figura 12. Clic en Raspberry Pi OS full (32-bit)



Como últimos pasos, clic en la opción **STORAGE (almacenamiento)**, como se ilustra en la figura 13, selecciona la microSD Card que va hacer su función como “disco duro” para el arranque del Raspberry Pi OS.

Luego, selecciona **WRITE** para la instalación en la microSD, como se muestra en la figura 14. Finalizado la instalación se procede a retirar la memoria de la computadora y colocar en la ranura microSD del Raspberry Pi 4 para su arranque.

Figura 13. Selección de microSD para su instalación

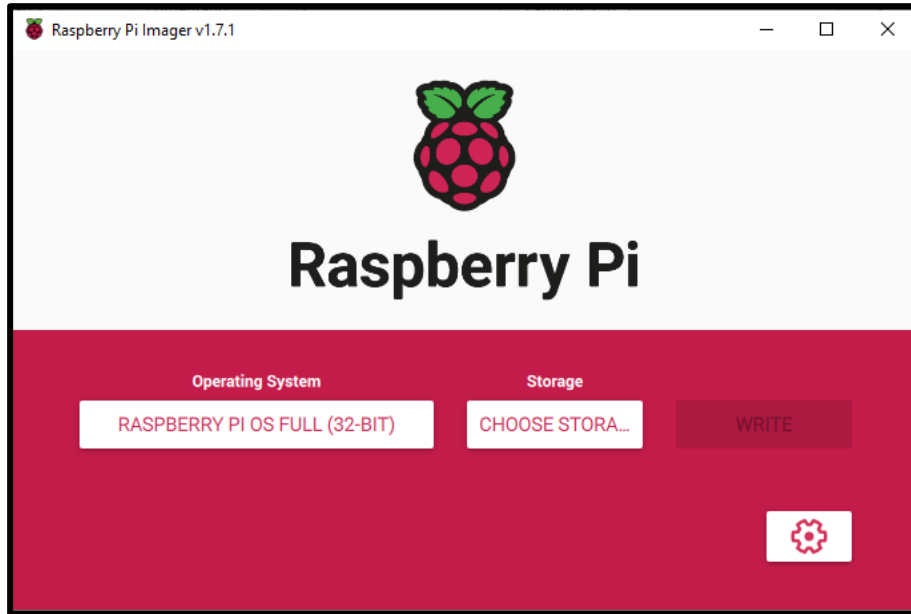
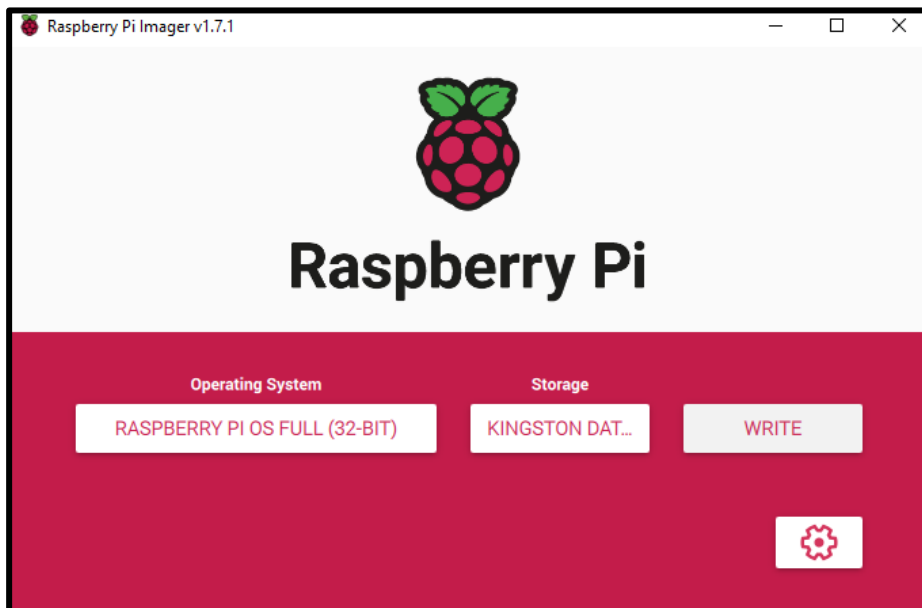


Figura 14. Escritura en la Micro SD



Desde que se detalló en este libro los pasos para su instalación, sin embargo, puede haber actualizaciones con el paso del tiempo por lo que ciertos pasos se tenga que omitir o agregar para mayor confianza visite la página Node-RED <https://nodered.org/> allí podrá consultar y buscar más información referente a su instalación y programación visual.

#### 2.12.4 Configuración de idioma en Raspberry Pi 4

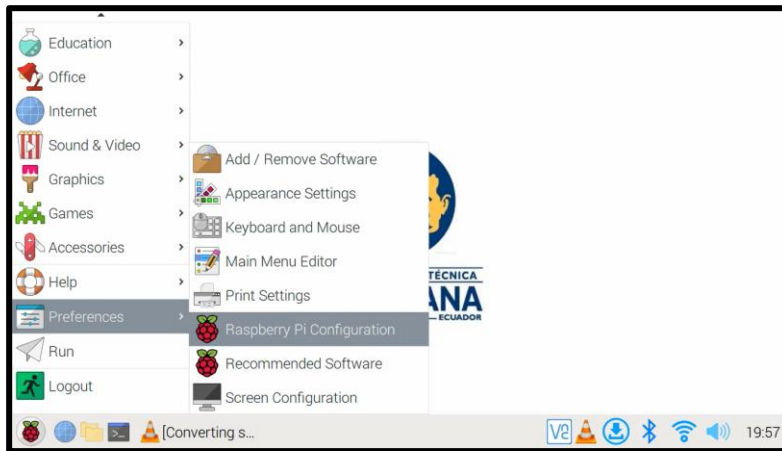
Una vez haya instalado el sistema operativo Raspberry Pi, inserte la Micro-SD a la Raspberry Pi 4, alimente y encienda el dispositivo, podrá visualizar mediante la pantalla LCD conectada al puerto USB de la Raspberry Pi 4, como se muestra en la figura 15.

Figura 15. Escritorio de Raspberry Pi OS



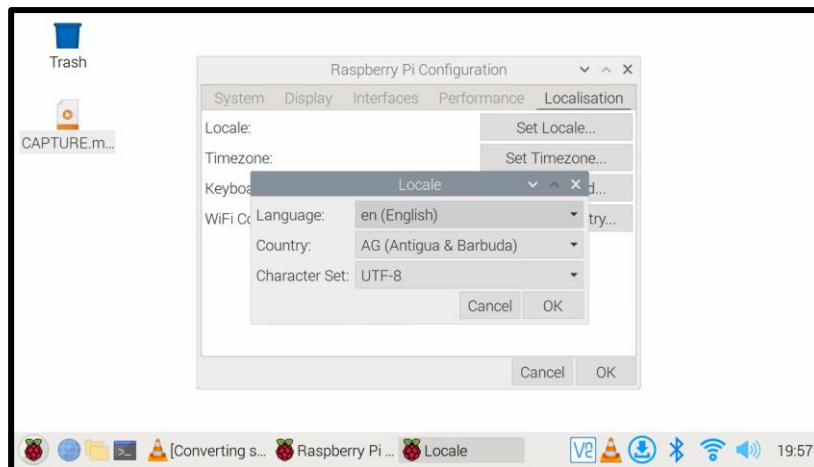
Haga clic en el cuadro de **Inicio**, se dirige **Preference** y selecciona **Raspberry Pi Configuration**, como se muestra en la figura 16.

Figura 16. Configuración del idioma en Raspberry Pi OS



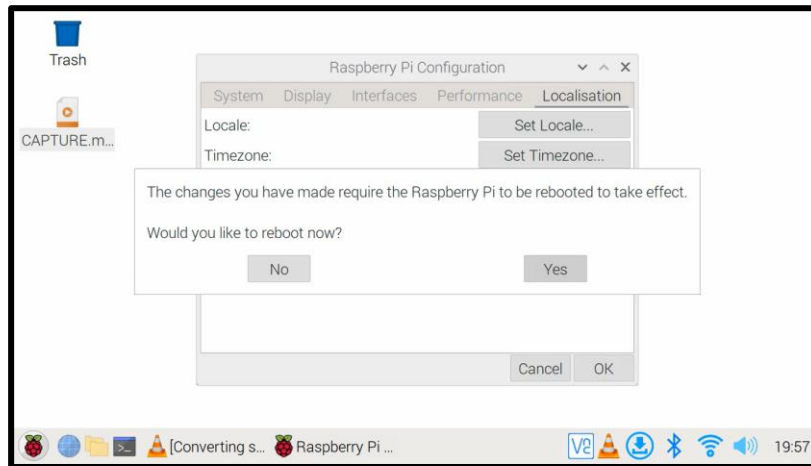
Se abrirá una ventana de opciones en la que se dirigirá a **Localisation**, allí selecciona **Locale**, para luego, buscar lo que es el idioma en spanish (español) en **Language**, de manera que se selecciona Ecuador en **Country**, como se ilustra en la figura 17.

Figura 17. Ventana de configuración Raspberry Pi



Una vez terminado la configuración, el propio sistema le envía un mensaje para el reinicio de esta para establecer la configuración, como se muestra en la figura 18, sino desea lo podría hacer al final del cierre si por algún motivo no requiera reiniciar.

Figura 18. Ventana de reinicio para establecer la configuración



Una vez reiniciado, el sistema operativo de Raspberry Pi 4 queda con la configuración establecida, por lo que, si se desea volver a configurar, tomar los pasos posteriores.

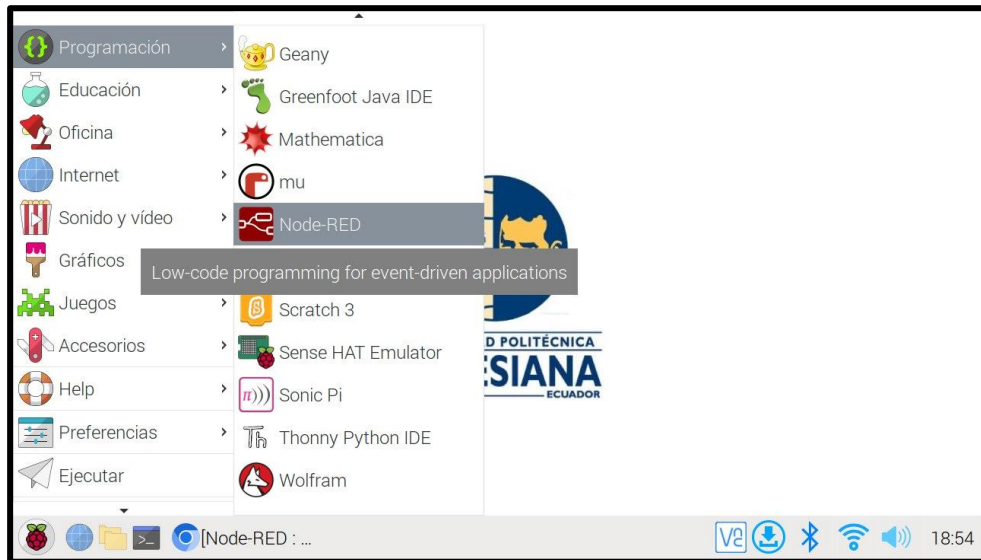
## 2.13 Node-RED

### 2.13.1 Ejecución del editor de flujos

Al momento de arrancar el sistema operativo Raspberry Pi lo ideal es que se ejecute automáticamente, pero no es así, por ende, el editor de flujos no se ejecutará sin que antes allí iniciado el *Node-RED console*. Existen dos maneras de la iniciación, de forma automática y manual.

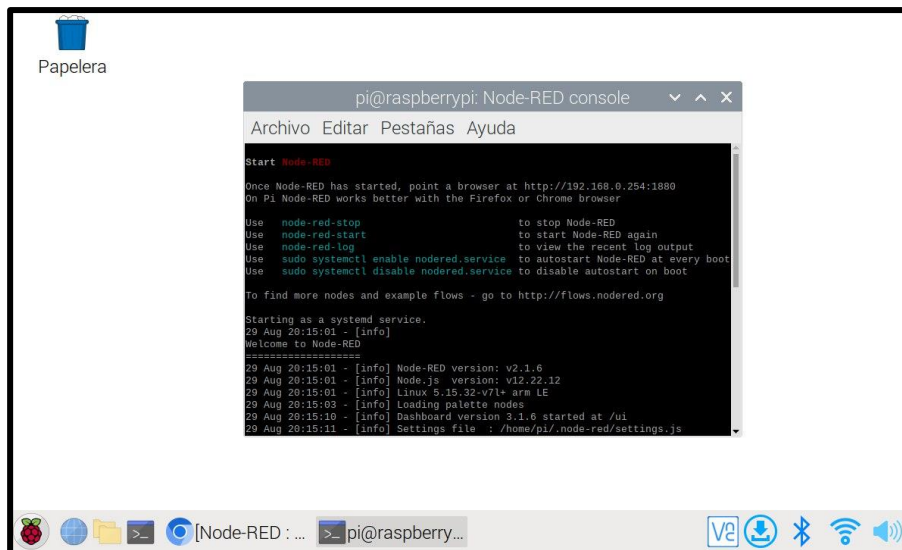
La primera, de manera manual como se muestra en la figura 19, será partiendo desde *opción menú>programación>Node-RED*

Figura 19. Inicio de Node-RED



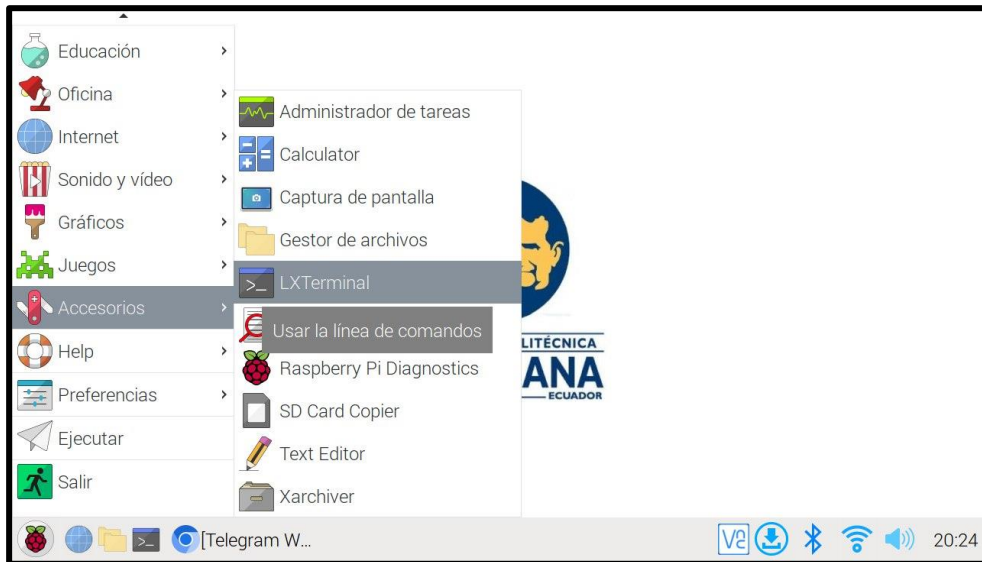
Una vez hecho esto, se muestra una ventana conocida como *Node-RED Console*, como se ilustra en la figura 20, se ejecutará Node-RED, si se cierra la ventana, deja de ejecutarse el entorno de trabajo Node-RED.

Figura 20. Terminal de Node-RED



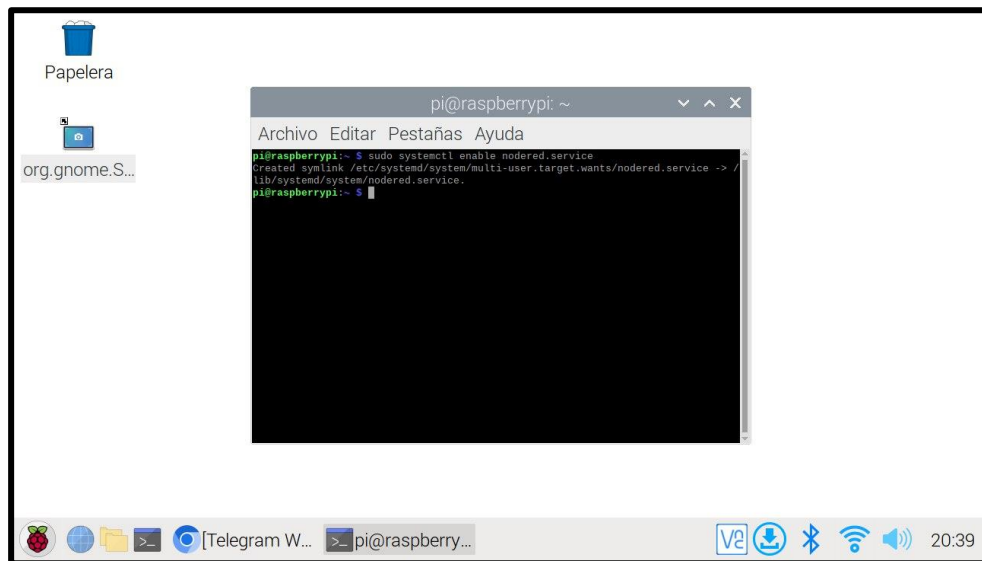
Segundo, inicio de forma automática, partiendo desde **menú>accesorio>LXTerminal**, clic y se ejecutara la terminal, como se muestra en la figura 21.

Figura 21. Ventana LXTerminal



Escribe la línea de comando *sudo systemctl enable nodered.service* se activará y ejecutará Node-RED de forma automática, sin necesidad de ejecutar *pi@raspberrypi: Node-RED console*, como se ilustra en la *figura 22* sin importar que se reinicie el sistema operativo el entorno de programación se ejecutará sin necesidad de abrir el *Node-RED console*.

Figura 22. Ventana LXTerminal



Si se desea desactivar la forma automática, en la misma ventana de **LXTerminal**, por el medio de la línea de comando *sudo systemctl disable nodered.service* se desactivará la ejecución.

### 2.13.2 Editor de flujos Node-RED

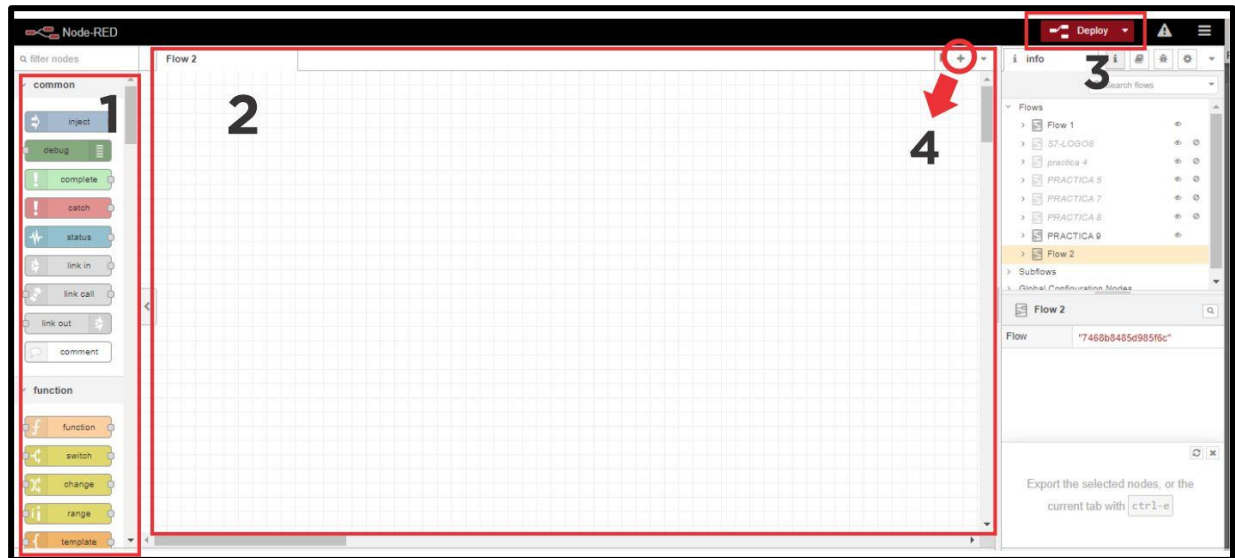
#### Espacio de trabajo

Para iniciar la IDE el desarrollo integrado de node-RED con la cual se basa en conexiones de flujos es necesario conocer las partes esenciales con un concepto general las secciones del espacio de trabajo de node-RED, como se ilustra en la figura 23 se describen las funciones principales.

1. **Palette (Paletas de nodos):** Contiene todos los nodos instalados y listos para el desarrollo de flujos, cabe señalar que están organizadas por categorías; common, fuction, network, dashboard, remote, plc, whatsApp, telegram, etc.
2. **Wokspace (Entorno de trabajo):** Es el espacio de trabajo principal en el que se desarrolla los flujos, arrastrando desde la paleta los nodos necesarios para la conexión entre sí.
3. **Deploy (implementar):** su uso es enviar y ejecutar un flujo (pestaña) de programación, cuando se ejecuta correctamente sin error sale el mensaje de **“Successfully deployed”** (implementado con éxito).
4. **Flow (flujo):** se denomina a la “pestaña” dentro del espacio de trabajo del editor node-RED, cabe señalar que, flujo se usa para describir a un conjunto de nodos conectados, por lo tanto, un flujo (pestaña ) puede contener múltiples flujos (conjunto de nodos conectados entre sí) pulsando donde se señala se puede agregar un flujo en la que se puede asignar un nombre y descripción



Figura 23. Campo de Node-RED



*Sidebar* o barra lateral contiene cuatro opciones en el que proporciona herramientas útiles dentro del editor. (Node-RED, 2022), como se ilustra en la figura 24

**5. Information (información):** aporta información acerca de los nodos.

**6. Help (ayuda):** brinda opción de ayuda.

**7. Debug (depurar):** se ven los mensajes enviados a los nodos de depurar dentro del flujo.

**8. Configuration Nodes (nodos de configuración):** esta opción brinda información por medio de una lista de todos los nodos de configuración que se utilizan y las que no, cada nodo muestra su tipo y etiqueta. (Node-RED, 2022)

**9. Context data (datos de contexto):** divisar el contenido del contexto.

**10. Dashboard (tablero):** proporciona nodos para crear interfaz de forma que se pueden visualizar y controlar equipos conectados en la misma. Para la visualización del dashboard se pulsa en la flecha superior derecho, a su vez este envía una ventana nueva con la dirección <http://localhost:1880/ui> (Local Host es la dirección ip en el cual fue iniciado node-RED ) como se muestra en la figura 25

Figura 24. Sidebar (barra lateral)

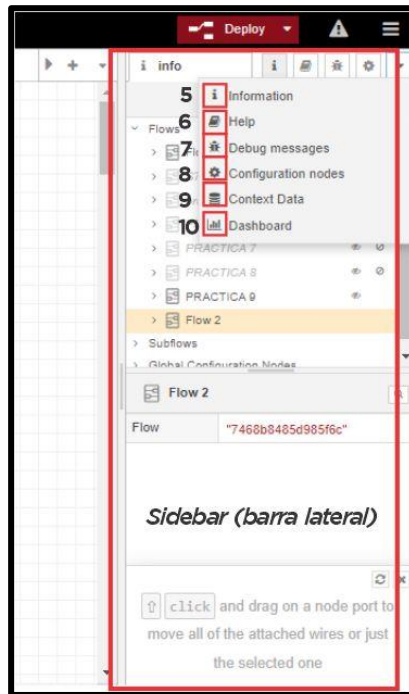
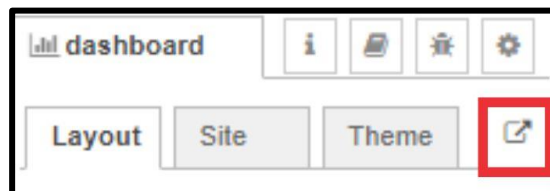


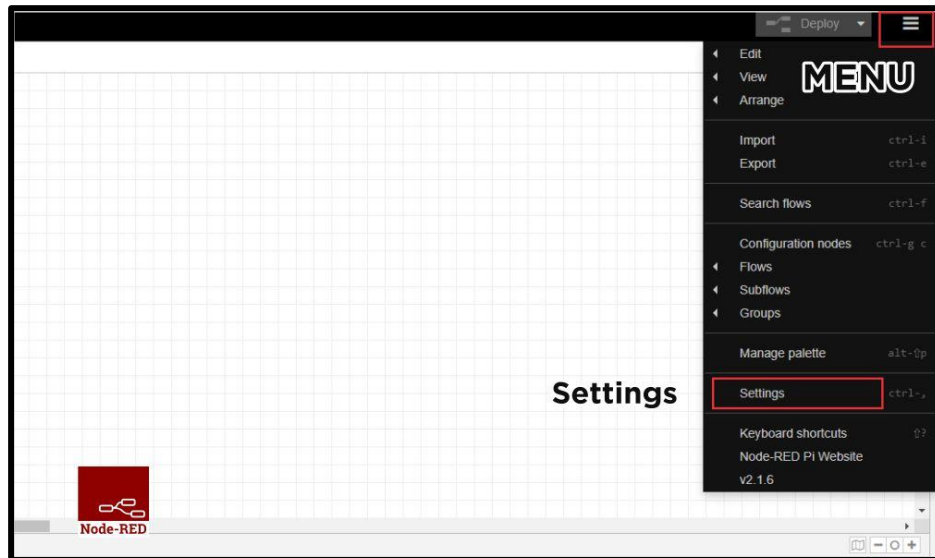
Figura 25. Arranque para la visualización del Dashboard



### 2.13.3 Instalación de complementos en el editor de flujo

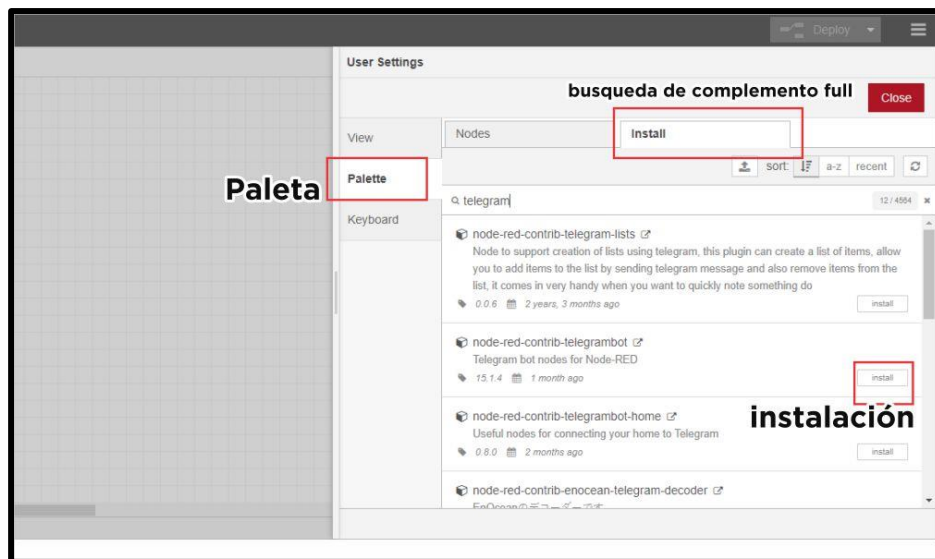
Al iniciar el entorno de trabajo en node-RED por primera vez, se observa que no existe ninguna instalación de nodos para la creación de flujos, por lo tanto, es necesario dirigirnos en la pestaña “Menu”, se dirige a la opción *Settings*, como se muestra en la figura 26

Figura 26. Pasos para instalación de complementos 1/2



Luego, se dirige a la opción **Palette - Install** puede hacer la búsqueda de los complementos de acuerdo a los que se desea instalar en la paleta de nodos, de manera que se escribe el nombre en **búsqueda** como se ilustra en la figura 27. Por último, se presiona en **install** donde empezará a instalar y al finalizar lo podrá ver en el entorno de trabajo.

Figura 27. Pasos para instalación de complementos 2/2

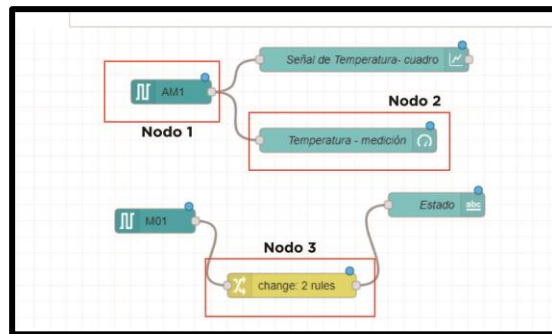


### 2.13.4 Node (Nodos)

Un nodo es un elemento básico de un flujo con la finalidad de hacer una tarea específica. (Node-RED, 2022)

Básicamente, los nodos se activan al recibir un mensaje del nodo posterior o mantiene un tiempo de espera hasta que haya recibido algún evento externo, como un temporizador, lectura de señal analógica o alguna orden desde un PLC. Procesan el mensaje o evento para luego enviar al siguiente nodo dentro del flujo. Figura 28

Figura 28. Nodos



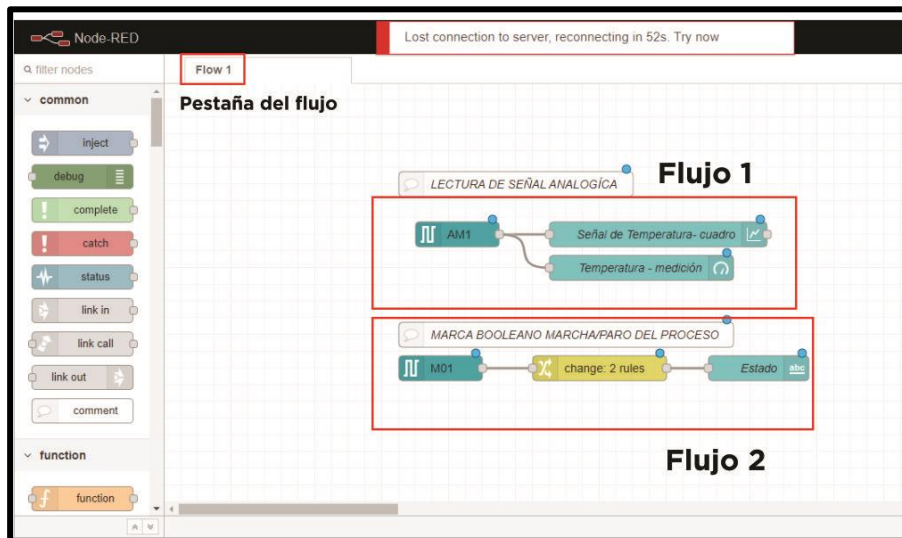
Cabe mencionar que un nodo puede tener como máximo un puerto de entrada y varios puertos de salidas como se requiera en la programación visual.

### 2.13.5 Flow (Flujo)

Se conoce a la pestaña dentro del espacio de trabajo del editor y es la forma principal de organizar los nodos. (Node-RED, 2022)

Pero, también se utiliza el término para describir a un solo conjunto de nodos conectados que conlleva la creación de un flujo más dimensionando, como se ilustra en la figura 29.

Figura 29. Diferencia de termino flujo



### 2.13.6 Mensaje

Se le denomina a toda información que pasa por los nodos de un flujo. Escritos en lenguaje JavaScript que contiene varios conjuntos de propiedades. Conocido dentro del editor *msg*.

### 2.13.7 Wire (Cable)

Son las conexiones que existen entre nodos, envía información.

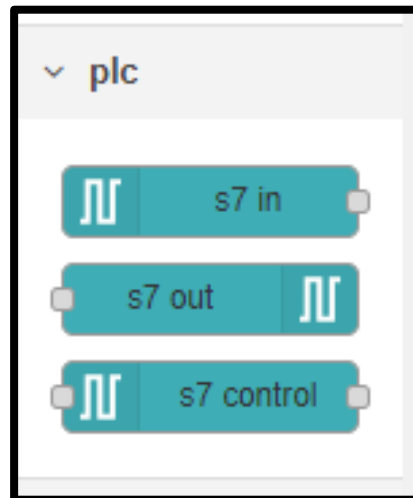
### 2.13.8 Widgets

Se define a los elementos gráficos a los que se pueden agregar a un flujo y crear interfaces interactivas a la vista del usuario, estás pueden mostrar datos, aceptar ejecuciones por parte del usuario y controlar el flujo dentro de Node-RED como son los botones, cuadro de textos, selectores, gráficos, etc. Con solo arrastrar al editor de flujo y hacer uso del lenguaje de programación como HTML, JavaScript, package.json.

### 2.13.9 node-red-contrib-s7

Es un complemento de paleta de nodos personalizado para uso en Node-RED que proporciona comunicación en PLC SIEMENS S7 a través del protocolo S7 que intercambian datos y comandos con otros dispositivos y sistemas. (Node-RED, 2022)

Figura 30. Complemento PLC S7 para node-RED



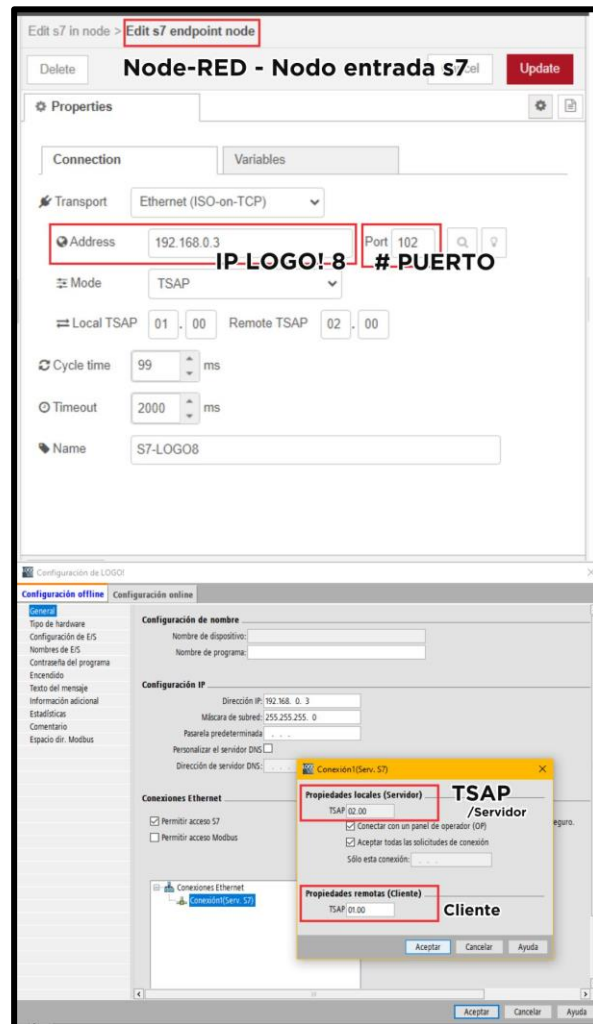
El complemento *node-red-contrib-s7* proporciona una cantidad de nodos que el usuario puede interactuar con los controladores S7 (Siemens), de los cuales pueden leer y escribir datos dando como resultado operación de control, supervisión y recopilación de datos en tiempo real.

- **Configuración de nodo *node-red-contrib-s7***

Para hacer uso de las variables del autómatas SIEMENS LOGO!8 en Node-RED es necesario realizar una serie de configuraciones específicas para comunicar entre ambos sistemas. Por ende, cumplido con la instalación del complemento *node-red-contrib-s7* se realiza el siguiente paso, que es la de especificar la dirección IP y el número de puerto del LOGO8, como ilustra en la figura 31.

Por último, el modo TSAP es parte de la conexión para determinar la comunicación entre el cliente/servidor, es decir entre Node-RED y LOGO!8, por ello ambas partes deben estar configurado.

Figura 31. Configuración IP, puerto y TSAP dentro del nodo entrada S7



- **Uso de Variables LOGO!8 en Node-RED**

Al hacer uso de variables del LOGO! 8 dentro de Node-RED respecto en la interacción con diferentes tipos de datos lectura o escritura, es necesario realizar unas configuraciones prescrita en la documentación de Node-RED se ilustra en la figura 32 nos detalla las direcciones de registro o las áreas de memoria que corresponden al LOGO!8.

Figura 32. Tabla para el uso de variables en Node-RED por S7

Bloque de logotipo	Rango de máquina virtual del logotipo	ejemplo de dirección Nodo-RED	Descripción
I	1024 - 1031	DB1, BYTE1024 o DB1, X1024.5 o DB1, WORD1024	Lee los terminales de entrada 1...8 o 6 o 1...16
AI	1032 - 1063	DB1, WORD1032	Lee el terminal de entrada analógica 1. Siempre tamaño palabra.
Q	1064 - 1071	DB1, BYTE1064 o DB1, X1064.5 o DB1, WORD1064	Lee los terminales de salida 1...8 o 6 o 1...16
AQ	1072 - 1103	DB1, WORD1072	Lee el terminal de salida analógica 1. Siempre tamaño palabra.

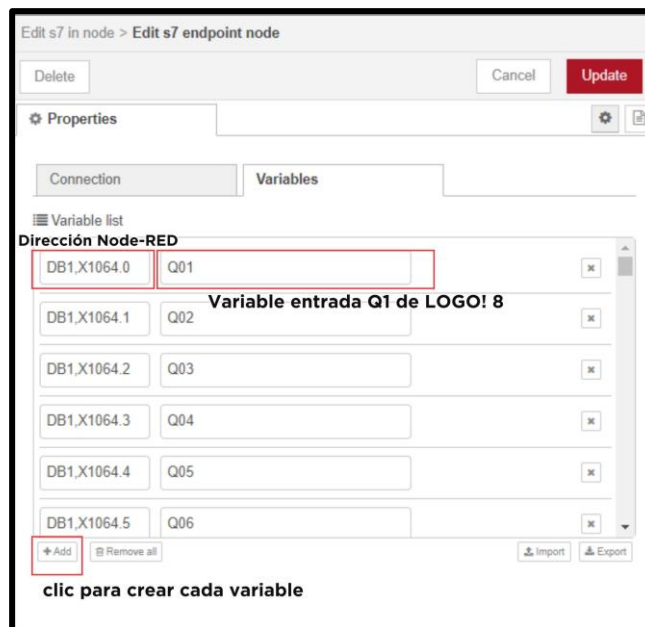
  

M	1104 - 1117	DB1, BYTE1104 o DB1, X1104.5 o DB1, WORD1104	Lee indicadores de bits M1...M8 o M6 o M1...16
AM	1118 - 1245	DB1, WORD1118	Lee la bandera analógica 1. Siempre tamaño palabra.
NI	1246 - 1061	DB1, BYTE1246 o DB1, X1246.5 o DB1, WORD1246	Lee la entrada de red 1...8 o 6 o 1...16
NAI	1262 - 1389	DB1, WORD1262	Lee la entrada de red analógica 1. Siempre tamaño palabra.
HQ	1390 - 1405	DB1, BYTE1390 o DB1, X1390.5 o DB1, WORD1390	Lee la salida de red 1...8 o 6 o 1...16
NAQ	1406 - 1469	DB1, WORD1406	Lee la salida de red 1. Siempre tamaño palabra.

Fuente: (Node-RED, 2022)

Para ello, se debe **crear una lista de variables** en el que especifique el nombre de cada entrada, salida, marca, entrada analógica o cualquier bloque de datos que requiera representar y hacer uso del autómatas, con el fin de no utilizar la direcciones que nos detalla en su documentación Node-RED, sino para tener una referencia más sencilla de las variables y tipos de datos a utilizar como se ilustra en la figura 33.

Figura 33. Lista de variables de entradas de LOGO!8





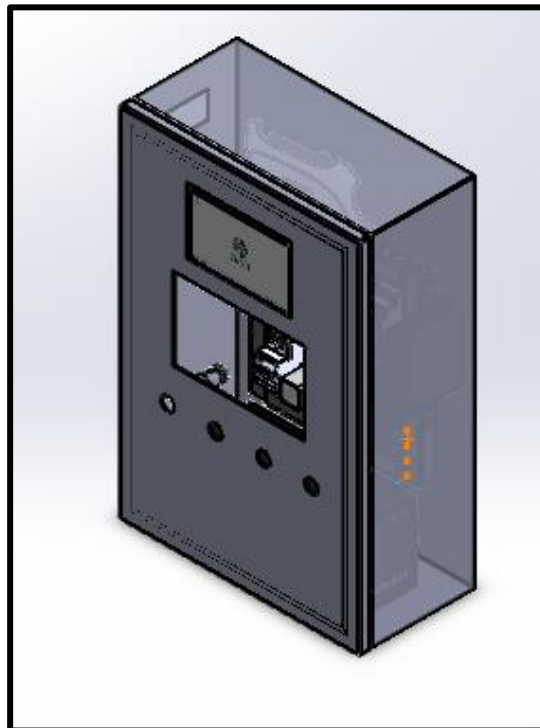
## 3. MARCO METODOLÓGICO

### 3.1 Diseño del módulo didáctico

#### 3.1.1 Desarrollo del Tablero de control

Para el desarrollo del módulo didáctico, como se ilustra en la figura 34, se realizó un estudio y cotizaciones de los equipos y materiales que serían parte de este proyecto técnico. Costos de los cuales los integrantes de este proyecto los cubrieron. Varias marcas de equipos fueron utilizadas de lo que es el Mini PLC, tarjeta mini PC, Disyuntores, Contactores, Relé térmico, sensores industriales, Fuente de alimentación, Pantalla LCD Touch, Luces pilotos, Regulador.

Figura 34. Vista del módulo didáctico



#### 3.1.2 Desarrollo de la Planta industrial

La estructura del módulo didáctico está hecha de metal, figura 35, así como la base que asienta el tablero de control y los tanques de reservorios como se ilustra en la figura 36.

Figura 35. Tanques del módulo didáctico

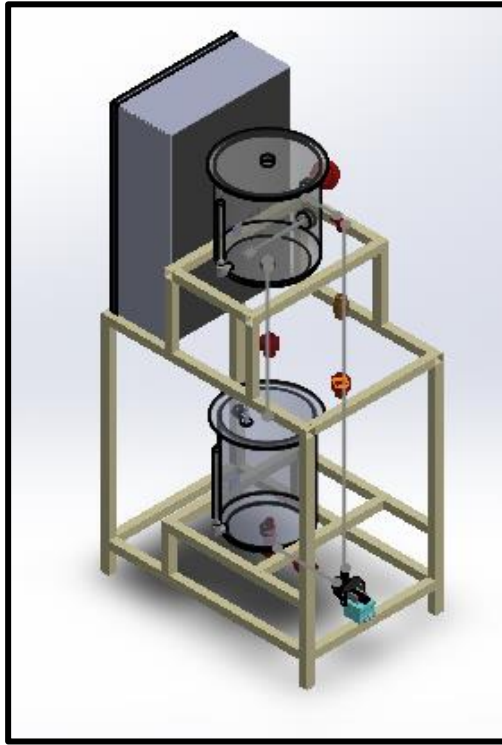


Figura 36. Diseño en 3D del módulo didáctico



## 3.2 Elementos que conforma el módulo didáctico

### 3.2.1 LOGO! 8 versión 8.3

Actualmente, la marca ha lanzado una nueva actualización LOGO! versión 8.3 como se ilustra en la figura 37, lo novedoso, es la posibilidad de activar y configurar la conexión a la nube, donde los usuarios pueden monitoreo y operarlos desde la nube por cloud computing (computación en la nube) estos servidores web gratuitos o pagados, a través de teléfonos inteligentes, tarjetas electrónicas inteligentes, PC. Otro aspecto para recalcar son las múltiples opciones de interfaz como MODBUS TCP/IP, Ethernet. (SIEMENS, 2020)

Figura 37. Mini PLC LOGO 8 a la versión 8.3



Fuente: (SIEMENS, 2020)

### 3.2.2 Módulo de expansión LOGO 8 AM2 RTD 2AI SIEMENS

El módulo de expansión AM2 RTD nos permite leer dos canales analógicos de termorresistencia (PT-100 y PT-1000) en un rango de temperatura desde -50 hasta 200 grados C, como se ilustra en la figura 38.

## Características

- Tensión de alimentación: 24 VDC
- Entradas analógicas: 2
- Tipo: RTD
- Grado de protección según EN 60529
- Temperatura de empleo: 0°C a 55°C
- Homologación CSA
- Homologación UL
- Homologación FM

Figura 38. Módulo de expansión 6ED1055-1MA00-0BA2



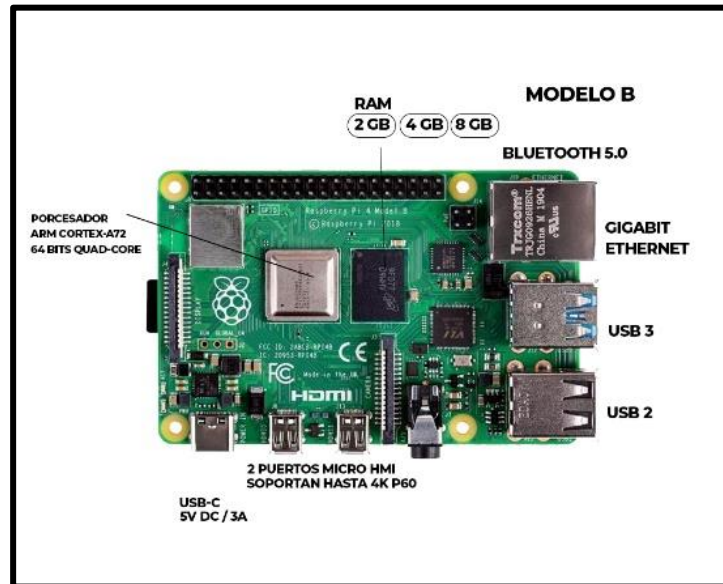
Fuente: (SIEMENS, 2020)

### 3.2.3 Raspberry Pi 4

La Raspberry Pi 4, Como se muestra en la figura 39, se puede usar como una Mini PC, con la apuesta de una memoria adicional funciona sin problema para tareas de escritorio, navegar en la web, usar en línea servicios, además, no resulta un problema al utilizar sitios web grandes ya que

con la memoria 4 GB RAM el proceso mucho más rápido. Las especificaciones y el peso ligero lo hacen un “poderoso mini PC”. (Richard, 2022)

Figura 39. Placa interna Raspberry Pi 4 Modelo B



### 3.2.4 ARGON ONE PI 4

Carcasa para la Raspberry Pi 4 fabricada en aleación de aluminio con un acabado que da una vista de elegancia y modernidad, su diseño permite la distribución de sus entradas USB, puerto C, botón de encendido, puerto Mini-HDMI de un solo lado que facilita la conexión de los cables, además actúa como disipador por su estructura, como se ilustra en la Figura 40.

Figura 40. Argon one pi 4



### 3.2.5 Interruptor Termomagnético tipo C

Es un dispositivo que resulta útil en protección contra sobrecargas y cortocircuitos, su funcionamiento se basa en el magnetismo y calor, interrumpe la corriente eléctrica cuando detecta ciertos límites, figura 41.

Figura 41. Interruptor termomagnético



### 3.2.6 El Sensor Ultrasonido UK1A/G7-0ESY

Es un sensor ultrasonido cilíndrico M18 de reflexión directa y retro reflectante, tiene un rango de detección hasta 400 mm, opera en una frecuencia de 40 kHz con un tiempo de respuesta de 500

ms y una precisión de  $\pm 0,5\%$  de la distancia medida, tiene una salida PNP +0...10V , temperatura de funcionamiento de - 20°C...+ 70°C, histéresis de 1%, error de linealidad 1% como se muestra en la figura 42

Figura 42. Sensor ultrasonido UK1A/G7-0ESY



### 3.2.7 Sensor PT-100 Ø 6 DIN / Cabezal – PIXSYS

La función principal del sensor consiste en el que la resistencia eléctrica cambia en función de la variación de temperatura. La composición de este sensor de temperatura, figura 43, conocido como termorresistencia están compuesto de sensores de platino, y además una capa delgada de platino en el sustrato de cerámico para su medición sea precisa, Su valor óhmico es de 0 a 100°C, dado que la resistencia de los cables se sumas con la resistencia de sensor (variable de temperatura), se utiliza un sistema de tres hilos para anular el error. (Pixsys Electronics, 2022)

Figura 43. Termorresistencia PT-100



### **3.2.8 Router inalámbrico TP-LINK TL-WR840N**

Es la responsable de llevar al ahora de una transferencia fiable de datos, sin errores, control de flujo de información a través de una red de comunicación. Las características de este dispositivo se desarrollan con cable/inalámbrico en la conexión de una red ya diseñada, con una velocidad de transmisión inalámbrica de 300 Mbps, control de banda ancha basada en IP permite que los administradores determinen cuanta banda ancha está distribuida a cada dispositivo conectado.

Equipado con 4 puertos ETHERNET rápidos, su alta velocidad y estabilidad permite tener un mejor desempeño al momento de conectar varios dispositivos, figura 44 es fácil su configuración, su uso permite hasta 4 modo; punto de acceso, extensor de rango, router, WISP (TP-LINK, 2022).



Figura 44. Router TP-LINK TL-WR840N



Fuente: (TP-LINK, 2022)

### **3.2.9 Fuente de alimentación AUTONIC 50 W 100/240 VAC /24VDC**

Aplicable a cualquier maquinaria y sistema de comunicación industrial en la que se requiere de baja potencia así mismo que sea de peso ligero para su instalación, figura 45

#### **Especificaciones**

- Protección sobre corriente
- Protección contra cortocircuito
- Protección corriente de irrupción
- Cumple con regulaciones de seguridad IEC60950 y IEC 50178
- Voltaje de salida 5VCC, 12VCC, 24VCC
- Frecuencia 50/60 Hz
- Alimentación: 100-240VCA
- Salida de corriente: 2.1 A
- Potencia de salida: 50W

- Medioambiente \_humedad 25 a 85% RH
- Ambiente Temperatura ambiente -10 a 50 °C
- Peso: Aprox 350g

Figura 45. Fuente de alimentación AUTONIC



Fuente: (AUTONIC, 2022)

### 3.2.10 UPS Serie NT-500 - FORZA

Sistema de alimentación ininterrumpida (UPS Sigla en inglés) es un equipo que permite la prolongación de la energía eléctrica por un tiempo limitado, figura 46, gracias a su batería y componentes ayudan a almacenar la energía, eso ayuda a evitar corte de manera inesperada o “apagón” de los dispositivos que se encuentra conectados en el UPS. (FORZA, 2022)

Especificaciones técnicas

- Entrada: 120V
- Margen de tensión: 78-145 VAC
- Tipo de enchufe: 1 NEMA 5-15P
- Frecuencia: 50/60 Hz
- Protección exclusiva contra sobretensión
- Tipo y numero de batería: 12v 4.5Ah (1)

- Temperatura de funcionamiento: 0-40°C
- Temperatura de almacenamiento: -15-45 °C
- Carcasa: cubierta de material termoplástica, retardador de llama
- Dimensión de la unidad: 279x101x142 mm
- Recarga automática: Si
- Tiempo de autonomía: 18 min
- Tiempo de recarga: hasta un 90% de su capacidad en 6 horas

Figura 46. Forza Serie NT-500



Fuente: (FORZA, 2022)

### 3.2.11 Relé de estado sólido 480 VAC/ 32 VDC – XURUI

Un relé de estado sólido es un conmutador electrónico que ofrece aislamiento eléctrico entre un circuito de control (entrada) y un circuito de carga (salida), su ventaja al no tener partes móviles ofrece una mejor conmutación en velocidad de tiempo nanosegundo a microsegundo, frente a los milisegundos, no produce ruido acústico, es invulnerable ante vibraciones y choques mecánicos, es poco la interferencia electromagnética, figura 47. (Areny, 2005)

#### Características

- Rango de voltaje de control: 3VDC ~ 32VDC
- Rango de corriente de control: 6mA ~ 25mA
- Voltaje reverso: 32VDC
- Rango de voltaje de carga: 75VAC ~ 480VAC
- Rango máximo de corriente de carga: 0.08A ~ 75<sup>a</sup>
- Temperatura de trabajo: -20°C ~ 70°C

Figura 47. Relé XURUI



### 3.2.12 Resistencia calefactora 2000W-220V

La función de la resistencia calefactora tiene como finalidad la de calentar distintos materiales, fluidos, aire o gases a distintas temperaturas, dependiendo de su aplicación en el proceso, por lo cual, existen diferentes tipos de resistencia calefactoras, figura 48.

Figura 48. Resistencia Calefactora para inmersión



### 3.2.13 Relé de bobina

Conocida como relé electromagnético, su específica función es la de separar entre la corriente de accionamiento y los circuitos controlados por los contactos, estos hacen que puedan operar altos voltajes con pequeñas tensiones de control, figura 49.

#### Características

- Marca: CGC
- Modelo: MK2P-I
- Voltaje de bobina: 28 VDC
- Capacidad de contacto: 7A 250VAC 10A 28VDC
- Números de pines: 8 pines

Figura 49. Relé de bobina



### 3.2.14 Relé de nivel CAMSCO

Apto para abastecimiento de agua y drenaje, con un montaje fácil de instalar y reemplazar, trabaja en función de 3 sondas E1-E2-E3 las que proporcionan la señal de activar y desactivar dependiendo de su configuración, el voltaje de señal que envía las sondas E1-E3 es bajo 8V AC, su tensión nominal es 110 V o 220 V AC, figura 50.

Figura 50. Relé de nivel y Sondas marca CAMSCO



### 3.2.15 Válvula Solenoide UNI-D UW-15

La función principal de esta válvula es la de abrir y cerrar el paso de fluido, aplicado en procesos residenciales como industriales, su construcción se basa en rosca hembra de ½ pulg NPT de 2 vías, cuerpo de Latón, material de sello NBR, bobina de 24V DC con un rango de temperatura de -5 a 85°C, figura 51.

Figura 51. Válvula solenoide UNI-D UW-15



### 3.2.16 Pantalla Touch 7" LCD


Pantalla LCD táctil en toda su composición, está diseñada básicamente para raspberry 3, 4. Su compatibilidad es estándar en todo los Mini Pc con un sistema Raspbian, Ubuntu y Windows 10, con una resolución de hardware 1024 x 600 hasta 1920 x 1080 por software, figura 52.

Figura 52. Pantalla touch 7"




## 4. PLANTEAMIENTO Y DESARROLLO DE LAS PRÁCTICAS

### 4.1 Práctica #1


	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA <b>#1</b>	TÍTULO DE LA PRÁCTICA Cableado de control y potencia de módulo de instrumentación.
OBJETIVOS	
<ul style="list-style-type: none"><li>• Reconocimiento del módulo didáctico por parte del estudiante como inicio de práctica.</li><li>• Interpretación de esquemas de control y potencia para realizar conexiones eléctricas</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Realizar esquemas de control y potencia.</li><li>• Armar cables tipo plug para realizar conexiones de manera segura sin contacto directo con las partes activas.</li><li>• Realizar conexiones eléctricas de los elementos según diagramas.</li><li>• Encendido y puesta en marcha.</li><li>• Verificación de funcionamiento de cada una de las partes del módulo.</li></ul>	
DESARROLLO ENCONTRARÁ EN ANEXO	




## 4.2 Práctica #2

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA #2	TÍTULO DE LA PRÁCTICA Elaboración de diagrama PI&D de módulo de instrumentación de procesos con el software AutoCAD electrical.
OBJETIVOS	
<ul style="list-style-type: none"><li>• Realizar diagrama de tuberías e instrumentación aplicando la norma ISA 5.1</li><li>• Definir tabla de símbolos según norma ISA 5.1</li><li>• Utilizar librería P&amp;ID de AutoCAD electrical</li><li>• Definir tipos de líneas según la función dentro del proceso</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Elaboración de diagrama de tuberías e instrumentación indicando el flujo de proceso y también el tipo de líneas que corresponda, ya sea una tubería, señal eléctrica, señal electromagnética.</li><li>• Utilizar la librería P&amp;ID para el desarrollo del diagrama en AutoCAD electrical.</li><li>• Crear formato de dibujo en AutoCAD electrical para cada una de las páginas del diagrama donde se mostrarán:<ul style="list-style-type: none"><li>• Diagrama P&amp;ID.</li><li>• Tabla de letras y símbolos.</li><li>• Tabla de tipos de línea</li></ul></li></ul>	
DESARROLLO ENCONTRARÁ EN ANEXO	

### 4.3 Práctica #3

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA #3	TÍTULO DE LA PRÁCTICA Elaboración de lazos de control con diagrama PI&D
OBJETIVOS	
<ul style="list-style-type: none"><li>• Definición del comportamiento del actuador según el tipo de lazo.</li><li>• Diseño de lazos de control.</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Elaboración de lazos de control y tipos de lazo de cada actuador, con su respectivo instrumento de proceso para la realimentación al controlador LOGO.</li><li>• Definir número de lazos.</li><li>• Definir tipo de control para cada actuador.</li></ul>	
DESARROLLO ENCONTRARÁ EN ANEXO	

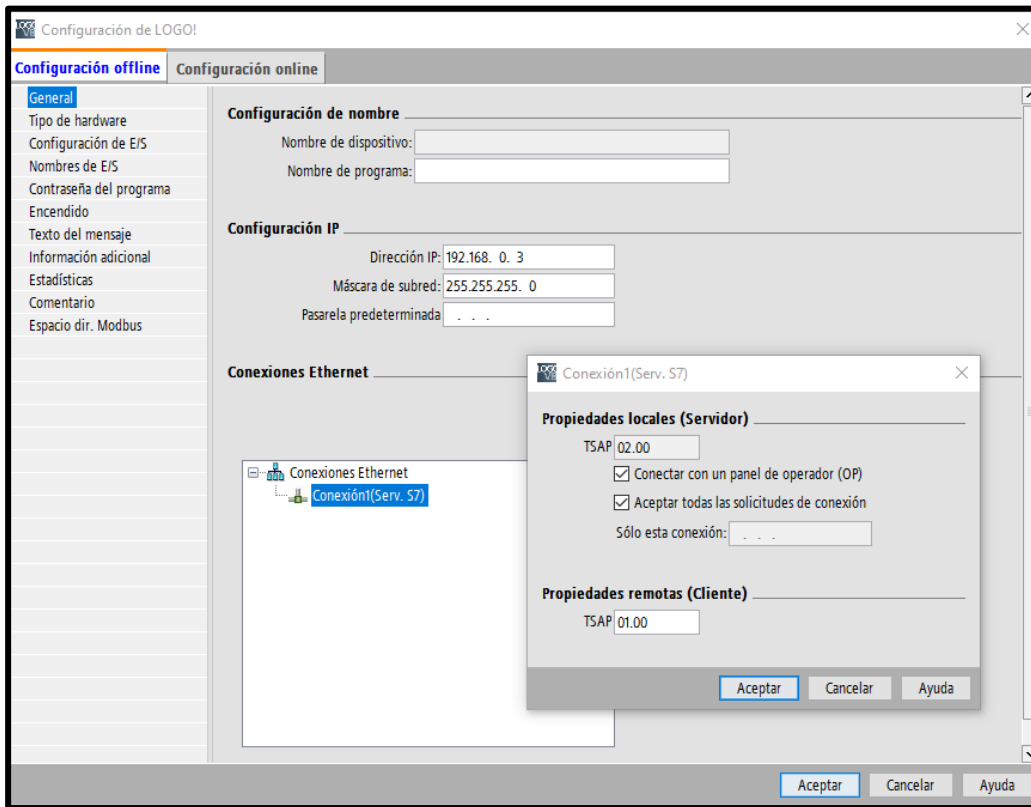
#### 4.4 Práctica # 4

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA #4	TÍTULO DE LA PRÁCTICA Control PWM de sistema de calentamiento con LOGO SIEMENS 8 y Node-red.
OBJETIVOS	
<ul style="list-style-type: none"><li>• Efectuar la comunicación entre Mini PLC LOGO V8.3 y Raspberry Pi 4</li><li>• Entender la funcionalidad de programación basada por motor de flujo por Node-RED.</li><li>• Desarrollar el control y visualización en tiempo real el proceso por medio de una interfaz pantalla touch.</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Crear conexión S7 de servidor entre LOGO! Siemens y Node-Red a través de TSAP.</li><li>• Instalar los complementos necesarios para creación de flujos.</li><li>• Adaptación del nodo S7 a LOGO! a través de importación de código JSON en el sitio web de Node Red.</li><li>• Desarrollo de programación FUP en LOGO! Soft Comfort para control de resistencia por modulación por ancho de impulso (PWM).</li><li>• Definir límites de temperatura no mayor a 50°C.</li><li>• Crear 3 grupos para organizar la interfaz del dashboard.</li><li>• Crear interfaz gráfica (dashboard) para la visualización de la temperatura por nodos Gauge y Chart.</li><li>• Crear texto de avisos de lectura en la pantalla LOGO! 8</li></ul>	

- **Resolución de la práctica 4**

Para el desarrollo de esta práctica, se sugiere que se empiece con la configuración IP para compilar de manera inalámbrica por medio de la Red que nos proporciona el router, y luego con la comunicación S7 que nos proporciona con Node-RED por TSAP, como se ilustra en la figura 53.

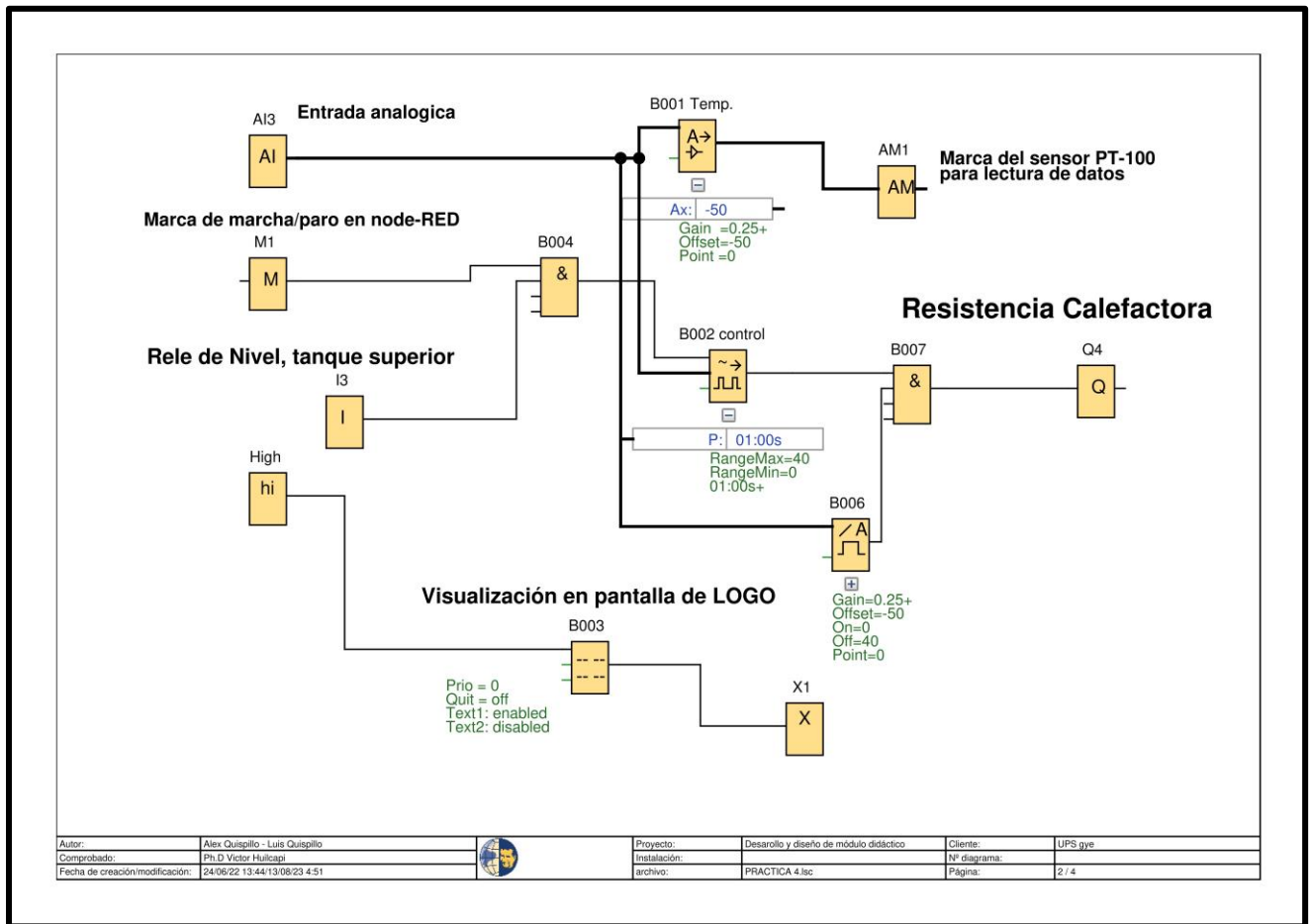
Figura 53. Comunicación entre LOGO 8 y Raspberry Pi 4



- **Programación FUP en LOGO! Soft Comfort práctica 4**

En el desarrollo de la programación se utilizaron compuertas lógicas como; amplificador analógico, entrada analógica para la lectura del sensor PT-100, control PWM, AND, salida, marcas, texto de aviso para la visualización en la pantalla de LOGO! 8, conmutador analógico de valor umbral, como se ilustra en la figura 54

Figura 54. Práctica 4 LOGO! Soft Comfort



- **Instalación de los complementos en el editor de flujo**

Como se observa en la tabla 2 se encuentran los complementos a utilizar en esta práctica para el desarrollo de los flujos en el entorno de trabajo Node-RED.

Tabla 2. Complementos de práctica 4

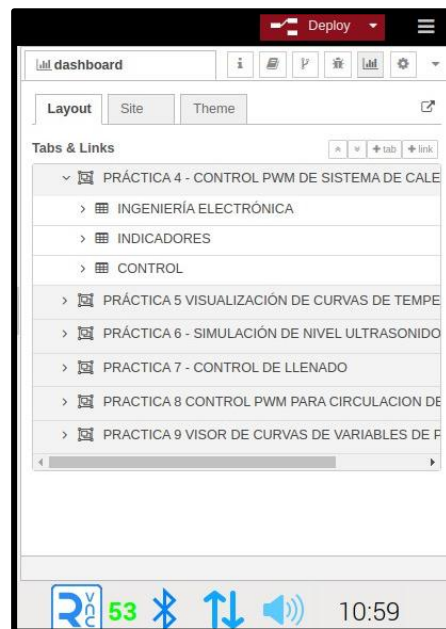
node-red	node-red-contrib-ui-led	node-red-dashboard
node-red-contrib-s7	node-red-contrib-ui-media	

- **Creación de grupo en Dashboard**

Para crear se dirige en el parte superior derecho encontrará los iconos *Dashboard >Layout >+tab* allí se crea la tabla.

Allí se crea los grupos que están dentro de la tabla que se darán nombre a *Práctica 4 - Control PWM De Sistema De Calentamiento*, de manera que, los nombres de los grupos creados para esta práctica darán él orden a los widgets para la interfaz, como se ilustra en la figura 55.

Figura 55. Creación de los grupos para el dashboard



- **Creación de flujo en workspace de Node-RED práctica 4**

Dentro del entorno de trabajo lo primero es crear un flujo (pestaña) es allí donde se comienza a desarrollar la programación visual por flujos.

Como se observa en la tabla 3 se adjunta la configuración de los nodos utilizados como *s7 in*, *change*, *text*, *button*. Cabe mencionar que cada nodo comprende una configuración distinta como se indica en la tabla.

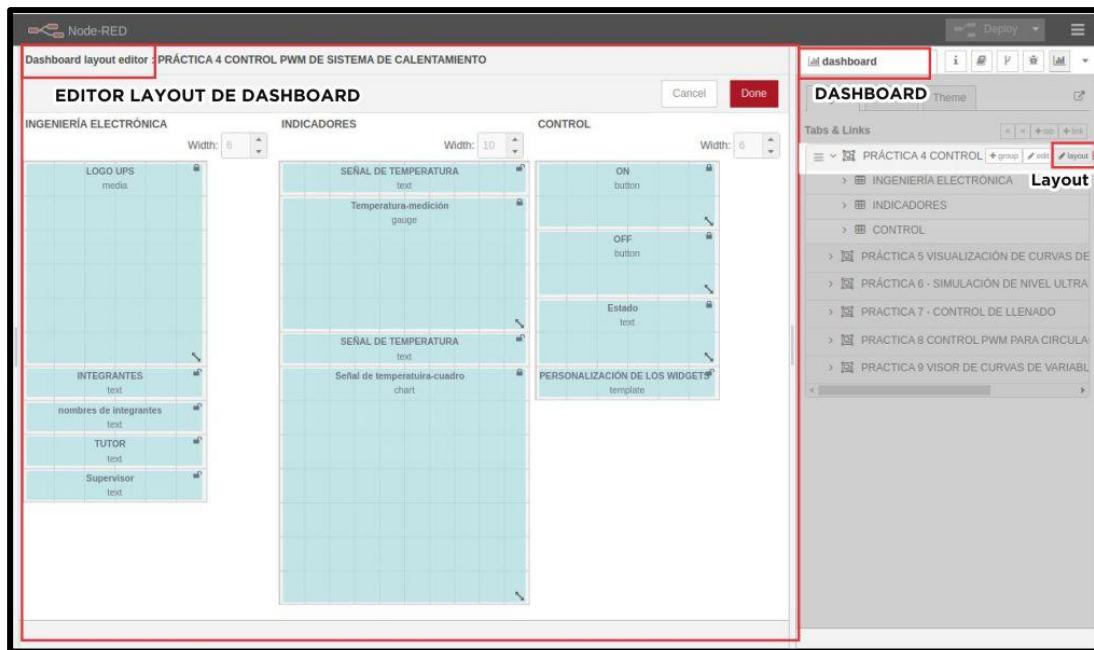
Tabla 3. Configuración de los nodos utilizados en la práctica 4

LECTURA DE SEÑAL ANALÓGICA		MARCA BOOLEANO MARCHA/PARO DEL PROCESO		LENGUAJE DE PROGRAMACIÓN JSON
<p><b>AM1 (S7 in)</b>                      PLC: S7-LOGO8                      Mode: Single variable                      Variable AM1</p>	<p><b>Señal de temperatura - cuadro (chart)</b>  <b>Group:</b>                      Type: line chart                      X-axis: last 3 /minute                      X-axis-Label: HH:mm:ss</p>	<p><b>ON (button)</b>                      Group: Nombre del grupo                      Size: 6x2                      Label: ON                      Background: green                      Payload: true                      &lt;/&gt;Class: boton</p>	<p><b>OFF (button)</b>                      Group: Nombre del grupo                      Size: 6x2                      Icon: code                      Label: OFF                      Background: red                      Payload: false                      &lt;/&gt;Class: boton</p>	<p><b>Nodo template</b>                      .estado {                          font-size: 0.6cm;                          color: white !important;                      }                      .nr-dashboard-text.estado {                          background-color: #1A237E;                      }                      .boton button {                          border: none;                          border-radius: 20px;                          text-align: center;                          font-size: 0.65cm;                      }                      .boton button:hover {                          /*width: 50%;*/                          background-color: #F57C00;                          color: black !important;                      }                      .nr-dashboard-text.otros {                          border.width: 4px;                          border-color: #3E2723;                          border-style: solid;                      }                      .nr-dashboard-text.encabezado {                          background-color: #3E2723;                          color: white !important;                      }</p>
<p><b>Temperatura – medición (gauge)</b>                      Group: nombre del grupo                      Type: Gauge                      Label: temperature                      Range: 0 – 100                      Sectors: 0...30...35...100</p>		<p><b>M01 (S7 in)</b>                      PLC: S7-LOGO8                      Variable: M01</p>		
CAMBIO DE TIPO DE SEÑAL BOOLEAN/STRING		NODOS DE TEXTOS		
<p><b>M01 (S7 in)</b>                      PLC: S7-LOGO8                      Mode: Single variable                      Variable M01</p>	<p><b>Change: 2 rules</b>                      Change/ msg/ payload                      Search for: Boolean (true)                      Replace with: string/Habilitado                       Change/ msg/ payload                      Search for: Boolean (false)                      Replace with: string/Desahabilitado</p>	<p><b>Integrantes (text)</b>                      Group: nombre del grupo                      Value format:  <b>INTEGRANTES</b>                      Layout: center                      &lt;/&gt;Class: encabezado</p>	<p><b>LOGO UPS configuration</b>                      Group: nombre del grupo                      Suze:6x6                      Layout: center  <b>Files</b>                      Category: Mis imagenes                      File: Salesiana.jpg</p>	
<p><b>Estado (text)</b>                      Group: Nombre del grupo                      Label: Estado                      Layout: center                      &lt;/&gt;Class: estado</p>		<p>Group: nombre del grupo                      Value format: <b>SEÑAL DE TEMPERATURA</b>                      Layout: center                      &lt;/&gt;Class: encabezado</p>		

Recordar que cada uno de los nodos del complemento *node-red-dashboard* se los deben asignar a los grupos ya antes creado dentro de la tabla.

Los *widgets* en la interfaz se ordenan de manera automática, pero lo ideal es personalizar, por ende, se debe configurar en *Layout* en la opción *dashboard*, como se ilustra en la figura 56.

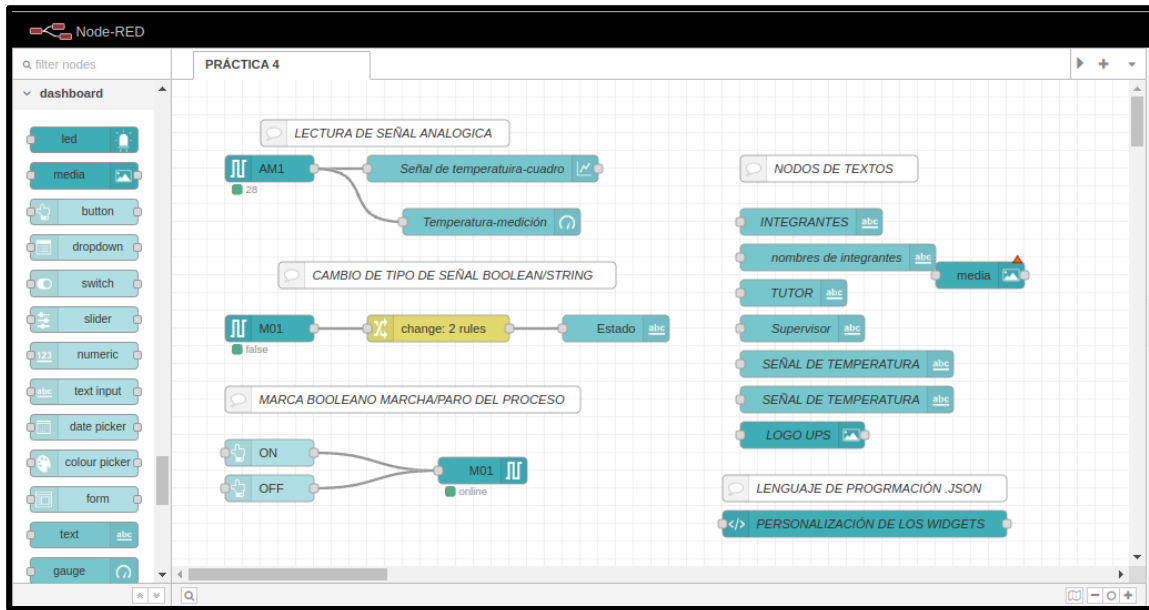
Figura 56. Configuración en Layout práctica 4



En la figura 57 se adjunta la programación en cómo se quedó detallado referente a esta práctica con sus respectivos comentarios, de esta manera se da orden a la programación y tener una referencia al momento de realizar alguna reestructuración.



Figura 57. Programación visual de la práctica 4 completa




Finalmente, en la figura 58 se ilustra la creación del dashboard de la práctica 4. La interacción con la interfaz gráfica se produce al tener programado y compilado en LOGO! 8.

Figura 58. Dashboard de la práctica 4



## 4.5 Práctica #5

	<p>GUÍA DE PRÁCTICA DE LABORATORIO</p>
<p>CARRERA Ingeniería en electrónica</p>	<p>MATERIA</p>
<p>PRÁCTICA  #5</p>	<p>TÍTULO DE LA PRÁCTICA  Creación de dashboard en motor de flujos servidor web Node-RED para visualización de curvas en tiempo real de temperatura y nivel</p>
<p>OBJETIVOS</p> <ul style="list-style-type: none"> <li>• Efectuar la comunicación entre Mini PLC LOGO V8.3 y Raspberry Pi 4</li> <li>• Entender la funcionalidad de programación basada por motor de flujo por Node-RED y la programación en LOGO! 8 en diagrama FUP para su respectiva simulación.</li> <li>• Simulación de proceso del control de llenado, calentamiento y recirculación de líquido en dos tanques.</li> <li>• Orientar al estudiante en el desarrollo, control y visualización en tiempo real del proceso por dashboard desde la pantalla Touch conectada a la Raspberry Pi 4.</li> </ul>	
<p>PLANTEAMIENTO DE PROBLEMAS</p>	
<ul style="list-style-type: none"> <li>• Establecer comunicación entre la Raspberry Pi 4 y LOGO! 8</li> <li>• Simular el proceso de llenado, calentamiento y recirculación del líquido, tomando en cuenta las recomendaciones del caso:</li> <li>• Primero, el tanque 1 debe estar lleno de agua hasta el último electrodo de nivel que es E1, un nivel ideal que el líquido llegue a la punta del electrodo, nomás de 5 cm pasado, este indica que el tanque esta llenó. La resistencia se activa automáticamente al compilar el editor de flujo, luego, se desactiva al activar de manera manual el swich Válvula Solenoide, este debe descarga en el tanque 2 hasta que se desactive el relé de nivel, desactiva la válvula para luego activar el swich de la Bomba la cual comenzara</li> </ul>	

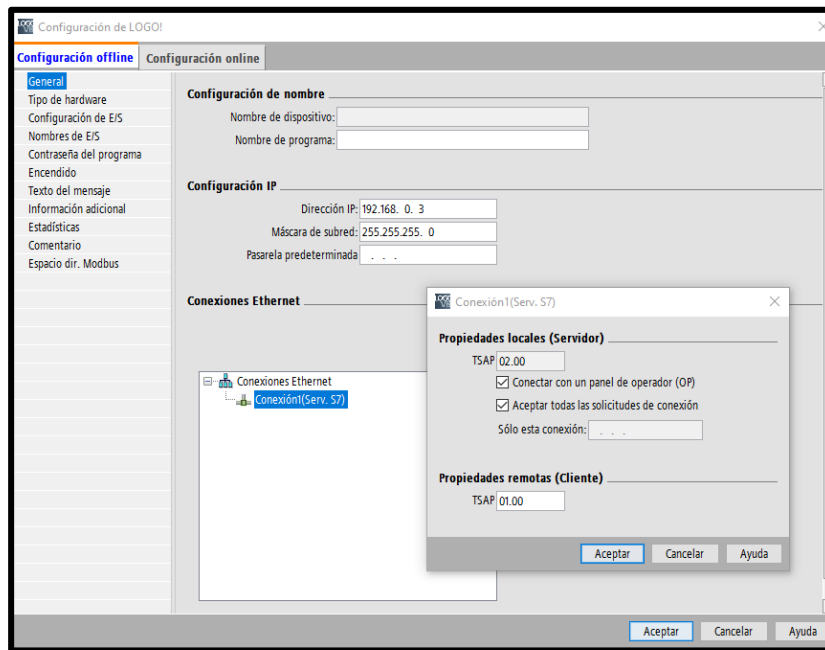
a llenar en el tanque 1 hasta que el relé de nivel se active indicando lleno en ese mismo instante de desactiva la bomba y se activa la Resistencia Calefactora automáticamente. Con el propósito de visualizar en el dashboard en tiempo real la señal analógica que envía la PT-100 y sensor ultrasónico. Por ultimo se pone un botón de PARO DE EMERGENCIA para desactivar todas las salidas.

- Desarrollar el diagrama de funciones FUP en LOGO! Soft Comfort y la programación visual en el editor de flujo Node-RED para la respectiva simulación de la práctica.
- Desarrollar los widgets para la visualización en el dashboard de esta práctica, tomando en cuenta lo que se debe mostrar: Logo de la institución, nombre de los integrantes y docente, la visualización por grafica el comportamiento de temperatura y nivel ultrasonido en tiempo real, los switches de accionamiento válvula y bomba, luces de piloto para la resistencia y relé de nivel.
- Identificar cada nodo utilizado en el editor de flujo Node-RED.

- **Resolución de la práctica 5**

Para el desarrollo de esta práctica, se sugiere que se empiece con la configuración IP para compilar de manera inalámbrica por medio de la Red que nos proporciona el router, y luego con la comunicación S7 que nos proporciona con Node-RED por TSAP, como se ilustra en la figura 59.

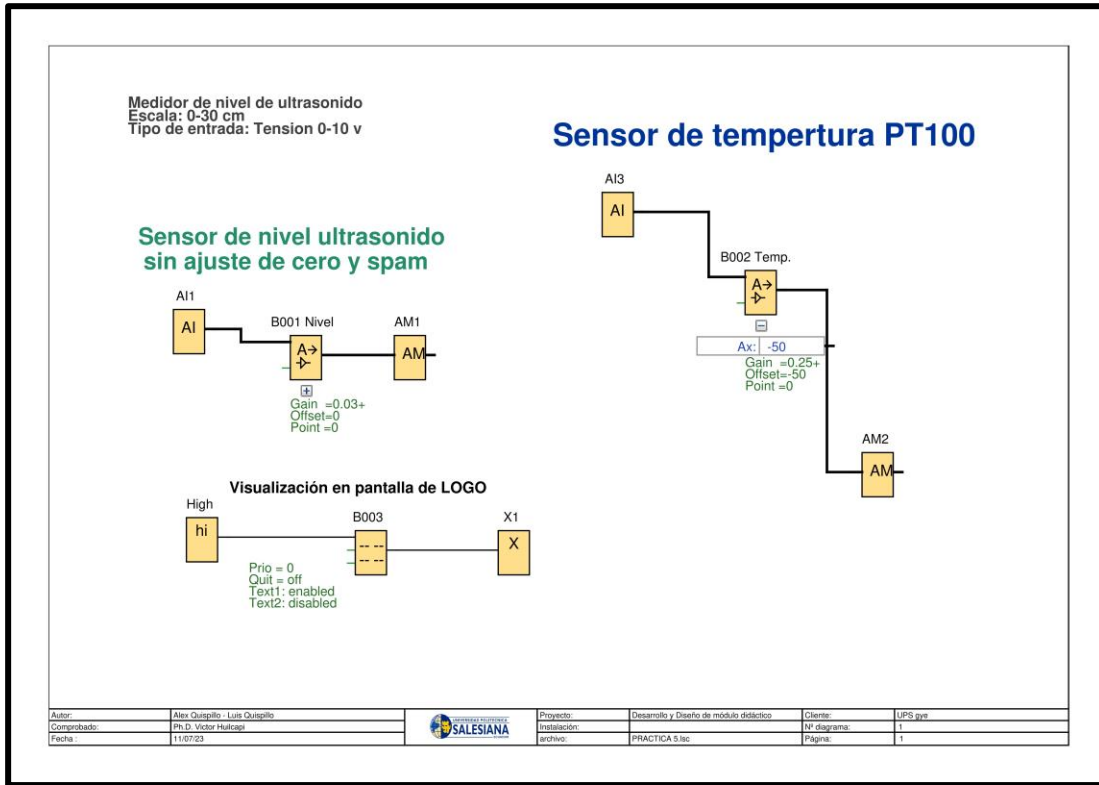
Figura 59. Comunicación entre LOGO! 8 y Raspberry Pi 4



- **Programación en LOGO! Soft Comfort de la práctica 5**

En el desarrollo de la programación se utilizaron compuertas lógicas como; amplificador analógico, control (PWM), salida, marcas, texto de aviso para la visualización en la pantalla de LOGO 8 como se muestra en la figura 60.

Figura 60. Programación en LOGO Soft Comfort práctica 5



- **Instalación de los complementos en el editor de flujo**

Como se observa en la tabla 4 se encuentran los complementos a utilizar en la práctica 5 para el desarrollo de los flujos en el entorno de trabajo en node-RED.

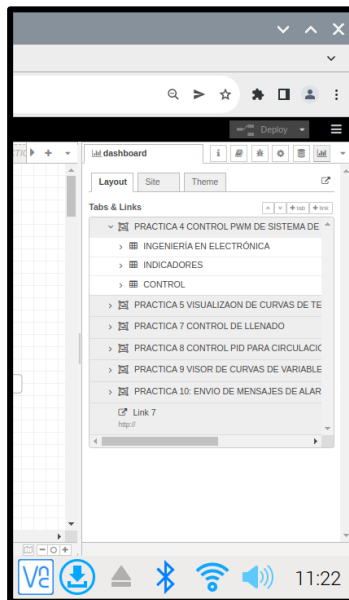
Tabla 4. Complementos de práctica 5

node-red	node-red-contrib-ui-led	node-red-dashboard
node-red-contrib-s7	node-red-contrib-ui-media	

- **Programación grafica en Node-RED sobre la práctica 5**

Una vez realizada la programación para lectura de sensores temperatura y nivel, se comienza con la creación de la tabla con tendrá el nombre *Práctica 5- visualización de curvas de temperatura y nivel* allí se crea los tres grupos donde agrupan los nodos de dashboard para darle forma a la interfaz, como se ilustra en la figura 61.

Figura 61. grupo práctica 5



- **Creación de flujo en workspace de Node-RED práctica 5**

Dentro del entorno de trabajo lo primero es crear un flujo (pestaña) para empezar a desarrollar la programación visual por flujos. Como se observa en la *tabla 5* se adjunta la configuración de los nodos utilizados como *s7 in*, *s7 out*, *chart*, *text*, *media*, *comment*, *change*, *switch*, *led*, *button*, *template*.

Cabe mencionar que cada nodo comprende una configuración distinta como se indica en la gráfica, sino se detalla otras opciones que se presenta en algunos nodos es porque no se utilizaron esas acciones o se utilizaron en prácticas anteriores.

Tabla 5. Configuración de los nodos utilizados en la práctica 5

SENSOR DE NIVEL		ACCIONAMIENTO MANUAL DE VALVULA Y BOMBA		LENGUAJE DE PROGRAMACIÓN JSON
<p><b>AM1 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable AM1</p>	<p><b>Nivel/tiempo - cuadro (chart)</b> <b>Group:</b> Type: line chart X-axis: last 30 /second X-axis-Label: HH:mm:ss Y-axis: min 0 max 30</p>	<p><b>I03 high freq (s7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable <b>High Freq</b></p>	<p><b>PARO DE EMERGENCIA (button)</b> Group: Nombre del grupo Label: PARO DE EMERGENCIA BackgroundD: RED Payload: false &lt;/&gt;Class: boton</p>	<p><b>Nodo template</b></p> <pre>.estado {     font-size: 0.6cm;     color: white !important; } .nr-dashboard-text.estado {     background-color: #1A237E; } .boton button {     border: none;     border-radius: 20px;     text-align: center;     font-size: 0.65cm; } .boton button:hover {     /*width: 50%;*/     background-color: #F57C00;     color: black !important; } .nr-dashboard-text.otros {     border.width: 4px;     border-color: #3E2723;     border-style: solid; } .nr-dashboard-text.encabezado {     background-color: #3E2723;     color: white !important; }</pre>
<p><b>Valor medido (node text)</b> Group: Nombre del grupo Label: Valor medido: Layout: center &lt;/&gt;Class: bordes</p>		<p><b>Válvula (switch)</b> Group: Nombre del grupo Label: VÁLVULA Background RED On Payload: true Off Payload: false &lt;/&gt;Class: pulsar</p>	<p><b>Bomba (switch)</b> Group: Nombre del grupo Label: BOMBA DE AGUA On Payload: true Off Payload: false &lt;/&gt;Class: pulsar</p>	
<p><b>LECTURA DE RTD – PT 100</b></p>		<p><b>ACCIONAMIENTO MANUAL DE VÁLVULA Y BOMBA</b></p>		
<p><b>AM2 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable AM2</p>	<p><b>Temperatura/tiempo - cuadro (chart)</b> <b>Group:</b> Type: line chart X-axis: last 1 /minute X-axis-Label: HH:mm:ss Y-axis: min 0 max 60</p>	<p><b>Q04 - 2 rules (change)</b> Change: msg payload Search for: true Replace with true Change: msg payload Search for: false Replace with: false</p>	<p><b>Resistencia - rules (change msg.payload)</b> Change: msg payload Search for: true Replace with false</p>	
<p><b>Estado (text)</b> Group: Nombre del grupo Label: Estado Layout: center &lt;/&gt;Class: estado</p>		<p><b>Bomba de Agua - 2 rules (change)</b> Change: msg payload Search for: false Replace with true Change: msg payload Search for: false Replace with: false</p>	<p><b>Q01 - BOMBA</b></p> <p><b>Q02 - VÁLVULA</b></p> <p><b>Q04 - RESISTENCIA</b></p>	

Realizada la programación visual en el editor de flujo Node-RED, tomar en cuenta que cada nodo puesto en el editor tiene una finalidad de conexión con otro nodo, por ende, un nodo no puede estar sin conectar para evitar error al compilar.

En la figura 62 y 63 se adjunta la programación en cómo se quedó detallado referente a esta práctica con sus respectivos comentarios, de esta manera se da orden a la programación y tener una dirección al momento de realizar alguna reestructuración.

Figura 62. programación visual en Node-RED sobre la práctica 5 parte 1

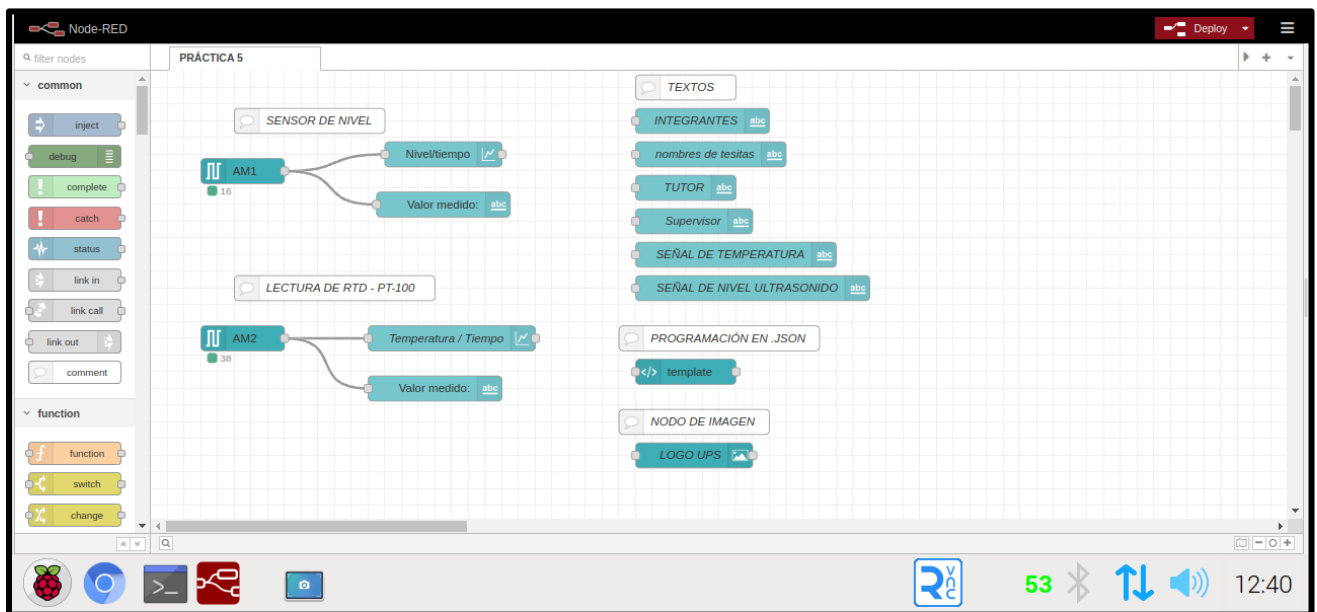
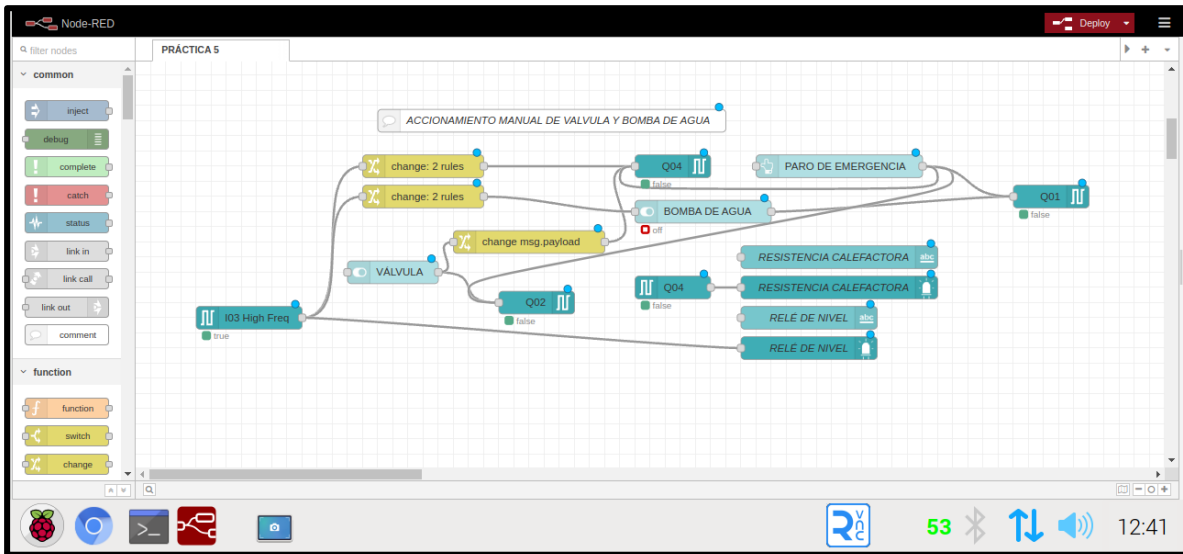




Figura 63. programación visual en Node-RED sobre la práctica 5 parte 2




Finalizado, como se muestra en la *figura 64* el desarrollo del dashboard para esta práctica 5, además se dio unos ajustes de personalizado en nodo *“button”* y *“switch”* con el nodo *“template”*.

Figura 64. Dashboard práctica 5



#### 4.6 Práctica #6

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA #6	TÍTULO DE LA PRÁCTICA Medición de nivel por ultrasónico del sistema de llenado.
<b>OBJETIVOS</b> <ul style="list-style-type: none"><li>• Interpretar medición de nivel por ultrasonido y definir porcentaje de error con una medición patrón.</li><li>• Entender la funcionalidad de programación en LOGO! Soft Comfort en diagrama FUP para su respectiva simulación.</li></ul>	
<b>PLANTEAMIENTO DE PROBLEMAS</b>	
<ul style="list-style-type: none"><li>• Conocer las características de medición del instrumento de medición de nivel por ultrasonido. Entre ellas: alcance o spam, repetibilidad, relación de ajuste, zona muerta, precisión, exactitud, histéresis y factores que generan error en la medición.</li><li>• Desarrollar el diagrama de funciones FUP en LOGO! Soft Comfort para el ajuste de medición cero y spam del sensor ultrasonido.</li><li>• Visualizar en la pantalla LOGO!8 para comparación de valor real y su respectivo ajuste en el uso de las funciones aritméticas.</li></ul>	

- **Resolución práctica 6**

Esta práctica consiste en el *ajuste de medición* del sensor ultrasonido utilizado, ya que en la práctica 5 se observó el desajuste del sensor de nivel en su medición correspondiente al fluido.

- **Programación en LOGO Soft Comfort V8.3 de la práctica 6**

- **Ajuste de Cero y Spam**

Conectar el sensor de nivel a la entrada I7, la cual debe estar configurada como analógica AI1, tomando en cuenta que los valores de lectura que toma el LOGO! 8 es 0-1000, por lo tanto, se conecta con una compuerta *amplificador analógico* para la conversión a un rango de 0-40 en seguida con una marca analógica para la salida del valor, figura 65 se puede observar desde la pantalla de simulación los valores enviados, figura 66.

Figura 65. Ajuste de valor analógico

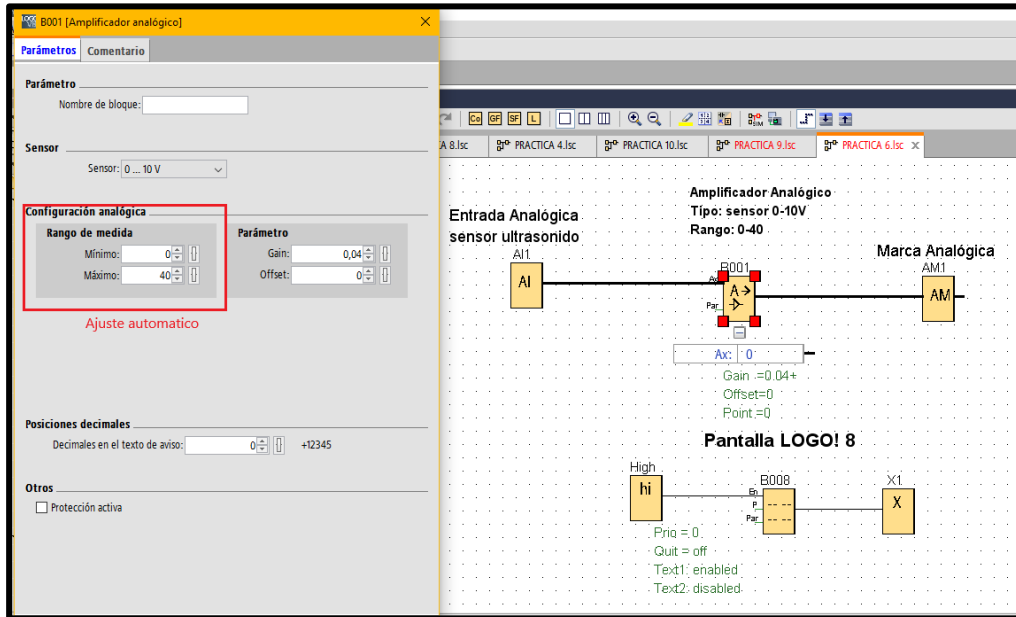


Figura 66. Texto de aviso en pantalla LOGO! 8



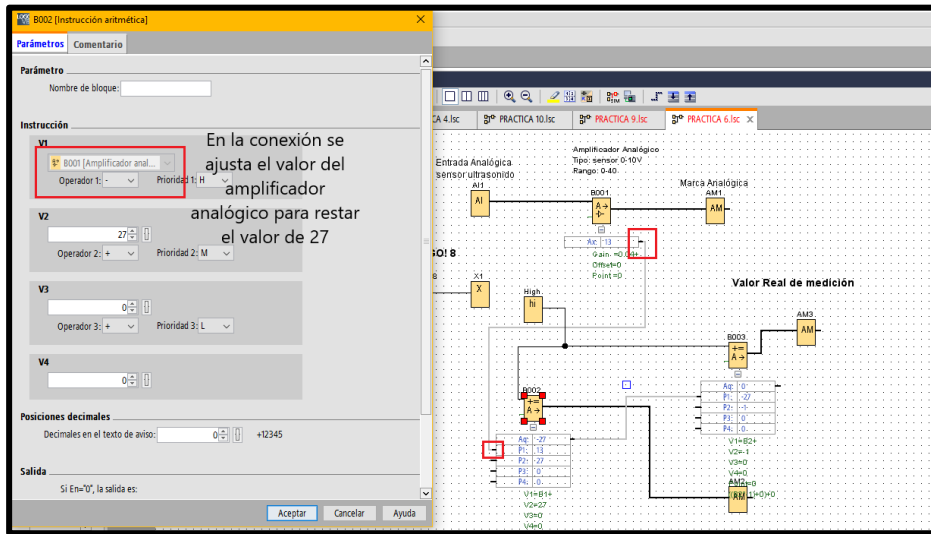
Segundo, Verificar el valor que se recibe en el canal analógico con el tanque vacío y restarlo, tomando en cuenta que el sensor no debe tener un acercamiento nomás 5 cm entre el fluido, para evitar error en la medición. En este caso la medición en el tanque vacío es de 27 cm, figura 67.

Figura 67. Medición del tanque vacío



Por último, El valor resultado de la medición se utiliza para restar con el valor de amplificador analógico, figura 68, para luego, multiplicar (-1) para resultado de un valor positivo en la medición.

Figura 68. Ajuste de instrucción aritmética



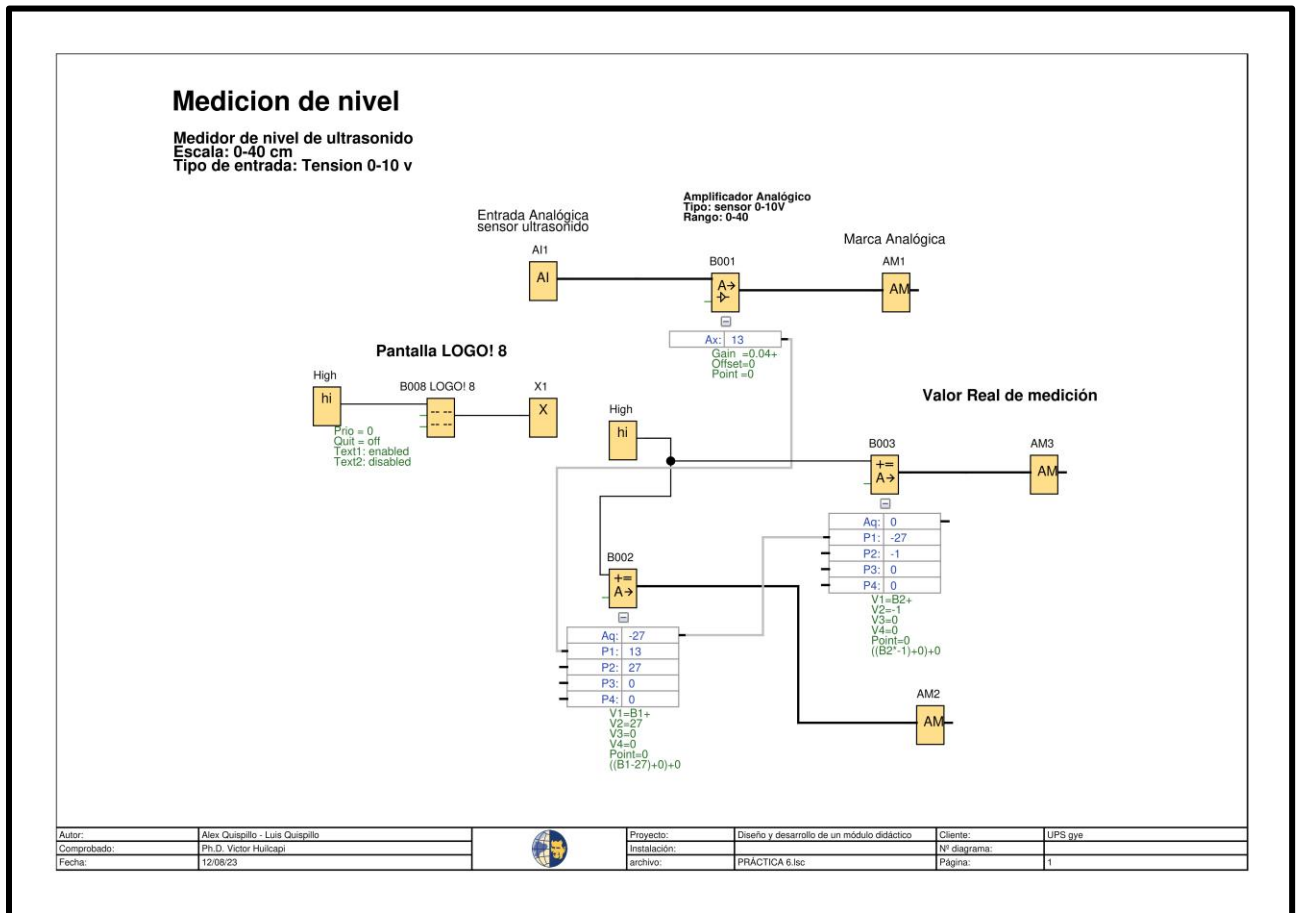
Como resultado final, desde la pantalla LOGO! 8 se realizó la respectiva comparación entre el valor real y valor ajustado con las funciones aritméticas, figura 69, se observa un pequeño valor de alcance cero, se puede decir que existe un porcentaje mínimo de error en la medición del sensor, justamente detallado en el *datasheet* del sensor.

Figura 69. Pantalla LOGO! 8



Este caso se adjunta la programación completa en la figura 70 de lo que es el ajuste del sensor ultrasonido realizado paso a paso en las instrucciones anteriores, tomar en cuenta las recomendaciones con respecto al ajuste, de todas forma esta práctica consiste para tener una noción del funcionamiento de este dispositivo muy usado en las industrias para medición de fluidos u otros materiales.

Figura 70. Programación en LOGO1 Soft Comfort práctica 6



#### 4.7 Práctica #7

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA <b>#7</b>	TÍTULO DE LA PRÁCTICA Control de llenado desde IoT NODE RED.
OBJETIVOS <ul style="list-style-type: none"><li>• Realizar control de bomba de agua para llenado automático de tanque con valores de consigna fijos, utilizando el medidor de nivel por ultrasonido como retroalimentación.</li><li>• Orientar al estudiante en el enfoque en el aprendizaje del editor de flujo Node-RED para aplicaciones IoT.</li><li>• Entender la funcionalidad de programación basada por motor de flujo por Node-RED y la programación en LOGO! 8 en diagrama FUP para su respectiva simulación.</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Establecer consigna de nivel máximo y mínimo de llenado en el tanque inferior.</li><li>• Desarrollar la interfaz gráfica en Node-RED para visualización de los niveles en los tanques y el estado funcional de los actuadores.</li><li>• Uso de las luces de piloto en la interfaz para ON/OFF del proceso, bomba de agua.</li><li>• Establecer un nodo switch para uso de paro de emergencia.</li><li>• Crear botón de descarga manual del fluido del tanque superior.</li><li>• Crear botón de encendido y apagado del proceso.</li><li>• Visualización de nivel por nodo dashboard Chart.</li></ul>	

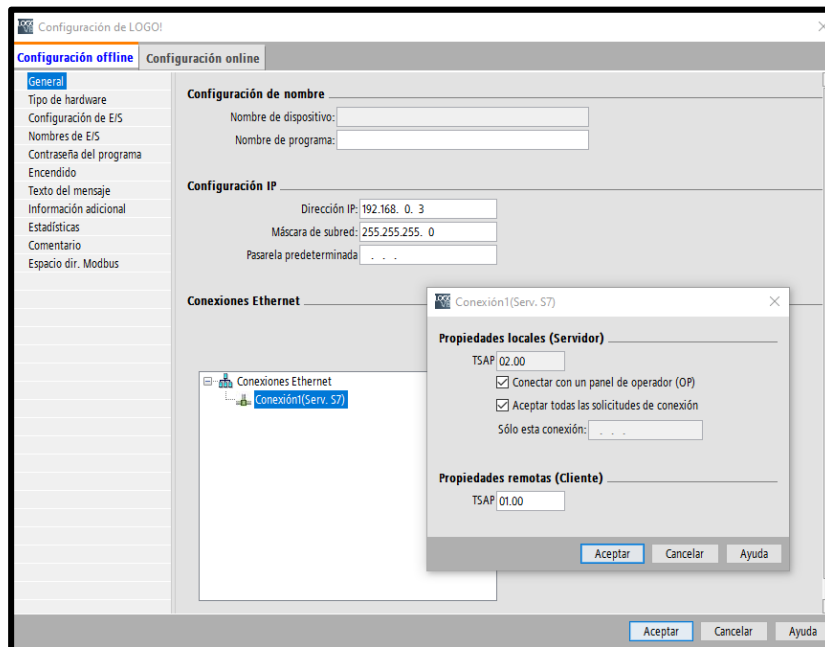
- El inicio del proceso comienza con la descarga manual del fluido por consigna de nivel máximo.
- Condición que el relé de nivel envíe una señal de 0 para el encendido del proceso, es decir, activa la bomba para el llenado de tanque superior, por lo tanto, el sensor ultrasonido debe ajustarse a un valor en el que desactive el relé de nivel por medio de la válvula.
- La bomba realiza una comparación de que si es  $\geq 5$  se activa el proceso, si es  $\leq 5$  el proceso se desactiva automáticamente desde las marcas en LOGO!
- El proceso finaliza al cumplir en la recirculación del fluido con el tanque superior lleno y el tanque inferior con un valor entre 5 a 6 cm de caudal por seguridad de la bomba y con válvula cerrada.
- Uso del nodo template para personalizar los widgets del dashboard.



- **Resolución práctica 7**

Para el desarrollo de esta práctica, se sugiere que se empiece con la configuración IP para compilar de manera inalámbrica por medio de la Red que nos proporciona el router, y luego con la comunicación S7 que nos proporciona con Node-RED por TSAP, como se ilustra en la figura 71.

Figura 71. Comunicación entre LOGO 8 y Raspberry Pi 4

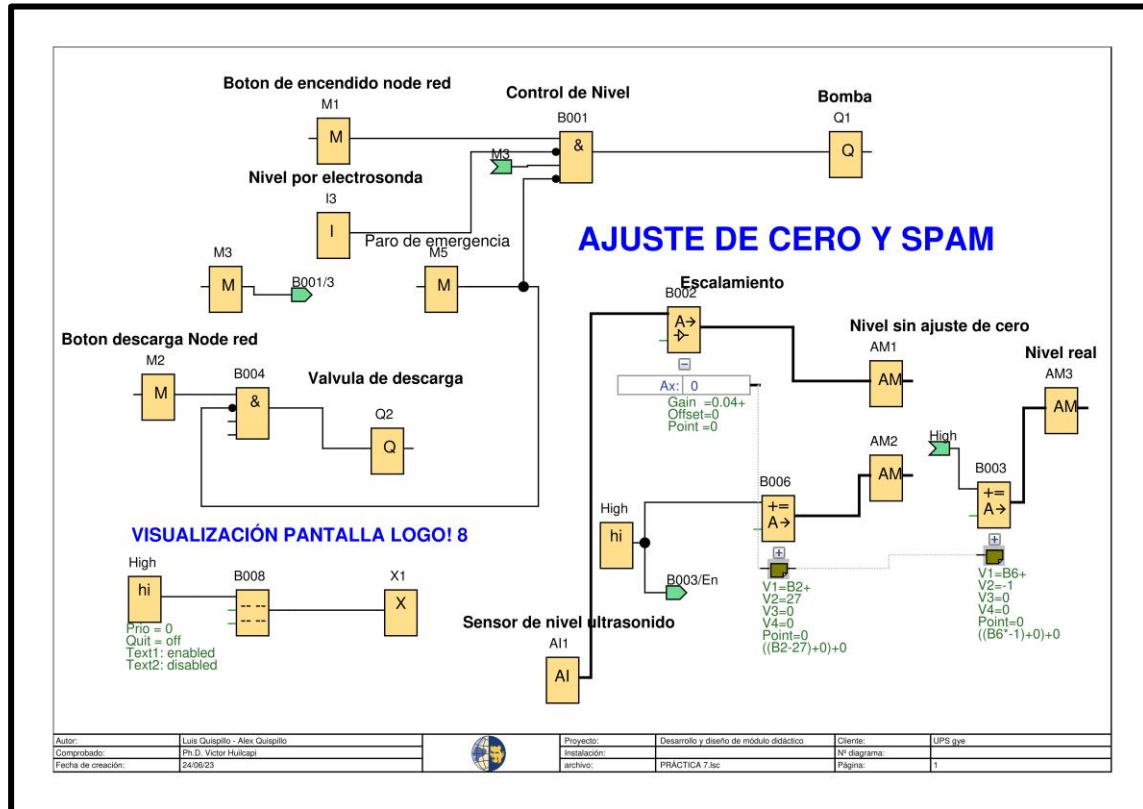


Fuente: Autores

- **Programación en LOGO Soft Comfort V8.3 de la práctica 7**

En el desarrollo de la programación se utilizaron algunos bloques como: amplificador analógico, conmutador analógico de valor umbral diferencial entrada, salida, marcas, marca analógica, como se ilustra en la figura 72.

Figura 72. Programación de la práctica 7 en LOGO Soft Comfort V8.3



- **Instalación de los complementos en el editor de flujo**

Como se observa en la *tabla 6* se encuentran los complementos a utilizar en la práctica 7 para el desarrollo de los flujos en el entorno de trabajo en Node-RED.

Tabla 6. Complementos de la práctica 7

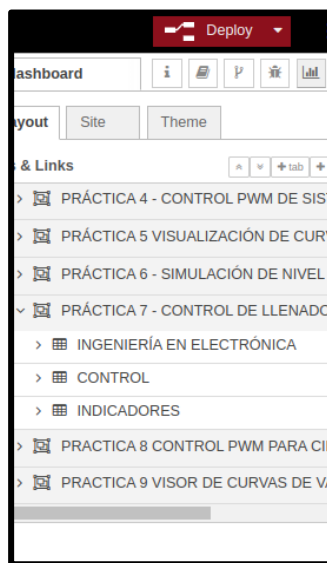
node-red	node-red-contrib-ui-led	node-red-dashboard
node-red-contrib-s7	node-red-contrib-ui-media	

- **Creación de grupos en dashboard**

En primer lugar, se crea la tabla, para eso se dirige en el parte superior derecho encontrará los iconos *Dashboard >Layout >+tab*.

Segundo, dentro de tabla que se define con el nombre *Práctica 7- Control de llenado desde IoT Node-RED*, se crean los grupos que están dentro de la tabla, de manera que, los nombres de los grupos ya creados, se define en la figura 73 para darle orden a los widgets de la interfaz gráfica.

Figura 73. Grupo práctica 7



- **Creación de flujo en workspace de Node-RED práctica 7**

Dentro del entorno de trabajo lo primero es crear un flujo (pestaña) para empezar a desarrollar la programación visual por flujos. Como se observa en la tabla 7 se adjunta la configuración de los nodos utilizados como: *s7 in, s7 out, chart, text, media, comment, change, switch, led, button, template*.

Cabe mencionar que cada nodo comprende una configuración distinta como se indica en la tabla, sino detalla otras opciones que se presenta en algunos nodos es porque no se utilizaron esas acciones o fueron utilizadas en prácticas anteriores.

Tabla 7. Configuración de los nodos utilizados en la práctica 7

CONTROL DE VÁLVULA		BOTONERA		LENGUAJE DE PROGRAMACIÓN JSON
<p><b>AM3 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable AM1</p>	<p><b>Q02 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable Q02</p>	<p><b>ON (button)</b> Group: Nombre del grupo Size: 3x2 Icon: fa-check Label: Encendido Background: green Payload: true &lt;/&gt;Class: pulsar</p>	<p><b>M01 (S7 in)</b> PLC: S7-LOGO8 Variable: M01 <b>DESACARGAR (button)</b> Group: Nombre del grupo Size: auto Label: Descargar Background: Green Payload: true &lt;/&gt;Class: pulsar</p>	<p><b>Nodo template</b> .pulsar button{ background-color: #D50000; border-radius: 15px; color: White !important; font-size: 0.5cm; text-align: center; } .pulsar_1{ background-color: #D50000; border-radius: 15px; color: White !important; font-size: 0.5cm; text-align: center; } .nr-dashboard-text.nivel_1{ background-color: #263238; font-size: 0.5cm; color: White !important; } .nr-dashboard-text.bordes{ border-width: 4px; border-color: #4E342E; font-size: 0.5cm; border-style: solid; } .nr-dashboard-text.encabezado{ background-color: #A1887F; } .nr-dashboard-switch.emergencia{ background-color: #D50000; font-size: 0.5cm; color: White !important; }</p>
<p><b>SWITCH</b> Property msg.payload &gt;= (number) 23</p>	<p><b>CHANGE</b> <b>Change: 2 rules</b> Change/ msg/ payload Search for: Boolean (true) Replace with: string/Válvula abierta  Change/ msg/ payload Search for: Boolean (false) Replace with: string/Válvula cerrada</p>	<p><b>OFF (button)</b> Group: Nombre del grupo Size: 6x2 Icon: fa-stop-circle Label: Apagado Background: red Payload: false &lt;/&gt;Class: pulsar</p>	<p><b>Set msg payload (Change)</b> Change: msg.payload Search for: true Replace with: false <b>(BOOLEAN)</b></p>	
<p><b>CHANGE</b> Set: msg.payload To the value: false <b>(BOOLEAN)</b></p>	<p><b>ESTADO DE VÁLVULA (text)</b> Group: nombre del grupo Value format: Layout: center &lt;/&gt;Class: encabezado Name: ESTADO DE VÁLVULA</p>	<p><b>NODOS DE TEXTOS</b></p>	<p><b>ENTRADA RELÉ DE NIVEL</b></p>	
<p><b>M02 (S7 out)</b> PLC: S7-LOGO8 Variable M02</p>		<p><b>Integrantes (text)</b> Group: nombre del grupo Value format: <b>INTEGRANTES</b> Layout: center &lt;/&gt;Class: encabezado</p>	<p><b>M01 (S7 in)</b> PLC: S7-LOGO8 Mode: <b>Single variable</b> Variable: <b>103 high freq</b></p>	
<p><b>CONTROL DE BOMBA</b></p>		<p><b>TUTOR (text)</b> Group: nombre del grupo Value format: <b>TUTOR</b> Layout: center &lt;/&gt;Class: encabezado</p>	<p><b>CHANGE</b> <b>Change: 2 rules</b> Change/ msg/ payload Search for: true (Boolean) Replace with: 1 Change/ msg/ payload Search for: false (Boolean) Replace with: 0</p>	
<p><b>AM3 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable AM3</p>	<p><b>SWITCH</b> Property msg.payload &lt;= (number) 5</p>		<p><b>Nodo Chard</b> Group: nombre del grupo Size: 4x7 Type: Bar chart Y-axis: min 0 max 1 Name: Tanque superior</p>	
<p><b>SWITCH</b> Property msg.payload &gt;= (number) 5</p>	<p><b>Set msg payload (CHANGE)</b> Set: msg.payload To the value: false <b>(BOOLEAN)</b></p>	<p><b>TANQUE INFERIOR (text)</b> Group: nombre del grupo Value format: TANQUE INFERIOR Layout: center &lt;/&gt;Class: encabezado</p>		
<p><b>Set msg payload (CHANGE)</b> Set: msg.payload To the value: true <b>(BOOLEAN)</b></p>	<p><b>M01 (S7 in)</b> PLC: S7-LOGO8 Variable: M01</p>	<p><b>SEÑAL DE BOMA -LED</b> Group: nombre del grupo Size: 3x1 Value format: BOMBA Layout: center &lt;/&gt;Class: encabezado Name: SEÑAL DE BOMA -LED</p>	<p><b>CHANGE</b> <b>Change: 2 rules</b> Change/ msg/ payload Search for: true (Boolean) Replace with: Lleno (string) Change/ msg/ payload Search for: false (Boolean) Replace with: vacio (string)</p>	
<p><b>AM3 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable AM3</p>		<p><b>LOGO UPS configuration</b> Group: nombre del grupo Suze: 6x6 Layout: center <b>Files</b> Category: Mis imagenes File: Salesiana.jpg</p>	<p><b>NIVEL (text)</b> Group: nombre del grupo Size 4x1 Label: NIVEL Layout: center &lt;/&gt;Class: nivel_1</p>	
<p><b>PARO DE EMERGENCIA</b></p>				
<p><b>Set msg.payload (node switch)</b> Group: nombre del grupo Size: auto Label: PARO DE EMERGENCIA On Payload: True Off Payload: False Topic msg topic &lt;/&gt;Class: emergencia</p>	<p><b>Set msg payload (node change)</b> Set: msg.payload To the value: false <b>(BOOLEAN)</b></p>			
	<p><b>M04 (S7 out)</b> PLC: S7-LOGO8 Variable: M04</p>	<p><b>M05 (S7 out)</b> PLC: S7-LOGO8 Variable: M05</p>		
	<p><b>M02 (S7 out)</b> PLC: S7-LOGO8 Variable: M02</p>	<p><b>M01 (S7 out)</b> PLC: S7-LOGO8 Variable: M01</p>		

Recordar que, cada nodo del complemento *node-red-dashboard* debe indicar en que grupo van a estar distribuidos ya que los *widets* en la interfaz se ordenan de manera automática al no darle indicación de tamaño, lo ideal es personalizar, lo cual se debe dirigir *Layout* en la opción *dashboard*.

En la figura 74, 75 y 76 se adjunta la programación visual detallada referente a esta práctica con sus respectivos comentarios, de esta manera se da orden a la programación y tener una guía al momento de realizar alguna reestructuración.

Figura 74. Entorno de trabajo de Node-RED de la práctica 7

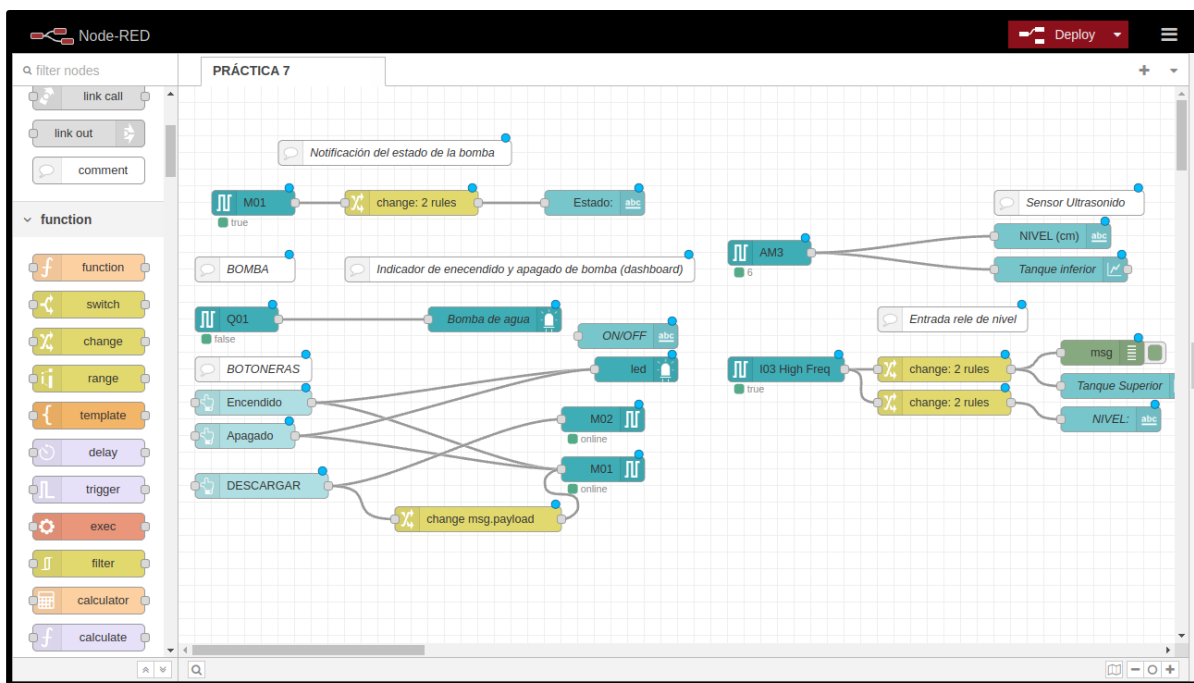


Figura 75. Entorno de trabajo de Node-RED de la práctica 7

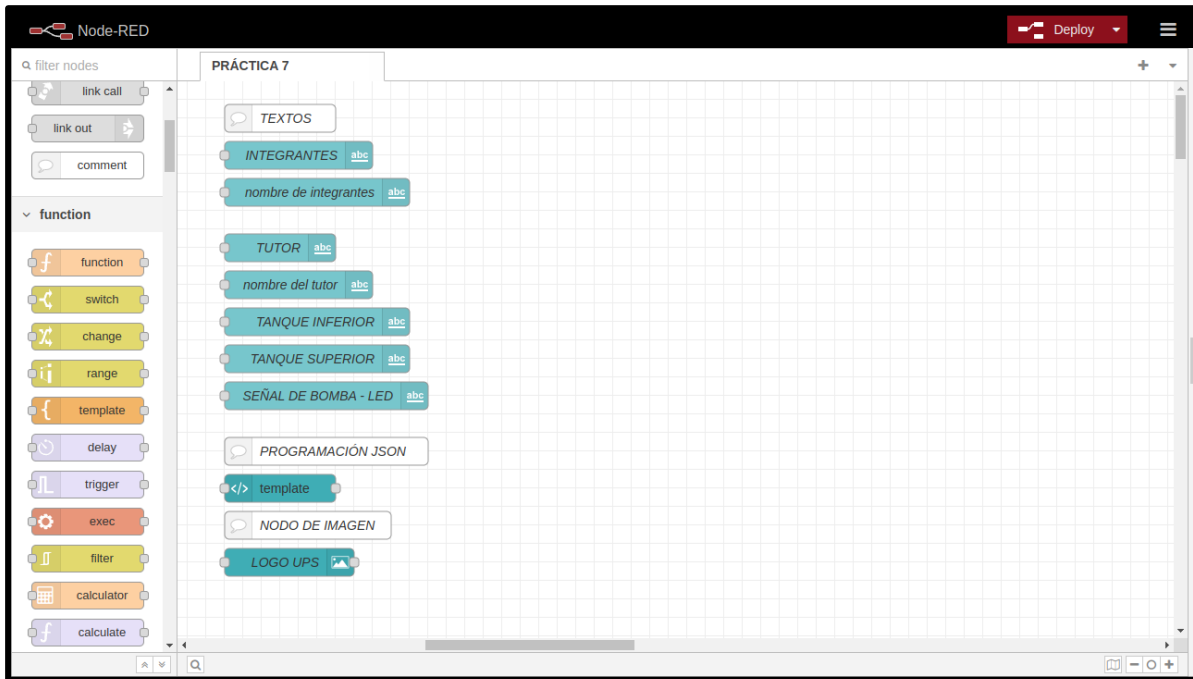
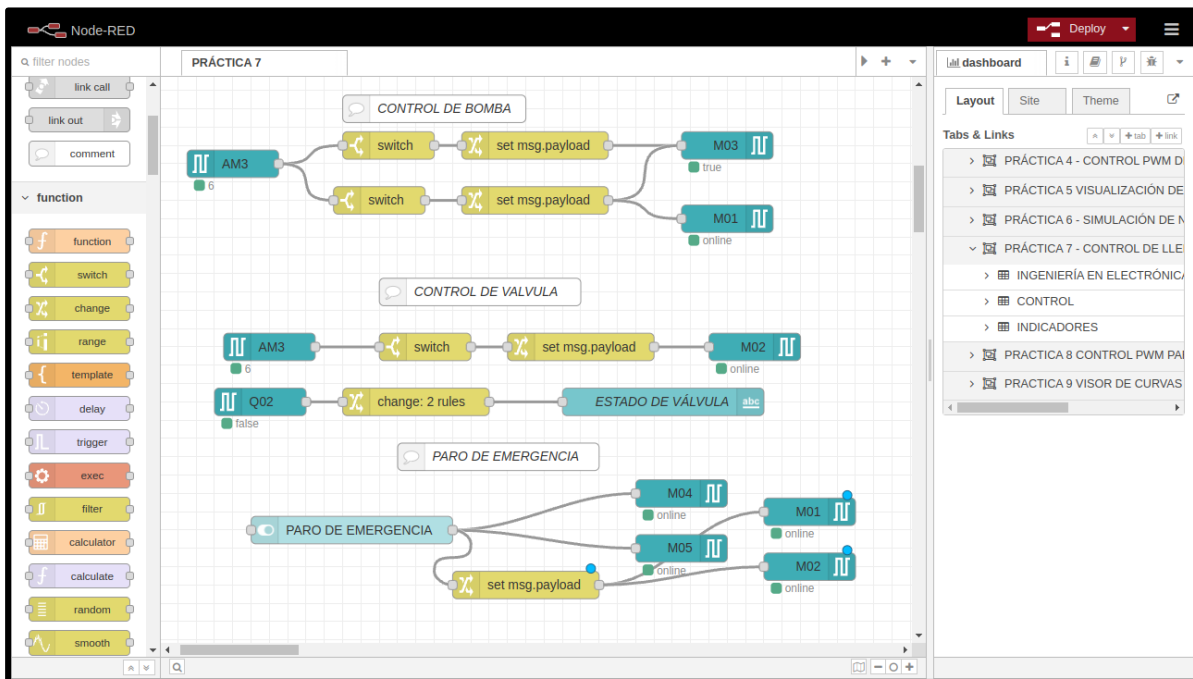
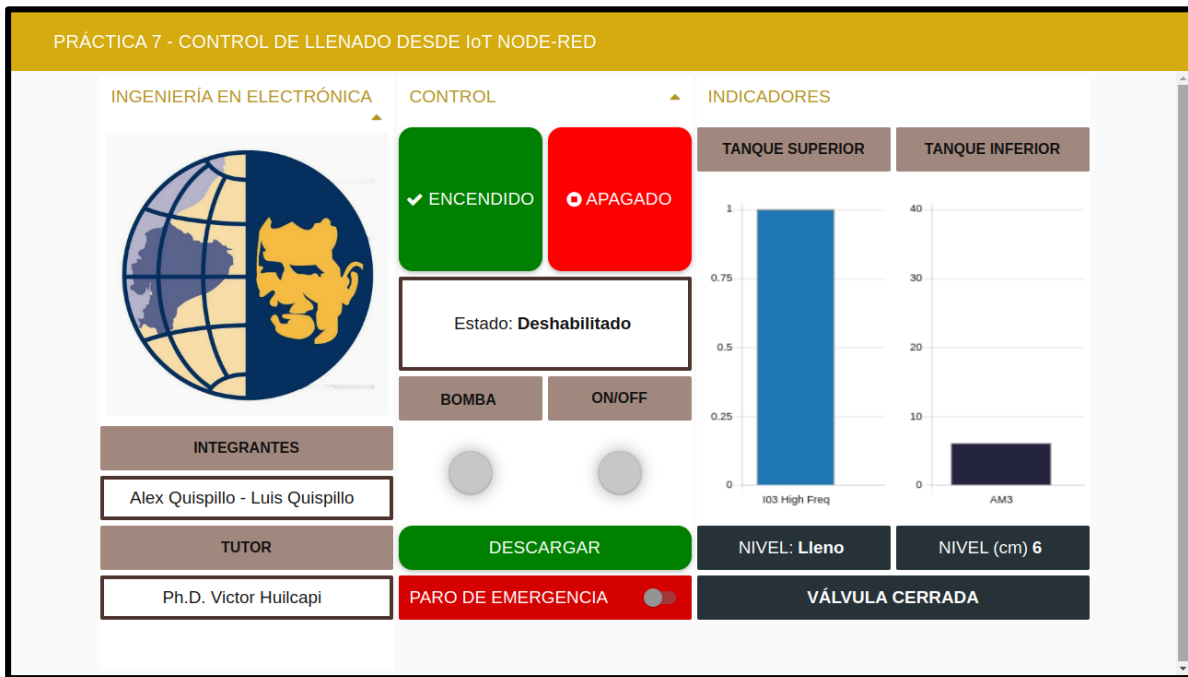


Figura 76. Entorno de trabajo de Node-RED de la práctica 7



Finalizado, como se muestra en la *figura 77* el desarrollo del dashboard para esta práctica 5, además se dio unos ajustes de personalizado en nodo “*button*” y “*switch*” con el nodo “*template*”.

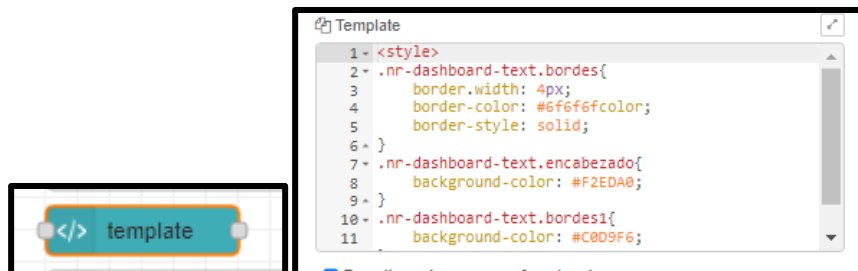
Figura 77. Dashboard de la práctica 7




- **Personalización de Widgets**

Por medio del nodo *Template* se personaliza los widgets de la interfaz. En esta práctica se utiliza lenguaje JavaScript para personalizar el color de bordes y fondo de ciertos nodos de texto, tamaño de la fuente, esquinas de widgets como se ilustra en la Figura 78.

Figura 78. Widgets personalizados práctica 7



## 4.8 Práctica #8

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA <b>#8</b>	TÍTULO DE LA PRÁCTICA Control de PWM desde Node-RED para circulación de agua caliente.
OBJETIVOS	
<ul style="list-style-type: none"><li>• Establecer la comunicación LOGO! 8 y Raspberry Pi 4.</li><li>• Diseñar un control de proceso aplicando programación FUP y programación Visual.</li><li>• Generar interacción del usuario con el dashboard creado desde editor de flujo Node-RED.</li><li>• Orientar al estudiante en el enfoque en el aprendizaje del editor de flujo Node-RED para aplicaciones IoT.</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Inicio del proceso comienza con la pulsación de botón encender, abre la válvula solenoide y drena el fluido desde el taque superior al tanque inferior por gravedad, comienza el sensor ultrasonido la medición de nivel que llena en el tanque 2, el máximo de llenado que es de 27 cm pero, lo recomendable para no tener un error de lectura en el exceso de líquido lo ideal es dejar en 26 cm en la medición, la válvula se cierra por un bloque temporizador de retardo a la desconexión, por lo tanto, se debe calcular el tiempo de llenado en el tanque hasta el nivel adecuado sin paro del proceso.</li></ul>	



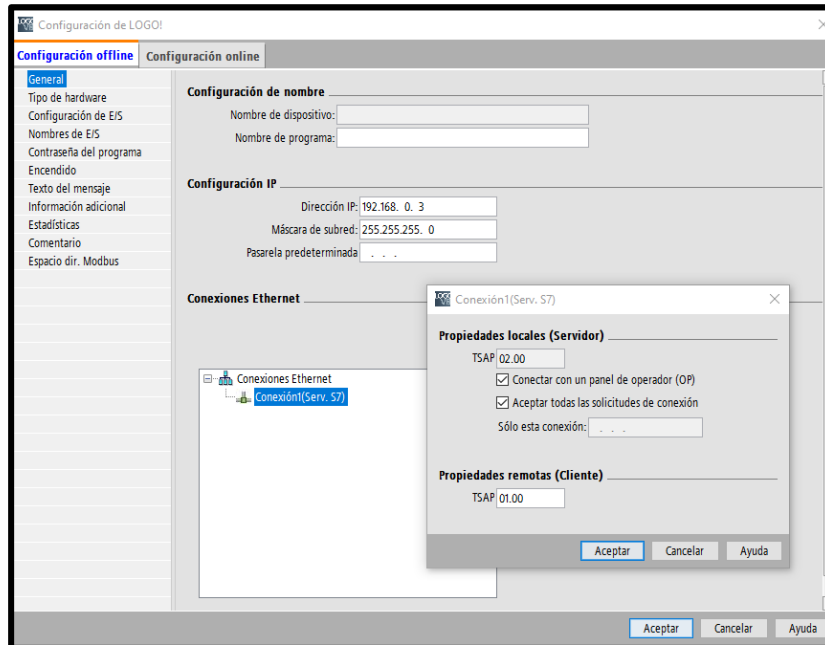
Se activa la bomba por un relé en una de las salidas de LOGO! 8 para recirculación del agua en el tanque 1, se desactiva la bomba al enviar una señal booleana en la entrada del LOGO1 8 por el relé de nivel, en seguida un relé activa la resistencia calefactora para el control de temperatura por PWM antes ya configurada en su set máximo y mínimo.

- Realizar la respectiva programación visual en el editor de flujo para el control y visualización por Node-RED.
- Diseño del dashboard para visualización de: llenado del tanque1 y 2, temperatura actual, resistencia calefactora, curva de temperatura, estados de bomba y válvula.
- Uso de las luces de piloto en el dashboard para visualizar el encendido de la resistencia calefactora.
- Crear texto de aviso en pantalla de LOGO! como: temperatura, bomba, válvula, resistencia, cada uno con la señalética de ON y OFF.
- Hacer uso del nodo *template* para la personalización de los widgets.

- **Resolución práctica 8**

Para el desarrollo de esta práctica, se sugiere que se empiece con la configuración IP para compilar de manera inalámbrica por medio de la Red que nos proporciona el router, y luego con la comunicación S7 que nos proporciona con Node-RED por TSAP, como se ilustra en la figura 79.

Figura 79. Comunicación entre LOGO! 8 y Raspberry Pi 4 práctica 8



- **Programación FUP en LOGO! Soft Comfort práctica 8**

En el desarrollo de la programación se utilizaron compuertas lógicas como; entradas analógicas, marca analógica, estado alto, amplificador analógico, marca analógica, entrada, salida, marcas, conector abierto, texto de aviso que se visualiza en la pantalla de LOGO! 8, como se muestra en la figura 80.

Además, como en las prácticas anteriores se tuvo que hacer el ajuste de medición (Cero y Span) de la señal analógica del sensor ultrasonido mediante programación FUP, figura 81.

Figura 80. Programación en FUP de la práctica 8 - 1/2

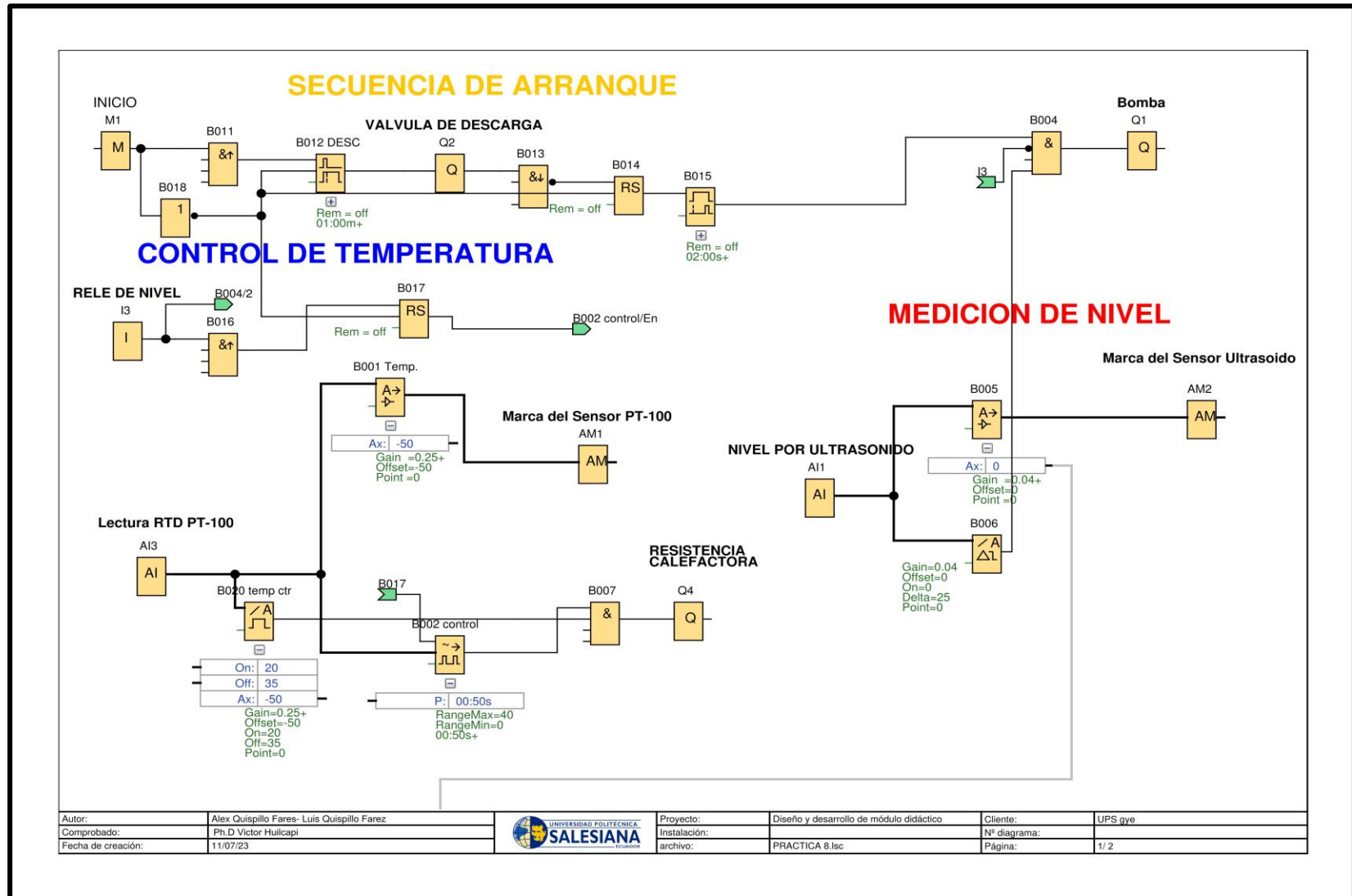
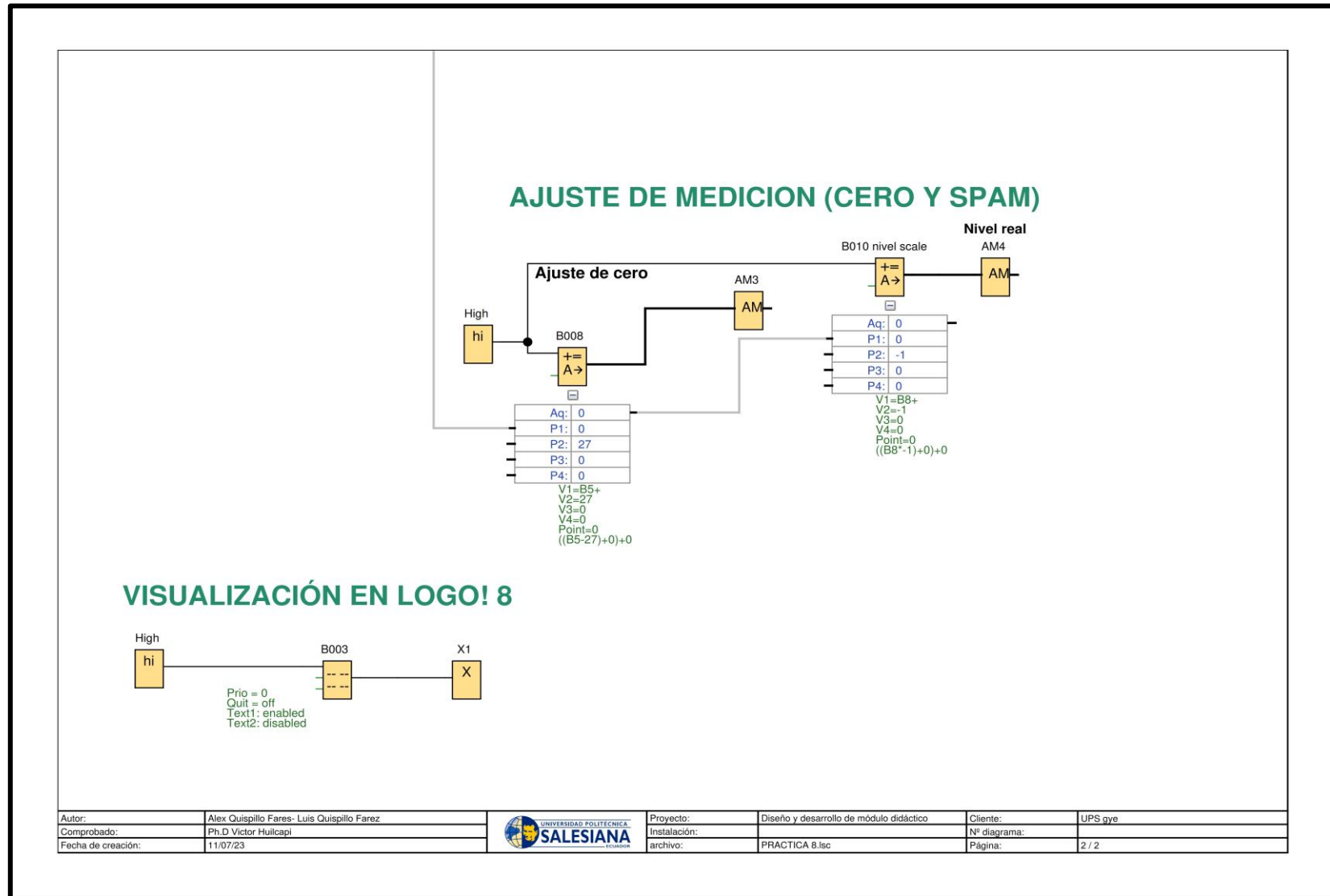


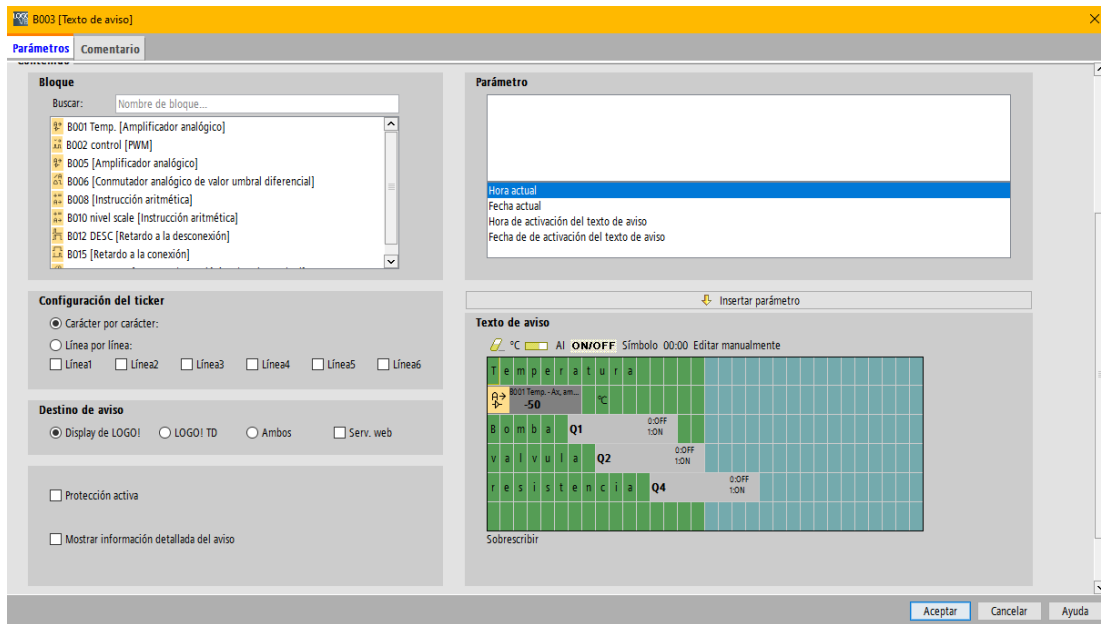
Figura 81. Programación en FUP práctica 8- 2/2



- **Creación de texto aviso en LOGO! 8**

Desde el bloque B003 de texto de aviso se puede agregar y editar manualmente. Los diferentes tipos de bloques se puede encontrar en la parte izquierda, como se ilustra en la figura 82.

Figura 82. Texto de aviso práctica 8



- **Instalación de los complementos en el editor de flujo**

Como se observa en la *tabla 8* se encuentran los complementos a utilizar en la práctica 8 para el desarrollo de los flujos en el entorno de trabajo de Node-RED.

Tabla 8. Complementos de práctica 8

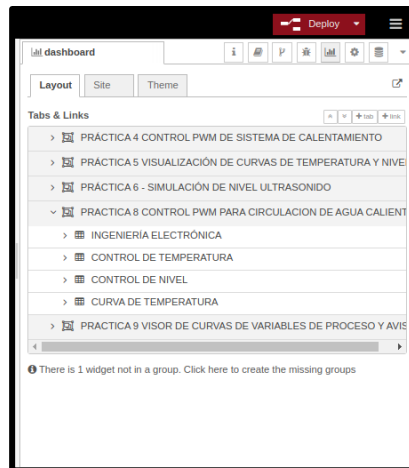
node-red	node-red-contrib-ui-led	node-red-dashboard
node-red-contrib-s7	node-red-contrib-ui-media	

- **Creación de grupos y visualización en dashboard**

Para el desarrollo del dashboard, lo primero es definir los tres grupos donde cada *node-dashboard* estarán agrupado, como se ilustra en la figura 83, para el orden de la interfaz, la cual se define.

- **Ingeniería en electrónica:** nombres de los practicaste y el logo de la UPS.
- **control de temperatura:** accionamientos del proceso, visualización de curva de temperatura, control PWM de la resistencia.
- **Control de nivel:** texto de aviso en el encendido de la bomba y válvula, por último, el widget de llenado del tanque 2.
- 

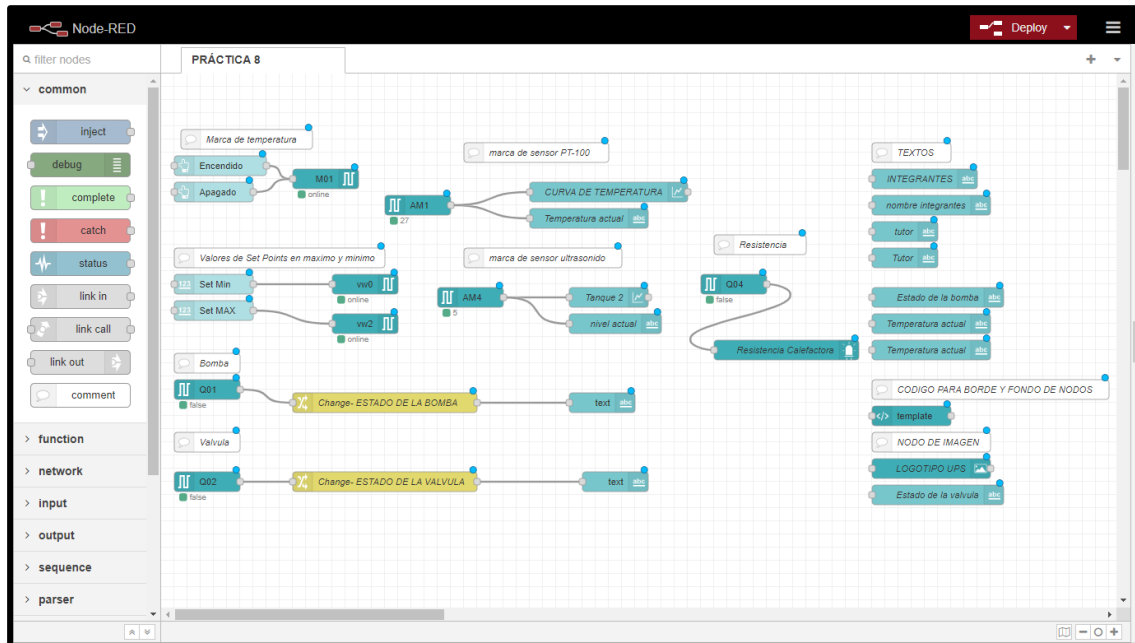
Figura 83. Grupo en dashboard práctica 8



- **Programación grafica en Node-RED sobre la práctica 8**

Realizada la programación en LOGO! Soft Comfort, se comienza con la programación visual en Node-RED como se ilustra en la figura 84 los cuales se utilizaron los nodos: *Change, text, media, s7 in, template, comment*.

Figura 84. Programación grafica de la práctica 8




Por último, se configura los nodos de dashboard que serán agrupados en cada grupo que fue creado para su orden, es así como los widgets se mostrarían en la interfaz gráfica, *figura 85*.

Figura 85. Dashboard de la práctica 8



#### 4.9 Práctica #9

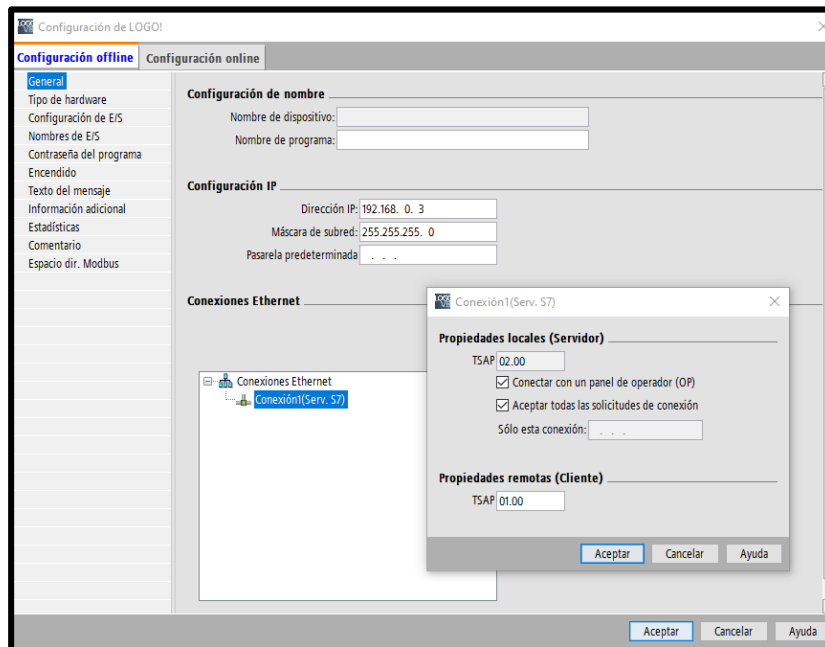
	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA #9	TÍTULO DE LA PRÁCTICA Visor de curvas de variables de procesos y generación de avisos de funcionamiento del sistema.
OBJETIVOS <ul style="list-style-type: none"><li>• Utilización en el editor de flujos los diferentes nodos: switch, range, change para generar avisos funcionales y avisos de alarma.</li></ul>	
PLANTEAMIENTO DEL PROBLEMA	
<ul style="list-style-type: none"><li>• Crear accionamientos manuales de los actuadores para poder simular errores del sistema en control de nivel y control de temperatura.</li><li>• Crear cuadros numéricos de entrada para definir límites máximos y mínimos de variables de procesos.</li><li>• Utilización de funciones en Node-RED y nodos de comparación y poder generar diferentes tipos de avisos: texto, led, gauge.</li><li>• El desarrollo de la programación en esta práctica será mayor parte en Node-RED, que en LOGO! Soft Comfort V8.3 para aprovechar los recursos de cada uno de los diferentes nodos.</li><li>• Crear acceso remoto y visualización de notificaciones en aplicación móvil.</li></ul>	



- **Resolución práctica 9**

Para el desarrollo de esta práctica, se sugiere que se empiece con la configuración IP para compilar de manera inalámbrica por medio de la Red que nos proporciona el router, y luego con la comunicación S7 que nos proporciona con Node-RED por TSAP, desde luego la IP y numero de puerto también forman parte, como se ilustra en la figura 86.

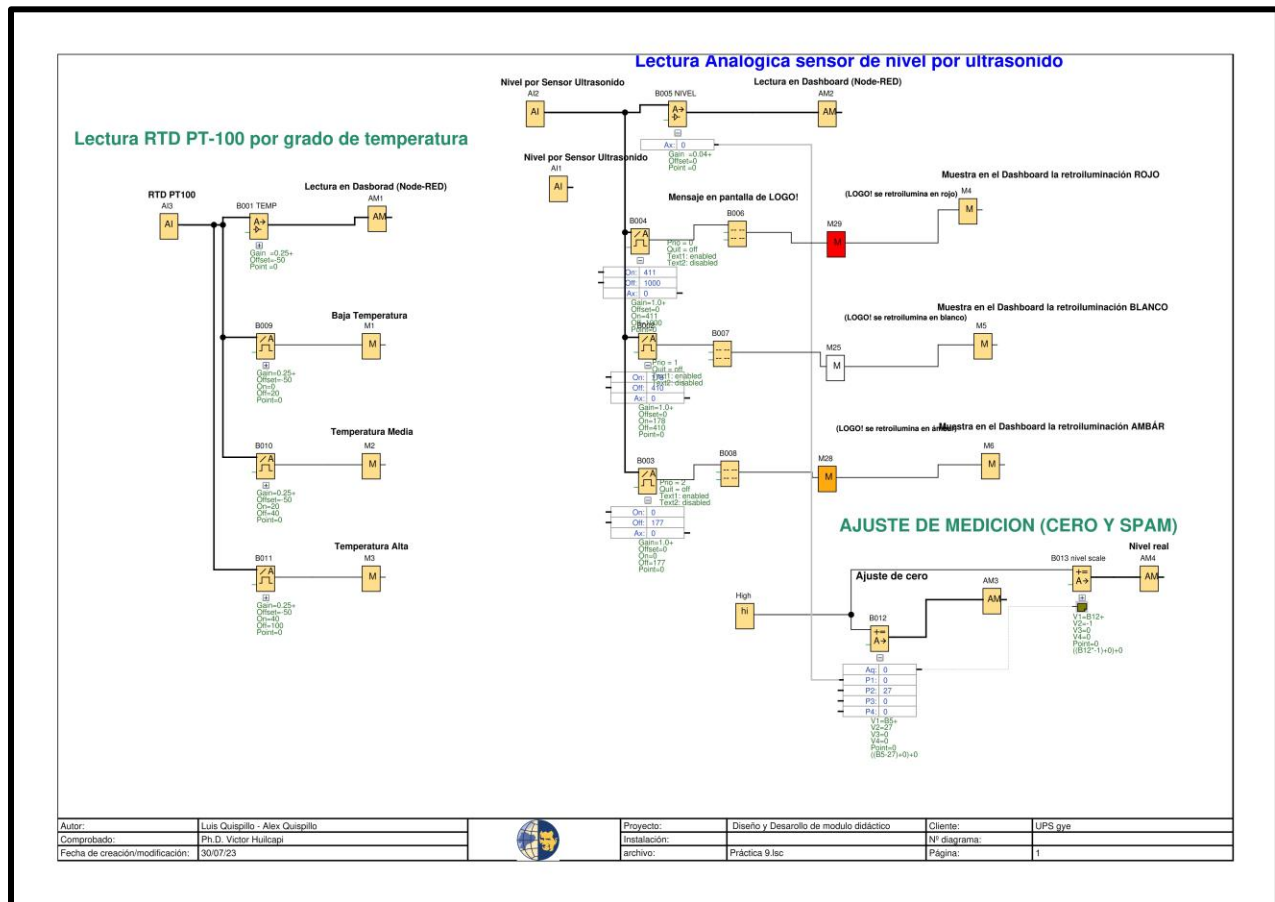
Figura 86. Ventana para la comunicación entre LOGO 8 y Raspberry Pi 4



- **Programación en LOGO! Soft Comfort V8.3 de la práctica 9**

En el desarrollo de la programación se utilizaron diferentes bloques como: entradas analógicas, estado alto, amplificador analógico, marca analógica, entrada, salida, marcas, conector abierto, texto de aviso y marcas para retroiluminación en LOGO!, figura 87.

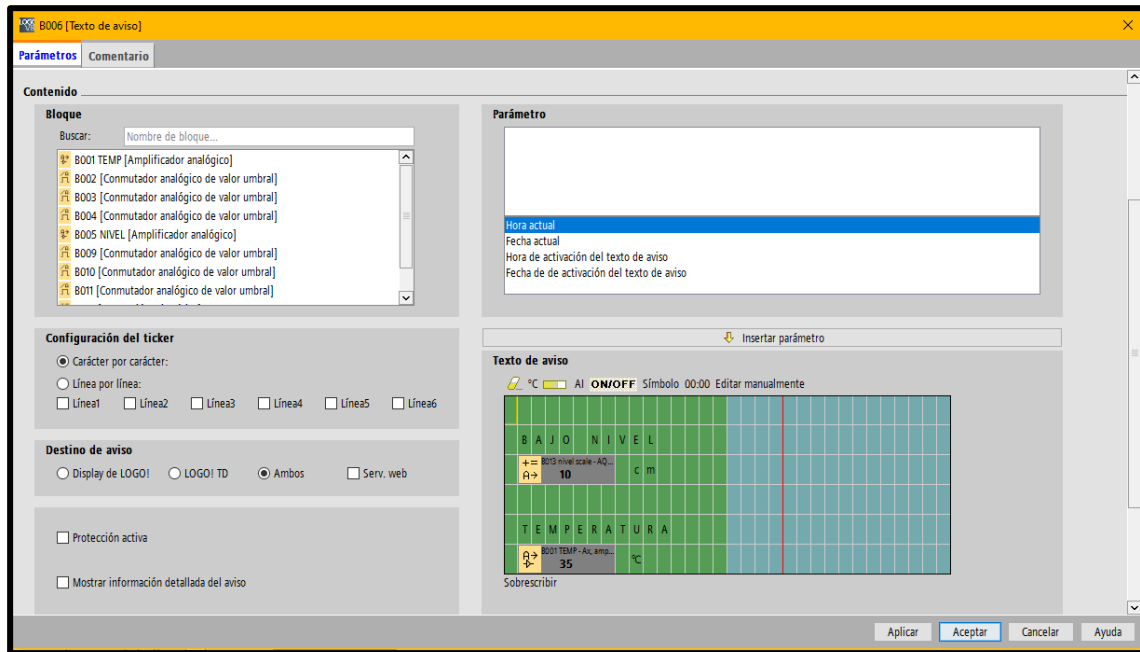
Figura 87. Programación en LOGO! Soft Comfort de la práctica 9



- Creación de texto aviso y retroiluminación en LOGO! 8

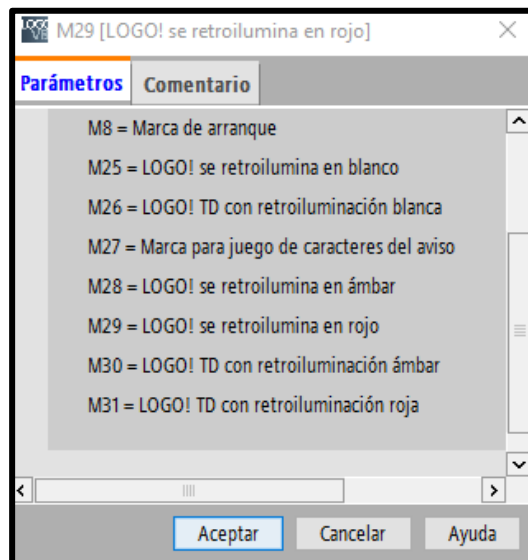
Desde el bloque B006 de texto de aviso se puede agregar y editar manualmente. Los diferentes tipos de bloques se puede encontrar en la parte izquierda, como se ilustra en la figura 88.

Figura 88. Texto de aviso nivel bajo práctica 9



Para programar las retroiluminaciones en LOGO! 8 se utilizan las respectivas marcas como se muestra en la figura 89.

Figura 89. Marca de retroiluminación en color rojo



- **Instalación de los complementos en el editor de flujo**

Como se observa en la tabla 9 se encuentran los complementos a utilizar en la práctica 9 para el desarrollo de los flujos en el entorno de trabajo de Node-RED.

Tabla 9. Complementos de práctica 9

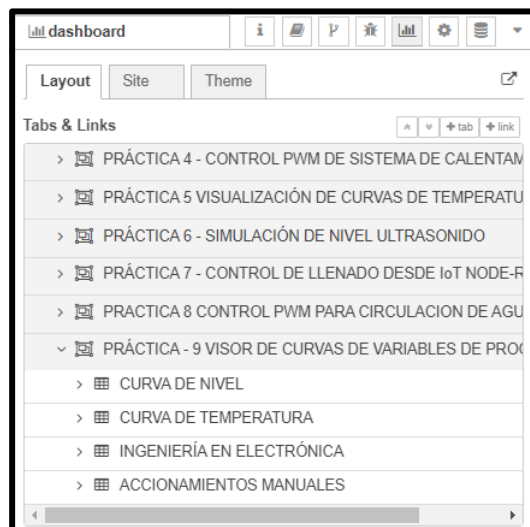
node-red	node-red-contrib-ui-led	node-red-dashboard
node-red-contrib-s7	node-red-contrib-ui-media	

- **Creación del grupo en dashboard**

En primer lugar, se crea la tabla, para eso se dirige en el parte superior derecho encontrará los iconos *Dashboard >Layout >+tab*.

Segundo, dentro de tabla se definió con el nombre *Práctica 9 – Visor de curvas de variables de proceso y avisos*, por último, se crean los grupos que están dentro de la tabla, como se ilustra en la figura 90.

Figura 90. Creación de grupo práctica 9



- **Programación grafica en Node-RED sobre la práctica 9**

Realizada la programación en LOGO! Soft Comfort, se comienza con la programación en el editor de flujo de Node-RED como se ilustra en las figuras 91 y 92 por los cuales se utilizaron los nodos: **s7 in, text, chart, gauge, change, template, comment, media**.

Figura 91. Práctica 9 primera parte de la programación visual

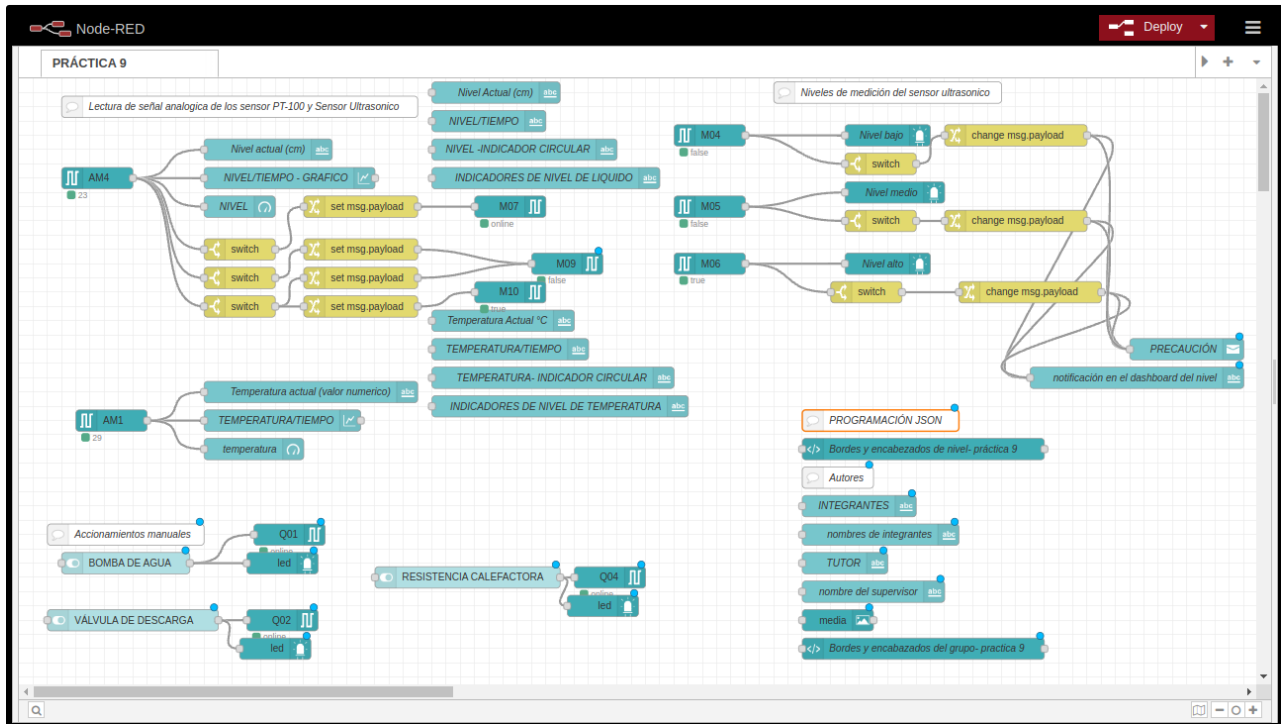
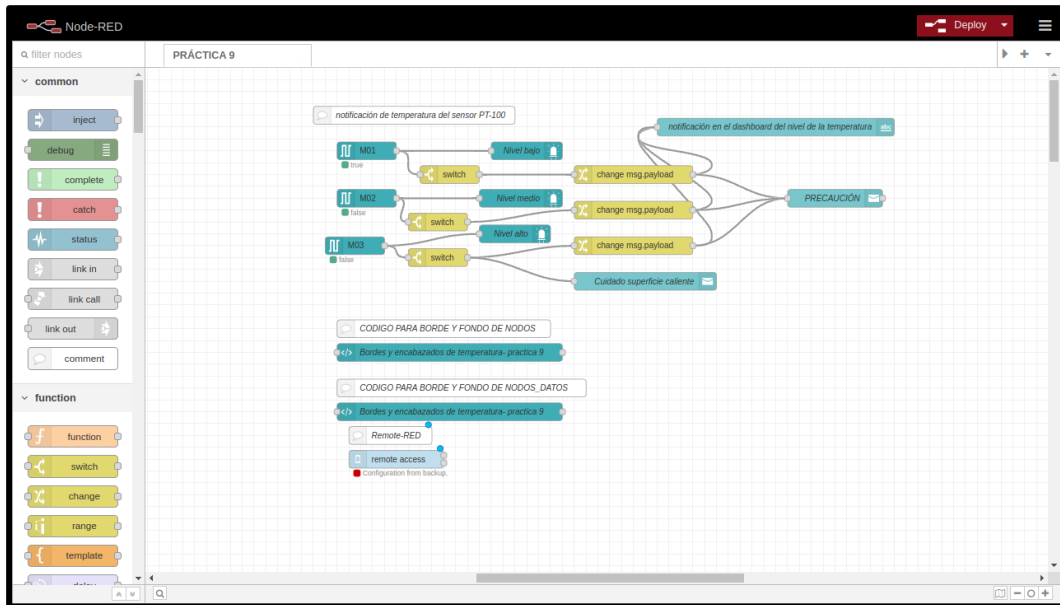


Figura 92. Práctica 9 segunda parte de la programación visual

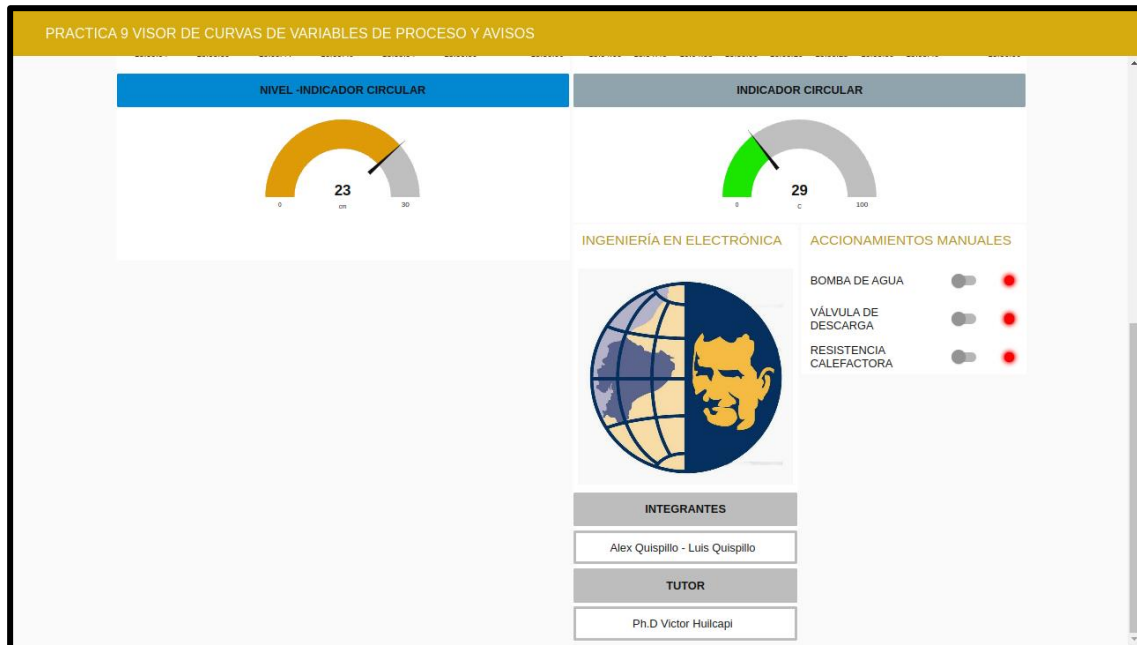


Finalizando, se observa el desarrollo del dashboard previamente asignada en cada nodo del complemento *node-red-dashboard* para creación de los widgets de la interfaz gráfica, como se ilustran en las figuras 93 y 94.


Figura 93. Dashboard de la práctica 9



Figura 94. Dashboard de la práctica 9



#### 4.10 Práctica #10

	GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA Ingeniería en electrónica	MATERIA
PRÁCTICA #10	TÍTULO DE LA PRÁCTICA Envío de mensajes de alarma por rebase de nivel por WhatsApp desde Node-red.
OBJETIVOS	
<ul style="list-style-type: none"><li>• Desarrollo de diagrama basado en flujo, utilizando los siguientes nodos: switch, range, change, remote red, notificación, WhatsApp, para generar avisos funcionales y avisos de alarmas.</li><li>• Entender la funcionalidad de programación basada por motor de flujo por Node-RED y la programación en LOGO! 8 en diagrama FUP para su respectiva simulación.</li></ul>	
PLANTEAMIENTO DE PROBLEMAS	
<ul style="list-style-type: none"><li>• Realizar lectura de temperatura a través de canal analógico de entrada de módulo AM2 RTD con termorresistencia PT100 de 3 hilos.</li><li>• Ajuste de cero y span del eco recibido en rango de tensión en el canal analógico AI1 que corresponde a la entrada I7 configurada como analógica en hardware (LOGO Soft y Node-RED).</li><li>• Utilizar instrucciones de comparador analógico de valor de umbral para establecer los diferentes rangos de nivel y temperatura.</li><li>• Mapeado de parámetros VM para poder escribir las consignas de comparación de temperatura y nivel.</li><li>• Crear la primera tabla en el dashboard donde se visualizarán los siguientes grupos:</li></ul>	

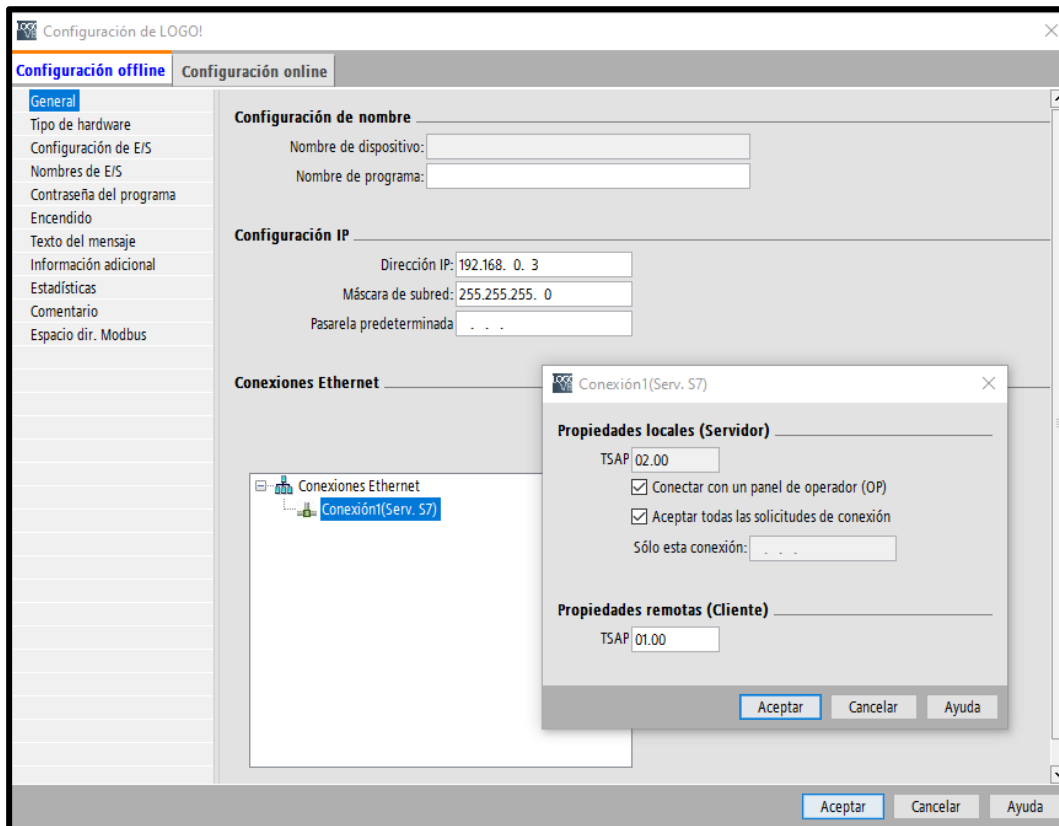


- a) Accionamientos manuales de los actuadores para poder simular errores del sistema en control de nivel y control de temperatura.
- b) Avisos de accionamiento para tener visualizar su estado de todos los actuadores.
- c) Alertas de: Estado y Nivel actual del tanque inferior.
- d) Notificaciones para el uso de imágenes de NODE y WHATSAPP, además tener una constancia del último mensaje enviado a las App.
- e) Nombre de los integrantes y supervisor de la práctica con el logo de la institución.
  - Crear la segunda tabla en el dashboard donde se visualizarán los siguientes widgets para la variable nivel:
    - f) Indicador tipo gauge.
    - g) Visor de curvas.
    - h) Luces piloto indicadoras de niveles.
  - i) Ajuste de rangos: bajo, intermedio y alto, escribiendo a través de nodos S7 OUT, a las variables mapeadas.
    - Crear la tercera tabla en el dashboard donde se visualizarán los siguientes widgets para la variable temperatura:
      - j) Indicador tipo gauge.
      - k) Visor de curvas.
      - l) Luces piloto indicadoras de niveles.
  - m) Ajuste de rangos: bajo, intermedio y alto, escribiendo a través de nodos S7 OUT, a las variables mapeadas.
    - Utilización de funciones en node red y nodos de comparación y poder generar diferentes tipos de avisos: texto, led, gauge.
    - Agregar bot Whatsapp CBM y obtener API KEY.
    - El desarrollo de la programación en esta práctica será en Node-RED, no en LOGO Soft Comfort V8.3 para aprovechar los recursos de cada uno de los diferentes nodos.
    - Crear acceso remoto y visualización de notificaciones en aplicación móvil Remote Red.

- **Programación en FUP de la práctica 10**

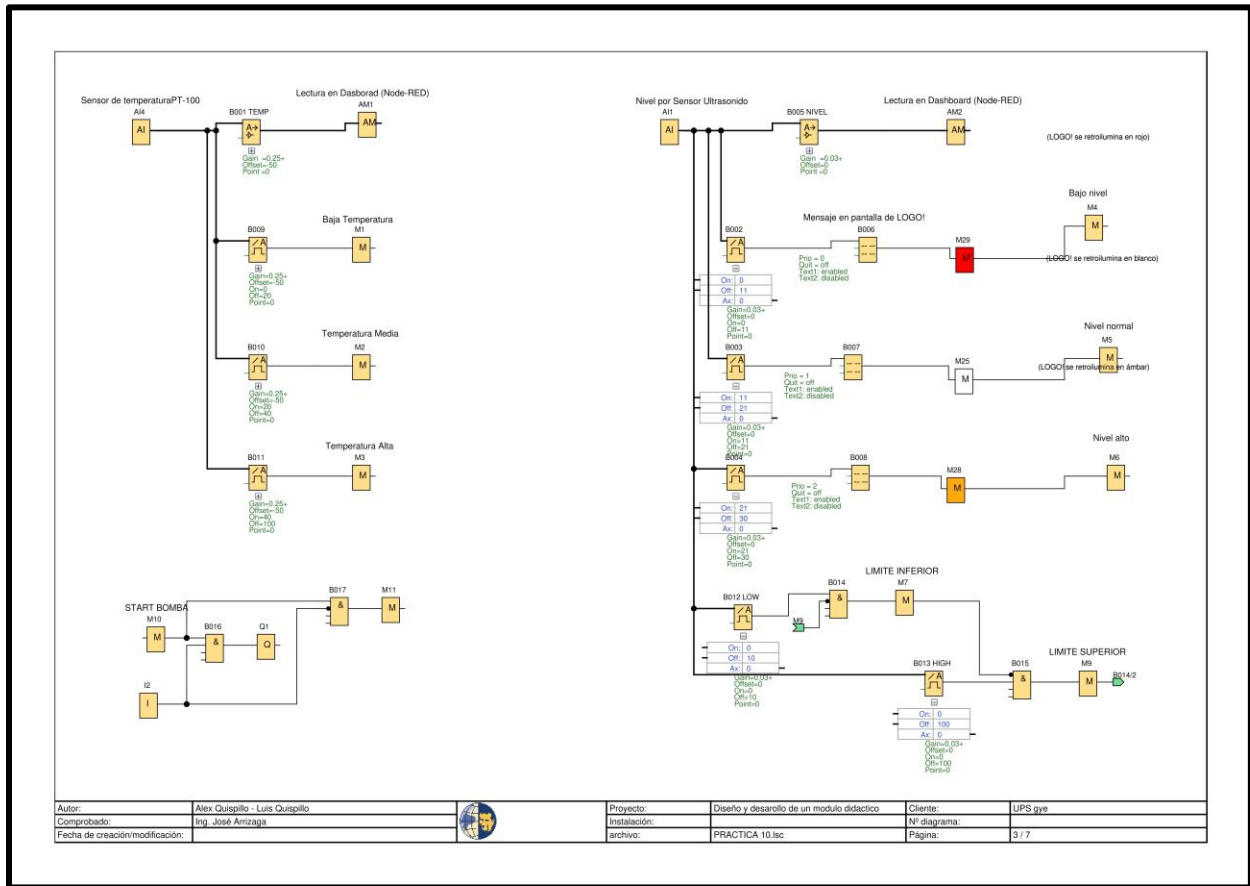
Para el desarrollo de esta práctica, se sugiere que se empiece con la configuración IP para compilar de manera inalámbrica por medio de la Red que nos proporciona el router, y luego con la comunicación S7 que nos proporciona con Node-RED por TSAP, como se ilustra en la figura 95.

Figura 95. Entorno de trabajo de LOGO Soft Comfort



Se analiza detenidamente el problema que se presenta y posteriormente se define las funciones y bloques a utilizar en la programación de LOGO! 8. Se adjunta los bloques utilizados para la respectiva práctica, como se ilustra en la figura 96.

Figura 96. Programación de la práctica 10



- **Instalación de los complementos en el editor de flujo**

Como se observa en la *Tabla 10* se encuentran los complementos a utilizar en la práctica 10 para el desarrollo de los flujos en el entorno de trabajo de Node-RED.

Tabla 10. Complementos de práctica 10

node-red	node-red-contrib-ui-led	node-red-dashboard
node-red-contrib-s7	node-red-contrib-ui-media	Node-red-contrib-whatsapp-cmb
Node-red-contrib-calc	Node-red-contrib-remote	

- **Creación de grupo en Dashboard**

En primer lugar, se crea la tabla, para eso se dirige en el parte superior derecho encontrará los iconos *Dashboard >Layout >+tab*.

Segundo, dentro de tabla que se define con el nombre *Práctica 10 - Envío de mensajes de alarma por rebase de nivel por WhatsApp*, se crean los grupos que están dentro de la tabla, de manera que, los nombres de los grupos ya creados, se define en la figura 97 para darle orden a los widgets de la interfaz gráfica.

Figura 97. Creación de los grupos para el dashboard práctica 10



- **Creación de flujo en workspace de Node-RED práctica 10**

Dentro del entorno de trabajo lo primero es, crear un flujo (pestaña) para empezar a desarrollar la programación visual por flujos. Como se ilustra en la tabla 11 se adjunta la configuración de los nodos utilizados como: *s7 in, s7 out, chart, text, media, comment, change, switch, led, button, template, sent WhatsApp*.

Cabe mencionar que cada nodo comprende una configuración distinta como se indica en la tabla, sino detalla las configuraciones de nodos porque fueron utilizadas en prácticas anteriores.

Tabla 11. Configuración nodos de práctica 10

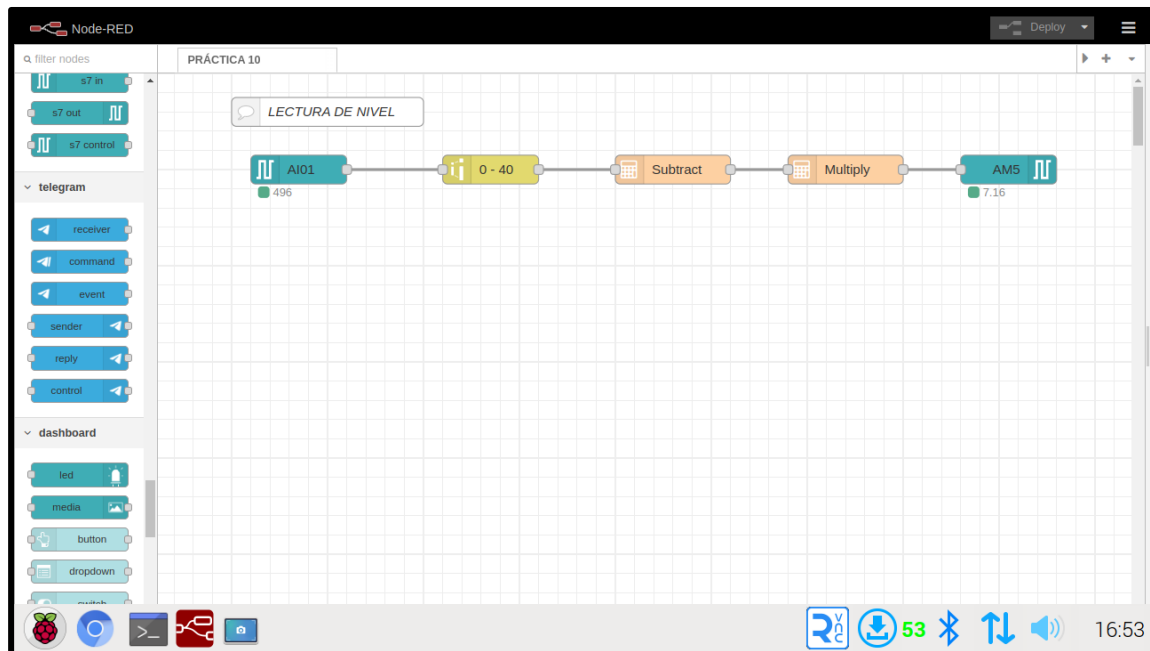
NIVELES DE TEMPERATURA		NIVELES DE TANQUE		LENGUAJE DE PROGRAMACIÓN JSON
<p><b>(S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable M01</p> <p><b>Nodo switch</b> Property msg.payload IS TRUE</p> <p><b>NIVEL BAJO (Nodo LED)</b> Group: nombre del grupo Size: 6x2 Label: Nivel bajo Color for value of msg. payload False: black True: Red</p> <p><b>Nodo Change</b> Change/ msg/ payload Search for: Boolean (true) Replace with: string/Temperatura nivel bajo.</p> <p><b>M03 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable M03</p> <p><b>Nodo Switch</b> Property msg.payload IS TRUE</p> <p><b>NIVEL MEDIO (Nodo LED)</b> Group: nombre del grupo Size: 7x2 Label: Nivel alto Color for value of msg. payload False: black True: orange</p> <p><b>Nodo Change</b> Change/ msg/ payload Search for: Boolean (true) Replace with: string/Temperatura nivel alto)</p>	<p><b>M02 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable M02</p> <p><b>Nodo Switch</b> Property msg.payload IS TRUE</p> <p><b>NIVEL MEDIO (Nodo LED)</b> Group: nombre del grupo Size: 6x2 Label: Nivel medio Color for value of msg. payload False: black True: Green</p> <p><b>Nodo Change</b> Change/ msg/ payload Search for: Boolean (true) Replace with: string/Temperatura nivel medio)</p> <p><b>Estado de temperatura (text)</b> Group: nombre del grupo Layout: center &lt;/&gt;Class: encabezado_1 Name: Estado de temperature</p> <p><b>Cuidado superficie caliente (Nodo notificación)</b> Layout: OK/Cancel Dialog Default: OK Topic: cuidado superficie caliente Name: cuidado superficie caliente</p>	<p><b>M04 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable M04</p> <p><b>Nodo switch</b> Property msg.payload IS TRUE</p> <p><b>NIVEL BAJO (Nodo LED)</b> Group: nombre del grupo Size: 6x2 Label: Nivel bajo Color for value of msg. payload False: black True: Red</p> <p><b>Nodo Change</b> Change/ msg/ payload Search for: Boolean (true) Replace with: string/bajo nivel</p>	<p><b>M06 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable M06</p> <p><b>Nodo switch</b> Property msg.payload IS TRUE</p> <p><b>NIVEL ALTO (Nodo LED)</b> Group: nombre del grupo Size: 7x2 Label: Nivel alto Color for value of msg. payload False: black True: Red</p> <p><b>Nodo Change</b> Change/ msg/ payload Search for: Boolean (true) Replace with: string/nivel exceso</p>	<p><b>Nodo template</b> .bordes { font-size: 0.4cm; color: white !important; border: 5px solid #2979FF; } .nr-dashboard-text.encabezado { background-color: #2979FF; color: black !important; font-size: 0.5cm; } .nr-dashboard-switch.resistencia { background-color: #FF3D00; color: white !important; } .nr-dashboard-switch.valvula { background-color: #283593; color: white !important; } .nr-dashboard-switch.bomba { background-color: #212121; color: white !important; } .nr-dashboard-text.encabezado_1 { background-color: #FF6D00; color: black !important; }</p>
		<p><b>M05 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable M05</p> <p><b>Nodo switch</b> Property msg.payload IS TRUE</p> <p><b>NIVEL MEDIO (Nodo LED)</b> Group: nombre del grupo Size: 6x2 Label: Nivel bajo Color for value of msg. payload False: black True: Red</p> <p><b>Nodo Change</b> Change/ msg/ payload Search for: Boolean (true) Replace with: nivel estable</p>	<p><b>Estado de nivel (text)</b> Group: nombre del grupo Size: 4x1 Layout: center &lt;/&gt;Class: encabezado Name: Estado de nivel</p> <p><b>PRECAUCIÓN (Nodo notificación)</b> Layout: Top Right Timeout(s): 6 Border: blue Topic: Precaución Name: Precaución</p>	

CONFIGURACIÓN DE LIMITE DE TEMPERATURA	CONFIGURACIÓN DE LIMITE DE NIVEL	AVISOS INDICADORES		LECTURA DE NIVEL (ajuste de cero y spam)
<p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 20 Label limite inferior °C Topic: msg.topic</p> <p><b>VW0 (S7 in)</b> PLC: S7-LOGO8 Variable VW0</p> <p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 20 Label limite superior °C Topic: msg.topic</p> <p><b>VW2 (S7 in)</b> PLC: S7-LOGO8 Variable VW2</p> <p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 40 Label limite inferior °C Topic: msg.topic</p> <p><b>VW4 (S7 in)</b> PLC: S7-LOGO8 Variable VW4</p> <p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 40 Label limite superior °C Topic: msg.topic</p> <p><b>VW6 (S7 in)</b> PLC: S7-LOGO8 Variable VW6</p>	<p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 20 Label limite inferior °C Topic: msg.topic</p> <p><b>VW12 (S7 in)</b> PLC: S7-LOGO8 Variable VW12</p> <p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 20 Label limite superior °C Topic: msg.topic</p> <p><b>VW14 (S7 in)</b> PLC: S7-LOGO8 Variable VW14</p> <p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 40 Label limite inferior °C Topic: msg.topic</p> <p><b>VW16 (S7 in)</b> PLC: S7-LOGO8 Variable VW16</p> <p><b>Nodo Numeric</b> Group: nombre del grupo Size: auto Range min 0 max 40 Label limite superior °C Topic: msg.topic</p> <p><b>VW18 (S7 in)</b> PLC: S7-LOGO8 Variable VW18</p>	<p><b>Q01 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable Q01</p> <p><b>Nodo switch</b> Property msg.payload IS TRUE IS FALSE</p> <p><b>Q01 - 2 rules (change)</b> Change: msg.payload Search for: true Replace with Encendido Change: msg.payload Search for: false Replace with: Apagado</p> <p><b>Nodo send message (WhatsApp)</b> Account: Add new Name: bomba On Message: Bomba encendida</p> <p><b>Nodo send message (WhatsApp)</b> Account: Add new Name: bomba off Message: Bomba desconectada</p> <p><b>BOMBA DE AGUA (text)</b> Group: nombre del grupo Size: auto Label: bomba de agua</p>	<p><b>Q02 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable Q02</p> <p><b>Nodo switch</b> Property msg.payload IS TRUE IS FALSE</p> <p><b>Q01 - 2 rules (change)</b> Change: msg.payload Search for: true Replace with Encendido Change: msg.payload Search for: false Replace with: Apagado</p> <p><b>Nodo send message (WhatsApp)</b> Account: Add new Name: Válvula On Message: válvula abierta</p> <p><b>Nodo send message (WhatsApp)</b> Account: Add new Name: valvula off Message: valvula cerrada</p> <p><b>VÁLVULA SOLENOIDE (text)</b> Group: nombre del grupo Size: auto Label: válvula solenoide</p>	<p><b>AI01 (S7 in)</b> PLC: S7-LOGO8 Mode: Single variable Variable AI01</p> <p><b>Nodo range</b> Property: msg.payload Action: scale the message property Map the input range From: 0 to: 1000 To the target range: From: 0 to: 40</p> <p><b>Nodo calculator</b> Input field: msg.payload Output field: msg.payload Operation: Subtract # constant 27 Round result to 2 decimals</p> <p><b>Nodo calculator</b> Input field: msg.payload Output field: msg.payload Operation: Multiply # constant -1 Round result to 2 decimals</p> <p><b>AM5 (S7 in)</b> PLC: S7-LOGO8 Variable AM5</p> <p><b>NOTIFICACIONES</b></p> <p><b>REMOTE RED</b></p> <p><b>Nodo switch</b> Property msg.payload IS TRUE IS FALSE</p> <p><b>Nodo change</b> Name: ON Change msg.payload Search for: true Replace with: JSON. { "title": "BOMBA", "body": "ENCENDIDA" }</p> <p><b>Nodo filter</b> Mode: block unless value changes Property: msg.payload Msg topic</p> <p><b>Nodo remote- notification</b> Config: nombre de configuración Title: msg.payload.title Body: msg.payload.body Sound: prostine</p> <p><b>Output: forward msg</b></p>

- Programación grafica en Node-RED sobre la práctica 10

Esta práctica se realiza el ajuste de cero y spam en el editor de flujo, ya que existen nodos para su aplicación de esta manera se muestra un ajuste más simplificado, pero no quiero desacreditar la programación el LOGO! Soft Comfort ya que cada programa de desarrollo tiene sus ventajas y desventajas, como se ilustra en la figura 98.

Figura 98. Ajuste de cero y spam en Node-RED



Realizada el ajuste de cero y spam, se comienza con la programación visual respecto al planteamiento del problema, como se muestra en la figura 99, figura 100 y figura 101.

Los nodos utilizados fueron: *s7 in*, *text*, *media*, *numeric*, *whin-send*, *change*, *switch*.

Figura 99. Práctica 10 primera parte

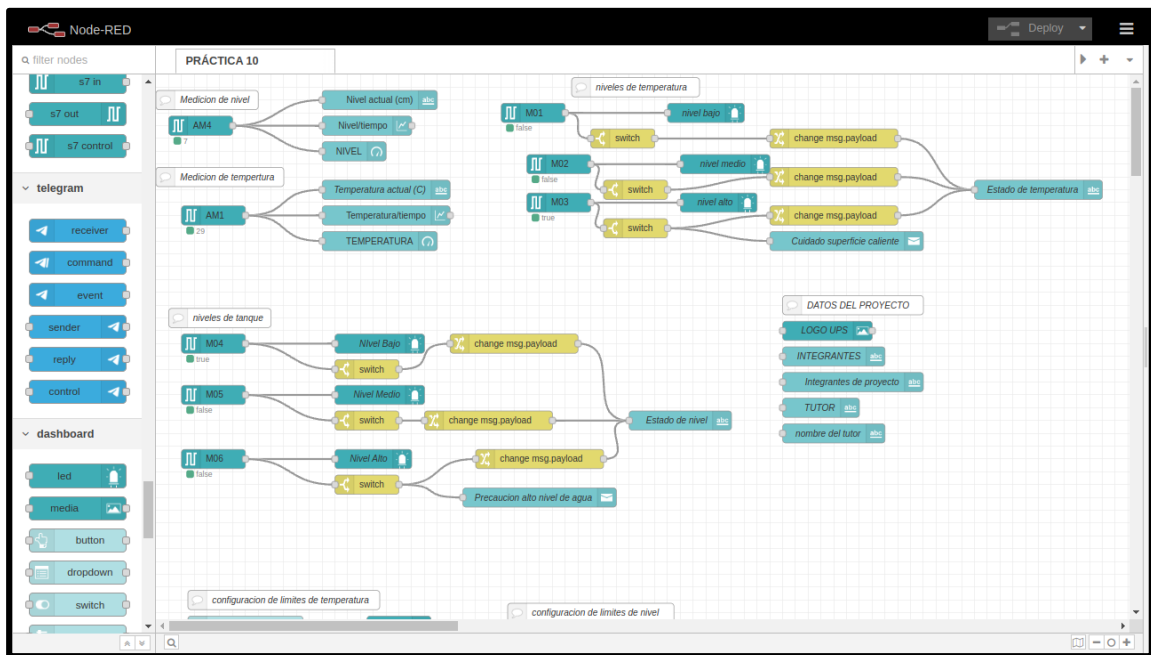


Figura 100. Práctica 10 segunda parte

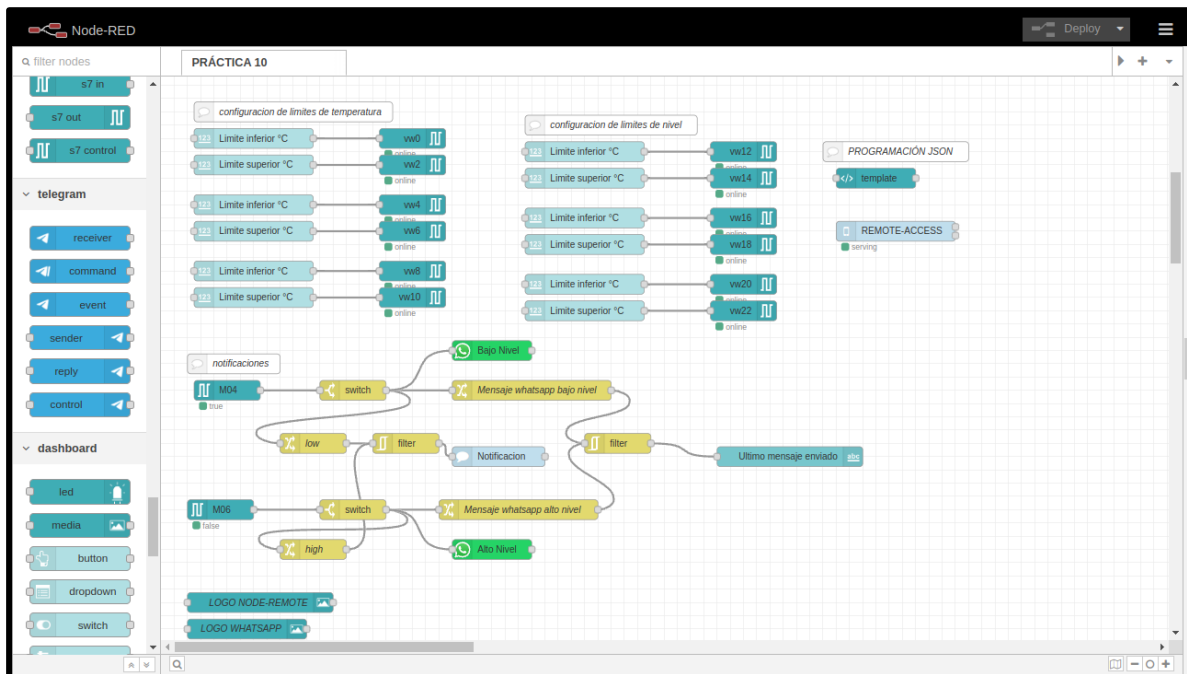
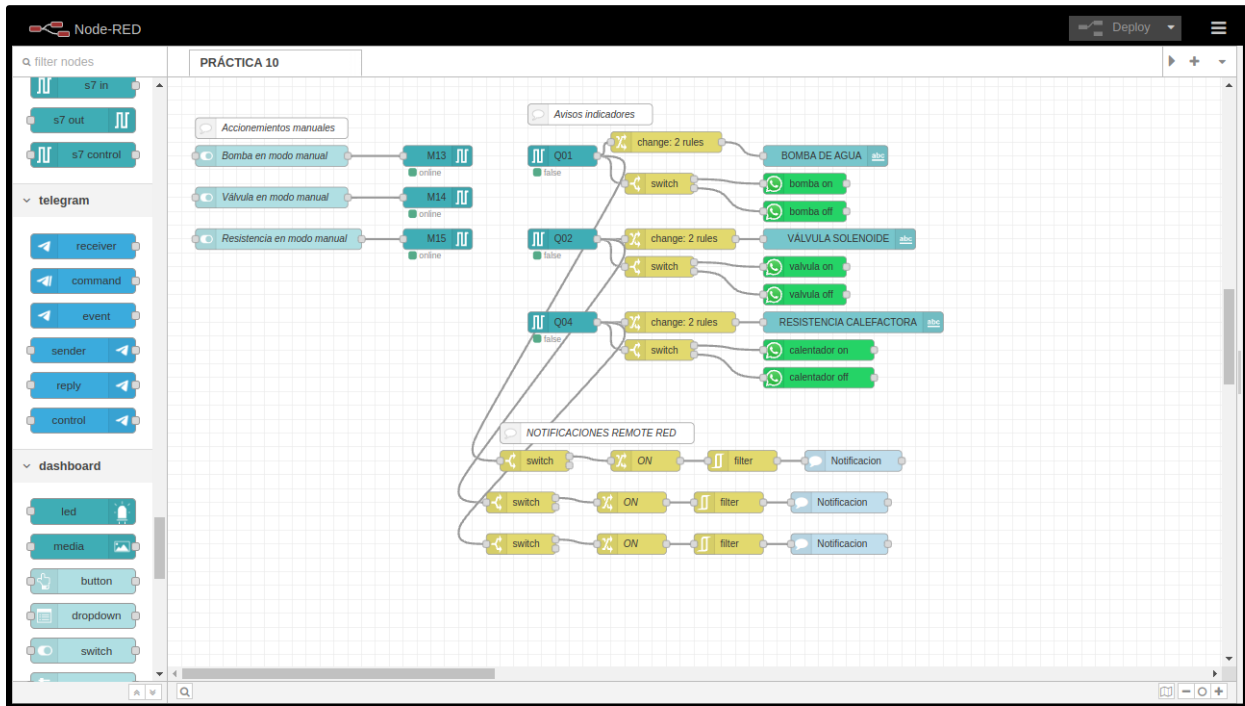




Figura 101. Práctica 10 tercera parte



Finalizado la programación en el editor de flujo, se compila el dashboard, este nos crea en una ventana nueva donde se visualizará. En esta práctica se crearon tres tablas de los cuales varios grupos fueron creados para agrupar los widgets para darle más secciones de ventana dentro del dashboard al seleccionar el menú en el parte superior izquierdo, figura 102.

Figura 102. menú en la ventana de dashboard práctica 10



En la figura 103 se muestra el dashboard principal en el que se encuentra los autores principales y accionamiento y avisos, en la figura 104 el dashboard muestra todo lo relacionado al control y monitoreo del nivel así como en la figura 105 el monitoreo y control de temperatura.

Figura 103. Tabla principal de la práctica 10

PRÁCTICA 10 - ENVIO DE MENSAJES DE ALARMA POR REBASE DE NIVEL POR WHATSAPP

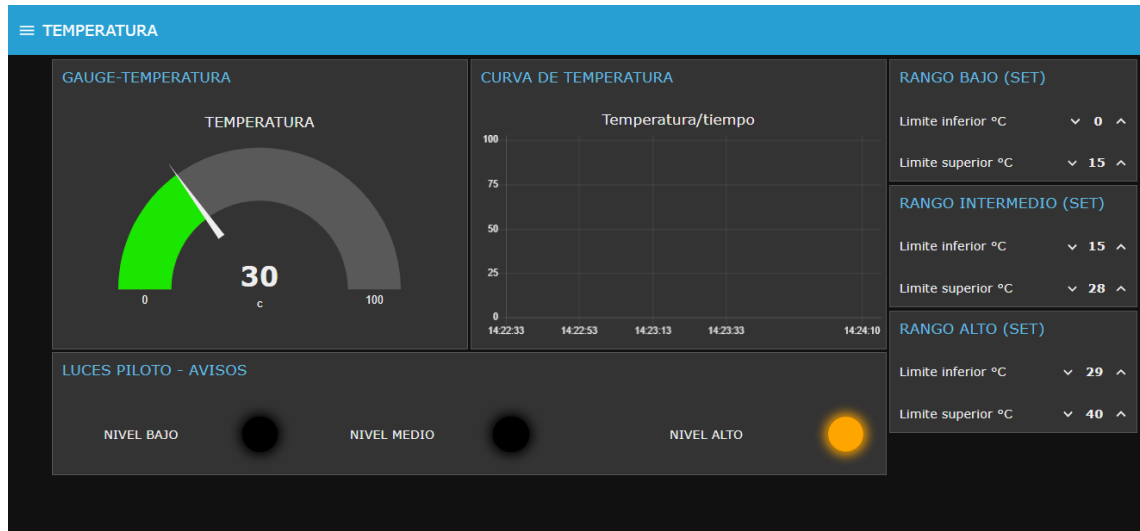
<p>INGENIERÍA ELECTRÓNICA</p> <p><b>INTEGRANTES</b></p> <p>Alex Quispillo - Luis Quispillo</p> <p><b>TUTOR</b></p> <p>Ph.D. Victor Huilcapi</p>	<p>ACCIONAMIENTO MANUALES</p> <p>BOMBA <input type="checkbox"/></p> <p>RESISTENCIA CALEFACTORA <input checked="" type="checkbox"/></p> <p>VÁLVULA SOLENOIDE <input type="checkbox"/></p>	<p>AVISOS DE ACCIONAMIENTOS</p> <p>BOMBA DE AGUA Apagado</p> <p>VÁLVULA SOLENOIDE Apagado</p> <p>RESISTENCIA CALEFACTORA Apagado</p>						
	<p>NOTIFICACIONES</p> <p><b>Remote RĒD</b></p> <p></p> <p>Ultimo mensaje enviado: Bajo nivel en el tanque.</p>	<p>ALERTAS</p> <table border="1"> <tr> <td>ESTADO: <b>Bajo nivel</b></td> <td>Nivel actual (cm) <b>7</b></td> </tr> <tr> <td colspan="2">ESTADO: <b>Temperatura nivel alto</b></td> </tr> <tr> <td colspan="2">Temperatura actual (C) <b>29</b></td> </tr> </table>	ESTADO: <b>Bajo nivel</b>	Nivel actual (cm) <b>7</b>	ESTADO: <b>Temperatura nivel alto</b>		Temperatura actual (C) <b>29</b>	
	ESTADO: <b>Bajo nivel</b>	Nivel actual (cm) <b>7</b>						
	ESTADO: <b>Temperatura nivel alto</b>							
Temperatura actual (C) <b>29</b>								

Figura 104. Tabla 2 de nivel de la práctica 10

NIVEL

<p>GAUGE-NIVEL</p> <p>NIVEL</p> <p>27 cm</p>	<p>CURVA DE NIVEL</p> <p>Nivel/tiempo</p>	<p>RANGO BAJO (SET)</p> <p>Limite inferior °C <b>0</b></p> <p>Limite superior °C <b>11</b></p> <p>RANGO INTERMEDIO (SET)</p> <p>Limite inferior °C <b>12</b></p> <p>Limite superior °C <b>20</b></p> <p>RANGO ALTO (SET)</p> <p>Limite inferior °C <b>21</b></p> <p>Limite superior °C <b>27</b></p>
<p>LUCES PILOTOS AVISOS</p> <p>NIVEL BAJO <input type="checkbox"/> NIVEL MEDIO <input type="checkbox"/> NIVEL ALTO <input checked="" type="checkbox"/></p>		

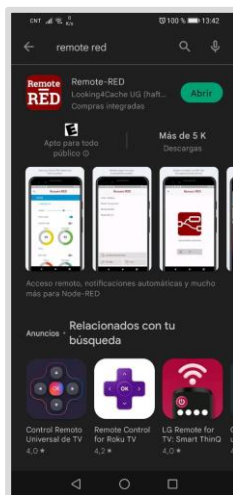
Figura 105. Tabla 3 de nivel de la práctica 10



- **Control del proceso por medio de la App Remote-RED**

Para el respectivo control y lectura por medio de esta App, es necesario tener instalado la aplicación, lo realiza desde Play Store Google, como se ilustra en la figura 106.

Figura 106. Descarga de Remote-RED desde la Play Store



Una vez instalado la App, se dirige al editor de flujo de Node-RED, dando clic en el nodo **remote Access**, luego, se dirige al icono de un “lápiz” para entrar a **Edit remote-config node**, da clic en el icono e ingresa, figura 107 A y B.

Tomar en cuenta que esta creada, sino deberá crear una nueva configuración con el nombre que desee, en este caso está escrito como se ilustra en la figura 108.

Figura 107. Dentro del nodo remote-access

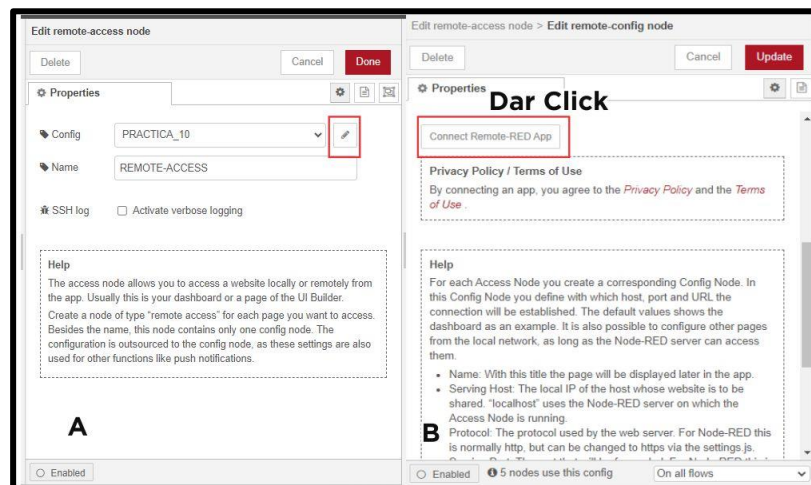
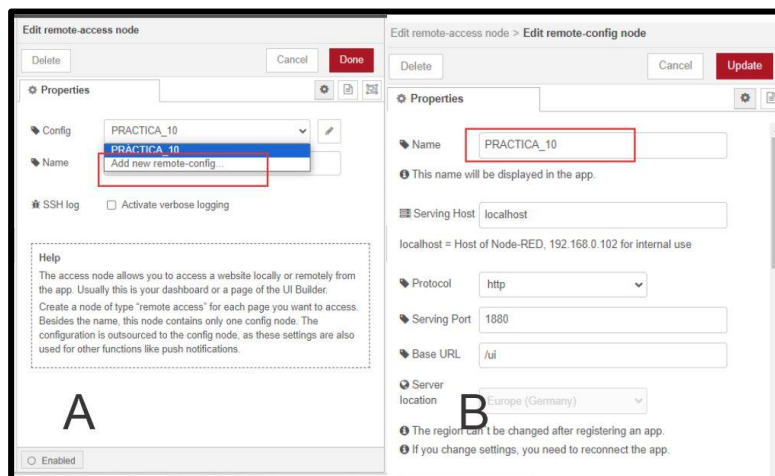


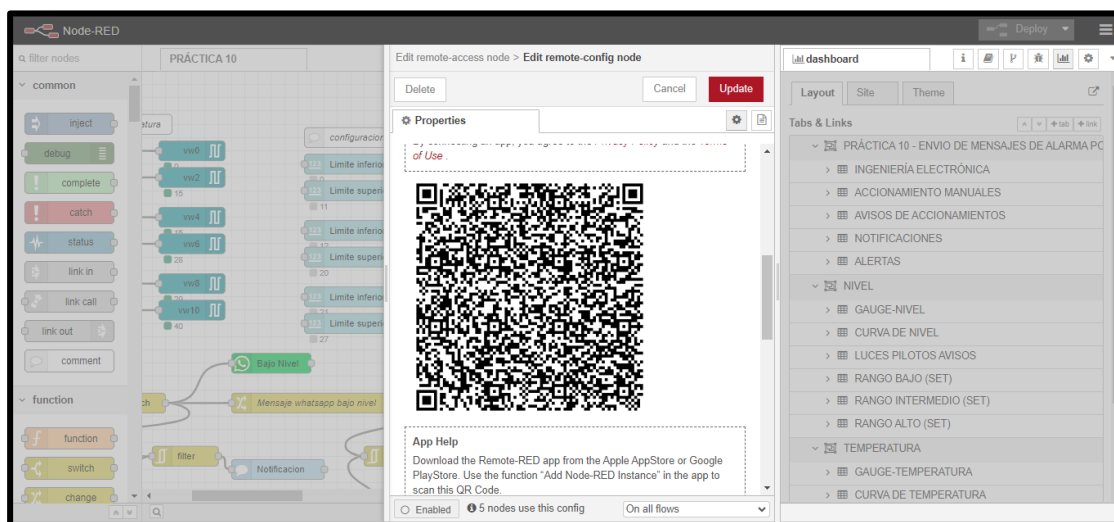
Figura 108. Crear nueva configuración



Se da clic en **Connect Remote-RED App** aparecerá el QR, como se muestra en la figura 109, lo escanea con la App Remote-RED y aparecerá el editor de flujo y la ventana de dashboard del proceso en este caso de la práctica 10.

Tomar en cuenta que para poder visualizar y editar, es necesario de tener una versión pagada, la versión gratuita solo dura un día.

Figura 109. QR de remote-access para visualización del dashboard de la práctica 10

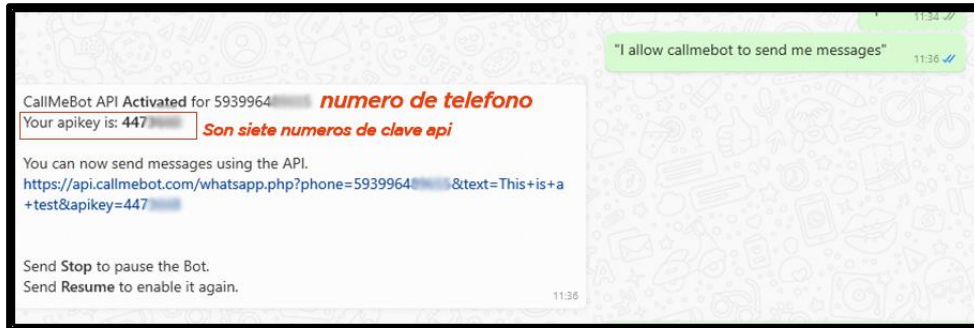


- **Uso del BOT del node-red-contrib-whatsapp-cmb**

Para el uso del BOT de este nodo es necesario seguir estos pasos:

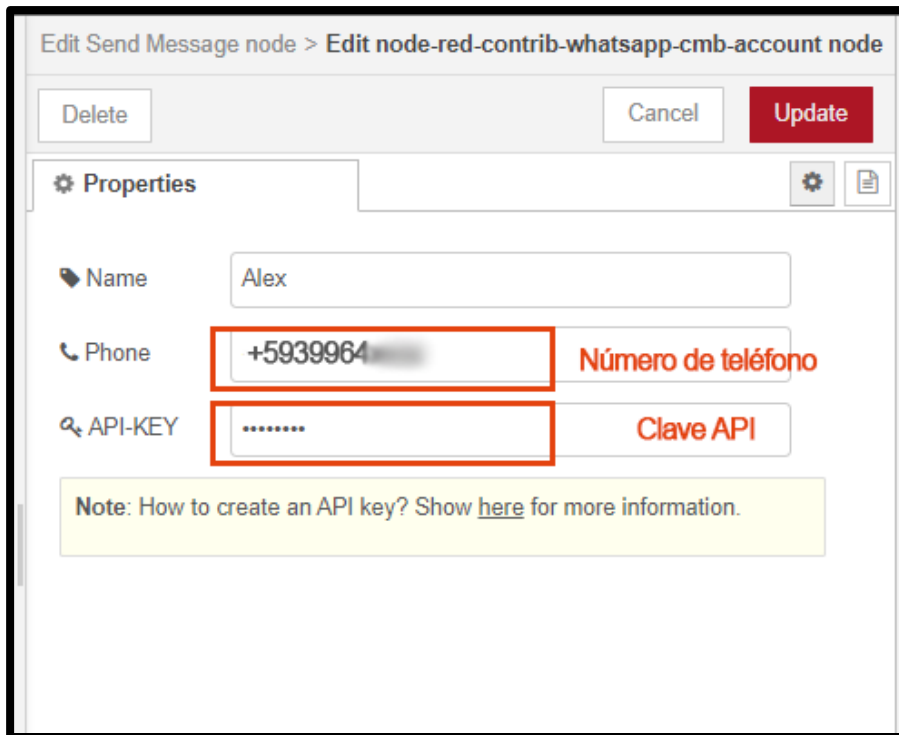
- Se debe registrar el numero **+34644537849** como contacto.
- Enviar un mensaje desde WhatsApp **"I allow callmebot to send me messages"**.
- El BOT le devuelve el mensaje indicando la **Apikey** como se muestra en la figura 110.
- Tomar en cuenta que cada número de teléfono personal envía una apikey personal.
- Para que el BOT te reenviarte tu **Apikey** enviar el mensaje **"Recover apikey"**.

Figura 110. Apikey del Bot de nodo WhatsApp



Desde luego al utilizar el nodo *send message* en primera instancia se debe configurar la recepción de los mensajes con el número de teléfono y la api-key, como se ilustra en la *figura 111*.

Figura 111. Configuración API-KEY de WhatsApp



## 5. ANEXOS

### 5.1 Resolución teórico y práctico del planteamiento #1:

Para realizar el cableado de control y potencia fueron necesarios los siguientes pasos:

1. Dimensionamiento de protecciones, según cargas eléctricas.
2. Selección de calibre de conductores.
3. Selección de estándar para cableado de red.
4. Elaboración de diagramas de control y potencia.
5. Ubicación de canaletas, rieles, borneras de conexión y demás.
6. Etiquetado y rotulado de conexiones.
7. Utilización de terminales.

- **Dimensionamiento de protecciones según cargas eléctricas:**

La protección de un circuito eléctrico, contra sobrecargas y cortocircuitos es indispensable, razón por la cual fue seleccionado un interruptor termomagnético de tipo C de 6 amperios para el circuito de control y de 16 amperios para el circuito de potencia.

El circuito de control consta de las siguientes cargas:

- Fuente conmutada AUTONIC de 24V/2.5A
- LOGO Siemens 24 RCE
- Relé de estado sólido 24 VDC
- 3 relés conmutadores 24 VDC
- 4 luces piloto de 24 VDC
- El circuito de potencia consta de las siguientes cargas:
- UPS de 500 VA
- Resistencia de calentamiento de 2000 w

- **Selección de calibre de conductores:**

La selección del calibre de conductores se realizó tomando como referencia la tabla 12 en el que describe los conductores de AWG que especifica la intensidad máxima de cada calibre.

Tabla 12. Tabla de medidas de cables AWG

<b>AMPERAJE QUE SOPORTAN LOS CABLES DE COBRE</b>					
Nivel de temperatura:	60°C	75°	90°C	60°C	
TIPO	TW	RHW, THW, THWN	THHN, XHHW-2, THWN-2	SPT	
MEDIDA/ CALIBRE DE CABLE	AMPERAJE SOPORTADO			MEDIDA/ CALIBRE DE CABLE	AMPERAJE SOPORTADO
14 AWG	15 A	15 A	15 A	20 AWG	2 A
12 AWG	20 A	20 A	20 A		
10 AWG	30 A	30 A	30 A		
8 AWG	40 A	50 A	55 A	18 AWG	10 A
6 AWG	55 A	65 A	75 A		
4 AWG	70 A	85 A	95 A		
3 AWG	85 A	100 A	115 A	16 AWG	13 A
2 AWG	95 A	115 A	130 A		
1 AWG	110 A	130 A	145 A		
1/0 AWG	125 A	150 A	170 A	14 AWG	18 A
2/0 AWG	145 A	175 A	195 A		
3/0 AWG	165 A	200 A	225 A	12 AWG	25 A
4/0 AWG	195 A	230 A	260 A		

Según la tabla de conductores, los calibres adecuados son: 18 AWG para el circuito de control y 14 AWG para el circuito de potencia.

El tipo de cable utilizado es THHN para el circuito de control y potencia, a excepción de la resistencia eléctrica de calentamiento, debido a la temperatura que esta puede llegar a alcanzar durante su funcionamiento, para esta se seleccionó un cable con aislante de asbesto de calibre 14 AWG.



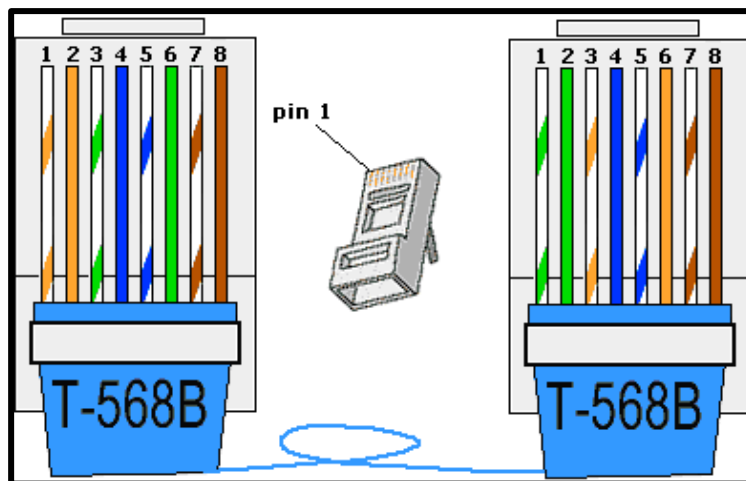
- **Selección de estándar para cableado de red**

El estándar seleccionado fue el T568B el cual se emplea en conexiones de red con dispositivos que no son iguales, donde se utiliza una topología de red de tipo estrella a través de un router de 4 puertos como se muestra en la *tabla 13* y de manera grafica como se expone en la figura 112. En este estándar la distribución de los colores es exactamente igual en ambos extremos.

Tabla 13. Cableado estándar T568B

<b>Pin 1</b>	TX+ Blanco-Naranja
<b>Pin 2</b>	TX- Naranja
<b>Pin 3</b>	RX+ Blanco-Verde
<b>Pin 4</b>	BDD- Azul
<b>Pin 5</b>	Blanco-Azul
<b>Pin 6</b>	RX- Verde
<b>Pin 7</b>	BDD+ Blanco-Marrón
<b>Pin 8</b>	BDD- Marrón

Figura 112. Estándar T568B



Fuente: (redes23, 2013)

- **Elaboración de diagramas de control y potencia**

Los diagramas fueron elaborados en una demo del software AutoCAD Electrical, el cual cuenta con una librería de símbolos eléctricos y que permite identificar de manera específica cada uno de los diferentes elementos.

En el diagrama de control, como se ilustra en la figura 113, se definieron las terminales de conexión de entrada y salida, conexión del RTD PT-100, su alimentación, así mismo se definieron el color que representa el + y – en DC.

En el diagrama de potencia se muestran las conexiones desde el disyuntor diferencial, hasta las diferentes conexiones como fuentes de alimentación para el sistema de control, así mismo las diferentes cargas, como se ilustra en la figura 114.

Figura 113. Diagrama de control

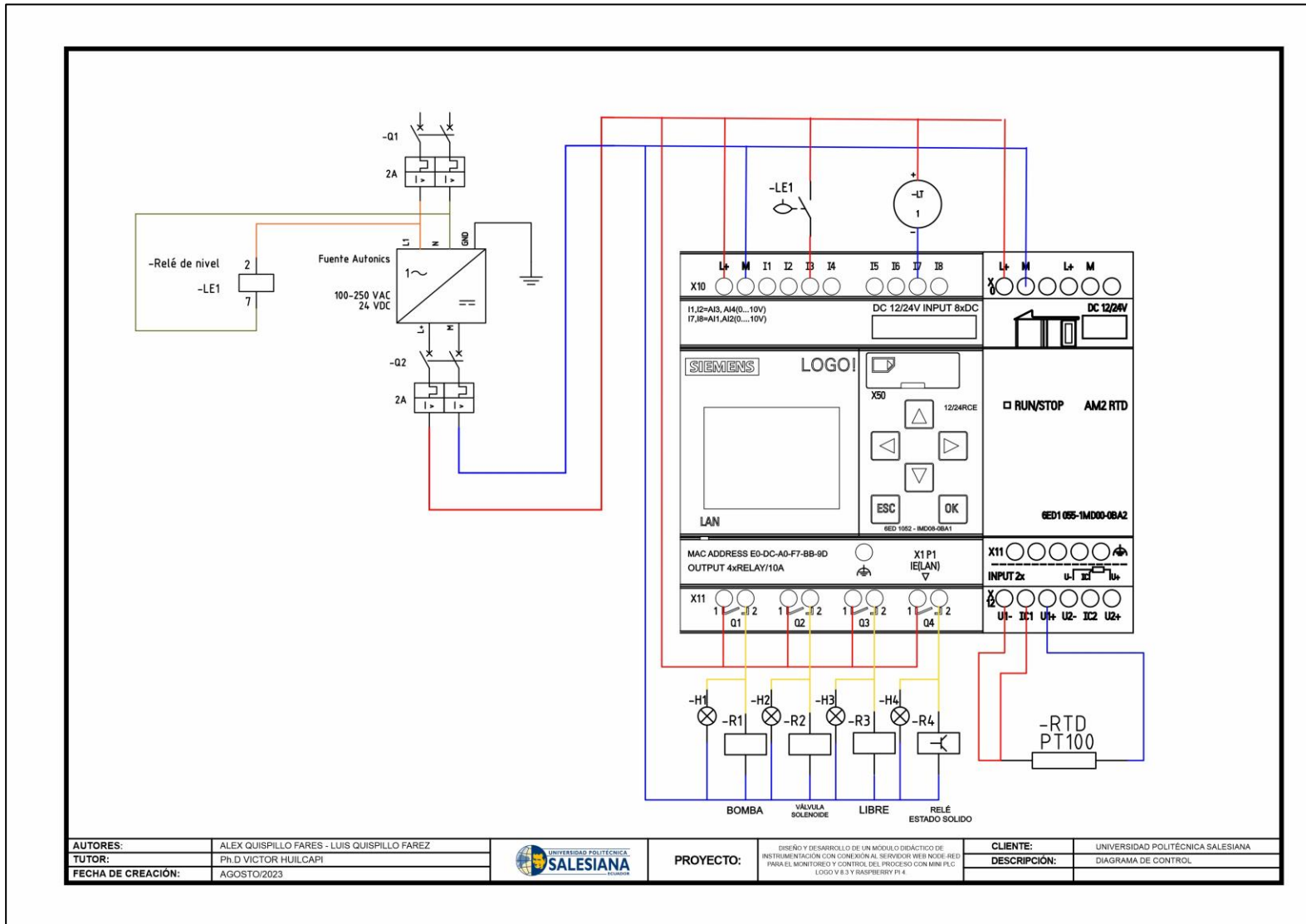
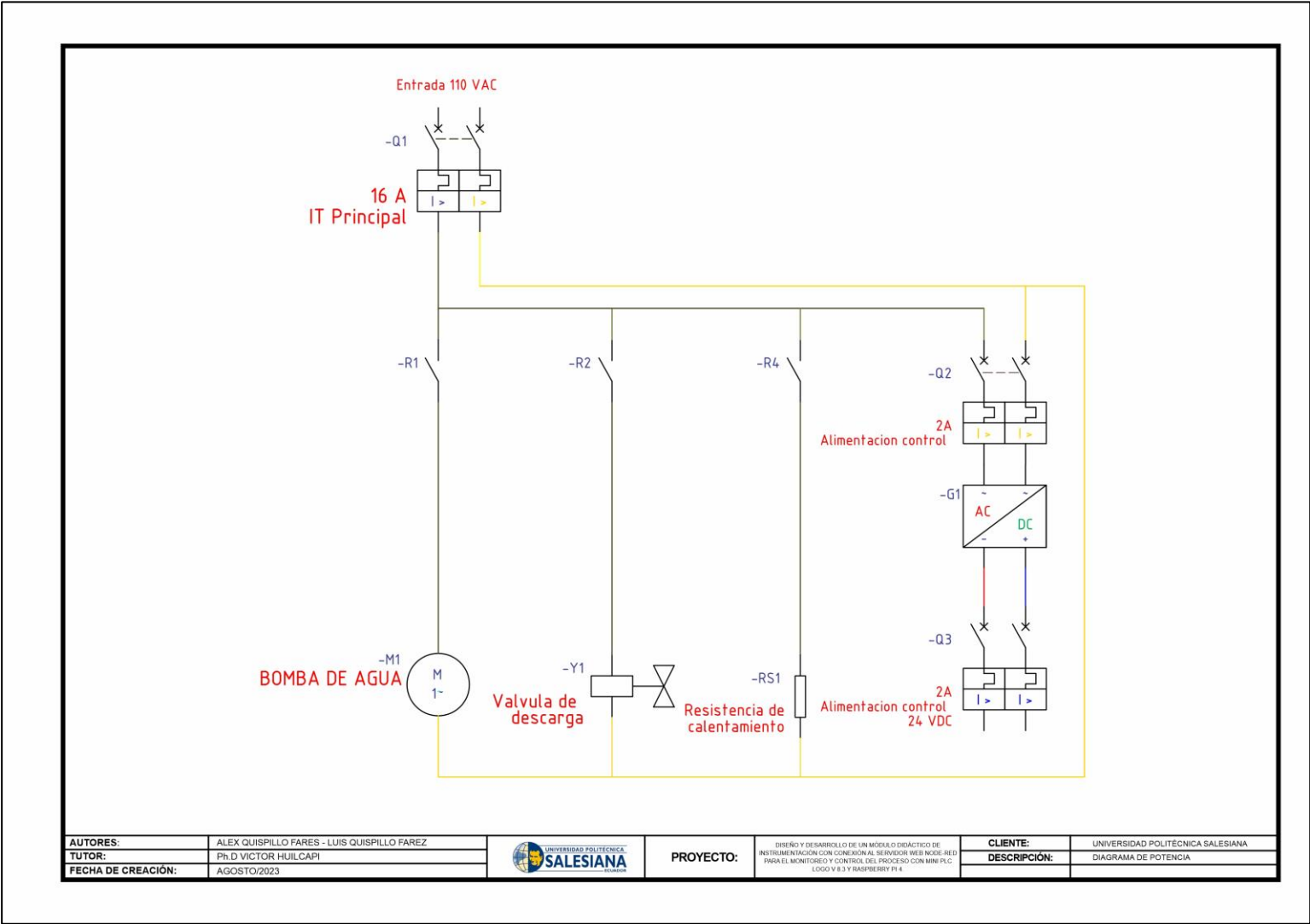


Figura 114. Diagrama de potencia



- **Ubicación de canaletas, rieles, borneras y demás elementos del tablero eléctrico**

Los elementos del tablero fueron distribuidos por **secciones**:

- Protección
- Control
- Accionamiento
- Interconexión en borneras

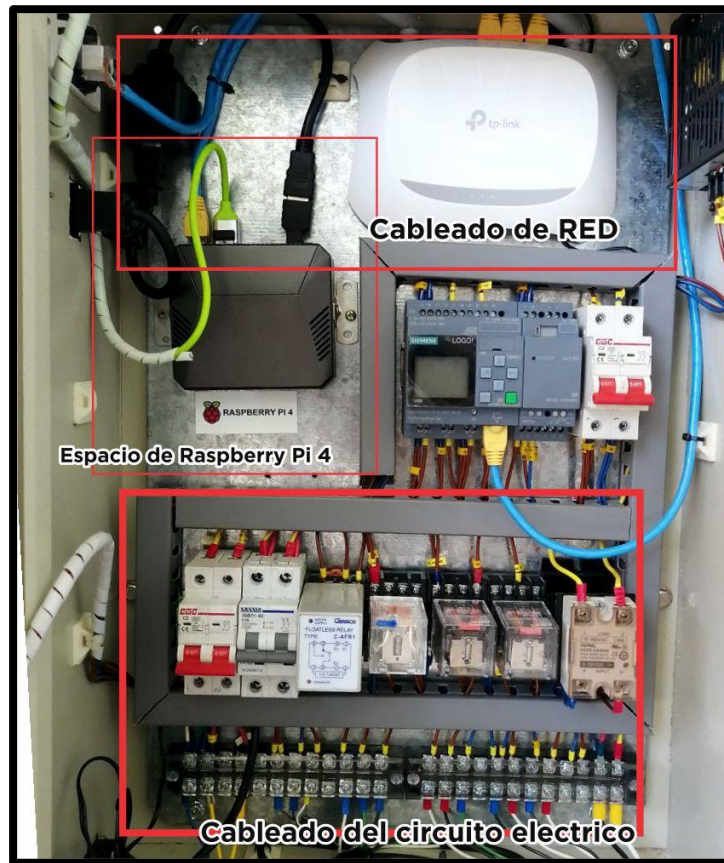
El relé de estado sólido fue montado sobre un disipador de calor, figura 115, para que sus semiconductores internos no sufran daños debido al calentamiento generado, durante su funcionamiento.

Figura 115. Disipador y Relé de estado solido



El cableado de red y el cableado del circuito eléctrico están separados, además el **Raspberry Pi 4** que fue instalado en el tablero, se proporcionó un espacio considerable entre los demás elementos para disipación del calor como se ilustra en la figura 116.

Figura 116. Distribución en el gabinete del proyecto



- **Etiquetado y rotulado de conexiones:**

El cableado de potencia utiliza en su identificación las siguientes letras

- L: Línea 1
- N: Línea 2

El cableado de control utiliza en su identificación las siguientes letras y números como se ilustra en la tabla 14.

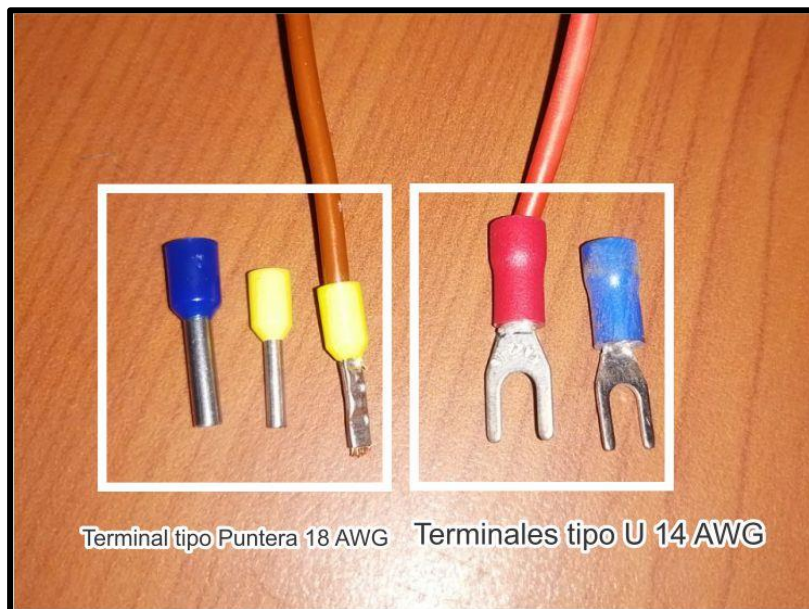
Tabla 14. identificación de letras y números

24: Línea de 24 VDC	5: Salida digital Q1
0: Línea negativa 0VDC	6: Salida digital Q2
1: Entrada digital I1	7: Salida digital Q3
2: Entrada digital I2	8: Salida digital Q4
3: Entrada digital I3	100: Terminal 1 RTD
4: Entrada analógica conectada en I7	101: Terminal 2 RTD
	102: Terminal 3 RTD

- **Utilización de terminales:**

En el cableado del circuito de control se utilizaron terminales de tipo puntera 18 AWG y el circuito de potencia terminales de tipo U 14 AWG, como se ilustra en la figura 117.

Figura 117. Tipos de terminales



## 5.2 Resolución teórico y práctico del planteamiento # 2:

El diagrama P&ID permite interpretar las conexiones hidráulicas y también los instrumentos de proceso utilizados en el módulo. Este diagrama se respalda con la norma ISA 5.1 utilizada en la industria de automatización y control, para estandarizar y proporcionar identificación a cada instrumentos y sistemas en proceso, que está vigente desde al año 2009, tabla 15.

Tabla 15. Tabla ISA-S5.1

	PRIMERA LETRA [4]		LETRAS SUCESIVAS [3]		
	Variable medida o inicial	Letra de modificación	Lectura o función de lectura pasiva	Función de salida	Letra de modificación
<b>A</b>	Análisis (3, 19)		Alarma		
<b>B</b>	Quemado, combustión		Libre (1)	Libre (1)	Libre (1)
<b>C</b>	Libre (1)			Control (1,3)	
<b>D</b>	Libre (1)	Diferencial (4)			
<b>E</b>	Temperatura (Te,m)		Sensor (Elemento primario)		
<b>F</b>	Caudal	Relación (4)			
<b>G</b>	Libre (1)		Viduo, Dispositivo visual (9)		
<b>H</b>	Manual				Alto (7, 13, 14)
<b>I</b>	Comenzar (electrica)		Indicar (10)		
<b>J</b>	Potencia	Explotación (7)			
<b>K</b>	Tiempo, programación tiempo	Variación de tiempo (4,2,1)		Estación de control (2,2)	
<b>L</b>	Nivel		Luz (1,1)		Bajo (7, 13, 14)
<b>M</b>	Libre (1)	Momentáneo (4)			Medio, Intermedio (7, 13)
<b>N</b>	Libre (1)		Libre (1)	Libre (1)	Libre (1)
<b>O</b>	Libre (1)		Orificio, Restricción		
<b>P</b>	Presión, Vacío		Punto (Ensayo) Conexión		
<b>Q</b>	Cantidad	Integral, Totalizar (4)			
<b>R</b>	Radiación		Registro (1,7)		
<b>S</b>	Velocidad, Frecuencia	Seguridad (8)		Interruptor (1,3)	
<b>T</b>	Temperatura			Transmisión (1,8)	
<b>U</b>	Multivariable (6)		Multifunción (1,2)	Multifunción (1,2)	Multifunción (1,2)
<b>V</b>	Vibración, Análisis mecánico (19)			Válvula, Regulador tipo, Prensado (1,3)	
<b>W</b>	Peso, fuerza		Válvula, Sonda		
<b>X</b>	sin clasificar (2)	Eje X	sin clasificar (2)	sin clasificar (2)	sin clasificar (2)
<b>Y</b>	Evento, Estado o Presencia (20)	Eje Y		Relé, Cálculo, Conversión (1,3, 14, 18)	
<b>Z</b>	Posición, Dimensión	Eje Z		Motor, Actuador, Elemento final de control sin clasificar	

Fuente: (Sole, 2010)

En el diagrama P&ID del módulo didáctico se observa el flujo de proceso, el tipo de señales que se manejan a través de cada uno de los instrumentos, como también su ubicación.

Las líneas que representan las tuberías especifican los diámetros a utilizar, así como cada uno de los accesorios, esto es de gran utilidad al momento de ensamblar.

El diagrama incluye dos tablas, la primera, la tabla donde se indica el significado de cada una de las letras, según su ubicación.



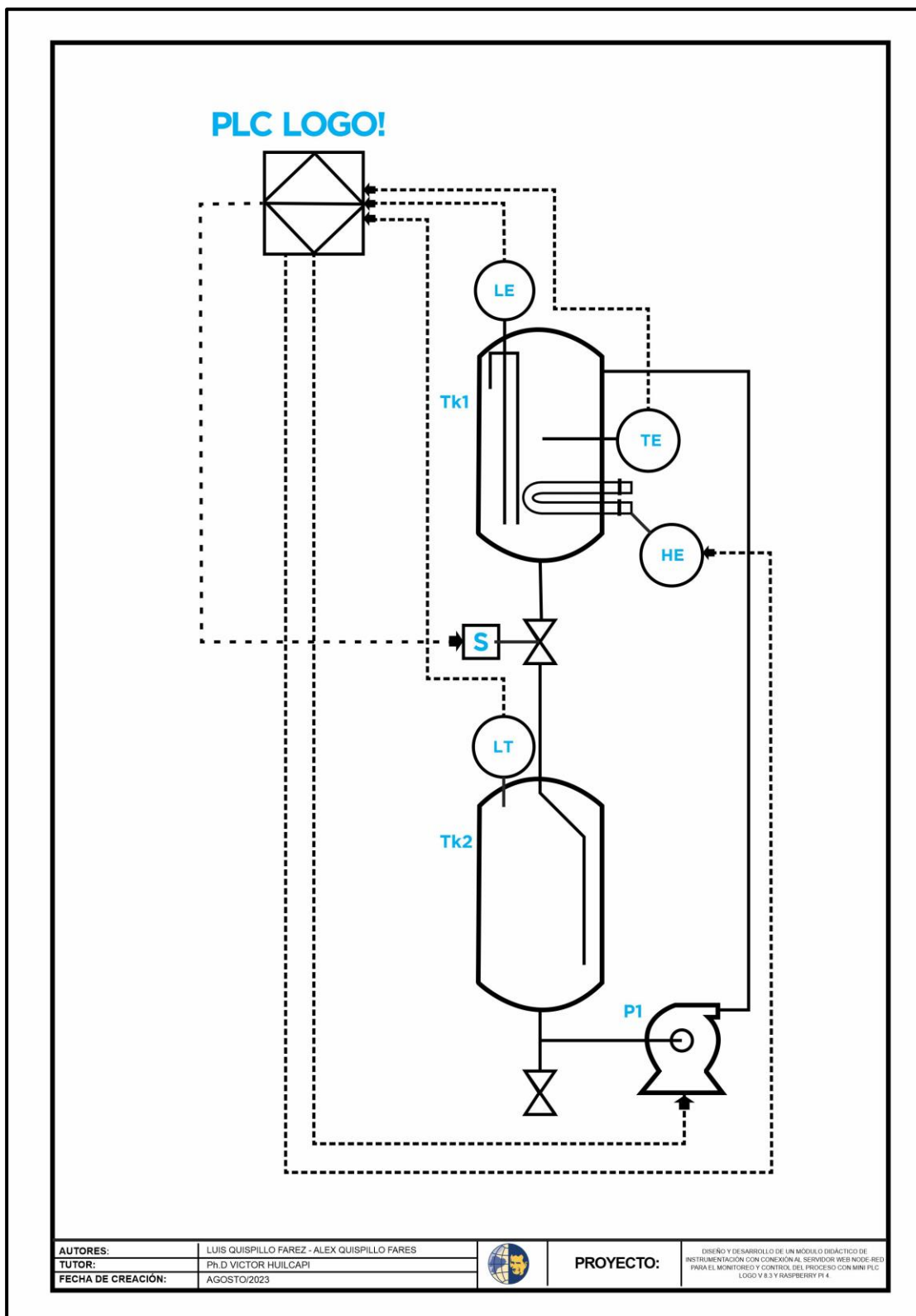
En la tabla 16 muestra un listado de partes, con su respectiva nomenclatura, entre ellas: instrumentos de proceso, controlador, actuadores y reservorios.

Tabla 16. Nomenclatura del módulo didáctico

<b>LISTA DE PARTES</b>	
<b>TK1</b>	Tanque superior
<b>TK2</b>	Tanque inferior
<b>P1</b>	Bomba de agua
<b>LT</b>	Transmisor de nivel ultrasónico
<b>LE</b>	Medidor de nivel con Electrosonda
<b>TE</b>	Transmisor de temperatura
<b>HE</b>	Resistencia de calentamiento
<b>S</b>	Electroválvula solenoide
<b>PLC LOGO!</b>	Controlador del módulo

A continuación, se observa en el diagrama de tuberías e instrumentación en el módulo didáctico, se ilustra en la *figura 118*.

Figura 118. Diagrama P&ID del módulo didáctico



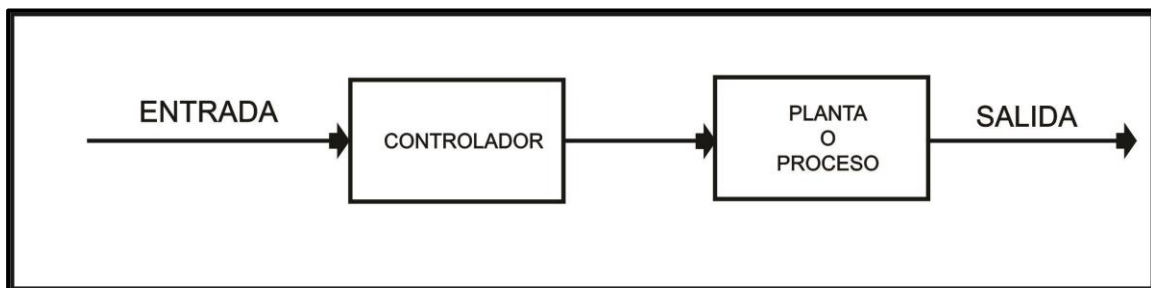
### 5.3 Resolución teórica del planteamiento # 3:

Existen dos tipos de lazos para el control de variables de proceso.

- **Control de lazo abierto:**

En este tipo de control no se cuenta con una retroalimentación de un instrumento para el control de una variable de proceso, por ejemplo, en el módulo se podría realizar un control de llenado de un tanque, tomando como referencia el caudal de la bomba, para establecer un nivel a través de temporización. Este tipo de control, *figura 119* tiene como desventaja, el no disponer de una medición como tal de la variable de proceso y se vuelve dependiente de la observación del operador.

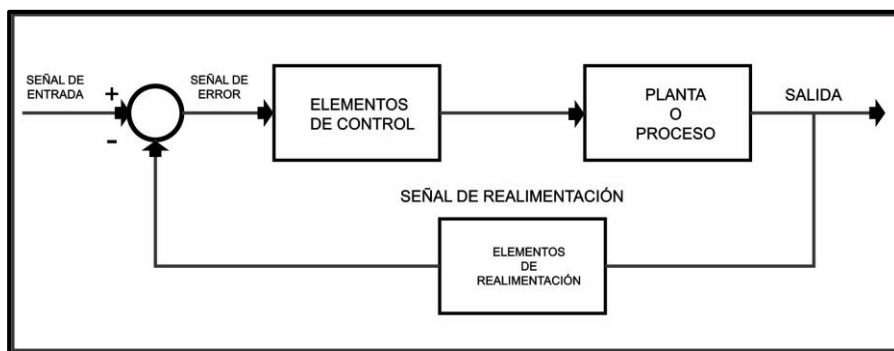
Figura 119. Control de lazo abierto



- **Control de lazo cerrado:**

El control de lazo cerrado utiliza una variable manipulada conocida como consigna o set point, que es el valor al cual quiero llevar a mi variable de proceso. Adicionalmente, se cuenta con una retroalimentación del instrumento de proceso, para poder generar un comportamiento en el actuador, de tal forma que no se rebase en exceso el límite configurado como consigna, ni tampoco existan tiempos muertos donde la variable tenga valores muy inferiores por debajo de la consigna, luego de haber sido esta alcanzada inicialmente, *Figura 120*.

Figura 120. Control de lazo cerrado



- **Lazos de control en el proceso**

Las variables de salida o variables controladas son aquellas que definen el número de lazos de control posibles dentro del proceso, en este caso se define los siguientes lazos de control que van a ser utilizados en las practicas posteriores, como se ilustra en la tabla 17.

Tabla 17. análisis de lazo de control del módulo didáctico

Lazo de control 100	Control de bomba de agua a través de instrumentos de nivel en tanque superior (nivel máximo y mínimo) e inferior (ultrasonido).
Lazo de control 101	Control de temperatura de resistencia de calentamiento, utilizando como retroalimentación una termorresistencia RTD PT100 y adicionalmente, dependiente del nivel en el tanque superior.
Lazo de control 102	Descarga de tanque superior
Lazo de control 103	Descarga de tanque inferior
Lazo de control 104	Control por recirculación de agua caliente donde intervienen la bomba de agua, resistencia de calentamiento. Como retroalimentación se utiliza la termorresistencia RTD PT100 y la confirmación del nivel en el tanque inferior

## 5.4 Simulación y desarrollo del módulo didáctico

- **Simulación de las prácticas correspondiente al módulo didáctico**

Las respectivas prácticas del módulo didáctico fueron realizadas desde el montaje de la habitación de unos de los integrantes del proyecto de vinculación, por lo tanto, se fueron trabajando desde la programación en LOGO! Soft Comfort V8.3 y el entorno de trabajo de Node-Red de la Raspberry Pi 4, como se muestra en la figura 121.

Figura 121. Montaje del tablero de control para desarrollo de las practicas



- **Diseño estructural del módulo didáctico**

El desarrollo del diseño se realizó en SOLIDWORKS como se ilustra en la *figura 122* y *figura 123*, para desarrollar la idea de su construcción para así evitar gastos en ajustes de los componentes a instalar, de esa manera se trabajó en conjunto con mi compañero en la realización estructural tomando en cuenta el tamaño y peso del gabinete, además las posiciones de cada actuador y sensores que estaría instalado en el módulo.

Figura 122. Tablero eléctrico 3D

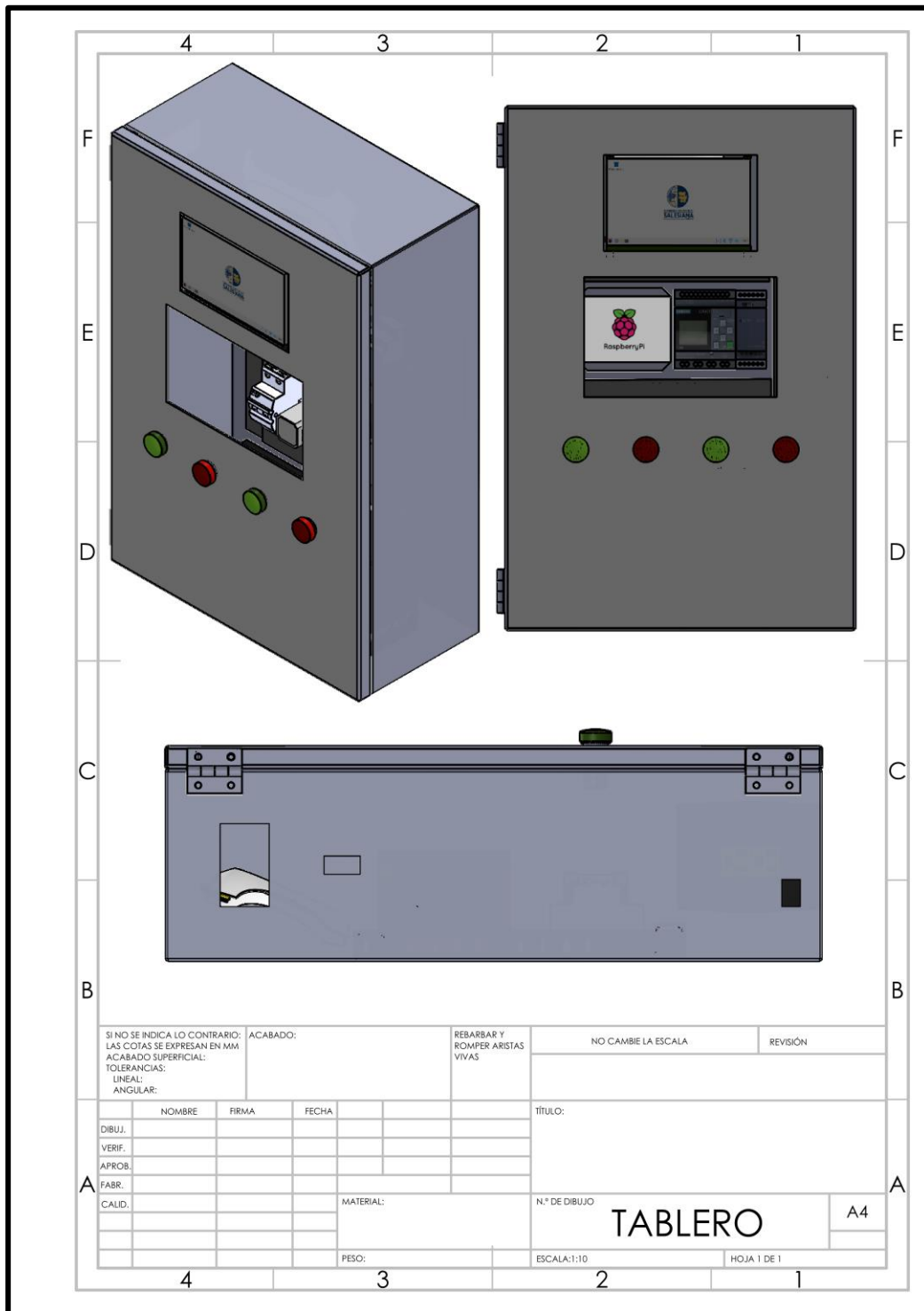
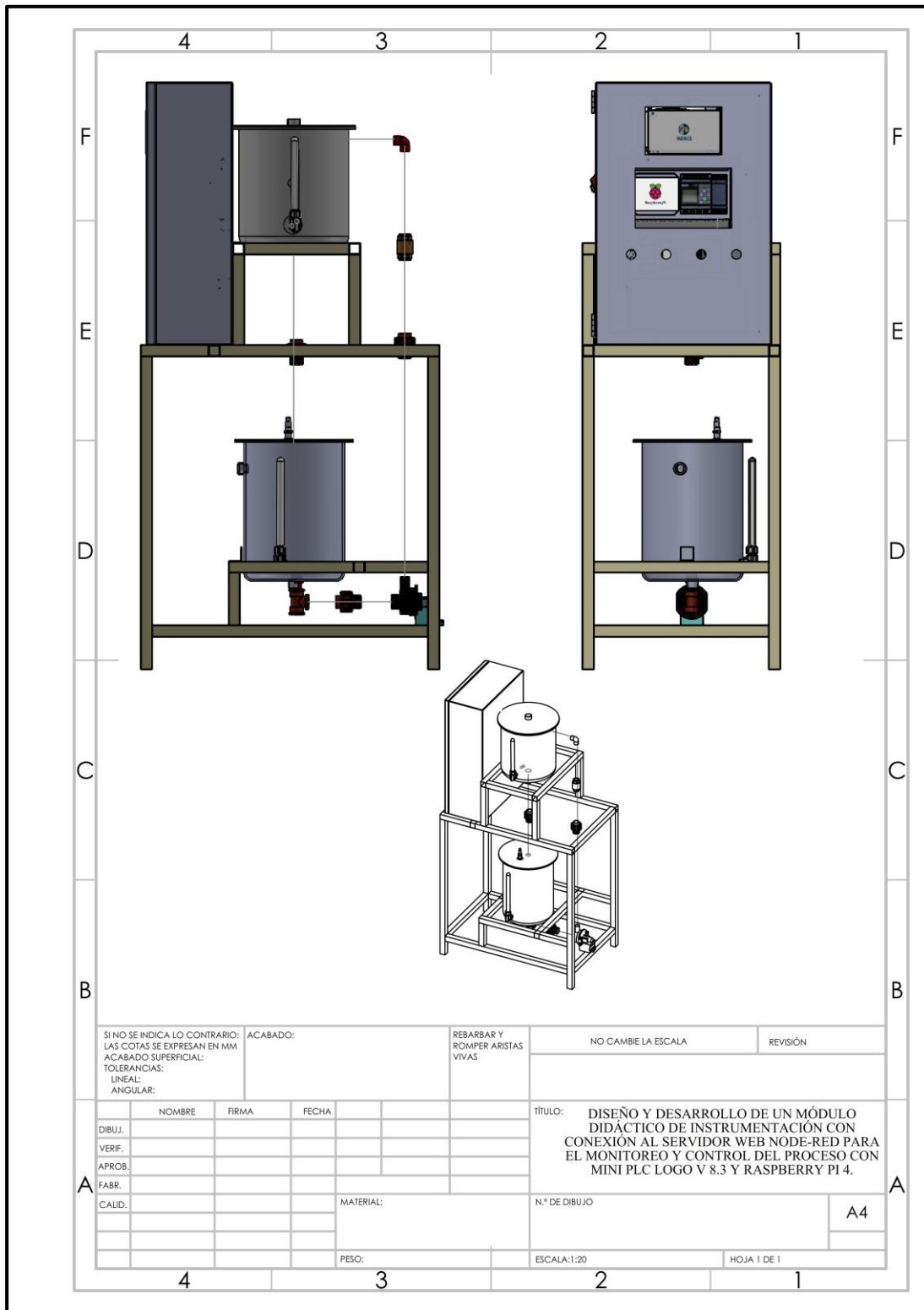


Figura 123. Diseño demodulo 3D



Como se muestra en *figura 124* es el módulo didáctico ya implementado desde el inicio de todo el proceso de este libro.

Figura 124. Modulo didáctico completo





## CONCLUSIÓN

La realización de este proyecto resulto algo desafiante en cuestión de presupuesto y desarrollo en la construcción y distribución del gabinete con sus equipos, el diseño de toda la parte metalúrgica con la idea de que el espacio que ocupe no sea tan inmenso, sino, algo más compacto y fácil de manipular con el que el estudiante tenga el interés de aprender y ampliar sus conocimientos de la carrera de ingeniería en electrónica.

Asimismo, tanto la programación visual en el editor de flujo Node-RED como el lenguaje de programación FUP de LOGO! Soft Comfort V8.3, fue todo un reto aprender algo nuevo sobre tecnología IoT e Industria 4.0 así mismo, recordar, estudiar y aplicar las herramientas utilizadas en los años de carrera como estudiantes, se sabe que la tecnología avanza de una manera sorprendente. Como estudiantes y futuros ingenieros se debe estar en constante aprendizaje ya sea siendo autodidacta, viendo tutoriales, asistiendo seminarios, escuchando consejos de otros colegas de la profesión, en fin, se tiene presente que nunca se deja de aprender en vida y así mismo es válido dar ese conocimiento lo que fue dado en alguna ocasión cuando no se sabía nada.

Dicho esto, se anima a que puedan tomar este proyecto didáctico enfocado a IoT e Industria 4.0 a realizar prácticas que se encuentra en este libro, y si se desarrolla nuevas prácticas bienvenido sea para la expansión del proyecto.

## RECOMENDACIÓN

- Tomar en cuenta que el valor de control máximo para la simulación de temperatura es de 0 – 50°C, por lo tanto, este valor no debe exceder para evitar presión de vapor ya que el diseño del tanque de acero inoxidable no tiene desfogue de vapores.
- Al realizar la programación visual en el editor de flujo Node-RED, tomare en cuenta que cada nodo puesto en el editor tiene una finalidad de conexión con otro nodo, por lo tanto al utilizar nodos que pertenece al dashboard deben agruparlos dentro del grupo en que van a estar los widgets en la interfaz, por lo que se recomienda antes, crear la tabla con sus diferentes grupos.
- SENSOR ULTRASONIDO utilizado en esta práctica, según su ficha técnica (*datasheet*) debe tener una distancia entre el material a medir en el tanque en este caso el agua un máximo de 50 mm para no tener una medición errónea que lo envía a cero, tomar en cuenta esta recomendación.
- RASPBERRY PI 4 no debe estar expuesto a ambiente húmedos, polvo, altas temperaturas. Por lo tanto, se sugiere que este ambiente en un acondicionado para no tener problema en el hardware, propiamente en los laboratorios de la UPS dispone de un ambiente apropiado para módulos didácticos. Así mismo, su alimentación debe estar antes conectado a un suministro respaldo y protección con el fin de evitar caída de tensión y variación de corriente en el sistema eléctrico.
- RASPBERRY PI 4 no debe ser reemplazado por cualquier fuente de alimentación de 5V, leer ficha técnica expuesto en este libro.
- Es recomendable que el encargado de los laboratorios de la UPS le facilite un puerto de acceso a internet, ya que están configuradas las IP y máscara de subred.
- Para el uso del puerto HDMI del módulo es necesario conectar desde el interior al puerto de doble vía.
- Dentro del tablero se deja un Relé conectado a una luz piloto enlazado con Q0.3 para simulaciones en cuestión.
- Para hacer uso de la Raspberry pi 4 es necesario abrir el tablero y encender el UPS y pulsar el botón de encender, si por algún motivo queda “congelado” el sistema operativo, con tener presionado el botón de encendido por 4s se apaga.

- Gestionar mantenimiento a los cables ETHERNET del módulo por cuestión de tiempo.
- Tomar en cuenta que dentro del tanque inferior está conectado una tubería para la caída libre del líquido por cuestión de medición del sensor ultrasonido.
- Para un mejor desenvolvimiento en las prácticas dentro del entorno de trabajo de Node-RED es importante conectar un mouse y teclado inalámbrico en la Raspberry Pi 4 o enlazar la IP que indica **RealVNC Server** desde un navegador que esté conectado a la misma red del módulo.
- La configuración de pantalla está dimensionada en 1280x720px para una mejor visualización del sistema operativo.

## BIBLIOGRAFÍA

- Areny, R. P. (2005). *Adquisición y distribución de señales*. Barcelona: Boixareu .
- AUTONIC. (2022). *AUTONIC*. Obtenido de <https://www.autonics.com/model/A1200000002>
- AUTRACEN. (julio de 2022). Obtenido de <https://www.autracen.com/blog/blog-autracen-1/estructura-de-plc-51>
- Baéz, S., & Hernandez, M. E. (2020). *Bomba de agua*. Universidad Nacional de Colombia, Manizales. Obtenido de <https://es.scribd.com/document/462478211/Bomba-de-agua>
- Belmonte, L. P. (2018). *Comunicaciones industriales y WinCC*. Marcombo.
- Brunete, A., San Segundo, P., & Herrero, R. (2020). En *Introducción a la automatización industrial*. Madrid. Obtenido de [https://bookdown.org/alberto\\_brunete/intro\\_automatica/sensores-industriales.html](https://bookdown.org/alberto_brunete/intro_automatica/sensores-industriales.html)
- Daneri, P. (junio de 2008). *scribd*. Obtenido de <https://es.scribd.com/document/357932319/Plc-FORZA>, t. p. (2022). Manual de usuario.
- Gombau, C. (29 de mayo de 2020). EL PC MÁS ECONÓMICO DEL MUNDO RASPBERRY PI 4. *Computer Hoy*. Obtenido de <https://es.scribd.com/article/475022110/El-Pc-Mas-Economico-Del-Mundo-Raspberry-Pi-4>
- Hermann, M., Pentek, T., & Otto, B. (2015). Design Principles for Industrie 4.0 Scenarios. *Working Paper*. Recuperado el Octubre de 2021, de [https://www.researchgate.net/publication/307864150\\_Design\\_Principles\\_for\\_Industrie\\_4\\_0\\_Scenarios\\_A\\_Literature\\_Review](https://www.researchgate.net/publication/307864150_Design_Principles_for_Industrie_4_0_Scenarios_A_Literature_Review)
- I+D *Electronica* . (s.f.). Obtenido de <https://www.didacticaselectronicas.com/index.php/elementos-electromecanicos/relays-1/estado-solido/rel%C3%A9-de-estado-s%C3%B3lido,-480vac-100a-xssr-da4875w6-relevos-relevadores-relay-reles-relays-de-estado-solido-xurui-detail>
- Kranz, M. (2017). *INTERNET OF THINGS: Construye nuevos modelos de negocio*. (J. M. Petisco, Trad.) LID. Obtenido de <https://es.scribd.com/book/361918975/INTERNET-OF-THINGS-Construye-nuevos-modelos-de-negocio>
- Morales, A., Alvarez, M., & Mircha, J. (2016). *Manual de NodeJs*. Obtenido de <https://es.scribd.com/document/373876519/NodeJS>

Node-RED. (agosto de 2022). *nodered.org*. Obtenido de <https://nodered.org/>

Ogata, K. (2010). *Ingeniería de control moderna* (quinta ed.). Madrid: PEARSON.

Ortiz, E. J. (2019). *NODE RED RASPBERRY Y ESP8266*. UNIVERSIDAD PRIVADA FRANZ TAMAYO, El Alto. Obtenido de <https://es.scribd.com/document/440334316/NODE-RED-RASPBERRY-Y-ESP8266-docx>

Pixsys Electronics. (Agosto de 2022). *www.pixsys.net*. Obtenido de <https://www.pixsys.net/it/prodotti/sensori-e-trasmittitori/sensori-di-temperatura/pt100-%c3%98-6-testa%c2%a0din%c2%a0-easy-up>

*raspberrypi.com/software*. (2022). Obtenido de <https://www.raspberrypi.com/software/>

*redes23*. (2013). Obtenido de <https://redes23.webnode.es/segundo-parcial/>

Richard, D. (2022). *scribd*. Obtenido de <https://es.scribd.com/read/569007231/Raspberry-Pi-4-For-Beginners-2022-A-Practical-Guide-to-Setup-Programming-Theory-and-Building-Innovative-Projects#>

SIEMENS. (06 de marzo de 2020). *infoPLC*. Obtenido de <https://www.infopl.net/noticias/item/107569-siemens-logo-8-3-conecta-directamente-con-la-nube>

Sole. (2010). *Instrumentación industrial* (octava ed.). Mexico: Alfa y Omega.

TP-LINK. (2022). *TP-LINK*. Obtenido de <https://www.tp-link.com/ec/home-networking/wifi-router/tl-wr840n/>

Tumbaco, & Villa. (2018). *Diseño e implementación de una planta didáctica industrial con aplicaciones para el control de nivel en un reservorio, para el laboratorio de automatización*. Universidad politecnica salesana, Guayaquil.

Vera, & Humberto. (2015). *Diseño e implementación de un aplnta didactica industrial con aplicaciones para el control e nivel en un reservorio, para el laboratorio de automatización*. Universidad Politecnica Salesiana , Guayaquil.