



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
ESCUELA NACIONAL DE ESTUDIOS SUPERIORES
UNIDAD JURIQUILLA
LICENCIATURA EN TECNOLOGÍA

ENLACE AL INTERNET DE LAS COSAS (IOT) DE UN SISTEMA DE MONITOREO Y CONTROL DE TEMPERATURA DE PASTEURIZACIÓN AGROINDUSTRIAL DE LECHE PARA PEQUEÑOS PRODUCTORES

**QUE PARA OBTENER EL GRADO DE:
LICENCIADO EN TECNOLOGÍA**

**PRESENTA:
RODRIGO DE BENAVENTE CARREÓN**

**DR. DOMINGO RANGEL MIRANDA
CENTRO DE FÍSICA APLICADA Y TECNOLOGÍA AVANZADA**

**DR. ULISES OLIVARES PINTO, ENES-JURIQUILLA
DRA. DIANA MARTÍNEZ CASILLAS, ENES-JURIQUILLA
DR. JESÚS EMMANUEL SOLÍS PÉREZ, ENES-JURIQUILLA
DRA. IRIS MARIELA LÓPEZ BAUTISTA, CENAM**

JURIQUILLA, QUERÉTARO, MÉXICO, NOVIEMBRE 2023



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

Agradezco a mi tutor por su orientación y motivación constante durante la última etapa de la licenciatura para seguir mis objetivos profesionales y por la oportunidad que me ofreció para desarrollar este proyecto. Me benefició enormemente las habilidades y conocimientos que adquirí en el laboratorio, con el instrumental y herramientas que puso a disposición.

Por ello también agradezco la oportunidad que me otorgó la Universidad Nacional Autónoma de México por estudiar en una de sus escuelas, aprovechar sus instalaciones para tomar las clases y las instalaciones del Centro de Física Aplicada y Tecnología Avanzada para llevar a cabo las pruebas y los proyectos en el laboratorio de Instrumentación y Desarrollo.

Les doy gracias a todos mis profesores, tanto del CFATA como de la ENES, por la diversidad de perspectivas, conocimientos y consejos que me brindaron, que sé que serán de gran importancia para mi desarrollo profesional. Agradezco al personal académico y administrativo por su asistencia en los distintos procesos imprescindibles para la conclusión de mis estudios. Agradezco el apoyo y la cooperación que me prestaron mis compañeros de generación, que fue cada vez más reconocida conforme nuestro grupo se redujo.

Agradezco al MVZ Ramón González Téllez Girón y al IAZ José Carlos González Rivera por su consentimiento para conocer el proceso de pasteurización y su rancho ganadero “El Rincón”, y por la leche brindada para la realización de las pruebas del sistema.

Finalmente, estoy profundamente agradecido por el incentivo, ayuda y atención de mi familia y mis amigos. Aunque continuamente tengo la fuerza, el enfoque y la tenacidad de lograr mis objetivos, en momentos de duda y adversidad su soporte me representa una gran diferencia.

DEDICATORIA

Le dedico este proyecto de tesis especialmente a mis padres, mi hermano, mis amigos más cercanos y mi tutor por brindarme su apoyo tanto directamente como indirectamente en el desarrollo del proyecto. Me aportaron herramientas, recursos, consejos y su tiempo, que si en ocasiones no fueron indispensables fueron de gran ayuda. Y no solamente en este proyecto, sino que también me apoyaron durante toda la licenciatura, por lo que, es para mí su reconocimiento sumamente importante.

ÍNDICE

AGRADECIMIENTOS	2
DEDICATORIA	3
ÍNDICE	4
ÍNDICE DE FIGURAS	9
ÍNDICE DE TABLAS	13
RESUMEN	15
ABSTRACT	16
1. INTRODUCCIÓN	17
1.1. PLANTEAMIENTO DEL PROBLEMA	18
1.1.1. Problemática Nacional	18
1.1.2. Problemática Local	19
1.2. ESTADO DEL ARTE	19
1.2.1. Tecnologías de la información aplicadas a producción agroindustrial	19
1.2.2. Controladores de temperatura en procesos industriales en la industria agroalimentaria de lácteos	19
1.2.3. Sistemas de control de temperatura PID en sistema de pasteurización	20
1.2.4. Sistemas de pasteurización industrial monitorizados en el Internet de las Cosas (IoT)	20
1.3. JUSTIFICACIÓN	21
1.4. HIPÓTESIS	22
1.5. OBJETIVOS	22
1.5.1. Objetivo General	22

1.5.2. Objetivos Particulares	22
2. MARCO TEÓRICO	23
2.1. EL INTERNET DE LAS COSAS (IOT)	23
2.1.1. Estructura IoT	24
2.1.2. Plataformas IoT	25
2.1.3. Requisitos de una plataforma IoT	26
2.1.4. MCU y MPU	27
2.1.5. MCU y Sistema en chip (SoC)	27
2.1.6. MPU y Computadoras de placa única (SBC)	28
2.1.7. SoC y SBC	29
2.1.8. Categorías de plataformas IoT	32
2.2. CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO (PID)	33
2.2.1. Estructura general del control PID	33
2.2.2. Método de ajuste Ziegler-Nichols	36
2.3. CONTROL ANALÓGICO INTEGRADO PAC	37
2.3.1. Comunicación RS232 y RS485	38
2.4. EL PROCESO DE PASTEURIZACIÓN	40
2.4.1. Principio	40
2.4.2. Tipos de pasteurización	41
2.4.3. Efecto de la temperatura en las bacterias	46
2.5. ESTADO ACTUAL DE SISTEMAS DE PASTEURIZACIÓN VAT	49
2.5.1. Sistemas de pasteurización actuales	49
2.5.2. Propuestas de sistemas de pasteurización en México y Latinoamérica para pequeños productores	50

3. METODOLOGÍA: PROPUESTA DEL PROTOTIPO DE ACTUALIZACIÓN MEDIANTE IOT	52
3.1. INFRAESTRUCTURA	55
3.1.1. Arquitectura del sistema de control	55
3.1.2. Arquitectura del sistema de monitoreo	61
3.1.3. Funcionamiento del sistema completo	66
3.2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA	69
3.2.1. Controlador PID	69
3.2.2. Tarjeta electrónica: Conexión de la tarjeta IoT al Max31865	70
3.3. DESARROLLO EXPERIMENTAL	72
3.3.1. Control de temperatura: PID	74
3.3.2. Adquisición de datos acondicionados	75
4. RESULTADOS Y DISCUSIÓN	76
4.1. DEFINICIÓN DE PARÁMETROS DEL CONTROLADOR PID	76
4.2. CONSTRUCCIÓN DE INFRAESTRUCTURA	81
4.3. COMPARATIVA Y SELECCIÓN DE HARDWARE IOT	83
4.4. COSTOS DEL SISTEMA	97
4.5. PARTICLE PHOTON	98
4.5.1. Photon	99
4.5.2. Software - Entorno de desarrollo (programación)	100
4.5.3. Centro de comando	100
4.5.4. Modelos de servicio en la nube	101
4.5.5. Planes de servicio de Particle Cloud	105
4.5.6. Registro de datos en la Particle Cloud	106

4.5.7.	Conexión a la nube	107
4.5.8.	Adquisición y procesamiento de los datos	111
4.5.9.	Procesamiento de datos	112
4.6.	INVESTIGACIÓN DEL SOFTWARE IOT Y ELABORACIÓN DE LA INTERFAZ DE MONITOREO	115
4.6.1.	Plotly	115
4.6.2.	Dash	115
4.6.3.	Librerías de componentes de código abierto	116
4.6.4.	Funciones reactivas	118
4.6.5.	Aplicaciones	118
4.6.6.	Elaboración de la interfaz	119
4.6.7.	Personalización y ajuste de la interfaz gráfica	123
4.6.8.	Configuración de parámetros	134
4.7.	CARACTERIZACIÓN DEL SISTEMA - PRUEBAS DEL SISTEMA CON AGUA	135
4.7.1.	VAT - 12L sin agitación	135
4.7.2.	VAT - 12L con agitación	138
4.7.3.	VAT - 16L sin agitación	140
4.7.4.	VAT - 16L con agitación	142
4.7.5.	HTST - 12L sin agitación	143
4.7.6.	HTST - 12L con agitación	145
4.7.7.	HTST - 16L sin agitación	146
4.7.8.	HTST - 16L con agitación	147
4.8.	PRUEBAS FINALES - PRUEBAS DEL SISTEMA CON LECHE	148

4.8.1. Prueba sin agitación	148
4.8.2. Prueba con agitación	150
4.8.3. Análisis microbiológico	152
5. CONCLUSIONES	157
6. TRABAJO FUTURO	161
REFERENCIAS	162
ANEXOS	174
ANEXO 1: DATOS TÉCNICOS DE LOS DISPOSITIVOS	174
1.1. Características técnicas del Mezclador de propósito general Talboys	174
1.2. Especificaciones del adaptador AC-1000 de Desin Instruments	175
1.3. Controlador PID LS-3400	177
1.4. Características técnicas del Particle Photon	182
ANEXO 2: DISEÑO Y CONSTRUCCIÓN DE LA INFRAESTRUCTURA	185
ANEXO 3: CÓDIGOS DE PROGRAMACIÓN	187
ANEXO 4: INFORMES DE RESULTADOS DE LAS PRUEBAS MICROBIOLÓGICAS	189

ÍNDICE DE FIGURAS

Figura 2.1. Diagrama de bloques de un SoC. Editado de [34].	29
Figura 2.2. Diagrama de bloques de una SBC. Editado de [34].	30
Figura 2.3. Diagrama del funcionamiento general de un control PID. Imagen tomada de [40].	33
Figura 2.4. Un controlador PID toma control de acción basado en errores de control del pasado, presente y futuro. Modificado de [41].	34
Figura 2.5. Efecto de la temperatura sobre el crecimiento microbiano. Modificada de [51].	46
Figura 3.1. Diagrama general de la metodología del prototipo de pasteurización IoT.	54
Figura 3.2. Prototipo de pasteurización VAT para procesar 50 litros de leche por hora. Tomado de [62].	55
Figura 3.3. Esquema de conexión del sistema de control de temperatura de la leche.	57
Figura 3.4. Imagen representativa del calentador de 1500 Watts de la empresa Hanaiette. Tomado de [69].	59
Figura 3.5. Imagen representativa del mezclador de servicio ligero Talboys modelo 102. Tomado de [70].	60
Figura 3.6. Imagen representativa del impulsor dispersante de disco. Tomado de [71].	61
Figura 3.7. Esquema del conexionado del sistema de monitoreo de temperatura de la leche.	62
Figura 3.8. Vista frontal del Max31865. Tomado de [73].	63
Figura 3.9. Diagrama de arquitectura del sistema de pasteurización completo.	66
Figura 3.10. Diagrama del funcionamiento del control PID en el proceso de pasteurización.	66
Figura 3.11. Configuración física del controlador PID. A: Controlador de potencia, B: Adaptador de comunicación, C: Controlador PAC PID.	69
Figura 3.12. Conexión del sistema de monitoreo, vista superior. A: Tarjeta IoT, B: Convertidor Max31865.	70

Figura 3.13. Conexión del sistema de monitoreo, vista lateral alzada. A: Tarjeta IoT, B: Convertidor Max31865, C: Fuente de alimentación regulada.	71
Figura 3.14. Conexión del sistema de monitoreo, vista lateral alzada ampliada. C: Fuente de alimentación regulada, D: Relé de estado sólido.	71
Figura 3.15. Captura de las ventanas de simulación y comunicaciones de PROASIS DCS-Win Configuración.	75
Figura 4.1. Esquema de la configuración implementada en Loop-Win.	77
Figura 4.2. Captura de la interfaz de monitoreo del controlador PID.	80
Figura 4.3. Vista isométrica del modelo de la estructura de soporte propuesta para el sistema de pasteurización.	82
Figura 4.4. Fotografía de la estructura de soporte terminada con los instrumentos de medición y el agitador montados.	82
Figura 4.5. Esquema de los distintos servicios de la nube. Tomado de [88].	103
Figura 4.6. Captura de la IDE de Particle en su sección de archivos de código.	108
Figura 4.7. Captura de consola de la nube de Particle con eventos subidos por el Photon.	110
Figura 4.8. Diagrama de asignación de pines del Photon. Tomado de [91].	111
Figura 4.9. Programación del Photon en forma de diagrama de flujo.	114
Figura 4.10. Captura de la pestaña de tablero de gráficos de control de la aplicación Dash utilizada como plantilla.	121
Figura 4.11. Captura de la pestaña de configuración de especificaciones de la aplicación Dash utilizada como plantilla.	123
Figura 4.12. Captura de la pestaña del tablero de gráficos de control de la aplicación para el sistema de pasteurización.	124
Figura 4.13. Captura de la pestaña del tablero de configuración de especificaciones de la aplicación para el sistema de pasteurización.	125
Figura 4.14. Visualización de la interfaz de monitoreo en un proceso de pasteurización VAT.	126

Figura 4.15. Diagrama de flujo del código de la interfaz dentro de la definición de contenedores y funciones de procesamiento de información.	128
Figura 4.16. Diagrama de flujo del código de la interfaz dentro del llamado a las funciones reactivas.	132
Figura 4.17. Captura de la pestaña de configuración de especificaciones de la aplicación con los parámetros necesarios para la pasteurización VAT.	134
Figura 4.18. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 12 litros de agua sin agitación.	137
Figura 4.19. Captura de la interfaz de monitoreo del proceso VAT (63°C) con 12 litros de agua sin agitación.	137
Figura 4.20. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 12 litros de agua y con agitación.	139
Figura 4.21. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 16 litros de agua sin agitación.	141
Figura 4.22. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 16 litros de agua y con agitación.	142
Figura 4.23. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 12 litros de agua sin agitación.	144
Figura 4.24. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 12 litros de agua y con agitación.	145
Figura 4.25. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 16 litros de agua sin agitación.	146
Figura 4.26. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 16 litros de agua y con agitación.	147
Figura 4.27. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 16 litros de leche sin agitación.	149
Figura 4.28. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63 °C) con 16 litros de leche y con agitación.	151

Figura A.1. Imagen representativa del mezclador de propósito general Talboys modelo 102. Tomado de [70].	174
Figura A.2. Esquema de conexión del convertidor AC-1000 para enlazar un controlador de la serie LS-3000 con una computadora. Tomado de [67].	175
Figura A.3. Esquema de conexiones del convertidor AC-1000. Tomado de [67].	176
Figura A.4. Ejemplo de la visualización de los datos y medidas en la forma de distintos paneles de registro lógico y analógico.	179
Figura A.5. Ejemplo de una aplicación configurada mediante Loop-Win. Tomado de [43].	180
Figura A.6. Marcas de entradas del Particle Photon. Tomada de [91].	182
Figura A.7. Vista frontal de la infraestructura del sistema de medición y control de temperatura.	185
Figura A.8. Vista lateral de la infraestructura del sistema de medición y control de temperatura.	185
Figura A.9. Vista superior de la infraestructura del sistema de medición y control de temperatura.	186
Figura A.10. Dibujo de la infraestructura del sistema de medición y control de temperatura, acotado en milímetros.	186
Figura A.11. Informe de resultados de los análisis microbiológicos realizados en la leche sin pasteurizar.	189
Figura A.12. Informe de resultados de los análisis microbiológicos realizados en la leche pasteurizada.	191

ÍNDICE DE TABLAS

Tabla 2.1. Diferencia entre SoC y SBC. Editado de la información de [34-36].	31
Tabla 2.2. Pros y contras de la pasteurización VAT/LTLT.	42
Tabla 2.3. Pros y contras de la pasteurización tipo HTST con el uso de un pasteurizador.	43
Tabla 2.4. Pros y contras de la pasteurización tipo HTST con el uso de un intercambiador de calor.	44
Tabla 2.5. Pros y contras de la pasteurización de tipo UHT.	45
Tabla 3.1. Comparativa de propiedades térmicas del agua y la leche. Información tomada de [74-76].	72
Tabla 3.2. Comparativa de propiedades reológicas del agua y la leche. Información tomada de [74],[76].	73
Tabla 4.1. Cuadro comparativo de tarjetas de desarrollo IoT. Extraído de [27].	83
Tabla 4.2. Cuadro comparativo de plataformas IoT en términos de computación. Extraído de [24].	86
Tabla 4.3. Cuadro comparativo de plataformas IoT en términos de entornos de desarrollo y estándares de comunicación. Extraído de [24].	88
Tabla 4.4. Cuadro comparativo de plataformas IoT en términos de conectividad. Extraído de [24].	91
Tabla 4.5. Cuadro comparativo de plataformas de software IoT. Extraído de [24].	94
Tabla 4.6. Detalle de costos de materiales por sistema.	97
Tabla 4.7. Límites máximos de contenido microbiano para leche y derivados lácteos. Tomada de [2].	152
Tabla 4.8. Cuadro comparativo de los resultados de los análisis microbiológicos de la leche sin pasteurizar en relación con los límites máximos de contenido microbiano de la NOM-243-SSA1-2010.	154

Tabla 4.9. Cuadro comparativo de los resultados de los análisis microbiológicos de la leche pasteurizada en relación con los límites máximos de contenido microbiano de la NOM-243-SSA1-2010.	155
Tabla A.1. Especificaciones del mezclador de propósito general Talboys modelo 102. Tomado de [70].	174
Tabla A.2. Descripción de entradas del Particle Photon. Tomada de [91].	182

RESUMEN

ENLACE AL INTERNET DE LAS COSAS (IOT) DE UN SISTEMA DE MONITOREO Y CONTROL DE TEMPERATURA DE PASTEURIZACIÓN AGROINDUSTRIAL DE LECHE PARA PEQUEÑOS PRODUCTORES

En este proyecto de tesis se plantea la actualización de un controlador de temperatura LS3400 basada en la interconexión de éste al internet de las cosas (IoT) para procesos de pasteurización de leche de bovino. Este prototipo fue diseñado para la tecnificación de la producción de pequeños productores de leche, utilizando dispositivos de bajo costo. Se utilizó la tarjeta IoT de Particle, que consta básicamente de un microcontrolador con entradas analógicas y digitales, lo que permitió acondicionar los sensores de temperatura tipo Pt100. También se hizo uso de software libre de Particle como Dash y HTML para la interconexión y manejo de datos. Dicha plataforma de desarrollo (IDE Particle) con conexión inalámbrica vía WiFi permitió utilizar servicios de procesamiento en la nube sin costo para un determinado número de operaciones de datos. La actualización del sistema consistió en realizar el control de la temperatura de pasteurización tipo PID con el LS3400 y al mismo tiempo monitorear la temperatura de manera remota enlazando la tarjeta de Particle vía WiFi a la red del internet de las cosas. Así mismo se desarrolló una interfaz de usuario para visualizar la temperatura de pasteurización en tiempo real de manera remota en una computadora conectada al internet. Posteriormente se realizaron pruebas para diferentes volúmenes y tiempos de pasteurización tipo VAT en agua y leche, lográndose con ello tener un sistema actualizado de monitoreo de la temperatura de pasteurización conectado al internet de las cosas con un buen rendimiento y calidad en la pasteurización de la leche.

ABSTRACT

LINK TO THE INTERNET OF THINGS (IOT) OF A TEMPERATURE MONITORING AND CONTROL SYSTEM FOR AGROINDUSTRIAL MILK PASTEURIZATION FOR SMALL PRODUCERS

This thesis project proposes the upgrade of an LS3400 temperature controller based on its interconnection to the Internet of Things (IoT) for pasteurization processes of bovine milk. This prototype was designed for the technification of the production of small milk producers, using low-cost devices. Particle's IoT card was used, which basically consists of a microcontroller with analog and digital inputs, that allowed conditioning the Pt100 temperature sensors. Particle's free software, such as Dash and HTML, was also used for interconnection and data management. This development platform (IDE Particle) with wireless connection via WiFi allowed the use of cloud processing services at no cost for a certain number of data operations. The system upgrade consisted of performing PID pasteurization temperature control with the LS3400 and at the same time monitoring the temperature remotely by linking the Particle card via WiFi to the internet of things network. A user interface was also developed to visualize the pasteurization temperature in real time remotely on a computer connected to the internet. Subsequently, tests were performed for different volumes and times of VAT pasteurization in water and milk, thus achieving an updated system for monitoring the pasteurization temperature connected to the internet of things with good performance and quality in the pasteurization of milk.

1. INTRODUCCIÓN

Dentro de la producción de leche, una de las etapas principales es la pasteurización. Proceso con el que se eliminan microorganismos del lácteo posteriormente a la extracción, originalmente a través de un aumento de la temperatura. El calentamiento debe permitir conservar el valor nutritivo y las propiedades organolépticas de la leche y sus derivados, tales como el contenido de proteínas, sabor, etc. [1]. Por este motivo, el control de la temperatura de pasteurización es crucial para una buena producción de leche.

Los métodos principalmente empleados en los pequeños y medianos productores son el método Low Temperature Long Time (LTLT) y High Temperature Short Time (HTST). El primer método se refiere a “baja temperatura por mucho tiempo”, el punto de control es de 63 °C y se mantiene por 30 minutos. El segundo método indica “alta temperatura por poco tiempo”, cuyo punto de temperatura es de 72 °C y debe mantenerse por 15 segundos. En ambas técnicas sólo se admite una tolerancia de 1 °C en el punto de control [2].

En Querétaro se considera como pequeño productor o rudimentario a todo aquel que produce entre 5 mil y 10 mil litros de leche diario [3], cuenta con tecnologías e instalaciones básicas sin un proceso de ordeña automatizado [4]. Por otro lado, los medianos productores o semitecnificados producen más de 30 mil litros [3]. Operan con un nivel tecnológico intermedio, realizan la extracción de leche manualmente, no disponen de dispositivos de enfriamiento y mantienen al ganado en un sistema de gestión semiconfinado [4]. Desde 2018 se encuentran registrados más de 300 mil pequeños y medianos ganaderos dedicados a la producción de leche, de acuerdo con los datos de SAGARPA (Secretaría de Agricultura, Ganadería, Desarrollo Rural, Pesca y Alimentación) [5]. Los productores con procesos parcialmente tecnificados aportan el 20% de la producción total, y la contribución de productores rudimentarios es de alrededor del 30% [4].

Por lo anterior, en este proyecto de tesis se propone utilizar un controlador electrónico integrado que incluye un algoritmo Proporcional, Integral, Derivativo (PID) para el control de la temperatura y su interconexión con el Internet de las Cosas (IoT), para mantener una supervisión automatizada a distancia del proceso de pasteurización; y al mismo tiempo proporcionar información digital de uno o varios procesos de producción simultáneos al pequeño productor conectado a internet, con el propósito de proveer elementos relevantes para mantener una buena calidad de su producto y un aumento en su producción.

1.1. PLANTEAMIENTO DEL PROBLEMA

1.1.1. Problemática Nacional

De acuerdo con la Encuesta Nacional Agropecuaria (ENA) 2019 por parte de la INEGI de 2019, en México el 37.7% de las unidades de producción (UP) hace uso de tecnologías informáticas de comunicación en las actividades agropecuarias; por su parte, el uso de Internet se reportó en apenas en el 7.9% de ellas [6]. Debido a este déficit tecnológico en las Pequeñas y Medianas Empresas (PyMES) nacionales del sector primario agroindustrial, la industria de producción de leche también se ha visto afectada en su modernización, ya que, los pequeños productores no utilizan sistemas de medición, procesamiento y monitoreo de datos de sus procesos de producción en el Internet de las Cosas. La falta de estas tecnologías en los procesos representa un rezago en el desarrollo industrial y económico en el sector primario. Los instrumentos de medición y control aumentan la precisión y, por medio de las certificaciones correspondientes, la confiabilidad de los sistemas. Los instrumentos aportan un mayor nivel de exactitud y, debido a que pueden ser calibrados e incluidos en un proceso de automatización, replicabilidad en el proceso industrial. Aunque los productores cuentan con un método que les ha sido útil por mucho tiempo, sin estas tecnologías no pueden asegurar que su

proceso cumple con los estándares actuales. Asimismo, no hacer uso de tecnologías actuales en sus procesos, les resta competitividad en el mercado.

1.1.2. Problemática Local

Según fuentes de noticias estatales, una de las principales prioridades del estado de Querétaro es el desarrollo del sector productivo primario, donde el sector pecuario en general y el de la leche en particular es el de mayor producción [7]. En relación con el plan estatal de desarrollo, es igualmente prioridad transformar a Querétaro en un estado con base tecnológica de clase mundial empleando eficazmente las tecnologías de la información (TI) [8].

1.2. ESTADO DEL ARTE

1.2.1. Tecnologías de la información aplicadas a producción agroindustrial

En el procesamiento de alimentos se encuentran diferentes aplicaciones para el monitoreo de las variables del proceso, tal es el caso del sistema propuesto por Paladino et al. [9]. Esta implementación abarca la adquisición, gestión, incorporación y visualización de datos de proceso en plantas micro y pequeñas. Otra aplicación es la propuesta por Karim, Hassan, Akanda y Mallik, en ésta se realiza un monitoreo de la humedad y la temperatura utilizando el sensor DHT-11 y un sistema implementado con Arduino [10]. Por otro lado, Yavari et al. proponen un sistema para la determinación de la calidad de la leche tras su procesamiento [11].

1.2.2. Controladores de temperatura en procesos industriales en la industria agroalimentaria de lácteos

En el artículo publicado por Alamirew et al. se exponen las características del controlador PID, uno de los sistemas industriales más relevantes por su diversidad de aplicaciones, y su comparación con el Modelo de Control Predictivo (MPC, por sus siglas en inglés), una propuesta que supera su desempeño en procesos complejos [12].

Esta misma comparativa es realizada por Anang et al. utilizando un Controlador Lógico Programable (PLC), quienes concluyen señalando las ventajas y desventajas que el algoritmo MPC muestra en relación con los controladores de cascada PID y PID estándar [13].

1.2.3. Sistemas de control de temperatura PID en sistema de pasteurización

Con respecto al ajuste de la temperatura de pasteurización de la leche se han instalado algunos sistemas tipo PID. Es por ello que Amole et al. abordan tres técnicas de ajuste diferentes para encontrar el control óptimo de parte del sistema PID, donde se destaca que el método Cohen-Coon (CC) exhibe el rendimiento preferido para la aplicación [14]. Firmansyah et al. han usado y comparado controladores multiloop (bucle múltiple) y en cascada en una miniplanta de pasteurización utilizando un algoritmo basado en control PID [15]. A partir de los resultados se infiere que el controlador de tipo bucle múltiple tiene un rendimiento superior de seguimiento del punto de ajuste, mientras que el controlador en cascada es mejor para el rechazo de perturbaciones. Cinco de los controladores más usados actualmente en intercambiadores de calor para leche son tratados en el artículo de Maulana et al. en el trabajo se abordan las tres diferentes implementaciones de los sistemas PID, el Sistema de Inferencia Difusa y el Sistema de Inferencia Difuso Neuro Adaptativo (ANFIS por sus siglas en inglés) [16].

1.2.4. Sistemas de pasteurización industrial monitorizados en el Internet de las Cosas (IoT)

En los ejemplos de sistemas industriales de pasteurización en los que se señala monitoreo por IoT se encuentra la industria del helado, intercambiadores de calor y uso de agua. Biscotti et al. comentan sobre la adopción de IoT para proporcionar valores de temperatura en la pasteurización de la leche en tiempo real para la elaboración de helados [17].

Sridharan et al. muestran el desarrollo de un sistema remoto de control y supervisión para un intercambiador de calor de doble tubo basado en IoT con Raspberry-Pi [18]. Otra aplicación encontrada fue desarrollada por Jadhav et al. donde plantearon un sistema de supervisión del volumen de agua empleado en los procesos de pasteurización, homogeneización y producción de lácteos [19].

1.3. JUSTIFICACIÓN

La producción de leche en México en 2020 según el Servicio de Información Agroalimentaria y Pesquera (SIAP) fue de 12 mil 553 millones de litros [20]. En particular, Querétaro pasó del lugar 18 al 12 a nivel nacional en producción de leche con una producción de 385 millones 629 mil litros al año en 2018 [21], y ha permanecido en ese puesto hasta 2020 [20]. Estos datos indican que la leche es un producto ampliamente consumido a nivel nacional y estatal como parte de la industria pecuaria, ocupando su producción el primer lugar en el estado de Querétaro. Por ello es importante contar con sistemas de control de temperatura actualizados y modernos que permitan el monitoreo y supervisión del proceso de pasteurización por medio de una plataforma de IoT, de modo que sea posible conocer el estado del proceso en tiempo real y en cualquier lugar con acceso a internet. La información contenida en el plan de desarrollo estatal 2023-2027, menciona que el gobierno del estado pretende apoyar a las pequeñas y medianas empresas (PyMES) con programas que aumenten la transferencia tecnológica y la tecnificación de los procesos en el sector primario [8].

Dado que es un proyecto enfocado a pequeños productores nacionales, es importante proponer un sistema económicamente accesible y amigable para el pequeño productor, que contenga una interfaz gráfica o aplicación celular (APP) con la cual pueda interactuar eficazmente el usuario. Por lo que se busca apoyar a las PyMES a tecnificar sus procesos y mejorar la producción en este sector.

1.4. HIPÓTESIS

La plataforma de enlace a IoT para la supervisión remota de datos de un proceso de pasteurización representará una mejora en la medición, procesamiento y monitoreo del control de temperatura en esta etapa de producción de la leche en pequeños productores.

1.5. OBJETIVOS

1.5.1. Objetivo General

Enlazar el monitoreo de la temperatura de pasteurización al Internet de las Cosas (IoT) para su supervisión de manera remota por parte de pequeños productores nacionales.

1.5.2. Objetivos Particulares

1.5.2.1. Establecer los parámetros del controlador PID y construir la infraestructura necesaria para el proceso de control de temperatura de pasteurización de la leche.

1.5.2.2. Investigar y elegir el hardware y software para el enlace al Internet de las Cosas.

1.5.2.3. Implementar una plataforma de enlace (Hardware y Software) al Internet de las Cosas y desarrollar una interfaz de monitoreo remoto de las mediciones de temperatura de pasteurización.

2. MARCO TEÓRICO

2.1. EL INTERNET DE LAS COSAS (IOT)

El Internet permite que diferentes sistemas eléctricos y electrónicos se conecten y comuniquen entre sí. Actualmente los dispositivos capaces de enlazarse a internet desempeñan funciones clave en la ayuda, la recopilación, la disección y la detección de información y datos sobre sus factores ambientales, y también son inteligentes para comunicarse entre sí a través de una gran organización vinculada llamada Internet de las cosas [22]. El crecimiento del internet de las cosas ha ido en aumento a causa de la gran accesibilidad de dispositivos de cómputo pequeños y accesibles, lo que consiste en tarjetas de desarrollo que conectan unidades de microcontrolador (MCU) y unidades de procesador (MPU) a un paquete preconstruido con muchas características necesarias para IoT [23]. El hardware IoT se puede clasificar en dos grandes categorías: dispositivos portátiles, tarjetas y sistemas integrados. Con respecto a dispositivos portátiles, existen muchas aplicaciones de equipos estándar preensambladas. El alcance en esta categoría se limita al software. Por el lado de las tarjetas y sistemas integrados, tanto el software como el hardware están abiertos para los desarrolladores [24]. Debido a la variedad de configuraciones existentes en el mercado, con respecto al tipo y número de características, con seguridad existe una plataforma que se adecúe exactamente a los requisitos del proyecto en cuestión.

2.1.1. Estructura IoT

La estructura IoT comprende una secuencia del proceso que se sigue desde la adquisición de datos por parte de los elementos de sensado hasta su manipulación o procesamiento en la nube. Todos los dispositivos IoT la incorporan o forman parte de ella. La estructura consta de cuatro capas, definidas de la siguiente manera [25]:

1. *Sensores o actuadores: Cosas*

La detección de los datos es necesaria para recolectar parámetros clave de la situación. Los cambios o eventos de un proceso o entorno de interés son recolectados en forma de datos por los sensores para su posterior transformación en información útil [26]. Los actuadores son considerados como un tipo de motor responsable del control y la acción de un sistema. Son de utilidad para transformar señales eléctricas o datos en movimiento para controlar un sistema [26]. Puede tomar información real y con ello cambiar el estado de varios dispositivos.

2. *La compuerta del internet*

Los datos adquiridos por el sensor deben ser transformados en un formato computarizado para su adecuado manejo. Con el acceso a internet los datos, ahora digitalizados, son dirigidos a través de diferentes puertas de enlace como Wi-Fi, redes de área local (LAN) cableadas, o el internet, a sistemas orientados a su procesamiento. Esta capa incluye protocolos y tecnologías orientadas a la conexión y comunicación entre dispositivos [25].

3. Procesamiento de datos

La información agregada en las plataformas IoT requiere preparación previa a su transferencia a la nube. Los sistemas IT funcionan cerca de la red de sensores para obtener información de manera directa y eficiente. El hardware y software en esta capa recopila, analiza e interpreta la información. El procesamiento en esta capa consiste en poner los datos a disposición para su posterior análisis o acción. Las tecnologías o herramientas en esta capa pueden abarcar sistemas de gestión de datos, plataformas de análisis o algoritmos de aprendizaje automático [25].

4. Centro de datos y la nube: Aplicación

Los servidores de bases de datos basados en la nube se utilizan para analizar a profundidad, administrar y almacenar datos de forma segura para obtener el resultado. Esto se puede hacer en las instalaciones o en el servidor de la nube. Ya en esta capa el procesamiento de los datos supone una interacción directa con el usuario final. Esta capa proporciona interfaces y funcionalidades para acceder y controlar los dispositivos IoT. Las tecnologías presentes en esta capa corresponden a aplicaciones móviles, interfaces de usuario para interactuar con la infraestructura IoT subyacente, servicios de middleware que habilitan la conexión y comunicación entre dispositivos, así como sistemas IoT [25].

2.1.2. Plataformas IoT

El desarrollo de proyectos en el área de Internet de las Cosas se compone de dispositivos capaces de enlazarse a internet o estar en línea. En particular, para el diseño de sistemas portátiles o el prototipado rápido y eficiente se hace uso de plataformas o placas de desarrollo, que consisten en MCU y MPU. La selección entre una MCU o una MPU se basa en los requerimientos de cada aplicación.

Los principales requerimientos tomados a consideración para la determinación de la plataforma de desarrollo son: procesador, velocidad de reloj, memoria de acceso aleatorio (RAM), memoria flash, memoria eléctricamente programable y borrable memoria de sólo lectura (EEPROM), voltaje de operación, comunicación, lenguaje de programación, número de puertos USB, número de pines de propósito general E/S (GPIO), dimensiones, precio y en algunos casos unidad de procesamiento gráfico.

2.1.3. Requisitos de una plataforma IoT

En el desarrollo de soluciones tecnológicas es necesario conocer los requisitos previos del proyecto para distinguir, analizar y evaluar las cualidades vitales en los nuevos dispositivos electrónicos IoT a medida que se abre el mercado o incrementa la oferta. Con el crecimiento del mercado y la aceptación de estos productos ante el público, la mayor parte de sus características se vuelven comunes. Debido a que los requisitos previos de los equipos para los proyectos de IoT generalmente varían entre sí, los dispositivos IoT cuentan con una alta precisión y un diseño que les permite trabajar en condiciones extremadamente distintas [27].

Las características del dispositivo IoT se organizan en cuanto a estas capacidades generales [27-29]:

- Protección y control de datos
- Tratamiento de datos y almacenamiento
- Conectividad: protocolos de comunicación
- Administración de energía
- Plano físico: arquitectura
- Seguridad
- Facilidad de mejora: puertos de conexión
- Comunidad y soporte para desarrolladores
- Requisitos previos de costos

2.1.4. MCU y MPU

Se define como unidad de microcontrolador o MCU a las pequeñas computadoras que se encuentran en un solo circuito integrado o microchip [30]. Tras ser programadas, son útiles para controlar tareas de manera automática. Estas computadoras, de manera similar a computadoras convencionales, contienen un núcleo procesador (CPU), RAM, ROM y periféricos equivalentes de entrada/salida, etc., que les permiten realizar tareas de forma independiente [31]. A diferencia de una computadora tradicional, generalmente una MCU se dedica a una sola función. Este aspecto ha ocasionado que frecuentemente se encuentre integrada en otros dispositivos. Por otro lado, una unidad móvil de procesamiento (MPU) se reconoce como una plataforma que contiene solamente una unidad central de procesamiento (CPU) [30]. Esto obliga a que las MPU requieran periféricos adicionales para realizar tareas.

Aunque en ocasiones poseen apariencias similares, los microcontroladores y los microprocesadores se diferencian en el poder de cómputo y autonomía. Los microprocesadores cuentan con mayor poder de cómputo, pero deben emplearse como componentes individuales en sistemas más grandes. En cambio, los microcontroladores son capaces de realizar funciones simples de forma independiente, a pesar de su potencia y funcionalidad limitadas [30].

2.1.5. MCU y Sistema en chip (SoC)

Es posible distinguir sencillamente la diferencia entre una MCU y una MPU. Sin embargo, un SoC posee más similitudes con una MCU. Las MCU se utilizan normalmente para pequeños sistemas de control integrados o aplicaciones de control, por lo que proporciona solo los componentes y potencia de procesamiento mínimas [31].

Un SoC suele ser una encapsulación de uno o más componentes encontrados en una computadora, inclusive microcontroladores, procesadores de señales digitales (DSP), aceleradores, procesador de gráficos (GPU) y hardware de soporte [31].

Puede tener memoria RAM, ROM, EEPROM o Flash como almacenamiento. Las mayores diferencias entre ambos sistemas radican en los periféricos y las aplicaciones.

Los periféricos incluidos en los SoC son más especializados que las MCU [31]. Por lo tanto, un SoC al contar con periféricos y componentes especializados dispone de un diseño más robusto para aplicaciones específicas. Los requisitos en aplicaciones especializadas pueden llegar a ser demasiado complejos para que los maneje una sola MCU. Una desventaja encontrada en estos sistemas es que no cumple con ningún estándar con respecto a los circuitos que lo contienen [31].

2.1.6. MPU y Computadoras de placa única (SBC)

A causa del avance en las tecnologías en el área IoT, nuevas propuestas de plataformas de desarrollo IoT han sido lanzadas comercialmente. Las computadoras de placa única (SBC), en contraste con las MPU, normalmente consisten en poseer gran poder de procesamiento y además varias opciones de comunicación en una placa de circuito impreso (PCB). En aplicaciones IoT, el propósito de las SBCs es el de realizar las tareas o procesos más complejos consiguiendo al mismo tiempo un bajo consumo y portabilidad [32]. El diseño de estas plataformas fue pensado para satisfacer las necesidades frecuentemente encontradas en proyectos IoT, por lo que conforme se proponen actualizaciones en estas plataformas, se añaden componentes adecuados para la comunicación, procesamiento, portabilidad, entre otros.

En relación con el número de funciones y la demanda de velocidad que suponen en las diversas aplicaciones IoT encontradas en el mercado, existe una variedad de opciones de plataformas. Las SBCs ampliamente utilizadas en la actualidad incluyen arduino, Raspberry Pi, Beagle Bone, Nvidia, Asus Tinker board, entre otros [33].

2.1.7. SoC y SBC

A partir de las descripciones anteriormente mencionadas, es posible establecer un marco útil para distinguir las aplicaciones entre ambos sistemas. Un SoC se refiere a un pequeño chip o circuito integrado que contiene todos los componentes y circuitos necesarios de un sistema particular (ver Figura 2.1).

El procesador es el centro del SoC, generalmente cuenta también con varios microcontroladores, microprocesadores y DSPs, los cuales se consideran como coprocesadores [34]. De manera nativa, la comunicación de datos en serie se realiza mediante el transmisor receptor asíncrono universal (UART). Componentes adicionales especializados encontrados en SoCs son reguladores de voltaje, osciladores, relojes y ADC/DAC (Analog-Digital Converter/Digital-Analog Converter) [34].

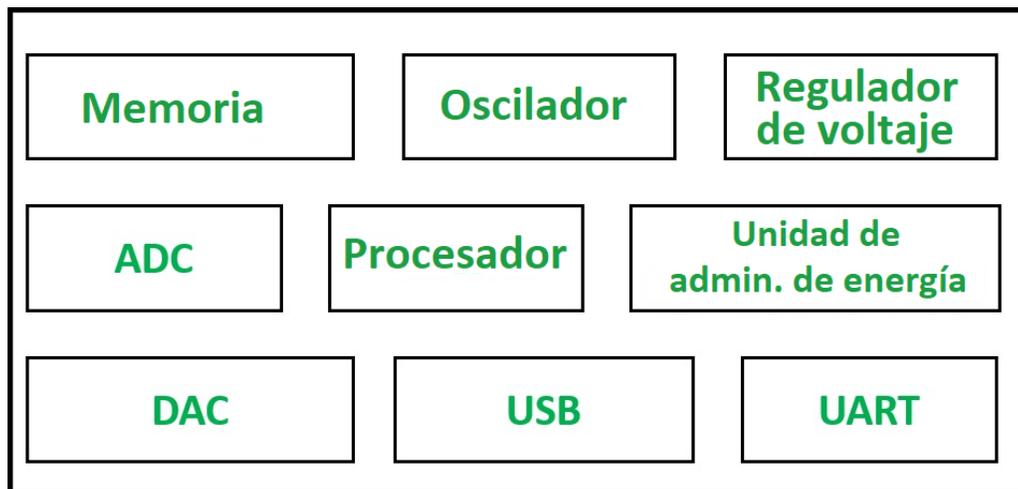


Figura 2.1. Diagrama de bloques de un SoC. Editado de [34].

Similarmente una SBC está construida en una sola placa de circuito impreso que contiene un microprocesador, memoria, dispositivos E/S y otras funciones que requieren una computadora funcional (Ver Figura 2.2). No obstante, una SBC se refiere a una computadora completa, es decir, que es capaz de realizar tareas de una computadora convencional, aunque no con la misma potencia o capacidad.



Figura 2.2. Diagrama de bloques de una SBC. Editado de [34].

El procesador es igualmente el centro de una SBC. En las especificaciones de una SBC, comúnmente cuentan con un procesador ARM, una memoria RAM estática de 8 bits o 16 bits, pines GPIO (para su uso con sensores, pantallas, teclados, etc.), puerto Ethernet y fuente de alimentación de 5 V CC [34]. Incluso uno o más SoCs pueden formar parte esencial de una SBC. Las características principales que distinguen ambas plataformas se observan en la Tabla 2.1.

Tabla 2.1. Diferencia entre SoC y SBC. Editado de la información de [34-36].

Base para la diferencia	SoC	SBC
Componentes	Tiene todos los componentes integrados	Muchos periféricos se conectan mediante ranuras
Adaptabilidad	Menor adaptabilidad	Mayor adaptabilidad
Precio	Más costoso	Menos costoso
Ventajas	<ul style="list-style-type: none"> - Es de un tamaño pequeño - Incluye muchas características y funciones - Bajo consumo de energía - Es flexible con respecto al tamaño y factor de potencia - Es económico 	<ul style="list-style-type: none"> - Fácil de usar - Tienen hardware verificado - Bajo consumo de energía - Buen rendimiento a bajo precio
Desventajas	<ul style="list-style-type: none"> - Mucho tiempo para el proceso de diseño (6 - 12 meses) - Recursos limitados - No recomendado para diseños de producto de bajo volumen/alto margen 	<ul style="list-style-type: none"> - Mucha personalización puede dificultar el diseño - Menos capacidad que las computadoras de placas múltiples - Requiere de mayor conocimiento
Áreas de aplicación	<ul style="list-style-type: none"> - Industria - IoT - Sistemas integrados/embebidos 	<ul style="list-style-type: none"> - Desarrollo de producto final - IoT - Inteligencia artificial

Aplicaciones	<ul style="list-style-type: none">- Teléfonos inteligentes- Relojes inteligentes- Tabletillas- Computadoras- Diversas aplicaciones de IoT- Sistemas integrados: uso de microcontroladores	<ul style="list-style-type: none">- Compuertas IoT flexibles- Monitoreo inteligente de activos- Aplicaciones de inteligencia artificial
--------------	--	---

2.1.8. Categorías de plataformas IoT

Los dispositivos IoT se encargan de proporcionar información útil sobre sus interacciones en la red. Debido a su portabilidad y almacenamiento de energía, estos dispositivos hacen uso de Bluetooth de baja energía (IEEE 802.15.4), Wi-Fi (IEEE 802.11), u otros estándares de comunicación, aunque en algunos casos pueden ser conectados igualmente mediante Ethernet (IEEE 802.3) [24].

Los servicios que proporcionan estos sistemas se pueden clasificar en tres subcategorías [24]:

1. Control de dispositivos: monitoreo, seguridad, actualizaciones de firmware.
2. Adquisición de datos: gestión y transformación en las diferentes capas.
3. Desarrollo de aplicaciones: análisis, lógica de eventos, visualización y programación de apps.

2.2. CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO (PID)

Un sistema de control PID consiste en un proceso retroalimentado que es ampliamente utilizado en la industria debido a su método de operación, el cual permite calcular la diferencia entre el valor de la medición y el punto de control [37]. Esta característica permite tener un control altamente preciso y eficiente.

El primer controlador PID fue desarrollado por Elmer Sperry en 1912. En conjunto con Morris E. Leeds y otros, entendieron intuitivamente que las acciones de control proporcional y de encendido y apagado generalmente no proporcionarían un encendido adecuado [38]. Por otro lado, el primer análisis teórico fue publicado por Nicolas Minorsky en el año de 1922. Minorsky realizó este análisis mientras desarrollaba sistemas de dirección automáticos para la armada de los Estados Unidos [39]. Basó su análisis en el error instantáneo del timonel y aprovechó datos del error pasado para crear un modelo matemático basado en la tasa de cambio de los errores. Su objetivo principal no era lograr el control sino la estabilidad.

2.2.1. Estructura general del control PID

Un controlador PID consiste en un circuito de control que trabaja con una entrada y genera una salida. Este circuito se describe con una función de transferencia que puede ser ajustada de forma independiente, esto se suele denominar un 'grado de libertad'. La Figura 2.3 presenta un diagrama convencional de un lazo de control PID.

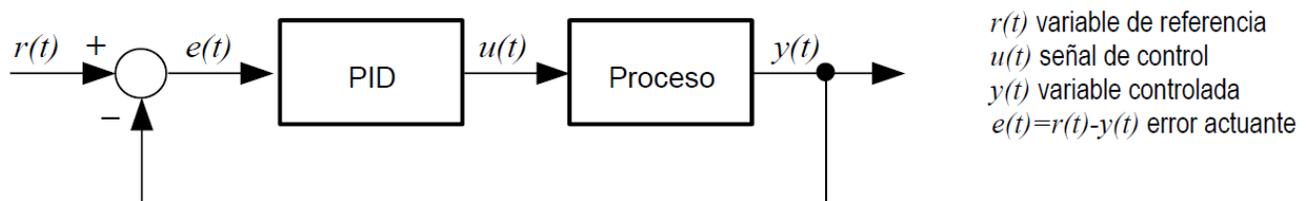


Figura 2.3. Diagrama del funcionamiento general de un control PID. Imagen tomada de [40].

donde K_p es una ganancia proporcional ajustable, C_p es el control proporcional. La ecuación 2 propone que el control proporcional es capaz de regular sistemas estables, aunque presenta un ajuste limitado y en algunas ocasiones un error en estado estacionario (normalmente llamado off-set).

2.2.1.2. Operación de control integral (I)

Produce una salida de control que varía según la acumulación del error, esto se expresa matemáticamente como la integral del error a lo largo del intervalo de tiempo durante el cual se efectúa el control. (ver Ecuación 3). Este método implica un modo de control lento.

$$u(t) = K_i \int_0^t e(\tau) d\tau, \quad (3)$$

Con la transformación de la función del error en el dominio de la frecuencia se obtiene la Ecuación 4, que representa la función de transferencia correspondiente al control integral.

$$C_i(s) = K_i/s, \quad (4)$$

La señal de salida es distinta de cero cuando la señal de error alcanza cero. Es decir, cuando se tiene una referencia constante o perturbaciones, no existe off-set.

2.2.1.3. Operación de control derivativa (D)

Esta aproximación se realiza a través de la serie de Taylor. De manera general, las ecuaciones del error para una aproximación de primer orden son de la forma:

$$e(t + T_d) \approx e(t) + T_d \left(\frac{de(t)}{dt} \right), \quad (5)$$

Por la forma de la Ecuación 5 esta es una acción de control difícil de sintonizar, lo que complica utilizarla en tiempo real. Por ello, este tipo de acción se utiliza cuando es necesario anticipar un efecto futuro sobre el sistema.

El control PID combina las tres acciones de control, descritas en las ecuaciones 1, 3 y 5, lo que implica el uso de ajustes **proporcionales, integrales y diferenciales**. Donde el ajuste de control **PID** se construye sumando estas tres operaciones de control, lo que entrega la forma:

$$u(t) = K \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (6)$$

2.2.2. Método de ajuste Ziegler-Nichols

La sintonización del controlador implica elegir los parámetros adecuados de manera que se ajusten a los requerimientos del proceso. Se recurre a los métodos de Ziegler y Nichols [41] para modificar las ganancias de un controlador PID, especialmente cuando no se dispone de conocimiento sobre los modelos matemáticos del proceso [41]. Este proceso involucra la estimación de la ganancia proporcional K , el tiempo integral T_i y el tiempo diferencial T_d presentes en la ecuación 6, en función de las propiedades de la señal de salida de un proceso específico (respuesta transitoria). El método de sintonización conduce a la calibración óptima de la señal de control del sistema.

La regla de sintonización de Ziegler-Nichols (ZN) [42] utiliza la ganancia y el período finales del proceso. Siguiendo la metodología de Ogata, se identifican dos enfoques que funcionan como directrices de ajuste según las reglas de Ziegler-Nichols. El objetivo de ambos enfoques es limitar el sobrepaso en la respuesta del sistema a un máximo del 25% [41].

2.3. CONTROL ANALÓGICO INTEGRADO PAC

La tecnología “Programmable Analogical Controller” (PAC) se refiere al control de procesos industriales con variables analógicas, y reúne al mismo tiempo funcionalidades de control lógico encontradas en los sistemas de tipo “Programmable Logic Control” (PLC) [43]. Esta tecnología fue un concepto nuevo en lo que respecta a los equipos de control avanzado en el año 2008. Un sistema PAC es capaz de ejecutar diferentes funciones en tiempo real, interrelacionando distintas variables de proceso simultáneamente. Así se generan varios resultados evitando que las distintas líneas de señal deban llegar a los PLC [44]. Actualmente los sistemas PLC ya cuentan con funciones analógicas. Sin embargo, aunque los sistemas PAC tienen características similares a los PLC, están orientados principalmente al control analógico. Por esta razón agilizan el proceso de diseño de complejidad baja/media y la verificación de circuitos analógicos. En un proceso industrial, se emplea el controlador PLC para supervisar las actividades [44].

Los sistemas PAC fueron propuestos con una programación por “bloques de función preprogramados”, para permitir una configuración rápida e intuitiva. Los módulos de funciones predefinidos son conectados virtualmente de acuerdo con los requisitos del sistema. La configuración involucra la interconexión de estos módulos de forma análoga a la conexión lógica entre componentes físicos [43].

A diferencia de la programación escalera en los sistemas PLC, no es necesario conocer lenguajes de programación, sino solamente conocer de las funciones y características del equipo a través del manual. La configuración y parametrización pueden realizarse mediante el teclado y pantalla del mismo equipo de tipo PAC o a través del bus de comunicación RS-485 con programas especializados de diseño gráfico por computadora [44].

2.3.1. Comunicación RS232 y RS485

Los estándares de conexión son imprescindibles en las comunicaciones industriales, puesto que permiten la comunicación entre una computadora, dispositivos de monitoreo y control, y sus periféricos. En aplicaciones industriales, se usan redes de bajo costo para la automatización. Las conexiones pueden basarse en corriente o tensión. Si se usan conexiones de tensión, la magnitud de la tensión representa la información transmitida entre los conductores conectados [45].

El progreso tecnológico y la expansión de la industria han generado una creciente demanda de redes locales con mayor velocidad y alcance, dichos cambios promueven la normalización de un nuevo estándar. Como resultado, surgieron varias directrices globales sugeridas por la EIA (Asociación de Industrias Electrónicas) [45].

2.3.1.1. Estándar RS-232

En la década de los 60, debido a los avances en las telecomunicaciones, la EIA propuso una norma de comunicación estándar. Esto fue necesario porque los sistemas requerían un dispositivo que pudiera convertir señales analógicas a digitales y viceversa para el intercambio de datos. En la norma propuesta, el protocolo hace uso de un modo asíncrono, en el que el receptor y el emisor deben tener la misma frecuencia. El estándar fue introducido como RS-232 y tras modificaciones en los 40 años siguientes pasó a llamarse EIA-232 [45].

Características

- Velocidad: 50 - 19,200 baudios
- Comunicación directa: sobre banda base digital
- Comunicación mediante módem: en banda base análoga modulando la portadora
- Utiliza un conector macho tipo D de 9 terminales
- Máximo alcance de conexión de 15 metros

- Tasa de comunicación de hasta 20 kbps
- La edición europea se regula bajo la norma CCITT V.24.
- El rango de tensión para representar el valor lógico '0' va desde +3 V hasta +15 V, mientras que para el valor lógico '1' varía entre -3 V y -15 V.

2.3.1.2. Estándar RS-485

El estándar RS-485 forma parte de la norma RS-449, la cual fue publicada en 1983. Con respecto al estándar RS-485, Forero explica que en el estándar RS-485 se especifican las características eléctricas para la transmisión de señales lógicas '0' o '1' a través de un medio de transmisión digital mediante una señal diferencial y una señal de alta impedancia, lo que permite conectar múltiples dispositivos emisores en la misma red. Debido a esta razón, el estándar resulta altamente beneficioso en canales de transferencia de información local industrial de dos hilos. Desde 2003 se conoce como la norma TIA-485-A.222 de acuerdo con la TIA (Asociación de la industria de las telecomunicaciones) [45].

Características

- Cable de par de cobre trenzado y terminales RJ11: menos interferencia y más velocidad de transmisión.
- Conecta un máximo de 32 dispositivos (emisor a receptor)
- Permite la transmisión bidireccional simultánea entre el dispositivo principal (maestro) y los secundarios (esclavos).
- Mantiene compatibilidad con RS-232
- Capacidad de transmisión se define en función de la relación entre la cantidad de datos a transferir y el tiempo que lleva la señal en el portador
- Hasta 1200 metros de extensión
- Velocidad máxima 10 Mbps en una distancia de 12 m
- Rango de voltaje para '0': +0.2 V a +12 V y para '1': -1.5 V a -5 V
- Voltaje máximo en la salida: -7 V a +12 V

2.4. EL PROCESO DE PASTEURIZACIÓN

La pasteurización es un proceso fundamental durante la producción de bebidas y alimentos, y está en gran medida asociada a los lácteos, ya que se trata de una de las primeras etapas del proceso. En particular, la leche procesada por medio de pasteurización a baja temperatura es la utilizada para producir quesos. La pasteurización de la leche implica la erradicación de microorganismos perjudiciales y la mayoría de los saprófitos (organismos que se nutren de materia orgánica en descomposición) mediante la variación de temperaturas y tiempos, con el propósito de mantener la calidad microbiológica y prevenir su deterioro [46]. Reducir la flora relacionada con el producto tiene como objetivo aumentar su vida útil sin afectar su composición química y sus características sensoriales [47].

Los límites críticos para la pasteurización se dan a 63°C durante 30 minutos, a 72°C durante 15 segundos, 135°C durante 8 segundos o 149°C durante 2 segundos [2]. Este proceso requiere de monitorización constante, puesto que, en caso de fallar, se deben tomar acciones correctivas a tiempo, de lo contrario, la leche ya no será útil para elaborar ningún tipo de producto. Posterior a este proceso, se recomienda que la leche sea refrigerada a una temperatura de -1 °C [48].

2.4.1. Principio

En Inglaterra tradicionalmente se propuso la pasteurización como mantener la leche a una temperatura entre 62.8 °C y 65.56 °C durante 30 minutos como mínimo, enfriándola de inmediato a una temperatura menor a 12.8 °C [49, p. 805].

Desde su inicio, se ha afirmado que la pasteurización ultra rápida, que implica calentar la leche rápidamente a 75 °C durante solo 15 segundos, es cuestionable e inestable, y no se considera satisfactoria [49]. A pesar de que en el laboratorio se cumplan las normas y se obtengan los resultados requeridos, existen muchos factores complejos que generan incertidumbre, lo que dificulta su aplicación con certeza.

Savage [49] señaló que expertos en el campo están de acuerdo en que la pasteurización a temperaturas más bajas, junto con el control de la temperatura durante un lapso específico, es el método más confiable y efectivo en lo que respecta a conservar las propiedades organolépticas y nutrientes de la leche. Esta pasteurización, en el intervalo de 62 °C y 65° C, es llamada “holder” o de retención.

Con respecto a la eficacia de la pasteurización lenta, se reconoce que temperaturas más bajas y tiempos más largos son más efectivos para eliminar patógenos, mayor control se tendrá y menos probabilidades habrá de efectos perjudiciales en la leche. Por ello, desde 1932 el Servicio de Sanidad Pública de los Estados Unidos define la pasteurización como el calentamiento de toda la leche o productos lácteos a al menos 61 °C, sosteniendo esta temperatura durante un mínimo de 30 minutos [49, p. 808].

2.4.2. Tipos de pasteurización

Los dos métodos de pasteurización más populares son la pasteurización tradicional o LTLT y la ultrapasteurización o UHT [50]. Con ambas técnicas se busca cumplir el mismo objetivo, eliminar los microorganismos patógenos presentes en los alimentos y evitar la descomposición del alimento. Esto se debe a que dichos microorganismos al ser ingeridos mediante la comida producen diversas enfermedades.

Algunos ejemplos de las enfermedades más frecuentes en microorganismos presentes en los alimentos son: tuberculosis, difteria, polio, salmonelosis, escarlatina, fiebre de Malta y la fiebre entérica [50].

El tiempo y la temperatura marcan la diferencia entre métodos de pasteurización. Se debe contemplar para todos ellos un calentamiento homogéneo de la sustancia a tratar, en este caso leche entera. De acuerdo con la empresa Tecnosoluciones Integrales S.A., los tres métodos más conocidos son VAT o LTLT, HTST y UHT y son caracterizados por los aspectos definidos en forma de pros y contras [50].

2.4.2.1. *Pasteurización lenta de baja temperatura (también VAT o LTLT)*

En este tipo de pasteurización, se calienta el líquido a una temperatura entre 63 °C y 65 °C y permanece a dicha temperatura durante 5 a 30 minutos. Acto seguido se realiza un enfriamiento lento, a veces más de 24 horas hasta el intervalo de 4 °C y 6 °C, luego envasado y sellado [50]. Las condiciones de esta pasteurización dan lugar a las ventajas y desventajas presentadas en la Tabla 2.2.

Tabla 2.2. Pros y contras de la pasteurización VAT/LTLT.

Pasteurización VAT/LTLT	
Pros	Contras
Conserva mejor el valor nutritivo de la leche	Requiere un tiempo muy largo y espacio muy extenso para el tratamiento
Elimina mohos, levaduras y la gran parte de bacterias	Cuenta con una eficacia de eliminación menor
La leche es apta para su consumo durante una semana	Exige modificaciones frecuentes

2.4.2.2. *Pasteurización rápida de alta temperatura (también HTST)*

Conlleva un calentamiento y enfriamiento a la leche circulante, lo que la lleva a una temperatura de 72°C durante 15 a 20 segundos y luego a una de 6°C [50]. Este método puede llevarse a cabo con todas tecnologías que se presentan a continuación.

2.4.2.2.1. Con pasteurizador

El proceso se lleva a cabo en un sistema pasteurizador. La Tabla 2.3 presenta los pros y contras.

Tabla 2.3. Pros y contras de la pasteurización tipo HTST con el uso de un pasteurizador.

Pasteurización HTST con pasteurizador	
Pros	Contras
No altera sustancialmente el producto en términos físicos, químicos o nutricionales	Es necesario refrigerar la leche para prevenir la proliferación de microorganismos
Logra eliminar completamente las bacterias dañinas y las alterantes casi en su totalidad, con un alto grado de efectividad	Tras abrir el contenedor, el tiempo de consumo se reduce a un periodo de 3 a 4 días
La duración máxima de frescura se extiende hasta 5 días más	

2.4.2.2.2. *Con intercambiadores de calor*

En este proceso los intercambiadores de calor, los cuales constan de sistemas de regulación de temperatura a través de tubos o placas metálicas, llevan a cabo el calentamiento a altas temperaturas durante un tiempo muy breve. Se pueden observar los pros y contras de este método en la Tabla 2.4.

Tabla 2.4. Pros y contras de la pasteurización tipo HTST con el uso de un intercambiador de calor.

Pasteurización HTST con intercambiador de calor	
Pros	Contras
Procesa de 1 000 l a 10 000 l de leche por hora	El proceso requiere personal calificado y controles de calidad rigurosos
Es posible recuperar tres cuartas partes del calor utilizado para calentar	
Se reducen los costos de mantenimiento con el poco equipo industrial necesario	

2.4.2.3. *Ultrapasteurización (también UHT)*

Mediante un pasteurizador, se calienta el flujo de leche a una temperatura de 135 °C a 150 °C en un período de 2 a 10 segundos. Inmediatamente después se refrigera el producto a 4 °C. A continuación, la leche se manda a un aparato que la usa para llenar y sellar los contenedores, encontrado por debajo del sistema de pasteurización [50]. En la Tabla 2.5 se desglosan todas las ventajas y desventajas que implica este método.

Tabla 2.5. Pros y contras de la pasteurización de tipo UHT.

Pasteurización UHT	
Pros	Contras
Elimina efectivamente los organismos dañinos y sus formas de resistencia	Implica equipo caro y complejo, una planta para empaque aséptico y operarios altamente calificados
Se produce una mínima degradación del alimento	Ocasiona la pérdida de características organolépticas de la leche
La leche se envasa en condiciones asépticas, lo que evita una contaminación posterior	Durante el proceso, se alteran proteínas, se dora el azúcar y se pierden algunas vitaminas, entre otros efectos negativos
No requiere refrigeración posterior	Influye en ciertos componentes de la leche, como el daño que causa a la lecitina y las vitaminas, la coagulación de la lacto albúmina debido a la concentración de sales, y altera la caseína, lo que provoca una apariencia oscura y un leve aroma a leche cocida
Se debe conservar la leche durante 6 meses	Para garantizar la eliminación de leucocitos, los grupos de caseína y los residuos orgánicos se recomienda purificar la leche a posteriori
Los lípidos del producto no sufren cambios	

2.4.3. Efecto de la temperatura en las bacterias

La velocidad de crecimiento, y con ello el tiempo de generación (g), están condicionados en gran medida por la temperatura. Suponiendo constantes las demás condiciones ambientales, cada bacteria muestra una curva típica que relaciona la tasa de crecimiento con la temperatura. En la curva se distinguen tres puntos característicos llamados *temperaturas cardinales* [51] (ver Figura 2.5).

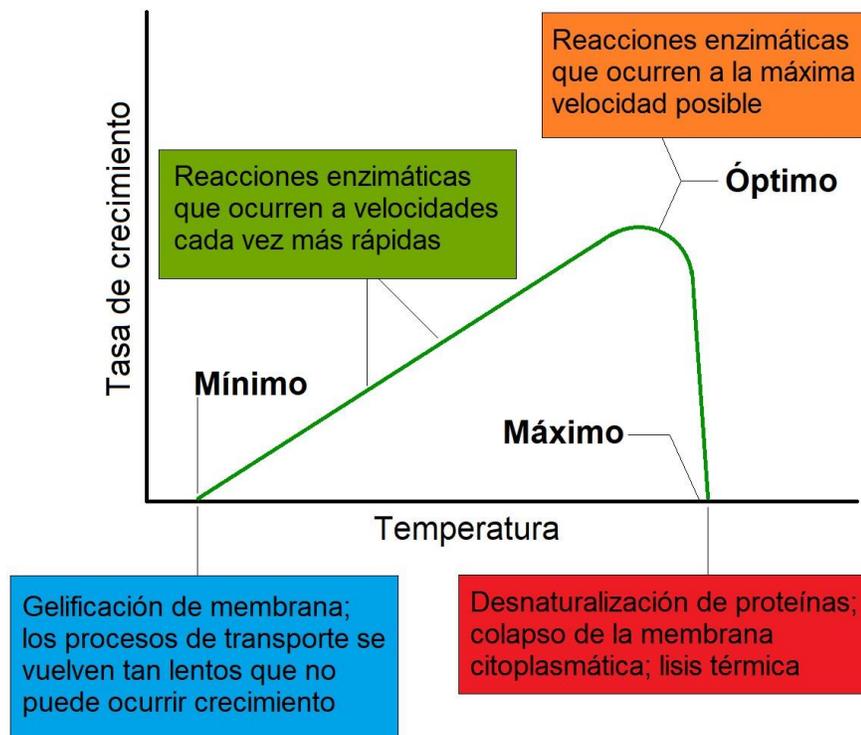


Figura 2.5. Efecto de la temperatura sobre el crecimiento microbiano. Modificada de [51].

Los puntos de temperatura presentes en la gráfica se denominan “temperaturas cardinales”, las cuales se componen de:

- T_{\min} : mínima, en el rango menor a esta no ocurre crecimiento.
- T_{\max} : máxima, en el rango superior a esto no ocurre crecimiento.

- T_{opt} : óptima, posibilita el crecimiento más rápido (donde tiempo de generación g es mínimo).

De acuerdo con la clase de microorganismos, existen diferentes rangos de temperatura aceptables [51].

- Psicrófilos: crecimiento óptimo entre 15 °C y 30 °C, temperatura máxima 35 °C.
- Mesófilos: crecimiento óptimo entre 25 °C y 40 °C, temperatura máxima de 35 °C a 47 °C.
- Termófilos: crecimiento óptimo entre 50 °C y 75 °C, temperatura máxima de 80 °C a 113 °C.

En las bacterias, la viabilidad se refiere a la capacidad que tienen estas de crecer y dividirse. Cuando se supera la temperatura máxima de crecimiento, las bacterias sufren los efectos de la pérdida de viabilidad. Aunque las bacterias mudaran a un medio idóneo, pierden la capacidad de desarrollarse o reproducirse. Según el análisis cinético, la muerte se debe a la aniquilación o desactivación definitiva de una molécula crucial [51]. La desactivación por calor en una suspensión bacteriana se caracteriza por los siguientes parámetros clave:

- Tiempo térmico mortal: tiempo mínimo para matar todas las bacterias a cierta temperatura.
- Tiempo de reducción decimal: tiempo para bajar un 10% de la densidad bacteriana a cierta temperatura.
- Punto térmico mortal o temperatura de inactivación: temperatura más baja que elimina completamente a la población bacteriana, regularmente 10 minutos.

Las temperaturas de inactivación de los principales patógenos transmitidos por la leche durante un tratamiento de 15 segundos son [52]:

- *Staphylococcus aureus* > 66.5 °C
- *Yersinia enterocolitica* > 62.5 °C
- *Escherichia coli* > 65 °C
- *Cronobacter sakazakii* > 67.5 °C
- *Listeria monocytogenes* > 65.5 °C
- *Salmonella ser. Typhimurium* > 61.5 °C

Sin embargo, tiempos de calentamiento, enfriamiento y mantenimiento más largos resultan en una mayor letalidad a una temperatura de proceso determinada [53], argumento que se confirma con una destrucción casi completa de las colonias de bacterias presentes en la leche producida por una pasteurización LTLT [54].

Temperaturas de inactivación en virus presentes en la leche durante tratamientos de 30 minutos [55]:

- Cytomegalovirus > 60 °C
- Retroviridae > 60 °C
- HIV-1 > 60 °C
- Ébola > 60 °C
- Marburg > 60 °C
- Zika > 60 °C
- Virus del bosque Semliki > 60 °C
- Papillomavirus > 60 °C
- Mirone norovirus > 60 °C
- SARS-CoV & SARS-CoV-2 > 56 °C
- MERS-CoV > 63 °C

2.5. ESTADO ACTUAL DE SISTEMAS DE PASTEURIZACIÓN VAT

La pasteurización es un proceso útil para destruir la mayor cantidad de microorganismos, para que sea poco probable la incidencia de enfermedades por el consumo del alimento, sin alterar la composición y cualidades del líquido. Se tiene registro de que este procedimiento ha sido estudiado desde 1864 [56]. Fue entonces que Louis Pasteur informó que la aplicación de calor al vino y la cerveza previene defectos ácidos, amargos y viscosos en el vino. Tres años más tarde, aplica también calor a la leche e informa que el proceso pospuso la acidificación de la leche. Veinte años después, Soxhlet y Jacobi [56] defendieron el calentamiento de la leche para la alimentación infantil, puesto que redujo las enfermedades y salvó vidas al eliminar los patógenos.

2.5.1. Sistemas de pasteurización actuales

Desde su creación ha habido una serie de mejoras relacionadas a la sofisticación de la tecnología involucrada. Es decir, sistemas de transferencia de calor, tecnologías de control, instrumentos y elementos de monitoreo, e interfaces de usuario. Estas mejoras se refieren a tiempos de pasteurización menores, un proceso más confiable, abaratamiento y mayor accesibilidad de la tecnología, y mayor eficiencia general.

Sin embargo, los sistemas ahora son fabricados acorde a los requerimientos de cada segmento del mercado. En específico, los pequeños y medianos productores tienen acceso a sistemas LTLT y HTST de acuerdo con sus capacidades económicas y las capacidades productivas que requieren. Camacho et al. identifica que en Chiapas, uno de los estados de mayor producción de leche de bovino, un productor de leche pequeño cuenta con 9 a 23 vacas, de donde de 5 a 11 son productoras y obtiene de 3 a 6 litros de leche por bovino [57].

Según estos datos, un pequeño productor en estas regiones produce un equivalente mínimo de 15 litros y un máximo de 66 litros de leche, que requiere pasteurizar en promedio al día. Los sistemas LTLT suponen un menor costo y nivel de conocimiento técnico para poder ser operados y darles mantenimiento, pero con el desarrollo de tecnologías de monitoreo y control la accesibilidad aumenta.

Estas pasteurizadoras cuentan con capacidades de hasta 1000 L según los requerimientos del cliente [58,59]. La inclusión de sistemas de transferencia de calor o erradicación de patógenos nuevos y sus respectivos dispositivos de control ha sido tarea de los últimos años, tales técnicas incluyen microondas, campo eléctrico pulsado, irradiación, calentamiento óhmico y energía solar [60]. No obstante, debido a su baja eficiencia y reciente investigación, no han sido adoptados formalmente. Las mejoras que han sido adoptadas gradualmente con el tiempo se relacionan con la transferencia de calor, tales como placas intercambiadoras de calor, y dispositivos de control, como sistemas PLC o PID [14-16].

2.5.2. Propuestas de sistemas de pasteurización en México y Latinoamérica para pequeños productores

Gutiérrez del CIMAV [61] llevó a cabo un análisis en el que propone diseñar, simular y optimizar un intercambiador de calor para satisfacer los requisitos de manufactura, costos y operación de un proceso de pasteurización lenta. El estudio propone un enfoque matemático a partir de modelos que expliquen cómo funcionan los sistemas y sustancias utilizados. Estos modelos proponen sistemas de ecuaciones cuyo objetivo es encontrar los valores que cumplan correctamente con las condiciones del sistema. A este conjunto de valores se le llama 'estado del sistema'. El estado del sistema es el punto donde trabaja, y puede o no funcionar como se espera. Cada estado del sistema se refiere a un punto de operación, y a su vez tiene un costo de producción y funcionamiento asociado. El estudio busca un diseño eficiente con el costo más bajo de producción y funcionamiento, lo cual se logra optimizando los puntos de operación.

Flores y Tipán de la Universidad Politécnica Salesiana de Quito [62] crearon una olla para pasteurizar de manera semiautomática leche cruda de la Asociación Campo Verde en Turucucho, encontrada al noreste del Cantón Cayambe. El equipo pasteuriza leche cruda para hacer productos lácteos e impulsar el crecimiento de la comunidad. La justificación del proyecto se debe a la competitividad y los estándares de calidad de la zona. Estos factores impulsaron a la asociación a usar nueva tecnología para procesar la leche y elaborar productos lácteos, como parte de un emprendimiento para mejorar la comunidad. Adicionalmente, con el proyecto se buscó aprovechar el producto primario que solía ser vendido a algunas industrias lecheras, ya que la cantidad era suficiente para ser procesado y consumido localmente.

Garbanzo, Moreira y Rodríguez de la Universidad de Costa Rica [63] desarrollaron un diseño de un equipo de pasteurización por lotes para pequeños productores de queso fresco artesanal con una capacidad de 300 litros de leche. El sistema diseñado se compone de un tanque de acero inoxidable con una chaqueta como intercambiador de calor, un sistema de agitación y los sistemas de calentamiento y enfriamiento que utilizan agua como fluido de trabajo. El sistema de calefacción consiste en un quemador de gas LP. Por otro lado, el enfriamiento se consigue por medio de agua fresca proveniente de un tanque de reserva. La misión de la Universidad de Costa Rica es aportar nuevas tecnologías a los diversos sectores productivos del país. En este proyecto, se buscó aportar al sector de producción artesanal de alimentos derivados de la leche. La aportación del sistema fue su bajo costo y mayor calidad en cuanto a equipos similares utilizados en la región. Además, se incluyeron manuales de mantenimiento, operación y limpieza del equipo para su uso adecuado en la planta.

3. METODOLOGÍA: PROPUESTA DEL PROTOTIPO DE ACTUALIZACIÓN MEDIANTE IOT

El diseño de un sistema de monitoreo IoT que cuenta con un sistema de pasteurización y de control PID para monitorear un proceso real plantea mayor complejidad. Por lo tanto, con base en el estado del arte y a la literatura de los sistemas que son de interés se propuso una guía conveniente para el proyecto. En la metodología se expone los pasos seguidos para llevar a cabo el diseño y elaboración del sistema de pasteurización IoT, igualmente el desglose de la arquitectura y funcionamiento completo de este (Ver Figura 3.1).

1. Sistemas de control tradicional PID retroalimentado y PID+Fuzzy
2. Método de Autosintonía por Zeigler-Nichols
3. Parámetros de control de temperatura para un sistema de pasteurización con Pt100
4. Revisión del Control Analógico Integrado PAC tipo LS3400 y Programación por Bloques de Función
5. Protocolos, interfaces de comunicación y potencia del controlador LS3400
6. Software de Programación Loop Win, Proasis DCS
7. Estado actual del sistema de pasteurización y propuesta de actualización mediante IoT
8. Extracción de datos de Temperatura desde la interfaz PC-Controlador LS3400
9. Requerimientos de datos y su procesamiento en IoT para un pequeño productor de Lácteos
10. Selección del Hardware para enlazar datos de temperatura a IoT y Servidor Web o HTTP (Hypertext Transfer Protocol)
11. Programación del Hardware o plataforma de enlace a IoT
12. Investigación y selección del lenguaje y librerías para la creación de la interfaz de monitoreo
13. Configuración y desarrollo de una interfaz para el monitoreo remoto

-
14. Pruebas de pasteurización de leche a diferentes temperaturas desde IoT
 15. Pruebas de enlace a internet
 16. Obtención de información y visualización de datos de la temperatura de pasteurización
 17. Interfaz virtual IoT para monitoreo del proceso de pasteurización por el usuario
 18. Pruebas de funcionamiento del sistema en tiempo real

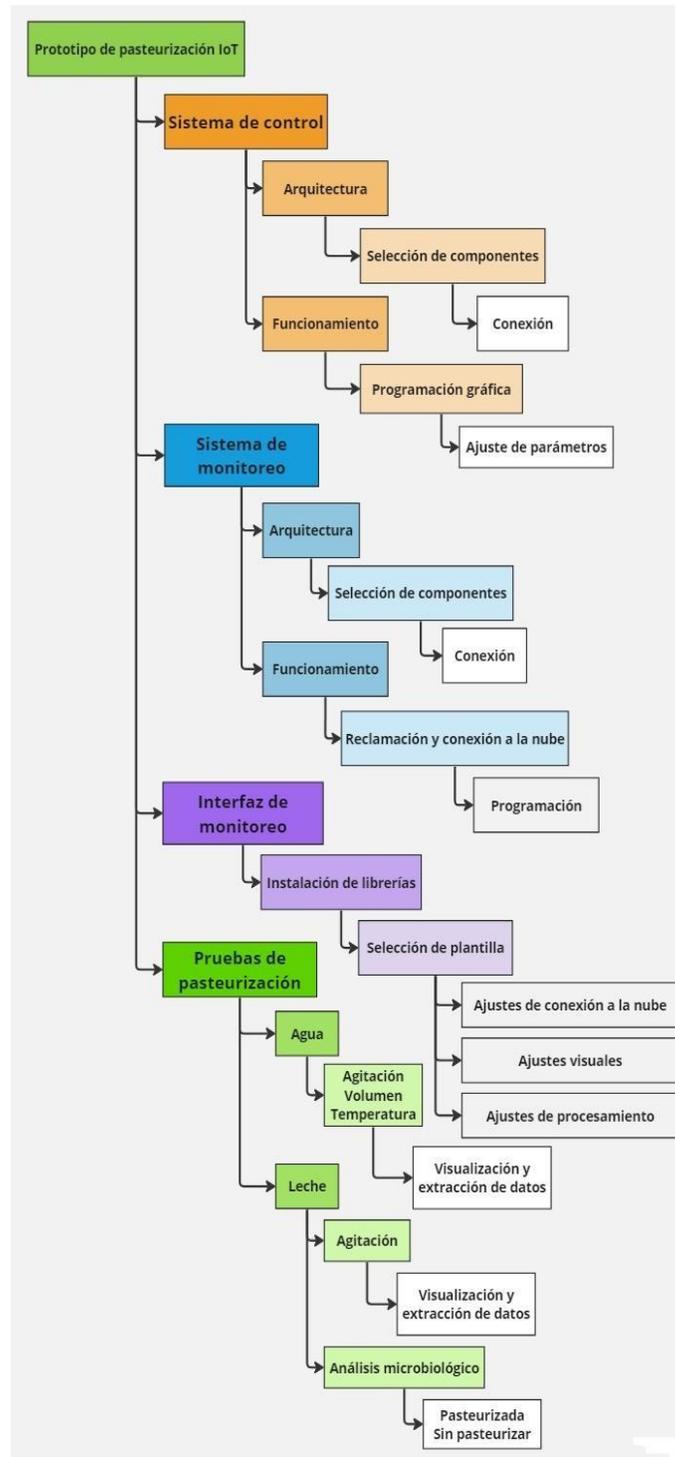


Figura 3.1. Diagrama general de la metodología del prototipo de pasteurización IoT.

3.1. INFRAESTRUCTURA

3.1.1. Arquitectura del sistema de control

Se consideró la temperatura como variable a monitorear con respecto al tiempo, acorde a los parámetros de control que son necesarios en el proceso de pasteurización [2]. Siguiendo esta línea se consultaron diseños dentro del estado del arte que reuniera los requisitos de un proceso de pasteurización LTLT, de donde se determinó el presentado por los ingenieros mecánicos Tipán y Flores [62] como el más adecuado para el propósito de este proyecto (Ver Figura 3.2). Se tomaron en cuenta los siguientes aspectos: costo de los materiales, tiempo de fabricación, volumen del contenedor, diseño análogo a los sistemas de pasteurización LTLT, diseño capaz de albergar los instrumentos de medición adecuadamente.

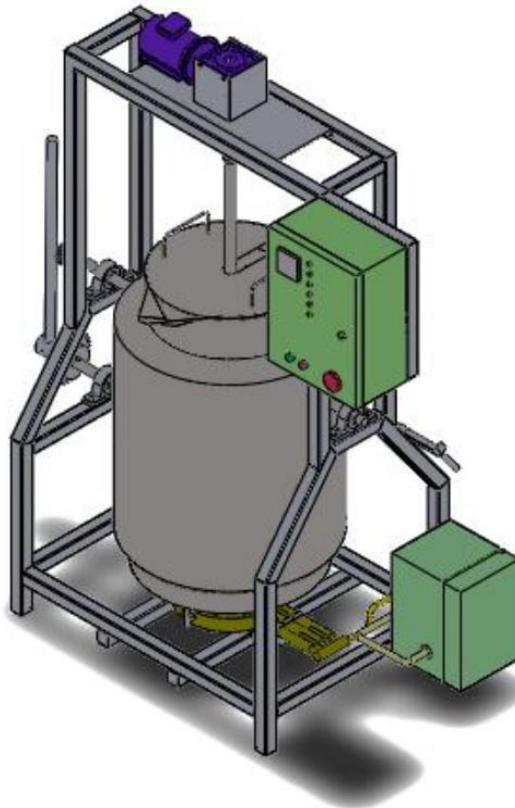


Figura 3.2. Prototipo de pasteurización VAT para procesar 50 litros de leche por hora.

Tomado de [62].

En el diagrama presentado en la Figura 3.3 se muestran las etapas del sistema de control, las cuales se dividen de la siguiente forma: etapa de control PID, etapa de potencia, instrumentos de medición, sistema mecánico y computadora (salida RS-232). En el diagrama se señalan los sistemas y etapas o instrumentos correspondientes, partiendo del sistema mecánico hasta el sistema de monitoreo y la etapa de visualización en la computadora. El sistema dispone de dos detectores de temperatura resistivos (RTD) de tipo Pt100, de modo que no se produjeran variaciones en la señal debido a un aumento en la resistencia de cables añadidos para la medición de temperatura con solamente un sensor de temperatura; un calentador de inmersión de 1500 W; un motor de 42W y 1/30 HP con un rango de 1000-7500 rpm a 120V-60Hz marca Talboys Engineering Corp para propulsar el agitador; un controlador tipo PID de Desin Instruments modelo LS-3400 para el control de temperatura; un controlador de potencia monofásico Eurotherm modelo 7100A para el manejo de la potencia del calentador; un dispositivo IoT de Particle modelo Photon alimentado por un chip Cypress Wi-Fi junto con un microcontrolador STM32 ARM Cortex M3; un convertidor de resistencia a digital modelo Max31865 para la lectura de las mediciones del Pt100 en el dispositivo IoT; una computadora para la visualización de los datos.

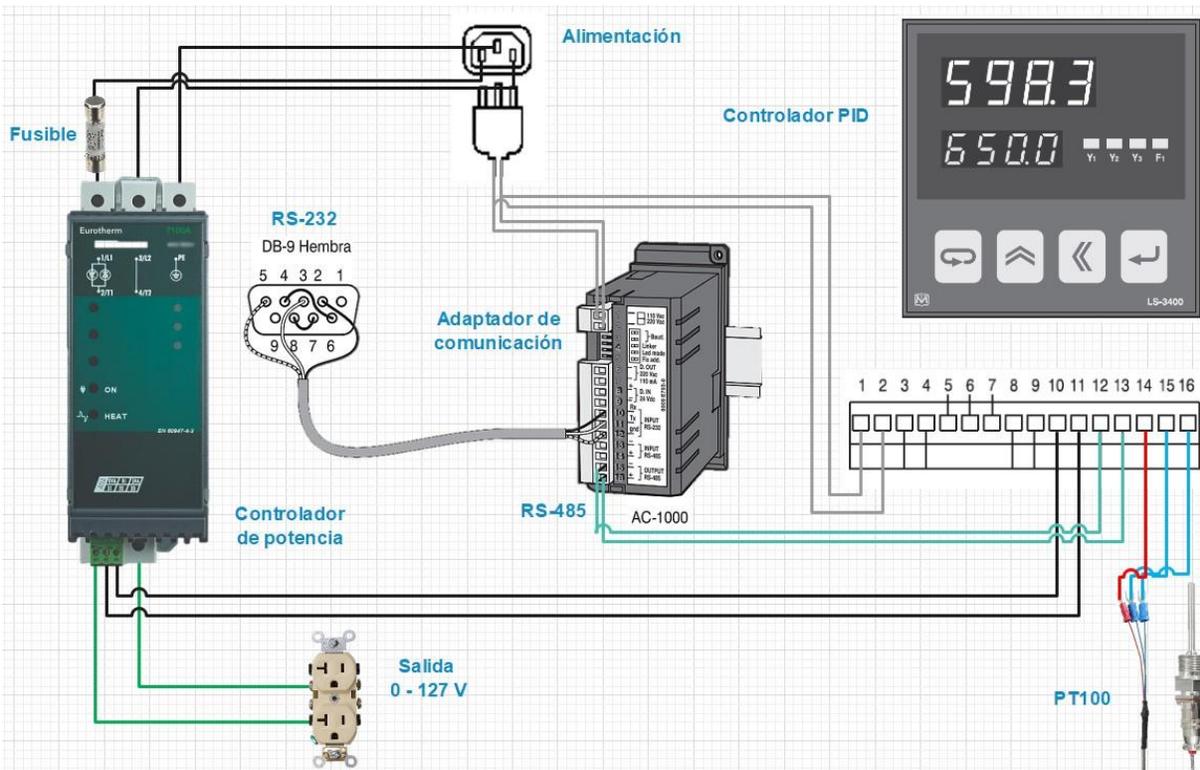


Figura 3.3. Esquema de conexión del sistema de control de temperatura de la leche.

3.1.1.1. Sensor de temperatura: RTD Pt100

Los RTD son sensores que han sido ampliamente utilizados, específicamente el PT100 es generalmente útil para la medición de temperaturas en el intervalo de $-200\text{ }^{\circ}\text{C}$ a $+800\text{ }^{\circ}\text{C}$, gracias a la linealidad que presenta en este intervalo [64]. El platino ofrece mayor linealidad y estabilidad que cualquier otro material. También tiene la ventaja de ser extremadamente resistente a la corrosión y al ataque químico y es inerte a altas temperaturas. Además de usarse el platino en termómetros de resistencia (RTD), también se usa en algunos termopares de alta temperatura [65]. La punta del Pt100 utilizado expuesta al medio líquido, en este caso leche, está compuesta de acero inoxidable para evitar corrosión y conseguir el funcionamiento esperado. El sensor se instaló dentro del sistema de pasteurización de modo que se encuentre completamente sumergido en el medio.

3.1.1.2. *Adquisición, procesamiento de los datos y control del proceso: Desin LS-3400*

A partir de la medición de la temperatura de uno de los Pt100, el controlador PID tiene la función de enviar una señal de control de tipo 4 mA a 20 mA para ajustar la salida de voltaje entregada por el controlador de potencia. Por medio del controlador de potencia sucede la regulación del calor entregado por el calentador de inmersión. Como resultado, el controlador PID, a través de la medición de temperatura y el controlador de potencia, efectúa un control de temperatura retroalimentado en la consigna asignada dentro de la configuración de parámetros. En este caso, la consigna es de 63 °C, con un intervalo de control de ± 1 °C, para la realizar el proceso de pasteurización de la leche según la norma [2].

3.1.1.3. *Adaptador de comunicación: Desin Instruments AC-1000*

El AC-1000 es un convertidor de comunicaciones de uso industrial que utiliza RS232/485 y el protocolo Modbus-RTU. Puede ser instalado en el riel DIN. Puesto que el conector del ordenador es un DB9 no se requirió modificación en el mismo. La mayor distancia permitida entre el convertidor y la computadora por normatividad es de 15 metros [66]. El adaptador convierte y transmite señales con una tasa máxima de 38.4 kilobaudios y un aislamiento de 1000 V [67]. El adaptador de comunicación AC-1000 conecta un dispositivo que se comunica mediante RS-485 a una computadora con puerto serial DB9 que permite la comunicación RS-232.

3.1.1.4. *Control de potencia: Eurotherm 7100A*

Un controlador de potencia monofásico emplea control de ángulo de fase para proporcionar una potencia de salida proporcional suave de 0 a 100 % a una fase, es decir, 0 a 127 V 60 Hz. El Eurotherm 7100A soporta un máximo de 250 A y 500 V. La salida del controlador PID le indica proporcionalmente de 0 a 100 % al controlador de potencia el voltaje suministrado al calentador eléctrico, el cual, acorde a esta proporción puede ajustarse de 0 a 500 V. Las entradas que permiten el control en este equipo son: 0 - 20 mA ó 4 - 20 mA, 0 - 5 V ó 0 - 10 V [68]. Para este proyecto, el dispositivo fue ajustado con entrada de 4 a 20 mA, con respecto a la salida disponible en el controlador PID, y salida de 127 V, puesto que el calentador utilizado funciona a estos valores de tensión.

3.1.1.5. *Calentador eléctrico de inmersión*

El calentador de agua empleado es uno de inmersión de 1500 W, utilizado generalmente en agua (ver Figura 3.4). Se compone de una cubierta de acero inoxidable 304, útil para evitar la oxidación y lograr un funcionamiento adecuado. Cuenta con un control de temperatura que corta la corriente cuando la temperatura del medio, en el que se encuentra sumergido, alcanza los 75 °C [69]. En el sistema de pasteurización, el calentador es conectado a la salida de 0 a 127 V del controlador de potencia, de este modo siendo controlado a través de la acción del controlador PID.



Figura 3.4. Imagen representativa del calentador de 1500 Watts de la empresa Hanaiette. Tomado de [69].

3.1.1.6. Mezclador de servicio liviano: Talboys Modelo 102

Mezclador útil para mezclas de baja viscosidad. Este tipo de mezcladores de servicio liviano ofrecen un par bajo para mezclas de uso general y tienen una clasificación de servicio continuo (ver Figura 3.5). Los mezcladores incluyen portabrocas y varilla de montaje de 8" (20,3 cm) con perno de carruaje y llave para montar el motor en soportes. Catalogados como de propósito general, estos mezcladores se consideran económicos, compactos y de alta durabilidad. El modelo 102 presenta control de velocidad básico montado en la parte superior, mayor par con reducción de engranajes de 10:1, la capacidad de dos ejes mezcladores para múltiples aplicaciones, mayor versatilidad y la mezcla de hasta dos galones de aceites livianos. El rango de velocidades va de 500 rpm a 7500 rpm mediante transmisión directa, mientras que con el mayor par (10:1) se reduce de 50 rpm a 750 rpm. Proporciona un torque de 0.15 in-lbs mediante transmisión directa y un torque de 1 in-lbs mediante la reducción 10:1. El mezclador se alimenta con 120 V con una corriente de 330 mA [70]. Las especificaciones completas del mezclador se encuentran en el Anexo 1.1.



Figura 3.5. Imagen representativa del mezclador de servicio ligero Talboys modelo 102.

Tomado de [70].

3.1.1.7. Impulsor de dispersión de disco

El impulsor dispersante seleccionado está hecho de acero inoxidable 304 (ver Figura 3.6). El disco de dispersión es pasivado con bordes afilados y superficie pulida después de ser formado. El disco y la varilla de agitación se fijan por una tuerca de brida 304 con una gran zona de sujeción y buena estabilidad. La barra de agitación de 8 x 300 milímetros está equipada con dos discos dispersantes de 3 mm y 8 mm. Para el caso de este proyecto se hizo uso del disco de 8 mm [71].



Figura 3.6. Imagen representativa del impulsor dispersante de disco. Tomado de [71].

3.1.2. Arquitectura del sistema de monitoreo

El monitoreo se realiza mediante la medición de la temperatura, que también es necesaria para el control. Con base en los factores a considerar en el proceso de pasteurización, el dispositivo IoT debe de tener una tasa de actualización más rápida que el controlador, ser de baja potencia, contar preferiblemente con conexión Wi-Fi y ser de bajo costo, entre otros requerimientos.

Adicionalmente a las capacidades que presente el Hardware de la plataforma IoT, se requiere que las librerías disponibles mediante la IDE permitan explotar el potencial de las funciones con las que cuenta por medio de sus diferentes pines. Mediante los pines de entrada analógica a digital y de Propósito General, un convertidor de resistencia a digital modelo Max31865, la librería correspondiente para la lectura de este convertidor y documentación para la configuración de la misma en la plataforma IoT, se busca lograr la interconexión correcta entre un Pt100 y el dispositivo IoT. En la Figura 3.7 se presenta un ejemplo del diagrama de conexiones del equipo de monitoreo con una de las tarjetas IoT de interés para el proyecto. En el diagrama se observa el sensor de temperatura tipo Pt100, que se encuentra instalado al sistema de pasteurización para las lecturas de temperatura de la leche, el convertidor Max31865 que adquiere la señal de salida del sensor y la convierte en una señal digital, y finalmente el dispositivo IoT que acondiciona la señal para desplegar y enviar el valor numérico de la temperatura a través de la nube.

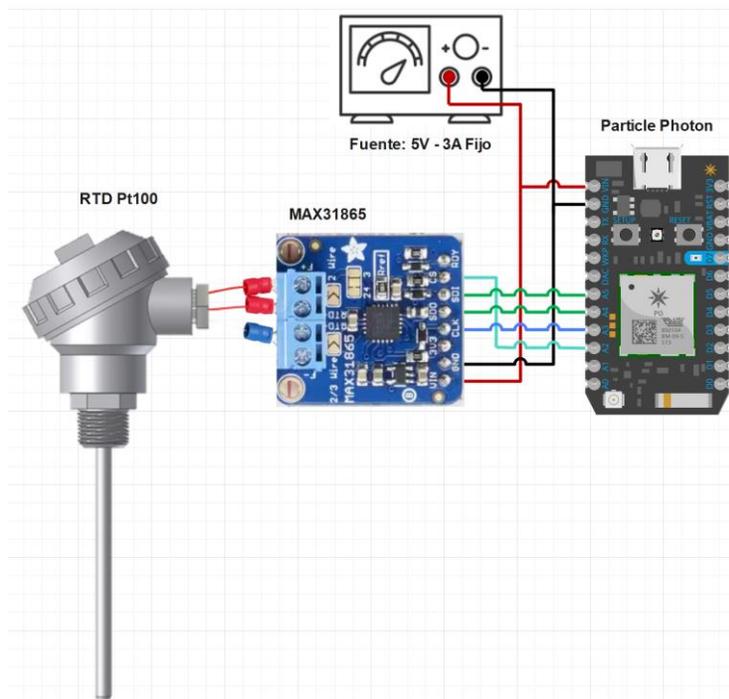


Figura 3.7. Esquema del conexionado del sistema de monitoreo de temperatura de la leche.

3.1.2.1. *Sensor de temperatura: RTD Pt100*

Del mismo modo que el sensor utilizado para el sistema de control, el RTD es instalado dentro del sistema para encontrarse completamente sumergido en el medio líquido. Las mediciones de la resistencia presentada son convertidas por el Max31865 para luego ser procesadas por el Photon.

3.1.2.2. *Acondicionamiento de la señal: Max31865*

El dispositivo Max31865 convierte resistencia en señales digitales y es ideal para detectores de temperatura (RTD) de platino (ver Figura 3.8). El funcionamiento comprende el uso de una resistencia que ajusta la sensibilidad del RTD y un ADC delta-sigma convierte el valor de resistencia en datos digitales con protección contra sobretensión y detección de problemas en el RTD o el cable [72].



Figura 3.8. Vista frontal del Max31865. Tomado de [73].

3.1.2.3. *Visualización en interfaz gráfica: Python*

La visualización de las mediciones obtenidas en los instrumentos puede lograrse de distintos modos según sean los requerimientos del sistema para hacer énfasis en

aspectos y parámetros en específico. Es bien sabido que elaborar una aplicación desde cero es una tarea compleja y que toma mucho tiempo. Por esta razón, es recomendable diseñar la interfaz mediante lenguajes de programación conocidos y con el soporte suficiente para llevar a cabo el desarrollo de interés. Durante la licenciatura se impartieron cursos de programación de C, C++, Matlab, LabView, R, Arduino y Python.

De estos lenguajes se destaca LabView y Python debido al soporte actual que poseen ambos para el desarrollo de interfaces gráficas de usuario. No obstante, LabView es software cerrado o de paga, por lo cual, Python supone mejor opción con base en las recomendaciones mencionadas.

3.1.2.4. Acceso a la nube

Cuando los datos se encuentran en la nube de Particle, es posible solicitarlos siempre y cuando se cuente con los códigos de identificación y seguridad del dispositivo de Particle. Con respecto a la identificación, se refiere a una serie de 24 caracteres que involucran números y letras y, por otro lado, el código de seguridad consta de una combinación de 40 caracteres, igualmente compuesto de números y letras. Adicionalmente se debe de especificar el nombre de la variable exactamente igual a como se encuentra publicada en la nube de Particle. Posteriormente, los códigos se deben introducir dentro de una liga que es denominada “liga de solicitud” para, a partir de la librería que corresponda a “HTTP requests” o “solicitudes HTTP” en el lenguaje de programación utilizado, extraer los datos de la nube.

Los recursos o datos en Internet o la nube se encuentran almacenados en diferentes servidores. Para acceder al contenido en Internet, el navegador debe solicitar a estos servidores los recursos que desea. Este protocolo de solicitudes y respuestas permite visualizar la página correspondiente en el navegador.

La transferencia de recursos se realiza a través del Protocolo de Control de Transmisión (TCP), que administra diversas conexiones en Internet cuando una computadora o dispositivo desea enviar datos a otra. HTTP (Protocolo de transferencia

de hipertexto) es el lenguaje de comandos que los dispositivos en ambos lados de la conexión deben seguir para comunicarse.

Se realiza una solicitud HTTP desde un cliente a un host ubicado en el servidor para recibir un recurso necesario para construir el contenido. Cuando realizan una solicitud, los clientes utilizan una URL (Localizador Uniforme de Recursos) que contiene la información necesaria para acceder a los recursos del servidor.

A partir de la URL es posible indicar en los servidores, donde se almacenan los recursos o datos de interés, la información, en este caso los códigos de identificación, seguridad y la variable, para acceder a ella de manera precisa. La URL empleada en las solicitudes de Particle se compone de la siguiente estructura general: `https://api.spark.io/v1/devices/ + deviceId + / + varName + /?access_token= + accessToken`. Dentro de los componentes que conforman a la liga URL, “deviceId” se refiere al código de identificación del dispositivo, “varName” es el nombre de la variable de la cual deseamos conocer su valor almacenado en la nube, “accessToken” es el código de seguridad o acceso creado específicamente para tener el pase o permiso a los datos que comparte el dispositivo.

En este proyecto, Python fue el lenguaje de programación utilizado para acceder a los datos y proveer una interfaz gráfica para su visualización. Python es versátil en cuanto a su accesibilidad en diversas plataformas, posee una amplia funcionalidad con el creciente número de librerías y soporte de las mismas, ofrece la posibilidad de realizar las solicitudes HTTP, y permite procesar los datos obtenidos para finalmente poderlos visualizar con el uso del mismo lenguaje. La librería para acceder a las funciones que se refieren a las solicitudes HTTP se llama “requests” y la documentación se encuentra en la siguiente liga: <https://pypi.org/project/requests/>.

3.1.3. Funcionamiento del sistema completo

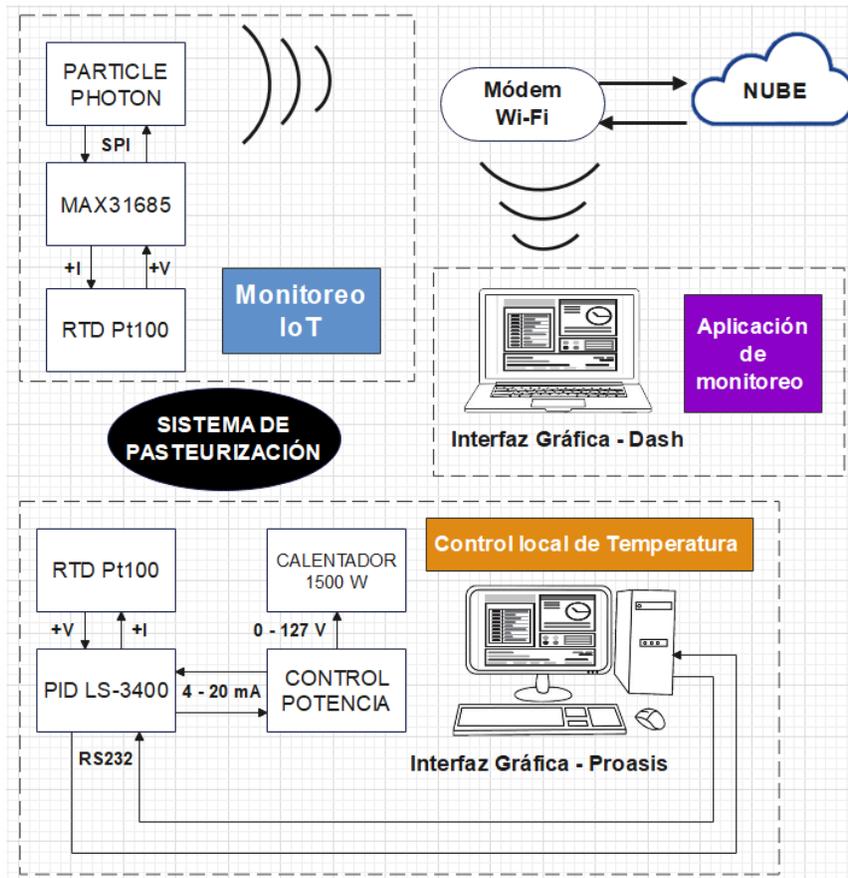


Figura 3.9. Diagrama de arquitectura del sistema de pasteurización completo.

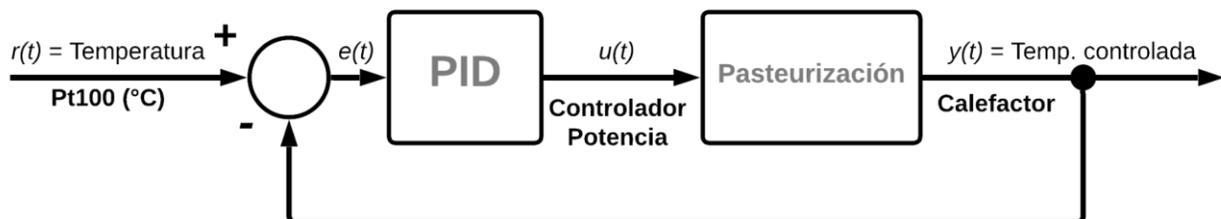


Figura 3.10. Diagrama del funcionamiento del control PID en el proceso de pasteurización.

Se desarrolló la arquitectura del sistema completo combinando las funciones de los tres sistemas presentados. En el esquema de la Figura 3.9 se presenta la arquitectura de todas las etapas que componen al sistema entero. En el diagrama, el sistema de pasteurización se refiere a la estructura metálica que contiene los sensores de temperatura, el calentador, el agitador, que son instrumentos útiles para el proceso de pasteurización, asimismo al contenedor en donde se vierte la leche para el proceso. El sistema de monitoreo y de control cuentan con un sensor RTD Pt100 de temperatura para la medición, la señal de salida de los sensores se dirige directamente a la etapa o dispositivo de procesamiento para posteriormente realizar la acción respectiva. En el caso del Particle Photon, a diferencia del controlador LS-3400 de Desin, se requiere de una resistencia de referencia para establecer la sensibilidad correcta, lo que se logra incluyendo al convertidor Max31865 en el proceso de adquisición en este sistema. En ambos sistemas se cuenta con el uso de una computadora, sin embargo, es en este punto donde residen las diferencias más importantes. El control de temperatura es local, el controlador PID requiere de una conexión al módulo de comunicación de la misma compañía para conseguir la conversión de la comunicación de RS-485 a RS-232. El protocolo de comunicación se adapta a RS-232 de modo que la computadora, que posee los programas de Proasis y Loop-Win para la configuración y monitoreo del controlador, sea capaz de adquirir y leer correctamente la señal. La adquisición de los datos en la computadora posibilita la visualización de los mismos mediante el software que provee la compañía, asimismo es posible configurar con mayor facilidad y rapidez el controlador mediante la computadora.

No obstante, el controlador no depende del uso de la computadora, puede funcionar sin la conexión a esta y tampoco es necesaria para su configuración, puesto que el controlador puede ser igualmente programado por teclado. La necesidad de esta conexión y programa específico para visualizar los datos recibidos en el controlador se elimina con la implementación del sistema de monitoreo IoT.

En el sistema de monitoreo IoT, la computadora no requiere la conexión ni de un programa o requisitos igualmente específicos para visualizar los datos del proceso. Adicionalmente, la interfaz de visualización y los elementos de visualización pueden ser modificados y ajustados a las necesidades del usuario y con un menor nivel de complejidad.

Los requisitos de la computadora o dispositivo de visualización se reducen a contar con conexión a Internet, con la versión actual o superior de Python, con las librerías utilizadas en el código instaladas en su versión correspondiente o superior, y los archivos de la aplicación. La aplicación en este caso no interfiere con el proceso que realiza el Photon ni ejecuta acciones de control, es decir, varios dispositivos que cumplen con los requisitos pueden ejecutar la aplicación y monitorear el proceso al mismo tiempo. Es posible también realizar acciones de control o mandar instrucciones al Photon, de manera que desde la interfaz se lleve a cabo cierto grado de control IoT. En este proyecto, no se establecieron acciones de control mediante la aplicación, puesto que la regulación del proceso se ejecuta automáticamente a través del controlador PID. Sin embargo, se añadió una acción preventiva automática en el Photon para cortar el suministro del controlador de potencia en caso de superar la temperatura de 70°C en las pruebas de pasteurización LTLT.

El diagrama presentado en la Figura 3.10 representa el funcionamiento del control PID del proceso de pasteurización llevado a cabo por el sistema. La variable de referencia $r(t)$ corresponde a la temperatura de pasteurización, de donde el sensor de temperatura Pt100 registra el valor de temperatura actual, el error $e(t)$ indica al controlador PID la diferencia entre la variable de referencia y la variable controlada $y(t)$, con base en esta diferencia acorde al algoritmo PID de Ziegler-Nichols envía una señal de control $u(t)$ al controlador de potencia con el fin de ajustar la variable con el actuador (calefactor), lo que resulta en la variable controlada $y(t)$ a la salida.

3.2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

3.2.1. Controlador PID

La Figura 3.11 muestra la configuración física del controlador PID, la cual consta de una estructura donde se colocaron los módulos o etapas necesarias para la regulación y observación de la información del proceso. El controlador de potencia (A) se instaló en uno de los extremos para disponer de los cables de alimentación con mayor facilidad. En el medio se colocó el adaptador de comunicación (B) para aprovechar los espacios de la estructura y conseguir un menor distanciamiento con el controlador PID (C). Igualmente se ubicó al controlador PID en uno de los extremos para contar con mayor libertad en el manejo de conexiones y cables, puesto que es aquí donde se conectan los módulos y el sensor de temperatura.

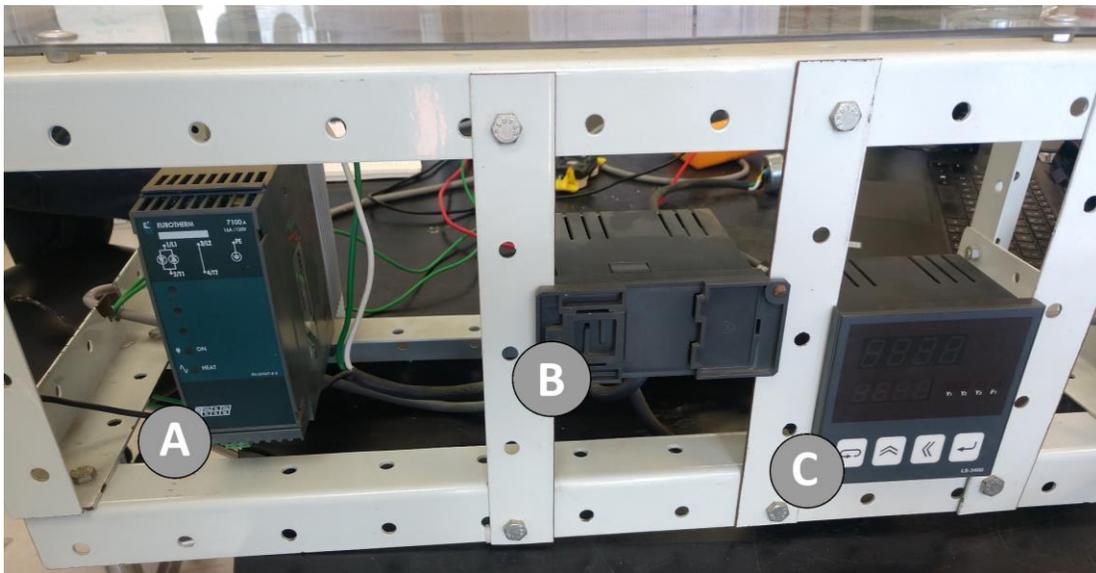


Figura 3.11. Configuración física del controlador PID. A: Controlador de potencia, B: Adaptador de comunicación, C: Controlador PAC PID.

Detalles de la conexión del controlador PID para lograr la comunicación entre los módulos se encuentra en el Anexo 1.2.

3.2.2. Tarjeta electrónica: Conexión de la tarjeta IoT al Max31865

La conexión fue realizada como se muestra en la Figura 3.12, en donde se montó el circuito en una protoboard para llevar a cabo las pruebas. En las Figuras 3.12, 3.13 y 3.14 se observa la conexión del sistema de monitoreo, se distingue el distingue la tarjeta de desarrollo IoT (A) en uno de los extremos de la placa, el Max31865 (B) al centro y a un costado de la placa las entradas y salidas de alimentación y tierra. La alimentación proviene de una salida constante de 5 V con máximo 1 A, de una fuente de alimentación regulada marca y modelo Tektronix PS280 (C). Se incluyó un relevador de estado sólido (D) para realizar la acción preventiva mencionada en el apartado anterior.

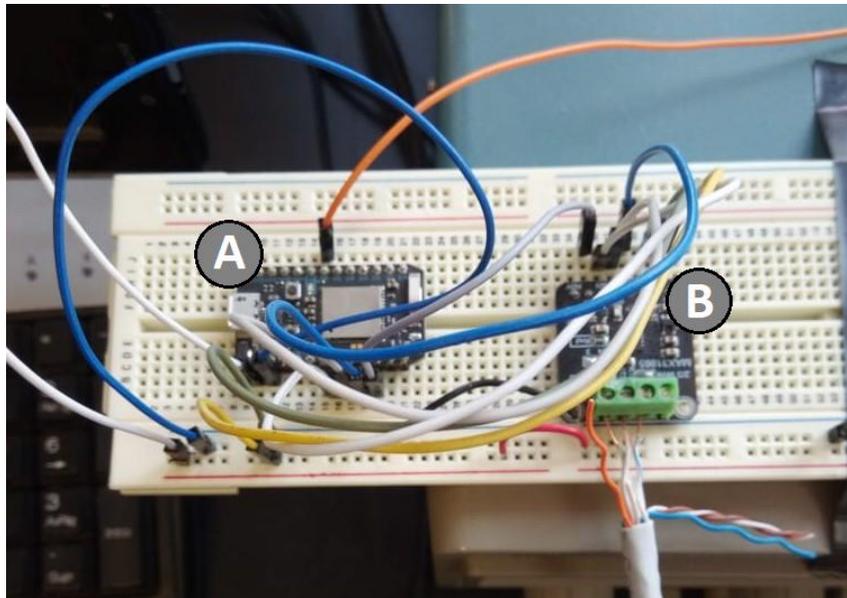


Figura 3.12. Conexión del sistema de monitoreo, vista superior. A: Tarjeta IoT, B: Convertidor Max31865.

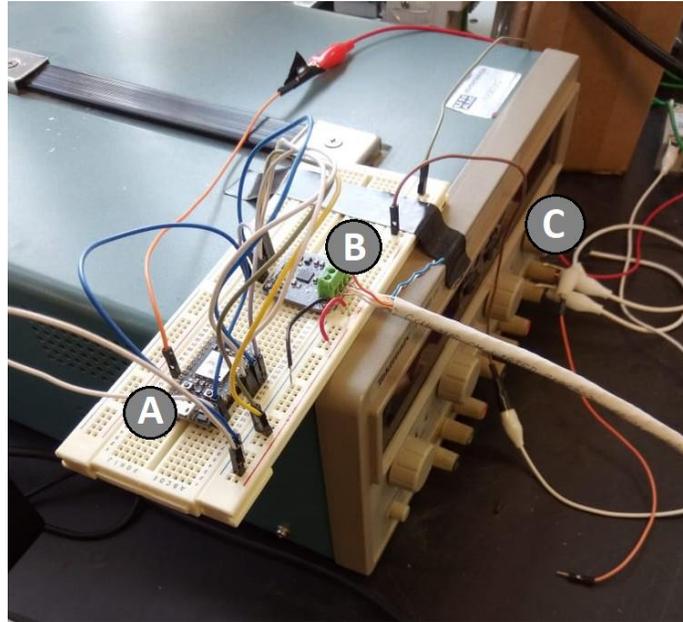


Figura 3.13. Conexión del sistema de monitoreo, vista lateral alzada. A: Tarjeta IoT, B: Convertidor Max31865, C: Fuente de alimentación regulada.

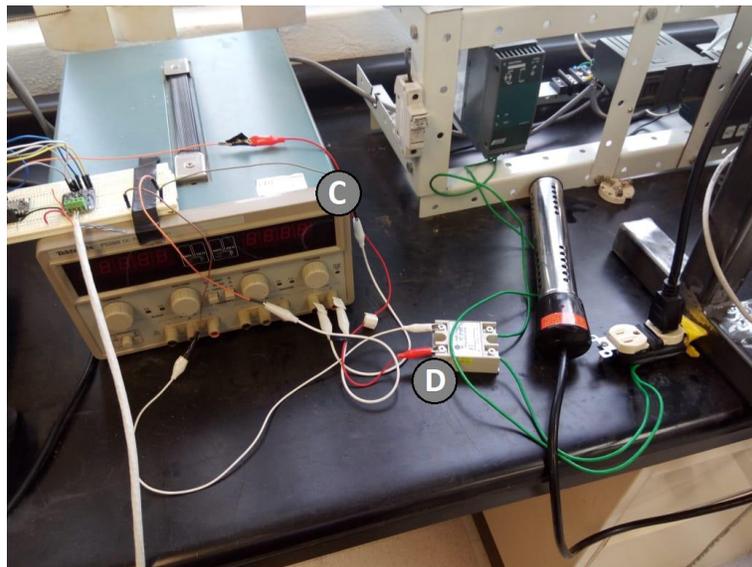


Figura 3.14. Conexión del sistema de monitoreo, vista lateral alzada ampliada. C: Fuente de alimentación regulada, D: Relé de estado sólido.

3.3. DESARROLLO EXPERIMENTAL

El agua y la leche presentan cierta similitud con respecto a características térmicas y reológicas. En el aspecto térmico es de interés el calor específico, capacidad calorífica específica, la conductividad y la difusividad térmicas, por el lado reológico la densidad y la viscosidad. Las tablas 3.1 y 3.2 muestran la comparativa de las características mencionadas entre el agua y la leche [74-76].

Tabla 3.1. Comparativa de propiedades térmicas del agua y la leche. Información tomada de [74-76].

Propiedades térmicas		
Propiedad	Agua	Leche (Entera)
Calor específico (kcal/kg °C)	1.00	0.93-0.94
Capacidad calorífica específica (kcal/kg °C)	1.00	0.94
Conductividad térmica [0-100 °C] (W/m °C)	0.57 - 0.67	0.53 - 0.61
Difusividad térmica [0-100 °C] (m ² /s)	1.35x10 ⁻⁷ - 1.65x10 ⁻⁷	1.3x10 ⁻⁷ - 1.5x10 ⁻⁷

Tabla 3.2. Comparativa de propiedades reológicas del agua y la leche. Información tomada de [74],[76].

Propiedades reológicas		
Propiedad	Agua	Leche
Densidad (kg/m ³)	997	1025 - 1035
Viscosidad (cP)	1	1.7 - 2.2

Debido a la similitud entre la leche y el agua en las características que tienen una influencia significativa en el proceso de pasteurización, se realizaron las pruebas del sistema con agua como el líquido a pasteurizar. Con esto en mente, se registraron los datos del proceso para medir la inercia térmica del sistema, así también la capacidad del sistema de alcanzar la temperatura de interés subiendo la temperatura y de mantenerla. Los parámetros que se variaron para determinar las aptitudes del sistema fueron temperatura, volumen y agitación.

La temperatura se modificó entre 63 °C y 72 °C, siguiendo el método VAT y HTST, para identificar la capacidad del sistema de alcanzar dicha temperatura y el tiempo necesario para lograrlo. La variación de temperatura se realizó en un volumen de 12 litros y uno de 16 litros, de aquí se puede extraer la relación de tiempo con respecto al volumen añadido para alcanzar el punto de temperatura deseado. Finalmente, se añadió el cambio en la agitación, pues es de interés conocer la importancia de un agitador en el proceso de pasteurización de este tipo. Tras llevar a cabo las pruebas con agua modificando estos parámetros, se realizaron únicamente dos pruebas con la leche, una sin agitación y otra con agitación.

Es de interés reconocer la similitud que suponen las propiedades del agua y la leche con respecto a las pruebas, es decir, la relación entre los datos y los resultados. Dada la complejidad de conseguir leche fresca para las pruebas, se diseñaron las pruebas en torno a un volumen de 16 litros a una temperatura de 63 °C, puesto que son las condiciones adecuadas para el diseño del sistema propuesto.

3.3.1. Control de temperatura: PID

El control de temperatura es realizado por el sistema PID LS-3400 de Desin Instruments. El sistema requiere de un ajuste de los parámetros para definir el método de operación y de ajuste para el algoritmo PID, las consignas o puntos de estabilidad, temporizadores, alarmas, operaciones lógicas y datos a desplegar en la pantalla del dispositivo. El ajuste puede ser realizado por medio del teclado o botones que ofrece el dispositivo, o por mediante el software especializado. El software de configuración se compone en primer lugar de PROASIS DCS-Win Configuración, y en segundo lugar por Loop-Win. La compañía recomienda utilizar el software, aunque ambos métodos conducen al mismo resultado. Es recomendado usar el programa especializado, puesto que es más rápido, intuitivo y gráfico, de modo que el usuario puede visualizar en todo momento la configuración completa que se encuentra programando. En la configuración de teclado, los parámetros se cambian y posteriormente se selecciona guardar la configuración realizada. Mediante PROASIS DCS-Win Configuración se definen los instrumentos, entradas y salidas que se desean habilitar, asimismo la comunicación para realizar la transmisión de los datos. La configuración realizada en este programa se puede probar, de modo que se despliega una ventana que muestra la información recibida del dispositivo acorde a los módulos habilitados. Si la configuración no fue realizada exitosamente, el programa indica las fallas y se debe repetir el proceso. La Figura 3.15 presenta las ventanas de simulación y comunicaciones del programa, en donde se reconoce que se puede revisar el funcionamiento de diferentes componentes: entrada analógica, salidas digitales, alarmas, integrador y salida analógica.

Adicionalmente, la configuración de las comunicaciones comprende el puerto del cual se dispondrá, la velocidad de Baudios y el tipo de ciclo que se desea implementar. En dicha ventana se encuentran seleccionadas las opciones utilizadas para este proyecto.

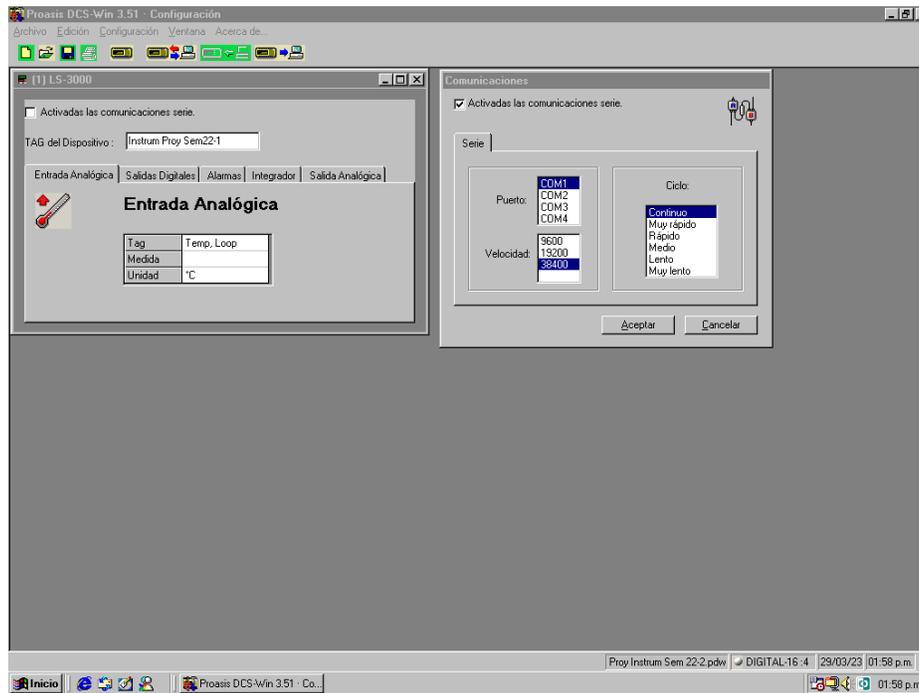


Figura 3.15. Captura de las ventanas de simulación y comunicaciones de PROASIS DCS-Win Configuración.

3.3.2. Adquisición de datos acondicionados

El dispositivo IoT de monitoreo se configura a partir de la programación que se envía a memoria por medio de la IDE. Para el proceso de monitoreo del sistema de pasteurización el dispositivo IoT requiere recibir la señal del Pt100, procesarla para determinar la temperatura en grados Celsius ($^{\circ}\text{C}$), revisar si es requerido abrir el circuito debido a un fallo en el control, enviar los datos a la nube y esperar un tiempo determinado, según la frecuencia de reloj del dispositivo y la frecuencia de actualización de la interfaz gráfica, para comenzar el ciclo de nuevo. Se requiere que el tiempo de actualización del dispositivo sea menor al de la aplicación, puesto que, de esta manera la aplicación no debe esperar más tiempo del programado para recibir datos nuevos.

4. RESULTADOS Y DISCUSIÓN

4.1. DEFINICIÓN DE PARÁMETROS DEL CONTROLADOR PID

En la configuración mediante Loop-Win, los parámetros se cambian y la configuración se guarda, pero a diferencia del teclado, la configuración se guarda en forma de archivos legibles por el programa. De este modo, se puede tener una carpeta de configuraciones distintas a disposición, por lo que se agiliza en gran medida este paso por medio del uso del software. Puesto que la configuración en Loop-Win se almacena en la computadora, se cuenta con una función para enviarla al dispositivo. Esta opción le permite al usuario revisar la programación antes de enviarla y guardarla en el dispositivo. Es importante tener en cuenta que, debido al protocolo de comunicación que se utiliza entre la computadora y el controlador PID para realizar la configuración, se deben detener los programas que estén haciendo uso del canal de comunicación. Es frecuente que se haga uso del PROASIS DCS-Win o dentro de PROASIS Configuración se realicen pruebas del dispositivo, es requerido detener y cerrar estos programas para que sea posible enviar la configuración de Loop-Win al LS-3400.

El esquema de configuración de parámetros del controlador se presenta en la Figura 4.1. El esquema en forma de diagrama de bloques se lee de izquierda a derecha, comenzando por la adquisición de los datos mediante la entrada analógica, que en este caso es la medición de temperatura del Pt100. Posteriormente, la señal se envía a los módulos de alarma y PID. Por un lado, el módulo de alarma en máximo superior señala si la señal rebasa la consigna establecida, cuando sucede el módulo pasa de enviar una señal en bajo o apagado a una señal en alto o encendido.

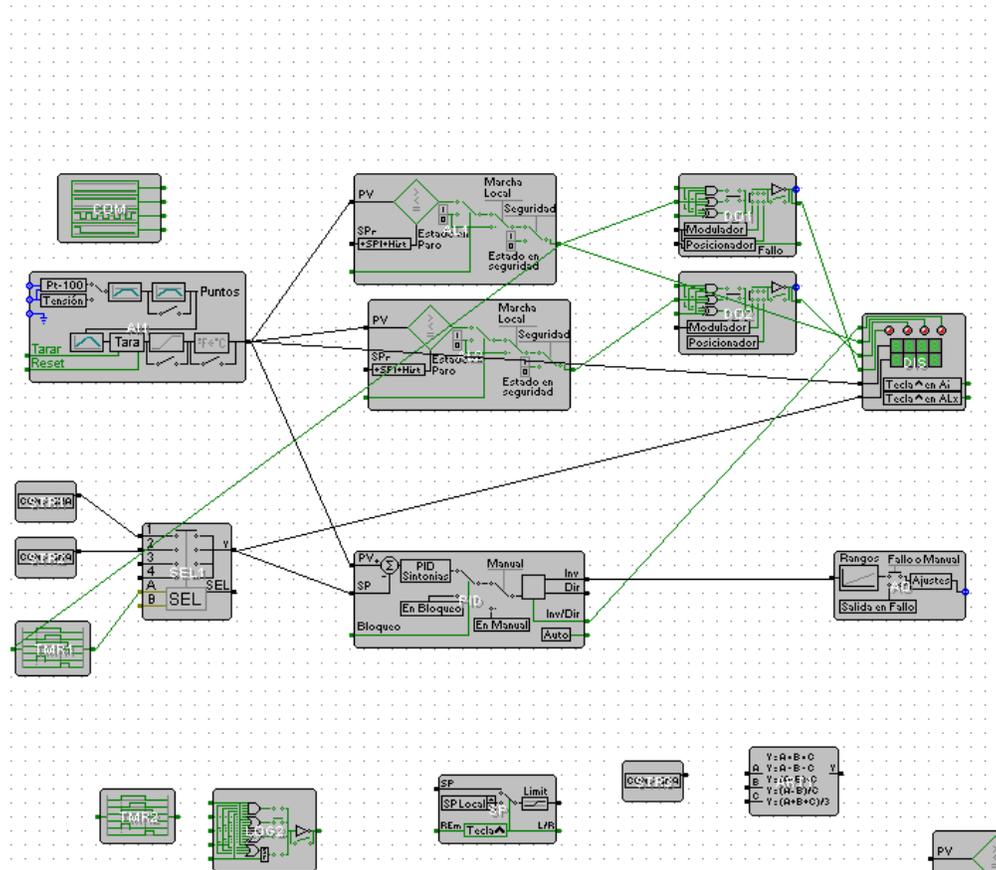


Figura 4.1. Esquema de la configuración implementada en Loop-Win.

La señal de encendido es de utilidad en este proyecto para marcar que ya se ha llegado a la temperatura de pasteurización. Dicha señal es recibida por un módulo temporizador, mientras que la señal esté en bajo el control PID se realiza con la consigna 1 del selector que es 63°C o 72 °C para las pruebas, cuando la señal está en bajo comienza el temporizador hasta terminar el tiempo y cambia el selector a la consigna 2 que es 30 °C. Puesto que se trata de pasteurización VAT, el temporizador tiene asignado un tiempo de 30 minutos, y 1 minuto para HTST, antes de mandar una señal en alto para cambiar en consecuencia el selector a la consigna 2. Puesto que solo se dispone de un calentador, pero no un sistema de enfriamiento, se realiza únicamente control inversor (indicado como "Inv"), que se refiere a la acción de calentar y también debe de asignarse en la configuración del módulo PID.

El control efectuado es de tipo PID siguiendo el primer método de Ziegler-Nichols debido a la forma de tipo S de la curva que se consigue con el proceso de control de temperatura de pasteurización.

La salida o respuesta del controlador es enviada directamente a la salida analógica, que se traduce en el controlador de potencia para realizar la acción de calentar. El control inversor también indica que una vez que se termina el proceso de pasteurización, el sistema no cuenta con la posibilidad de realizar el proceso de enfriamiento, por lo que, se debe de proceder a trasvasar el líquido a otro contenedor o sistema en donde se pueda realizar esta función. En la última sección del diagrama de bloques se visualiza la configuración que se tiene de la pantalla del dispositivo, en donde se indica cuáles señales se despliegan y qué LEDs o focos representan dichas señales. En la pantalla del controlador se cuenta con información importante del proceso, sin embargo, dicha información representa el estado del sistema en cada segundo.

Si se requiere un seguimiento continuo del estado del sistema a lo largo del proceso, además de revisar gráficos y descargar los históricos, es necesario hacer uso de PROASIS DCS-Win. El programa de PROASIS DCS-Win no es necesario para el funcionamiento del dispositivo, pero sí lo es para tener acceso a la información almacenada del proceso o de los procesos que se hayan realizado con el dispositivo. En la Figura 4.2 se muestra la interfaz gráfica de PROASIS DCS-Win. La interfaz gráfica se presenta en la computadora con el uso de la información adquirida mediante la comunicación RS-232. Se cuenta con distintas opciones para desplegar la información en forma de gráficos o exportar los datos registrados en los históricos. En “Archivo” se guarda o abre la configuración de visualización actual. “Visualizar” despliega opciones de las ventanas que se desea visualizar en la interfaz. Mediante “Históricos” se configura el registro y la presentación de los datos. En “Parámetros” se seleccionan los parámetros que se desean observar, es decir, alarma, entrada analógica, salidas digitales, entre otros.

“Gráfico” presenta opciones de visualización del gráfico que muestra los datos en forma de curvas, por ejemplo, rango en el eje de tiempo, rango en el eje de intensidad y curvas presentadas. Finalmente “Ventana” muestra opciones de visualización de cada ventana. La interfaz gráfica permite conocer la información del estado actual del proceso registrada en el controlador y también del estado durante todo el lapso de tiempo que comprende el proceso. Las ventanas superiores mostradas en la Figura 4.2 indican los datos de entrada y salida analógica, es decir, temperatura y control de temperatura. En la ventana izquierda la información es indicada por minutos por escalas de tiempo pequeñas, por el lado derecho la información es indicada en una escala de tiempo mayor para ser capaces de visualizar dichos datos en el periodo que comprende todo el proceso. La ventana inferior izquierda presenta la visualización de los históricos del proceso, mostrando el día, la hora y el valor de la entrada y salida analógicas. En las ventanas inferiores restantes, se encuentran los datos que se reflejan en el monitor del controlador. Esta información se refiere al estado actual del proceso en forma de alarmas y medidas de las entradas y salidas analógicas.

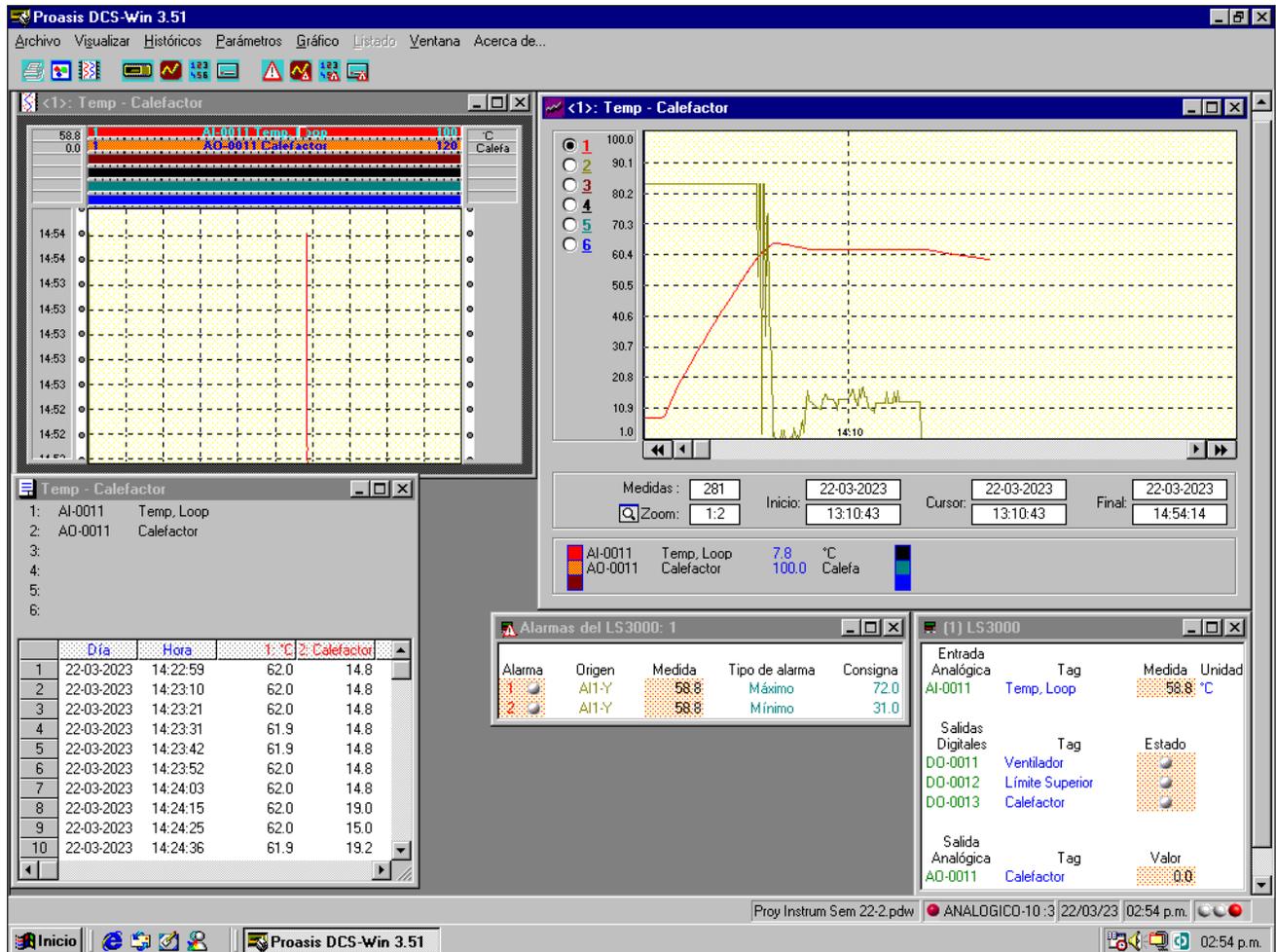


Figura 4.2. Captura de la interfaz de monitoreo del controlador PID.

Más información de los programas de los programas de parametrización y ajuste del controlador PID se encuentra en el Anexo 1.3.

4.2. CONSTRUCCIÓN DE INFRAESTRUCTURA

Con base en el diseño presentado por Tipán y Flores [62] y un bajo presupuesto, se adquirió un contenedor de leche de aproximadamente 17 litros y con respecto a este se elaboró un diseño de la estructura que lo soportara y sujetara los instrumentos de medición y control. En la Figura 4.3, se puede ver una vista en 3D del modelo de armazón planteado con el uso de herramientas CAD (Computer-Aided Design). El resto de las vistas y el dibujo con las cotas del modelo se encuentran en el Anexo 2. En primer lugar, con la adición del agitador se decidió montar al contenedor en la estructura y colocar bases de goma en los postes o columnas de soporte para disminuir la vibración transmitida al suelo. Posteriormente, se pensó en el montaje de los sensores de temperatura. Dado que los sensores son Pt100 que disponen de una estructura cilíndrica con cuerda milimétrica, se optó por instalar una ménsula con solo un barreno para fijar los RTD. El propósito de las barras cruzadas es de sostener los RTD y el agitador. La barra inferior soporta los RTD, mientras que la superior al agitador, de donde se puede visualizar un barreno para colocar el tornillo de sujeción. La estructura se compone de dos secciones, una inferior y una superior, de este modo evitar la necesidad de desensamblar los instrumentos cuando se requiera hacer mantenimiento, revisiones o cambios. La unión de ambas secciones se halla en los laterales de las columnas en forma de escuadras. Cada columna cuenta con un par de escuadras, donde se fijan los tornillos con sus respectivas rondanas para evitar el movimiento o caída de la sección superior de la estructura. El resultado final de la construcción de la infraestructura del sistema se visualiza en la Figura 4.4.

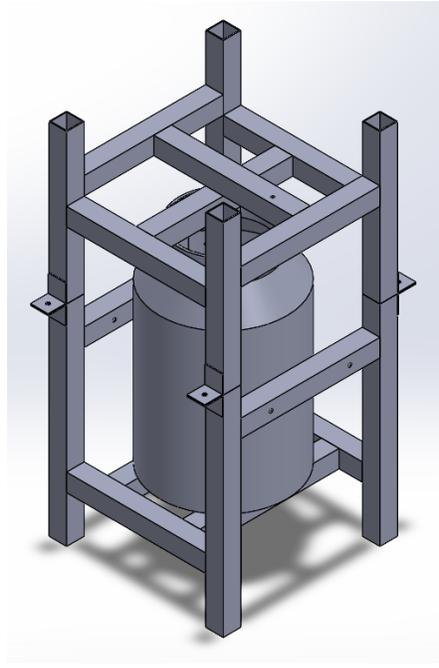


Figura 4.3. Vista isométrica del modelo de la estructura de soporte propuesta para el sistema de pasteurización.



Figura 4.4. Fotografía de la estructura de soporte terminada con los instrumentos de medición y el agitador montados.

4.3. COMPARATIVA Y SELECCIÓN DE HARDWARE IOT

En relación con los requerimientos del proyecto se determinan las características necesarias que debe incluir la plataforma IoT. Es con base a la literatura planteada en el marco teórico sobre los dispositivos IoT que se realizó la consulta y selección de la plataforma IoT más adecuada para el sistema de monitoreo del proceso de pasteurización. Acorde a la literatura, se presentan las plataformas más populares y avanzadas para la creación de prototipos y dispositivos inteligentes, y el desarrollo de proyectos y productos. Tewari et al. [27] exponen en la Tabla 4.1 las características de cinco plataformas IoT distintas de gran utilidad para aplicaciones específicas.

Tabla 4.1. Cuadro comparativo de tarjetas de desarrollo IoT. Extraído de [27].

Parámetros	Particle.io - ARGON	ESP8266-01 WiFi	Adafruit Feather 32u4 Bluefruit LE	Arduino 101	Raspberry Pi 4
Procesador	ARM Cortex-M4F procesador 32-bit @ 64 MHz	L106 32 bit Núcleo de microprocesador RISC basado en Tensilica Xtensa Diamond Standard 106Micro corriendo a 80 MHz	ATmega32u4 @ 8 MHz con 3.3 V lógico/energía	Intel CUrie x86 (Quark) y un núcleo de arquitectura 32-bit ARC	Broadcom BCM2711 cuatro núcleos Cortex-A72 (ARM v8) 64-bit SoC @ 1.5 GHz
Voltaje operativo	3.6 VDC a 5.5 VDC	3 a 3.6 V	3.7 V	3.3 V (E/S resistente a 5 V)	5 V
Velocidad de reloj	120 MHz	26 MHz - 52 MHz	8 MHz	32 MHz	1.5 Ghz

RAM	256 KB	-	2 KB	24 KB	1 GB, 2 GB, 4 GB
Memoria Flash	4 MB	512 KB a 128 MB	32 KB	196 KB	-
EEPROM	4096 bytes	-	-	1024 bytes	512 KB
Comunicación	IEEE 802.11 b/g/n	IEEE 802.11 b/g/n WiFi	BTLE	Bluetooth LE	2.4 GHz y 5 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
Soporte de Lenguaje de programación	C/C++, ARM assembly	Micro Python, Circuit Python, C, C++	Python	Python, C/C++	Python, Java, C/C++, Node.js
Conectividad E/S	20 GPIO señal mezclada (6 analógicas, 8 PWM – Pulse Width Modulation), UART, I2C, SPI	SDIO 2.0, (H) SPI, UART, I2C, I2S, IRDA, PWM, 16 GPIO PINS	7 pines PWM, 10 entradas analógicas, 20 pines GPIO	14 (4 proveen salida PWM)	Raspberry Pi, 40 pines estándar, USB - 2 puertos USB 3.0, 2 puertos USB 2.0
Precio	~\$615 MXN [77]	\$115 MXN [78]	\$860 MXN [79]	Discontinuado [80]	Desde \$1115 a \$2,352 MXN [81]

A partir de la comparativa y análisis de las características de las plataformas se extrae que Argon de Particle es útil para la programación de código simple y creación de prototipos tipo IoT. Consiste en un dispositivo de bajo consumo, hardware de bajo costo, alta confiabilidad de la red y comunicaciones locales que garantiza una transmisión de mensajes rápida y estable. Tanto el ESP8266 como el Arduino 101 tienen un diseño básico de equipo y se utilizan en aplicaciones relacionadas al registro de datos de sensores y el control de actuadores desde aplicaciones de trabajadores en línea.

El Adafruit Feather 32u4 Bluefruit LE es útil en caso de requerir asociar un dispositivo con un teléfono celular, puesto que cuenta con la capacidad de conectar muchos dispositivos USB, además de HID (Human Interface Device) o MIDI (Musical Instrument Interface Device). Por último, el Raspberry Pi se clasifica como una SBC, por lo que es la plataforma con el apoyo más robusto y por ello con el precio más elevado. Tewari et al. [27] indican que es ideal para aplicaciones IoT de trabajo, puesto que cuenta con mucho espacio adicional, RAM y un procesador de gran capacidad, y admite muchos lenguajes de programación que pueden crear aplicaciones del lado del trabajador, como Node.js. Adicionalmente ofrece la conexión a internet tanto por LAN como también mediante Wi-Fi.

Por otro lado, Singh y Kapoor [24] comparan lo que consideran las plataformas más populares y avanzadas, que rápidamente se han convertido en la primera opción de varios desarrolladores de IoT. Los autores contemplan prioritariamente el enfoque hacia la creación inicial de prototipos, la creación de objetos inteligentes y el desarrollo de proyectos y productos del área IoT. En términos generales, los autores toman en cuenta tres aspectos: especificaciones, API (Application Programming Interface) abierta y hardware. Las especificaciones del proyecto forman parte de la primera y vital etapa en la selección de la plataforma que se utilizará para una aplicación en específico. Aspectos fundamentales del Hardware son el procesador/microcontrolador, la velocidad de reloj, GPIO, ADC/DAC, conectividad (Wi-Fi, Bluetooth o Ethernet), comunicación (I2C, UART, SPI) y el precio (Tabla 4.2).

Se recomienda una sesión previa de la revisión de los requerimientos de la plataforma IoT de acuerdo con las actividades que debe realizar en la aplicación específica.

Tabla 4.2. Cuadro comparativo de plataformas IoT en términos de computación.

Extraído de [24].

Características	Procesador /MCU	Unidad de procesamiento gráfico	Velocidad de reloj	Tamaño	Memoria	RAM	Voltaje de alimentación	Precio de lista
Tarjeta SparkFun Blynk	Tensilica L106 32-b	No	26 MHz	51 mm x 42 mm	4 MB	128 KB	5 V mediante micro-USB/C conector Li-Po y circuito de carga	US\$29.95
Arduino Yun	ATmega32u4 y Atheros AR9331	No	16 MHz y 400 MHz	73 mm x 53 mm	32 KB y 16 MB + micro-SD	64 MB DDR2	5 V vía micro-USB	US\$58
Raspberry Pi 3	Broadcom BCM2837 y ARM Cortex-A53 64-b Quad Core	VideoCore IV 300/400 MHz	1.2 GHz	85 mm x 56 mm	Micro-SD	1 GB LPD DR2	5 V vía micro-USB	US\$35

cloudBit	Freescall iMX233 (ARM926EJ- S core)	No	454 MHz	55 mm x 19 mm	Ranura Micro-SD con 4-GB micro-SD	64 MB	5 V vía micro- USB	US\$59. 95
Photon	STM32F205 120 Mhz ARM Cortex M3	No	120 MHz	36.5 mm x 20.3 mm	1 MB	128 KB	5 V vía micro- USB	US\$19
BeagleBone Black	AM335x ARM Cortex- A8	PowerVR SGX530	1 GHz	86 mm x 56 mm	4 GB 8-b eMMC, micro-SD	512 MB DDR 3	5 V vía micro- USB	US\$49
Pinoccio	ATmega256 RFR2	No	16 MHz	70 mm x 25 mm	256 KB	32 KB	5 V vía micro- USB/c onecto r Li-Po	US\$109
UDOO	Freescall iMX 6 ARM Cortex-A9 y Atmel SAM3X8E ARM Cortex- M3	Vivante GC 2000 para 3- D + GC355 para 2-D + GC320 para 2-D	1 GHz	110 mm x 85 mm	Micro-SD	1 GB DDR 3	12 V	US\$135
Samsung Artik 10	ARM A15x4 y A7x4	Mali-T628 MP6 core	1.3 GHz y 1.0 GHz	39 mm x 29 mm	16 GB	2 GB LPD DR3	3.4 - 5 V	US\$100

Dadas las características de las plataformas IoT más atractivas según Singh y Kapoor [24], el Photon se coloca como el dispositivo más adecuado para la creación inicial de proyectos que requieren pocos recursos de cómputo. El rendimiento que ofrece de acuerdo con su precio y tamaño supone una plataforma IoT muy versátil, siempre y cuando no se requiera procesamiento gráfico y el proyecto no represente un aumento considerable en los recursos computacionales en el futuro.

Posteriormente, el siguiente aspecto clave es la API abierta. Para evitar posibles bloqueos, es preferible seleccionar una plataforma IoT que esté provisto de estándares abiertos, soporte comunitario y bibliotecas abiertas. El uso de bibliotecas abiertas y soporte fácilmente disponible permite el desarrollo de aplicaciones en menos tiempo con un mejor uso de recursos y eficiencia. Finalmente, el hardware abierto se vuelve considerablemente importante en el desarrollo de prototipos iniciales y la transformación del proyecto en un producto. Una plataforma con esquemas abiertos y soporte de hardware siempre es útil para desarrollar una aplicación e implementar un producto, que necesita del diseño de una solución personalizada y rentable. El resto de especificaciones de hardware y software se revisan y comparan en las Tablas 4.3 y 4.4.

Tabla 4.3. Cuadro comparativo de plataformas IoT en términos de entornos de desarrollo y estándares de comunicación. Extraído de [24].

Características	Sistemas Operativos Soportados	IDE	Estándar de video	Puertos de video/audio	Puertos USB	Comunicación
Tarjeta SparkFun Blynk	No	Arduino, Blynk	No	No	No	I2C, SPI, UART
Arduino Yun	OpenWrt-Yun	Arduino	No	No	1x USB 2.0	I2C, UART

Raspberry Pi 3	Raspbian, Windows 10 IoT Core, OpenELEC, OSMC (Open-Source Media Center), Pidora, Arch Linux, RISC OC (Reduced-Instruction-Set Computer Operating System), Ubuntu	C#, Python, Java, Scratch, entre otros	MPEG-2, VC-1, H. 264 AVC (1080p @ 30 fps, frames per second)	HDMI 1.4 con CEC (Consumer Electronics Control), 4 conectores pole 3.5 mm, Raw LCD (Liquid Crystal Display) (DSI, Digital Serial Interface)	4x USB 2.0	1x SPI, 2x I2C, PCM (Pulse Code Modulation)/I2S, 1x UART
cloudBit	Distribución Arch Linux ARM personalizada	Cloud API, Arduino	No	No	No	UART
Photon	FreeRTOS	Particle Build (En línea), Particle Deb (Local)	No	No	No	2x SPI, 1x I2S, 1x I2C, 1x CAN, 1x UART
BeagleBone Black	Debian, Android, Ubuntu	C++, Perl, Python, Cloud9 IDE	Soporte de decodificación NEON (Nonprofit Enterprise Online Network) software	Micro-HDMI	1x USB	4x UART, 2x SPI, 2x I2C, 2x CAN (Controller Area Network) BUS

Pinoccio	No	Arduino, Pinoccio, HQ (En línea/Fuera de línea)	No	No	No	I2C, SPI, 2x UART
UDOO	UDOOuntu, Android, XMBC (Xbox Media Center), Yocto, Arch Linux, OMV (Open-Media Vault)	Arduino	MPEG-2, H.264 (1080p60)	HDMI, Audio análogo y jacks de mic, LVDS (Low- Voltage Differential Signaling) + Touch	1x USB OTG (On- The- Go), 2x USB 2.0, 1x USB a Serial	SPI, I2C, UART, CAN BUS
Samsung Artik 10	Yocto 1.6 (Fedora)	Arduino IDE, Samsung SDK (kit de desarrollo de software), C/C++, Java, Groovy	1080p @ 120 fps H.263/H.264/MPE G-4/VP8 + MPEG- 2	MIPI (Mobile Industry Processor Interface) DSI (Display Serial Interface) de cuatro canales hasta WUXGA (Wide Ultra- Extended Graphics Array), HDMI, un canal PCM y dos canales I2S	1x USB 2.0, 1x USB 3.0	1x SPI, 6x I2C, 1x I2S, 3x UART

Tabla 4.4. Cuadro comparativo de plataformas IoT en términos de conectividad.
Extraído de [24].

Características	GPIO	Pines PWM	Pines Analógicos	ADC/DAC	Ethernet	Wi-Fi	Bluetooth	Interfaz de cámara	Sensores y pantalla integrados
Tarjeta SparkFun Blynk	9	3	1	10 b	No	IEEE 802.11 b/g/n	No	No	Temperatura y humedad, LED RGB
Arduino Yun	20	7	12	10 b	IEEE 802.3 10/100 Mb/s	IEEE 802.11 b/g/n	No	No	No
Raspberry Pi 3	26	2	No	No	IEEE 802.3 10/100 Mb/s	IEEE 802.11 b/g/n	Bluetooth 4.1 LE	CSI	No
cloudBit	No	No	No	No	No	IEEE 802.11 b/g/n	No	No	No
Photon	18	9	8	12-b ADC, 12-b DAC	No	IEEE 802.11 b/g/n	Sí	No	Reloj en tiempo real
BeagleBone Black	69	8	7	12-b ADC	IEEE 802.3 10/100 Mb/s	No	No	No	No

Pinoccio	17	4	8	10-b ADC	No	Wi-Fi Backpa ck IEEE 802.11 b/g/n	No	No	Temperatur a, LED RGB
UDOO	76	13		12-b ADC, 12-b DAC	IEEE 802.3 10/100 Mb/s	IEEE 802.11 b/g/n	No	CSI	No
Samsung Artik 10	51	2	6	Seis canales	No	IEEE 802.11 b/g/n	Bluetooth 4.1 LE	1x dos pistas MIPI CSI, 1x cuatro pistas MPI CSI, 8 MP @ 30 fps	No

Es crucial alinear las especificaciones de los sistemas, dispositivos o plataformas a los requerimientos del proyecto, lo que indica un cierto grado de personalización según la especialización o complejidad del mismo. El sistema de monitoreo de pasteurización no precisa de sistemas o procesamiento de video o audio, tampoco de dispositivos con comunicación USB. Con respecto a la comunicación, se necesita de entradas equipadas con comunicación SPI para la transferencia de datos entre el sensor de temperatura Pt100 y la plataforma IoT. Puesto que el intercambio de datos no es demandante, no implica tiempos de respuesta menores a 5 segundos ni altas velocidades de transferencia, contar con comunicación Wi-Fi y Bluetooth es suficiente. En relación con el número de pines GPIO y analógicos, es útil contar con 4 o más en caso de un aumento en las capacidades de monitoreo y control en el sistema mediante futuras actualizaciones. Por último, es conveniente que la plataforma posea una IDE de tipo abierto, que admita la incorporación de librerías y funciones desarrolladas por la comunidad.

Una vez seleccionada las plataformas de acuerdo con el hardware, el siguiente paso es elegir con respecto al software de aplicación. Existe una amplia gama de compatibilidad con lenguajes de programación para aplicaciones de software como C, C++ y Python. Se ha vuelto imprescindible aprender estos lenguajes (C/C++) para el desarrollo de hardware IoT, así también para aplicaciones de middleware y desarrollo de API (Python, Java, NodeJS, NET) (Tabla 4.3). Agregados útiles para el diseño de interfaces gráficas, es útil conocer HTML5, CSS3, Java, kits de desarrollo de software (SDK) de Android y Javascript se catalogan como las mejores opciones. Por último, la selección de la plataforma IoT finaliza con la revisión de las plataformas de software disponibles.

En relación con la plataforma de hardware IoT existen plataformas de software compatibles. La elección de la plataforma de software óptima está determinada por los instrumentos proporcionados, el IDE para la programación y la API disponible para el acceso a los datos y las notificaciones. La API se relaciona con la integración y los protocolos disponibles en la plataforma correspondiente.

Adicionalmente, es importante tomar en consideración la eficiencia de las herramientas de gestión de datos, como aplicaciones móviles y de escritorio, los paneles y similares, el precio, el soporte de la comunidad y la documentación disponible. Las características de las distintas plataformas de software compatibles con las plataformas IoT mencionadas por los autores se presentan en la Tabla 4.5. Las plataformas comparadas en la tabla son de código abierto u ofrecen cuentas gratuitas.

Tabla 4.5. Cuadro comparativo de plataformas de software IoT. Extraído de [24].

Plataforma de software IoT	Integración	Protocolos	Analítica	Visualización	Precio	Plataforma de hardware compatible
Temboo	REST API, XML, JSON	MQTT (Message Queuing Telemetry Transport), CoAP (Constrained Application Protocol), HTTP	Microsoft Power BI, Google BigQuery	Transmisión, tablero, app móvil	Cuenta gratuita y planes de pago	Samsung Artik, Arduino, Texas Instruments
Kaa	REST API, Web UI (User Interface), JSON	MQTT, CoAP, XMPP, HTTP	Hadoop, mongoDB, Oracle, Cassandra, Spark, Couchbase, CDAP	IU administrativa, tablero, app móvil	“Open source” y gratuito	Intel Edison, BeagleBone, Raspberry Pi, Econais, LeafLabs, Texas Instruments
Carriots	REST API, XML, JSON	MQTT, HTTP	Microsoft Azure, IBM, Ducksboard, Nibodha	Tablero, app móvil	Cuenta gratuita y planes de pago	Arduino, Raspberry Pi
Cayenne	REST API, XML, JSON	MQTT, CoAP	Análisis en tiempo real	Tablero, app móvil	Cuenta gratuita y	Raspberry Pi

			dentro del tablero		planes de pago	
Ubidots	REST API, XML, JSON	UDP (User Datagram Protocol), HTTP, MQTT, CoAP	Distimo, Google Analytics	Tablero	Cuenta gratuita y planes de pago	Arduino, Raspberry Pi, Spark Core, Microchip WCM, Adafruit FONA
ThingSpeak	ThingSpeak API, XML, JSON	HTTP, MQTT	MATLAB	MATLAB, tablero, app móvil	“Open source” y gratuito	Arduino, Particle Photon & Core, Raspberry Pi, Electric Imp
Leylan	API del dispositivo, API física, API tipo	HTTP, MQTT, OAuth 2.0	IFTTT (apps de terceros)	App web, IFTTT (apps de terceros)	“Open source” y gratuito	Arduino Yun, Raspberry Pi, Electric Imp, Spark Core, Netduino, Texas Instruments
The ThingBox	Node Red	HTTP, MQTT	Emoncms	Emoncms	“Open source” y gratuito	Raspberry Pi
Smartliving	REST API, XML, JSON	HTTP, MQTT, Stomp	Análisis en tiempo real dentro del tablero	Tablero, app móvil	Gratuito	Arduino, Raspberry Pi, Intel Edison, LoRa
Artik Cloud	REST API, XML, JSON	HTTP, MQTT, CoAP, Websocket	Apoyo de muchos proveedores de servicios de terceros	Tablero, apps móviles, apps de terceros	Cuenta gratuita y planes de pago	Arduino, Raspberry Pi, Samsung Artik

Wylidrin	Wylidrin API, JSON	HTTP, MQTT	Análisis en tiempo real dentro del tablero y soporte de R	Tablero, app móvil, App Google Chrome	Cuenta gratuita y planes de pago	Raspberry Pi, Intel Galileo, Intel Edison, UDOO, BeagleBone Black, Arduino
----------	-----------------------	---------------	---	--	---	--

A partir de la información de la Tabla 4.5 se reconocen las integraciones y protocolos más comunes en aplicaciones IoT. En específico, plataformas de software como Ubidots y ThingSpeak pueden ser de gran utilidad debido al uso de integraciones presentes en Python, el uso de analítica de Matlab y Google, que son herramientas poderosas y familiares, las opciones de uso gratuitas con las que cuentan y la lista de compatibilidad con las plataformas de hardware de la que disponen.

En relación con el análisis de las diversas plataformas IoT, existen opciones útiles y atractivas en términos de hardware, software, soporte y precio. Para el caso del monitoreo de procesos que involucren pocas variables, periodos de actualización mayores a 5 segundos, uso de instrumentación que requiera de comunicación SPI, UART o I2C, servicio en la nube para desarrollar software (tipo PAAS) y un bajo presupuesto, Particle es una empresa que ofrece opciones que cuentan con estas características.

4.4. COSTOS DEL SISTEMA

En esta sección, se examinan los elementos requeridos, tanto en términos físicos como económicos, para llevar a cabo el proyecto. Se dividen los costos por sección del sistema completo en la Tabla 4.6, de este modo se busca conocer el costo total de cada una de ellas.

Tabla 4.6. Detalle de costos de materiales por sistema.

Infraestructura	
Descripción	Precio [MXN]
Estructura hecha con PTR de media pulgada de acero	\$ 2 500
Tanque para almacenamiento de leche de 17.5 L	\$ 700
Total	\$ 3 200
Sistema de control	
Descripción	Precio [MXN]
Controlador PID LS-3400 Desin Instruments	\$ 5 000 a \$ 7 000
Adaptador de comunicación AC1000 Desin Instruments	~ \$ 2 500
Controlador de potencia 7100A Eurotherm	~ \$ 2 800
Sensor de temperatura Pt100	\$ 338
Calentador eléctrico de inmersión de 1500 W	\$ 760
Mezclador de servicio liviano Talboys Modelo 102	~ \$ 2 700
Impulsor de dispersión de disco	\$ 840

Total	\$ 14 938 a \$ 16 938
Sistema de monitoreo	
Descripción	Precio [MXN]
Tarjeta IoT Photon Particle	\$ 800
Convertidor de resistencia a digital Max31865	\$ 190
Sensor de temperatura Pt100	\$ 338
Total	\$ 1 328

Es notorio que el costo del sistema de control sobresale por encima del costo total sumado del sistema de monitoreo y la infraestructura. Sin embargo, el enfoque del proyecto es el monitoreo de tipo IoT y el sistema de control puede ser desarrollado con tecnologías más asequibles en una actualización futura.

4.5. PARTICLE PHOTON

Una plataforma de IoT empresarial es una pila de tecnologías que permiten que los productos y dispositivos conectados a Internet envíen y reciban datos. IoT ayuda a las empresas a desbloquear nuevos valores de los activos físicos, utilizando datos para mejorar las ofertas de productos, generar nuevas fuentes de ingresos y administrar mejor los activos físicos. Según el caso de uso y la vertical, IoT puede ayudar a controlar los costos a través del mantenimiento preventivo y el monitoreo de activos, o proporcionar nuevos modelos de ingresos con servicios basados en suscripción que permiten ingresos recurrentes en lugar de compras únicas [82]. Este tipo de soluciones debe de incluir:

- Hardware (sensores, microcontroladores, periféricos)
- Sistema operativo (aplicaciones de usuario, capa de abstracción de hardware, bibliotecas de firmware)

- Conectividad (acceso a red inalámbrica, certificaciones, planes de datos)
- Solución en la nube (administración de dispositivos, almacenamiento de datos, servidores de aplicaciones)

Particle es una empresa con sede en San Francisco dedicada a proporcionar una plataforma IoT integrada como servicio que ayuda a las empresas a conectar, administrar e implementar aplicaciones de software en dispositivos conectados, desde el borde hasta la nube y viceversa. Más de 240,000 desarrolladores y más de 160 clientes empresariales se basan en Particle [83].

4.5.1. Photon

Uno de los dispositivos o tarjetas IoT insignia de la compañía es el Photon. El Photon es un diminuto dispositivo Wi-Fi IoT con un precio de lanzamiento de \$19 USD (aprox. \$350 MXN), diseñado para la creación de aplicaciones y soluciones conectadas al Internet de las cosas [24]. Con conexión a la nube de Particle, resulta sencillo de configurar y mantiene una excelente relación entre capacidad de procesamiento y costo [82].

Características del Photon [84]

Particle PØ módulo Wi-Fi

- STM32F205RGY6 120Mhz ARM Cortex M3
- Broadcom BCM43362 Wi-Fi chip
- 802.11b/g/n Wi-Fi
- 1MB flash, 128KB RAM

Especificaciones generales

- LED de estado RGB incorporado
- 18 GPIO de señal mixta y periféricos avanzados
- Sistema operativo en tiempo real (FreeRTOS)
- Diseño de código abierto

- Configuración suave de AP
- Certificación FCC, CE e IC

El diagrama del diseño de los pines del Photon con las funciones correspondientes a cada uno se encuentra en el Anexo 1.4.

4.5.2. Software - Entorno de desarrollo (programación)

Particle Workbench es el nombre del entorno de desarrollo integrado (IDE en inglés) de la compañía. Workbench proporciona todas las herramientas, bibliotecas y extensiones necesarias para el desarrollo de IoT en un solo paquete fácil de instalar. Utiliza Visual Studio Code de Microsoft como editor, lo que permite crear, compilar e implementar código para productos de Particle [85].

Incluye:

- Cadena de herramientas gestionada: Admite compilación fuera de línea y en la nube, además de actualizaciones Over-The-Air (OTA) inalámbricas
- 3000+ bibliotecas: Sistema Operativo del dispositivo para un desarrollo más rápido
- Totalmente personalizable: Personaliza el flujo de trabajo con más de 8000 extensiones de Visual Studio

4.5.3. Centro de comando

Particle se refiere al centro de comando como la interfaz que ofrece para administrar los dispositivos de la compañía. La consola de dispositivos brinda funciones una interfaz en la nube y una API Rest para administrar y monitorear los dispositivos IoT [86]. Es posible realizar las actualizaciones de firmware de los dispositivos IoT aprovechando herramientas que ofrece la interfaz:

- Conexión directa a la nube: no requiere configuración, las actualizaciones de firmware son parte de la plataforma.

- Sin interrupciones: permite indicar si el dispositivo se encuentra listo para la actualización o si se encuentra en uso.
- Despliegue rápido: las actualizaciones de firmware pueden ser enviadas a los dispositivos de manera inmediata.
- Seguridad: la plataforma ofrece encriptación entre el Sistema Operativo (SO) y la nube de Particle.

4.5.4. Modelos de servicio en la nube

La computación en la nube ha sido un servicio, cuya popularidad y adopción ha ido en aumento por las herramientas y nuevas posibilidades que ofrece. No solamente se refiere al almacenamiento de los archivos sino a la administración o disposición que se tiene de los mismos.

Amazon define la computación en la nube de la siguiente forma:

La computación en la nube provee a los desarrolladores y departamentos de diversos recursos para evitar arduas tareas como el aprovisionamiento, el mantenimiento y la planificación de capacidad. Con el crecimiento de la informática en la nube se han desarrollado varios modelos y estrategias de implementación para satisfacer las necesidades de los usuarios. Cada tipo de servicio en la nube y método de implementación aporta distintos niveles de control, flexibilidad y administración [87].

Es recomendable que un diseñador de dispositivos o aplicaciones entienda la diferencia entre los distintos tipos de servicios. Cada servicio además cuenta con diferentes estrategias de implementación. Los creadores de productos o servicios deben analizar cada opción con su estrategia de implementación para identificar la mejor combinación de soluciones que se ajuste a los requerimientos del proyecto. Los distintos niveles de servicios en la nube se definen a continuación y pueden observarse de manera gráfica en la Figura 4.5.

1. Infraestructura como Servicio (IaaS)

El servicio IaaS es una oferta en la nube que proporciona componentes de infraestructura de TI, como servidores virtuales, almacenamiento y redes, de manera flexible y escalable. Los usuarios pueden gestionar y controlar estos recursos de manera más independiente que en otros modelos de servicio en la nube, lo que les permite adaptar su infraestructura según sus necesidades sin invertir en hardware físico. Este modelo es especialmente útil para empresas que desean mantener un mayor control sobre su infraestructura de TI y desean escalabilidad bajo demanda.

2. Plataforma como servicio (PaaS)

El servicio PaaS es una oferta en la nube que proporciona una plataforma de desarrollo y despliegue de aplicaciones. Incluye herramientas, servicios y entornos de desarrollo que permiten a los desarrolladores crear, probar y lanzar aplicaciones sin preocuparse por la infraestructura subyacente. Los usuarios de PaaS se benefician de una mayor eficiencia en el desarrollo de aplicaciones y la capacidad de escalar fácilmente, ya que la plataforma se encarga de tareas como el mantenimiento y la administración de servidores y bases de datos. Esto acelera el tiempo de desarrollo y permite a las organizaciones centrarse en la creación de aplicaciones de manera más ágil.

3. Software como servicio (SaaS)

El servicio SaaS es una modalidad de la nube que ofrece aplicaciones de software a través de internet. Los usuarios acceden a estas aplicaciones a través de un navegador web, sin necesidad de descargar ni instalar software localmente. SaaS se encarga del alojamiento, mantenimiento y actualizaciones del software, lo que permite a las organizaciones utilizar aplicaciones sin preocuparse por la infraestructura tecnológica subyacente. Este enfoque brinda flexibilidad, escalabilidad y accesibilidad a las aplicaciones, simplificando su gestión y permitiendo un uso eficiente.

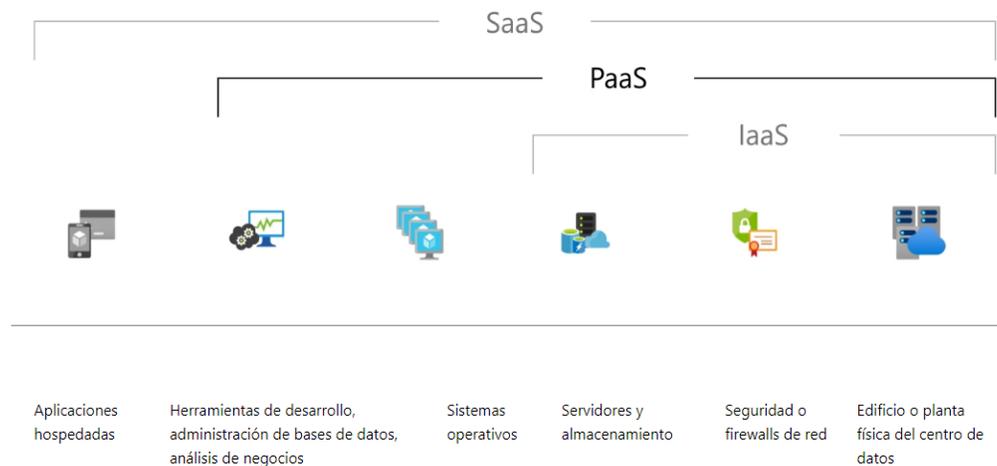


Figura 4.5. Esquema de los distintos servicios de la nube. Tomado de [88].

Particle ofrece un servicio tipo Plataforma como Servicio (PaaS) de pila completa que facilita crear productos conectados a internet acelerando el crecimiento y optimizando las operaciones en la nube [89].

Microsoft describe el servicio como:

PaaS es un entorno de desarrollo e implementación completo en la nube, con recursos que permiten entregar todo, desde aplicaciones sencillas basadas en la nube hasta aplicaciones empresariales sofisticadas habilitadas para la nube [88].

En este tipo de servicio, se paga solamente por el uso de los recursos que se necesitan del proveedor, a los que se acceden a través de una conexión a Internet. Como presenta la Figura 4.5, el servicio PaaS posee características de IaaS, lo que comprende infraestructura (servidores, almacenamiento y redes), pero escala un nivel más alto al añadir también middleware. El término “middleware” en este contexto se suele referir a las capas de software que se sitúan entre el sistema operativo y las aplicaciones, proporcionando una serie de servicios y funcionalidades que ayudan a las aplicaciones a comunicarse entre sí y con otros sistemas.

El middleware es muy útil porque simplifica el desarrollo, la implementación y la administración de aplicaciones en la plataforma, permitiendo a los desarrolladores concentrarse en la lógica de la aplicación. Ejemplos de middleware son servicios de: comunicación, bases de datos, seguridad, escalabilidad y gestión.

El ciclo de vida de aplicaciones web se refiere al proceso completo de desarrollo, implementación, mantenimiento y eventual retirada de una aplicación web, desde su concepción inicial hasta su finalización. Este ciclo suele seguir una serie de etapas clave: definición de requisitos, diseño, desarrollo, pruebas y control de calidad, implementación, operación y mantenimiento, evolución y mejora, y retiro [88]. Según Microsoft, al usar *PaaS* las empresas pueden deshacerse de la necesidad de comprar y gestionar por separado licencias de software, infraestructura de aplicaciones y software de intermediación (middleware) que son necesarios para desarrollar y ejecutar aplicaciones. Por lo tanto, el usuario usualmente administra solo las aplicaciones y los servicios que desarrolla [88].

El servicio que ofrece Particle a través de sus dispositivos incluye lo siguiente [89]:

- *PaaS* totalmente integrada: software, conectividad y hardware IoT son conectados fácilmente a través de la plataforma; seguridad, escalabilidad y conectividad son gestionados por parte de Particle.
- IoT personalizable: los dispositivos, módulos y funciones son reprogramables y reconfigurables para el desarrollo de aplicaciones personalizadas.
- Soporte: Particle permite acelerar el proceso de lanzamiento del producto al proporcionar hojas de datos y bibliografía detallada, una API y un SDK a disposición de los usuarios.

4.5.5. Planes de servicio de Particle Cloud

El servicio PaaS que provee Particle se clasifica en diferentes planes de crecimiento según sea el número de operaciones de datos realizadas, tanto para celular como para Wi-Fi. La selección del plan correspondiente abarca los siguientes puntos [90]:

- Cada publicación, suscripción, función o variable consume una operación de datos independientemente del tamaño.
- Los datos tienen un tamaño máximo de 622 a 1024 bytes de caracteres UTF-8.
- Los datos almacenados, como los datos de geolocalización del rastreador, consumen una operación de datos por punto de ubicación guardado.
- Ciertas retransmisiones, como se describe a continuación.

El plan gratuito incluye varios dispositivos, con un máximo de 100,000 operaciones de datos mensuales. Por lo tanto, es posible repartir la cantidad de operaciones de datos según se requiera sin superar el límite para mantenerse en el plan gratuito. Cuando el usuario desea subir al plan de crecimiento, Particle plantea el plan de acuerdo con el número de bloques solicitados. Cada bloque corresponde a un número máximo de Operaciones de Datos y dispositivos por mes en el plan de crecimiento. Cada bloque supone [90]:

- Hasta 720 000 operaciones de datos al mes.
- Hasta 100 dispositivos.
- Hasta 540 MB de datos móviles por mes, agrupados en todos los dispositivos, por cada bloque comprado.
- El precio varía para Wi-Fi, celular y rastreador.
- Añade tantos bloques como necesite el usuario.

Si el usuario supera alguno de los puntos estipulados y desea seguir haciendo uso del servicio, requiere solicitar otro bloque dentro del plan de crecimiento. Es decir, si el usuario cuenta con 150 dispositivos necesitará 2 bloques, aunque sus operaciones de datos aún no superen los 720,000.

4.5.6. Registro de datos en la Particle Cloud

El Photon tiene la capacidad de enviar los datos a la nube por medio de una librería especializada que provee Particle para enlazarlo. El enlace se logra por medio de la conexión a internet que establece el chip Wi-Fi Broadcom del dispositivo con el módem del proveedor de servicio de internet (ISP en inglés) [91].

De acuerdo a los datos compartidos en el sitio web oficial de la documentación ©Particle Industries, Inc. [92] para los servicios de encriptación, interconexión y manejo de datos en la nube es como sigue:

El servicio de la nube de Particle utiliza la autenticación mutua mediante pares de claves públicas y privadas de RSA (Rivest-Shamir-Adleman), un algoritmo útil para encriptación, para asegurarse de que el dispositivo sea el del usuario y no un suplantador, y la nube de Particle es la verdadera y no un impostor intermediario. Así, ambas partes pueden estar seguras de que la otra es quien dice ser. El proceso de negociación inicial crea una sesión cifrada usando DTLS (Datagram Transport Layer Security) sobre UDP (User Datagram Protocol) en dispositivos celulares y Gen 3, o AES (Advanced Encryption Standard) sobre TCP (Transmission Control Protocol) en Photon y P1 (módulo Wi-Fi). Esto asegura que sus datos no puedan ser monitoreados o manipulados en tránsito. La conexión de nube de Particle utiliza el CoAP (Constrained Application Protocol) sobre DTLS o AES. Todas las funciones, como publicación, suscripción, funciones, variables y actualizaciones de firmware OTA (Over-The-Air), se realizan a través de una única conexión CoAP.

Para lograr una transmisión de datos eficiente, se pueden reanudar las conexiones DTLS de la nube de Particle. Esto permite que un dispositivo se vuelva a conectar a la nube usando menos de 200 bytes de datos, frente a hasta 5K de datos para un protocolo de enlace completo.

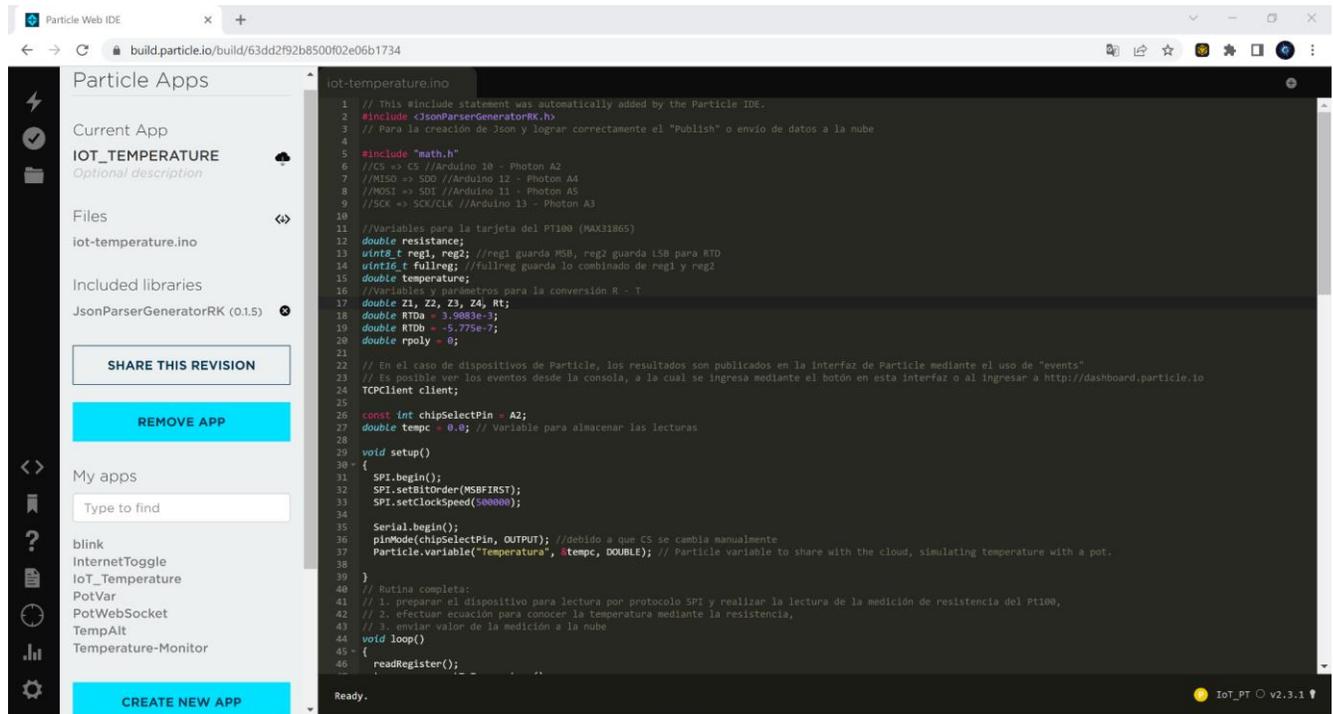
El uso de CoAP sobre DTLS con reanudación de sesión permite que la conexión a la nube se reanude de manera muy eficiente, a diferencia de otros protocolos como MQTT (Message Queuing Telemetry Transport) sobre TLS/SSL (Transport Layer Security/Secure Sockets Layer) que requieren un protocolo de enlace TLS completo de 5K en la reconexión. El uso

de funciones como Particle Publish permite que los datos se envíen a servidores externos usando tan solo 150 a 200 bytes de datos.

Establecer una conexión TLS/SSL a un servidor externo directamente desde un dispositivo podría usar 5000 bytes de datos para cada dato enviado cuando se incluye el protocolo de enlace TLS/SSL. Esto es posible porque la autenticación TLS/SSL se realiza fuera del dispositivo mediante webhooks o el flujo de eventos enviados por el servidor.

4.5.7. Conexión a la nube

En la Figura 4.6 se presenta la IDE de Particle con la visualización de una sección del código utilizado para la programación del Photon. Las funciones disponibles dentro del código son escribir, guardar, verificar, “Flash” o subir código al Photon y abrir otro código guardado en la cuenta. Como herramientas útiles de edición se iluminan los términos reservados, algunos argumentos dentro de funciones y se indica los errores de compilación encontrados al realizar la acción de verificación. De la selección de botones al costado o funciones adicionales en la IDE, se cuenta con acceso a las librerías compartidas por la comunidad, una sección de ayuda, documentación, visualización de los dispositivos enlazados a la cuenta, acceso a la consola de la nube de Particle y acceso a la configuración de la cuenta. En la configuración que se ofrece en este apartado, solamente se dispone de cambio de contraseña, cierre de sesión y liberar caché.



```
1 // This #include statement was automatically added by the Particle IDE.
2 #include <JsonParserGenerator.h>
3 // Para la creación de json y lograr correctamente el "Publish" o envío de datos a la nube
4
5 #include "math.h"
6 //CS => CS //Arduino 10 - Photon A2
7 //MISO => 500 //Arduino 12 - Photon A4
8 //MOSI => 501 //Arduino 11 - Photon A5
9 //SCK => SCK/CLK //Arduino 13 - Photon A3
10
11 //Variables para la tarjeta del PT100 (MAX31865)
12 double resistance;
13 uint16_t reg1, reg2; //reg1 guarda MSB, reg2 guarda LSB para RTD
14 uint16_t fullreg; //fullreg guarda lo combinado de reg1 y reg2
15 double temperature;
16 //Variables y parámetros para la conversión R - T
17 double Z1, Z2, Z3, Z4, RT;
18 double RTDh = 3.98816e-3;
19 double RTDl = -5.775e-7;
20 double rpoly = 0;
21
22 // En el caso de dispositivos de Particle, los resultados son publicados en la interfaz de Particle mediante el uso de "events"
23 // Es posible ver los eventos desde la consola, a la cual se ingresa mediante el botón en esta interfaz o al ingresar a http://dashboard.particle.io
24 TCPClient client;
25
26 const int chipSelectPin = A2;
27 double tempc = 0.0; // Variable para almacenar las lecturas
28
29 void setup()
30 {
31   SPI.begin();
32   SPI.setBitOrder(MSBFIRST);
33   SPI.setClockSpeed(500000);
34
35   Serial.begin();
36   pinMode(chipSelectPin, OUTPUT); //debido a que CS se cambia manualmente
37   Particle.variable("temperatura", &tempc, DOUBLE); // Particle variable to share with the cloud, simulating temperature with a pot.
38
39 }
40 // Rutina completa:
41 // 1. preparar el dispositivo para lectura por protocolo SPI y realizar la lectura de la medición de resistencia del PT100,
42 // 2. efectuar ecuación para conocer la temperatura mediante la resistencia,
43 // 3. enviar valor de la medición a la nube
44 void loop()
45 {
46   readRegister();
47 }
```

Figura 4.6. Captura de la IDE de Particle en su sección de archivos de código.

El Photon cuenta con capacidades y funciones similares a un microcontrolador. Con esto se indica que la programación fue similar a la programación desarrollada en C. Lo que se distingue en este aspecto son las funciones IoT que proporciona la compañía de manera predeterminada. Con predeterminado se interpreta que las funciones ya se encuentran integradas sin la necesidad de incluir librería alguna. Las funciones empleadas para el envío de datos a la nube se muestran en el Anexo 3 Código A.1, en donde se hace énfasis en las funciones *JsonWriterStatic* y *Particle.publish()*.

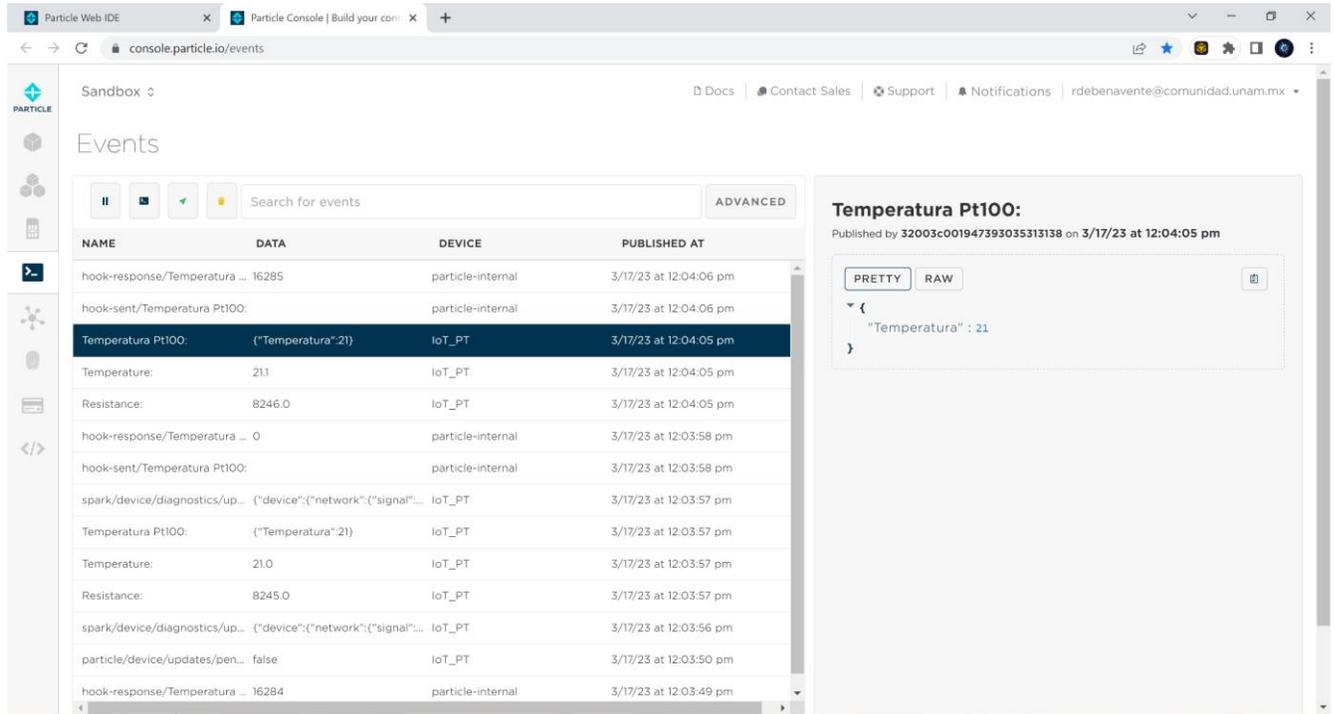
JsonWriter se usa para construir cadenas JSON válidas. Si bien puede compilar JSON con *sprintf*, el generador de JSON puede usar cadenas de escape entre comillas dobles y comillas dobles dentro de las cadenas. También puede generar caracteres Unicode correctos. Acorde a la documentación referente al *Particle.publish()*, el uso más común que se le da a *JsonWriter* es construir un búfer estático para contener los datos JSON para *Particle.publish()*.

Dado que estos datos están limitados a 256 bytes, este es un enfoque razonable utilizando `JsonWriterStatic` de la siguiente manera: “`JsonWriterStatic<256> jw;`”. También se puede asignar dinámicamente un búfer usando el simple `JsonWriter`. `JsonWriter` maneja matrices y objetos anidados, pero lo hace sin crear copias temporales de los objetos. Debido a esto, es necesario usar `startObject()`, `startArray()` y `finishObjectOrArray()` para que los objetos se equilibren correctamente. Para facilitar esto, se puede crear una instancia de `JsonWriterAutoObject` en la pila [93]. Cuando el objeto sale del alcance, se cerrará automáticamente [93]. El código que crea la cadena JSON para la publicación de eventos se puede observar en el Anexo 3.

Con base en la sintaxis o el modo de empleo de las funciones, se creó un bloque de función que agrupara las instrucciones que se requieren ejecutar en cada iteración para conformar un objeto en formato JSON que almacene los datos de la variable de interés y los envíe a la nube de Particle. El bloque de función fue denominado “`createEventPayload`”, como se puede notar en el Código A.2. El concepto de “evento” se refiere a cada dato subido a la nube, lo cual genera una operación en los servidores y se puede considerar como un estado del sistema. Particle define que un evento posee las siguientes propiedades [94]:

- nombre de 1-64 caracteres ASCII
- datos opcionales: Los datos tienen un tamaño máximo de 255 a 1024 bytes de caracteres UTF-8; se deben de consultar los límites de campo de API, ya que el límite varía según la versión del sistema operativo del dispositivo y, a veces, el dispositivo.

La interfaz de usuario de la consola de la nube de Particle se observa en la Figura 4.7. Es de interés la sección de “Events” o eventos, pues como ya se ha mencionado, es en este espacio donde se visualizan los datos subidos a la nube. Existen otras secciones que ofrecen más funciones de enlace y conexión a más servicios de Particle, pero acorde a los alcances de este proyecto no fueron empleadas.



The screenshot shows the Particle Console interface. On the left, there is a sidebar with navigation icons. The main area displays a table of events with columns for NAME, DATA, DEVICE, and PUBLISHED AT. The event 'Temperatura Pt100' is highlighted. To the right, a detailed view of this event is shown, including a 'PRETTY' button and a JSON representation of the event data: `{ "Temperatura" : 21 }`.

NAME	DATA	DEVICE	PUBLISHED AT
hook-response/Temperatura ...	16285	particle-internal	3/17/23 at 12:04:06 pm
hook-sent/Temperatura Pt100:		particle-internal	3/17/23 at 12:04:06 pm
Temperatura Pt100:	{ "Temperatura": 21 }	loT_PT	3/17/23 at 12:04:05 pm
Temperature:	21.1	loT_PT	3/17/23 at 12:04:05 pm
Resistance:	8246.0	loT_PT	3/17/23 at 12:04:05 pm
hook-response/Temperatura ...	0	particle-internal	3/17/23 at 12:03:58 pm
hook-sent/Temperatura Pt100:		particle-internal	3/17/23 at 12:03:58 pm
spark/device/diagnostics/up...	{ "device": { "network": { "signal": ...	loT_PT	3/17/23 at 12:03:57 pm
Temperatura Pt100:	{ "Temperatura": 21 }	loT_PT	3/17/23 at 12:03:57 pm
Temperature:	21.0	loT_PT	3/17/23 at 12:03:57 pm
Resistance:	8245.0	loT_PT	3/17/23 at 12:03:57 pm
spark/device/diagnostics/up...	{ "device": { "network": { "signal": ...	loT_PT	3/17/23 at 12:03:56 pm
particle/device/updates/pen...	false	loT_PT	3/17/23 at 12:03:50 pm
hook-response/Temperatura ...	16284	particle-internal	3/17/23 at 12:03:49 pm

Figura 4.7. Captura de consola de la nube de Particle con eventos subidos por el Photon.

Antes de poder escribir y flashear el código en el Photon, o incluso poder usarlo, se debe de reclamar a una cuenta de Particle. Dicho proceso se realizó siguiendo los pasos indicados para el uso de la aplicación móvil, los pasos pueden encontrarse en la siguiente página: <https://docs.particle.io/community/phonon-maker-kit-workshop/ch1/>. También es mediante la aplicación que se configuró el acceso a internet por medio de la señal Wi-Fi disponible. Tras efectuar los cambios y montar el código en el Photon, se suben los datos a la nube al encontrarse el dispositivo en operación. Los datos subidos a la nube se visualizan en la consola de Particle como se presenta en la Figura 4.7. Se despliegan cuatro columnas donde se indica el nombre, el dato, el dispositivo y el tiempo en que se publicó el evento. Se pueden seleccionar los eventos individualmente para ver todos los datos referentes en el espacio encontrado a la derecha.

Además, se puede elegir el formato en que se visualizan los datos “Pretty” o “Raw”, que corresponden a un formato que puede ser leído con mayor facilidad por una persona y uno que muestra los datos exactamente igual a como son enviados por el Photon.

4.5.8. Adquisición y procesamiento de los datos

El Photon es un dispositivo Wi-Fi IoT para crear proyectos conectados al Internet de las cosas. La placa se alimenta mediante un chip Wi-Fi de Cypress, en combinación con un STM32 ARM Cortex M3, el cual es un MCU robusto. Por lo que la tarjeta de desarrollo Photon puede ser programada a través de su IDE personalizada de manera similar a un microcontrolador en C. Gracias a las entradas con las que cuenta, el Photon tiene la capacidad de realizar comunicación asíncrona UART (Universal Asynchronous Receiver/Transmitter) o síncrona como es SPI (Serial Peripheral Interface). Cuenta con pines de entrada/salida de propósito general, módulos PWM, ADC y DAC (ver Figura 4.8) [91]. Por medio del dispositivo IoT se realiza la adquisición, procesamiento y envío de la lectura en grados Celsius (°C) de uno de los RTD Pt100 instalado en el sistema de pasteurización.

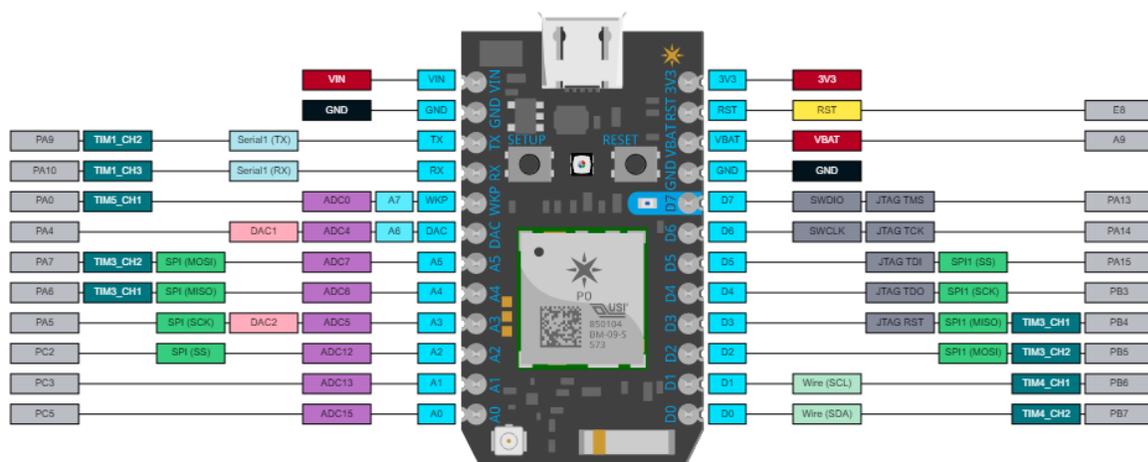


Figura 4.8. Diagrama de asignación de pines del Photon. Tomado de [91].

4.5.9. Procesamiento de datos

El proceso de monitoreo requiere que el Photon reciba la señal del Pt100, procesarla, enviar los datos a la nube y esperar a la actualización de la interfaz gráfica. El código elaborado que consigue el proceso de monitoreo se explica en el diagrama de bloques mostrado en la Figura 4.9.

Al inicio del programa se incluyen las librerías que se usarán para el código, en el proyecto se utilizó “JsonParserGeneratorRK.h” y “math.h”. La primera librería es de utilidad para crear el objeto en formato Json y formar el evento en la nube, la segunda se requiere para llamar funciones matemáticas necesarias para realizar con mayor facilidad algunos cálculos. Adicionalmente se definen los variables globales que son de utilidad para los registros que se acceden en el Max31865 y los valores que conforman al polinomio en el cálculo de la temperatura. Acto seguido se realiza la configuración, que es marcada como “setup” en la programación del Photon.

En este apartado se llaman las funciones que establecen la comunicación serial (SPI), se definen los pines que funcionarán como salida o entrada y las variables cuyos datos serán enviados a la nube, y opcionalmente se puede asignar algún valor a los pines o variables globales antes de comenzar el ciclo iterativo o “loop”. Ya en la sección del “loop” se encuentra el llamado de cuatro funciones principales y un delay o retraso de 8 segundos para comenzar el nuevo ciclo.

Sin entrar en detalles técnicos para conocer solamente la función del código, se muestra el desglose de las acciones que se ejecutan esencialmente en cada función principal. Las cuatro funciones se encuentran presentadas por recuadros de color azul. El proceso de monitoreo es iterativo y tiene un periodo de aproximadamente 8 segundos. El proceso consiste en obtener los datos del Pt100 mediante el convertidor Max31865, el cual los envía mediante comunicación SPI. La comunicación es abierta por el Photon para a continuación recibir y guardar los datos en dos registros debido al tamaño de la cadena. Se guardan después estos datos en una variable capaz de almacenar la cadena completa. El Photon cierra de nuevo la comunicación con el convertidor.

Los datos son procesados por el Photon siguiendo el cálculo de un polímero que define la conversión de datos de resistencia a temperatura en grados celsius. Según el valor calculado por medio de los primeros términos del polímero, se determina si se procede a aumentar la precisión o si ya es precisión suficiente para enviar los datos a la nube.

Terminado el proceso de cálculo, el resultado del cálculo es comparado con un valor máximo, según el proceso que se esté llevando a cabo, para abrir o mantener cerrado el circuito de potencia del calentador como medida preventiva. Si el proceso es VAT, el máximo es de 65 °C, si el proceso es HTST, el máximo es de 75 °C. Finalmente, los datos de temperatura son guardados en formato JSON en un objeto para ser enviados correctamente a la nube. En el apartado de *Conexión a la nube* se describe que es necesario contener los datos en formato JSON para hacer uso correctamente de la función `Particle.Publish()`, la cual enlaza los datos a la nube.

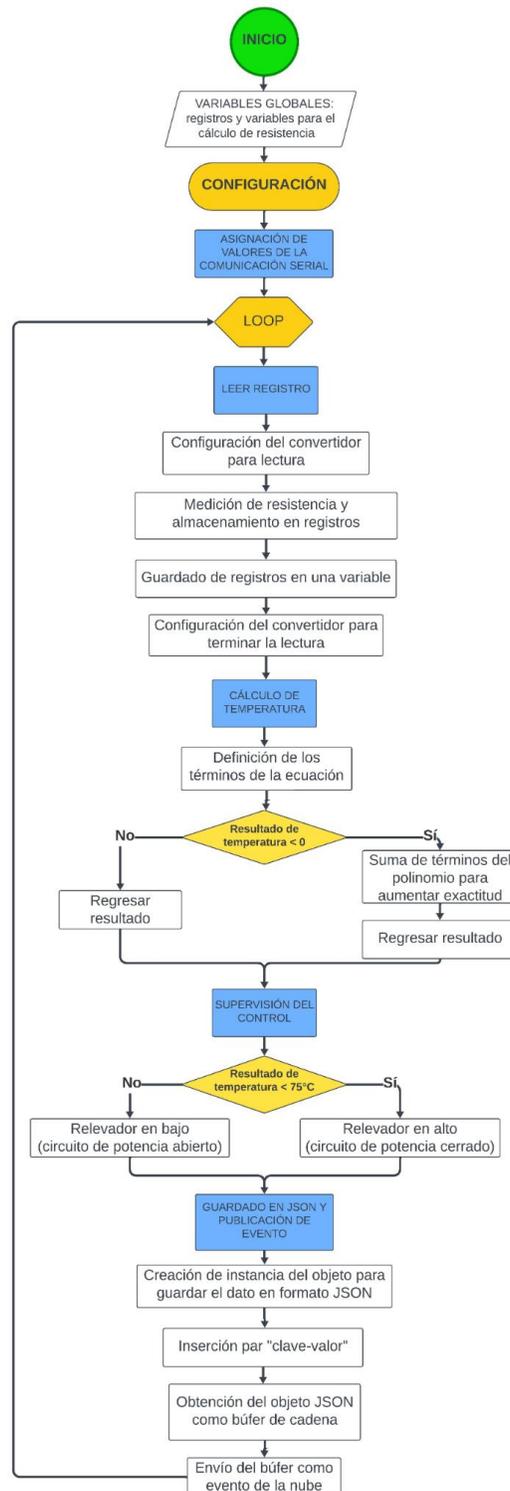


Figura 4.9. Programación del Photon en forma de diagrama de flujo.

4.6. INVESTIGACIÓN DEL SOFTWARE IOT Y ELABORACIÓN DE LA INTERFAZ DE MONITOREO

Con el uso de Python es posible desarrollar una gran variedad de aplicaciones debido al gran soporte que se le ha dado por medio de la comunidad en los últimos años. Por ello la selección de las librerías especializadas se basó en su capacidad de diseñar interfaces gráficas de usuario reactivas, con los elementos suficientes para el monitoreo adecuado del proceso de pasteurización y con compatibilidad de comunicación con el dispositivo IoT. La librería escogida cumple con los criterios de selección, además de un repositorio de plantillas en Github que sirven de manera didáctica y eficiente para agilizar los tiempos de desarrollo.

4.6.1. Plotly

Plotly crea y actualiza Dash y proporciona un medio y las herramientas para desarrollar y lanzar apps en contextos corporativos. Plotly es una empresa informática técnica con sede en Montreal, Quebec, fundada en el año 2012, que desarrolla herramientas de análisis y visualización de datos en línea [95]. Plotly proporciona herramientas de gráficos, análisis y estadísticas en línea para individuos y colaboración, así como bibliotecas de gráficos científicos para Python, R, MATLAB, Perl, Julia, Arduino y REST [96].

4.6.2. Dash

Dash es una biblioteca de código abierto que opera conforme la licencia del MIT, la cual permite un amplio uso y modificación [97]. Además, actúa como la estructura fundamental de código para desarrollar de forma experimental apps de procesamiento de datos en F#, Julia, R y Python de manera eficiente. Dash utiliza Plotly.js y React.js para crear apps personalizadas de visualización de datos. Es útil para quienes manejen datos y diseñen interfaces. Dash se encarga de la tecnología necesaria para crear aplicaciones web con gráficos interactivos [98].

Es posible ver estas aplicaciones en el navegador web, con ello compartirlas a través de URL y usarlas en diferentes dispositivos, incluyendo móviles.

4.6.3. Librerías de componentes de código abierto

Dash tiene la posibilidad de incluir componentes para interfaces de usuario interactivas, de los cuales existen de código cerrado y de código abierto. Los componentes de código cerrado son de pago y son aportados por la compañía, por otro lado, los componentes de código abierto son actualizados y ampliados por la comunidad. El código abierto permite una gran variedad de opciones, puesto que pueden verse modificadas las versiones actuales o pueden proponerse nuevas en actualizaciones posteriores. Existe una variedad de librerías de código abierto dentro de Dash, de las cuales las más destacadas son:

- *Dash Core Components*: brinda acceso a muchos componentes interactivos de nivel superior, incluidos menús desplegables, listas de verificación y controles deslizantes. Las opciones disponibles en esta librería incluyen: menús desplegables, controles deslizantes, entradas numéricas, entradas de texto, casillas de verificación, botones, selector de fechas, “Markdown”, subir/cargar componentes, bajar/descargar componentes, tabuladores, gráficas, diálogos de confirmación, almacenamiento de datos en el navegador, componentes de ubicación [99].
- *Dash DAQ (Data Acquisition)*: marco de aplicación web que proporciona abstracción pura de Python en torno a HTML, CSS y JavaScript. Dash DAQ comprende un sólido conjunto de controles que simplifican la integración de la adquisición de datos y los controles en aplicaciones Dash.

Componentes que se encuentran disponibles en la versión 0.5.0 son: interruptores booleanos, selectores de color, medidores, barras graduadas, indicadores, palancas de mando, perillas, monitores led, entradas numéricas, botones de encendido, botones de apagado, entradas de precisión, menús deslizantes, tanques de llenado, termómetros, interruptor de palanca, tema oscuro [100].

- *Dash HTML Components*: marco de aplicación web que proporciona abstracción pura de Python en torno a HTML, CSS y JavaScript. En lugar de escribir HTML o usar un motor de plantillas HTML, compone su diseño usando Python con el módulo de Componentes HTML de Dash (`dash.html`). La librería también tiene acceso a propiedades como “`style`”, “`class`” y “`id`”. Todos estos atributos están disponibles en las clases de Python [101].

Los elementos HTML y las clases de Dash son en su mayoría iguales, pero hay algunas diferencias clave:

- La propiedad de “`style`” es un diccionario.
- Las propiedades en el diccionario de “`style`” están en “`camelCased`”.
- La clave de “`class`” se renombra como `className`.
- Las propiedades de “`style`” en unidades de píxeles se pueden proporcionar como números sin la unidad “`px`”.

Se presenta un ejemplo de una estructura HTML simple con el uso de la librería de Dash en Python (ver Código A.3) para exponer la forma en que cambia la programación tradicional de páginas web (ver Código A.4).

El Código A.4 se convierte, mediante la programación de la librería, en código HTML que puede ser leído por el navegador web. El uso de la estructura de programación propuesta por Dash puede resultar más eficiente y práctica para códigos más extensos y complejos en comparación con la estructura ramificada original.

4.6.4. Funciones reactivas

De acuerdo con Lettier [102] la programación reactiva funcional, FRP en inglés, “es un marco general para programar sistemas híbridos de manera declarativa y de alto nivel, las ideas clave en FRP son sus nociones de comportamientos y eventos”. Se refiere a la creación, uso y combinación de comportamientos y eventos para crear sistemas o aplicaciones reactivas a base de funciones. El término reactivo en este contexto denota una “reacción” automática por parte de la salida (output) debido a un cambio o cambios en la entrada (input) [103].

Los comportamientos, tal como una variable, son valores que varían en el tiempo. Un comportamiento siempre tiene un valor sin importar la hora que sea y su valor puede cambiar con el tiempo [102]. Los comportamientos son variables continuas con respecto al tiempo. Los eventos son valores con marca de tiempo o pares (tiempo, valor). En otras palabras, los eventos son variables discretas con respecto al tiempo [102]. Al igual que los comportamientos, la biblioteca de FRP dentro del lenguaje de programación debe incluir alguna función que permita realizar alguna acción eficaz, con el valor del evento, siempre que ocurra el evento.

En Dash, las funciones que producen la programación reactiva son llamadas “callbacks”, lo que se traduce como “devolución de llamada”. Las funciones tipo *callback* se ejecutan sistemáticamente cuando se modifica un valor o propiedad de un componente de entrada (input), y esto se utiliza en programación funcional para actualizar un valor en otro componente (output) [103].

4.6.5. Aplicaciones

Dash se encuentra disponible a los usuarios desde 2016 con su versión 1.0.0. Hasta la fecha se han realizado una diversidad de aplicaciones con el uso de esta librería, un repositorio que ha sido de utilidad para demostrar los alcances de esta herramienta es el siguiente: <https://dash.gallery/Portal/> .

En el repositorio es posible encontrar aplicaciones de distintas áreas, tanto de la ciencia, como de la industria, como también del deporte. El hecho de ser una librería de Python otorga a los usuarios todas las capacidades que este lenguaje de programación contiene. Es decir, se tiene acceso a las herramientas de Dash y a todas las librerías que se encuentran en Python. El cúmulo de posibilidades creativas y productivas en el entorno que se genera con aplicaciones de Dash es inmenso. No obstante, existe la posibilidad de tener acceso al código de aproximadamente 90 aplicaciones Dash, ya sea para expandir el conocimiento de capacidades que tiene la librería como también tomar como plantilla una de estas Demos. El repositorio de las Demos de Dash de código abierto se encuentra en el enlace siguiente: <https://github.com/plotly/dash-sample-apps> .

Las pruebas consisten en modificaciones de los parámetros de referencia, es decir, temperatura de pasteurización, volumen y agitación. Las pruebas funcionan como método de caracterización del sistema con los instrumentos de medición, monitoreo y control propuestos. Se conoce que los instrumentos del sistema no lo vuelven capaz de llevar a cabo un proceso de pasteurización HTST exitosamente, sin embargo, se realizaron las pruebas a la temperatura de pasteurización correspondiente para identificar los alcances del sistema. Con las pruebas se pretende también reconocer las áreas de oportunidad del sistema.

4.6.6. Elaboración de la interfaz

Tomando en cuenta la disponibilidad de tiempo y las opciones de desarrollo, se elaboró la interfaz partiendo de una plantilla, es decir, tomando el código fuente de una de las demos. El código de la demo elegida fue tomado directamente del repositorio de Github de las demos de Dash.

Siguiendo esta línea de base, para la aplicación se escogió una demo denominada “Panel de Control de Procesos Estadísticos de Fabricación”.

En el proceso de selección se decidió por esta aplicación puesto que cuenta con la representación más similar a la que se apuntaba desde un inicio al buscar una interfaz adecuada para presentar los datos del sistema de pasteurización. La demo se refiere a un proceso que puede considerarse industrial, al mismo tiempo que despliega la información de manera más compacta y sintetizada en forma de datos estadísticos para tener una comprensión del proceso sin recurrir al uso de muchos elementos de texto o gráficos.

La aplicación se compone de dos pestañas o páginas de visualización, una donde se configura los límites de especificación y control de cada variable, y la otra donde se encuentran los gráficos de control. En la Figura 4.10 se observa una captura de la pestaña del tablero de gráficos de control de la demo, de donde se destaca el uso de cuatro paneles para estructurar la visualización de los gráficos estadísticos. En el lado izquierdo se encuentra el panel de estadísticas rápidas, en la sección media el panel superior de estadísticas, en la derecha el gráfico de pastel y en el inferior el panel del gráfico de control. El panel de estadísticas rápidas muestra la identificación que se le haya dado al sistema y una manecilla o disco mostrando el valor de un dato con respecto a un rango, se añade en esta sección un botón de “start” para iniciar el proceso de monitoreo.

Todas las variables de proceso de interés son denominadas “Parameter” en el panel superior de estadísticas, se presenta aquí el conteo o número de datos registrados de esa variable, un minigráfico de línea 2D, el porcentaje fuera de control (Out Of Control) de los datos, y con respecto a ello, un gráfico de barra representando dicho porcentaje, finalmente se cuenta con un indicador “Pass/Fail” para señalar si el proceso ya se ha salido de control fuera del rango permitido. Dicho rango de “pase” es de 8% por encima o por debajo del parámetro superior o inferior de control respectivamente. En el panel a un lado se visualiza el gráfico de pastel, que muestra de manera sintetizada con colores el porcentaje en el que las variables se encuentran fuera de control en el proceso.

Por último, el panel del gráfico de control despliega el valor de las variables con respecto a número de lote o del conteo mediante una línea con marcadores, encima se sitúan las líneas punteadas de los parámetros de especificación y control de cada variable, la media de los datos obtenidos y un histograma por el lado derecho. La selección de los datos de la variable que se presentan en el gráfico de control se realiza con el click en el nombre de la variable correspondiente en el panel de gráfico superior.

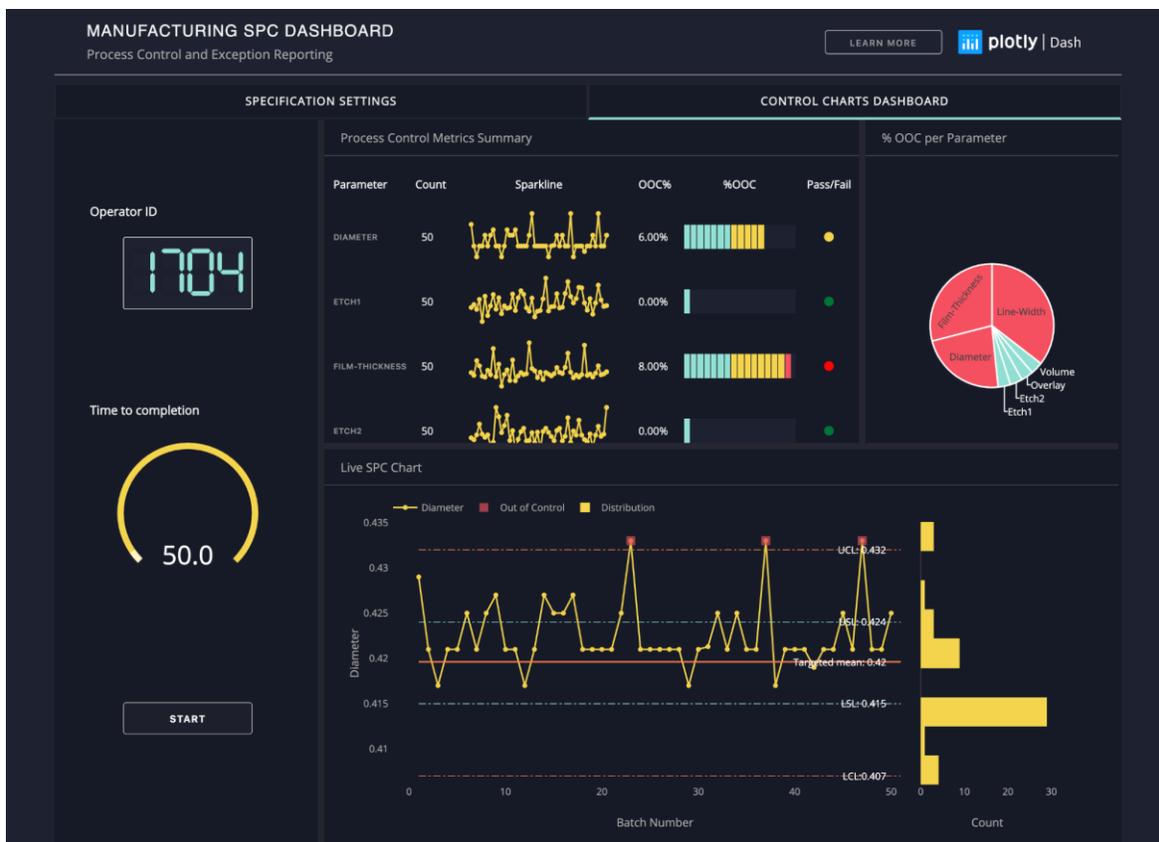


Figura 4.10. Captura de la pestaña de tablero de gráficos de control de la aplicación Dash utilizada como plantilla.

En la Figura 4.11 se visualiza la pestaña correspondiente a la configuración de especificaciones, las cuales se refieren a los límites de especificación y control de la variable de interés. Puede optarse por definir límites específicos nuevos para cada variable, para algunas o por ninguna en lo absoluto. Como puede observarse en una de las columnas de la tabla de la pestaña, ya cuentan desde el inicio con valores históricos para cada variable. Los valores históricos en la aplicación se obtienen mediante un cálculo que se basa en porcentajes con respecto a los datos que son leídos al iniciar la ejecución del código. Es decir, de los primeros datos con los que se cuenta de la variable se determina estadísticamente cuáles pueden ser los límites adecuados.

Este método es la forma predeterminada de cómo se establecen los límites, si no se realizan las modificaciones, el gráfico de control es definido y elaborado con los límites obtenidos mediante los cálculos. El método predeterminado es útil cuando el conjunto de datos es representativo del patrón que se desea o que se tendrá para el resto de las mediciones, es decir, es representativo o característico del comportamiento usual de la variable. Cuando se cuenta con pocos datos, no son característicos del proceso del sistema, presentan valores muy alejados del rango deseado o se desea definir límites específicos, el método predeterminado puede ser reemplazado por valores que el usuario introduce en los campos de la última columna.

En estos campos, es posible escribir los datos numéricos o cambiar los actuales presionando las flechas en el borde para aumentar o disminuir el valor. Cuando se hayan introducido todos los límites deseados para la variable en específico, se presiona el botón de “Update” para registrar o guardar los cambios. Si se desea corroborar que los cambios fueron realizados, se presiona el botón de “View current setup”, lo que despliega una tabla por debajo que muestra los límites que se encuentran guardados para definir el gráfico de control.

MANUFACTURING SPC DASHBOARD
Process Control and Exception Reporting

LEARN MORE  plotly | Dash

SPECIFICATION SETTINGS

Use historical control limits to establish a benchmark, or set new values.

Select Metrics

Diameter x

CONTROL CHARTS DASHBOARD

Specs	Historical Value	Set new value
Upper Specification limit	0.424	0.424
Lower Specification limit	0.415	0.415
Upper Control limit	0.432	0.432
Lower Control limit	0.407	0.407

UPDATE VIEW CURRENT SETUP

Specs	Current Setup
Upper Specification Limit	0.424
Lower Specification Limit	0.415
Upper Control Limit	0.432
Lower Control Limit	0.407

Figura 4.11. Captura de la pestaña de configuración de especificaciones de la aplicación Dash utilizada como plantilla.

4.6.7. Personalización y ajuste de la interfaz gráfica

En relación con la plantilla utilizada, presentada en el apartado de *Visualización en interfaz gráfica*, se diseñó la nueva interfaz o aplicación considerando las necesidades de monitoreo para el sistema de pasteurización. La Figura 4.12 presenta la captura de la pestaña del tablero de control de la aplicación al iniciar. En comparación, se modificó el número de variables presentadas en el panel superior a uno para mostrar solamente la temperatura. El panel de estadísticas rápidas fue modificado casi por completo, a excepción del componente de display led, donde se muestra el supuesto número del sistema, y el botón de inicio y paro del proceso de monitoreo en la interfaz.

Debajo del número del ID del sistema se incluyó un texto de color para representar el estado de control del proceso, “Normal” y color verde en condiciones adecuadas, “Fault” y “Check” de manera intermitente alternando de amarillo a rojo en condiciones fuera de control. Al indicador se le aumentó el tamaño, se le añadieron detalles para aumentar la legibilidad como una manecilla, color blanco en los valores para contrastar, un gradiente de color para representar el cambio de temperatura, se añadieron marcas de los valores y que además se muestre el valor numérico que se está representando en el indicador. Y como última adición en este panel, se incluyó un contador del tiempo restante del proceso una vez que se alcance la temperatura de pasteurización. El otro panel que recibió los mayores cambios fue el del gráfico de pastel, que en lugar de mostrar el estado fuera de control de las variables ahora indica el tiempo transcurrido por cada etapa: calentamiento, pasteurización, enfriamiento.

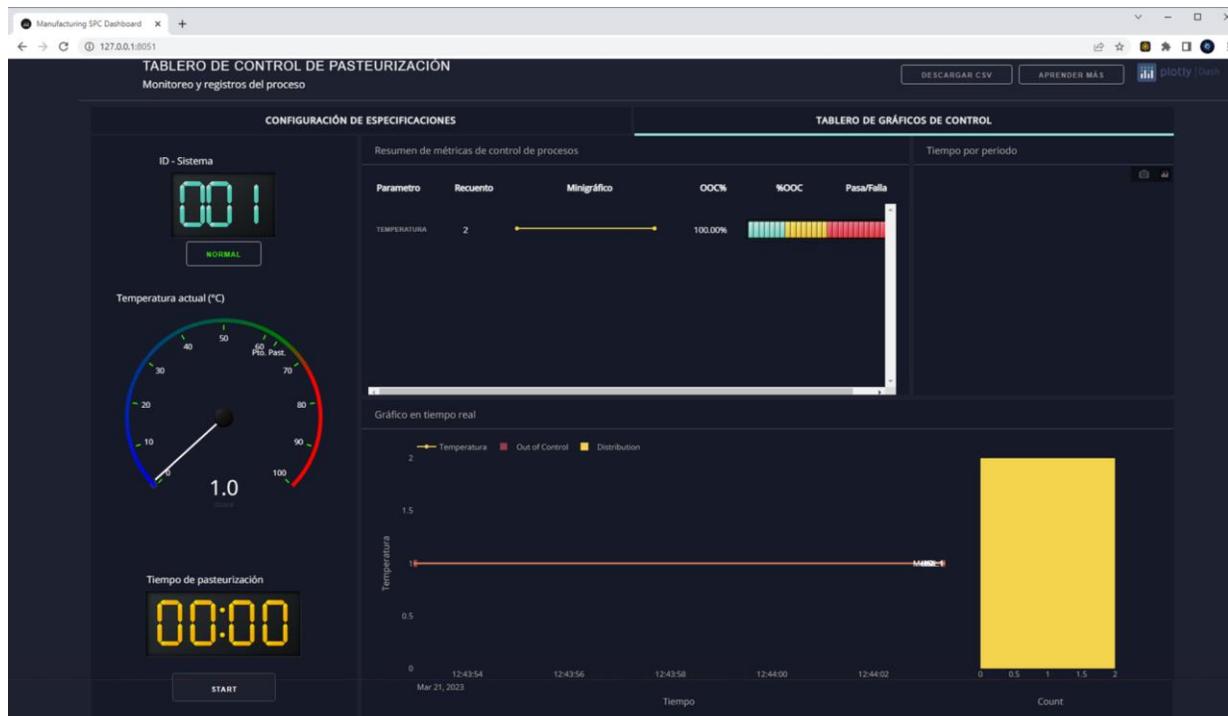


Figura 4.12. Captura de la pestaña del tablero de gráficos de control de la aplicación para el sistema de pasteurización.

Cambios menores pero útiles en el proceso de monitoreo se realizaron en la pestaña de configuración de especificaciones. Como se puede notar en la Figura 4.13, se retiró una de las columnas, puesto que, para fines prácticos del sistema de pasteurización, no se requieren los valores históricos de los primeros datos obtenidos para determinar los límites de especificación y de control. Se añadieron dos espacios más para definir la temperatura de pasteurización (en grados Celsius) y el tiempo de pasteurización (en segundos), para dar mayor flexibilidad a quien desee llevar a cabo otro tipo de pasteurización.

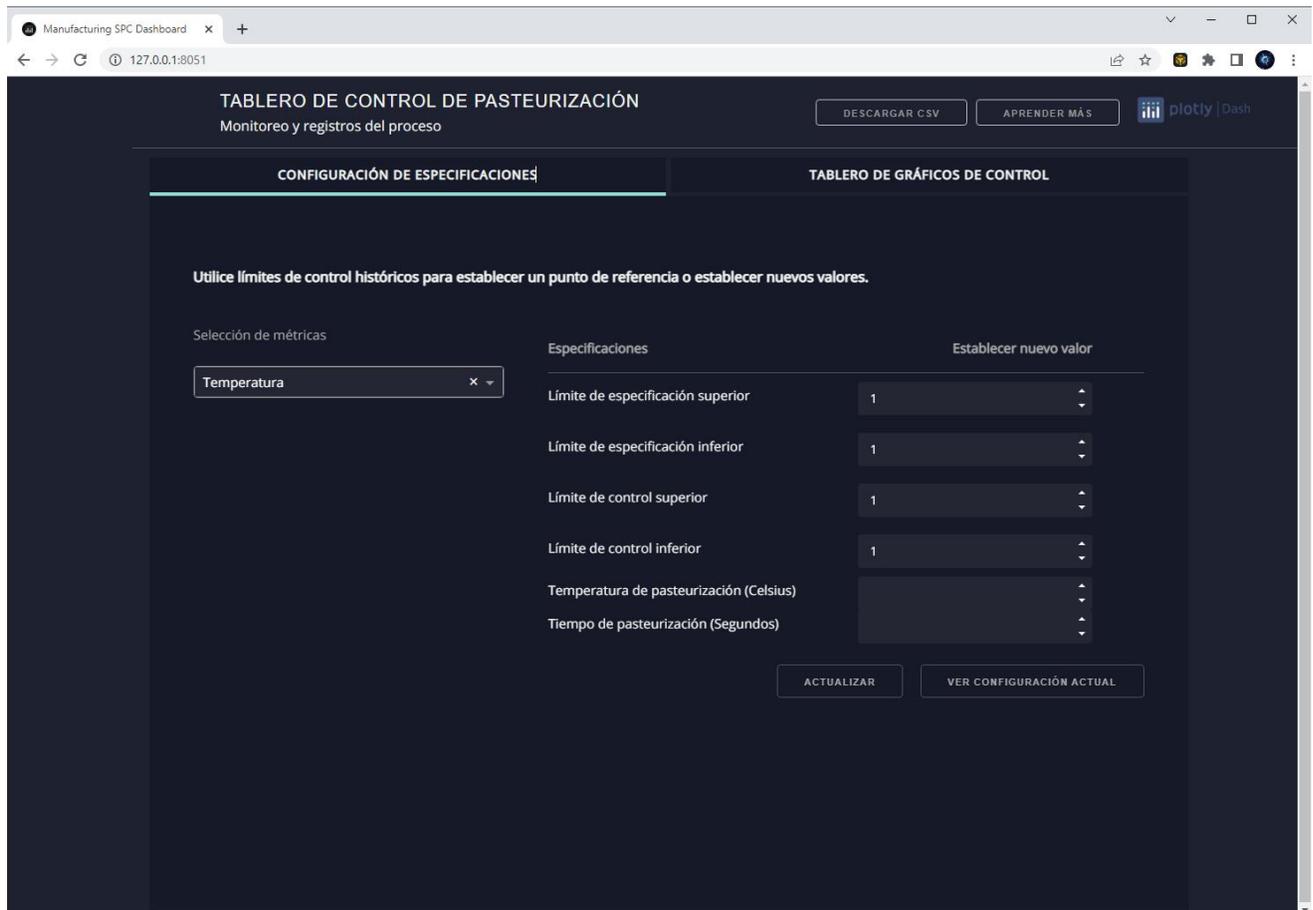


Figura 4.13. Captura de la pestaña del tablero de configuración de especificaciones de la aplicación para el sistema de pasteurización.

La visualización de los resultados presentados por la aplicación en un proceso de pasteurización VAT o LTLT se puede observar en la Figura 4.14. Se destaca que la curva presenta un comportamiento usual, que se puede analizar en conjunto con el gráfico de pastel, que indica precisamente que el tiempo de calentamiento fue más corto que el de pasteurización. Haciendo uso de la interfaz, es posible pasar la flecha del ratón por encima del gráfico de pastel para que se muestre el tiempo en segundos que es representado en el gráfico, así también el porcentaje en número que representa una etapa con respecto al tiempo total.

El histograma en el lado derecho indica que el mayor número de datos se encuentra en el rango de pasteurización, demostrando que no se desvió significativamente o en lo absoluto la temperatura del rango de interés.

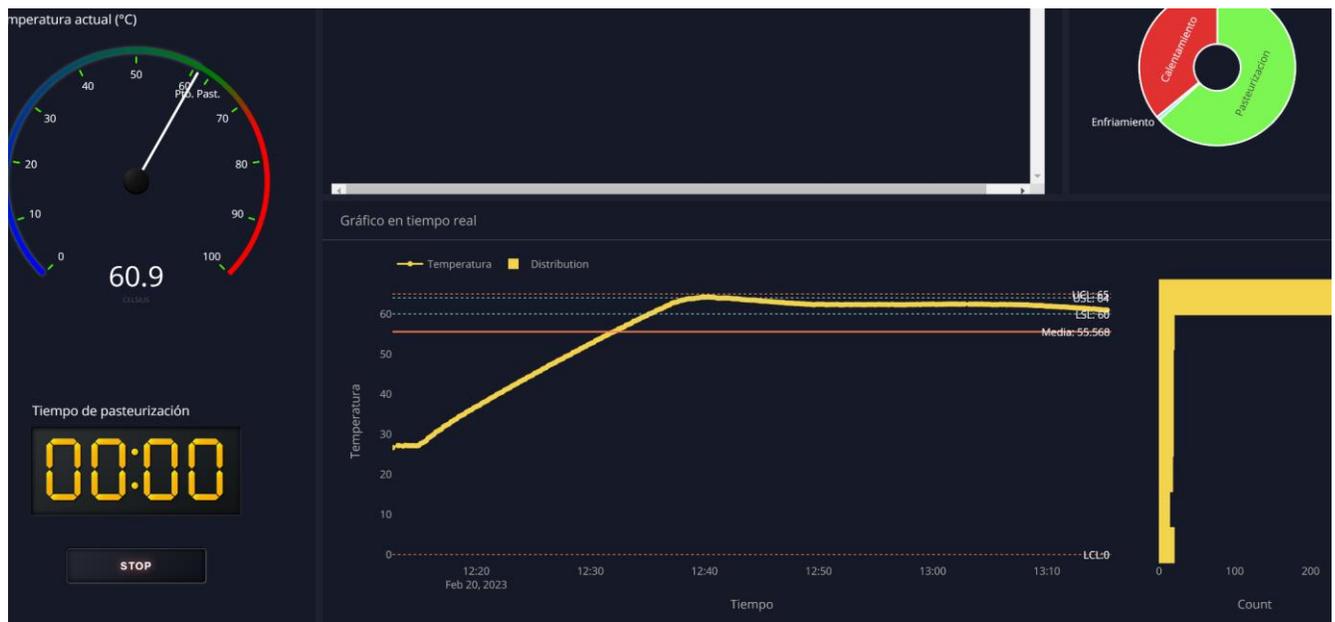


Figura 4.14. Visualización de la interfaz de monitoreo en un proceso de pasteurización VAT.

El desarrollo principal en la aplicación fue el cambio de fuente para los datos, la demo originalmente hace uso de un archivo que guarda los datos que se presentarán, sin embargo, no cuenta con la programación para recibir datos en tiempo real y presentarlos de la misma forma. En la interfaz no es posible reconocer cómo se realiza el procesamiento de la información o los datos, aspecto que es posible de comprender a través del código o de un esquema que presente el funcionamiento del código. Puesto que el código se compone de más de 1500 líneas, resulta más conveniente mostrar un esquema que resume las funciones dentro de la programación. La Figura 4.15 presenta lo que compone a la primera sección del código, es decir, la definición de los contenedores que conforman los paneles y de las funciones que procesan información, definen elementos dentro de los contenedores y elementos dentro de los componentes gráficos de la aplicación.

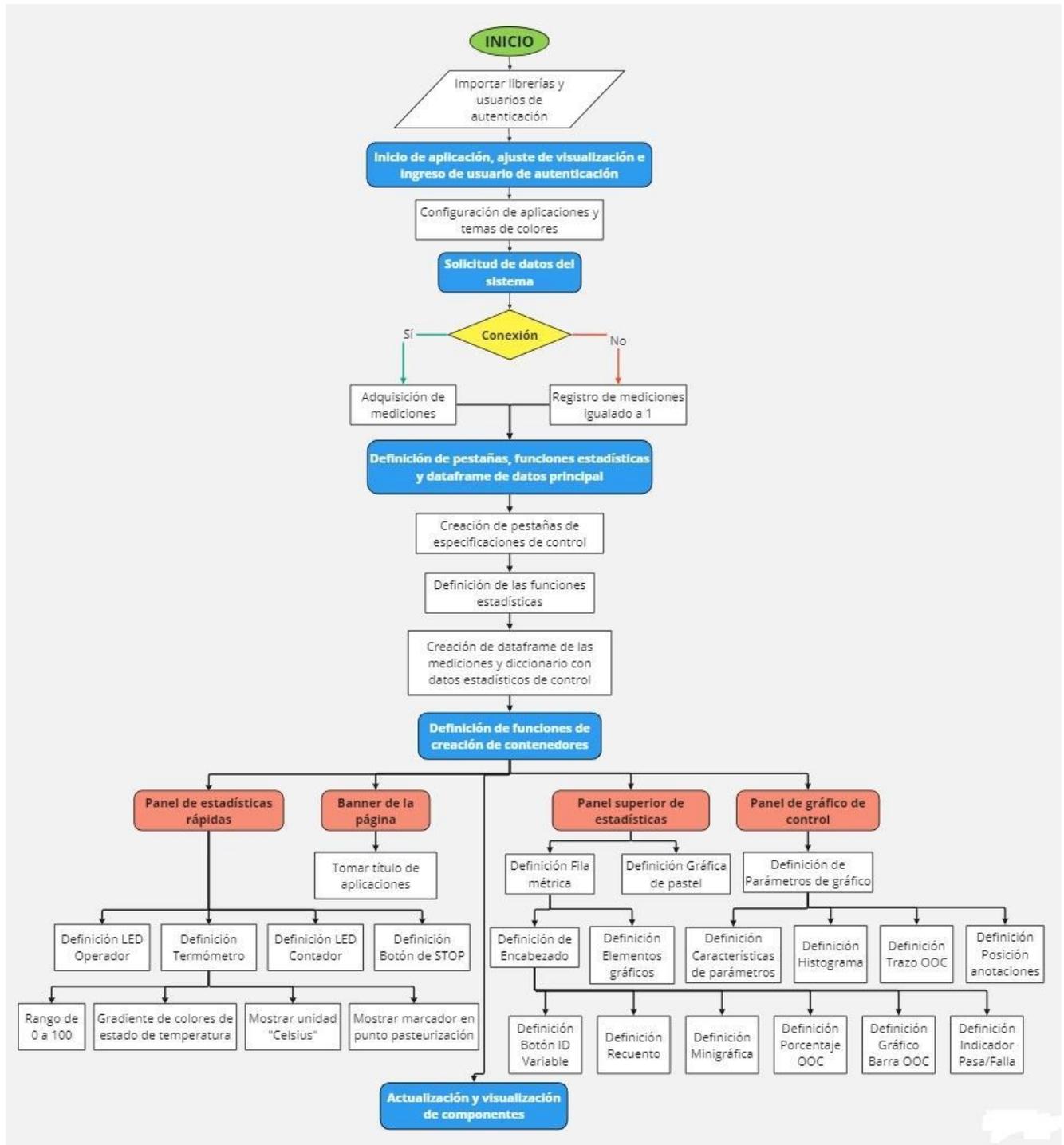


Figura 4.15. Diagrama de flujo del código de la interfaz dentro de la definición de contenedores y funciones de procesamiento de información.

En la primera porción del código se importan las librerías de las funciones especializadas que serán llamadas más adelante. Posteriormente, se ejecuta el proceso de autenticación, que significa introducir un usuario y su contraseña. La aplicación toma los valores de la paleta de colores asignados para definir los colores de los elementos gráficos. A continuación, para definir y conformar los “dataframe” o estructuras de datos, que sirven para almacenar los datos adquiridos de la nube y los parámetros estadísticos que se extraen a partir de ellos, se toman dos datos de la nube. Este proceso no solo sirve para definir la estructura de datos, sino también para confirmar la conexión del dispositivo IoT a la nube.

Con estos primeros pasos ejecutados, antes de definir la plantilla o contenedor principal de la aplicación, es necesario definir los contenedores y funciones que definirán los aspectos gráficos de la información desplegada. Con aspectos gráficos se entiende colores, posición, tamaño o espacio ocupado, tipo de elemento y estilo como los más importantes. Los contenedores principales o paneles, después del que comprende toda la aplicación, son tres y el contenedor del banner. Estos paneles se refieren a diferentes datos, que son desplegados a partir de la información adquirida.

El primero a observar es el panel de estadísticas rápidas, que contiene el LED del ID del Operador o Sistema, el Termómetro o Indicador de la temperatura, el LED que sirve de contador del tiempo de pasteurización y el Botón de STOP o inicio y paro del proceso de monitoreo. El siguiente panel que se define es el superior de estadísticas, el cual es dividido en dos, por un lado, contiene un encabezado y filas de métricas o variables, y por el otro un gráfico de pastel. Cada fila métrica se compone del ID o identificación de dicha variable, el recuento de los datos recabados, un minigráfico de líneas y marcadores en forma de puntos, y los indicadores del estado fuera de control del proceso con respecto a la métrica en específico. En el caso del gráfico de pastel, se toma el tiempo de cada etapa del proceso y se almacena en forma de arreglo con tamaño del número de secciones que se desean en el gráfico, en este sistema se representan tres etapas: calentamiento, pasteurización y enfriamiento.

Finalmente, el gráfico de control es lo más llamativo de la interfaz, pues representa de la manera más completa y compacta el estado del sistema con respecto a una variable. Los parámetros que lo caracterizan, además de los aspectos gráficos, son los estadísticos, de donde una función anterior ya calculó y organizó dichos parámetros a partir de los datos que ya han sido adquiridos. En el gráfico de control se representan los parámetros estadísticos por medio de “anotaciones” y “marcas” que se configuran en el gráfico, lo que significa en su posición, color y estilo. Adicionalmente, se incluye el histograma de los datos para tener una visualización de la frecuencia en diferentes rangos, y en caso de que los datos se salgan de los límites de control, se define un arreglo para tomar los datos fuera de control y presentarlos en el gráfico con marcas de distinto color. El gráfico presenta los datos, pero solamente de una variable o métrica a la vez, para poder alternar entre la visualización de diferentes variables en caso de contar con más de una, el usuario debe de hacer clic en el identificador de la métrica de interés en el panel superior de estadísticas. De este modo, el gráfico de control es modificado para mostrar los datos de la métrica seleccionada.

En la Figura 4.16 se observa la segunda porción del código. Una vez que se han definido los aspectos gráficos y las funciones de procesamiento, que serán llamadas cada que se requiera hacer una actualización en el apartado correspondiente, se procede a definir el contenedor principal que define los parámetros de la aplicación. En el código, al pasar de la definición de la aplicación, ya se pueden reconocer las funciones reactivas y se pueden realizar cambios en la aplicación. Dentro de la definición de la aplicación también se definen ciertos componentes (dcc = dash core component), los cuales son: cuadro de diálogo (dcc.ConfirmDialog), componente de intervalo (dcc.Interval), componente de descarga de datos (dcc.Download), componente Markdown (dcc.Markdown) y componente de almacenamiento (dcc.Store). El componente de intervalo es una actualización iterativa que se realiza cada tiempo determinado, es decir, cada que dicho tiempo transcurra todas las funciones que tengan este componente como input o entrada para realizar cambios serán modificadas.

El componente de descarga guarda la información que por medio de alguna acción será descargada posteriormente. El componente Markdown se puede observar como un cuadro de texto, similar al cuadro de diálogo, pero que tiene otra presentación, muestra más texto y usualmente se muestra con la acción de un botón.

El componente de almacenamiento, a diferencia del de descarga, almacena los datos de algún arreglo, estructura de datos o variable, pero no para descarga y solo durante el tiempo de ejecución de la variable. Este componente se utiliza para guardar información que no forme parte de algún componente gráfico, es decir, datos de texto o numéricos. Los componentes gráficos cuentan con un identificador, lo que posibilita que se llamen elementos que se deseen cambiar de los mismos como se hace con los diccionarios. Los cambios se guardan dentro de las características de los componentes, lo cual significa que los cambios se observan de manera gráfica. Para el resto de información que deba de ser procesada y almacenada, aunque sea temporalmente, se hace uso de un componente de almacenamiento.

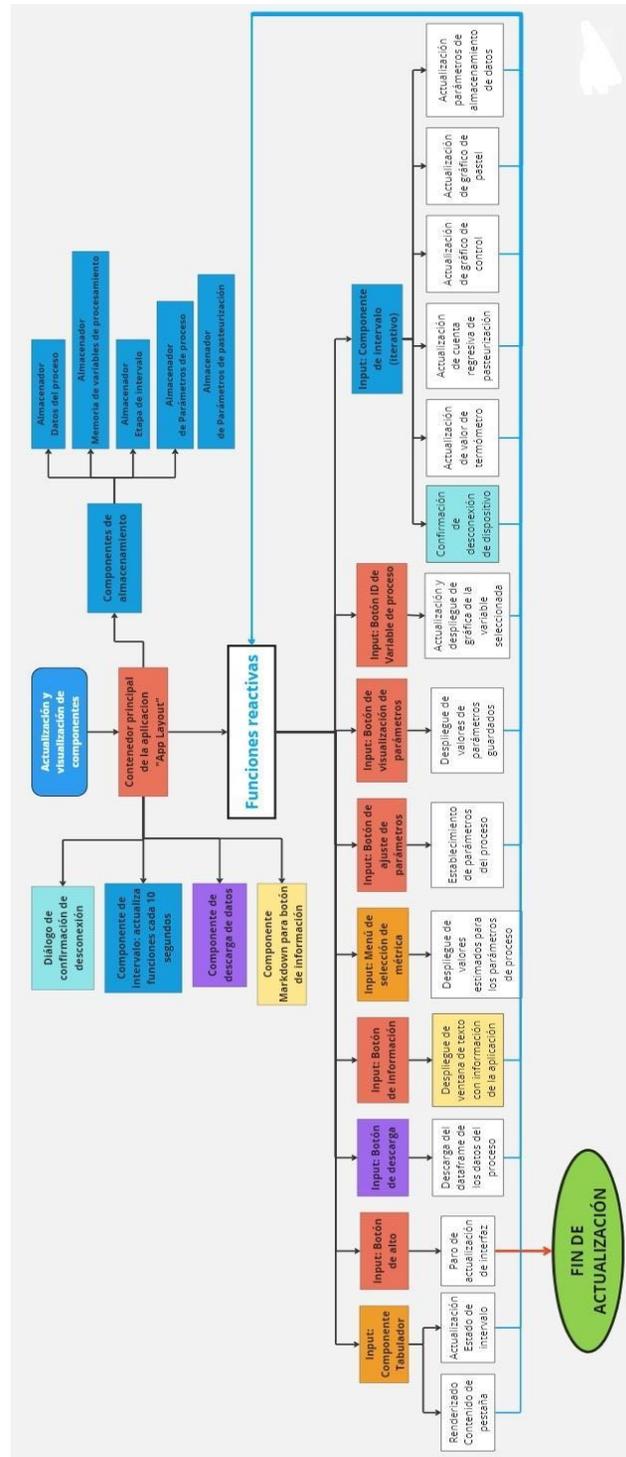


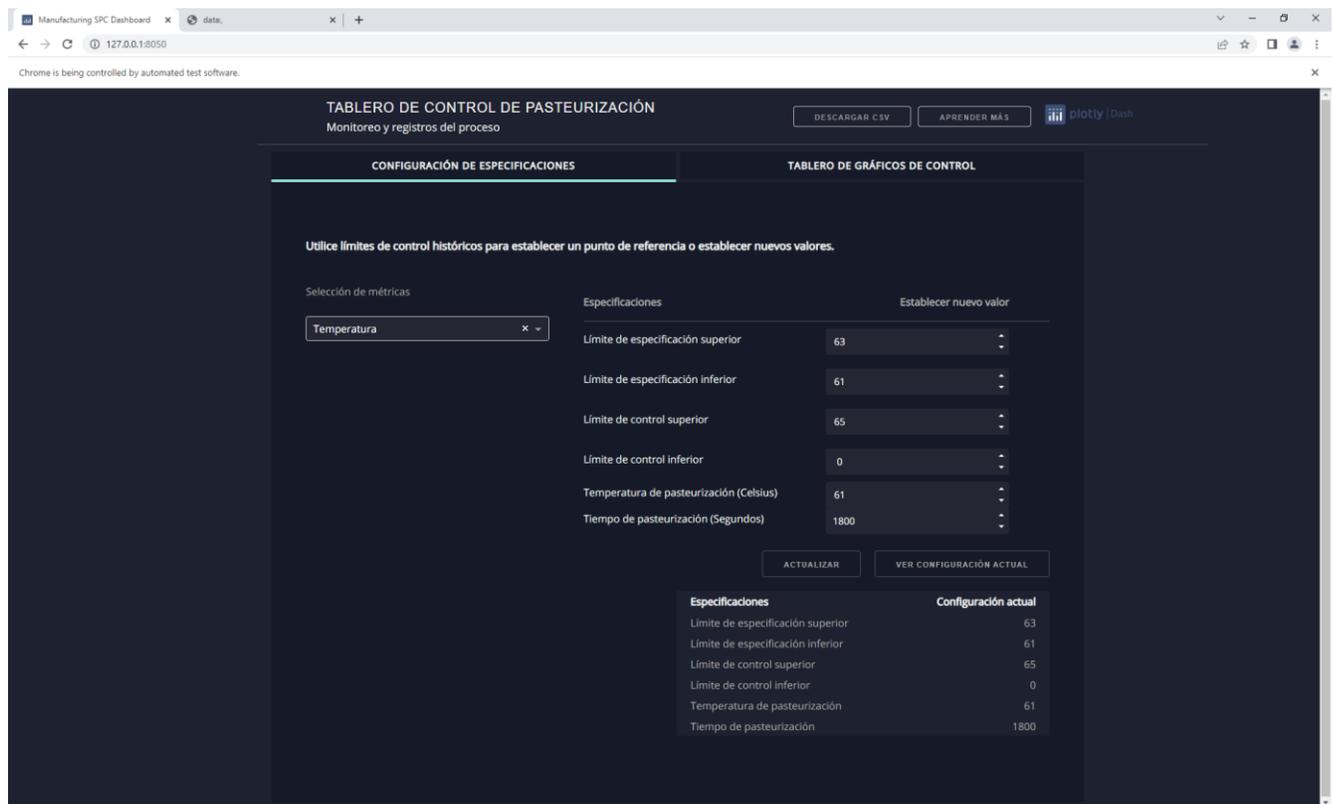
Figura 4.16. Diagrama de flujo del código de la interfaz dentro del llamado a las funciones reactivas.

Cuando el programador termina de definir el contenedor principal de la aplicación o “App Layout” y los componentes que se definen también en este apartado, continúa con la definición de las funciones reactivas, puesto que dependen de la configuración de la aplicación para entrar en funcionamiento. Las funciones reactivas responden a una entrada o “input” entregando una salida o “output”, si no ocurren cambios en dicha entrada, no ocurren cambios en la respectiva función. Por esta razón se incluyó en el diagrama de flujo las entradas que conducen a cambios en las respectivas funciones, en las cuales se describen los cambios más importantes. Los colores en el diagrama son útiles para distinguir entre distintos componentes, de donde los recuadros con mismos colores comparten una misma acción o funcionamiento.

En el diagrama se incluyen todas las funciones reactivas llamadas en el código. En Dash Plotly dichas funciones son referidas como “@app.callback”. Se establecen las variables de salida, las de estado y las de entrada, en ese orden, y posteriormente se define la función tal y como se realiza normalmente en Python. Las variables de estado funcionan de la misma manera que las de entrada, a excepción de ocasionar cambios en la función si sufren modificaciones en sus valores. En otras palabras, las variables de entrada ocasionan que una función reactiva produzca cambios o que el código entre a la función, mientras que las variables de estado no. Entonces solo se hará uso de las variables de estado cuando las variables de entrada ocasionen que la función reactiva produzca una salida [102]. En el lado izquierdo de la Figura 4.16 se muestran todas las funciones que no dependen del componente de intervalo, mientras que el lado derecho presenta todas las funciones que sí son modificadas mediante este componente, o asimismo que son actualizadas cada cierto tiempo. Para la aplicación, debido a la variación presentada en este sistema de pasteurización, se escogió un periodo de actualización de 10 segundos.

4.6.8. Configuración de parámetros

La aplicación requiere la asignación de límites de especificación y control, y los parámetros de pasteurización, es decir, temperatura y tiempo de pasteurización, para la realización de cada prueba. Este proceso varía acorde a las especificaciones correspondientes a la prueba a realizar. La Figura 4.17 presenta la pestaña de especificaciones, a donde el usuario siempre se debe de dirigir al iniciar la aplicación, de modo que la información desplegada coincida con los parámetros de supervisión y control del proceso.



The screenshot shows a web browser window displaying the 'Manufacturing SPC Dashboard' at the URL '127.0.0.1:8050'. The page title is 'TABLERO DE CONTROL DE PASTEURIZACIÓN' and it includes a 'plotly Dash' logo. The main content area is titled 'CONFIGURACIÓN DE ESPECIFICACIONES' and contains the following elements:

- A header: 'Utilice límites de control históricos para establecer un punto de referencia o establecer nuevos valores.'
- A 'Selección de métricas' dropdown menu with 'Temperatura' selected.
- A table for 'Especificaciones' with input fields for:
 - Límite de especificación superior: 63
 - Límite de especificación inferior: 61
 - Límite de control superior: 65
 - Límite de control inferior: 0
 - Temperatura de pasteurización (Celsius): 61
 - Tiempo de pasteurización (Segundos): 1800
- Buttons for 'ACTUALIZAR' and 'VER CONFIGURACIÓN ACTUAL'.
- A summary table at the bottom:

Especificaciones	Configuración actual
Límite de especificación superior	63
Límite de especificación inferior	61
Límite de control superior	65
Límite de control inferior	0
Temperatura de pasteurización	61
Tiempo de pasteurización	1800

Figura 4.17. Captura de la pestaña de configuración de especificaciones de la aplicación con los parámetros necesarios para la pasteurización VAT.

4.7. CARACTERIZACIÓN DEL SISTEMA - PRUEBAS DEL SISTEMA CON AGUA

En esta sección se presentan los resultados obtenidos de las pruebas, las cuales fueron realizadas una sola vez. La medición de la temperatura es útil para realizar el monitoreo y la acción de control con el método PID efectuado por el controlador PAC. Se muestran los gráficos de los datos recabados por el sistema de control y de monitoreo. La presentación de los datos recopilados al final del proceso es suficiente para realizar el análisis, no obstante, en la primera prueba se expone igualmente una captura de la interfaz durante el proceso. Con la presentación de la captura de la interfaz de monitoreo se busca reconocer su utilidad, puesto que, presenta igualmente los datos pero de manera precisa, clara y rápida.

4.7.1. VAT - 12L sin agitación

En la primera prueba se reconocieron las capacidades del calentador inmersivo para lograr que el agua alcanzara la temperatura de 63 °C sin la acción del agitador (ver Figuras 4.18 y 4.19). Debido a las condiciones atmosféricas al momento de realizar las pruebas, se comenzó el proceso con una temperatura aproximada de 30 °C. La etapa de calentamiento fue breve, tomando 10 minutos para alcanzar la temperatura de pasteurización. Sucedió lo anterior, puesto que, la posición de los sensores de temperatura hasta la cima del volumen del agua y el efecto de convección ocasionó que el agua más caliente se encontrara en la parte superior. Al encontrarse en la zona superior, los sensores solamente reportaron la temperatura más alta del líquido debido a la acción de la convección, ignorando de este modo la temperatura del resto del volumen.

El control de la temperatura fue activo durante toda la etapa de pasteurización, por lo que, encontramos varias formas de cresta en el gráfico a causa de la temperatura oscilante del agua entre los valores de 61.5 °C y 64 °C. Posiblemente el movimiento en el agua, provocado por el efecto de convección, se refleja en una disminución de temperatura, puesto que, los sensores de temperatura en un volumen de 12 litros no se encuentran totalmente sumergidos en el fluido.

La medición de un descenso en la temperatura resulta en un aumento en la potencia entregada al calefactor por la acción del controlador, por ello es por lo que en esta prueba el control de temperatura es activo durante toda esta etapa. El control fue preciso debido a que el sobrepaso en esta prueba fue de 1.5 °C por arriba de la consigna.

De la Figura 4.19 es posible identificar la temperatura en el momento de tomar la captura de la interfaz, asimismo el tiempo restante de la etapa de pasteurización que corresponde a "00:00", por ya haber finalizado, el gráfico de temperatura con respecto al tiempo, el histograma de los datos de temperatura en el lado derecho, y el gráfico de pastel con las proporciones del tiempo transcurrido en segundos en cada etapa del proceso. A causa de una desconexión del Photon con el Wi-Fi, la gráfica muestra una caída en los datos, puesto que, como es explicado en la Figura 4.15, se grafica un valor de 1 siempre que el Photon se encuentre desconectado. Sin embargo, la desconexión no significó gran diferencia debido al poco tiempo de desconexión durante la etapa de inicio. Para empatar los datos del Photon con los del controlador PID, se omitieron los datos iniciales del Photon para que coincidieran con el tiempo de inicio del PID, por ello no se encuentra la desconexión en la Figura 4.18.

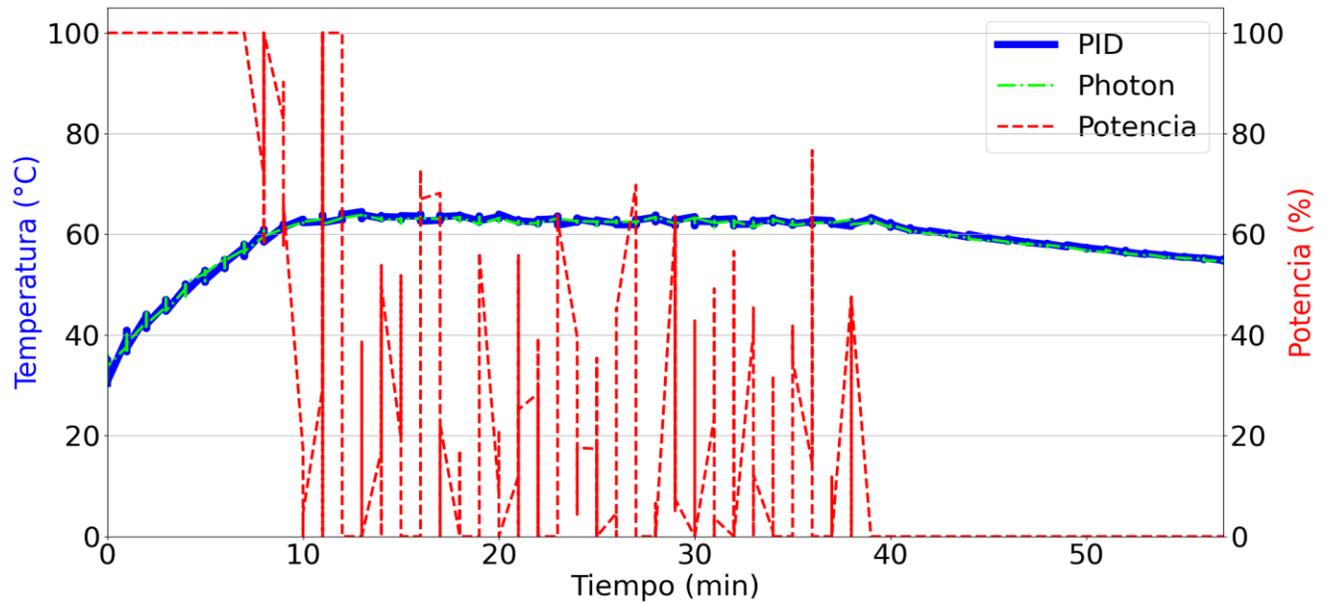


Figura 4.18. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 12 litros de agua sin agitación.

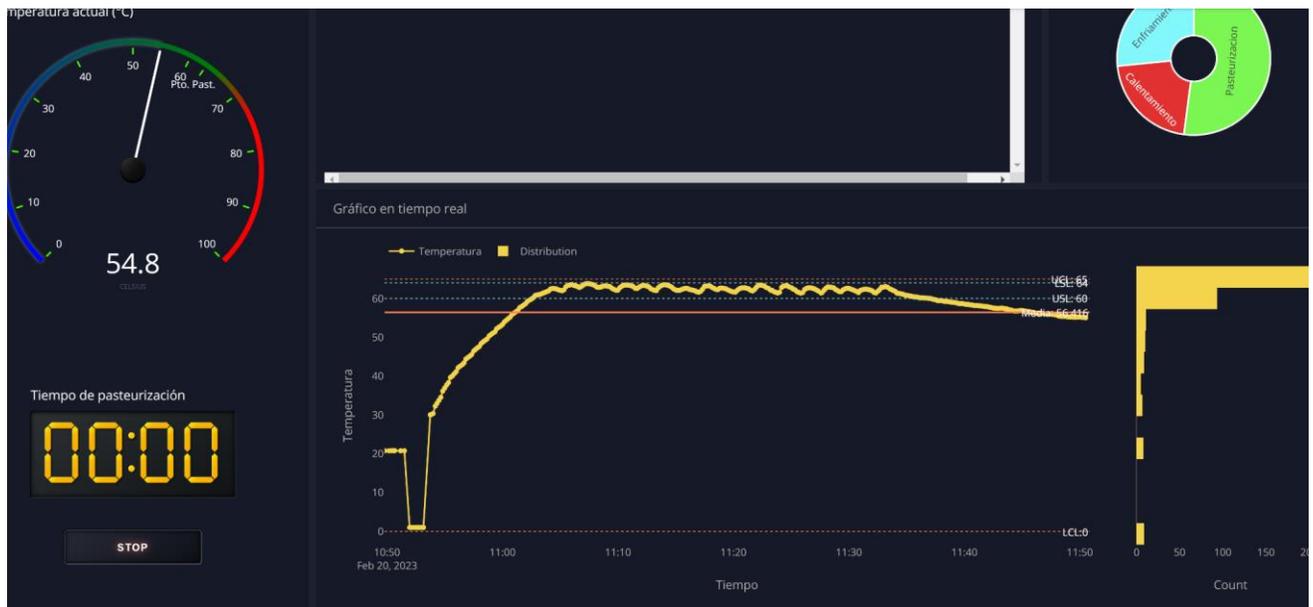


Figura 4.19. Captura de la interfaz de monitoreo del proceso VAT (63°C) con 12 litros de agua sin agitación.

4.7.2. VAT - 12L con agitación

La segunda prueba consistió en la misma configuración de la primera a excepción de la agitación (ver Figura 4.20). En esta como en todas las pruebas que incluyen la acción del agitador, se llevaron a cabo con el 15% al 20% de la capacidad que puede entregar el mezclador. Si se toma de referencia los datos técnicos que comparte el fabricante, el rango de velocidad en transmisión directa es de 500 a 7500 rpm, entonces se utilizó en un rango de 1550 a 1900 rpm. En comparación con la prueba anterior, se observa que la temperatura inicial es de aproximadamente 25 °C y el tiempo de calentamiento es de alrededor de 20 minutos. La acción de control es mucho menor después de los primeros 10 minutos de pasteurización, manteniendo la temperatura con menos de 20% de la potencia que puede entregar el calefactor. Es notable que la acción del agitador ocasiona una calefacción del líquido uniforme, por lo que la curva del gráfico es más suave y no se encuentran cambios abruptos a simple vista. Con el uso del agitador se nota también que el tiempo de calentamiento aumentó por 10 minutos, pero debe de tomarse a consideración que la temperatura inicial fue 5 °C menor que la primera prueba. Se continuó con la prueba después del tiempo de pasteurización, para conocer el enfriamiento pasivo que ocurre sin realizar cambios en el sistema. Se puede reconocer que la temperatura se mantiene de tal manera que para bajar 10 °C de 63°C a 53°C se requirió que transcurrieran alrededor de 60 minutos. El control fue de mayor precisión con respecto a la primera prueba, dado que se obtuvo un sobrepaso de 0.6 °C.

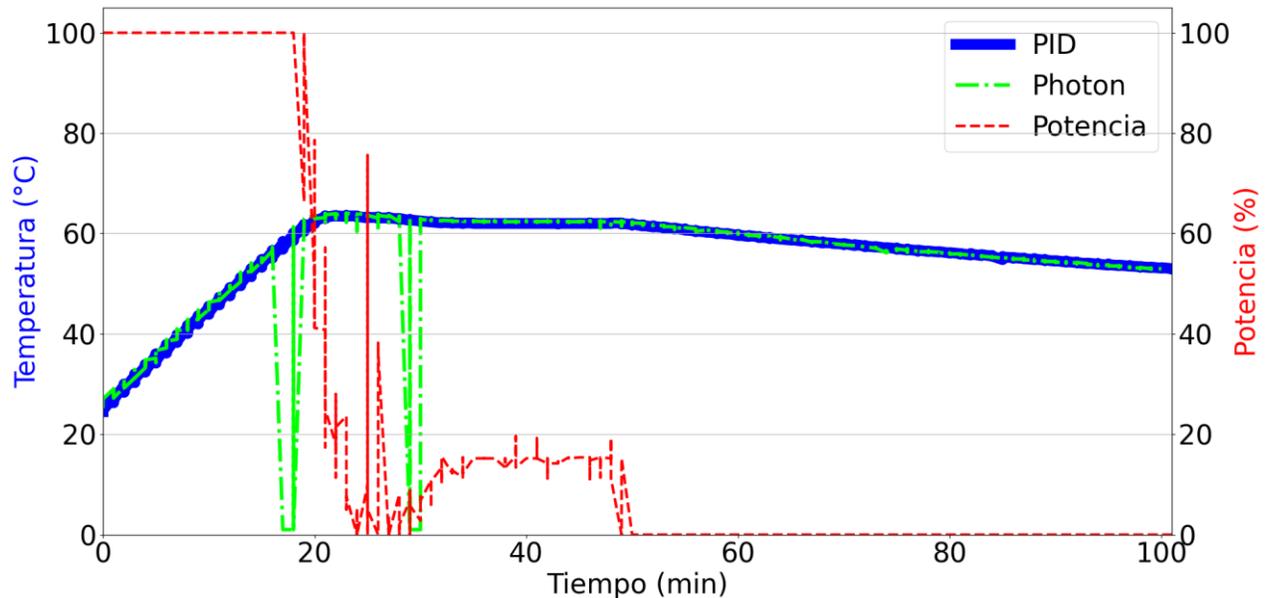


Figura 4.20. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 12 litros de agua y con agitación.

En esta prueba se encontraron fallos de conexión por parte del Photon, puesto que, no está exento de sufrir interferencias que conduzcan a pérdidas de señal. Dichos fallos de conexión se representan con un valor de “1” en el gráfico, es decir, en cada ocasión o tiempo en el que el Photon se encontrara “desconectado” de la red, el valor correspondiente a dicho tiempo es igual a “1”. En el momento de la prueba se reconoció a tiempo el problema, lo que se muestra posteriormente como fallos menores o que transcurrieron unos cuantos segundos.

4.7.3. VAT - 16L sin agitación

Posterior a las pruebas con 12 litros, se realizaron pruebas con la misma configuración, pero con un volumen mayor de modo que ocupara toda la capacidad del recipiente (ver Figura 4.21). La capacidad calculada a partir de las dimensiones del recipiente (30 cm de diámetro por 28 cm de alto en su espacio cilíndrico) es de aproximadamente 20 litros. Sin embargo, se decidió verter 16 litros de líquido para evitar el contacto con el cuerpo superior de los sensores de temperatura. Tomando las nuevas dimensiones en consideración, es razonable que el tiempo de calentamiento se extendiera y los cambios sucedieran con mayor suavidad, aún sin la acción del agitador. La temperatura inicial fue similar a la anterior, alrededor de 25 °C y el tiempo de calentamiento se extendió a aproximadamente 25 minutos. Puede interpretarse que con el aumento del volumen se tiene mayor inercia térmica, lo que ocasiona un mayor tiempo de calentamiento para alcanzar la temperatura de pasteurización y menores cambios en la temperatura en general. Se rescata que en esta prueba el monitoreo de los datos de temperatura fue preciso y sin fallos de conexión. La auto-sintonización del controlador mantuvo el mismo nivel de precisión en el control que en la segunda prueba con un sobrepaso máximo de 0.7 °C, siendo entonces la temperatura máxima de 63.7 °C. Es notable que las curvas de temperatura no coinciden completamente entre el PID y el Photon, esto puede deberse a la ausencia de agitación. La posición del sensor del controlador fue ligeramente más cercana a la del Photon, de este modo el sensor de temperatura del controlador probablemente experimentó primero los cambios, que por acción de la convección tras algunos segundos o un minuto se reflejaron en el sensor del Photon.

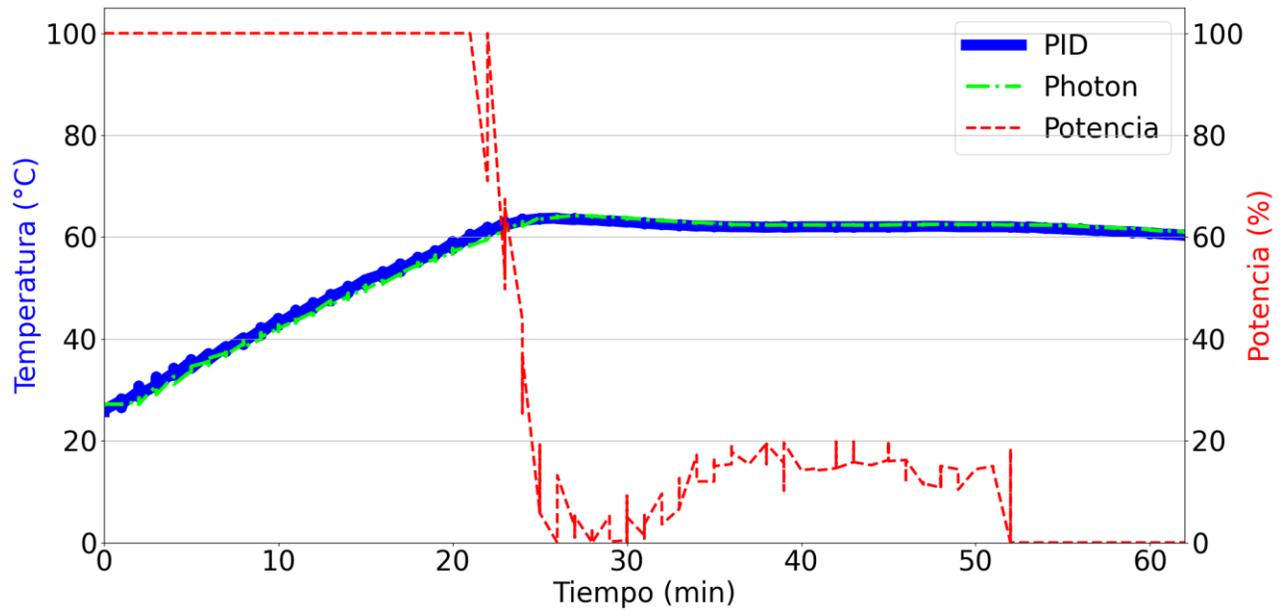


Figura 4.21. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 16 litros de agua sin agitación.

4.7.4. VAT - 16L con agitación

Al igual que en la prueba con 12 litros, se añade en este caso la acción del agitador para identificar cambios en el proceso de pasteurización (ver Figura 4.22). La temperatura de inicio fue ligeramente menor a 25 °C y el tiempo de calentamiento se extendió 5 minutos más que sin el uso del agitador. Con la inercia térmica de los 16 litros y una temperatura homogénea gracias al agitador, la participación del controlador fue ligeramente menor a la prueba sin agitación. Con esta configuración el tiempo de calentamiento y de pasteurización son prácticamente los mismos con 30 minutos por cada etapa. Igualmente, el monitoreo fue preciso, con dos disparidades en la medición de la temperatura por parte del Pt100 conectado al Photon, y sin fallos de conexión. Es posible que estas disparidades o cambios bruscos en la medición, exactamente en los minutos 4 y 35, surjan de la agitación constante que uniformiza la temperatura en el contenedor, lo que da a lugar a temperaturas distintas según sea la velocidad del agitador. La precisión del controlador en calentar el fluido a la temperatura y mantenerla es consistente con las dos pruebas anteriores, en vista de que se obtuvo un sobrepaso de 0.7 °C de nuevo.

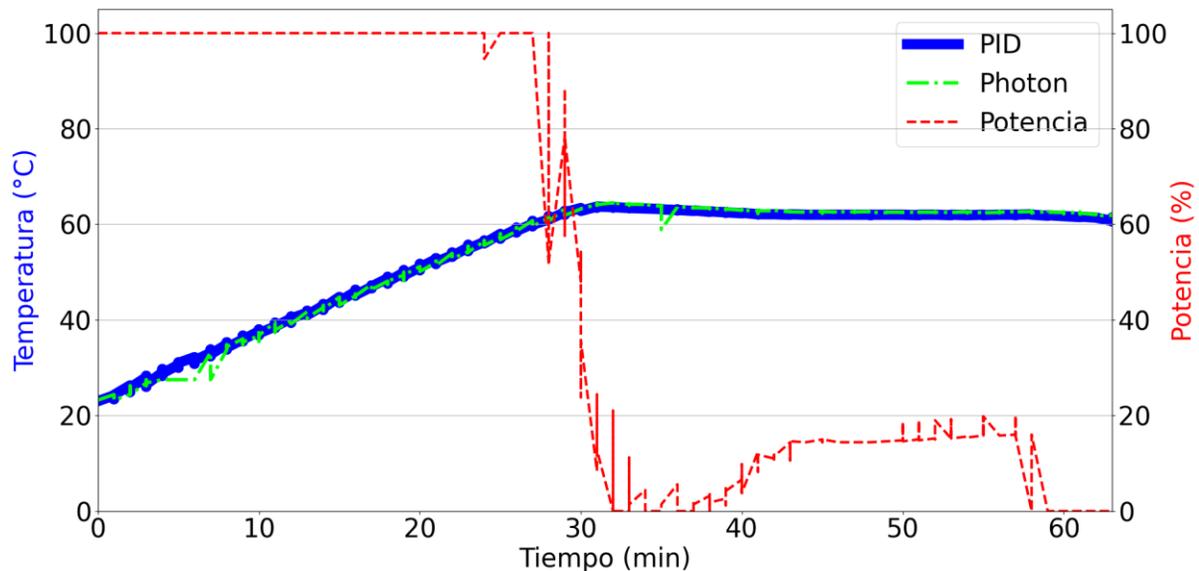


Figura 4.22. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 16 litros de agua y con agitación.

4.7.5. HTST - 12L sin agitación

A pesar de disponer de un calefactor, en lugar de un sistema más sofisticado para el intercambio de calor, para fines de caracterización se propuso de igual manera llevar a cabo pruebas de pasteurización lo más parecido al método HTST (72 °C, 15 segundos) (ver Figura 4.23). Con las posibilidades que ofrece el sistema elaborado, se buscó que el sistema alcanzara dicha temperatura y fuera capaz de mantenerla durante 1 minuto. En este caso, el factor limitante es la capacidad del calefactor, puesto que, de acuerdo con las especificaciones, la temperatura máxima que soporta es de 75 °C. La temperatura límite ocasiona que al detectarse dicha temperatura en el calefactor se bloquea el dispositivo hasta bajar la temperatura considerablemente, aunque el agua se encuentre a 70 °C o una temperatura menor a la limitante. No obstante, esta temperatura límite no ocasionó inconvenientes durante el proceso de esta prueba. Se alcanzó la temperatura de 72 °C después de 17 minutos, se mantuvo la temperatura por 1 minuto y se finalizó la prueba después de esperar a que el sistema se enfriara unos minutos. La auto-sintonización en el controlador debe darse de nuevo cambiando significativamente sus parámetros, puesto que, es desde esta prueba que se empleó el controlador para calentar a un punto de temperatura mayor. A pesar de la reconfiguración de los parámetros del controlador, el control es altamente preciso con un sobrepaso de 1.2 °C. Este sobrepaso ocasiona que la temperatura para este tipo de proceso se encuentre dentro del rango aceptable (71 °C a 73 °C). En lo que respecta al monitoreo, durante la etapa de calentamiento se observa una leve diferencia entre las temperaturas registradas, lo que es consistente con las pruebas sin agitación anteriores. En esta ocasión es la temperatura medida por el sensor del Photon mayor. Sin embargo, la diferencia en las temperaturas disminuye al acercarse al punto de pasteurización.

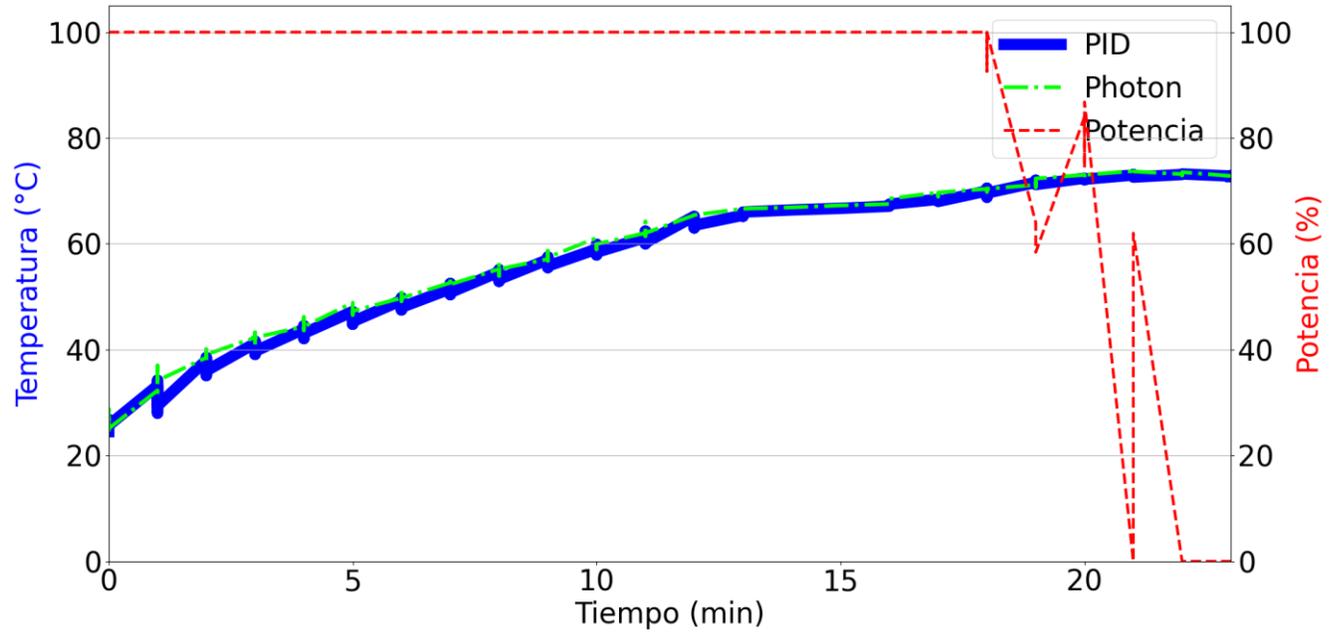


Figura 4.23. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 12 litros de agua sin agitación.

4.7.6. HTST - 12L con agitación

Incluyendo la agitación en el proceso, se consiguió una curva más similar a una recta con diferentes pendientes de acuerdo con la etapa del proceso (ver Figura 4.24). En comparación con la prueba sin la agitación, en esta prueba la etapa de calentamiento tuvo una duración cercana a los 27 minutos, lo que significó 10 minutos más. Es más notorio el cambio de la tasa de calentamiento con respecto a la de enfriamiento a causa del valor de ascenso y descenso de las pendientes. La potencia entregada por el controlador bajó a 90% al terminar la etapa de calentamiento, y a 50% aproximadamente durante la etapa de pasteurización. El control fue exitoso manteniendo la temperatura de pasteurización (72 °C) únicamente 1 minuto con un sobrepaso de 0.8 °C. En cuestión del monitoreo, se encontraron menos variaciones en el sistema de medición que en las mediciones tomadas por el sistema de control, lo que dio lugar a una curva más recta y un fallo de conexión en el minuto 30. La agitación dio lugar a una temperatura uniforme en el líquido, hecho que se refleja en la similitud de las temperaturas registradas por ambos sensores en comparación con la prueba sin agitación.

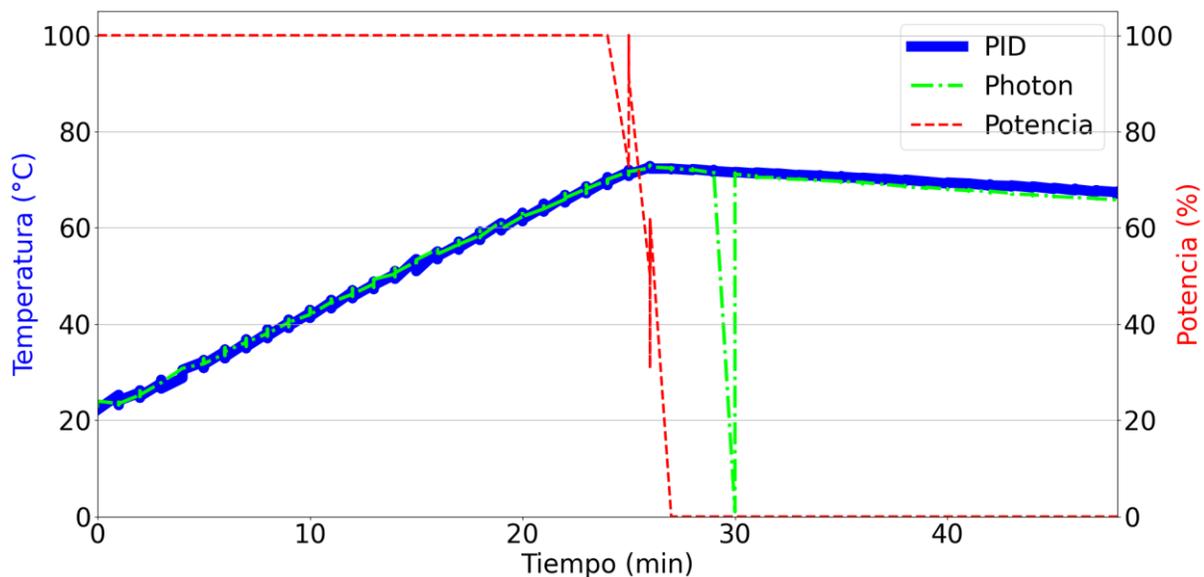


Figura 4.24. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 12 litros de agua y con agitación.

4.7.7. HTST - 16L sin agitación

Es en esta prueba con una temperatura de 72 °C de pasteurización y con 16 litros que se advierte que el sistema ya no consigue calentar el agua lo suficiente, puesto que logró calentar hasta llegar a un máximo de 70 °C (ver Figura 4.25). El calefactor llegó a la temperatura máxima de operación, pero debido al volumen del líquido y a la nula agitación la temperatura del agua no alcanzó los 72 °C, lo que ocasionó que el calefactor se bloqueara y cesara de calentar. Por lo tanto, se observan dos intentos de calentamiento o crestas, de donde sólo una alcanza el máximo. Esto significa que en cuanto el calefactor sufre cierta disminución en su temperatura, entra en operación de nuevo. Debido a la limitante del calefactor y no haber llegado a la temperatura de pasteurización, se observa que el suministro de potencia de parte del controlador es el máximo durante toda la prueba. En esta prueba las curvas del controlador y el Photon son prácticamente iguales sin fallos de conexión.

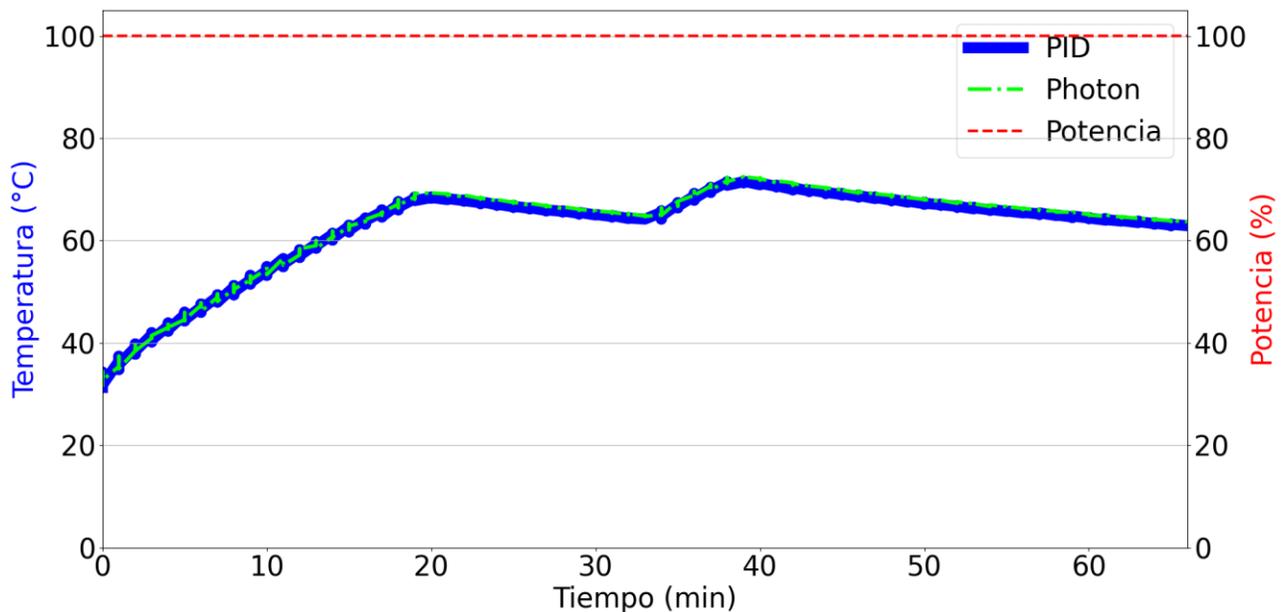


Figura 4.25. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 16 litros de agua sin agitación.

4.7.8. HTST - 16L con agitación

Similar a la prueba de 12 litros con agitación y a la de 16 litros sin agitación, por un lado, se obtuvo una curva más recta, y por el otro lado, la temperatura máxima alcanzada fue de 70 °C. La mejora en el calentamiento es resultado probablemente de la autosintonización. La temperatura máxima sin alcanzar el punto de pasteurización en la Figura 4.26, señala que con un volumen mayor el sistema es incapaz de alcanzar temperaturas superiores a 70 °C, por lo que, para ello se debe reemplazar el calefactor e implementar un sistema más robusto de intercambio de calor. Asimismo, se observa que la potencia entregada es del 100% durante todo el proceso. En este caso se obtuvieron 3 fallos de conexión de muy corta duración a los 17, 37 y 89 minutos aproximadamente, puesto que cada uno tuvo una duración de menos de 1 minuto. Por lo tanto, el monitoreo es muy preciso con respecto a la medición del controlador. Sin embargo, se deben tomar medidas para mejorar la conexión Wi-Fi entre el Photon y el modem de internet para evitar desconexiones en la medida de lo posible, sin prescindir de la ventaja que supone el uso de Wi-Fi con respecto a cables Ethernet.

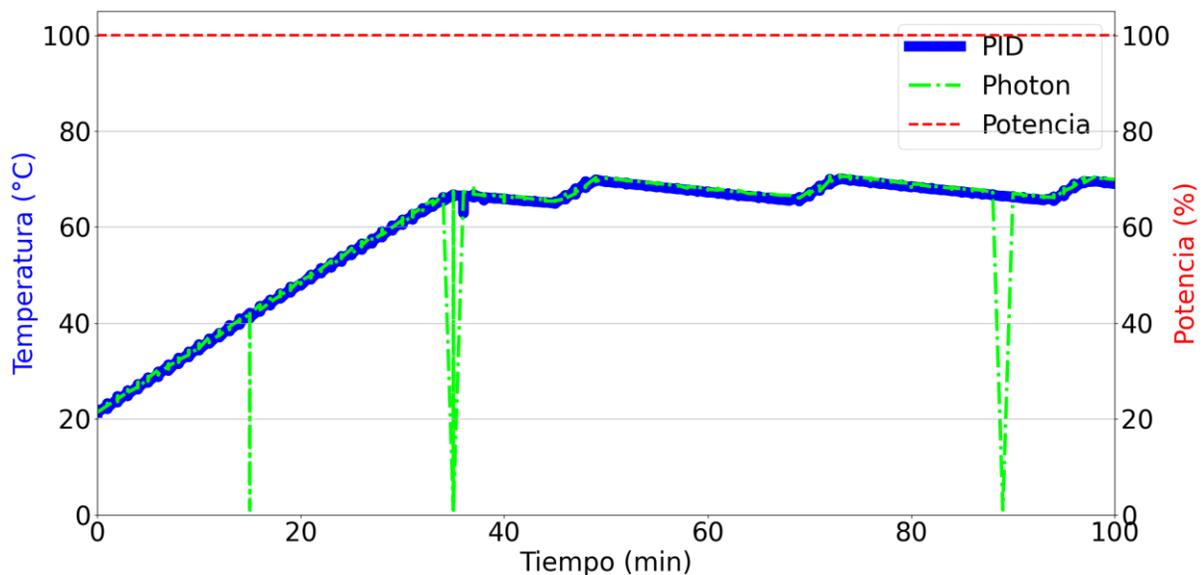


Figura 4.26. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso HTST (72°C) con 16 litros de agua y con agitación.

4.8. PRUEBAS FINALES - PRUEBAS DEL SISTEMA CON LECHE

Con la conclusión de las pruebas con agua para fines de caracterización del sistema, se identificó que la configuración adecuada incluye 16 litros de líquido para la correcta medición y control en el proceso de pasteurización. En las últimas pruebas se modificó únicamente el líquido por leche y la acción del agitador, con el fin de reconocer el comportamiento del sistema.

4.8.1. Prueba sin agitación

El gráfico de la prueba de pasteurización VAT con leche y sin agitación se presenta en las Figura 4.27, en donde la curva se asemeja al gráfico de la prueba con las mismas condiciones, pero con uso de agua (Ver Figura 4.21). Dicha diferencia produjo un tiempo de 29 minutos para alcanzar la temperatura de pasteurización y un sobrepaso de 1.1 °C. Esto significó 4 minutos y 0.4 °C más que en la misma prueba realizada con agua. En comparación, en este caso el suministro de potencia comenzó a disminuir alrededor de 4 minutos antes de alcanzar la consigna, con agua la potencia comenzó a disminuir poco más de 1 minuto antes de llegar a la consigna. La disminución abrupta en la potencia entregada por el controlador se debe al aumento rápido en la temperatura durante el final de la etapa de calentamiento. La acción de calentamiento sumada a la inercia térmica en la leche ocasionó que la temperatura del líquido alcanzara los 64 °C. El descenso de la temperatura fue muy lento tanto con agua como con leche, no se observaron cambios notorios a excepción de las diferencias antes mencionadas. La diferencia en el tiempo registrado de ambas gráficas, entre el controlador y el Photon, significa únicamente una desconexión temprana para visualizar por parte del controlador el descenso de la temperatura por un tiempo más prolongado. Igualmente, el monitoreo del Photon es muy preciso, con una ligera desviación durante la etapa de pasteurización, que, por experiencia en las pruebas con agua, puede reconocerse como un efecto de la ausencia de agitación en la prueba. En esta prueba no existieron fallas de conexión.

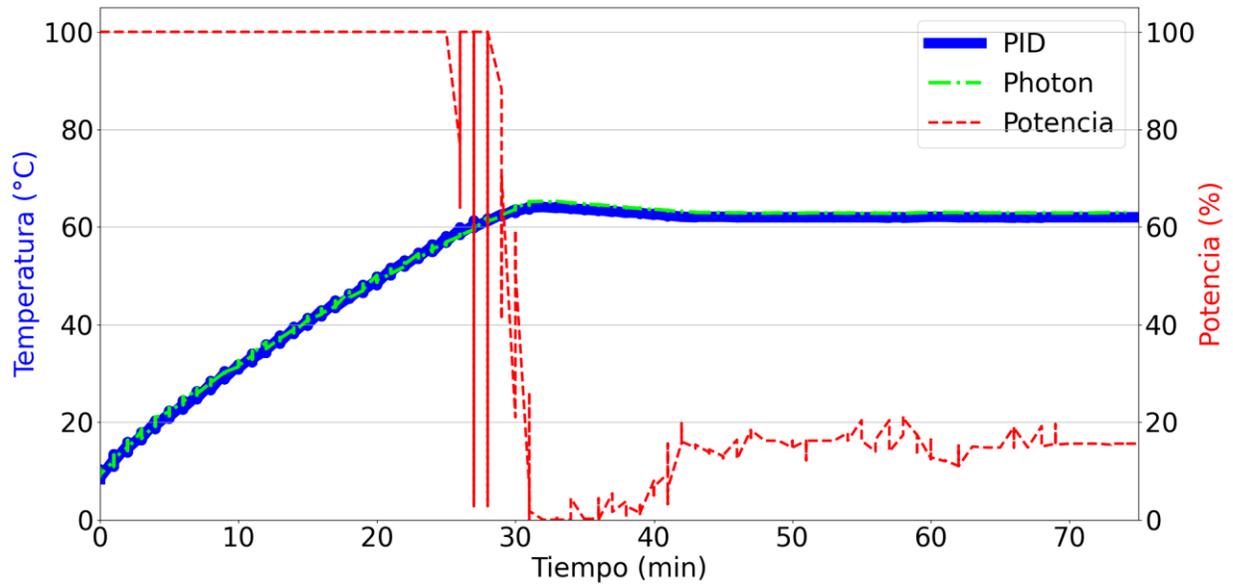


Figura 4.27. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63°C) con 16 litros de leche sin agitación.

4.8.2. Prueba con agitación

La prueba con agitación arrojó varias oscilaciones y cambios bruscos en la etapa de calentamiento y en la etapa de pasteurización (ver Figura 4.28). El flujo constante de la leche dentro del contenedor generado por el agitador para uniformizar la temperatura en el contenedor ocasionó en esta prueba que la temperatura medida por los sensores variara más que en pruebas anteriores. Estas variaciones son notables también en las mediciones registradas por el sistema de monitoreo, por lo que, el cambio brusco de temperatura es consistente en ambos sensores. Es probable que en lugar de que ocurrieran fallas en los sensores, se registraran dichas mediciones a causa del movimiento del líquido en el recipiente y los componentes presentes en la leche, los cuales no se encuentran uniformemente distribuidos en el líquido de manera natural. En la prueba anterior, puesto que no hubo agitación, posiblemente los componentes más densos se precipitaron al fondo del recipiente, dando a lugar un calentamiento en los componentes más ligeros. Es decir, solamente la porción líquida de la leche sufrió los cambios de temperatura en las pruebas sin agitación. De este modo, al agitar y uniformizar los componentes de la leche en esta prueba, se consiguen cambios más bruscos en las mediciones de temperatura de la leche. Se podría sugerir una agitación previa para que la distribución o difusión de la parte sólida de la leche fuera homogénea en todo el volumen al momento de realizar el proceso de pasteurización. Aunque se registraron cambios bruscos de temperatura, el controlador realizó su operación de manera adecuada con una rampa de calentamiento de 1.8 °C/min y un sobrepaso de 1.9 °C con respecto a las mediciones del sensor del controlador. El sistema de monitoreo sufrió 5 fallos de conexión y registró una medición de temperatura excesivamente fuera del rango cerca de los 25 minutos del proceso. Sin embargo, los fallos por lo general tuvieron una duración de 1 minuto al igual que en las últimas pruebas hechas con agua.

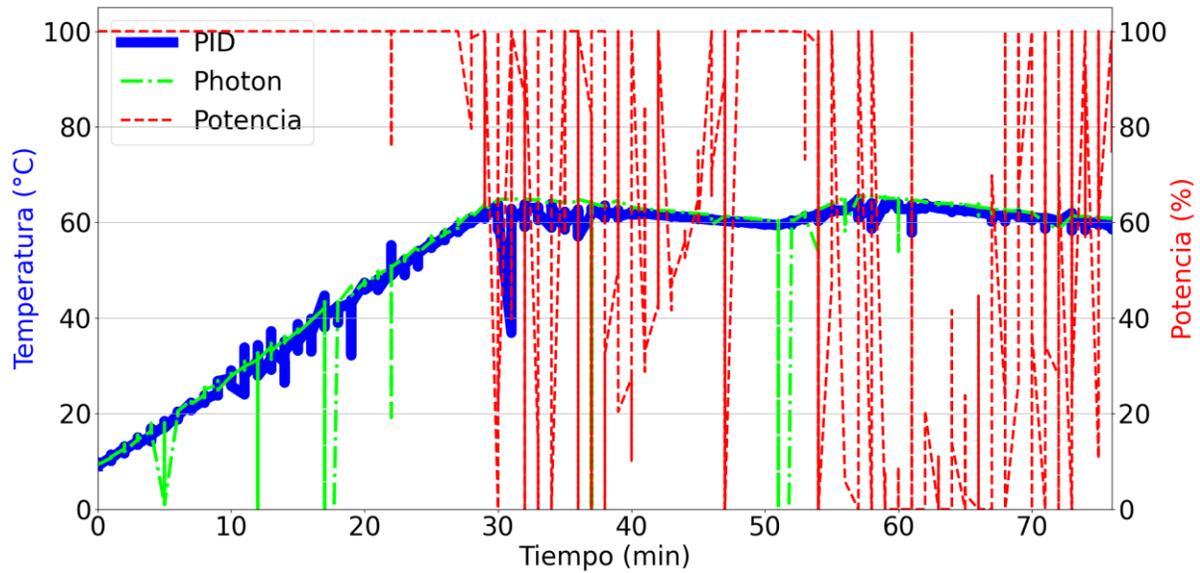


Figura 4.28. Datos de temperatura del controlador PID y del Photon, y la potencia entregada en el proceso VAT (63 °C) con 16 litros de leche y con agitación.

4.8.3. Análisis microbiológico

Adicional a las pruebas se realizaron análisis microbiológicos para otorgar la validez del control adecuado y preciso del sistema en el proceso de pasteurización con leche. Los análisis seleccionados para las muestras acorde a la NORMA Oficial Mexicana NOM-243-SSA1-2010 fueron: cuenta de organismos Coliformes totales, Salmonella spp, Escherichia coli, Listeria monocytogenes y cuenta de bacterias mesofílicas aerobias. En la Tabla 4.6 se observan los límites máximos de contenido microbiano para leche con respecto a la NOM-243-SSA1-2010. Se utilizan dichos límites para determinar con los resultados de los análisis el estado de la leche sin tratar y la leche pasteurizada.

Tabla 4.7. Límites máximos de contenido microbiano para leche y derivados lácteos.

Tomada de [2].

Microorganismo	Límite máximo	Productos
Organismos Coliformes totales	≤ 20 UFC/g o mL	En punto de venta: Leche, fórmula láctea, producto lácteo combinado; pasteurizados.
	≤ 10 UFC/g o mL	En planta: Leche, fórmula láctea, producto lácteo combinado; pasteurizados o deshidratados. Mantequilla, cremas, leche condensada azucarada, leche fermentada o acidificada, dulces a base de leche.
Staphylococcus aureus	≤ 10 UFC/ mL por siembra directa	Leche, fórmula y producto lácteos combinado pasteurizado.
	≤ 100 UFC/g o mL	Mantequilla, cremas, leche condensada azucarada, leche fermentada o acidificada, dulces a base de leche. Quesos madurados y quesos procesados

Salmonella spp	Ausente en 25g o mL	Leche, fórmula láctea, producto lácteo combinado: pasteurizados y deshidratados. Quesos frescos, madurados y procesados. Quesos de suero. Cremas, leche fermentada o acidificada, dulces a base de leche*, helados, sorbetes y bases para helados. Mantequillas.
Escherichia coli	≤ 3 NMP/g o mL	Leche utilizada como materia prima para la elaboración de quesos. Leche, fórmula láctea, producto lácteo combinado; deshidratados.
Listeria monocytogenes	Ausente en 25g o mL	Leche, fórmula láctea, producto lácteo combinado; pasteurizados ** Quesos. Quesos de suero. Helados, bases para helados y sorbetes.**.
Enterotoxina estafilococcica	Negativa	Leche, fórmula y producto lácteos combinado; deshidratados y la que se emplee como materia prima para elaboración de quesos. Quesos frescos, madurados y procesados. Helados, sorbetes y bases para helados.
Mesofílicos aerobios	100 000 UFC/g o mL	Bases para helado.

* UFC: Unidades Formadoras de Colonias

* NMP: Número Más Probable

Tabla 4.8. Cuadro comparativo de los resultados de los análisis microbiológicos de la leche sin pasteurizar en relación con los límites máximos de contenido microbiano de la NOM-243-SSA1-2010.

Microorganismo	Límite permitido	Muestra sin pasteurizar	Recomendada para ingesta
Organismos Coliformes totales	≤ 10 UFC/g o mL	140 UFC/mL	No
Salmonella spp	Ausente en 25 g o mL	Presencia en 25 g	No
Escherichia coli	≤ 3 NMP/g o mL	20 (V.E.) UFC/mL	No
Listeria monocytogenes	Ausente en 25 g o mL	Ausencia en 25 mL	Sí
Mesofílicos aerobios	100 000 UFC/g o mL	Mayor a 25 000 UFC/mL	Sí

* UFC: Unidades Formadoras de Colonias

* NMP: Número Más Probable

* V.E.: Valor estimado

Tabla 4.9. Cuadro comparativo de los resultados de los análisis microbiológicos de la leche pasteurizada en relación con los límites máximos de contenido microbiano de la NOM-243-SSA1-2010.

Microorganismo	Límite permitido	Muestra pasteurizada	Recomendada para ingesta
Organismos Coliformes totales	≤ 10 UFC/g o mL	3 (V.E.) UFC/mL	Sí
Salmonella spp	Ausente en 25g o mL	Ausencia en 25 mL	Sí
Escherichia coli	≤ 3 NMP/g o mL	10 (V.E.) UFC/mL	No
Listeria monocytogenes	Ausente en 25g o mL	Ausencia en 25 mL	Sí
Mesofílicos aerobios	100 000 UFC/g o mL	280 UFC/mL	Sí

Los informes de resultados de las pruebas microbiológicas, de donde se extrajeron los resultados presentados en las tablas de esta sección, pueden ser observados en el Anexo 4.

4.8.3.1. Muestra sin pasteurizar

El primer aspecto por revisar de acuerdo con la tabla de la norma y los resultados de los análisis (Ver Anexo 4) es la cuenta de organismos coliformes totales. De este análisis el resultado supera por 40 unidades el valor máximo aceptado en helados y por 130 unidades el valor máximo aceptado en planta.

Posteriormente, el resultado en Salmonella es presunto positivo denotando presencia en 25 g, el límite máximo aceptado es ausencia en 25g o mL. En el caso de E. Coli, se tiene un conteo de 20 UFC/mL (Valor estimado), y se reconoce ≤ 3 NMP/g como ningún tubo positivo en el análisis de dicha técnica, por lo que se encuentre por encima de lo permitido. Se reportó ausencia de Listeria monocytogenes en 25 mL, que coincide con los límites máximos. Por último, la cuenta de bacterias mesofílicas aerobias fue mayor a 25000 UFC/mL, el límite máximo es de 200,000 UFC/mL en Helados y sorbetes y de 100,000 UFC/mL en bases para helado. De la comparación de los resultados, se realizó la Tabla 9.

4.8.3.2. *Muestra pasteurizada*

En el caso de leche pasteurizada los métodos de análisis y comparación son los mismos. Los resultados obtenidos en los análisis marcan una diferencia significativa con relación a lo aceptado por la norma. En los organismos coliformes totales, la cuenta fue 7 unidades menor al límite. Con Salmonella en cambio se obtuvo ausencia. El resultado de E. Coli se acerca a 0 con respecto a la muestra sin pasteurizar, lo cual puede considerarse como no permitido, pero con un conteo reducido de bacterias. El análisis de Listeria presentó ausencia en 25 mL. Finalmente, la cuenta de bacterias mesofílicas aerobias es sustancialmente menor siendo 99,720 unidades menor al límite. La comparativa de los resultados con los límites de la norma se observa en la Tabla 10.

5. CONCLUSIONES

El sistema de pasteurización diseñado es capaz de enlazarse correctamente al Internet de las Cosas para fines de monitoreo remoto. El sistema permite que cualquier persona con acceso a internet con la aplicación de monitoreo, el lenguaje de programación y las librerías instaladas logre monitorear su proceso de pasteurización en tiempo real. El sistema fue orientado a los pequeños productores con relación al presupuesto de los materiales e infraestructura, y el nivel de complejidad que requiere para su empleo.

Se construyó la infraestructura del sistema con el uso de herramientas CAD para el diseño y el apoyo de colaboradores en el área de herrería para su fabricación. Se dispuso inicialmente de la infraestructura y el controlador PID para el ajuste de los parámetros PID por medio de pruebas con agua.

Se realizó una consulta minuciosa con el fin de conocer las características de hardware y software de la variedad de plataformas IoT disponibles en el mercado útiles para el monitoreo de la temperatura en el proceso de pasteurización. La revisión del estado del arte proporcionó diferentes opciones que consiguen el procesamiento y enlace IoT adecuados para el monitoreo del proceso. Sin embargo, tras la evaluación de los requerimientos del sistema de monitoreo se optó por el Photon de la compañía Particle.

La implementación del Photon en el sistema de monitoreo implicó la conexión de un convertidor análogo-digital de comunicación SPI para registrar las mediciones del sensor de temperatura Pt100. Posteriormente, el enlace al Internet de las Cosas precisó la reclamación del Photon con una cuenta de la misma compañía para asimismo vincular el dispositivo a la cuenta. La programación del método de adquisición y procesamiento de los datos se complementó con el número de identificación, la llave o “token” de acceso del dispositivo y la aplicación de las funciones especializadas para la publicación de eventos con el propósito de habilitar el envío de los datos a la nube de Particle.

Adicionalmente, se integró la visualización de los datos en una app desarrollada en Python con el uso de la librería Dash de la compañía Plotly para el monitoreo remoto desde cualquier dispositivo en el que Python pueda ser instalado.

La plataforma IoT representó una mejora en la medición, procesamiento y monitoreo del control de temperatura en el proceso de pasteurización. El Photon es capaz de registrar los datos de medición de diferentes variables y procesos independientemente y con ayuda de componentes, dispositivos o herramientas auxiliares de manera precisa y con alta resolución. El procesamiento de los datos puede ser realizado por varios dispositivos y de mayor potencia una vez que la información se encuentra en la nube. Del mismo modo, si la información se encuentra en la nube, todo dispositivo conectado a internet y con suficiente poder de cómputo es capaz de monitorear los datos, ya sea mediante la visualización presentada en la aplicación, el acceso a los datos o una visualización personalizada con acceso a los datos de la nube.

El prototipo de sistema de pasteurización propuesto es viable en relación con los análisis microbiológicos para su implementación en plantas de pequeños productores, puesto que, consigue cumplir cuatro de cinco parámetros microbiológicos de calidad satisfactoriamente establecidos por la norma oficial mexicana con respecto a la presencia de agentes microbianos. El parámetro no alcanzado se debe a condiciones de la leche, la infraestructura y la norma. Sin embargo, de acuerdo con la literatura [52] subiendo la temperatura uno o dos grados Celsius con el tiempo de la pasteurización LTLT, con relación a la norma aceptada, es probable que pueda erradicarse por completo la presencia de E. Coli y otros virus o bacterias [53]. Además de que en las pruebas se obtuvo un valor muy cercano a cero, de acuerdo con la normatividad que acepta valores menores a 3 NMP/g o mL.

En relación con los conceptos más relevantes de este proyecto que son el uso de tecnologías IoT, la pasteurización de leche y el monitoreo, se encontraron productos y trabajos que coinciden con estos términos de búsqueda. Por el lado comercial, Inducom [104] presenta el pasteurizador de leche marca GEA como preparado para IoT con opción de acceso remoto.

Este tipo de pasteurizador está disponible para capacidades de 3 000 hasta 10 000 litros por hora. Puesto que un pasteurizador de estas capacidades indica un tipo de pasteurización HTST o UHT, puede resultar muy productivo pero inasequible para pequeños productores.

En la academia se encontraron tres coincidencias. Como se presentó en el estado del arte, uno de los trabajos se relaciona a la pasteurización de leche para producción de helado [17]. A diferencia del enfoque de este proyecto, el método de registro de datos sigue una estrategia centrada en la evaluación de peligros y puntos críticos de control para rastrear las operaciones de procesamiento de productos alimenticios. Además, los datos de temperatura son registrados y enviados hasta terminar el proceso de pasteurización, por lo que no tienen relación con el proceso de monitoreo.

El segundo trabajo se refiere al monitoreo y control del rendimiento basado en IoT de un intercambiador de calor [18]. Aunque no se refiere explícitamente al proceso de pasteurización, es verdad que el uso de este tipo de sistemas se puede extender a sistemas de pasteurización de tipo HTST o UHT según sea el caso. No obstante, se desconocen los costos del sistema, de igual forma su integración en un sistema de pasteurización de leche.

El tercer y último trabajo encontrado en la literatura es el único que coincide con los conceptos principales del proyecto. De parte de la Universidad Antonio Nariño, ubicada en Cartagena, Colombia, Remberto A. Silgado presentó en el año en curso un sistema de monitoreo de temperatura de un pasteurizador de leche basado en IoT [105]. En contraste con este proyecto, Remberto Silgado hizo uso de un controlador PLC, una Raspberry Pi, un transmisor de temperatura y switches ethernet, en lugar de una antena Wi-Fi. Además, el proceso en que se realiza el monitoreo es de tipo HTST en lugar de LTLT. Por último, para la adquisición y almacenamiento de los datos se hizo uso de software MySQL como nube, y para la visualización de datos se implementó GRAFANA. Cabe destacar que estas soluciones corresponden al uso de una Raspberry Pi, igualmente software libre y otro tipo de control.

El uso de Raspberry Pi puede ser el adecuado para un sistema HTST o superior, pero ese no es el caso para uno de tipo LTLT. En caso de que la plataforma IoT seleccionada para el proyecto no cuente con una nube especializada para almacenar los datos por medio del internet, el uso de software libre se muestra como una propuesta viable. El software de visualización resulta atractivo a razón de su compatibilidad con el software de la nube utilizado. Sin embargo, es indispensable realizar una revisión profunda en las características que ofrece en relación con el uso de Dash en Python para presentar un comparativo detallado de ambas opciones. Finalmente, el sistema PLC, que es presentado como parte del dominio de control no está basado en IoT, al igual que el controlador PID propuesto en este proyecto. Por lo tanto, resta revisar el control en el esquema IoT como trabajo futuro.

El uso de un sistema de monitoreo basado en IoT puede expandirse a aplicaciones como el montaje de talleres de prensas, sistemas de electrosíntesis, predicción de la acción humana, detección de enfermedades de las hojas de las plantas, sistemas de lavado automático, conformado de metales inteligente, hogares inteligentes, predicción del tiempo en áreas que van de la ingeniería de manufactura hasta las ciencias biológicas, de acuerdo con Sridharan et al [18]. Debido a las aplicaciones presentadas, al amplio campo de aplicación que plantea este sistema y al crecimiento industrial en el estado de Querétaro, acorde al plan estatal de desarrollo [8], este sistema de monitoreo señala ser una solución conveniente para impulsar la tecnificación productiva que busca el gobierno del estado.

6. TRABAJO FUTURO

En esta sección se presentan las áreas de oportunidad con base en los resultados obtenidos en la realización de este proyecto.

1. Implementar el sistema de monitoreo con tecnologías de menor costo, mayor flexibilidad y versatilidad en su programación, módulos y acceso a la nube. Un ejemplo de este tipo de plataformas es el ESP32, un SoC creado por Espressif Systems, el cual dispone de varios módulos, programación en tres lenguajes y flexibilidad en el envío de los datos por medio de internet.
2. Ampliar el desarrollo de tecnologías IoT en pasteurización UHT, la cual es ampliamente utilizada por su incidencia en la productividad y economía.
3. Realizar una revisión comparativa profunda de las tecnologías de la información para el almacenamiento y administración de datos, además de tecnologías para su visualización y despliegue.
4. Es posible aplicar este sistema en otras áreas de innovación tecnológica adecuando otro tipo de plataformas, como puede ser tecnología de monitoreo biomédica, domótica, protección civil, entre otros.
5. Sustituir el controlador integrado LS-3400 por un Circuito Integrado de Aplicación Específico (ASICs), por ejemplo, un FPGA, del cual se puede diseñar y construir un controlador PID *ad hoc* con las necesidades particulares del usuario.

REFERENCIAS

- [1] G. J. Tortora, B. R. Funke, and C. L. Case, *Introducción a la microbiología*, 9th ed. Panamericana, 2013.
- [2] “NORMA Oficial Mexicana NOM-243-SSA1-2010, Productos y servicios. Leche, fórmula láctea, producto lácteo combinado y derivados lácteos. Disposiciones y especificaciones sanitarias. Métodos de prueba”, Secretaría de Salud, 2010. [Online]. Disponible en: <https://dof.gob.mx/normasOficiales/4156/salud2a/salud2a.htm#:~:text=3.40%20Pasteurizaci%C3%B3n%2C%20al%20tratamiento%20t%C3%A9rmico,algunas%20enzimas%20de%20los%20alimentos>.
- [3] Z. López, “Destaca estado en producción de leche,” Querétaro, 11-Jul-2019. [Online]. Disponible en: <https://www.eluniversalqueretaro.mx/cartera/destaca-estado-en-produccion-de-leche>. [Fecha de acceso: 24-Jun-2022].
- [4] J. J. Bravo, “Producción de leche en México y su comercio de lácteos con países del APEC,” in *Teoría, Impactos Externos Y Políticas Públicas Para El Desarrollo Regional*, R. Rosales, A. Mercado, A. Sánchez, D. Amparo, and C. Venegas, Eds. Universidad Nacional Autónoma de México, 2018.
- [5] R. A. Colima, “Crece la producción de leche en México: SAGARPA,” gob.mx. [Online]. Disponible en: <https://www.gob.mx/agricultura/colima/articulos/crece-la-produccion-de-leche-en-mexico-sagarpa-158944?idiom=es>. [Fecha de acceso: 24-Jun-2022].
- [6] O. Sotomayor, E. Ramírez y H. Martínez (coords.), “Digitalización y cambio tecnológico en las mipymes agrícolas y agroindustriales en América Latina”, *Documentos de Proyectos (LC/TS.2021/65)*, Santiago, Comisión Económica para América Latina y el Caribe (CEPAL)/Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO), 2021.
- [7] Secretaría de Salud del Estado de Querétaro (SESEQ), “Noticias: Pese a pandemia, sector primario incrementó 22.7% en ocupación,” Gob.mx, 2021. [Online]. Disponible en:

<https://www.queretaro.gob.mx/covid19/contenido/noticiasContenido.aspx?q=vUYGbsxLnliusSwAEKRALK3Db7X/wCKG>. [Fecha de acceso: 24-Jun-2022].

[8] Poder Ejecutivo del Estado de Querétaro, “Plan Estatal de Desarrollo 2021-2027,” 2021.

[9] Paladino, Fissore, and Neviani, “A low-cost monitoring system and operating database for quality control in small food processing industry,” *J. Sens. Actuator Netw.*, vol. 8, no. 4, p. 52, 2019.

[10] A. Mallik, A. B. Karim, Z. H. Md, and M. A. Md, “Monitoring food storage humidity and temperature data using IoT,” *MOJ food process. technol.*, vol. 6, no. 4, 2018.

[11] A. Yavari, D. Georgakopoulos, H. Agrawal, H. Korala, P. P. Jayaraman, and J. Karabotic Milovac, “Internet of things milk spectrum profiling for industry 4.0 dairy and milk manufacturing,” in *2020 International Conference on Information Networking (ICOIN)*, 2020.

[12] T. Alamirew, V. Balaji, and N. Gabbeye, “Comparison of PID controller with model predictive controller for milk pasteurization process,” *Bull. Electr. Eng. Inform.*, vol. 6, no. 1, pp. 24–35, 2017.

[13] Anang, S. Hadisupadmo, and E. Leksono, “Model predictive control design and performance analysis of a pasteurization process plant,” in *2016 International Conference on Instrumentation, Control and Automation (ICA)*, 2016.

[14] A. O. Amole, O. E. Olabode, D. O. Akinyele, and S. G. Akinjobi, “Optimal temperature control scheme for milk pasteurization process using different tuning techniques for a proportional integral derivative controller,” *Iranian Journal of Electrical and Electronic Engineering*, pp. 2170–2170.

[15] F. Firmansyah, I. A. Sulistyawan, S. Hadisupadmo, and A. Widyotriatmo, “Temperature control for a pasteurization miniplant using multiloop and cascade controllers,” in *2016 International Conference on Instrumentation, Control and Automation (ICA)*, 2016.

- [16] Y. Z. Maulana, S. Hadisupadmo and E. Leksono, "Performance analysis of PID controller, fuzzy and ANFIS in pasteurization process," 2016 International Conference on Instrumentation, Control and Automation (ICA), 2016, pp. 171-177.
- [17] A. Biscotti et al., "Internet of things and blockchain technologies for food safety systems," in 2020 IEEE International Conference on Smart Computing (SMARTCOMP), 2020.
- [18] M. Sridharan, R. Devi, C. S. Dharshini, and M. Bhavadarani, "IoT based performance monitoring and control in counter flow double pipe heat exchanger," Internet of Things, vol. 5, pp. 34–40, 2019.
- [19] S. Jadhav, S. V. Patil, T. C. Thanuja, M. P. Shivu and G. Shankar, "Monitoring of Industrial Water Usage by using Internet of Things," 2018 International Conference on Information, Communication, Engineering and Technology (ICICET), 2018, pp. 1-4.
- [20] Servicio de Información Agroalimentaria y Pesquera, "Escenario mensual de productos agroalimentarios," 2021.
- [21] A. Contreras, "Impulsan en Querétaro la producción de leche bovina," El Financiero, 05-Nov-2018. [Online]. Disponible en: <https://www.elfinanciero.com.mx/bajio/impulsan-en-queretaro-la-produccion-de-leche-bovina/>. [Fecha de acceso: 24-Jun-2022].
- [22] P. P. Ray, "A survey on Internet of Things architectures", Journal of King Saud University-Computer and Information Sciences, 2016.
- [23] Di Nisio, A., et al. "Design of a low cost multipurpose wireless sensor network." Measurements & Networking (M&N), 2015 IEEE International Workshop on. IEEE, 2015.
- [24] K. J. Singh and D. S. Kapoor, "Create Your Own Internet of Things: A survey of IoT platforms.," in IEEE Consumer Electronics Magazine, vol. 6, no. 2, pp. 57-68, April 2017, doi: 10.1109/MCE.2016.2640718.
- [25] "Architecture of internet of things (IoT)," GeeksforGeeks, 12-Jun-2020. [Online]. Disponible en: <https://www.geeksforgeeks.org/architecture-of-internet-of-things-iot/>. [Fecha de acceso: 27-May-2023].
- [26] A. Rayes and S. Salam, "The Things in IoT: Sensors and Actuators," in Internet of Things from Hype to Reality, London, UK: Springer, 2022, pp. 63–82.

- [27] N. Tewari, N. Deepak, M. Joshi and J. S. bhatt, "Comparative Study of IoT Development Boards in 2021: Choosing right Hardware for IoT Projects," 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), 2021, pp. 357-361, doi: 10.1109/ICIEM51511.2021.9445290.
- [28] Celesti A, Fazio M, Giacobbe M, Puliafito A, Villari M., "Characterizing cloud federation in IoT", In: 30th international conference on advanced information networking and applications workshops, 2016, pp. 93–98.
- [29] C. Gregersen, "A complete guide to microcontrollers for IoT," Nabto, 23-Jul-2020. [Online]. Disponible en: <https://www.nabto.com/iot-microcontroller-guide/>. [Fecha de acceso: 30-Sep-2022].
- [30] "How to select a Microcontroller for your IoT Devices," Particle. [Online]. Disponible en: <https://www.particle.io/iot-guides-and-resources/mcu-vs-soc-vs-microprocessor-for-iot/>. [Fecha de acceso: 22-Sep-2022].
- [31] "Difference between MCU and SoC," GeeksforGeeks, 04-Sep-2020. [Online]. Disponible en: <https://www.geeksforgeeks.org/difference-between-mcu-and-soc/>. [Fecha de acceso: 22-Sep-2022].
- [32] P. Abrahamsson et al., "Affordable and energy-efficient cloud computing clusters: The Bolzano raspberry pi cloud cluster experiment," in 2013 IEEE 5th International Conference on Cloud Computing Technology and Science, 2013.
- [33] G. R. Kanagachidambaresan, "Introduction to internet of things and SBCs," in Internet of Things, Cham: Springer International Publishing, 2021, pp. 1–18.
- [34] "System-on-chip vs single board computers," GeeksforGeeks, 29-Jul-2022. [Online]. Disponible en: <https://www.geeksforgeeks.org/system-on-chip-vs-single-board-computers/>. [Fecha de acceso: 26-Sep-2022].
- [35] "System-on-A-chip and single board computer: Advantages and disadvantages," Linear Microsystems (LMI), 04-Oct-2019. [Online]. Disponible en: <https://linearmicrosystems.com/system-on-a-chip-single-board-computers/>. [Fecha de acceso: 26-Sep-2022].

- [36] “System-on-Chip vs Single-Board Computer: A comparison guide,” Symmetry Electronics. [Online]. Disponible en: <https://www.symmetryelectronics.com/blog/system-on-chip-vs-single-board-computer/>. [Fecha de acceso: 26-Sep-2022].
- [37] K. J. Astrom and T. Hagglund, Advanced PID Control. Research Triangle Park: ISA, 2006.
- [38] S. Bennett, “The past of PID controllers,” IFAC Proc. Vol., vol. 33, no. 4, pp. 1–11, 2000.
- [39] N. Minorsky, Directional stability of automatically steered bodies, Journal of the American Society of Naval Engineers, vol. 342, pp. 280-309, 1922.
- [40] W. Cova, “Control PID Un Enfoque Descriptivo”, La Rioja: Universidad Tecnológica Nacional, 2005. [Online]. Disponible en: <https://docplayer.es/8308482-Un-enfoque-descriptivo-ing-walter-j-d-cova-universidad-tecnologica-nacional-facultad-regional-la-rioja-departamento-de-electronica.html>.
- [41] K. Ogata, Modern Control Engineering, 3a ed. Filadelfia, PA, Estados Unidos de América: Pearson Education, 1996.
- [42] Ziegler, J. G., & Nichols, N. B. Optimum settings for automatic controllers. trans. ASME, vol. 64, no. 11, 1942. Disponible en: <https://asmedigitalcollection.asme.org/fluidsengineering/article-abstract/64/8/759/1155342/Optimum-Settings-for-Automatic-Controllers?redirectedFrom=PDF>.
- [43] *Instrumentación inteligente LS-3000*, Desin Instruments, Barc., España, 2000.
- [44] J. W. Vásquez C., “Evolución y nuevas tendencias de los sistemas de control – PAC (Controladores analógicos programables),” Rev. inven., vol. 3, no. 5, p. 80, 2008.
- [45] N. G. Forero, “Normas de Comunicación en Serie: RS-232, RS-422 y RS-485,” Revista De La Facultad De Ingeniería De La Universidad Libre. [Online]. Disponible en: <https://docplayer.es/2707497-Normas-de-comunicacion-en-serie-rs-232-rs-422-y-rs-485.html>. [Fecha de acceso: 10-Mar-2023].

- [46] “Pasteurización”, SENASA, 28-Feb-2010. [Online]. Disponible en: <https://www.senasa.gob.ar/senasa-comunica/noticias/pasteurizacion>. [Fecha de acceso: 29-May-2022].
- [47] G. J. Tortora, B. R. Funke, and C. L. Case, “Introducción a la microbiología”, 9. ed. Panamericana, 2013.
- [48] M. J. Lewis, Physical properties of foods and food processing systems. Cambridge, England: Woodhead Publishing, 1990.
- [49] W. G. Savage, “La Pasteurización De La Leche Como Problema Técnico,” Oficina Sanitaria Panamericana, pp. 804–823, 1933.
- [50] K. Echeverria, “Ventajas y desventajas de los diferentes tipos de pasteurización,” TecnoSoluciones Integrales - TSI Group, 03-Jun-2020. [Online]. Disponible en: <https://tecnosolucionescr.net/blog/226-ventajas-y-desventajas-de-los-diferentes-tipos-de-pasteurizacion>. [Fecha de acceso: 29-May-2022].
- [51] Iáñez, E. (2003) “Agentes físicos”, Instituto de Biotecnología - Universidad de Granada. [Online]. Disponible en: <https://www.ugr.es/~eianez/Microbiologia/13agfisicos.htm>. [Fecha de acceso: 29-May-2022].
- [52] L. E. Pearce, B. W. Smythe, R. A. Crawford, E. Oakley, S. C. Hathaway, and J. M. Shepherd, “Pasteurization of milk: The heat inactivation kinetics of milk-borne dairy pathogens under commercial-type conditions of turbulent flow,” J. Dairy Sci., vol. 95, no. 1, pp. 20–35, 2012.
- [53] D. A. Evans, D. J. Hankinson, and W. Litsky, “Heat resistance of certain pathogenic bacteria in milk using a commercial plate heat exchanger,” J. Dairy Sci., vol. 53, no. 12, pp. 1659–1665, 1970.
- [54] P. Dumalisile, R. C. Witthuhn, and T. J. Britz, “Impact of different pasteurization temperatures on the survival of microbial contaminants isolated from pasteurized milk,” Int. J. Dairy Technol., vol. 58, no. 2, pp. 74–82, 2005.

- [55] M. A. Pitino, D. L. O'Connor, A. J. McGeer, and S. Unger, "The impact of thermal pasteurization on viral load and detectable live viruses in human milk and other matrices: a rapid review," *Appl. Physiol. Nutr. Metab.*, vol. 46, no. 1, pp. 10–26, 2021.
- [56] S. Watts, "A mini review on technique of milk pasteurization", *Journal of Pharmacognosy and Phytochemistry*, vol. 5, no. 5, p. 99, 2016. Fecha de acceso: Oct. 10, 2022. E-ISSN: 2278-4136. [Online]. Disponible en: <https://www.phytojournal.com/archives/2016/vol5issue5/PartB/5-4-36-907.pdf>.
- [57] J. H. Camacho-Vera, J. M. Vargas-Canales, L. Quintero-Salazar, and G. W. Apan-Salcedo, "Características de la producción de leche en La Frailesca, Chiapas, México," *Rev. Mex. Cienc. Pecu.*, vol. 12, no. 3, pp. 845–860, 2021.
- [58] "Pasteurizador LTLT Eléctrico - Tauber," Tauber - Equipamiento para industria alimenticia, 30-Apr-2021. [Online]. Disponible en: <https://tauber.mx/producto/pasteurizador-ltlt-electrico/>. [Fecha de acceso: 19-May-2023].
- [59] "eDairy Market," Edairymarket.com. [Online]. Disponible en: <https://edairymarket.com/products/id/es-asema--pasteurizador-duo-vat->. [Fecha de acceso: 19-May-2023].
- [60] M. Azizi-Lalabadi, N. R. Moghaddam, and S. M. Jafari, "Pasteurization in the food industry," in *Thermal Processing of Food Products by Steam and Hot Water*, Elsevier, 2023, pp. 247–273.
- [61] L. Gutiérrez, "Diseño y optimización de un sistema pasteurizador de leche para productores pequeños, mediante simulación en EES (Engineering Equation Solver)", Tesis de maestría, Centro de Investigación en Materiales Avanzados, Laboratorio Nacional de Nanotecnología, Nuevo León, México, 2010.
- [62] M. Tipán y D. Flores, "Diseño y construcción de un prototipo de pasteurizadora para el procesamiento de 50 litros de leche/hora", Tesis de licenciatura, Sede Quito, Universidad Politécnica Salesiana, Quito, Ecuador, 2018.
- [63] J. D. Garbanzo, E. G. Moreira y J. R. Rodríguez, "Diseño de un pasteurizador por lotes para la elaboración artesanal de quesos", Tesis de licenciatura, Escuela de ingeniería mecánica, Universidad de Costa Rica, San José, Costa Rica, 2015.

- [64] “What is Platinum Temperature Sensor – Introduction to PT100,” Milesight IoT - IoT Solution Provider, 2020. [Online]. Disponible en: <https://www.milesight-iot.com/blog/em500-pt100-lorawan-temperature-sensor-features-benefits-and-applications/>. [Fecha de acceso: 02-Feb-2023].
- [65] “Pt100 sensor working principle,” Process Parameters Ltd, 24-Jun-2019. [Online]. Disponible en: <https://www.processparameters.co.uk/pt100-sensor-working-principle/>. [Fecha de acceso: 02-Feb-2023].
- [66] “Comunicaciones RS-485 Modbus: Instrucciones avanzadas”, Desin Instruments, Barcelona, España. [Online] Disponible en: <https://www.manualslib.es/download/290912/Desin-Instruments-Das-8000-Serie.html>. [Fecha de acceso: 14-Mar-2023].
- [67] “Instrucciones: Adaptador de comunicaciones AC-1000”, Desin Instruments, Barcelona, España. [Online] Disponible en: <http://desin.es/pdf/HI-AC-1000-es-v1.pdf>. [Fecha de acceso: 14-Mar-2023].
- [68] “7100A User Manual”, Issue 4.4, Eurotherm, Worthing, Reino Unido, 2011. [Online] Disponible en: <https://www.eurotherm.com/products/discontinued-eurotherm-products/7100a-single-phase-power-controller/>. [Fecha de acceso: 14-Mar-2023].
- [69] “Calentador de agua de inmersión con cubierta de acero inoxidable 304, control inteligente de temperatura y termómetro digital LCD portátil, calentador de cubo portátil para el hogar y los viajes, camping, calefacción, esenciales, 1500 W”, Amazon, 3-Ago-2022. [Online]. Disponible en: https://www.amazon.com.mx/Calentador-inoxidable-inteligente-temperatura-calefacci%C3%B3n/dp/B0B7H5MDSS/ref=sr_1_2?keywords=calentador+de+agua+de+inmersi%C3%B3n&qid=1678981599&sr=8-2. [Fecha de acceso: 16-Mar-2023].
- [70] “Laboratory Equipment Catalog”, Troemner’s Talboys, New Jersey, EE.UU.. [Online] Disponible en: <https://www.novatech-usa.com/pdf/Talboys%20Laboratory%20Equipment%20Catalog.pdf>. [Fecha de acceso: 16-Mar-2023].

- [71] “Impulsor de dispersión de disco – Paleta de acero inoxidable para máquina de dispersión, varilla de 8 x 300 mm con disco de 3 cm y 8 cm”, Amazon, 14-Mar-2022. [Online]. Disponible en: <https://www.amazon.com.mx/Impulsor-dispersi%C3%B3n-dispersante-inoxidable-varilla/dp/B09SSSDPR9?th=1>. [Fecha de acceso: 16-Mar-2023].
- [72] “MAX31865: RTD-to-Digital Converter”, Rev 3, Maxim Integrated, California, EE.UU., 2015. [Online] Disponible en: <https://www.mouser.mx/datasheet/2/256/MAX31865-775735.pdf>. [Fecha de acceso: 15-Mar-2023].
- [73] “Módulo transmisor RTD PT100 MAX31865”, Naylamp Mechatronics - Perú. [Online]. Disponible en: <https://naylampmechatronics.com/sensores-temperatura-y-humedad/430-modulo-transmisor-rtd-pt100-max31865.html>. [Fecha de acceso: 16-mar-2023].
- [74] R.K. Robinson, “Modern Dairy Technology: Advances in Milk Products”, Vol. 2, 2nd Edition, United Kingdom: Blackie Academic & Professional, 1993. [Online]. Disponible en: https://link.springer.com/chapter/10.1007/978-1-4684-8172-3_5. [Fecha de acceso: 24-Mar-2023].
- [75] “Dairy and food engineering: Lesson 3. Thermal properties of milk and milk products,” Victoria University. [Online]. Disponible en: <http://ecoursesonline.iasri.res.in/mod/page/view.php?id=124053>. [Fecha de acceso: 24-Mar-2023].
- [76] J. I. Lucero, UNAD, Definición, Composición, Estructura y Propiedades de la Leche, 2017. [Online]. Disponible en: <https://silo.tips/download/definicion-composicion-estructura-y-propiedades-de-la-leche> [Fecha de acceso: 24-Mar-2023].
- [77] “Argon Dev Board, Bluetooth and Wifi”, Newark.com. [Online]. Available: <https://mexico.newark.com/particle/argn-h/argon-dev-board-bluetoothwifi/dp/84AC6511>. [Accessed: 06-Sep-2023].
- [78] “ESP8266 módulo Lot WiFi Compatible con arduino IDE,” Amazon. [Online]. Available: https://www.amazon.com.mx/tresd-print-tech-ESP8266-Compatible/dp/B08LSDS2S2/ref=asc_df_B08LSDS2S2/?tag=gledskshopmx-20&linkCode=df0&hvadid=486955325997&hvpos=&hvnetw=g&hvrnd=1218506608275

[514636&hvppone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=1010149&hvtargid=pla-1419815481312&psc=1](https://www.adafruit.com/product/3033). [Accessed: 06-Sep-2023].

[79] “Adafruit Feather 32u4 registrador de datos,” Hetpro-store.com. [Online]. Available: <https://hetpro-store.com/adafruit-feather-32u4-registrador-de-datos/>. [Accessed: 06-Sep-2023].

[80] Adafruit Industries, “Arduino 101 with Intel Curie,” Adafruit.com. [Online]. Available: <https://www.adafruit.com/product/3033>. [Accessed: 06-Sep-2023].

[81] “Raspberry Pi 4,” Agelectronica.com. [Online]. Available: <https://www.agelectronica.com/?n=RASPBERRY%20PI%203&ca=N&r=10&v=1>. [Accessed: 06-Sep-2023].

[82] “Smart IoT solutions and services for enterprises,” Particle. [Online]. Disponible en: <https://www.particle.io/enterprise-iot-platform/>. [Fecha de acceso: 15-Feb-2023].

[83] “About,” Particle. [Online]. Disponible en: <https://www.particle.io/about-particle/>. [Fecha de acceso: 15-Feb-2023].

[84] “Photon,” Particle. [Online]. Disponible en: <https://docs.particle.io/photon/>. [Fecha de acceso: 15-Feb-2023].

[85] “Workbench,” Particle. [Online]. Disponible en: <https://www.particle.io/workbench/>. [Fecha de acceso: 27-Feb-2023].

[86] “IoT command center,” Particle. [Online]. Disponible en: <https://www.particle.io/iot-command-center/>. [Fecha de acceso: 01-Mar-2023].

[87] Amazon.com. [Online]. Disponible en: <https://aws.amazon.com/es/types-of-cloud-computing/>. [Fecha de acceso: 07-Mar-2023].

[88] “¿Qué es PaaS?,” Microsoft.com. [Online]. Disponible en: <https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-paas>. [Fecha de acceso: 01-Mar-2023].

[89] “An integrated IoT platform-as-a-service,” Particle. [Online]. Disponible en: <https://www.particle.io/>. [Fecha de acceso: 01-Mar-2023].

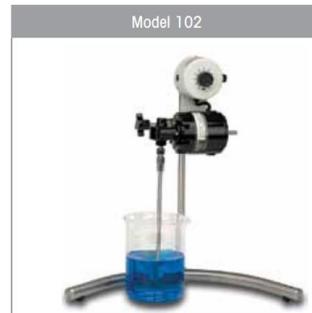
- [90] “Data operations,” Particle.io. [Online]. Disponible en: <https://docs.particle.io/getting-started/billing/data-operations/>. [Fecha de acceso: 19-May-2023].
- [91] “Photon datasheet,” Particle.io. [Online]. Disponible en: <https://docs.particle.io/reference/datasheets/wi-fi/photon-datasheet/>. [Fecha de acceso: 15-Mar-2023].
- [92] “Introduction,” Particle.io. [Online]. Disponible en: <https://docs.particle.io/getting-started/cloud/introduction/>. [Fecha de acceso: 15-Mar-2023].
- [93] “JsonParserGeneratorRK,” Particle.io. [Online]. Disponible en: <https://docs.particle.io/reference/device-os/libraries/j/JsonParserGeneratorRK/>. [Fecha de acceso: 17-Mar-2023].
- [94] “Particle.Publish() - cloud functions,” Particle.io. [Online]. Disponible en: <https://docs.particle.io/reference/device-os/api/cloud-functions/particle-publish/>. [Fecha de acceso: 17-Mar-2023].
- [95] “About Plotly,” Plotly.com. [Online]. Disponible en: <https://plotly.com/about-us/>. [Fecha de acceso: 02-Mar-2023].
- [96] “Collaboration,” Plotly.com. [Online]. Disponible en: <https://plotly.com/python/v3/ipython-notebooks/collaboration/>. [Fecha de acceso: 02-Mar-2023].
- [97] “Is Plotly for Python Free?,” Plotly.com. [Online]. Disponible en: <https://plotly.com/python/is-plotly-free/>. [Fecha de acceso: 02-Mar-2023].
- [98] P. Tinz, “A comprehensive guide to building enterprise-level Plotly Dash apps,” Towards Data Science, 02-Mar-2023. [Online]. Disponible en: <https://towardsdatascience.com/a-comprehensive-guide-to-building-enterprise-level-plotly-dash-apps-bd40dfe1313c>. [Fecha de acceso: 02-Mar-2023].
- [99] “Dash core components,” Plotly.com. [Online]. Disponible en: <https://dash.plotly.com/dash-core-components>. [Fecha de acceso: 02-Mar-2023].
- [100] “Dash DAQ,” Plotly.com. [Online]. Disponible en: <https://dash.plotly.com/dash-daq>. [Fecha de acceso: 02-Mar-2023].

- [101] “Dash HTML Components,” Plotly.com. [Online]. Disponible en: <https://dash.plotly.com/dash-html-components>. [Fecha de acceso: 02-Mar-2023].
- [102] D. Lettier, “Your easy guide to functional reactive programming (FRP),” Medium, 23-Oct-2018. [Online]. Disponible en: <https://medium.com/@lettier/functional-reactive-programming-a0c7b08f6b67>. [Fecha de acceso: 10-Mar-2023].
- [103] “Dash Tutorial: Part 3. Basic Callbacks,” Plotly.com. [Online]. Disponible en: <https://dash.plotly.com/basic-callbacks>. [Fecha de acceso: 10-Mar-2023].
- [104] “Pausterizador de leche – Sistema sanitario,” Inducom Bolivia, 02-Jun-2023. [Online]. Disponible en: <https://www.grupo-inducom.com/bo/producto/pausterizador-de-leche-sistema-sanitario/>. [Fecha de acceso: 28-Sep-2023].
- [105] R. A. Silgado, “Desarrollo de un sistema de Internet de las Cosas Industrial para el monitoreo de temperatura de un pasteurizador de leche,” Tesis de licenciatura, Universidad Antonio Nariño, Cartagena, Colombia, 2023. Disponible en: <http://repositorio.uan.edu.co/handle/123456789/8552>.

ANEXOS

ANEXO 1: DATOS TÉCNICOS DE LOS DISPOSITIVOS

1.1. Características técnicas del Mezclador de propósito general Talboys



Model 102

- Top mounted Basic Speed Control
- Provides increased torque with 10:1 gear reduction
- Offers the capability of two mixing shafts for multiple applications and added versatility
- Mixes up to two gallons of lightweight oils

Recommended accessory shafts:
151, 153B, 156, 161, 166

Figura A.1. Imagen representativa del mezclador de propósito general Talboys modelo 102. Tomado de [70].

Tabla A.1. Especificaciones del mezclador de propósito general Talboys modelo 102. Tomado de [70].

Modelo		102
Caballos de fuerza		1/75
Rango de velocidad	Transmisión directa	500 a 7500 rpm
	10:1	50 a 750 rpm
	Transmisión directa	0.15 in-lbs

Torque	10:1	1 in-lbs
Eléctrico		120 volt, 0.33 amps
Tamaño del eje		5/16" (7.9 mm)
Tipo de mandril		Coronilla
Dimensiones totales		6.5 x 8.5 x 7" (16.5 x 21.6 x 17.8 cm)
Peso de envío		5.1 lbs (2.3 kg)

1.2. Especificaciones del adaptador AC-1000 de Desin Instruments

1.2.1. Conexión en RS-485 con un LS-3000

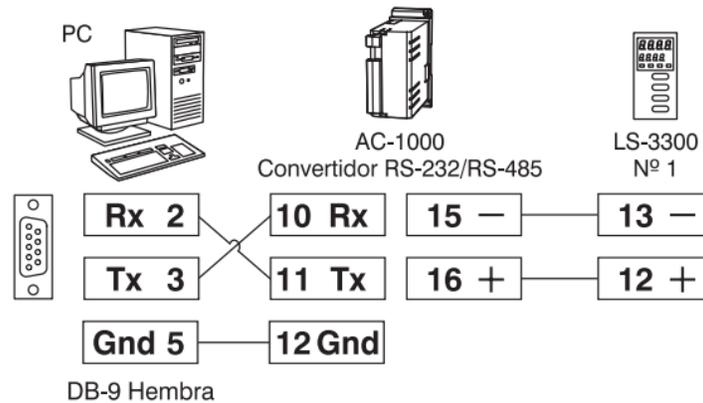


Figura A.2. Esquema de conexión del convertidor AC-1000 para enlazar un controlador de la serie LS-3000 con una computadora. Tomado de [67].

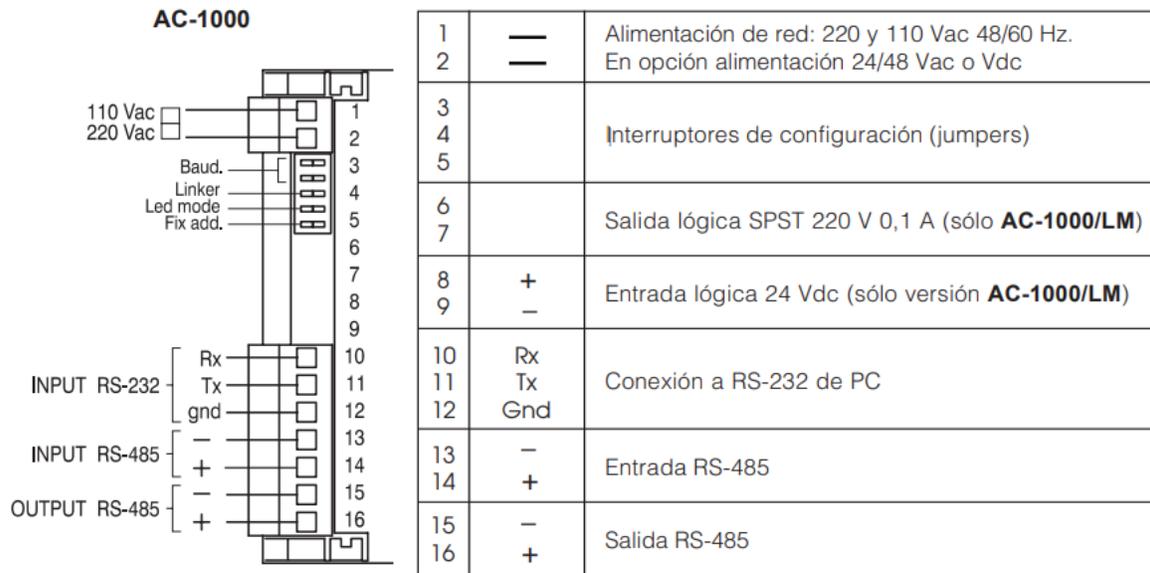


Figura A.3. Esquema de conexiones del convertidor AC-1000. Tomado de [67].

1.3. Controlador PID LS-3400

Características técnicas del controlador LS-3400 pueden ser encontradas en el manual *Instrumentación inteligente LS-3000* [43].

1.3.1. Software de Programación de apoyo para LS-3400

La generación LS-3000 es un conjunto de dispositivos programables de tipo PAC, destinados al control de procesos industriales con variables analógicas y lógicas. Todos los controladores de esta generación disponen de serie de un puerto de comunicación RS-485 MODBUS con capacidad de conectarse a cualquier red de datos en campo. La personalización y ajuste de parámetros puede realizarse localmente por el teclado o por el bus de comunicación RS-485 MODBUS, con el programa especial Loop-Win (en Windows 95/98 o NT) desde la computadora. En caso de que la computadora carezca de un puerto RS-485, es posible hacer uso del convertidor AC-1000, para realizar la conversión de RS-485 a RS-232. La monitorización o visualización de los datos adquiridos por el sistema puede realizarse mediante el programa de PROASIS DCS-Win [43].

1.3.1.1. PROASIS DAS-Win

Este programa sirve como apoyo para la visualización y almacenamiento de los datos del controlador en una computadora. Es un software de gestión y presentación de datos de medidas y señales lógicas, adquiridas por una red de hasta 253 módulos DAS-8000 y/o LS-3000 comunicados por RS-485 a una computadora compatible para aplicación en procesos industriales [43].

El programa realiza diagramas y gráficos para representar las medidas analógicas digitales de forma continua en tiempo real. Genera simultáneamente ficheros históricos, para su gestión posterior de forma gráfica o numérica. Dispone de una aplicación gráfica para el diseño de sinópticos de proceso y un simulador de registro gráfico de hasta 999

cartas, que simula configurables y visualizables en múltiples ventanas con datos históricos, sinópticos, alarmas, etc.

El programa se encuentra dividido en dos aplicaciones: el programa de configuración y el programa de ejecución.

- PROASIS DCS-Win Configuración: por medio de este se modifican y configuran todos los parámetros del sistema.
- PROASIS DCS-Win: encargado de la presentación de la información en forma sinóptica, registro de medidas y gestión del resto de funciones.

1.3.1.2. PROASIS DCS-Win

Este software es de gestión de control del sistema de control desde la computadora. Por medio del programa, el usuario tiene la posibilidad de observar ventanas con diagramas sinópticos con información digital de los datos y las medidas. Además, el programa permite utilizar fotografías del proceso como gráfico sinóptico con las mismas posibilidades que los dibujos, utilizando todas las posibilidades de Windows en la gestión de gráficos, simultaneidad de pantallas, íconos, etc [43]. Con las herramientas que se facilitan con el programa, pueden abrirse campos dentro de la imagen del sinóptico, en forma de pequeñas ventanas asignables a datos, variables, textos o niveles analógicos del proceso. Estos campos pueden corresponder a variables medidas, datos lógicos del estado de las entradas o salidas digitales, textos de denominaciones del proceso y además campos de nivel analógico proporcional a la señal indicada.

El programa cuenta con diagramas y gráficos prediseñados, emulando a los clásicos multipunto de papel, totalmente configurables por grupos, tags y denominación del proceso, así como parametrizaciones en canales, rangos, unidades, velocidad del registro (Figura A.4). De acuerdo a las capacidades del programa, se pueden generar hasta 999 gráficos en tiempo real de 6 vías de trazo continuo, tanto de señales analógicas como señales lógicas en múltiples ventanas.

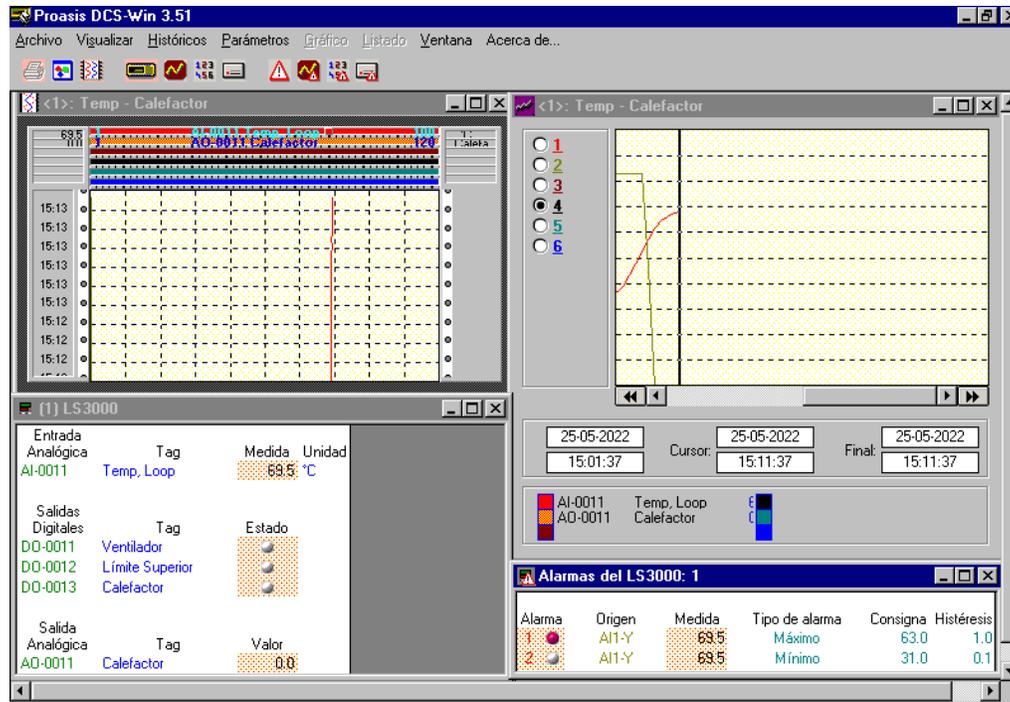


Figura A.4. Ejemplo de la visualización de los datos y medidas en la forma de distintos paneles de registro lógico y analógico.

Los datos registrados en forma de históricos pueden ser revisados mediante pantallas gráficas con facilidades para búsqueda por fechas, manual y automática, cambio de rango y zoom infinito hasta representar los límites mínimos de una medida guardados en el histórico. El número de registros guardados depende de la frecuencia de grabación que puede ser preseleccionada desde 1 segundos hasta 9999 segundos.

1.3.1.3. Loop-Win

Se refiere al programa de configuración desde la computadora, que consta de los bloques funcionales preprogramados. La interconexión de estos bloques puede realizarse por el teclado del controlador siguiendo los mnemónicos de la pantalla o por comunicación con la computadora a través del software de Loop-Win.

El programa sirve de herramienta de apoyo para programar, configurar y parametrizar todos los modelos de la serie LS-3000 de DESIN Instruments. El software permite adicionalmente calibrar los equipos. El programa permite programar los instrumentos físicamente instalados en el sistema por medio de los bloques funcionales preprogramados (Ver Figura A.5) [43]. El programa Loop-Win no es imprescindible para la configuración del dispositivo, no obstante, se recomienda por parte del fabricante utilizarlo, de modo que se consiga la configuración con mayor rapidez y eficacia en contraste con el uso del teclado.

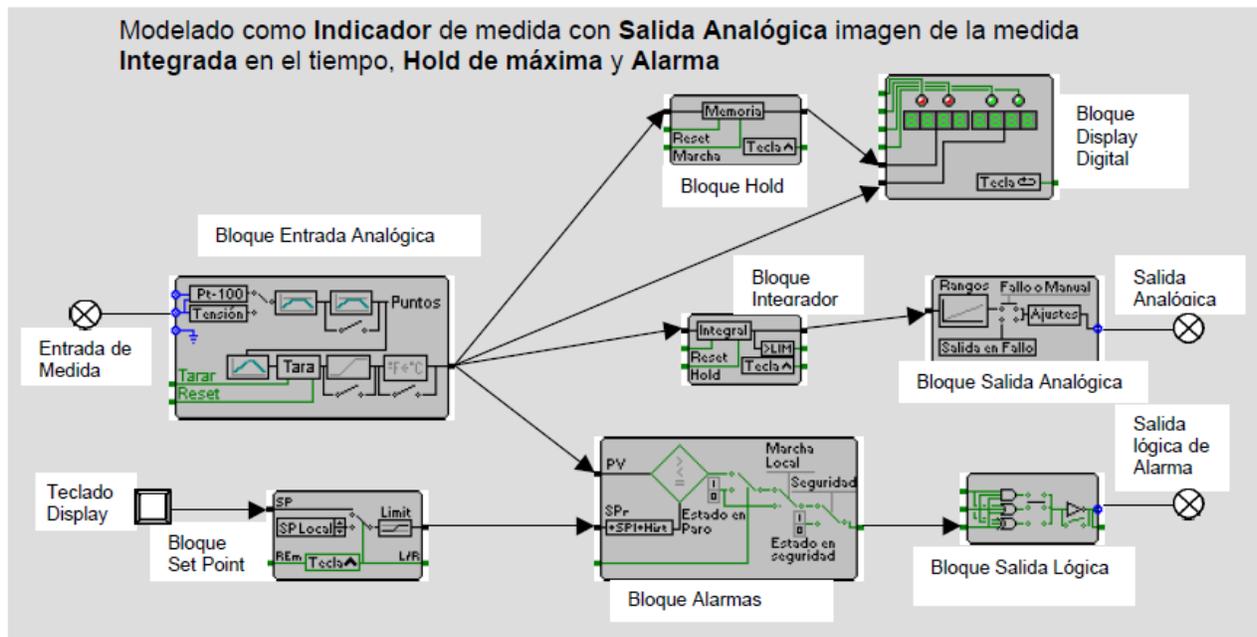


Figura A.5. Ejemplo de una aplicación configurada mediante Loop-Win. Tomado de [43].

1.3.1.4. Proceso de programación

La programación de Loop-Win para el control de un nuevo proceso puede entenderse como la selección y configuración de instrumentos de medición y control. Antes de pasar a los instrumentos, el programa ya posee una variedad de procesos prediseñados y precargados de fábrica, de modo que se omite el paso de selección de instrumentos y conexión lógica entre los mismos. Se le denomina como “estructura” a la conexión que forman entre sí los bloques funcionales mediante hilos virtuales después de ser configurados. En caso de no elegir una configuración prediseñada, se muestran todos los bloques, pudiendo ser movidos por la pantalla y unidos hasta formar la aplicación requerida. Como resultado se genera una estructura de bloques funcionales que definen la estructura o secuencia del proceso que representa la función precisada. Cuando se ha diseñado la estructura y se desea reprogramar con ella al controlador, es necesario cargarla en su memoria enviando el archivo desde la computadora.

1.4. Características técnicas del Particle Photon

1.4.1. Marcas de entradas

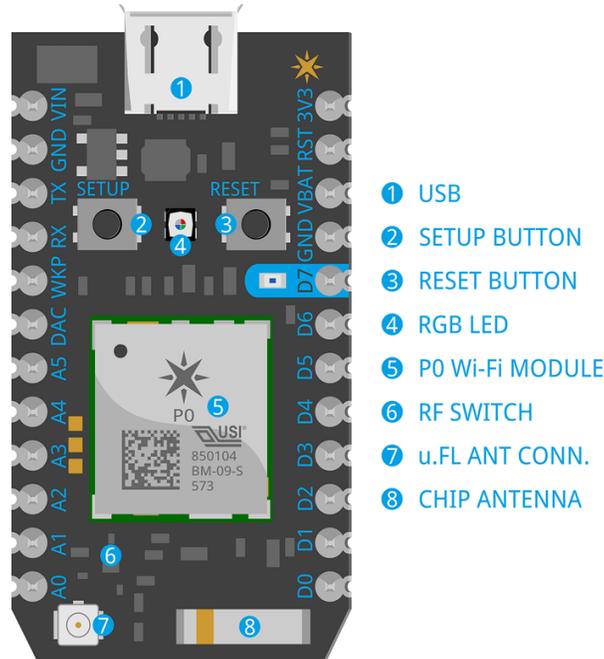


Figura A.6. Marcas de entradas del Particle Photon. Tomada de [91].

1.4.2. Descripción de entradas

Tabla A.2. Descripción de entradas del Particle Photon. Tomada de [91].

Pin	Descripción
VIN	Este pin se puede utilizar como entrada o salida. Como entrada, suministre de 3,6 a 5,5 V CC para alimentar el Photon. Cuando el Photon se alimenta a través del puerto USB, este pin generará un voltaje de aproximadamente 4,8 V CC debido a un diodo Schottky en serie de protección de polaridad inversa entre VUSB y VIN. Cuando se usa como salida, la carga máxima en VIN es 1A.
RST	Entrada de reinicio activo-bajo. El circuito integrado contiene una resistencia pull-up de 1k ohm entre RST y 3V3, y un condensador de 0,1 uF entre RST y GND.

VBAT	Suministro al RTC interno, registros de respaldo y SRAM cuando 3V3 no está presente (1,65 a 3,6 VCC).
3V3	Este pin es la salida del regulador integrado y está conectado internamente al VDD del módulo Wi-Fi. Al alimentar el Photon a través de VIN o el puerto USB, este pin generará un voltaje de 3,3 V CC. Este pin también se puede utilizar para alimentar el Photon directamente (entrada máxima de 3,3 V CC). Cuando se usa como salida, la carga máxima en 3V3 es de 100 mA. NOTA: Cuando encienda el Photon a través de este pin, asegúrese de que la alimentación esté desconectada del VIN y del USB.
RX	Se usa principalmente como UART RX, pero también se puede usar como propósito general (GPIO) digital o PWM.
TX	Se usa principalmente como UART TX, pero también se puede usar como GPIO digital o PWM.
WKP	Pin de activación alto activo, activa el módulo de los modos de suspensión/espera. Cuando no se usa como WAKEUP, este pin también se puede usar como GPIO digital, entrada ADC o PWM. Se puede referir a A7 cuando se usa como ADC.
DAC	Salida digital a analógica (D/A) de 12 bits (0-4095), denominada DAC o DAC1 en el software. También se puede utilizar como GPIO o ADC digital. Se puede denominar A6 cuando se usa como ADC. A3 es una segunda salida DAC utilizada como DAC2 en el software
A0- A7	Entradas de analógico a digital (A/D) de 12 bits (0-4095) y también GPIO digitales. A6 y A7 son asignaciones de conveniencia de código, lo que significa que los pines en realidad no están etiquetados como tales, pero puede usar código como analogRead(A7). A6 se asigna al pin DAC y A7 se asigna al pin WKP. A4,A5,A7 también se pueden utilizar como salida PWM.
D0- D7	Pines GPIO solo digitales. D0~D3 también se puede utilizar como salida PWM.

Notas: Además de los 24 pines alrededor del exterior del Photon, hay 7 bloques en la parte inferior del PCB Photon que se pueden usar para conectar señales adicionales: salidas LED RGB, botón SETUP, línea de habilitación SMPS y USB D+/D-. Los pines del Photon #25-31 se describen en los diagramas de salida de pines. Se recomienda consultar también la sección del Photon de patrón de tierra de PCB recomendado sin encabezados para conocer su ubicación en la parte inferior del fotón. Hay módulos PWM disponibles también en D0, D1, D2, D3, A4, A5, WKP, RX, TX con una advertencia: el periférico del temporizador PWM está duplicado en dos pines (A5/D2) y (A4/D3) para un total de 7 Salidas PWM independientes. Por ejemplo: PWM se puede usar en A5 mientras que D2 se usa como GPIO, o D2 como PWM mientras que A5 se usa como entrada analógica. Sin embargo, A5 y D2 no se pueden usar como salidas PWM controladas de forma independiente al mismo tiempo.

ANEXO 2: DISEÑO Y CONSTRUCCIÓN DE LA INFRAESTRUCTURA

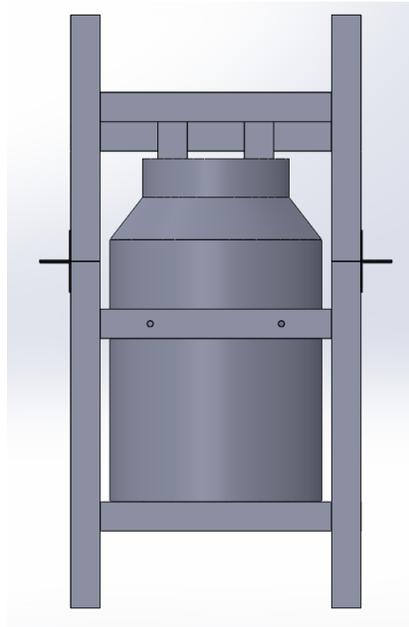


Figura A.7. Vista frontal de la infraestructura del sistema de medición y control de temperatura.

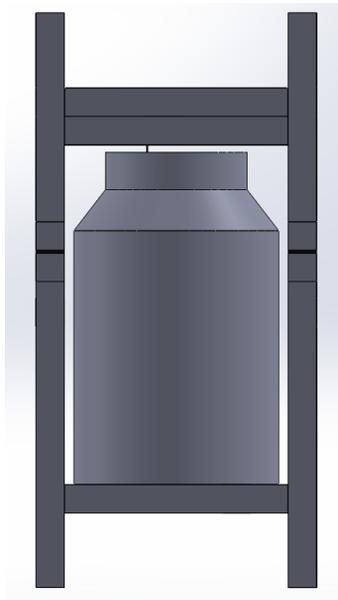


Figura A.8. Vista lateral de la infraestructura del sistema de medición y control de temperatura.

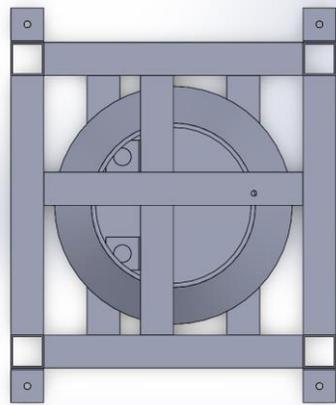


Figura A.9. Vista superior de la infraestructura del sistema de medición y control de temperatura.

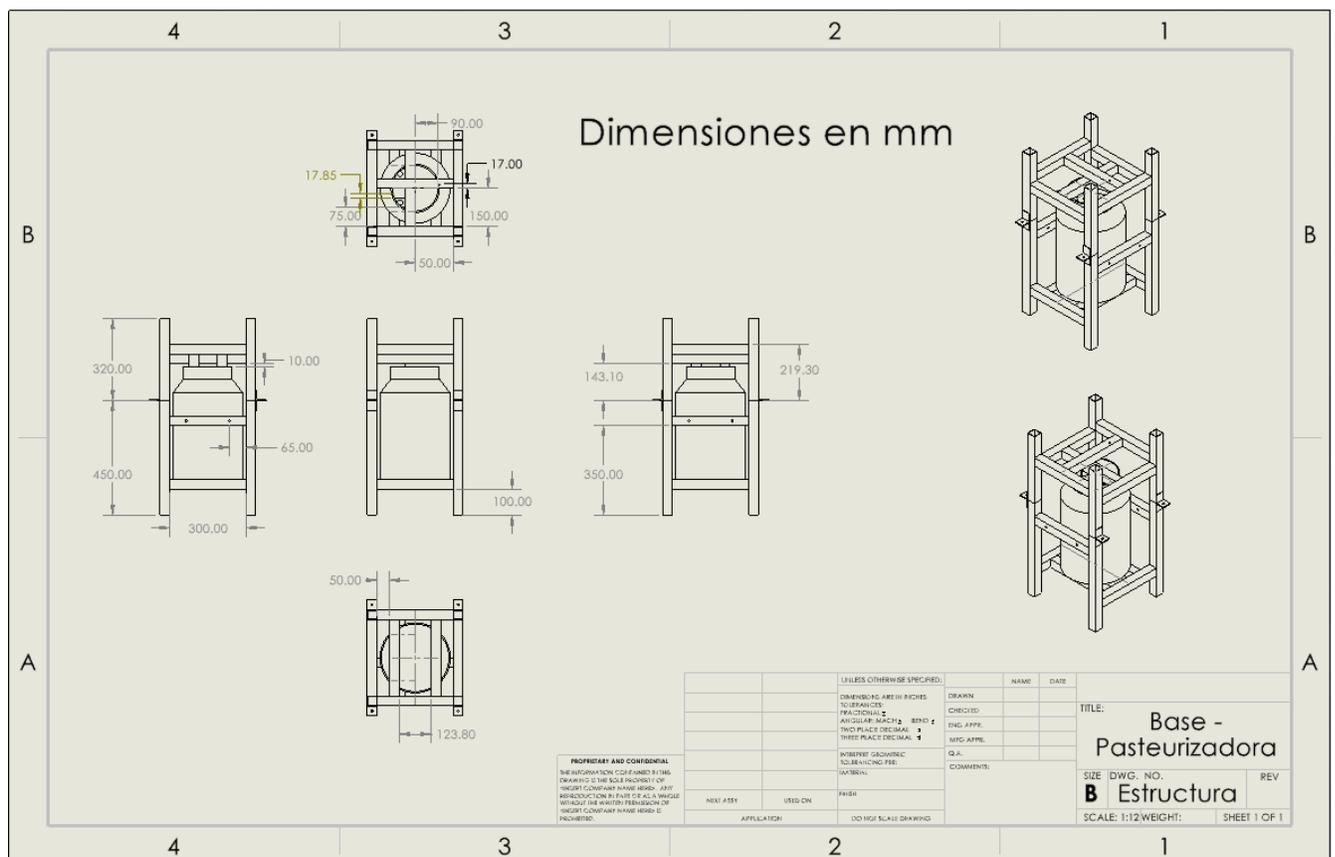


Figura A.10. Dibujo de la infraestructura del sistema de medición y control de temperatura, acotado en milímetros.

ANEXO 3: CÓDIGOS DE PROGRAMACIÓN

```
void createEventPayload(int tempc)
{
    JsonWriterStatic<256> jw;
    {
        JsonWriterAutoObject obj(&jw);
        jw.insertKeyValue("Temperatura", tempc);
    }
    Particle.publish("Temperatura Pt100: ", jw.getBuffer(),
PRIVATE);
}
```

Código A.1. Función empleada en la programación del Photon, la cual crea un objeto tipo JSON para subir a la nube los datos de la variable de interés.

```
{
JsonWriterAutoObject obj(&jw);
jw.insertKeyValue("a", true);
jw.insertKeyValue("b", 1234);
jw.insertKeyValue("c", "test");
}
```

Output: {"a":true,"b":1234,"c":"test"}

Código A.2. Captura del código y salida a modo de ejemplo para el uso de la función *JsonWriterAutoObject*. Tomado de [93].

```
from dash import html
html.Div([
    html.H1('Hello Dash'),
    html.Div([
        html.P('Dash converts Python classes into HTML').
        html.P("This conversion happens behind the scenes by
Dash's JavaScript front-end")
    ])
])
```

Código A.3. Código de muestra de estructura HTML básica utilizando la librería Dash de Plotly. Tomado de [101].

```
<div>
  <h1>Hello Dash<\h1>
  <div>
    <p>Dash converts Python classes into HTML<\p>
    <p>This conversion happens behind the scenes by Dash's
JavaScript front-end<\p>
  <\div>
<\div>
```

Código A.4. Código de muestra de estructura HTML básica, convertido a partir del Código C.3. Tomado de [101].

Para más información acerca del uso de los códigos presentados, así como el código fuente implementado en el Photon y en la aplicación de monitoreo, referirse a: rdebenavente@gmail.com o ranged@fata.unam.mx.

ANEXO 4: INFORMES DE RESULTADOS DE LAS PRUEBAS MICROBIOLÓGICAS



INFORME DE RESULTADOS

LAB-FR-083 Rev.10

LABORATORIO EURO-NUTEC

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO
AV. UNIVERSIDAD 3000 DISTRITO FEDERAL MEXICO

A la atención de: A QUIEN CORRESPONDA

Identificación de la muestra

Producto:	00LECHE	Origen:	NO DEFINIDO
Nombre:	LECHE	Ref Cliente:	SIN PASTEURIZAR
N°Crono:	2311987	Número de Lote:	muestra 1
Recibido el:	03/05/2023	Fecha Elaboración:	
Fecha Muestreo:	03/05/2023	Fecha Caducidad:	
Proveedor:	NO DEFINIDO	Comentario:	ALMACENAMIENTO: REFRIGERACION; EMPAQUE: BOTE DE PLASTICO;630g
Entregado por:	ACID		

Resultados de análisis de la muestra

Análisis	Método	Resultado	Unidad
CUENTA DE BACTERIAS MESOFILICAS AEROBIAS*1 05/05/2023	NOM-092-SSA1-1994	MAYOR A 25000	UFC/ml
CUENTA DE MICROORGANISMOS COLIFORMES TOTALES*2 04/05/2023	NOM-113-SSA1-1994	140.00	UFC/ml
E. COLI 04/05/2023	NOM-210-SSA1-2014 APENDICE J	20 V. E.	UFC/ml
LISTERIA MONOCYTOGENES 05/05/2023	AOAC Official Method 2013.10 Listeria (LPT) Method First action 2013	AUSENCIA EN 25 ml	-
SALMONELLA 04/05/2023	VIDAS AOAC 2013.01	PRESUNTO POSITIVO	-
SALMONELLA VIDAS CONFIRMACIÓN 09/05/2023	Confirmación Salmonella Vidas NOM-210 Apéndice A	PRESENCIA EN 25 g	-

Agradecemos su preferencia

Page.1 / 2

Este informe solo ampara la muestra analizada y no podrá ser reproducido de manera parcial sin previa autorización del Laboratorio EURO NUTEC. El muestreo fue llevado a cabo por parte del cliente. La muestra se resguarda durante 10 días hábiles (Muestras de productos perecederos y aceites una vez analizados se desechan). **El laboratorio recibirá solicitudes de revisión de resultados o reclamos de servicios únicamente en los 10 días hábiles posteriores al envío del informe de resultados en formato electrónico.**

Euro-Nutec Premix, S.A de C.V.
Av. del Marqués No.32, Fracc. Ind. Bernardo Quintana, El Marqués, Qro., Mex. C.P. 76246
Tel.- (442) 196 0100 Fax: (442) 196 0101 www.gponutec.com

Figura A.11. Informe de resultados de los análisis microbiológicos realizados en la leche sin pasteurizar.



INFORME DE RESULTADOS

LAB-FR-083 Rev.10

Atentamente



MICP Lucia Robles
Responsable de Laboratorio

*1 EN AGAR TRIFTONA EXTRACTO DE LEVADURA (AGAR PARA CUENTA ESTANDAR) INCUBADO 48HRS A 35°C
*2 EN AGAR ROJO DE VIOLETA BLIS INCUBADO 24HRS A 35°C
*3 EN AGAR DEXTROSA ACIDIFICADAS INCUBADAS A 25°C ± 1°C (SI NO SE ESPECIFICA EN EL RESULTADO LA INCUBACION FUE DE 24 HRS)
V= VALOR ESTIMADO
* RESULTADOS FUERA DE TOLERANCIA O RESULTADOS SIN TOLERANCIA (µg/Kg) = 250
CAMBIO= SE CONSIDERA UN PRODUCTO NO APTO PARA SU CONSUMO
MENOR A 10 UPG/g o mL = NO SE DETECTO CRECIMIENTO DE MICROORGANISMOS
MENOR A 3 NMP/g o mL = NO SE OBSERVÓ FORMACION DE GAS EN LOS TUBOS
MENOR A 1.1 NMP/mL = NO SE OBSERVÓ FORMACION DE GAS EN LOS TUBOS
% = g/100g

Agradecemos su preferencia

Page 2 / 2

Este informe solo ampara la muestra analizada y no podrá ser reproducido de manera parcial sin previa autorización del Laboratorio EURO NUTEC. El muestreo fue llevado a cabo por parte del cliente. La muestra se resguarda durante 10 días hábiles (Muestras de productos perecederos y aceites una vez analizados se desechan). **El laboratorio recibirá solicitudes de revisión de resultados o reclamos de servicios únicamente en los 10 días hábiles posteriores al envío del informe de resultados en formato electrónico.**

Euro-Nutec Premix, S.A de C.V.
Av. del Marqués No.32, Fracc. Ind. Bernardo Quintana, El Marqués, Qro., Mex. C.P. 76246
Tel.- (442) 196 0100 Fax: (442) 196 0101 www.gponutec.com

Figura A.11.2. Informe de resultados de los análisis microbiológicos realizados en la leche sin pasteurizar.



INFORME DE RESULTADOS

LAB-FR-083 Rev.10

LABORATORIO EURO-NUTEC

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO
AV. UNIVERSIDAD 3000 DISTRITO FEDERAL MEXICO

A la atención de: A QUIEN CORRESPONDA

Identificación de la muestra

Producto:	00LECHE	Origen:	NO DEFINIDO
Nombre:	LECHE	Ref Cliente:	PASTEURIZADA
N°Crono:	2311986	Número de Lote:	muestra 2
Recibido el:	03/05/2023	Fecha Elaboración:	
Fecha Muestreo:	03/05/2023	Fecha Caducidad:	
Proveedor:	NO DEFINIDO	Comentario:	ALMACENAMIENTO: REFRIGERACION; EMPAQUE: BOTE DE PLASTICO;612g
Entregado por:	ACID		

Resultados de análisis de la muestra

Análisis	Método	Resultado	Unidad
CUENTA DE BACTERIAS MESOFILICAS AEROBIAS*1 05/05/2023	NOM-092-SSA1-1994	280.00	UFC/ml
CUENTA DE MICROORGANISMOS COLIFORMES TOTALES*2 04/05/2023	NOM-113-SSA1-1994	3 V. E.	UFC/ml
E. COLI 04/05/2023	NOM-210-SSA1-2014 APENDICE J	10 V. E.	UFC/ml
LISTERIA MONOCYTOGENES 05/05/2023	AOAC Official Method 2013.10 Listeria (LPT) Method First action 2013	AUSENCIA EN 25 ml	-
SALMONELLA 04/05/2023	VIDAS AOAC 2013.01	AUSENCIA EN 25 ml	-

Atentamente



MICP Lucia Robles
Responsable de Laboratorio

*1 EN ASAR TRIPTONA EXTRACTO DE LEVADURA (ASAR PARA CUENTA ESTANDAR) INCUBADO 48HRD A 35°C
*2 EN ASAR ROJO DE VIOLETA BILIS INCUBADO 24HRD A 35°C
*3 EN ASAR DEXTROSA ACIDIFICADAS INCUBADAS A 35°C± 1°C (SI NO SE ESPECIFICA EN EL RESULTADO LA INCUBACION FUE DE 5 DIAS)
V= VALOR ESTIMADO
* RESULTADOS FUERA DE TOLERANCIA O RESULTADOS SIN TOLERANCIA (µg/Kg < 1000)
CAMBIO= SE CONSIDERA UN PRODUCTO NO APTO PARA SU CONSUMO
MENOR A 10 UFC/g o mL = NO SE DETECTO CRECIMIENTO DE MICROORGANISMOS
MENOR A 3 NMP/g o mL = NO SE OBSERVO FORMACION DE GAS EN LOS TUBOS
MENOR A 1.1 NMP/mL = NO SE OBSERVO FORMACION DE GAS EN LOS TUBOS
N=9/1000

Agradecemos su preferencia

Page 1 / 1

Este informe solo ampara la muestra analizada y no podrá ser reproducido de manera parcial sin previa autorización del Laboratorio EURO NUTEC. El muestreo fue llevado a cabo por parte del cliente. La muestra se resguarda durante 10 días hábiles (Muestras de productos perecederos y aceites una vez analizados se desechan). **El laboratorio recibirá solicitudes de revisión de resultados o reclamos de servicios únicamente en los 10 días hábiles posteriores al envío del informe de resultados en formato electrónico.**

Euro-Nutec Premix, S.A de C.V.
Av. del Marqués No.32, Fracc. Ind. Bernardo Quintana, El Marqués, Gro., Mex. C.P. 76246
Tel.- (442) 196 0100 Fax: (442) 196 0101 www.gponutec.com

Figura A.12. Informe de resultados de los análisis microbiológicos realizados en la leche pasteurizada.