

Desarrollo de un controlador lógico programable de bajo costo basado en Raspberry PI.

Development of a low-cost programmable logic controller based on Raspberry Pi.

Raúl Moreno Rincón* (1).
Tecnológico Nacional de México/Instituto Tecnológico de Tuxtla Gutiérrez.
raull.mr@tuxtla.tecnm.mx.

Luis Antonio Orellana Águila (2). Estudiante del Tecnológico Nacional de México/ Instituto Tecnológico de Tuxtla Gutiérrez, L20270626@tuxtla.tecnm.mx.

Álvaro Hernández Sol (3). Tecnológico Nacional de México//Instituto Tecnológico de Tuxtla Gutiérrez,
alvaro.hs@tuxtla.tecnm.mx.

Osbaldo Ysaac García Ramos (4) Tecnológico Nacional de México//Instituto Tecnológico de Tuxtla Gutiérrez,
ogarcia@tuxtla.tecnm.mx.

Alexis de Jesús Flores García (5). Tecnológico Nacional de México/ Instituto Tecnológico de Tuxtla Gutiérrez,
alexis.fg@tuxtla.tecnm.mx.

Roberto Ibáñez Córdova (6) Tecnológico Nacional de México//Instituto Tecnológico de Tuxtla Gutiérrez,
roberto.ic@tuxtla.tecnm.mx.

*corresponding author.

Artículo recibido en noviembre 24, 2025; aceptado en diciembre 16, 2025.

Resumen.

El proyecto consiste en el diseño y construcción de un prototipo de controlador lógico programable (PLC) basado en una Raspberry PI que nos resulte a menor costo que los PLC's comerciales equivalentes en tamaño. Y además aproveche las características tecnológicas de la Raspberry PI la cual es una tarjeta con capacidad de manipular dispositivos de audio, video, memorias externas, etc. que la colocan cercana a un procesador muy por encima de los controladores. Durante el desarrollo de este proyecto se le dio al prototipo, las ventajas de construcción que presentan y potencializan a los PLC's comerciales como son; el diseño tipo modular que separan al CPU, A los módulos de Entrada/Salida y a la fuente de poder, así también se le instalo un software que proporcionara un acceso a su programación parecido a los que presentan los PLC's comerciales, el cual es amigable, simple y eficiente, para nuestro caso se podrá programar con el lenguaje denominado escalera, que es el mas utilizado actualmente y con una interface que simula a los que proporcionan los fabricantes de PLC.

Palabras claves: Diseño, modular, Raspberry, simular,

Abstract.

The project consists of the design and construction of a prototype programmable logic controller (PLC) based on a Raspberry Pi, which will be less expensive than equivalent-sized commercial PLCs. Furthermore, it leverages the technological characteristics of the Raspberry Pi, a board capable of handling audio and video devices, external storage, etc., placing it close to a processor and far superior to other controllers. During the development of this project, the prototype incorporates the construction advantages that commercial PLCs offer and enhance, such as a modular design that separates the CPU, the input/output modules, and the power supply. Software was also installed that provides programming access similar to that of commercial PLCs; this software is user-friendly, simple, and efficient. In our case, it can be programmed using ladder logic, the most widely used language today, with an interface that simulates those provided by PLC manufacturers.

Keywords: Design, modular, Raspberry Pi, simulate.

1. Introducción.

El sector tecnológico e industrial, con su marcado crecimiento a lo largo de los años, ha ido evolucionando en diferentes etapas hasta el punto actual, donde su siguiente migración se inclina a lo que se llama industria 4.0. La alta demanda de estos dispositivos, llamados PLC (por sus siglas en inglés) que son unidades de control de alto desempeño específicos al desarrollo industrial, esto conlleva a ser dispositivos de alta potencia y eficiencia, pero con un costo elevado. En este contexto, surge la necesidad de crear soluciones que permita al usuario tener una alternativa de un dispositivo (PLC) de costo más bajo, pero sin sacrificar la potencia y la eficiencia.

Esta investigación nos permitió adentrarnos en el desarrollo de un controlador lógico programable de bajo costo basado en Raspberry Pi y la plataforma CODESYS. La elección de la Raspberry Pi como plataforma central para el controlador lógico programable se fundamenta en su renombrada versatilidad y su adopción generalizada en proyectos de ingeniería y desarrollo. Complementando esta elección, la implementación del sistema CODESYS la cual añade una capa de sofisticación al permitir una programación avanzada y una gestión eficiente de los procesos industriales

Durante el desarrollo de esta investigación se exploró la reunión de estas tecnología y componentes que se fusionan en la creación de este PLC. Se analizaron los elementos claves, desde el diseño de las tarjetas de acoplamiento I/O, la configuración de la Raspberry Pi, la programación del sistema CODESYS y la implementación del prototipo demostrativo. A medida que avanzamos en el desarrollo de la investigación nos permitió caracterizar las limitaciones y alcances de esta propuesta, proporcionando una visión exhaustiva de la viabilidad y aplicabilidad de esta solución e incorporarnos al desarrollo en este sector en constante evolución.

2. Métodos.

Diseño, construcción e implementación de hardware del sistema.

Se optó por una Raspberry Pi 3B versión 2.0, por su bajo costo, portabilidad, disponibilidad de documentación y compatibilidad con software de automatización. Esta tarjeta posee un procesador ARM Cortex-A53 de cuatro núcleos, puertos USB, HDMI, ranura para tarjeta microSD, y lo más importante: 40 pines GPIO, que permiten manipular señales digitales equivalentes a las entradas y salidas típicas de un PLC industrial, Halfacree (2020), ver figura 1.

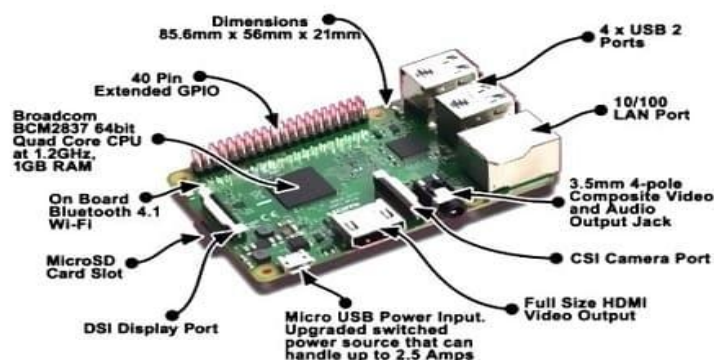


Figura 1. Raspberry PI 3B

Se determinó darle a nuestro PLC una estructura modular como la que presentan los PLC's comerciales, Salas (2017), por lo que se decidió diseñar y generar una Shell principal que permita conectarse de manera directa y fácil a la Raspberry y dar acceso a todos los recursos que esta tiene, así también se conecten las terminales de GPIO de la Raspberry a una serie de puertos de conexión para los módulos de entrada/salida (E/S) de nuestro PLC, dichos puertos serán de 8 E o de 8 S (GPIO) cada uno, ya que es el estándar que se utiliza en los PLC's comerciales, para nuestro caso solamente consideraremos un módulo de 8 entrada y un modulo de 8 salidas

Debido a que en el estándar industrial es la fuente de 24Vcd, Daneri (2008), este Shell recibirá a través de la tarjeta de entrada este valor de voltaje y por medio de 3 reguladores bajará la señal; uno para bajar este voltaje a 3.3Vcd que se requiere en el módulo de entrada, Otro para bajar el voltaje a 5Vcd que se requiere en el módulo de salida y un tercero para bajar el voltaje a 5Vcd para el circuito del propio Shell y para la Raspberry. Por lo que, como los módulos de E/S van a acoplar las señales ya sea leídas de la planta para nuestro PLC o señales de modificación que envíe nuestro PLC a la planta, necesitaran hacer el cambio de valor de voltaje y por lo tanto se le va a enviar al módulo de entrada la señal de 3.3Vcd para comunicarse con la Raspberry ya que el voltaje de 24Vcd para la comunicarse con la Planta lo recibe directamente de ella y él se lo envía al Shell principal, el módulo de salida solamente se le enviara la señal de 5Vcd.

Para el módulo de entrada, considerando que tienen optoacopladores en cada entrada, utilizara la fuente de 24Vcd antes del optoacoplador que es la parte del circuito que recibe la señal de la planta y la de 3.3Vcd después del optoacoplador que es la parte del circuito que envía señal a la Raspberry.

En el caso del módulo de salida que tiene acoplador por relevador, solamente requiere la señal de 5Vcd para activar el solenoide del relevador, ya que el cierre de contactos se encargara de conectar la fuente de 24Vcd presentes en la planta.

Instalación y Configuración del Sistema Operativo en Raspberry PI.

Para la instalación del sistema operativo, primeramente, se necesita una tarjeta microSD ya que ahí es almacenada. Para el proyecto se tomó la decisión de instalar Raspberry Pi OS, y la forma más simple de obtenerlo es mediante



Figura 2. Raspberry Pi Imager, software oficial de Raspberry Pi para la instalación de sistemas operativos en tarjetas microSD.

Se puede descargar el software desde la web oficial de Raspberry Pi, donde se encuentra disponible diferentes versiones como Windows y MacOS. Desde la interfaz se puede seleccionar el sistema operativo dando clic en CHOOSE OS, ver figura 2. Existen diferentes sistemas operativos que se pueden instalar en una tarjeta microSD, entre los cuales se muestran las versiones de Raspberry Pi OS, Emulation and game o Media player OS, León (2021).

El siguiente paso es conectar la microSD a la computadora, y seleccionar que sistema operativo y seleccionar nuestra microSD en el cual será insertado el sistema operativo. Luego el software Raspberry Pi Imager descarga el sistema operativo, lo copia a la tarjeta microSD y verifica que la instalación se haya completado con éxito. Posteriormente se nos indica que se debe remover la microSD e insertarla en la Raspberry Pi. A continuación, se enciende la Raspberry Pi y se hace la configuración previa en el cual se pone usuario y contraseña para el acceso. Al estar dentro de la interfaz de Raspberry Pi OS, se procede a configurar la dirección IP estática y el SSH el cual serviría para comunicación entre CODESYS y Raspberry Pi.

3. Desarrollo.

Diseño, construcción e implementación de hardware del sistema.

Para el diseño de las PCB'S se necesitó tener previamente todos los diseños de los circuitos para así poder realizar de manera más fácil un esquemático. Como se definió que el PLC será de tipo modular y contara con 3 módulos de los cuales son entradas, Shell principal y salidas. En este caso en el esquemático se refleja la topología de la tarjeta, estableciendo conexiones eléctricas entre todos los componentes a emplear. El diseño implicó utilizar un software especializado para trasladar el esquemático a una disposición física, en este caso se utilizó la plataforma EasyEDA.

Como primer paso fue realizar el esquemático del módulo de entrada, en la figura 3, se puede observar que cuenta con sus borneras de alimentación para el funcionamiento general del PLC de cual es de 24VDC y con sus borneras de entrada para los componentes que funcionan en ese voltaje. También cuenta con su protección eléctrica, basada por optoacopladores para que posteriormente lleguen las señales a los GPIO correspondientes de la Raspberry Pi y estas no sufran algún daño ya que funcionan a 3.3VDC. En la figura 3 se muestra el esquemático del módulo de entradas.

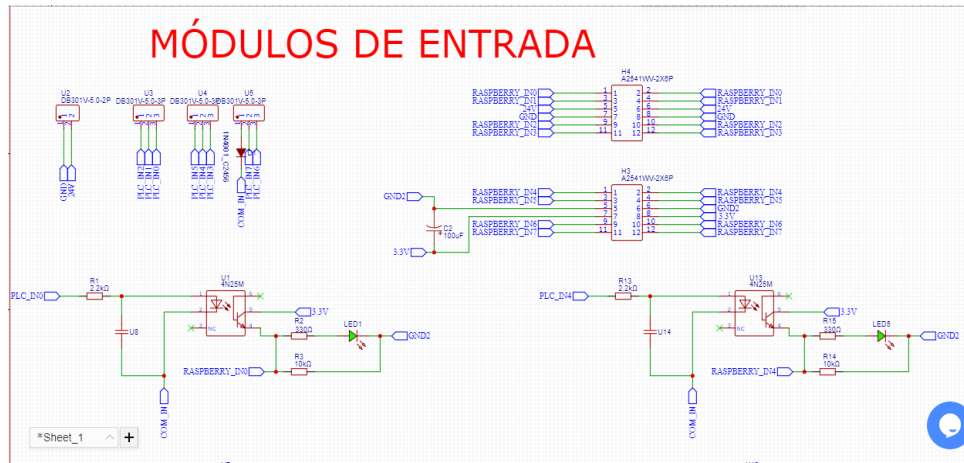


Figura 3. Esquemático del módulo de entrada.

En la figura 4 se puede observar el diseño final de la PCB, es decir, como quedaron acomodados los componentes, viendo que en la parte superior quedaron las borneras donde va la alimentación del PLC y las entradas para las señales de entrada y los componentes electrónico que conforma cada entrada, cabe mencionar que los pines en la parte inferior son los encargados de conectarse con la PCB del Shell principal. En esta imagen se muestra cómo se ve la PCB en 3D y tener una idea de cómo se verá físicamente, Halfacree (2020)

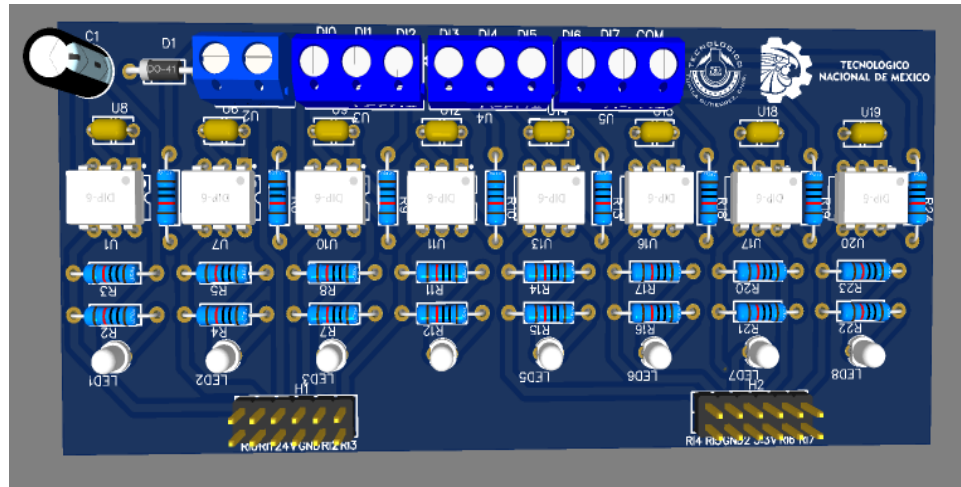


Figura 4. Diseño en 3D de la PCB del módulo de entrada.

De la misma forma, para la placa base del Shell principal, en la imagen 5 se muestran el circuito que lo conforma y podemos observar que están presentes los reguladores de voltaje, encargados de regular el voltaje de la fuente de alimentación de 24VDC que recibe a través de la tarjeta de entrada y alimenta a todos los componentes referenciándose al voltaje general del PLC (red de tierra) y así obtener un buen funcionamiento, de igual manera se observa en que forma será insertada la Raspberry Pi, cabe mencionar que será fácil montaje y desmontaje mediante sus pines haciéndolo una forma modular, figura 5.

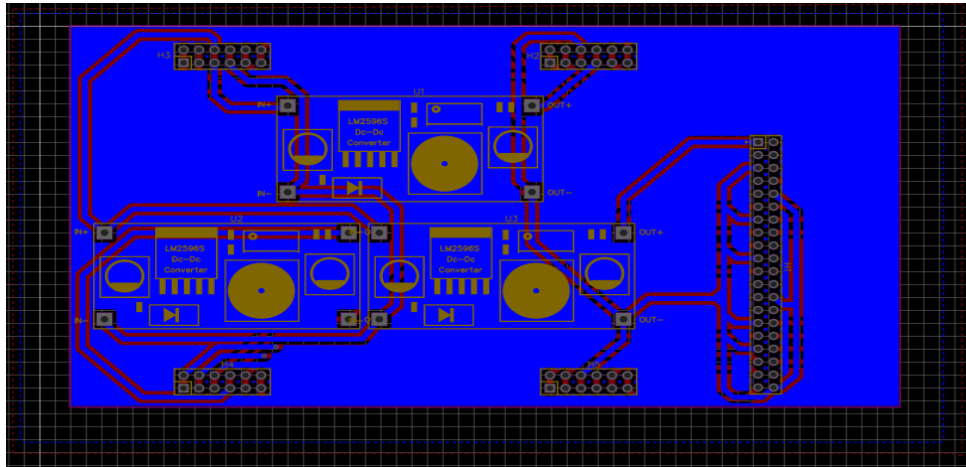


Figura 5. Esquemático del Shell principal.

En la figura 6 se puede tener el diseño final de la PCB, es decir, como quedaron acomodados los componentes, viendo que en la parte superior quedaron las borneras donde se va a insertar la tarjeta de entrada y en la parte inferior las borneras de la tarjeta de salida, también se observan los componentes electrónicos que conforma esta tarjeta, destacando los tres reguladores de voltaje antes mencionados. En esta imagen se muestra cómo se ve la PCB en 3D y tener una idea de cómo se verá físicamente

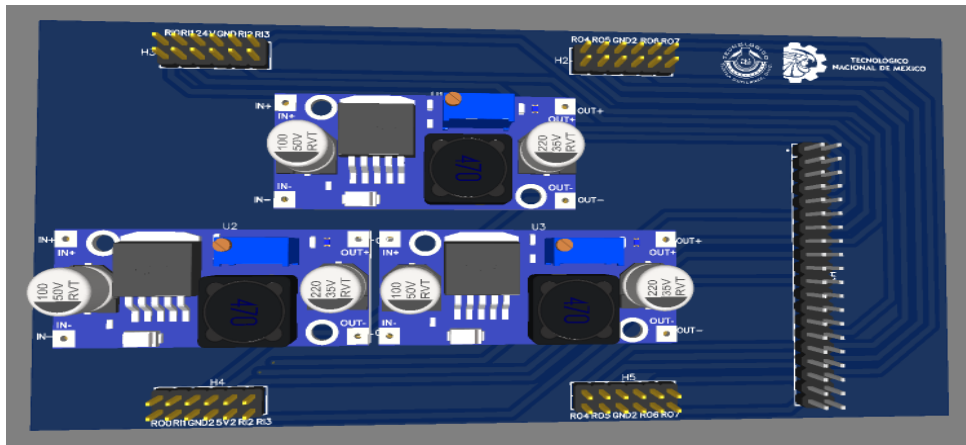


Figura 6. Diseño en 3D de la PCB del Shell principal.

Finalmente se realizó la PCB de la etapa de salidas, en la figura 7 se puede observar de qué manera quedó, en la parte inferior quedaron los pines de conexión los cuales serán conectados a la PCB de la Shell principal, también se muestra de qué forma se acomodaron los componentes electrónicos y principalmente las salidas a relevadores. En la parte inferior se muestra las borneras las cuales serán para conectar actuadores o indicadores.



Figura 9. Distribución de las GPIO de la Raspberry PI con respecto a los módulos de E/S del PLC.

Tabla 1. Lista de componentes utilizados.

Módulo de Entrada		Shell Principal		Módulo de Salida	
Cantidad	Descripción	Cantidad	Descripción	Cantidad	Descripción
8	Resistencia de 2.2Kohms	3	Regulador de voltaje LM2596	8	Resistencia de 1Kohms
8	Capacitor de 10nF			8	Transistor 2N2222
8	Optoacopladores 4N25	2	Pin header hembra	8	Resistencia de 2.2 Kohms
8	Base para integrado 6 pin	1	Raspberry PI Model 4		
8	Resistencia de 330ohms			8	Led verde 3mm
8	Resistencia de 10Kohms			8	Diodo 1N4001
8	Led verde 3mm			1	Bornera 2 pines
1	Diodo 1N4001			3	Bornera 3 pines
1	Bornera 2 pines			1	Pin header macho
3	Bornera 3 pines			2	Módulo de relevadores
2	Capacitor electrolítico 100uF				
1	Pin header macho				

Instalación y configuración de la plataforma CODESYS en la Raspberry PI.

Antes de empezar con la instalación del programa CODESYS se debe cambiar la IP de nuestra computadora desde donde vamos a realizar la programación de CODESYS, ver figura 10.

- Para esto se debe tener encendida la Raspberry Pi conectada mediante un cable ethernet hacia la computadora,
- Posteriormente se va al apartado de conexiones de red y ahí seleccionar nuestro adaptador ethernet. Al momento de darle siguiente saldrá un apartado de propiedades de ethernet
- Ahí debemos buscar el protocolo de internet versión 4 (TCP/IPv4).

- Al realizar esto se configuro la IP, la máscara de subred y la puerta de enlace predeterminada
- En este caso en la Figura 10 se muestra la configuración propuesta.
- Al terminar esto se debe entrar al símbolo del sistema y verificar si hay comunicación con la Raspberry Pi.

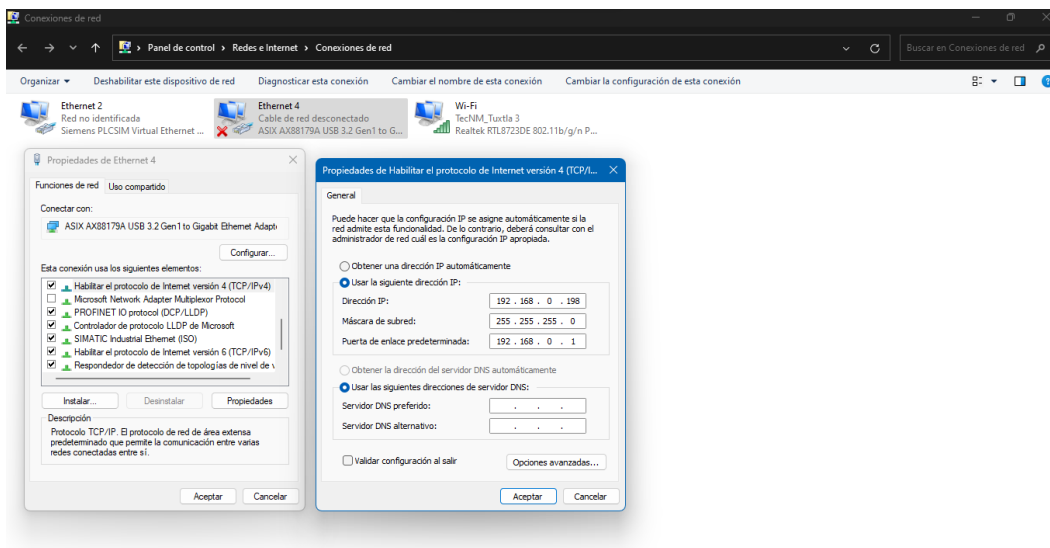


Figura 10. Configuración IP de la computadora.

Ahora si descargamos del sitio oficial al programa CODESYS y sus correspondientes sistemas SoftPLC de acuerdo con la versión determinada y los instalamos. También se descarga e instala desde la misma página la paquetería Control for Raspberry PI SL.

Al terminar la instalación, abrimos el apartado de administración de paquetería en CODESYS e insertamos las librerías deseadas, ver figura 11.

Y por último Solicitamos en el apartado de herramientas “actualizar Raspberry PI” buscamos a nuestra Raspberry y al encontrarla nos muestra su dirección IP lo cual nos permite saber que si se conectaron y continua la instalación.

Una vez configurado el entorno, ya podemos realizar algún programa para transferirlo a nuestro PLC lo cual se realiza en el programa CODESYS y respetando las direcciones que se conectaron como entradas y salidas en la Raspberry, se realizó una prueba con un programa de funcionamiento de un semáforo, ver figura 12.

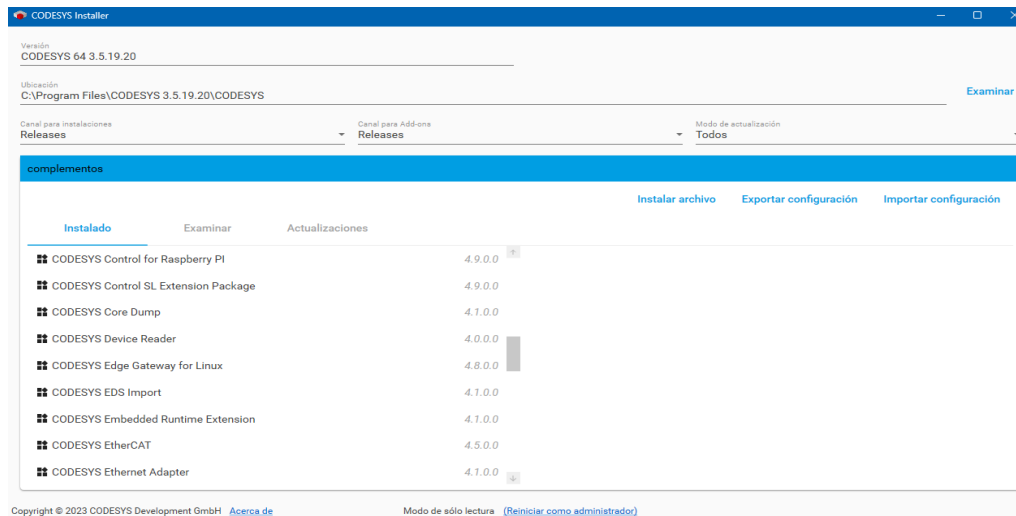


Figura 11. Instalación de la Paquetería de Raspberry Pi en CODESYS.

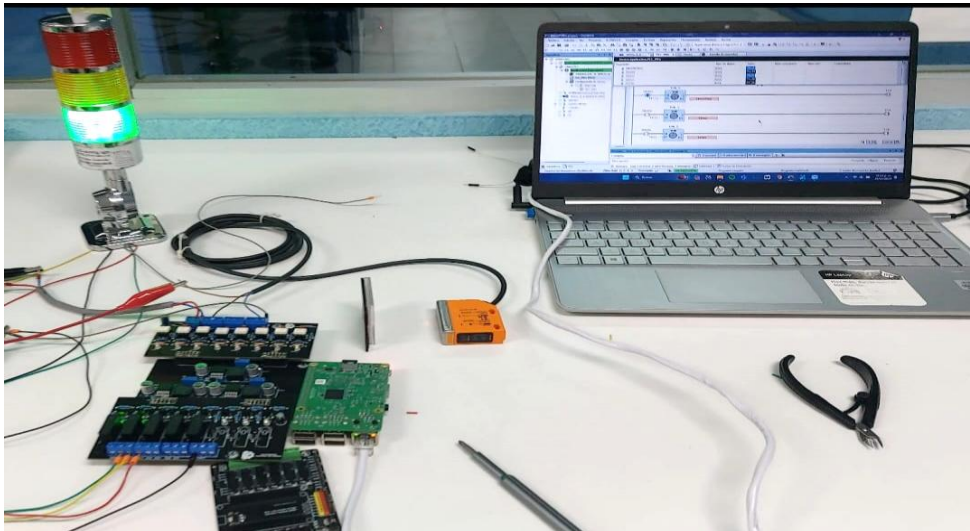


Figura 12. Prueba de funcionamiento, implementando un programa con CODESYS.

Ajustes finales.

Se diseñó una carcasa protectora del PLC que abrazara a las 3 placas que conforman nuestro PLC, pero dejando descubiertas las borneras de conexión de los recursos de la Raspberry PI, así como las borneras de conexión de las tarjetas de entrada y salida en donde se conectarán los dispositivos externos del sistema a controlar. La carcasa permite observar los leds de las tarjetas de entrada y salida y la ventilación del PLC.

Resultados:

Para probar el funcionamiento de nuestro PLC y validarlo se necesitó realizar varias pruebas. Para esto se diseñó un kit de entrenamiento para prácticas de automatización en el cual se incorporó a nuestro PLC. Este kit cuenta con diversos componentes usados en la industria 4.0, como son 2 sensores de la marca IFM uno es de tipo inductivo y el otro de tipo fotoeléctrico los cuales operan a 24Vcd. Cuenta con 3 pistones neumáticos de la marca SMC el cual tienen sus electroválvulas 5/2 de la marca Festo que operan a 24Vcd. También el kit cuenta con una botonera de arranque y

paro, tiene un juego de indicadores en este caso es una torreta la cual consta de 3 diferentes colores y un buzzer. Este kit es alimentado a través de una fuente de 24Vcd, Sanchis (2010). Ver figura 13.

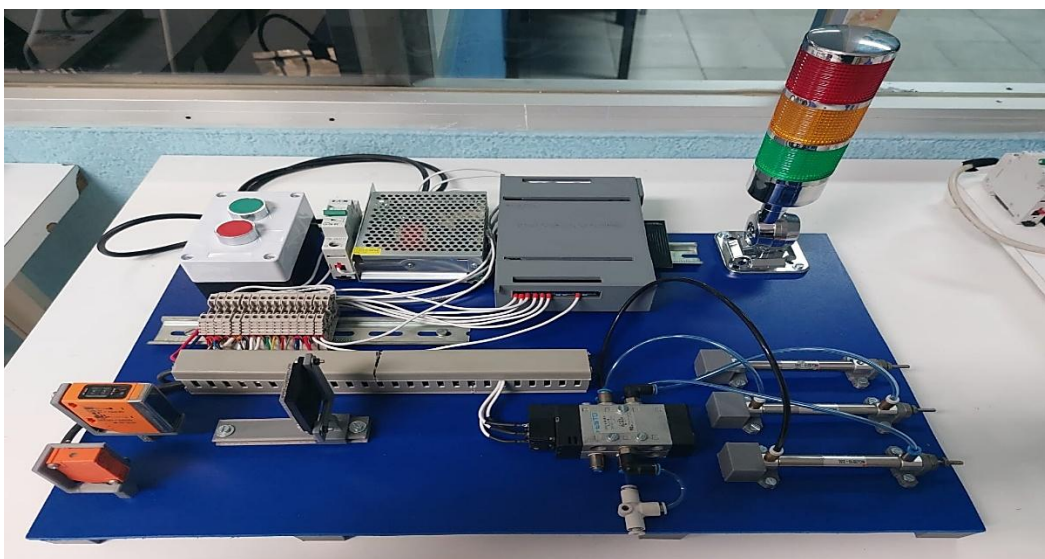


Figura 13. Kit de Entrenamiento Usando el PLC Basado en Raspberry Pi.

Una de las primeras pruebas fue la evaluación rigurosa de la complementación de hardware y software. La Raspberry Pi demostró ser una plataforma sólida y capaz de manejar las tareas asignadas con eficacia, mientras que CODESYS proporciono un entorno de desarrollo flexible y potente.

Se verifico el rendimiento del controlador mediante pruebas de respuesta en tiempo real y análisis de carga, los resultados destacaron un tiempo de respuesta dentro de los parámetros aceptables para entradas y salidas, cumpliendo con los requisitos estándar para la industria. Además, se observó una capacidad de procesamiento adecuada para manejar múltiples tareas simultaneas, esto lo hace un sistema para aplicaciones industriales de mayor complejidad.

También se llevó a cabo un análisis de estabilidad a largo plazo y la consistencia del controlador en condiciones de funcionamiento continuo. Durante el periodo de prueba extendido, no se observaron fallos significativos ni degradación del rendimiento, respaldando la fiabilidad del sistema en entornos industriales exigentes, Barrera (2004).

En una comparación con un PLC comercial equivalente, se determino que nuestro PLC que debido a los recursos que presenta la Raspberry PI presenta mayor accesibilidad, versatilidad y flexibilidad, mayor adaptabilidad al entorno industrial, alta gama de herramientas de programación y estabilidad.

Conclusiones.

El desarrollo de este proyecto ha alcanzado su objetivo al lograr un controlador lógico programable que aprovecha la potencia de una Raspberry Pi y la eficacia del sistema CODESYS. Este PLC representa un avance en el mundo de la automatización industrial 4.0 aplicado a un enfoque económico.

Al unir la flexibilidad de la Raspberry Pi y la solidez operativa de CODESYS ha resultado en un sistema de control versátil y eficiente, el cual es capaz de desempeñarse a buen nivel en distintos entornos industriales. El uso de un hardware costeable y un software eficiente, no solo se ha logrado una reducción de costos, sino que se ha creado una solución que se adapta de manera óptima.

Esto se traduce en un recurso valioso para empresas de menor nivel que buscan optimizar sus operaciones. Este proyecto posibilita el comienzo de oportunidad en la que la tecnología se aplique como un facilitador accesible y poderoso para la mejora los procesos industriales en pequeñas empresas o en comunidades rurales que quieran automatizar las soluciones de sus necesidades.

Créditos.

Los autores agradecen al Tecnológico Nacional de México por el financiamiento del proyecto a través de la convocatoria de Proyecto de Investigación Científica, Desarrollo Tecnológico e Innovación 2025.

Referencias bibliográficas.

- Barrera, R., Cartagena, J. A., & Pocasangre, C. O.** (2004.). *Diseño y construcción de un controlador lógico Programable PLC a base de un microcontrolador*. <https://dspace.unl.edu.ec/jspui/handle/123456789/18186>
- Daneri, P. A.** (2008). *PLC Automatización y Control Industrial*. https://www.academia.edu/47770770/Kupdf_net_plc_automatizacion_y_control_industrial_pablo_daneri
- Halfacree, G.** (2020). *LA GUÍA OFICIAL DE Raspberry Pi para principiantes*. McLibre.org. <https://www.mclibre.org/descargar/docs/revistas/magpi-books/raspberry-pi-beginners-book-4-es-202011.pdf>
- León, J. D.** (2021). *Diseño e implementación de un PLC basado en Raspberry Pi & Node-RED para aplicaciones industriales IOTT*. <https://repositorio.urp.edu.pe/handle/20.500.14138/4861>
- Salas, G. E.** (2017). *HISTORIA DE LOS INICIOS DEL PLC*. https://www.academia.edu/34381291/historia_del_plc
- Sanchis, R., Romero, J. A., & Ariño Latorre, C. V.** (2010). *Automatización industrial*. www.academia.edu. https://www.academia.edu/39128729/Automatizaci%C3%B3n_industrial

Información de los autores.



Raúl Moreno Rincón, Ingeniero en Comunicaciones y Electrónica, egresado de la ESIME-IPN, en la ciudad de México, D.F. Maestro en Ciencias en Ingeniería Electrónica egresado del Instituto Tecnológico de Toluca. Maestro en Educación Superior por la Universidad Autónoma de Chiapas. Certificado en SolidWorks Associate. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez y jefe de Proyectos de Docencia de Ingeniería Electrónica. Es investigador desde 1999 y Colabora en la línea de investigación “Robótica” de Ingeniería electrónica y es parte del cuerpo académico en formación “Sistemas de control inteligentes”. Ha realizado proyectos como: Sistema de alarma para personas con deficiencia auditiva basado en XBEE, robot de cafetería, araña hexápoda, sistema de control de animatronic, entre otros.



Luis Antonio Orellana Águila, Estudiante de Ingeniería Electrónica en el Tecnológico Nacional de México, campus Tuxtla Gutiérrez, participo en el proyecto de investigación nacional denominado Diseño de un Controlador Lógico Programable de bajo costo basado en Raspberry PI y el Sistema Codesys para aplicaciones en tratamiento de agua durante su residencia, a participado también en proyectos de fin de materia que se han expuestos en el evento académico semestral denominado Exproyecca en el TecNM-ITTG



Álvaro Hernández Sol, es Ingeniero en electrónica, egresado del Instituto Tecnológico de Tuxtla Gutiérrez ITTG, en Tuxtla Gutiérrez, Chiapas. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez y es investigador desde 1997. Certificado en SolidWorks Associate. Jefe de proyectos de investigación de ingeniería electrónica desde el 2001. Fundador y asesor del “Club de robótica del ITTG”. Colabora en la línea de investigación de “Robótica”



Osbaldo Ysaac García Ramos, Maestro en Ciencias en Ingeniería Mecatrónica egresado del Instituto Tecnológico de Tuxtla Gutiérrez. Académico de Tiempo Completo en UVM desde 2005. Tiene 12 años de experiencia docente. Tiene Cursos especialización en Mecatrónica en Universidad de Esslingen Alemania, en la empresa Emco Salzburgo Austria, en la empresa Festo Estados Unidos y Festo México. Responsable técnico del proyecto de investigación “Consolidar un centro de investigación y desarrollo de vehículos eléctricos funcionales y confortables e impulsados por energía limpia” junto con la empresa Invemex S.A. de C.V., 2015. Ha desarrollado proyectos de investigación como ‘Sistema de medición de hélices de barco para la secretaria de marina financiado por Conacyt y Semar’, ‘Diseño y caracterización de Biomateriales a partir de macromicetos como alternativa al uso del unicel’ en el año 2014 (financiados por Conacyt). Tiene participaciones en congresos internacionales como congreso Tecno láser Habana Cuba 2013 con artículo y ponencia ‘Método de registro automático de imágenes de rango tridimensionales para restauración de piezas arqueológicas’.



Alexis de Jesús Flores García es Ingeniero en Electrónica por el Instituto Tecnológico de Tuxtla Gutiérrez (ITTG, 2010) y Maestro en Ciencias en Optomecatrónica por el Centro de Investigaciones en Óptica A.C. Se desempeña como Docente en el Departamento de Ingeniería Eléctrica y Electrónica del ITTG, con 11 años de experiencia docente. Posee certificaciones CSWA, CSWP y en Manufactura Aditiva por SolidWorks, además de un diplomado en Microsoft Teams. Es profesor con perfil deseable PRODEP, colaborador del cuerpo académico “Sistemas de Control Inteligentes”, presidente de la Academia de Ingeniería Electrónica y miembro de la Academia de Ingeniería Mecatrónica.



Roberto Ibáñez Córdova, Ingeniero Industrial en Eléctrica por el I.T. Tuxtla Gutiérrez, Maestro en Ciencias en Ingeniería Electrónica por el Instituto Tecnológico de Toluca, Doctor en Ingeniería Aplicada por el Colegio de Formación Educativa TENAM en Tuxtla Gutiérrez. Áreas de interés son: Automatización de procesos mediante el desarrollo de Sistemas embebidos Inteligentes (Hardware-Software), Sistemas de Control remoto vía WEB.