



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE INGENIERÍA ELECTRÓNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE PLC
USANDO LATTEPANDA Y OPENPLC”**

Trabajo de titulación previo a la obtención del
Título de Ingeniero Electrónico

AUTORES:

**GEORGE RODRIGO ORDOÑEZ LARA
DANIEL STEVEN YÉPEZ MÉNDEZ**

TUTOR: ING. VÍCTOR DAVID LARCO TORRES MSC.

GUAYAQUIL – ECUADOR

2023

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA

Nosotros, **GEORGE RODRIGO ORDOÑEZ LARA** con documento de identificación N° 0940169196 con documento de identificación y **DANIEL STEVEN YÉPEZ MÉNDEZ** con documento de identificación N° 0927191114; manifestó que:

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

Guayaquil, 25 de marzo del 2024

Atentamente,



GEORGE RODRIGO ORDOÑEZ LARA

0940169196



DANIEL STEVEN YÉPEZ MÉNDEZ

0927191114

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR

Nosotros, **GEORGE RODRIGO ORDOÑEZ LARA** con documento de identificación N° 0940169196 y **DANIEL STEVEN YÉPEZ MÉNDEZ** con documento de identificación N° 0927191114;, expresamos voluntad y por medio del presente documento de ceder a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de grado titulado: **“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE PLC USANDO LATTEPANDA Y OPENPLC”**, el cual ha sido desarrollado para optar por el título de: **INGENIERO ELECTRÓNICO**, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 25 de marzo del 2024

Atentamente,



GEORGE RODRIGO ORDOÑEZ LARA

0940169196



DANIEL STEVEN YÉPEZ MÉNDEZ


0927191114

CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN

Yo, **Víctor David Larco Torres**, con documento de identificación N° , docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: “**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE PLC USANDO LATTEPANDA Y OPENPLC**”, realizado por , **GEORGE RODRIGO ORDOÑEZ LARA** con documento de identificación N° 0940169196 y **DANIEL STEVEN YÉPEZ MÉNDEZ** con documento de identificación N° 0927191114, obteniendo como resultado final el trabajo de titulación bajo la opción de trabajo de grado que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 25 de Marzo del 2024

Atentamente,

A handwritten signature in blue ink, appearing to read 'Victor Larco Torres', is written over a horizontal dashed line.

Ing. Víctor David Larco Torres MSc.

0912419611

DEDICATORIA

Este proyecto técnico se la dedico a Dios quien supo guiarme por el buen camino, por darme la fuerza para seguir adelante con mis estudios.

A mi familia por brindarme el apoyo porque gracias a ellos soy lo que soy, a mi Madre porque a pesar de las dificultades supo sacarme adelante y así pude culminar con mis estudios, a mis Tíos que me aconsejaron y me apoyaron económicamente desde el inicio de mi carrera, a mi Hija Eva Georgina Ordoñez ya que ella fue la inspiración para seguir adelante con mis estudios.

DEDICATORIA

Este proyecto técnico va con dedicatoria a Dios quien me guio y me dio la suficiente fuerza de voluntad para no desestimar la carrera y poder culminarla con éxito pese a todos los obstáculos que se presentan con normalidad en el proceso, esto también va dedicado a los docentes que forzaron mi aprendizaje y mi camino profesional, por instruirme con conocimiento, y a mis padres por haber aportado en lo económico y social, ya que ellos han sido y serán parte esencial en mi ser y formación.

AGRADECIMIENTO

Agradezco primeramente a Dios por darme la oportunidad de haber culminado este proyecto y por darme la fuerza para superar los obstáculos que en el transcurso de este proyecto se fueron dando.

Deseo agradecer a todas las personas que de una forma u otra permitieron la realización de este proyecto, en especial a mi hija Eva Georgina Ordoñez y mi Esposa Solange Dayanna Regalado por el amor y la paciencia en los momentos de arduo trabajo.

A mis Padres y Hermanos por el amor, comprensión y paciencia.

A mis Abuelitos y Tíos que siempre me han dado el apoyo sin condición alguna.

A mi Madrina Elena Erazo que me brindo un espacio en su hogar cuando más lo necesitaba y me trato como un hijo.

AGRADECIMIENTO

Agradezco primeramente a Dios por haber culminado mi proyecto y de cierta manera también agradezco a todos mis Docentes que me despejaron de cualquier duda en todo el transcurso de la carrera.

Agradezco a mis Familiares que me brindaron el apoyo de cualquier manera también por todo el amor y comprensión en todo el transcurso de mi proyecto.

RESUMEN

AÑO	ALUMNOS	DIRECTOR DE PROYECTO TÉCNICO	TEMA DE PROYECTO TÉCNICO
2024	<ul style="list-style-type: none">• George Rodrigo Ordoñez Lara• Daniel Steven Yépez Méndez	Ing. Víctor David Larco Torres, MSc.	Diseño e Implementación de un Prototipo de PLC Usando LattePanda y OpenPLC

El presente trabajo detalla el desarrollo y diseño de un prototipo de controlador lógico programable (PLC) utilizando LattePanda y OpenPLC. Este proyecto tiene como objetivo proporcionar una solución económica y eficiente para la automatización industrial. Utilizando la versatilidad de LattePanda, un ordenador de placa única de alto rendimiento, y la flexibilidad de OpenPLC, una plataforma de software libre para automatización, se crea un sistema robusto y adaptable para diferentes aplicaciones industriales.

Este prototipo demuestra ser una solución viable y rentable para la automatización industrial. La combinación de LattePanda y OpenPLC abre nuevas posibilidades para el diseño de sistemas de control personalizados, accesibles y eficientes. Este trabajo no solo aporta a la comunidad industrial, sino que también sirve como una herramienta educativa valiosa para estudiantes e ingenieros interesados en el campo de la automatización.

PALABRAS CLAVES: OpenPLC, Automatización Industrial, Controlador Lógico Programable, Software Libre, Programación de PLC, Integración de Sensores y Actuadores, Ladder Logic.

ABSTRACT

YEAR	STUDENTS	DIRECTOR OF TECHNICAL PROJECT	TECHNICAL PROJECT THEME
2024	<ul style="list-style-type: none">• George Rodrigo Ordoñez Lara• Daniel Steven Yépez Méndez	Ing. Víctor David Larco Torres, MSc.	Diseño e Implementación de un Prototipo de PLC Usando LattePanda y OpenPLC

This paper details the development and design of a prototype programmable logic controller (PLC) using LattePanda and OpenPLC. This project aims to provide an economical and efficient solution for industrial automation. Utilizing the versatility of LattePanda, a high-performance single-board computer, and the flexibility of OpenPLC, an open-source software platform for automation, a robust and adaptable system is created for various industrial applications.

This prototype proves to be a viable and cost-effective solution for industrial automation. The combination of LattePanda and OpenPLC opens new possibilities for the design of personalized, accessible, and efficient control systems. This work not only contributes to the industrial community but also serves as a valuable educational tool for students and engineers interested in the field of automation.

KEYWORDS: OpenPLC, Industrial Automation, Programmable Logic Controller, Open-Source Software, PLC Programming, Sensor and Actuator Integration, Ladder Logic.

ÍNDICE GENERAL

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA ...	¡Error! Marcador no definido.
CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR	¡Error! Marcador no definido.
CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN.....	II
DEDICATORIA	IV
DEDICATORIA	V
AGRADECIMIENTO	VI
AGRADECIMIENTO	VII
RESUMEN.....	VIII
ABSTRACT	IX
ÍNDICE GENERAL.....	X
ÍNDICE DE FIGURAS	XIII
ÍNDICE DE TABLAS.....	XV
INTRODUCCIÓN	1
1. EL PROBLEMA	2
1.1. Antecedentes.....	2
1.2. Importancia y Alcances.....	2
1.3. Delimitación	3
1.3.1. Temporal	3
1.3.2. Espacial.....	3
1.3.3. Académica	3
1.4. Innovación.....	3
1.5. Objetivos.....	4
1.5.1. Objetivo general.....	4
1.5.2. Objetivos específicos	4
2. FUNDAMENTOS TEÓRICOS.....	5
2.1. Código libre	5
2.2. Teoría de Control Automático	5
2.3. Fundamentos de los PLCs	6
2.4. Arquitectura y Funcionamiento de LattePanda	8
2.4.1. Arquitectura de Hardware.....	8
2.4.2. Capacidades de Entrada/Salida (E/S).....	9
2.4.3. Compatibilidad de Software	10
2.4.4. Plataforma de Desarrollo Integrado	11

2.4.5. Aplicaciones en Robótica y Automatización.....	11
2.4.6. Arduino Co-Processor.....	11
2.5. Entradas digitales	11
2.6. Salidas digitales	12
2.7. Entradas análogas.....	12
2.8. Salidas análogas.....	13
2.9. Protocolos de comunicación	13
2.9.1. UART	14
2.9.2. SPI.....	14
2.9.3. CAN.....	15
2.9.4. I2C.....	16
2.10. Software de Código Abierto y OpenPLC.....	17
2.11. Aplicaciones Industriales de PLCs.....	18
2.11.1. Control de maquinaria	19
2.11.2. Automatización de procesos:	19
2.11.3. Control de sistemas HVAC	19
2.11.4. Automatización de edificios	19
2.11.5. Control de bombas y válvulas	19
2.11.6. Control de procesos en la industria alimentaria	19
2.11.7. Automatización en la industria automotriz	20
2.11.8. Control de hornos industriales	20
2.11.9. Gestión de sistemas de energía	20
2.11.10. Control de sistemas de logística y transporte.....	20
2.12. Aspectos de Seguridad en PLCs y Sistemas Automatizados.....	20
2.13. Beneficios entre OpenPLC y plc.....	21
3. MARCO METODOLÓGICO	22
3.1. Diseño del prototipo.....	22
3.2. Funcionamiento del prototipo.....	22
3.3. Componentes del prototipo	23
3.3.1. Caja de control	23
3.3.2. Banda de transporte.....	26
3.4. OPENPLC	26
3.5. Programación en OPENPLC.....	27
3.5.1. Lenguajes de programación en OPENPLC.....	27
3.6. Declaración de variables en OPENPLC en LattePanda.....	30
3.7. Programación OPENPLC en LattePanda	34
3.8. Esquema de tarjetas PCB	35

3.9. Tarjeta de distribución.....	35
3.10. Esquema de potencia para LattePanda.....	36
4. MANUAL DE PRÁCTICA.....	37
Práctica: Control de una banda transportadora y conteo de productos	37
5. RESULTADOS	38
CRONOGRAMA.....	44
PRESUPUESTO	45
CONCLUSIONES.....	46
RECOMENDACIONES	47
REFERENCIAS BIBLIOGRÁFICAS.....	48
Anexo A.- PARTES DE BANDA DE TRANSPORTE.....	¡Error! Marcador no definido.
Anexo B.- Resolución	55

ÍNDICE DE FIGURAS

Figura 1 : Fundamentos de los PLCs (J. Torres Ventura, 2023).....	6
Figura 2 : Arquitectura y Funcionamiento de LattePanda (Briones García, 2023)....	8
Figura 3 : Arquitectura de Hardware (Rahmad, 2020).....	9
Figura 4 : Capacidades de Entrada/Salida (E/S) (Montenegro Garófalo, 2021).....	10
Figura 5 : Compatibilidad de Software (Gómez, 2022).....	10
Figura 6 : Entrada digital (Fonseca-Beltrán, 2021).....	12
Figura 7 : Salida digital (Molina Araujo, 2018).	12
Figura 8 : Diseño del prototipo (Romero-Ortells Labrada, 2023).	13
Figura 9 : Salida analógica (Huera Lomas, 2023).	13
Figura 10 : UART (Rico, 2021).	14
Figura 11 : SPI.....	15
Figura 12 : CAN (Martinez, 2023).....	16
Figura 13 : I2C (Quispe, 2019).	17
Figura 14 : OPEN PLC.....	18
Figura 15 : Diseño del prototipo.....	22
Figura 16 : Funcionamiento del proceso de la banda.....	23
Figura 17 : Conexiones de tarjeta de distribución en LattePanda.....	24
Figura 18 : Conexiones de tarjeta de distribución en LattePanda.....	24
Figura 19 : Conexiones de LattePanda y controlador de motor.....	25
Figura 20 : Conexiones de LattePanda y display de 7 segmentos.....	25
Figura 21 : Conexiones de LattePanda y pantalla 7 pulgadas.....	25
Figura 22 : Conexiones de LattePanda y pantalla 7 pulgadas.....	26
Figura 23 : OPENPLC.....	27
Figura 24 : <i>Function Block Diagram (FBD)</i>	28
Figura 25 : <i>Ladder Logic(LD)</i>	28
Figura 26 : Structured Text (ST).....	29
Figura 27 : Instruction List (IL).....	30
Figura 28 : Sequential Function Chart.....	30
Figura 29 : Creación de proyecto y selección de lenguaje en OPENPLC.....	31
Figura 30 : Secciones de OPENPLC.....	31
Figura 31 : Declaración de variables en OPENPLC para LattePanda.....	33
Figura 32 : Programación en OPENPLC para LattePanda.....	34
Figura 33 : Configuración de puertos en OPENPLC para LattePanda.....	34
Figura 34 : Esquema de tarjeta.....	35
Figura 35 : Diseño de tarjeta.....	35
Figura 36 : Esquema de la potencia.....	36
Figura 37 : Prototipo final.....	38
Figura 38 : Detección de producto en banda transportadora.....	39
Figura 39 : Programa de detección de sensor y conteo de productos.....	39
Figura 40 : Programa de activación y paro del sistema.....	40
Figura 41 : Activación de marcha y activación de paro del sistema.....	40
Figura 42 : Control del sentido del motor.....	40
Figura 43 : Control de la velocidad en Ladder en OPENPLC.....	41
Figura 44 : Control del display en Ladder en OPENPLC.....	41
Figura 45 : Accionamiento de código binario display 0-1.....	42
Figura 46 : Accionamiento de código binario display 2-3.....	42
Figura 47 : Accionamiento de código binario display 4-5.....	42
Figura 48 : Accionamiento de código binario display 6-7.....	43

Figura 49 : Accionamiento de código binario display 8- reinicio.....	43
Figura 50 : Botón de encendido.....	56
Figura 51 : Ejecución de programa OPENPLC	57
Figura 52 : Ejecución de programa OPENPLC	57
Figura 53 : Selección de lenguaje de programación en OPENPLC	58
Figura 54 : Configuración de pines en el módulo.	59
Figura 55 : Configuración de pines en el OPENPLC	59
Figura 56 : Programa de detección de sensor y conteo de productos	59
Figura 57 : Programa de activación y paro del sistema.....	60
Figura 58 : Control del sentido del motor	60
Figura 59 : Control de la velocidad en Ladder en OPENPLC.....	60
Figura 60 : Control del display en Ladder en OPENPLC.....	61
Figura 61 : Compilación del programa.....	61

ÍNDICE DE TABLAS

Tabla 1: Tabla de direccionamiento en OPENPLC.....	32
Tabla 2: Tabla de direccionamiento del proyecto en OPENPLC	33
Tabla 3: Tabla de cronograma de actividades.....	45
Tabla 4: Tabla de presupuesto del proyecto técnico	586
Tabla 5: Tabla de direccionamiento del proyecto en OPENPLC	581

INTRODUCCIÓN

El proyecto está basado en la implementación y diseño de un prototipo para la automatización industrial juega un papel crucial en la mejora de la eficiencia y la productividad de los procesos, dichos controladores lógicos programables (PLC) son componentes fundamentales en este ámbito.

A pesar de la importancia de los PLCs en la industria, su alto costo y complejidad de programación pueden ser barreras para pequeñas y medianas empresas o para fines educativos. En este contexto, el uso de plataformas de hardware y software abiertos, como LattePanda y OpenPLC, emerge como una alternativa viable y económica.

Este estudio se enfoca en el diseño y la implementación de un prototipo de PLC utilizando LattePanda, una computadora de placa única, y OpenPLC, un proyecto de software libre para automatización industrial, teniendo como objetivo un sistema de control económico, accesible y flexible, adecuado para aplicaciones educativas y pequeñas aplicaciones industriales.

El proyecto está enfocado en los estudiantes de la carrera de Ingeniería en Automatización Industrial de la Universidad Politécnica Salesiana Sede Guayaquil, por lo cual contribuye al campo de la automatización industrial al proporcionar una solución de bajo costo para el desarrollo de habilidades prácticas en la programación de PLCs, así como para la implementación de sistemas de control en pequeña escala. Además, demuestra la viabilidad de utilizar plataformas abiertas en aplicaciones industriales, lo que puede fomentar la innovación y la adaptabilidad en entornos de enseñanza y producción.

El documento está organizado en las siguientes secciones: revisión de literatura, descripción del sistema, metodología, resultados y discusión, conclusiones y recomendaciones para futuras investigaciones.

1. EL PROBLEMA

1.1. Antecedentes

El problema del constante crecimiento tecnológico es el aprendizaje en diferentes áreas de la ingeniería debido a que cada empresa crea sus programas con licencias con firmas digitales la cuales tienen un costo elevado para el usuario o estudiante de ingeniería Electrónica.

El inconveniente que tienen los estudiantes al momento de utilizar programas con licencias es el alto costo de estas para usar equipos enfocados en ingeniería electrónica fuera de los laboratorios de la Universidad Politécnica salesiana.

Teniendo cuenta esta problemática se plantea el uso de programas con licencias libres o OPENSOURCE en el área de autómatas programables mediante la simulación de un PLC mediante un sistema embebido utilizando la plataforma OPENPLC.

1.2. Importancia y Alcances

El uso de programas con licencias comerciales en ingeniería electrónica representa una barrera económica significativa para estudiantes y profesionales que no disponen de los recursos financieros para adquirir estas herramientas, donde los estudiantes necesitan acceder a estas herramientas para su formación, pero se encuentran con el obstáculo de los altos costos.

La dependencia de las instituciones educativas en programas de licencia limita las oportunidades de los estudiantes para practicar y experimentar fuera del entorno del laboratorio universitario. Esto puede restringir el aprendizaje práctico y la experimentación independiente, aspectos cruciales en campos como la ingeniería electrónica.

La facilitación del acceso a herramientas tecnológicas a través del software de código abierto puede acelerar la innovación y el desarrollo tecnológico. Al tener más personas accediendo y colaborando en estas plataformas, se fomenta un ambiente de innovación más rico y diverso.

1.3. Delimitación

1.3.1. Temporal

Se considera un periodo de seis meses distribuidos entre la implementación del modelo y desarrollo, pruebas de funcionamiento a fin de brindar un producto terminado.

1.3.2. Espacial

Se enfoca hacia el estudiante a fin de brindar un medio en el cual puedan trabajar sin la necesidad de usar licencias a fin de brindar la mejor experiencia.

1.3.3. Académica

El proyecto contara con lo solicitado por la Universidad Politécnica Salesiana referente a los grados de investigación y desarrollo conjunto a los sustentos requeridos e inmersión de las asignaturas impartidas a lo largo de la trayectoria de grado.

1.4. Innovación

La innovación derivada de la situación descrita radica en la adopción y promoción del software libre y de código abierto en la educación de ingeniería electrónica, particularmente en el aprendizaje práctico de la programación de autómatas y la simulación de PLCs. Esta innovación no solo aborda la barrera económica impuesta por las licencias de software costosas, democratizando así el acceso a herramientas educativas avanzadas, sino que también

fomenta un enfoque colaborativo y participativo en la educación técnica. Al integrar plataformas como OPENPLC en la enseñanza, se estimula un entorno donde los estudiantes no solo aprenden a usar estas tecnologías, sino que también se involucran en la mejora y desarrollo de estas, nutriendo así un ecosistema de innovación y aprendizaje continuo. Este cambio hacia un modelo educativo más inclusivo y colaborativo tiene el potencial de transformar radicalmente la formación en ingeniería electrónica, haciéndola más accesible, interactiva y alineada con las tendencias actuales de la tecnología abierta y compartida.

1.5. Objetivos

1.5.1. Objetivo general

- Diseñar un prototipo con LATTEPANDA y OPENPLC

1.5.2. Objetivos específicos

- Diseñar los esquemas electrónicos para el módulo en PROTEUS.
- Diseñar el esquema de potencia del prototipo basado en PLC con LATTEPANDA.
- Armar el prototipo.
- Diseñar una práctica de un proceso industrial mediante un sistema embebido LATTEPANDA y OpenPLC.

2. FUNDAMENTOS TEÓRICOS

2.1. Código libre

El código libre, un concepto central en la informática y desarrollo de software, se refiere a programas cuyo código fuente está disponible para ser usado, modificado y distribuido por cualquier usuario (Echavarría-Flórez, 2020).

Esta característica fomenta la colaboración y el intercambio de conocimientos, permitiendo a desarrolladores de todo el mundo contribuir y mejorar el software, se rigen por licencias específicas que garantizan estas libertades, como la General Public License (PANTOJA, 2013).

Este enfoque contrasta con el software propietario, donde el código fuente es privado y su modificación o redistribución está restringida. El código libre es fundamental en la promoción de la innovación tecnológica, la seguridad informática mediante la revisión comunitaria y la educación, al permitir el acceso a herramientas de desarrollo sin restricciones económicas (Suarez, 2023).

2.2. Teoría de Control Automático

Es un campo fundamental de la ingeniería y la matemática aplicada que se enfoca en el comportamiento de sistemas dinámicos y la manera de manipularlos para obtener un comportamiento deseado (Aquieta Tibán, 2023).

Utilizando modelos matemáticos para representar sistemas físicos, esta teoría aborda cómo los sistemas pueden ser controlados por medio de realimentación y otros métodos de control. Incluye el análisis de la estabilidad, respuesta en frecuencia, y diseño de controladores (Martínez Rodríguez, 2023).

Los controladores PID (Proporcional, Integral, Derivativo) son un ejemplo clásico, donde la salida del sistema se ajusta en función de la diferencia entre la salida deseada y la medida (RIERA ABELLÁN, 2023).

Esta teoría es aplicable en diversos campos, desde la ingeniería electrónica y mecánica hasta la economía y la biología, siendo crucial en el diseño y operación de sistemas como los PLCs (Controladores Lógicos Programables), donde se busca automatizar procesos industriales de manera eficiente y precisa (Andino, 2023).

2.3. Fundamentos de los PLCs

Los fundamentos de los PLCs (Controladores Lógicos Programables) abarcan conceptos clave y principios que definen su funcionamiento y aplicabilidad en la automatización industrial (J. Torres Ventura, 2023). Como se muestra en la Figura 1.

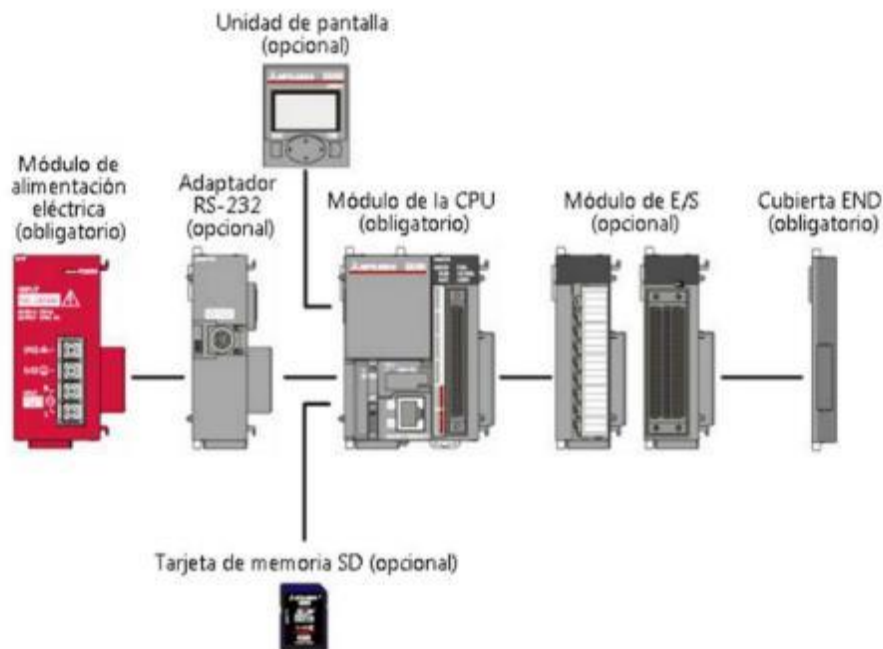


Figura 1 : Fundamentos de los PLCs (J. Torres Ventura, 2023).

2.3.1.1. Function Block Diagram (FBD)

Definición y Propósito: Un PLC es un dispositivo electrónico digital utilizado para automatizar procesos industriales, como controlar maquinaria o líneas de producción. Está diseñado para operar en entornos industriales robustos y realizar tareas de control basadas en programas personalizados (Gómez Borges, 2020).

2.3.1.2. Arquitectura y Componentes:

Los PLCs constan de una unidad central de procesamiento (CPU), memoria, entradas/salidas (E/S) y módulos de interfaz. La CPU ejecuta instrucciones del programa almacenado en la memoria, controlando así las operaciones de la maquinaria mediante señales de E/S (Martínez, 2015).

2.3.1.3. Programación:

Los PLCs se programan utilizando varios lenguajes estandarizados, como Ladder Logic, Structured Text, y Function Block Diagram. Estos lenguajes permiten a los ingenieros y técnicos desarrollar programas que definen cómo el PLC responderá a las entradas y controlará las salidas (Villalobos Toro, 2023).

2.3.1.4. Principios de Operación:

Funcionan leyendo las señales de sus entradas (sensores, interruptores), procesando estos datos según el programa, y activando las salidas correspondientes (actuadores, válvulas) para controlar un proceso o máquina (Saldaña Calderon, 2023).

2.3.1.5. Interfaces de Comunicación

Los PLCs modernos incluyen capacidades de comunicación para interactuar con otros sistemas de control, como redes industriales y sistemas SCADA (Supervisory Control and Data Acquisition), permitiendo un control y monitoreo centralizados (Poma Quispe, 2023).

2.4. Arquitectura y Funcionamiento de LattePanda

La arquitectura y funcionamiento del LattePanda que se muestra en la Figura 2, un computador de una sola placa, son aspectos clave que lo diferencian en el campo de la microinformática y la automatización (Briones García, 2023).

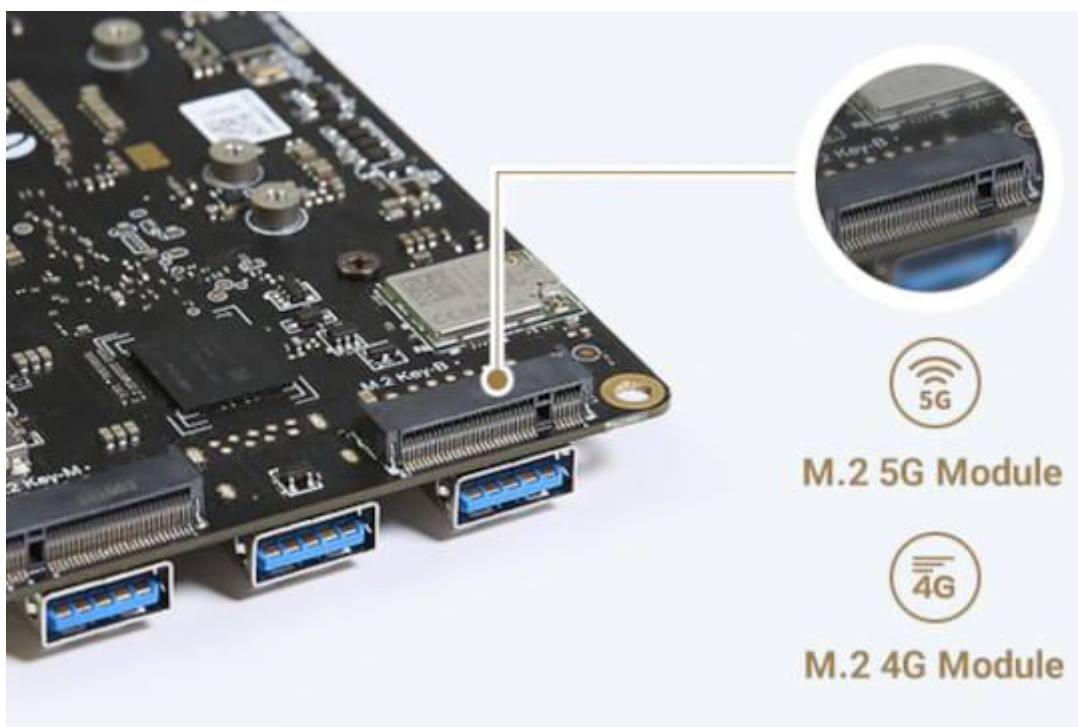


Figura 2 : Arquitectura y Funcionamiento de LattePanda (Briones García, 2023).

2.4.1. Arquitectura de Hardware

LattePanda está equipado con un procesador Intel, usualmente Atom o Celeron, que proporciona una capacidad de cómputo significativa para tareas de procesamiento de datos y control. Incorpora RAM integrada, normalmente

en variantes de 2 GB o 4 GB, y almacenamiento interno eMMC, que varía en capacidad (Rahmad, 2020). Como se muestra en la Figura 3.

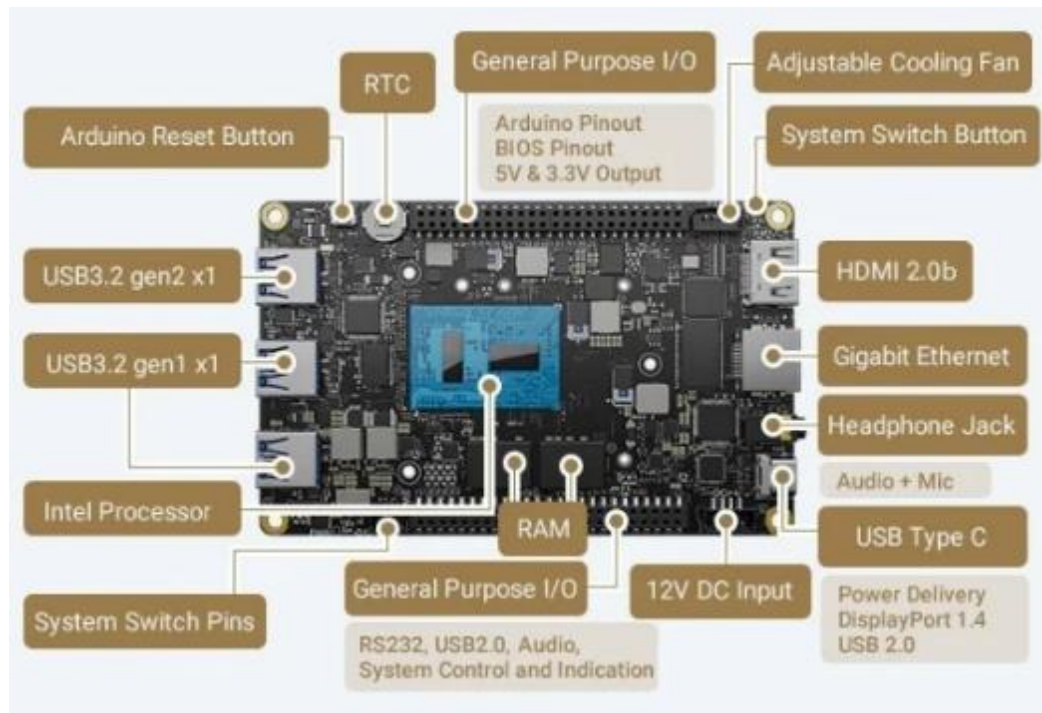


Figura 2 : Arquitectura de Hardware (Rahmad, 2020).

2.4.2. Capacidades de Entrada/Salida (E/S)

Como se muestra en la Figura 4, posee múltiples puertos USB, HDMI para salida de video, y conexiones de red como Ethernet y Wi-Fi. Además, incluye un conjunto de pines GPIO (General Purpose Input/Output) que permite la interacción con una variedad de sensores y actuadores, similar a otros dispositivos de microcontrolador (Montenegro Garófalo, 2021).

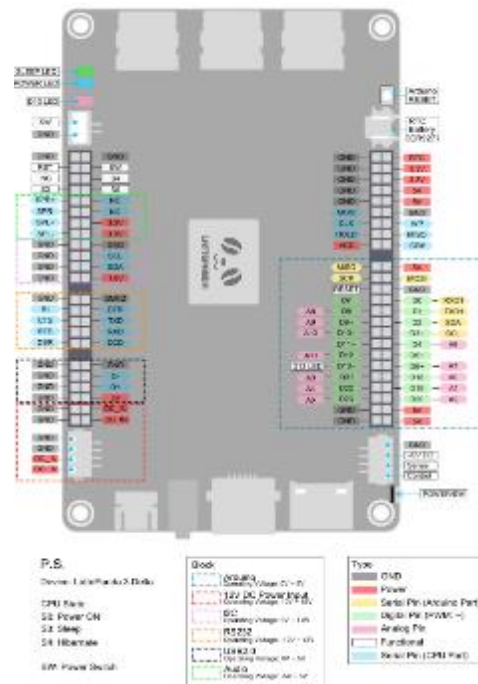


Figura 3 : Capacidades de Entrada/Salida (E/S) (Montenegro Garófalo, 2021).

2.4.3. Compatibilidad de Software

Como se muestra en la Figura 5, una de las características distintivas del LattePanda es su compatibilidad con Windows 10, además de soportar sistemas operativos basados en Linux. Esto lo hace atractivo para aplicaciones que requieren software específico de Windows o la robustez y flexibilidad de Linux (Gómez, 2022).



Figura 4 : Compatibilidad de Software (Gómez, 2022).

2.4.4. Plataforma de Desarrollo Integrado

Ofrece una plataforma integrada para el desarrollo de proyectos de computación y automatización, combinando la potencia de un computador completo con la versatilidad de las tarjetas de desarrollo (Borjas Garcia, 2023).

2.4.5. Aplicaciones en Robótica y Automatización

Su capacidad para ejecutar sistemas operativos completos y su amplia gama de E/S lo hacen ideal para aplicaciones complejas en robótica, automatización, y proyectos de IoT (MARTEL, 2021).

2.4.6. Arduino Co-Processor

Algunas versiones del LattePanda incluyen un coprocesador Arduino, lo que permite una fácil integración con el ecosistema Arduino para control y sensores, ampliando su aplicabilidad en proyectos de automatización y control (Acosta Contreras, 2023).

2.5. Entradas digitales

Las entradas digitales en sistemas de control como los PLCs (Controladores Lógicos Programables) son fundamentales para interpretar señales de dispositivos externos (Fonseca-Beltrán, 2021).

En la figura 6 se detallan estas entradas procesan señales binarias, reconociendo únicamente dos estados: alto (1) o bajo (0), que corresponden generalmente a la presencia o ausencia de voltaje. Son utilizadas para leer el estado de sensores, interruptores, y otros dispositivos digitales (Fonseca-Beltrán, 2021)..

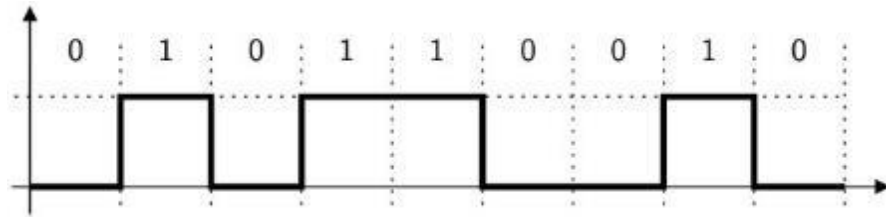


Figura 5 : Entrada digital (Fonseca-Beltrán, 2021).

2.6. Salidas digitales

Las salidas digitales en sistemas de control y automatización, como los PLCs (Controladores Lógicos Programables), son mecanismos que actúan como interfaz entre el controlador y los dispositivos externos (Molina Araujo, 2018).

En la figura 7 se detallan Estas salidas operan en dos estados: alto (ON) o bajo (OFF), correspondiendo a señales binarias de 1 y 0 respectivamente. Se utilizan para activar o desactivar dispositivos como relés, luces indicadoras, solenoides o motores (Molina Araujo, 2018)..

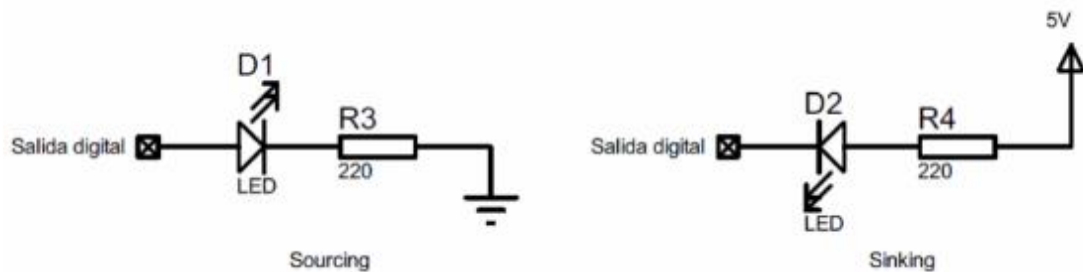


Figura 6 : Salida digital (Molina Araujo, 2018).

2.7. Entradas análogas

Las entradas analógicas se refieren a señales eléctricas continuas que varían en intensidad o voltaje en función de la magnitud de la variable que se está midiendo (Romero-Ortells Labrada, 2023).

En la figura 8 se detallan estas señales se utilizan comúnmente en sistemas de control y adquisición de datos para medir variables como la temperatura, la presión, la luz, entre otras (Romero-Ortells Labrada, 2023).

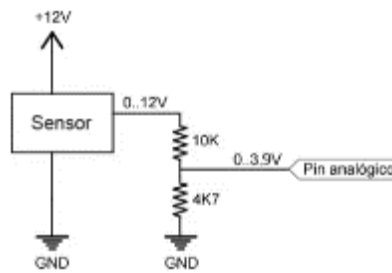


Figura 7 : Diseño del prototipo (Romero-Ortells Labrada, 2023).

2.8. Salidas análogas

Las salidas analógicas se refieren a señales eléctricas continuas que varían en intensidad o voltaje de acuerdo con la información digital proporcionada por un sistema de control o microcontrolador (Huera Lomas, 2023).

En la figura 9 se detallan estas señales se utilizan para controlar dispositivos que requieren niveles variables de voltaje o corriente para su funcionamiento, como motores, válvulas, luces regulables, entre otros (Huera Lomas, 2023)..

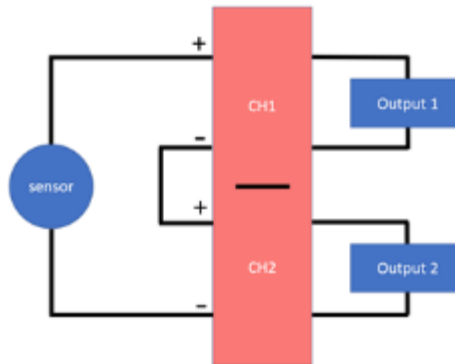


Figura 8 : Salida analógica (Huera Lomas, 2023).

2.9. Protocolos de comunicación

Los protocolos de comunicación son esenciales para permitir la transferencia de datos entre el microcontrolador y otros dispositivos, como sensores, actuadores o incluso otros microcontroladores (Zumba Ortiz, 2022).

Algunos protocolos comunes utilizados en microcontroladores incluyen UART (Universal Asynchronous Receiver/Transmitter), SPI (Serial Peripheral Interface), I2C (Inter-Integrated Circuit), y CAN (Controller Area Network), entre otros. Cada uno de estos protocolos tiene sus propias características y ventajas, lo que permite la comunicación en diferentes aplicaciones y entornos (Zumba Ortiz, 2022)..

2.9.1. UART

Es un protocolo de comunicación comúnmente utilizado en microcontroladores para la transmisión y recepción de datos serie asincrónicos (Rico, 2021).

En la figura 10 se detallan que los dos pines, uno para transmitir (TX) y otro para recibir (RX), y se emplea para establecer comunicación punto a punto entre el microcontrolador y otros dispositivos, como sensores, módulos de comunicación, pantallas, entre otros. La comunicación UART se basa en la transmisión de bits de datos junto con bits de inicio y parada para sincronizar la comunicación, lo que la hace simple y ampliamente compatible (Rico, 2021).

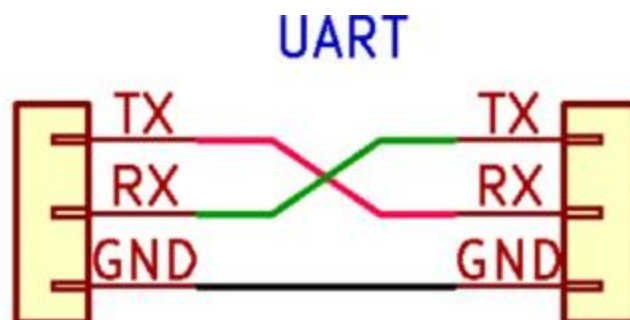


Figura 9 : UART (Rico, 2021).

2.9.2. SPI

Es un protocolo de comunicación utilizado en microcontroladores para la transferencia de datos serie síncronos entre el microcontrolador y otros

dispositivos, como sensores, memoria flash, pantallas, módulos de comunicación y otros periféricos (Arias Ronquillo, 2023).

En la figura 11 se detalla el SPI que utiliza al menos cuatro líneas de señal: MOSI (Master Out Slave In) para la transmisión de datos desde el microcontrolador al dispositivo, MISO (Master In Slave Out) para la recepción de datos desde el dispositivo al microcontrolador, SCK (Serial Clock) para la sincronización de la comunicación, y SS/CS (Slave Select o Chip Select) para seleccionar el dispositivo con el que se desea comunicar, lo que permite la conexión de varios dispositivos en un bus SPI. Este protocolo es rápido y adecuado para aplicaciones donde se necesita una alta velocidad de transferencia de datos (Sánchez, 2022).

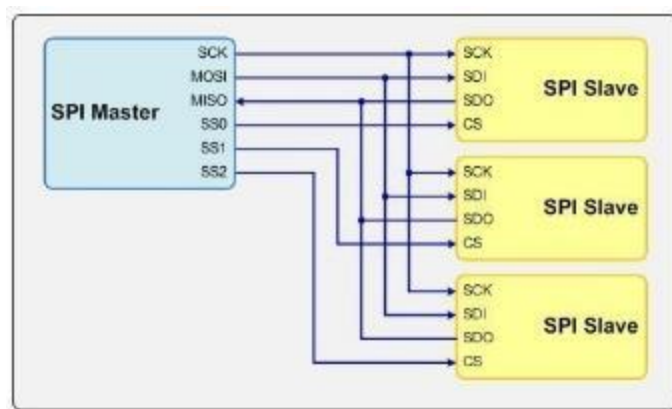


Figura 10 : SPI

2.9.3. CAN

Es un protocolo de comunicación utilizado en microcontroladores y sistemas embebidos, especialmente en aplicaciones automotrices e industriales, donde se requiere una comunicación confiable y robusta entre dispositivos distribuidos (Martinez, 2023).

En la figura 12 se detalla el protocolo CAN se basa en un bus de comunicación que permite la transmisión de datos entre múltiples nodos (microcontroladores u otros dispositivos) de manera diferencial, lo que aumenta la inmunidad a

ruidos eléctricos y garantiza la integridad de los datos en entornos ruidosos (Martinez, 2023).

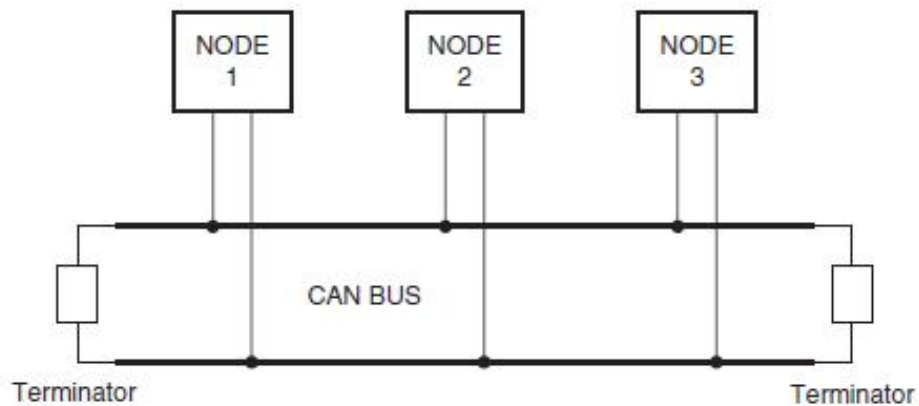


Figura 11 : CAN (Martinez, 2023).

2.9.4. I2C

Es un protocolo de comunicación ampliamente utilizado en microcontroladores para la transferencia de datos serie síncronos entre el microcontrolador y otros dispositivos, como sensores, pantallas, memoria EEPROM, y otros periféricos. I2C utiliza dos líneas de señal: SDA (Serial Data) para la transmisión de datos y SCL (Serial Clock) para la sincronización de la comunicación (Quispe, 2019).

En la figura 13 se muestra una característica importante de I2C es que permite conectar múltiples dispositivos en un único bus, cada uno identificado por una dirección única, lo que lo hace adecuado para aplicaciones con varios periféricos (Quispe, 2019)..

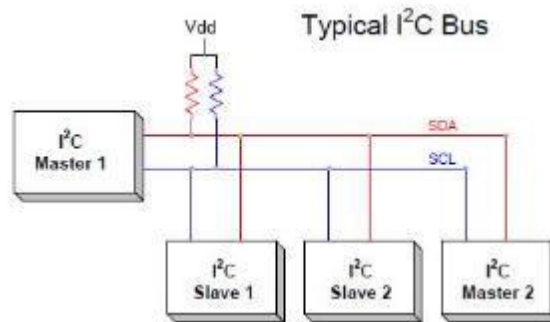


Figura 12 : I2C (Quispe, 2019).

2.10. Software de Código Abierto y OpenPLC

El software de código abierto es un tipo de software cuyo código fuente está disponible públicamente y se distribuye bajo una licencia que permite a los usuarios ver, modificar y distribuir el código según sus necesidades (Ramírez, 2022).

En la figura 14 se muestra el OpenPLC es un proyecto que combina estos dos conceptos al ofrecer un entorno de programación basado en código abierto para controladores lógicos programables (PLC) ,proporciona una plataforma de software que permite a los usuarios desarrollar aplicaciones de automatización industrial utilizando lenguajes de programación estándar como Ladder Logic o Structured Text y luego cargar estas aplicaciones en hardware de PLC compatible (Inchusta, 2023).

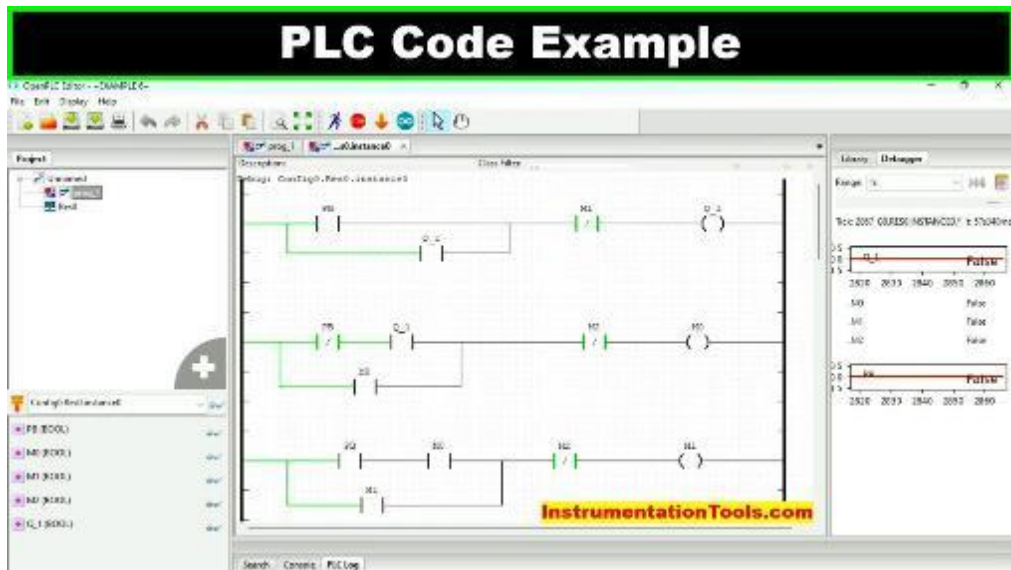


Figura 13 : OPEN PLC

Al ser de código abierto, OpenPLC ofrece flexibilidad y transparencia, lo que permite a los usuarios personalizar su lógica de control y adaptarla a sus necesidades específicas sin depender de software propietario costoso. Esto lo convierte en una herramienta valiosa en el campo de la automatización industrial y en aplicaciones de control (Aceitón, 2020).

2.11. Aplicaciones Industriales de PLCs

Las aplicaciones industriales de PLC (Controladores Lógicos Programables) con OpenPLC son diversas y abarcan una amplia gama de sectores industriales. OpenPLC, al ser una plataforma de código abierto, permite la flexibilidad y personalización en el desarrollo de soluciones de automatización (Aceitón, 2020)..

2.11.1. Control de maquinaria

OpenPLC se utiliza para controlar y supervisar maquinaria industrial, como líneas de ensamblaje, máquinas de procesamiento y equipos de fabricación (Aceitón, 2020)..

2.11.2. Automatización de procesos:

En la industria química, petroquímica y de procesamiento, OpenPLC se emplea para automatizar y controlar procesos de producción, asegurando la consistencia y la seguridad (Aceitón, 2020)..

2.11.3. Control de sistemas HVAC

OpenPLC se utiliza para gestionar sistemas de calefacción, ventilación y aire acondicionado en edificios comerciales e industriales, optimizando el consumo de energía (Aceitón, 2020)..

2.11.4. Automatización de edificios

OpenPLC se aplica en la gestión de sistemas de iluminación, control de accesos, seguridad y monitoreo en edificios inteligentes (Aceitón, 2020)..

2.11.5. Control de bombas y válvulas

Se utiliza en aplicaciones de control de flujo, como la distribución de agua, sistemas de riego agrícola y tratamiento de aguas residuales (Aceitón, 2020)..

2.11.6. Control de procesos en la industria alimentaria

OpenPLC asegura la precisión y eficiencia en la producción de alimentos, desde la mezcla de ingredientes hasta el envasado (Aceitón, 2020)..

2.11.7. Automatización en la industria automotriz

OpenPLC se utiliza en líneas de producción de automóviles para controlar robots, transportadores y sistemas de ensamblaje (Aceitón, 2020)..

2.11.8. Control de hornos industriales

Se emplea para gestionar y monitorear hornos de fundición, secado y tratamiento térmico en la industria metalúrgica (Aceitón, 2020)..

2.11.9. Gestión de sistemas de energía

OpenPLC se utiliza para controlar la generación, distribución y consumo de energía en plantas de energía y subestaciones eléctricas (Aceitón, 2020)..

2.11.10. Control de sistemas de logística y transporte

En almacenes y centros de distribución, OpenPLC se utiliza para optimizar el movimiento de mercancías y la gestión de inventario (Aceitón, 2020)..

2.12. Aspectos de Seguridad en PLCs y Sistemas Automatizados

La seguridad en PLCs y sistemas automatizados es crucial para mitigar riesgos potenciales. Esto implica la implementación de funciones de seguridad en los PLCs, como parada de emergencia y limitación de velocidad, control de acceso para evitar manipulaciones no autorizadas, protección contra intrusiones mediante firewalls y autenticación, actualizaciones regulares para abordar vulnerabilidades, cifrado de comunicaciones, sistemas de detección de intrusiones, cumplimiento de estándares eléctricos y, fundamentalmente, la capacitación adecuada del personal para garantizar el uso seguro de estos sistemas (Andrango Pachacama, 2023).

2.13. Beneficios entre OpenPLC y plc

Los beneficios entre OpenPLC y PLCs comerciales es esencial para la toma de decisiones en aplicaciones industriales. OpenPLC, al ser de código abierto, ofrece ventajas como la gratuidad del software y la flexibilidad en la personalización. Sin embargo, puede requerir más tiempo de desarrollo y capacitación del personal.

Los PLCs comerciales, aunque tienen costos iniciales más altos, a menudo cuentan con soporte técnico, certificaciones y características específicas de seguridad, lo que puede ser crítico en entornos industriales.

El análisis debe considerar los costos totales de propiedad, incluyendo implementación, mantenimiento, actualizaciones y la importancia de la fiabilidad y seguridad en la operación. En resumen, la elección entre OpenPLC y PLC comercial depende de las necesidades específicas de la aplicación, los recursos disponibles y los requisitos de seguridad y soporte.

3. MARCO METODOLÓGICO

3.1. Diseño del prototipo

Para el diseño del prototipo se optó por una LattePanda delta 432 el cual posee un sistema operativo Windows 10 integrado, permitiendo la instalación de programas como el OPENPLC y teniendo como puertos GPIO un embebido Atmel 32UA o Arduino Leonardo, con un sistema de banda transportadora, entradas digitales, salidas digitales y salidas pwm para el control de una banda transportadora como se muestra en la Figura 15.



Figura 15 : Diseño del prototipo

3.2. Funcionamiento del prototipo

El proyecto tendrá como función el ayudar al aprender las herramientas como OPENPLC para la integración de un proceso industrial a escala, el cual es una banda lineal de transporte que se muestra en la Figura 16, teniendo un sensor digital para lectura de objetos y un motor como actuador, conectados a un sistema LattePanda con un Microcontrolador Atmel 32UA o Arduino Leonardo.

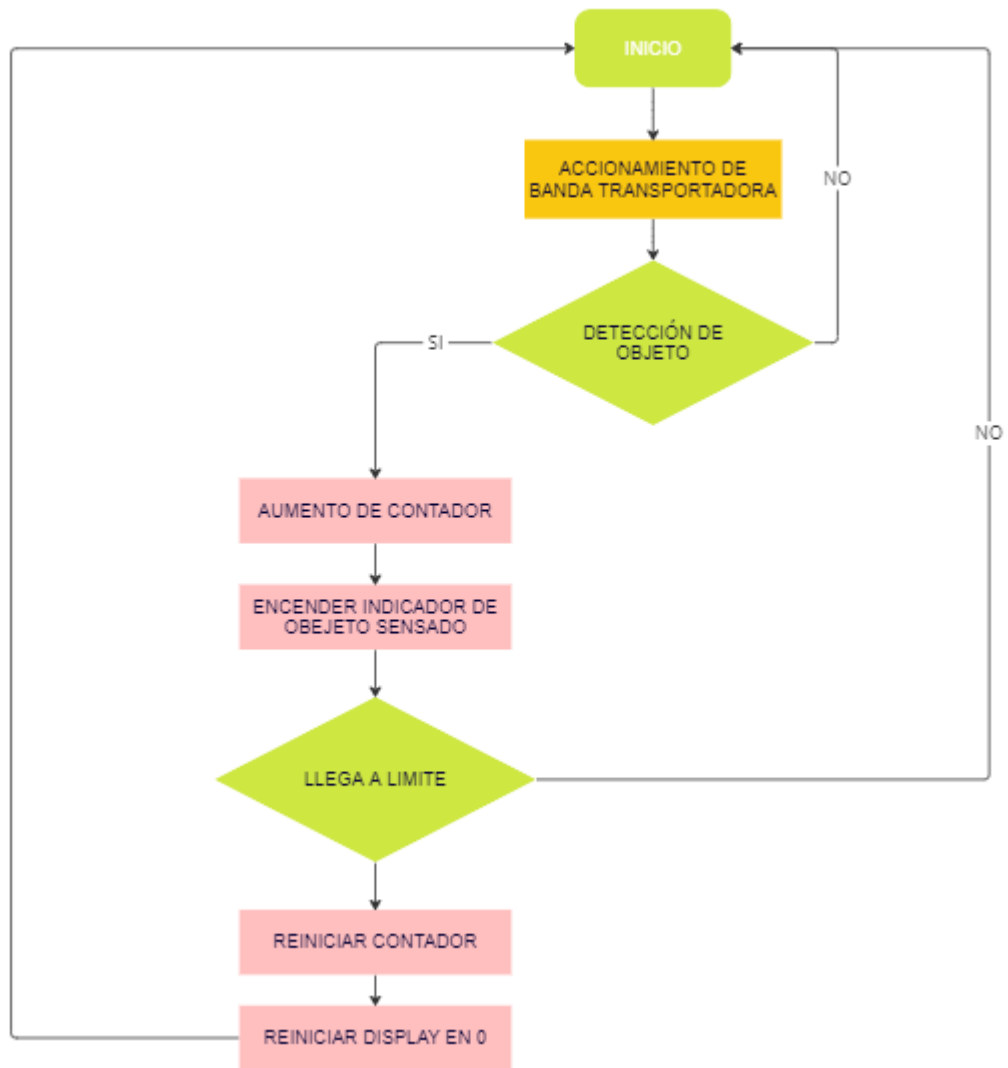


Figura 16 : Funcionamiento del proceso de la banda

3.3. Componentes del prototipo

Los componentes del prototipo son 2 una caja de control donde estarán la electrónica como el LattePanda, reguladores de voltaje, fuentes de voltajes, controladores de motores y tarjetas de acoples. El otro elemento es una banda transportadora que contara con un motor y un sensor infrarrojo.

3.3.1. Caja de control

Consta de una placa central la cual se conecta al LattePanda a manera que sirva como tarjeta de distribución para los demás periféricos como

controladores de motores, pulsantes, indicadores led, indicadores numéricos y conexión con sensores como se muestra en la Figura 17.

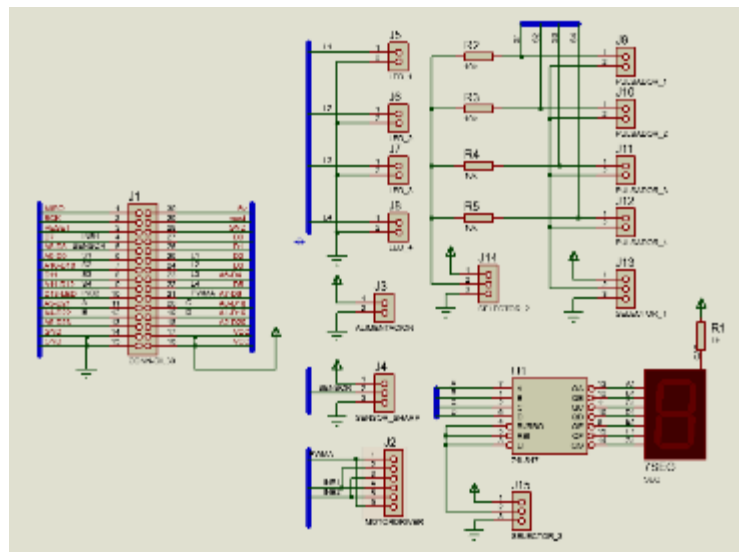


Figura 17 : Conexiones de tarjeta de distribución en LattePanda

El LattePanda se conecta a la tarjeta de distribución mediante sus salidas GPIO teniendo entradas digitales, analógicas y comunicación como se muestra en la Figura 18.

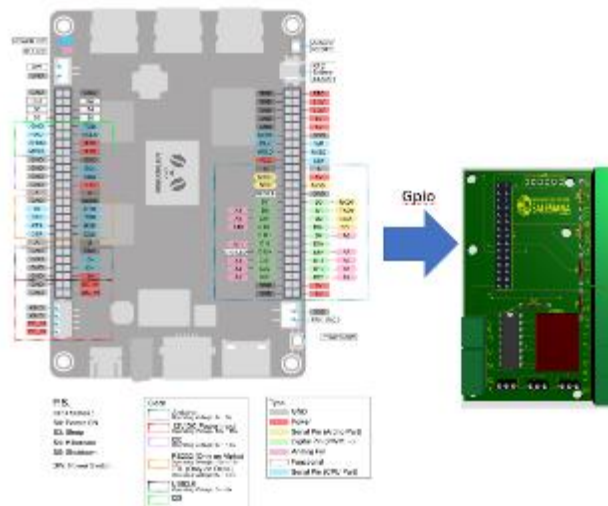


Figura 18 : Conexiones de tarjeta de distribución en LattePanda

La placa de distribución se conecta al controlador de motores VNHS2P30 que sirve para controlar el voltaje mediante una señal PWM y el sentido de giro de un actuador siguiendo la distribución que se muestra en el esquema de la Figura 19.



Figura 19 : Conexiones de LattePanda y controlador de motor

El sistema consta de una visualización mediante un display de 7 segmentos conectado a un convertor de binario a código bcd a manera de ahorro de pines como se muestra en la Figura 20.

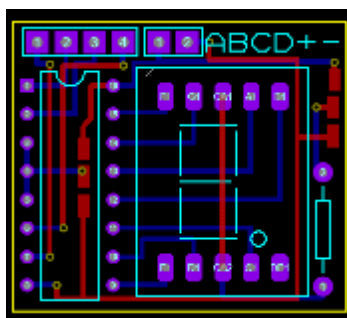


Figura 20 : Conexiones de LattePanda y display de 7 segmentos

Para el manejo del prototipo se opta por una pantalla táctil de 7 pulgadas conectada mediante un cable HDMI a la latepanda y un cable USB V8 como se muestra en la Figura 21.



Figura 21 : Conexiones de LattePanda y pantalla 7 pulgadas

3.3.2. Banda de transporte

Para la sección de la banda de transporte se optó por usar el material PLA ya que la banda en su totalidad es fabricada en una impresora 3D en la sección de anexos A se encuentran las partes y componentes que se muestran en la Figura 22.



Figura 22 : Conexiones de LattePanda y pantalla 7 pulgadas

3.4. OPENPLC

En la Figura 23 se muestra el ide de programación que es OPENPLC es una plataforma de código abierto que sirve como un controlador lógico programable en aplicaciones de automatización industriales mediante uso de microcontroladores .

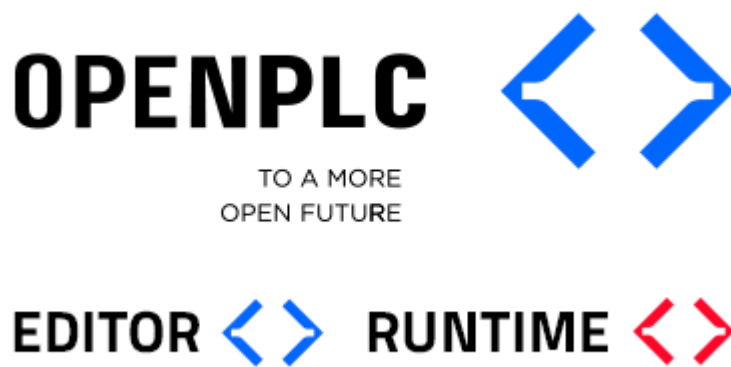


Figura 23 : OPENPLC

3.5. Programación en OPENPLC

3.5.1. Lenguajes de programación en OPENPLC

Los lenguajes de programación en OPENPLC son herramientas clave para definir el comportamiento de estos dispositivos en aplicaciones de automatización utilizando los microcontroladores (Rendon Castillo, 2023).

3.5.1.1. Function Block Diagram (FBD)

Es un lenguaje de programación gráfico que se utiliza en la automatización industrial, definido según la norma IEC 61131-3. Este lenguaje es especialmente efectivo para representar sistemas de control lógico y procesos mediante bloques funcionales, lo que facilita la comprensión y el diseño de sistemas complejos (González Gimeno, 2023). Como se muestra en la Figura 24.

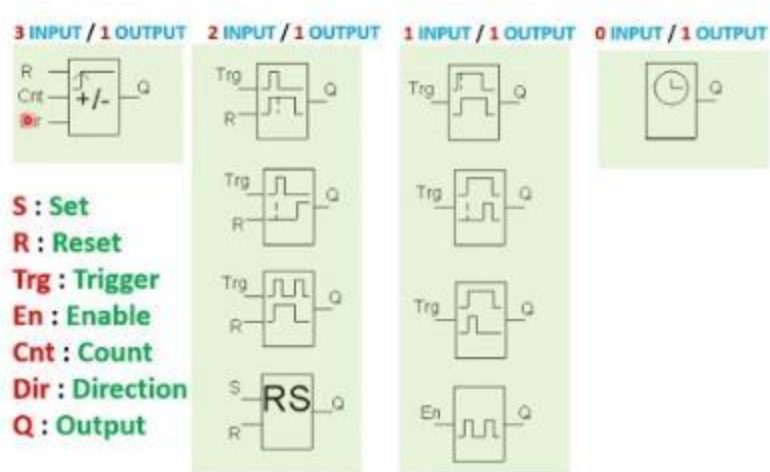


Figura 24 : Function Block Diagram (FBD)

3.5.1.2. Ladder Logic(LD)

Es un lenguaje de programación gráfico que emula los esquemas eléctricos de relés. Es uno de los lenguajes más antiguos y comúnmente usados en la programación de PLC, establecido bajo la norma IEC 61131-3. Su diseño e interfaz intuitiva lo hacen particularmente accesible para ingenieros y técnicos con experiencia en electricidad y sistemas de control convencionales (González Gimeno, 2023). Como se muestra en la Figura 25.

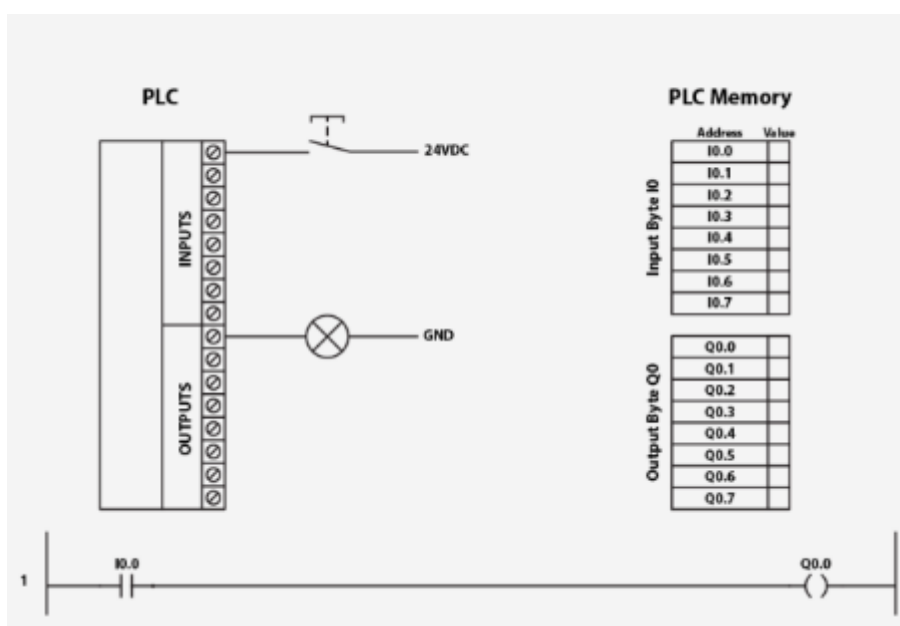


Figura 25 :Ladder Logic(LD)

3.5.1.3. Structured Text (ST)

Es un lenguaje de programación de alto nivel definido por la norma IEC 61131-3. Es similar a lenguajes de programación de computadoras como Pascal, C y ADA, y se utiliza para implementar algoritmos complejos y tareas de control que serían difíciles o engorrosas de expresar en lenguajes gráficos como Ladder Logic o Function Block Diagram (González Gimeno, 2023). Como se muestra en la Figura 26.

```
1 PROGRAM FCU
2 VAR
3   MotorOverload : ARRAY[1..3] OF BOOL;
4   MotorOverCurrent : ARRAY[1..3] OF BOOL;
5   MotorFBStart : ARRAY[1..3] OF BOOL;
6   MotorFBStop : ARRAY[1..3] OF BOOL;
7   MotorRoll : ARRAY[1..3] OF BOOL;
8   i : INT;
9
10 END_VAR
11
12
13
14 FOR i: 1 TO 3 BY 1 DO
15   IF MotorFBStop[i] OR MotorOverload[i] OR MotorOverCurrent[i] THEN
16     MotorRoll[i]:=0;
17   ELSEIF MotorFBStart[i] THEN
18     MotorRoll[i]:=1;
19   END_IF
20 END_FOR
```

Figura 26 : Structured Text (ST)

3.5.1.4. Instruction List (IL)

Es un lenguaje de programación de bajo nivel modelado en gran medida a partir de los lenguajes de ensamblador utilizados en la programación de computadoras. Se caracteriza por su estructura lineal y uso de instrucciones sucesivas, cada una representando una operación elemental. En el contexto de PLCs, IL ofrece un control detallado y granular del proceso de automatización (González Gimeno, 2023). Como se muestra en la Figura 27.

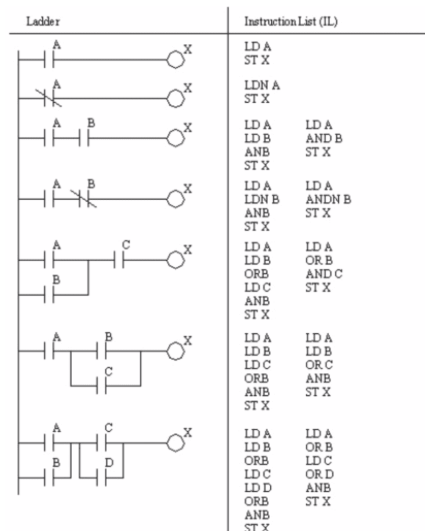


Figura 27 : Instruction List (IL)

3.5.1.5. Sequential Function Chart (SFC):

El SFC permite una visualización clara y estructurada del proceso de control, facilitando el diseño, la comprensión y el mantenimiento de programas complejos (González Gimeno, 2023). Como se muestra en la Figura 28.

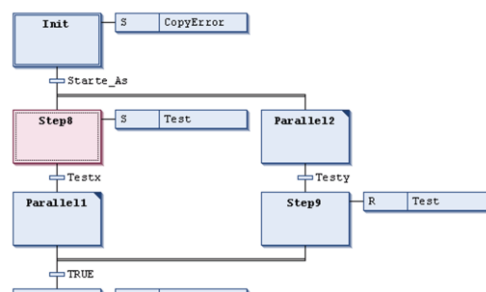


Figura 28 : Sequential Function Chart

3.6. Declaración de variables en OPENPLC en LattePanda

Al iniciar la programación es necesario que previamente se debe crear un proyecto en OPENPLC indicando el tipo de lenguaje de programación a utilizar como se muestra en la Figura 29.

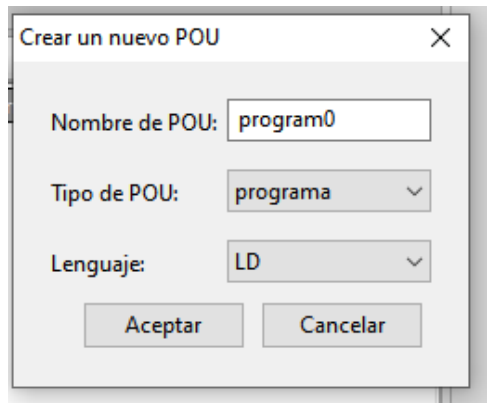


Figura 29 : Creación de proyecto y selección de lenguaje en OPENPLC

En la sección de OPENPLC se tendrán diferentes ventanas una donde se muestra el proyecto como su lenguaje de programación, la ventana donde se programa y se crean las variables a utilizar con sus direcciones, la ventana de librerías que contendrán los elementos para la programación como se muestra en la Figura 30.

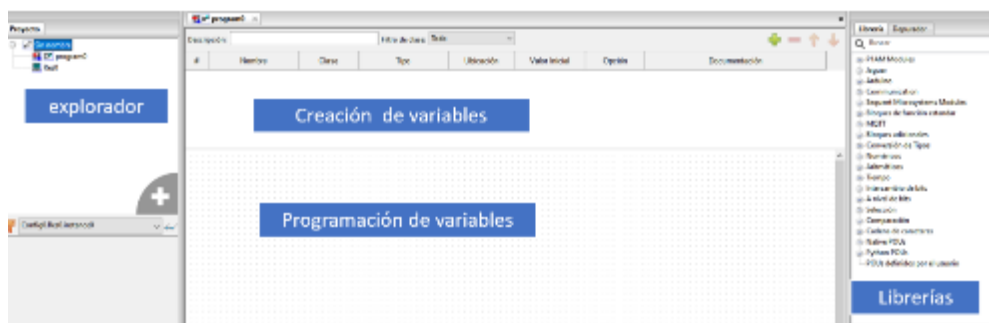


Figura 30 : Secciones de OPENPLC

Para el direccionamiento de ubicaciones de E/S se realiza mediante el uso de secuencias de caracteres especiales.

Estas secuencias son una concatenación del signo de porcentaje “%”, un prefijo de ubicación, un prefijo de tamaño y uno o más números naturales separados por espacios en blanco.

Se admiten los siguientes prefijos de ubicación:

- I para la entrada
- Q para salida
- M de memoria
- X Por bit (1 bit)
- B para byte (8 bits)
- W para palabra (16 bits)
- D para palabra doble (32 bits)
- L para palabra larga (64 bits)

Por ejemplo, al leer el estado de la primera entrada digital en una variable BOOL, se declara su variable ubicada en: %IX0.0.

Al escribir el contenido de una variable UINT en la segunda salida analógica, se declara su variable UINT ubicada en %QW2.

En la tabla 1 se muestra un ejemplo con las variables, pin y dirección.

Dato	Pines	Dirección
Digital In	2, 3, 4, 5, 6	%IX0.0 – %IX0.7
Digital Out	7, 8, 10, 11, 12, 13	%QX0.0 – %QX0.7
Analog In	A1, A2, A3, A4, A5, A6, A7	%IW0 – %IW3
Analog Out	9, 14	%QW0 – %QW3

Tabla 1: Tabla de direccionamiento en OPENPLC

Se procede a crear la dirección de los pines a utilizar en la configuración previamente se debe configurar las salidas de acuerdo con la siguiente tabla 2.

Descripción	Pin	Dirección
ENTRADA_1	D-9	%IX0.0
ENTRADA_2	D-10	%IX0.1
ENTRADA_3	D-11	%IX0.2
ENTRADA_4	D-12	%IX0.3
SENSOR	D-8	%IX0.4
SALIDA_1	D-2	%QX0.0
SALIDA_2	D-3	%QX0.1
SALIDA_3	D-4	%QX0.2
SALIDA_4	D-5	%QX0.3
DISPLAY_1	D-21	%QX0.4
DISPLAY_2	D-22	%QX0.5
DISPLAY_3	D-18	%QX0.6
DISPLAY_4	D-19	%QX0.7
INB_1	D-7	%QX1.0
INB_2	D-13	%QX1.1
Pwm_A	D-6	%QW0

Tabla 2: Tabla de direccionamiento del proyecto en OPENPLC

En OPENPLC en la sección de declaración para la utilización de los pines, también se procede a iniciar todas las variables con un estado bajo como se muestra en la Figura 31.

#	Nombre	Clase	Tipo	Ubicación	Valor Inicial	Opción	Documentación
1	ENTRADA_1	Local	BOOL	%IX0.0	0		
2	ENTRADA_2	Local	BOOL	%IX0.1	0		
3	ENTRADA_3	Local	BOOL	%IX0.2	0		
4	ENTRADA_4	Local	BOOL	%IX0.3	0		
5	SENSOR	Local	BOOL	%IX0.4	0		
6	SALIDA_1	Local	BOOL	%QX0.0	0		
7	SALIDA_2	Local	BOOL	%QX0.1	0		
8	SALIDA_3	Local	BOOL	%QX0.2	0		
9	SALIDA_4	Local	BOOL	%QX0.3	0		
10	DISPLAY_1	Local	BOOL	%QX0.4	0		
11	DISPLAY_2	Local	BOOL	%QX0.5	0		
12	DISPLAY_3	Local	BOOL	%QX0.6	0		
13	DISPLAY_4	Local	BOOL	%QX0.7	0		
14	INB_1	Local	BOOL	%QX1.0	0		
15	INB_2	Local	BOOL	%QX1.1	0		
16	PWM_A	Local	WORD	%QW0	0		

Figura 31 : Declaración de variables en OPENPLC para LattePanda

3.7. Programación OPENPLC en LattePanda

En la programación del prototipo se configura en el OPENPLC el tipo de tarjeta o microcontrolador a utilizar en este caso Arduino Leonardo, el medio de comunicación serial como se muestra en la Figura 32.

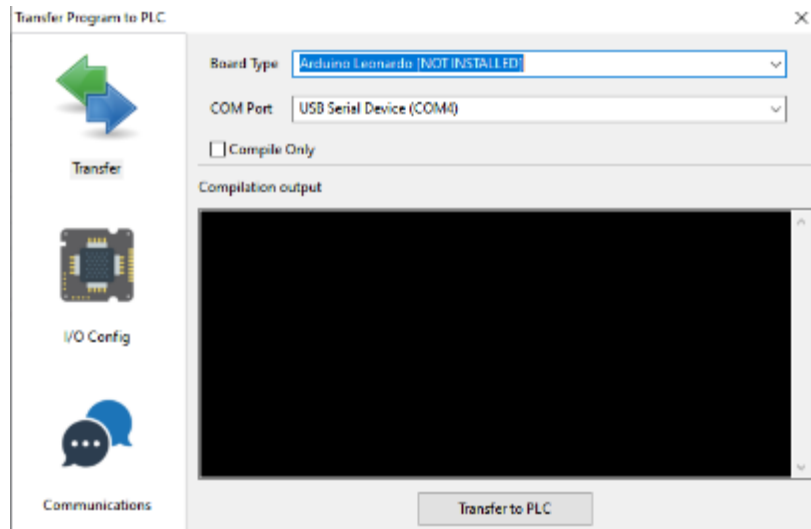


Figura 32 : Programación en OPENPLC para LattePanda

Posterior se debe configurar las instancias de los puertos de entrada y salida de acuerdo con la declaración de puerto de la sección 3.6 como se muestra en la Figura 33.

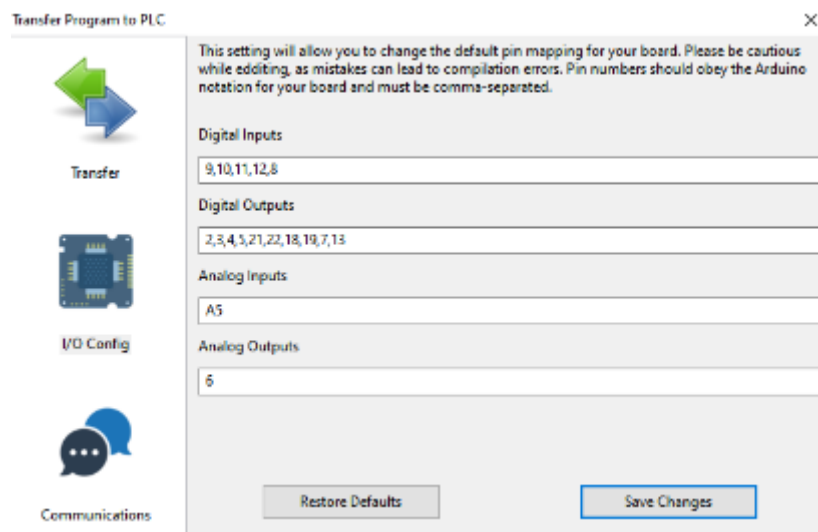


Figura 33 : Configuración de puertos en OPENPLC para LattePanda

3.8. Esquema de tarjetas PCB

3.9. Tarjeta de distribución

El sistema cuenta con una tarjeta de distribución que se detalla en la Figura 34 la cual se conecta a los puertos GPIO del sistema embebido LattePanda y permite la conexión con los visualizadores led, entradas digitales, controlador de motor y entrada del sensor. En la Figura 35 se detalla el diseño de pcb en proteus tomando en cuenta el diagrama descrito.

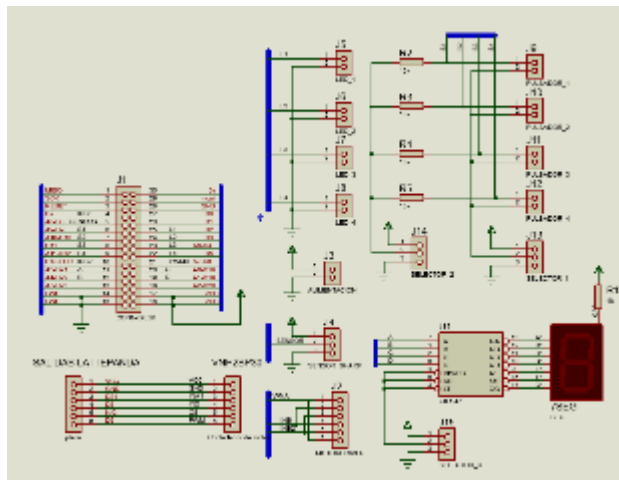


Figura 34 : Esquema de tarjeta

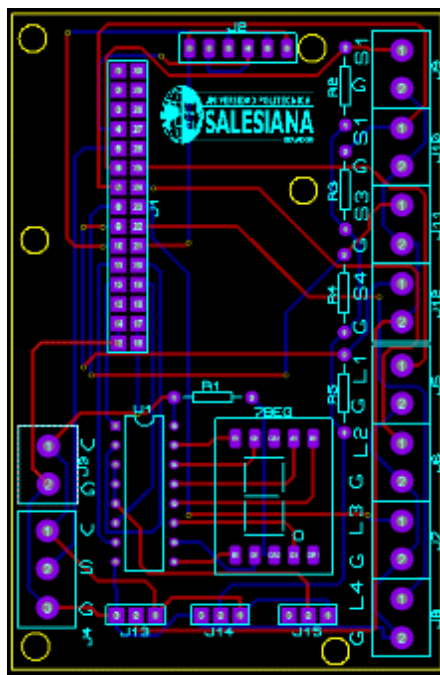


Figura 35 : Diseño de tarjeta

3.10. Esquema de potencia para LattePanda

El esquema de la conexión del prototipo se detalla en la Figura 36 teniendo como entrada una señal análoga de 120VAC que mediante los reguladores en las fuentes de voltaje se conectarán al sistema embebido, tarjeta de control de motores, tarjeta de relé y también se detalla la conexión de las entradas al sistema LattePanda.

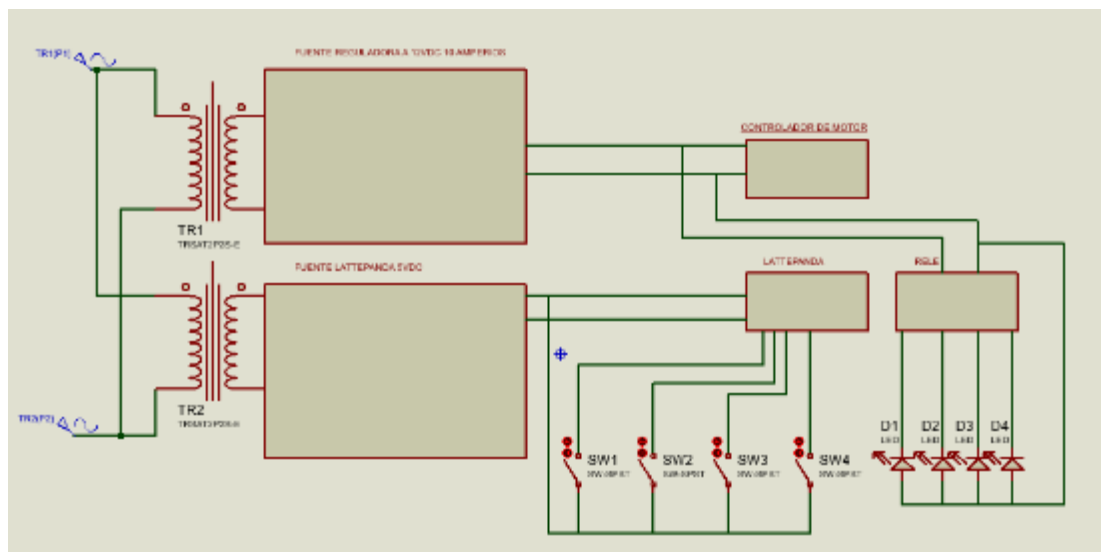



Figura 36 : Esquema de la potencia

4. MANUAL DE PRÁCTICA

Práctica: Control de una banda transportadora y conteo de productos

		GUÍA DE PRÁCTICA DE LABORATORIO	
CARRERA: Ingeniería Electrónica		ASIGNATURA:	
No.:	1	TÍTULO: Control de una banda transportadora y conteo de productos	
OBJETIVOS:			
<ul style="list-style-type: none">Realizar la configuración de un autómata programable usando microcontrolador del sistema LattePanda con OPENPLC IUtilizar OPENPLC para la lectura de datos digitales mediante LattePandaUtilizar LattePanda para controlar una banda transportadora.Realizar el conteo de productos en un display de 7 segmentos			
INSTRUCCIONES:		1. Analizar la solución propuesta de la práctica ubicada en los anexos de la memoria técnica.	
		2. Conectar el módulo a la red eléctrica 110 VAC	
		3. Realizar las conexiones correctas del módulo de acuerdo con el esquemático planteado en la resolución del anexo de la PRÁCTICA	
ACTIVIDADES POR DESARROLLAR			
1. Configurar OPENPLC para utilizar el microcontrolador en la LattePanda			
2. Realizar el control de la banda transportadora			
3. Aprender el entorno de programación			
RESULTADOS:			
<ul style="list-style-type: none">Desarrollo de técnicas para programar en OPENPLCComprender el manejo de las herramientas en OPENPLC.			
CONCLUSIONES:			
<ul style="list-style-type: none">Al implementar el control del sistema ayuda al aprendizaje en programas de licencia libre para autómatas programables basados en microcontroladores.Para el óptimo funcionamiento es necesario configurar las direcciones de entrada y salidas pertinentes en el microcontrolador en OPENPLC			
RECOMENDACIONES:			
<ul style="list-style-type: none">Revisar la alimentación del sistemaConsiderar la configuración de la práctica a realizarse en el módulo didáctico.Revisar el marco metodológico y el anexo de programación para mayor aprendizaje del desarrollo y programación interna del modulo			

Docente:

Firma: _

5. RESULTADOS

En el desarrollo del proyecto se tendrá como meta la creación de un módulo que integra la tecnología de un sistema embebido LattePanda mediante su microcontrolador Atmega 32UA y un programa de software libre OPENPLC para la automatización de un proceso a escala de una banda transportadora como se muestra en la Figura 37.

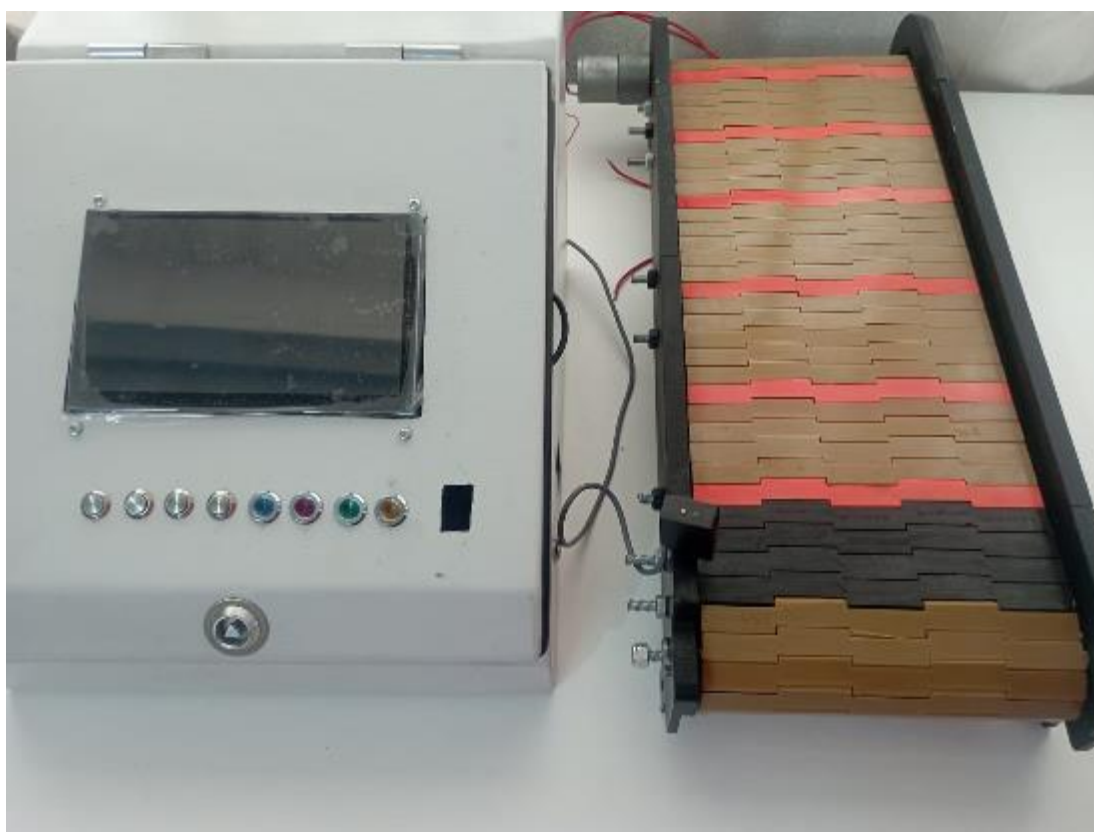


Figura 37 : Prototipo final

El proceso de la configuración y el manual de como programar el prototipo se encuentra en el anexo B, se detalla el manual para utilizar el OPENPLC y la LattePanda mediante el direccionamiento descrito en la sección 3.5.

El sistema al momento de captar una señal en el sensor procede a aumentar una instancia que almacena el número de datos o productos en la banda como se muestra en la Figura 38.

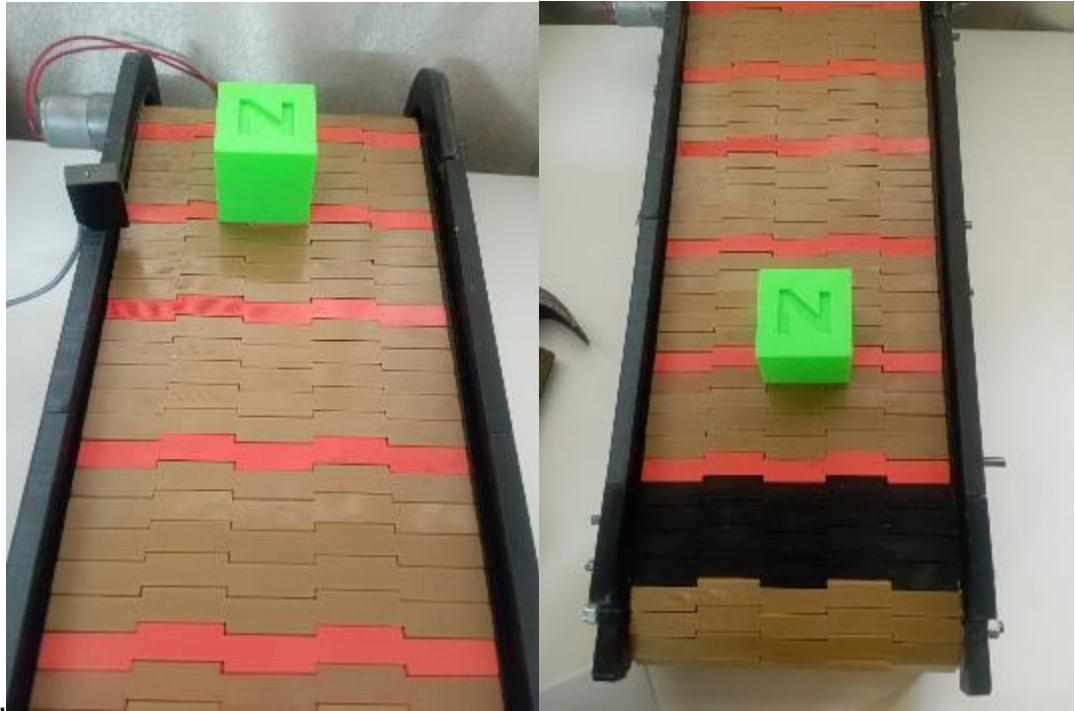


Figura 38 : Detección de producto en banda transportadora

Las instancias en el OPENPLC se programan mediante el lenguaje Ladder teniendo como entradas el valor del sensor y un bloque contador con una bandera que reinicia la función de cuenta al llegar al límite como se muestra en la Figura 39.

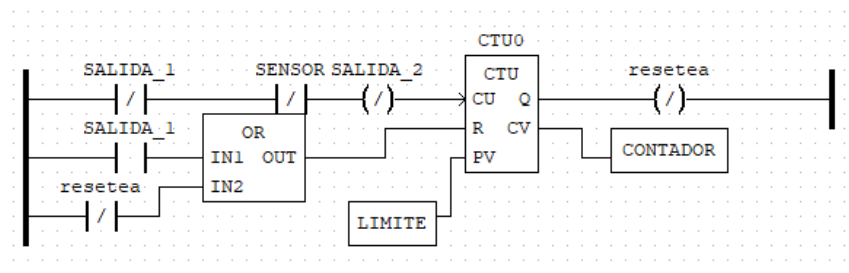


Figura 39 : Programa de detección de sensor y conteo de productos

Al iniciar el sistema consta con un enclavamiento de variables para el encendido y el apagado del sistema teniendo como datos un botón que hace la función de inicio, a la par otro botón que realiza la función de paro del sistema como se muestra en la Figura 40 y Figura 41.

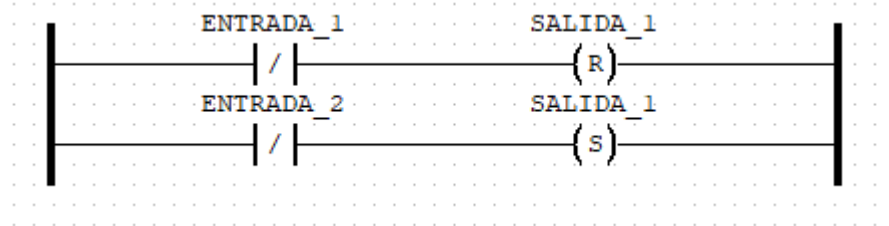


Figura 40 : Programa de activación y paro del sistema

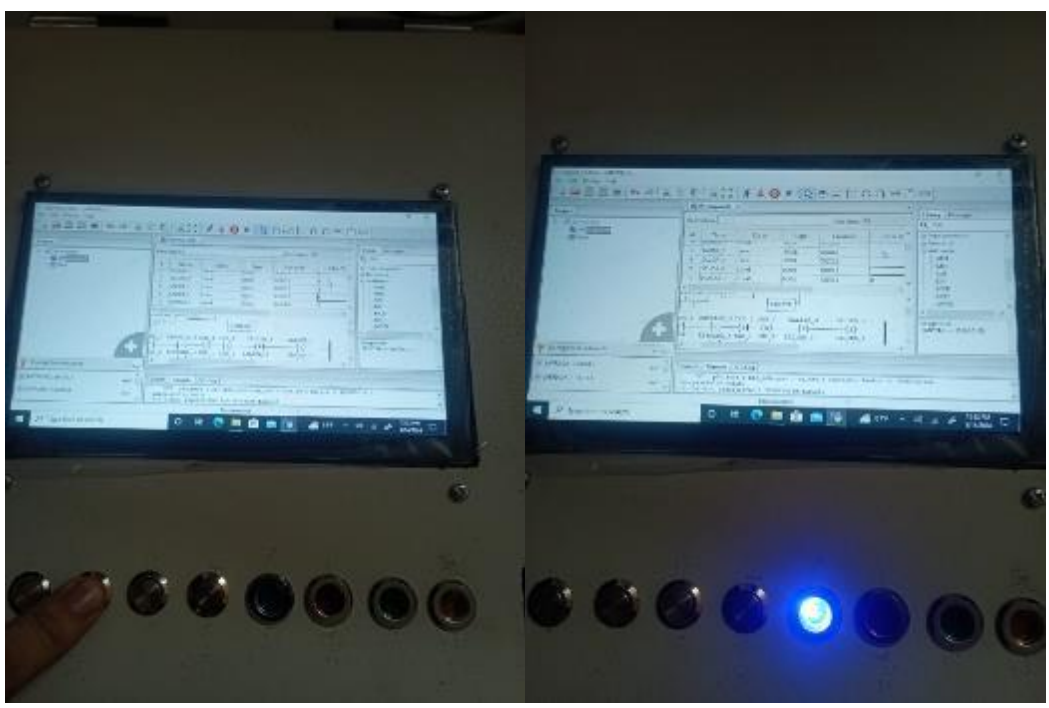


Figura 41 : Activación de marcha y activación de paro del sistema

En el control del sistema de banda transportadora se utilizan dos pulsantes para controlar el sentido de giro del motor antihorario o horario del sistema como se muestra en la Figura 42 del programa en Ladder en OPENPLC.

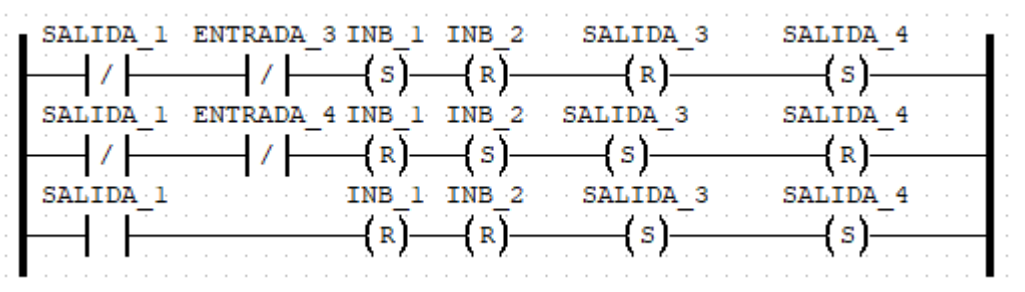


Figura 42 : Control del sentido del motor

Al momento del accionamiento del motor es necesario configurar una velocidad mediante una salida analógica en el direccionamiento se obtiene un valor de 27706 ya que al ser una salida tipo Word se configura con un rango de 0 a 65535 correspondiente a 10 bits de conversión DAC teniendo que usar un bloque MOVE para configurar la salida como se muestra en la Figura, que cuando el sistema está en reposo debe configurar una velocidad mínima de 0 y al ejecutar la máxima que se detalla en la Figura 43.

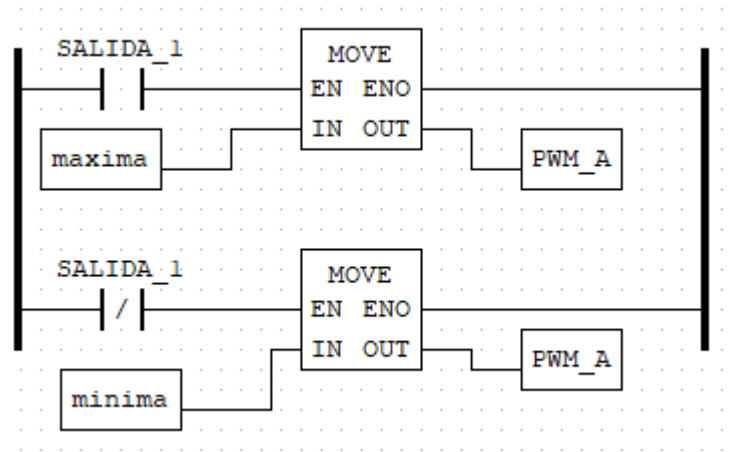


Figura 43 : Control de la velocidad en Ladder en OPENPLC

El método de visualización para el conteo es por medio un display de 7 segmentos el cual se programa mediante salidas digitales enviando un código binario por las salidas digitales en el LattePanda utilizando el código Ladder en OPENPLC como se muestra en la Figura 44.

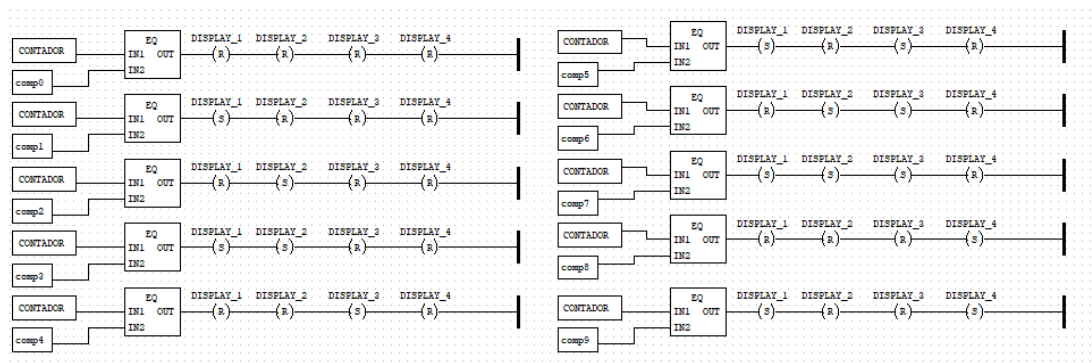


Figura 44 : Control del display en Ladder en OPENPLC

En la Figura 45, se observan el conteo de los productos en el display del 0 a 1.



Figura 45 : Accionamiento de código binario display 0-1

En la Figura 46, se observan el conteo de los productos en el display del 2 a 3.



Figura 46 : Accionamiento de código binario display 2-3

En la Figura 47, se observan el conteo de los productos en el display del 4 a 5.



Figura 47 : Accionamiento de código binario display 4-5

En la Figura 48, se observan el conteo de los productos en el display de 6 a 7.

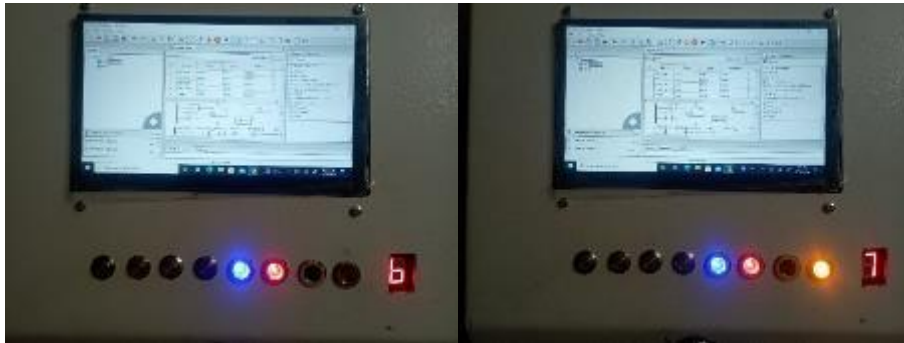


Figura 48: Accionamiento de código binario display 6-7

En la Figura 49, se observan el conteo de los productos en el display de 8 hasta que el contador reinicia.



Figura 49 : Accionamiento de código binario display 8- reinicio

CRONOGRAMA

En la tabla 3 se detalla el cronograma de actividades realizadas y el tiempo de duración del proyecto técnico.

<i>DURACIÓN DEL PROYECTO</i>		
ITEM	ACTIVIDAD	MES
1	ADQUISICIÓN DE LOS COMPONENTES PARA EL MODULO	1
2	DISEÑO DEL MODULO	1
3	PRUEBAS CON OPENPLC	2
4	PRUEBAS DE CONEXIÓN CON OPENPLC	2
5	RECOPIACIÓN DE INFORMACIÓN PARA REALIZACIÓN DE MONOGRAFIA	3
6	PRUEBAS Y REVISIÓN DEL PROYECTO	3
7	CORRECCIÓN DE DOCUMENTO CURRICULAR Y TÉCNICO	4

Tabla 3: Tabla de cronograma de actividades

PRESUPUESTO

En la tabla 4 se detalla el presupuesto y el costo de los materiales del proyecto técnico.

Descripción	Cantidad	Valor	Total
Estructura metálica o tablero de control	1	60	60
Banda Transportadora (Impresa en maquina 3D)	1	350	350
LattePanda Delta 432	1	349	40
Módulo de Relé	1	10	10
Pulsadores metálicos 22mm	4	3	12
Indicadores led cubierta metálica 22 mm	4	3	32
Tarjeta PCB distribución	1	20	20
Tarjeta PCB alimentación +5VDC	2	10	20
Tarjeta PCB alimentación 5GND	2	10	20
Tarjeta PCB alimentación +12VDC	2	10	20
Tarjeta PCB alimentación 12GND	2	10	20
Pantalla táctil 7 pulgadas	1	80	80
Canal riel din	5	5	25
Canal de distribución plástico	5	5	25
Fuente de alimentación de 12VDC	1	40	40
Fuente de alimentación de 5VDC	1	30	30
Motor POLOLU 59:1 12VDC	1	50	50
Controlador motor Vnh2sp30	1	40	40
Rollo de cables 26 AWG (multicolor)	1	80	80
Tornillo M4 X 20mm	100	0.05	4
Tuercas M4	200	0.05	4
Disipador LattePanda	1	55	55
Materiales Varios	1	90	90
TOTAL			1037

Tabla 4: Tabla de presupuesto del proyecto

CONCLUSIONES

- Se ha demostrado que la elección de una LattePanda como plataforma base para el PLC es adecuada, ya que proporciona suficiente potencia de cómputo y versatilidad para ejecutar OpenPLC de manera efectiva.
- La implementación exitosa de OpenPLC requiere un conocimiento de la programación lógica y de control, así como de las capacidades y limitaciones de la plataforma LattePanda que están descritas en el presente documento.
- En el desarrollo del programa es necesario tener en cuenta que la meta es simular un proceso industrial utilizando el sistema embebido LattePanda y OpenPLC por lo que es necesario tener claro que direcciones se programan en OPENPLC para el funcionamiento del prototipo.

RECOMENDACIONES

- Revisar la documentación del proyecto, desde el diseño hasta la implementación, ya que facilitará la revisión y realización de la práctica.
- Proporcionar capacitación adecuada al equipo de desarrollo en el uso de LattePanda y OpenPLC, así como en la programación lógica de control.
- Seleccionar el modelo de microcontrolador establecido en el manual de practica en el software OpenPLC configurado en LattePanda.
- Revisar el estado del equipo y dar un mantenimiento preventivo al sistema embebido LattePanda revisando pastas térmicas.

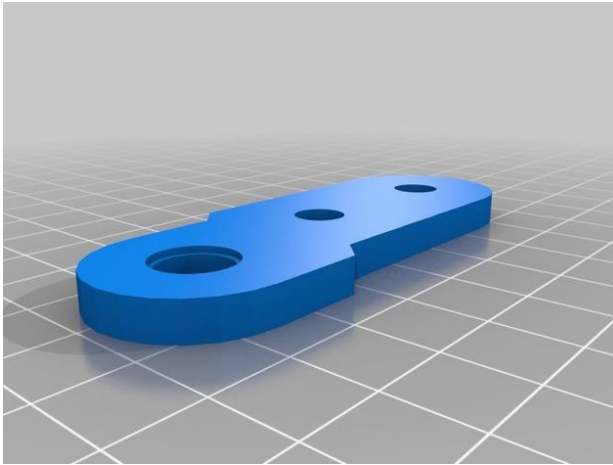
REFERENCIAS BIBLIOGRÁFICAS

- Aceitón, J. E. (2020). *El IOT-PLC: una nueva generación de controladores lógicos programables para la industria 4.0*.
- Acosta Contreras, I. P. (2023). *Diseño De Un Sistema Para Un Robot Autónomo Que Permita Optimizar El Cableado Estructurado Horizontal (Doctoral dissertation*. Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Networking y Telecomunicaciones).
- Andino, A. D. (2023). *Diseño y fabricación de robot entrenador de bádminton con reconocimiento de atleta*.
- Andrango Pachacama, J. E. (2023). . *Diseño y construcción de un sistema automatizado de dosificación de elementos controlado por un PLC S7 1200*.
- Aquieta Tibán, E. B. (2023). *Estudio comparativo de control PID, predictivo y difuso en un proceso de flujo*. Universidad Técnica de Cotopaxi; UTC.
- Arias Ronquillo, C. I. (2023). *Desarrollo de un entorno en Unity para la prueba y verificación de sensores*.
- Borjas Garcia, H. A. (2023). *Construcción de un sistema de acceso automático empleando una placa ordenador Raspberry PI para casilleros*.
- Briones García, M. R. (2023). *Diseño e implementación con algoritmo Fuzzy para control de velocidad en motor de corriente continua en sistemas embebidos LattePanda con monitoreo en plataforma IoT*.
- Echavarría-Flórez, I. &.C. (2020). *Métricas de legibilidad del código fuente: revisión sistemática de literatura*. Facultad de Ingeniería, 29(54), .
<https://doi.org/10.19053/01211129.v29.n54.2020.11756>.
- Fonseca-Beltrán, J. C.-A. (2021). *NUEVAS TECNOLOGÍAS EN LOS CONTROLADORES DIGITALES DE SEÑALES, ESTUDIO Y APLICACIONES*. . Ra Ximhai, 8(2), 249-256.
- Gómez Borges, M. I. (2020). *Programación de un controlador lógico difuso en un PLC M241: Aplicación práctica a un túnel de lavado industrial de textil*. Revista Cubana de Ciencias Informáticas, 14(3), 105-123.
- Gómez, G. (2022). *Diseño de un sistema de monitoreo de un prototipo de máquina industrial, para la presentación de datos en tiempo real con recursos de la nube de IBM en Colombia*.
- González Gimeno, O. (2023). *Uso de gemelos digitales para la validación y verificación de software en el estándar de programación industrial IEC-61131*.
- Huera Lomas, J. J. (2023). *Elaboración de una red multiplexada can en base a microcontroladores arduino para el control de sistemas embebidos automotrices*.
- Inchusta, A. M.-G. (2023). *ONTROL INTELIGENTE DE TURBINAS EÓLICAS EN TIEMPO REAL BASADO EN CONTROLADOR INDUSTRIAL COMPATIBLE CON IEC 61131*.
- J. Torres Ventura, A. H. (2023). *Rendimiento para la interoperabilidad entre Raspberry pi, ESP8266 y PLC con Node-RED para el IIoT*.
<https://doi.org/10.17163/ings.n29.2023.08> .
- MARTEL, I. V. (2021). *Sistema de monitorización y control remoto de las condiciones de cultivo para un invernadero con tecnología IoT y energía solar*.
- Martínez Rodríguez, A. J. (2023). *Sistema barra-esfera para prácticas de sistemas de control*.
- Martinez, D. M. (2023). *Sistema de emulación por software embebido de baterías Li-ion con bajo error*.
- Martínez, H. (2015). *Modulo didáctico para prácticas de laboratorio con controladores lógicos programables*. Doctoral dissertation, Universidad Autónoma de Nuevo León.

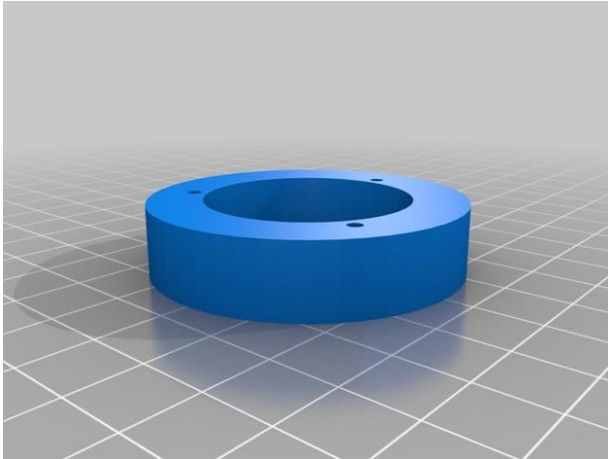
- Molina Araujo, M. J. (2018). *Automatización del sistema de control de la máquina empaquetadora de Blíster*. Revista Politécnica, 41(1), 25-28.
- Montenegro Garófalo, F. J. (2021). *onitorización de parámetros eléctricos de un manipulador móvil (Bachelor's thesis, Universidad Técnica de Ambato. Facultad de Ingeniería en Sistemas, Electrónica e Industrial. . Carrera de Ingeniería Industrial en Procesos de Automatización*.
- PANTOJA, W. L. (2013). *ENTORNO COLABORATIVO DE APOYO A LA MEJORA DE PROCESOS DE SOFTWARE EN PEQUEÑAS ORGANIZACIONES DE SOFTWARE*. Dyna, 80(177), 40-48.
- Poma Quispe, E. I. (2023). *Diseño e implementación de un sistema SCADA para la automatización de una planta industrial Caso de Estudio: planta industrial didáctica de un proceso termodinámico*.
- Quispe, U. (2019). *Diseño e implementación de una red de dispensadores automáticos de agua, basada en internet de las cosas, para el análisis de venta de agua por parte de la división de purificación de agua del Grupo Gundlach* .
- Rahmad, I. F. (2020). *Automatic Monitoring and Control System in Aeroponic Plant Agriculture. In 2020 8th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-5). IEEE*.
- Ramírez, J. G. (2022). *El uso del software de código abierto en el IoT*. Ciencia Latina Revista Científica Multidisciplinar, 6(6), 1867-1704.
- Rendon Castillo, P. A. (2023). *Comparación en el desempeño de plc físico marca siemens® respecto a un plc virtualizado aplicado al control de procesos*.
- Rico, H. E. (2021). *mplementación de protocolo de comunicación para pruebas funcionales entre sistema embebido de bajo costo y secadora convencional de ropa*.
- RIERA ABELLÁN, L. A. (2023). *Desarrollo de una aplicación interactiva para el diseño de controladores PID basada en Python*.
- Romero-Ortells Labrada, F. (2023). *Diseño e implementación de un sistema de medidas eléctricas para equipos de hasta 2, 3 kW*.
- Saldaña Calderon, P. D. (2023). *Diseño e implementación de un sistema automatizado para la producción de concreto premezclado para la empresa Movicreto*.
- Sánchez, J. F. (2022). *Sistema de Control de Agua y Nutrientes en Invernadero* .
- Suarez, V. I. (2023). *Los videojuegos comerciales como herramientas mediadoras de prácticas científicas y habilidades socioemocionales en educación primaria,. dissertation, Universidad Catolica de Cordoba* .
- Villalobos Toro, J. F. (2023). *iseño y puesta en marcha de una plataforma de control para el proceso de limpieza de la planta de cocimiento de la compañía de cervezas nacionales CA*.
- Y. LeCun, Y. B. (2015). *Deep learning*. Nature, vol. 521, no. 7553, pp. 436-444.
- Zhang, Y. L. (2018). *Color-pattern recognition based on fuzzy neural network and its applications to image segmentation*. IEEE Access, 6, 8169-8179.
- Zumba Ortiz, I. F. (2022). *Diseño de una red inteligente con alimentación fotovoltaica utilizando microcontroladores, comunicación Inalámbrica y una infraestructura IoT. Para monitoreo de variables de proceso en tableros eléctricos de diferentes áreas de una empresa de logística*. Doctoral dissertation, ESPOL. FIEC.

ANEXOS

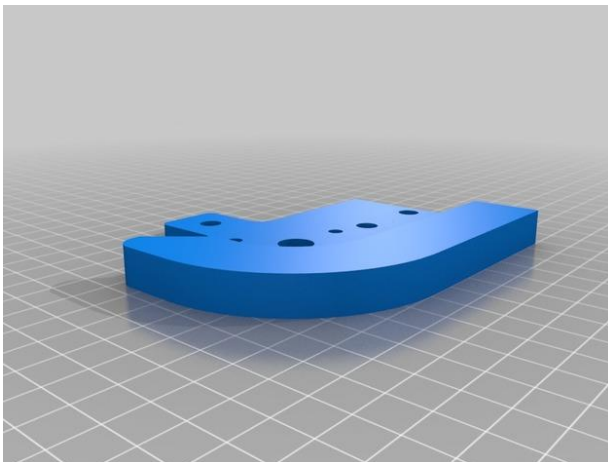
Anexo 1 : Retenedor de banda de transporte lateral derecho.



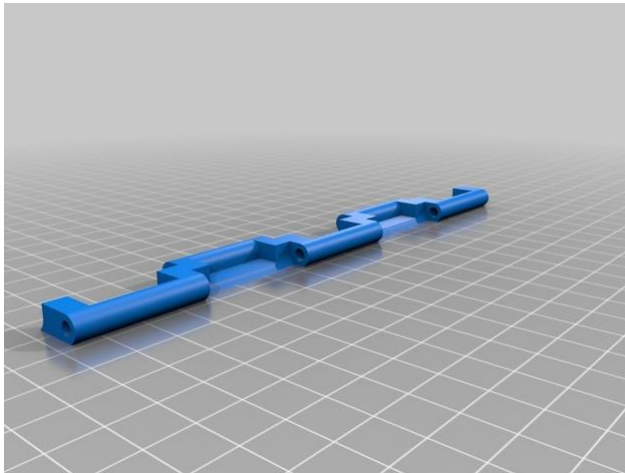
Anexo 2 : Espaciador de banda de transporte.



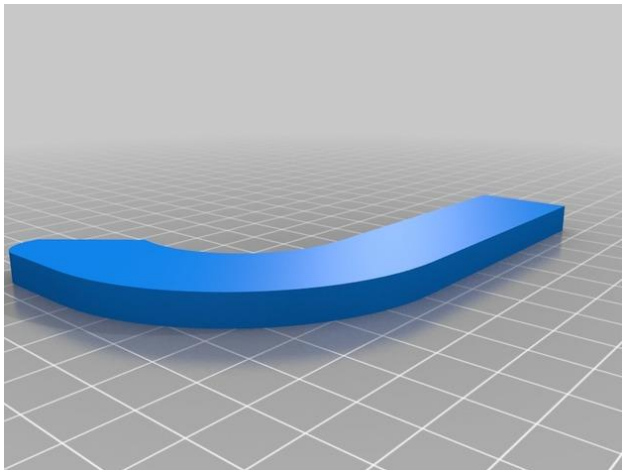
Anexo 1 : Lateral de banda de transporte.



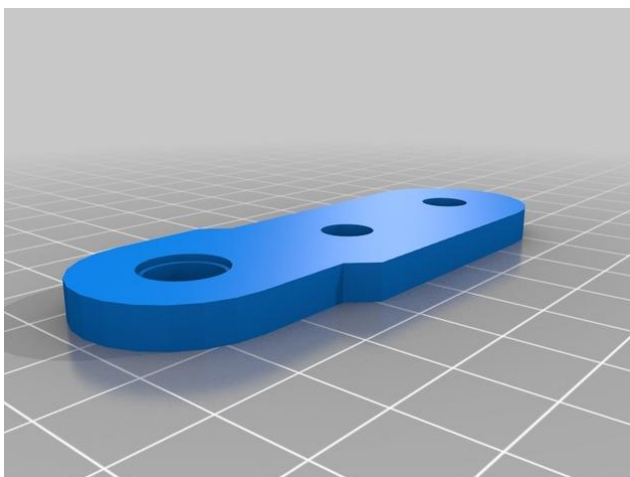
Anexo 2 : *Componente banda de transporte.*



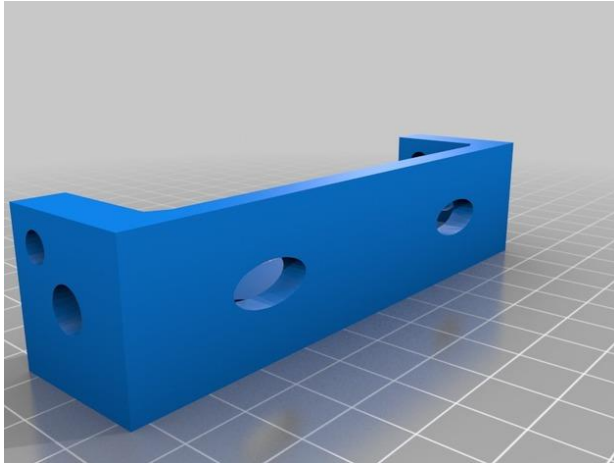
Anexo 5 : *Espaciador lateral banda de transporte.*



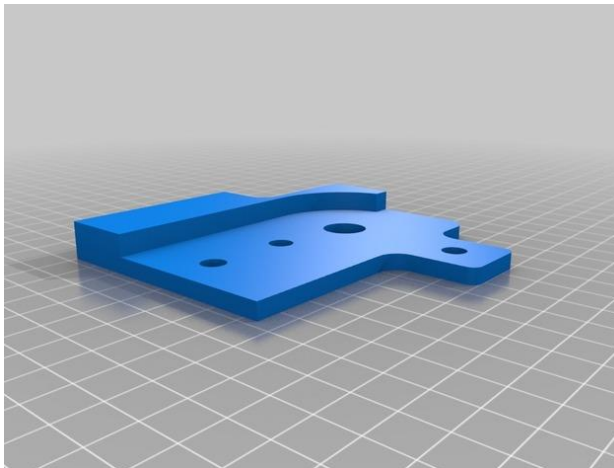
Anexo 3 : *Retenedor de banda de transporte lateral izquierdo.*



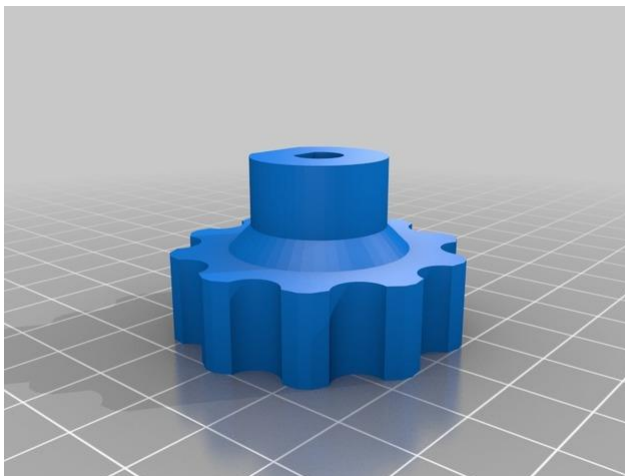
Anexo 4 : Cuadrante para distancia de banda transportadora.



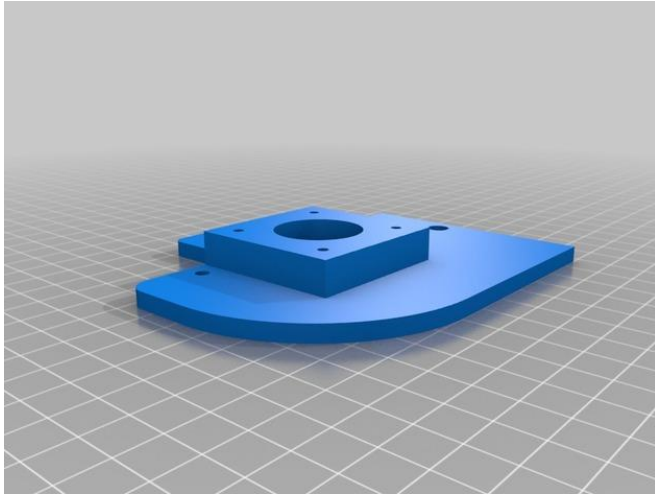
Anexo 5 : Lateral para distancia de banda de transporte.



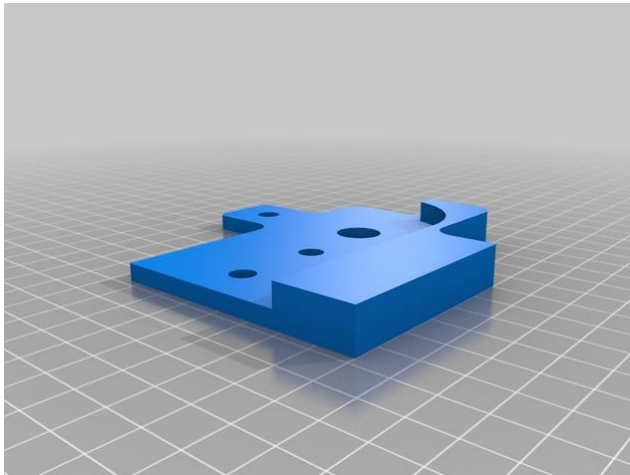
Anexo 6 : Rodamiento de banda de transporte.



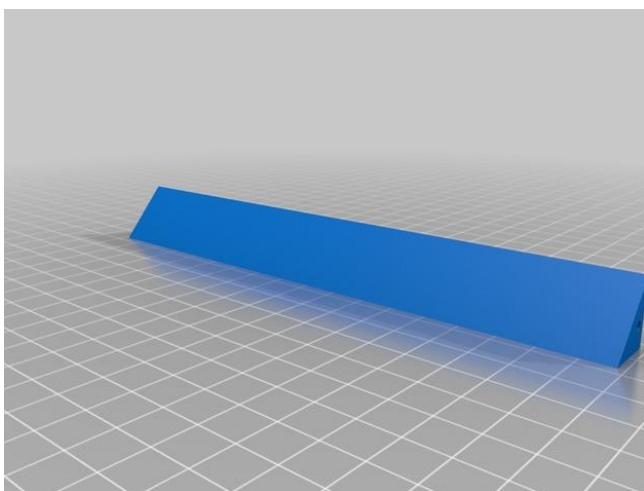
Anexo 7 : Lateral del motor de banda de transporte.



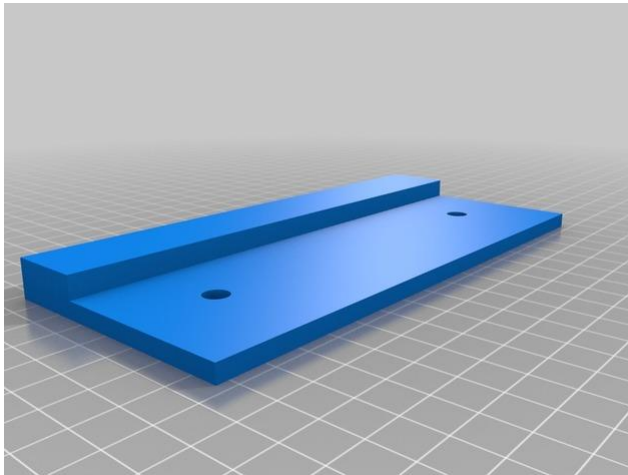
Anexo 8 : Parte lateral inferior del motor de banda transportadora.



Anexo 9 : Espaciador interno del motor de banda de transporte.



Anexo 10 : base lateral del motor de banda de transporte.



Anexo 14.- Resolución

RESOLUCIÓN DE PRÁCTICA

PRÁCTICA 1

NÚMERO DE ESTUDIANTES: 2

DOCENTE

ING. VÍCTOR DAVID LARCO TORRES MSC. MSc.

TIEMPO ESTIMADO: 1 HORAS

**TEMA: “CONTROL DE UNA BANDA TRANSPORTADORA
Y CONTEO DE PRODUCTOS”**

A. Objetivo General

- Controlar una banda transportadora

B. Objetivos Específicos

- Realizar la configuración de un autómata programable usando microcontrolador del sistema LattePanda con OPENPLC I
- Utilizar OPENPLC para la lectura de datos digitales mediante LattePanda
- Utilizar LattePanda para controlar una banda transportadora.
- Realizar el conteo de productos en un display de 7 segmentos

C. Marco Procedimental

1. Realizar la conexión del módulo a una alimentación de voltaje 110VAC .
2. Encere la LattePanda como se muestra en la Figura 50



Figura 50 : Botón de encendido

- 3.** El entorno en la LattePanda está basado Windows 10, como recomendación seguir los pasos del Anexo C donde se muestra la instalación del programa.
- 4.** Ejecutar el programa OPENPLC como se muestra en la Figura 51



Figura 51 : Ejecución de programa OPENPLC

- 5.** En el programa OPENPLC se procede a crear un nuevo proyecto como se muestra en la Figura 52.

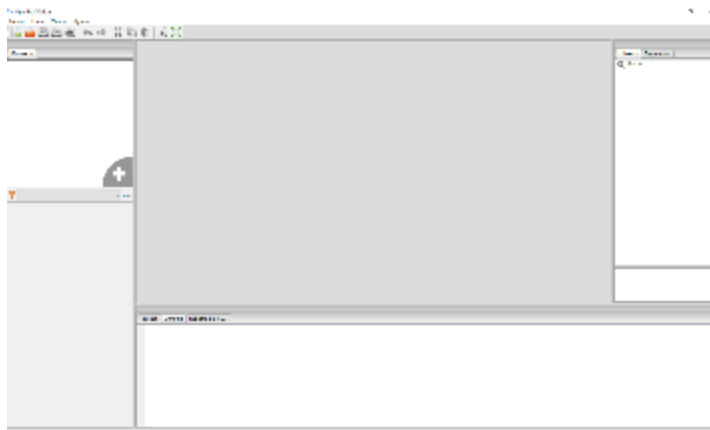


Figura 52 : Ejecución de programa OPENPLC

6. Al crear un nuevo programa se debe seleccionar el tipo de lenguaje a programar. En este caso se selecciona Ladder o LD como se muestra en la Figura 53.

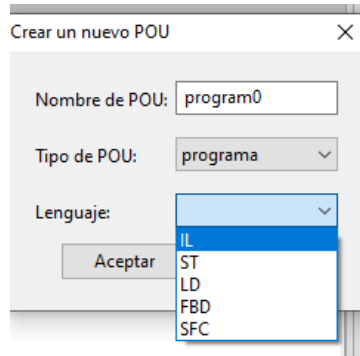


Figura 53 : Selección de lenguaje de programación en OPENPLC

7. Se enlaza los valores de las entradas y salidas a direcciones en el PLC de acuerdo con la tabla 5.

Descripción	Pin	Dirección
ENTRADA_1	D-9	%IX0.0
ENTRADA_2	D-10	%IX0.1
ENTRADA_3	D-11	%IX0.2
ENTRADA_4	D-12	%IX0.3
SENSOR	D-8	%IX0.4
SALIDA_1	D-2	%QX0.0
SALIDA_2	D-3	%QX0.1
SALIDA_3	D-4	%QX0.2
SALIDA_4	D-5	%QX0.3
DISPLAY_1	D-21	%QX0.4
DISPLAY_2	D-22	%QX0.5
DISPLAY_3	D-18	%QX0.6
DISPLAY_4	D-19	%QX0.7
INB_1	D-7	%QX1.0
INB_2	D-13	%QX1.1
Pwm_A	D-6	%QW0

Tabla 5: Tabla de direccionamiento del proyecto en OPENPLC Resolución

En la figura 54 se muestra la configuración de los pines en el módulo, y la configuración de los pines en el OPEN PLC como se muestra en la figura 55.



Figura 54 : Configuración de pines en el módulo.

#	Nombre	Clase	Tipo	Ubicación	Valor Inicial	Opción	Documentación
1	ENTRADA_1	Local	BOOL	%X0.0	0		
2	ENTRADA_2	Local	BOOL	%X0.1	0		
3	ENTRADA_3	Local	BOOL	%X0.2	0		
4	ENTRADA_4	Local	BOOL	%X0.3	0		
5	SENSOR	Local	BOOL	%X0.4	0		
6	SALIDA_1	Local	BOOL	%Q0.0	0		
7	SALIDA_2	Local	BOOL	%Q0.1	0		
8	SALIDA_3	Local	BOOL	%Q0.2	0		
9	SALIDA_4	Local	BOOL	%Q0.3	0		
10	DISPLAY_1	Local	BOOL	%Q0.4	0		
11	DISPLAY_2	Local	BOOL	%Q0.5	0		
12	DISPLAY_3	Local	BOOL	%Q0.6	0		
13	DISPLAY_4	Local	BOOL	%Q0.7	0		
14	INB_1	Local	BOOL	%QX1.0	0		
15	INB_2	Local	BOOL	%QX1.1	0		
16	PWM_A	Local	WORD	%QW0	0		

Figura 55: Configuración de pines en el OPENPLC

8. Se declara el contador con la instancia del sensor como se muestra en la Figura 56.

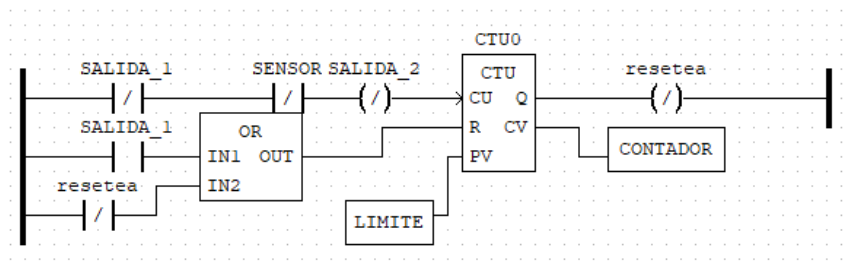


Figura 56 : Programa de detección de sensor y conteo de productos

9. Se declara el marcha y paro del sistema como se muestra en la Figura 57.

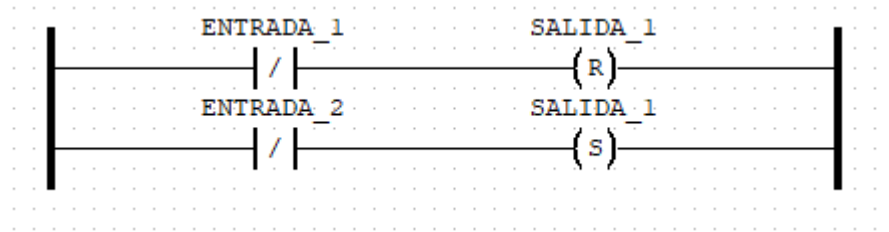


Figura 57: Programa de activación y paro del sistema

10. Declarar las instancias para activación del motor y control de giro como se muestra en la Figura 58.

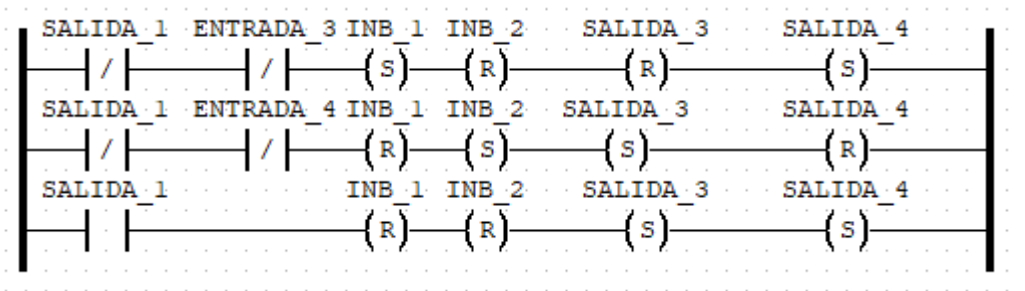


Figura 58: Control del sentido del motor

11. Declara el bloque de función para controlar la salida del controlador de motor mediante una señal PWM como se muestra en la Figura 59.

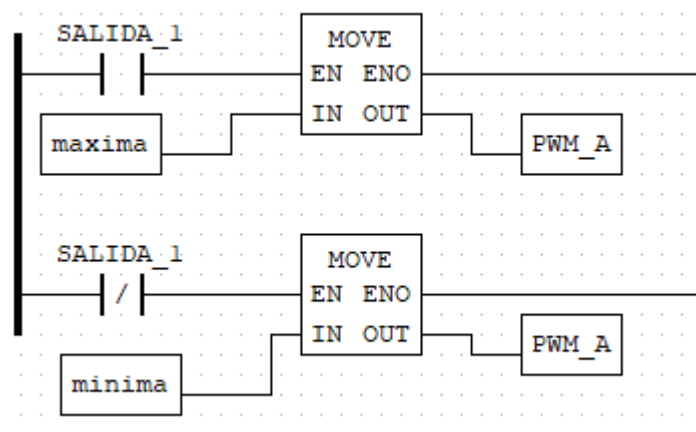


Figura 59: Control de la velocidad en Ladder en OPENPLC

12. Configurar la salida para el display de 7 segmentos tomando en consideración la variable de conteo como se muestra en la Figura 60.

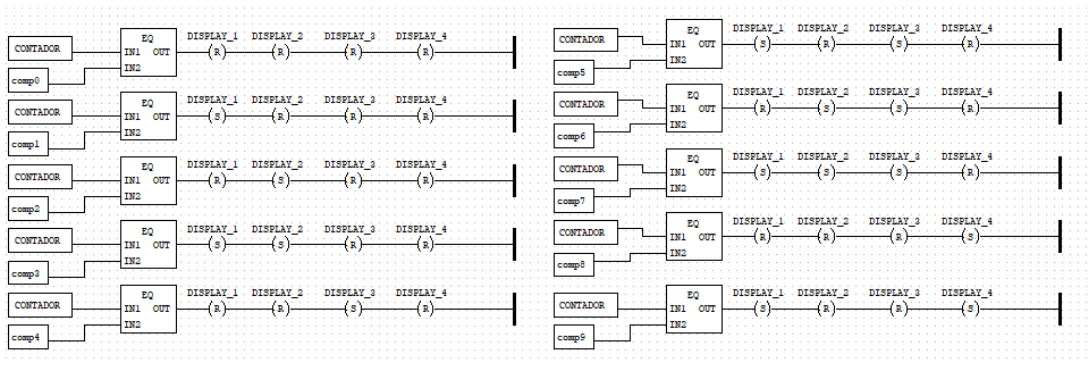


Figura 60 Control del display en Ladder en OPENPLC

13. Al finalizar se procede a cargar el programa como se muestra en la Figura 61.

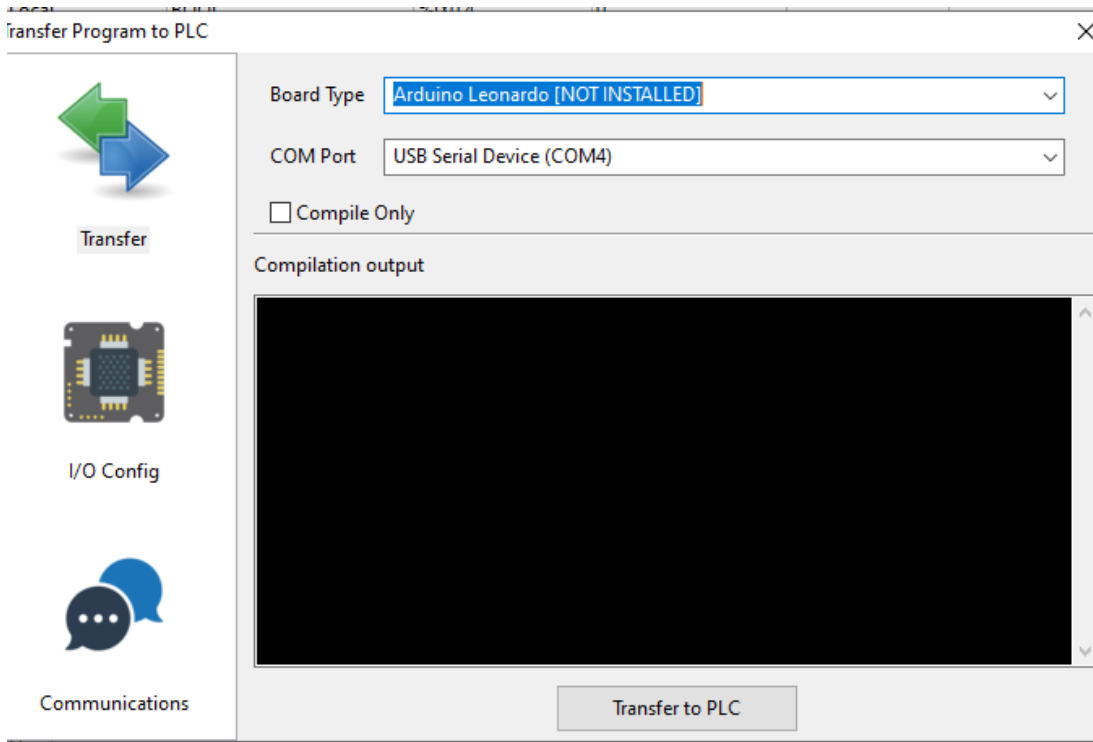


Figura 61: Compilación del programa