

Control and supervision for a bottling process prototype under the IEC 61131-3 standard

Braulio Raygoza-Pérez* Hoover Mujica-Ortega**

* Facultad de Ingeniería - UNAM, Departamento de Control y Robótica, Lab. Automatización, C.U., 04510, México CDMX e-mail: braulio.raygoza.perez@gmail.com

** Facultad de Ingeniería - UNAM, Departamento de Control y Robótica, Lab. Automatización, C.U., 04510, México CDMX e-mail: hmujica@unam.mx

Resumen:

In this article, the implementation of a control and supervision system for the recycling module in a prototype model, which represents an industrial bottling process, is presented. This module includes a robotic arm, conveyor belts, a rail with a transport cart, sensors, DC motors, and pistons. The recycling process uses the robot to move boxes with bottles to a conveyor belt, which takes them to the emptying zone and then to the recycling zone, where the robot separates the bottles from the packaging to reintegrate each element into its respective stage of the process. As part of the development, a motion control system was designed for the conveyor belt motors to reduce mechanical wear, optimize the energy consumption of the motors as well as reduce abrupt movements that could affect product quality. A monitoring system for the electrical variables of the system was also implemented, providing useful information for decision-making that contributes to increasing the process efficiency. For this, the IEC 61131 standard, the GRAFCET methodology, and new IoT technologies, such as View of Things, were used. After the implementation, the module was successfully automated and optimized, with monitoring and management from mobile devices, validating the effectiveness of the control system for current large-scale industrial environment.

Palabras clave: Control secuencial, Industria 4.0, PLC, HMI web.

1. INTRODUCCIÓN

La automatización industrial se ha desarrollado para mejorar la eficiencia de los procesos, la calidad de los productos, reducir costos y evitar riesgos para los trabajadores. La automatización de procesos industriales es uno de los principales objetivos de las empresas que buscan ser cada vez más competitivas en un entorno cambiante y agresivo (García Moreno, 2014). Es aquí donde surge la necesidad de tener un sistema de control y supervisión que permita garantizar el óptimo control y desempeño de un proceso.

La introducción del PLC (Programmable Logic Controller) transformó la automatización industrial, mejorando la productividad y siendo ampliamente utilizado en la industria (Ramanathan, 2014). La norma IEC 61131 de 1993 estandarizó la programación de los PLC proporcionando un lenguaje común e independiente del fabricante. Dentro de la norma destaca su tercera parte, la IEC 61131-3, donde se especifica la sintaxis y semántica de los 5 lenguajes de programación que la conforman (IEC, 2003).

El control secuencial, como lo explica (García Moreno, 2014), consiste en un sistema cuyo funcionamiento emplea una secuencia de fases claramente diferenciadas con condiciones de transición lógicas que permiten efectuar cambios de estado en el sistema controlado.

Según este paradigma, la norma IEC 60848 describe el uso de un lenguaje gráfico llamado GRAFCET (Graphe Fonctionnel de Commande Étape/Transition). Este lenguaje se emplea para el diseño y modelado de sistemas de automatización, control secuencial y procesos industriales, <https://doi.org/10.58571/CNCA.AMCA.2024.097>

permitiendo representar de manera secuencial el comportamiento del sistema a través de etapas y transiciones. La estructura gráfica de GRAFCET facilita una visualización clara de las relaciones entre las distintas etapas del proceso y las condiciones bajo las cuales se producen las transiciones entre ellas. (IEC, 2013)

Específicamente, como describe (Ramanathan, 2014), SFC (Sequential Function Chart) es un lenguaje gráfico de alto nivel con mucha similitud al estándar francés GRAFCET, el cual representa los principales estados de un proceso, el cambio de estados y la razón detrás del mismo. Las principales ventajas de este lenguaje gráfico son que proporciona una representación clara y sencilla de procesos secuenciales complejos, así como la capacidad de ser integrado a cualquiera de los otros lenguajes de programación del estándar IEC 61131-3 (Castellanos and Solaque, 2010).

Actualmente las industrias se encuentran con grandes retos debido al sistema de producción complejo y cambiante, en respuesta a la creciente necesidad de productos innovadores y personalizados. Las empresas requieren una excelente conectividad y congruencia con sus procesos para resolver estos problemas de manera efectiva y esto se logra con la implementación de la digitalización y la industria 4.0 (Hassan Reza et al., 2021). A su vez esto implica la implementación de tecnologías emergentes que permitan la reestructuración de la industria convencional para tener acceso a información en tiempo real acerca de los procesos de producción y logística.

Con el crecimiento económico global y la búsqueda constante de eficiencia han impulsado a los países más industrializados a promover la producción de bienes cerca de los mercados de destino. Este enfoque ha generado la adopción de estrategias como la sustitución de importaciones y la deslocalización como medios efectivos para reducir costos, adaptarse a demandas locales y fortalecer economías (Quiroz Cuenca, 2021).

Actualmente, México vive un momento de oportunidad para su desarrollo industrial, pues la integración comercial con América del Norte a través del *nearshoring* de las cadenas de producción hoy es una realidad que permitirá la diversificación industrial, innovación, formación de personal, desarrollo de tecnología y apoyo a pequeñas y medianas empresas, dada su relevancia en nuestra economía. En general, el *nearshoring* en nuestro país implica la oportunidad de impulsar la exportación, la inversión productiva, la generación de empleos y la mejora de los niveles de vida en la población (Hacienda, 2023).

2. CELDA DE MANUFACTURA AUTOMATIZADA

Una celda de manufactura robotizada es un sistema automatizado que utiliza robots industriales para realizar tareas específicas en un área de trabajo definida. Estas celdas mejoran la eficiencia, precisión y consistencia en los procesos de manufactura. La automatización se puede organizar en cinco niveles:

1. Nivel de dispositivos: manejo de sensores y actuadores.
2. Nivel de máquina: coordinación de varios dispositivos.
3. Nivel de sistema o celda: organización y operación de una celda de manufactura, que incluye líneas de producción y ensamblaje.
4. Nivel de planta: generación de planes operacionales como control de inventarios y calidad.
5. Nivel de empresa: gestión de información corporativa relacionada con mercadeo, ventas, contabilidad, etc.

La automatización de una celda de manufactura robotizada se clasifica en el tercer nivel, que implica la coordinación de máquinas y/o unidades entre las cuales se trasladan piezas para producir productos terminados. El control de cada unidad corresponde al nivel de máquina, y la operación de cada actuador se sitúa en el nivel de dispositivo (Sánchez-Chávez and Martell-Chávez, 2019).

3. PLATAFORMA PROTOTIPO DE EMBOTELLADO

En el laboratorio de automatización, se dispone de una plataforma prototipo de naturaleza secuencial que representa un proceso de embotellado industrial. Esta plataforma cuenta con 70 entradas y 112 salidas. Como se observa en la Figura 1, está dividida en cuatro módulos: llenado, tapado, paletizado y reciclado.

La falta de sensores y/o transductores limita la capacidad de medir variables eléctricas clave, como el voltaje y la corriente consumida por los dispositivos. Asimismo, la ausencia de una interfaz de supervisión en tiempo real dificulta la monitorización de variables de interés, como los estados de los actuadores y los consumos eléctricos. Por último, la carencia de una estrategia y arquitectura de control impide la medición y la actuación sobre los dispositivos de la plataforma. Por lo anterior, el objetivo

se centra en diseñar e implementar un sistema de control y supervisión para la puesta en operación del módulo de reciclaje.

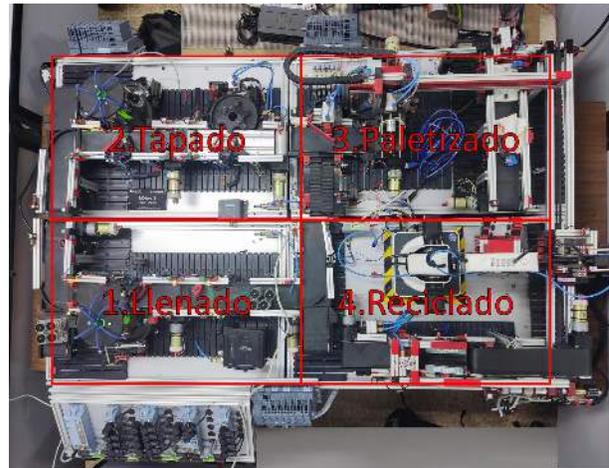


Figura 1. Segmentación de la plataforma prototipo

4. MÓDULO DE RECICLAJE

Como se observa en la Figura 2, el cuarto módulo está principalmente conformado por un brazo robótico de 4 grados de libertad, 3 bandas y 1 riel de transporte con control bidireccional, 5 motores de DC, 5 electroválvulas y 7 sensores. Para ello, cuenta con 20 señales de entrada y 20 de salida en un sistema remoto de entradas y salidas, o también llamado, periferia descentralizada.

Por otra parte, la función de este módulo consiste en que, con ayuda del robot manipulador, se tome un empaque con 4 botellas que simulan estar llenas mediante un LED en su interior. Posteriormente, se apaga dicho LED para simular un proceso de vaciado y, después, el robot separa las botellas del empaque para reintegrar cada elemento a su respectiva etapa del proceso.



Figura 2. Módulo de reciclaje del proceso

5. ARQUITECTURA DE CONTROL

La arquitectura de control se diseñó pensando en el futuro, considerando la eventual operatividad de todos los módulos y la posibilidad de añadir nuevos dispositivos. Para cumplir con este propósito, se implementó una arquitectura que permitiera un acceso inalámbrico al controlador desde cualquier punto dentro de la zona de trabajo, a la vez que facilitará la comunicación del controlador con la

periferia descentralizada de los cuatro módulos del prototipo. Es por ello que, como se muestra en la Figura 3, se optó por una topología en forma de árbol con un protocolo de comunicación PROFINET, proporcionando una estructura flexible, escalable y reconfigurable que garantiza una gestión eficiente y una futura expansión sin complicaciones.

Lo primordial para esta arquitectura fue implementar el PLC, el cual es un SIMATIC S7-1500 con CPU 1516F-3 PN/DP. Este controlador cuenta con la capacidad de integrar interfaces de supervisión web gracias a la herramienta View of Things. Además, su alta capacidad de procesamiento lo hace adecuado para controlar las 70 señales de entrada y 112 señales de salida de todo el proceso. Por otra parte, para conectar los sensores, motores, electroválvulas, el brazo robótico y todos los dispositivos a nivel planta, se utilizó una periferia descentralizada ET 200SP, un sistema remoto de entradas y salidas que facilita la conexión del controlador con los dispositivos a nivel de campo. Por otra parte, para poder interconectar la periferia descentralizada de la etapa de reciclado, así como cualquier otro dispositivo de automatización a futuro, se decidió incorporar un Switch Ethernet Industrial SIMATIC SCALANCE X308-2LD el cual permite tanto conmutar así como enrutar paquetes de datos. Esta elección proporciona una sólida base para la expansión y la integración de nuevos componentes de manera eficiente en un entorno industrial con condiciones desafiantes. A este mismo switch se le conectó una estación de ingeniería con el entorno de desarrollo TIA Portal, el cual permite acceder y programar al PLC. Finalmente, se aprovechó la infraestructura existente en el laboratorio, lo que permitió conectar la arquitectura propuesta a otro switch Ethernet y, a su vez, a un punto de acceso inalámbrico, facilitando la conexión de dispositivos inalámbricos a la interfaz web.

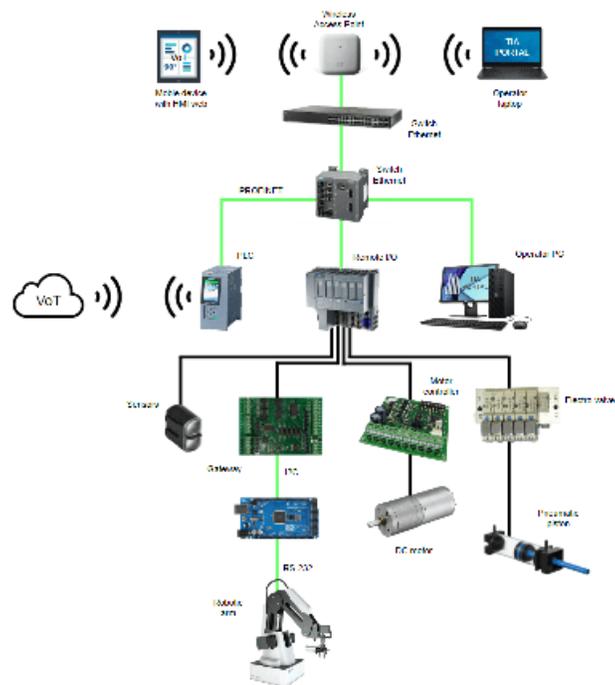


Figura 3. Arquitectura de control para la plataforma
<https://doi.org/10.58571/CNCA.AMCA.2024.097>

6. SISTEMA DE CONTROL Y SUPERVISIÓN

6.1 Rutinas de control

Las rutinas de control buscan operar el módulo según su descripción funcional. La estrategia principal utiliza dos lenguajes del estándar IEC 61131-3. Por una parte, como se muestra en la Figura 4, se observa parte de la rutina de control desarrollada con el lenguaje SFC. Este lenguaje es útil para definir de manera clara y sencilla las etapas, acciones y transiciones de un proceso secuencial complejo, lo que facilita el diseño, la implementación, la detección de errores, la documentación, el entendimiento y la reconfiguración de nuestra estrategia de control.

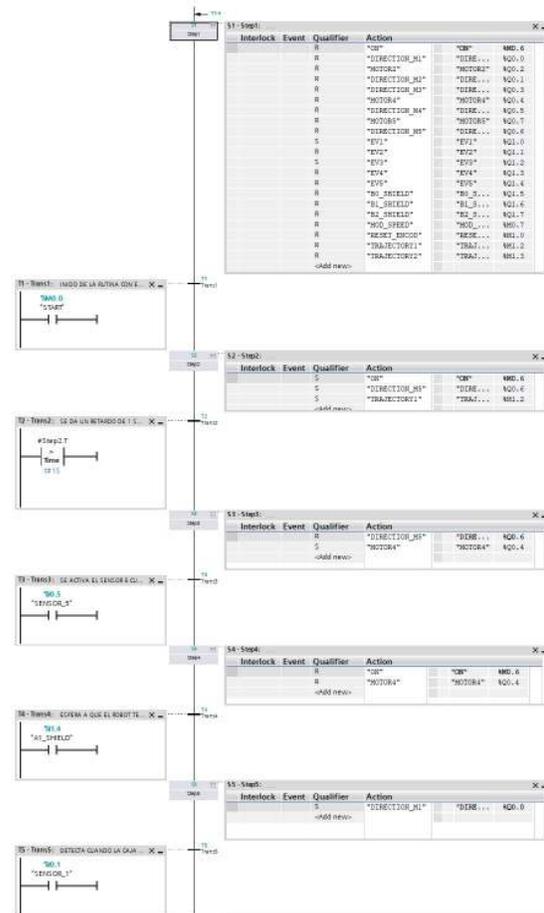


Figura 4. Fragmento de la rutina de control SFC implementada

En la primera etapa de esta rutina de control, se definen las condiciones iniciales del proceso, tal como los actuadores y las variables auxiliares. La primera condición de transición está determinada por la variable *START* que se activa al arrancar el proceso. La segunda etapa especifica qué motor y variables auxiliares se activarán, en este caso se activa el motor 5 y las variables *ON* y *TRAJECTORY1*. En su respectiva transición se define un retardo de 1 segundo. En la tercera etapa, se apaga el motor 5 y se enciende el motor 4, mientras que la tercera transición está definida por el sensor 5 que se activa cuando un empaque con botellas llega a la zona de recepción. La cuarta etapa desactiva al motor 4 y a la variable auxiliar *ON*. En su respectiva transición se define la variable *A1_SHIELD* la

cual se activa cuando el brazo robótico termina de ejecutar su secuencia de movimientos. En la quinta etapa, se activa el motor 1 que mueve a la banda transportadora superior, mientras que la quinta transición está definida por el sensor 1 que se activa cuando detecta el paso del empaque por la mitad del recorrido de la banda transportadora.

Por otra parte, En la Figura 5 se muestra una parte de la rutina de control desarrollada en lenguaje de escalera. Esta rutina incluye funciones adicionales como arranque, paro, reinicio, configuración de velocidad y la lectura e interpretación de variables eléctricas.

En el primer peldaño, se invoca el bloque de función asociado a la rutina en SFC. Este bloque permite ejecutar la rutina de control SFC de manera paralela y gestionar bits de entrada y salida que afectan la rutina, como el bit *INIT_SQ*, que reinicia la rutina; el bit *SW_AUTO*, que reanuda la rutina desde una pausa; y el bit *SW_MAN*, que pausa la rutina. En el segundo peldaño, se establecen las condiciones lógicas para el arranque del proceso mediante la variable *START*. Esta condición lógica está definida por un botón físico y un arranque digital en la HMI (Human Machine Interface). El tercer peldaño define las condiciones lógicas para el paro mediante la variable *STOP*. Esta condición es un enclavamiento con prioridad a la desconexión, definido por un botón físico y un paro digital en la HMI.

Es importante señalar que el lenguaje de escalera, al igual que el lenguaje SFC, forma parte del estándar IEC 61131-3. Esto permite desarrollar rutinas que interactúan y se complementan entre sí, compartiendo y operando variables en ambos lenguajes. Esto resulta de mucha utilidad al diseñar, implementar, dar mantenimiento y mejorar la estrategia de control propuesta.

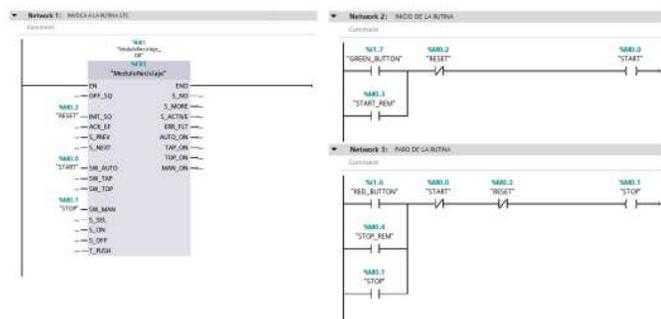


Figura 5. Fragmento de la rutina de control en lenguaje escalera implementada

Finalmente, en la Figura 6, se muestra parte de la subrutina de control de movimiento para los motores, también llamada arranque suave. Esta rutina consiste en otro diagrama de escalera, donde mediante un control gradual del PWM de los motores, se generan perfiles de movimiento de acuerdo al tiempo de arranque seleccionado por el usuario. Esto tiene la finalidad de reducir el desgaste mecánico generado por el estrés de un arranque con el máximo torque del motor, optimizar el consumo de corriente, y, en un caso más realista, evitar que el producto sufra perturbaciones a causa de movimientos abruptos en las bandas transportadoras, lo que podría afectar la calidad del producto final.

En el primer peldaño, se define la variable *START_M1* que arrancará al motor 1. En el segundo peldaño, se define la condición para activar el arranque suave. La variable *SOFT_M1* activa el control de variación del PWM en los motores, permitiendo generar una rampa de aceleración según el tiempo seleccionado. Si la variable está desactivada, el PWM se establece al 100%, resultando en un arranque directo. En el tercer peldaño, se define la instrucción para asignar el tiempo de arranque seleccionado por el usuario. En el cuarto peldaño, se calcula el factor de variación del PWM en función del tiempo de arranque seleccionado.

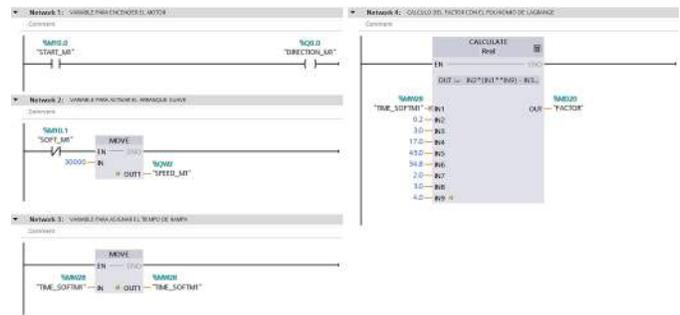


Figura 6. Fragmento de la rutina en lenguaje escalera para el control de movimiento de los motores

6.2 Control del robot

Para ejecutar los movimientos del robot, se utiliza una secuencia de puntos específicos hacia los cuales se debe desplazar el manipulador. Estos puntos se definen mediante coordenadas cartesianas obtenidas directamente del software DobotStudio, proporcionado por el fabricante. En la Figura 7 se muestra la interfaz del software, donde se configuran los puntos de desplazamiento, los tiempos de pausa, las acciones del efector final instalado, así como los puertos de entrada y salida.

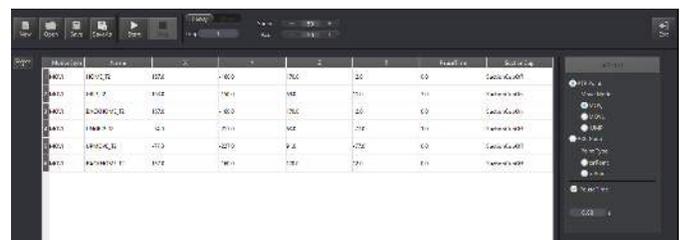


Figura 7. Interfaz de usuario de DobotStudio

Una vez definidos los puntos deseados se realiza la comunicación con el controlador. Esto permite que el controlador mande un bit lógico al puerto serie del robot para iniciar la secuencia de movimientos, y recibir otro bit para confirmar que el robot termino la secuencia. Es importante destacar que el robot no puede recibir señales digitales mayores a 5 [V], ni enviar señales digitales mayores a 12 [V]. Al no tener la capacidad de conectarse directamente a un controlador que, en este caso, recibe y envía señales digitales a 24 [V], fue necesario implementar una pasarela de conexión entre el controlador y el brazo robótico. Al igual que en las redes de comunicación, una puerta de enlace facilita la interconexión y la comunicación entre redes distintas, esta pasarela permite comunicar al PLC y al brazo robótico sin alterar sus respectivos voltajes de operación.

6.3 Pasarela de conexión

En este caso particular, existen diferentes formas de implementar una pasarela de conexión. Sin embargo, la solución implementada utiliza una placa de desarrollo Arduino MEGA y un módulo de expansión industrial, el 24 [V] Industrial I/O Shield. De esta manera, el PLC puede enviar señales digitales de 24 [V] a través de la periferia descentralizada, que son recibidas por el módulo de expansión y convertidas en señales de control que la placa de desarrollo procesa e interpreta. Posteriormente, mediante el puerto serie, se envían comandos de control de 5 [V] al robot. De forma análoga, el robot envía señales de control de 5 [V] al Arduino, el cual a su vez utiliza el módulo de expansión para emitir señales digitales de 24 [V] hacia el controlador. El esquema de conexiones de esta pasarela se muestra en la Figura 8.

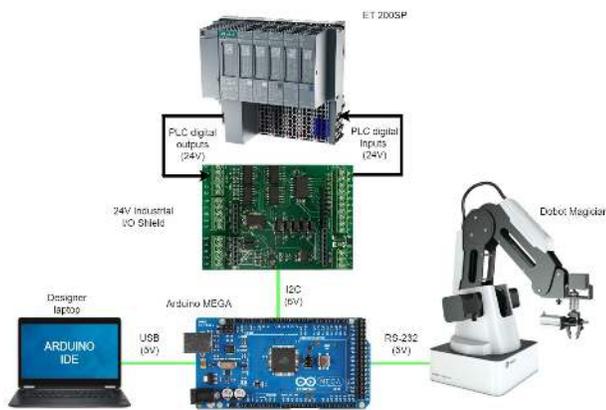


Figura 8. Pasarela de conexión

6.4 HMI web

La interfaz web se desarrolló con la herramienta View of Things integrada al software TIA Portal. Esta es una HMI descentralizada que tiene como propósito ser simple y eficiente. Como se puede observar en la Figura 9, la interfaz se divide en 3 paneles: el primero es el panel de control del proceso, que incluye botones para arranque, paro y reinicio, un indicador de la etapa actual de la rutina SFC, y un indicador del estado del robot; el segundo es el panel de monitoreo de la eficiencia energética, que muestra indicadores clave de rendimiento como voltaje, corriente y potencia; y el tercero es el panel de supervisión de actuadores, que muestra el estado actual de cada motor y electroválvula del módulo.



Figura 9. Interfaz de control y supervisión web
<https://doi.org/10.58571/CNCA.AMCA.2024.097>

7. RESULTADOS

7.1 Nivel de campo

Al comenzar el proyecto, se enfrentó con un prototipo el cual carecía de documentación. La primera tarea fue identificar, investigar, documentar y dar mantenimiento a todos los dispositivos del módulo. El siguiente desafío fue comprender las conexiones eléctricas, esto se logró verificando todo el cableado y creando un diagrama detallado con AutoCAD Electrical, lo que permitió tener una visión clara y sistemática de la configuración eléctrica del sistema, facilitando así cualquier tarea de diagnóstico o mantenimiento futuro. El uso del brazo robótico también fue un reto debido a la falta de experiencia previa con este tipo de tecnologías. Primeramente, se comprendió el alcance y limitaciones del robot, y después se procedió a programarlo con su interfaz de usuario, DobotStudio, permitiendo su operación en línea, fuera de línea y mediante comandos por puerto serie. Por otra parte, se integró un transformador de corriente necesario para medir la corriente alterna que consume este cuarto módulo, debido a que el medidor de energía de la periferia descentralizada no podía hacerlo directamente de la fuente de alimentación. Por otra parte, también se implementó un transductor de corriente para monitorear y comparar el consumo de corriente de los motores, mejorando así la gestión energética del sistema.

7.2 Nivel de control

La fase de puesta en marcha del control fue desafiante debido a varios obstáculos. Uno de los principales retos fue controlar los movimientos del robot sin la interfaz de usuario de DobotStudio, y utilizando solo dos señales digitales del PLC. Esto se solucionó con la pasarela de conexión instalada y desarrollando una rutina de control en Arduino que recibe las señales del controlador y envía comandos de movimiento hacia el robot a través del puerto serie para finalmente confirmar la ejecución de los movimientos al PLC. Por otra parte, la creación de un proyecto en TIA Portal presentó desafíos, especialmente en la configuración de la periferia descentralizada. Pues cada módulo debe ser seleccionado y configurado adecuadamente para evitar problemas de comunicación y garantizar un correcto funcionamiento del sistema de control. Por otra parte, con la rutina de control de movimiento para motores implementada, se realizaron pruebas comparativas con y sin control de arranque suave en los motores. Como se puede apreciar en la Figura 10, se muestra el perfil de consumo de corriente tanto en un arranque directo como en un arranque suave. La principal diferencia entre estos dos perfiles es que, en el arranque suave, se evita el pico máximo de consumo de corriente, generándose en su lugar una rampa de consumo. Esto reduce el desgaste en los motores y las bandas transportadoras, al minimizar el estrés mecánico que provoca un arranque directo con el torque máximo del motor. Además, al evitar el sobrepaso inicial en el consumo de corriente, se garantiza una mayor estabilidad en el sistema eléctrico. El análisis de los resultados muestra una reducción en el consumo de corriente de 50 [mA]. Aunque esta mejora pueda parecer modesta, es importante recordar que estamos hablando de motores de 24 [V]. Sin embargo, al extrapolar este ahorro a motores industriales de mayor potencia, el impacto será significativo, con potenciales ahorros energéticos considerables.

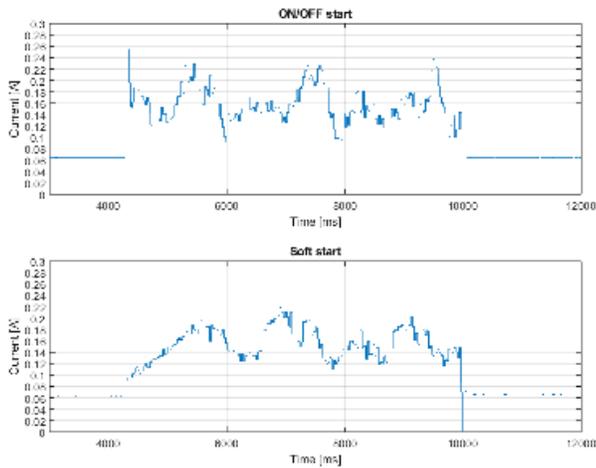


Figura 10. Comparativa del consumo de corriente con y sin el control de movimiento para el arranque

7.3 Nivel de supervisión

La etapa de implementación a nivel supervisión resultó ser fluida y sin contratiempos significativos, lo que ahora posibilita monitorear e incluso controlar el proceso desde una laptop o cualquier dispositivo móvil con un navegador web, tal como se puede apreciar en la Figura 11. Esto se logra gracias a la herramienta View of Things, la cual agrega un gran valor al proceso, acercándolo un paso más hacia la Industria 4.0 de una manera sencilla y eficiente. Considerando su naturaleza descentralizada, esta herramienta ofrece múltiples ventajas al no estar vinculada a una arquitectura específica. En consecuencia, es posible acceder a la interfaz desde prácticamente cualquier dispositivo con un navegador web conectado a la misma red del controlador. Es importante destacar que las interfaces desarrolladas con View of Things son seguras. Al igual que con las interfaces tradicionales, se requiere de un usuario y contraseña para poder acceder, y cada usuario es configurado previamente con sus respectivos privilegios. Además, el servidor web utiliza HTTPS (Hypertext Transfer Protocol Secure), basado en el uso de certificados SSL (Secure Sockets Layer) y TLS (Transport Layer Security), que autentican la identidad del sitio web y cifran la información transmitida entre el usuario y el servidor.



Figura 11. HMI web en dispositivos móviles
<https://doi.org/10.58571/CNCA.AMCA.2024.097>

8. CONCLUSIONES

Este trabajo presenta el desarrollo e implementación de un sistema de control secuencial para una plataforma experimental de embotellado. A través de un levantamiento técnico detallado, se ha comprendido el funcionamiento del módulo de reciclaje y se ha diseñado una arquitectura de control confiable, flexible y reconfigurable que facilita la programación e integración de los dispositivos de automatización actuales, así como los futuros que se pudieran integrar.

La estrategia y la arquitectura de control, desarrolladas en el lenguaje secuencial SFC y en los diagramas de escalera, de acuerdo con la metodología GRAFCET y la norma IEC 61131-3, garantizan un funcionamiento correcto y seguro del sistema, además de optimizar el consumo eléctrico de los motores y reducir el desgaste prematuro. Asimismo, la arquitectura permite una fácil mejora y reconfiguración del sistema de control en caso de ser necesario en el futuro.

Por otra parte, la interfaz de control y supervisión, implementada en un servidor web, permite monitorear y gestionar el proceso en tiempo real desde cualquier dispositivo móvil conectado a la misma red del controlador, sin incurrir en gastos adicionales de software y hardware. Es importante destacar que el uso de estas nuevas tecnologías basadas en el IoT (Internet of Things) apoya la digitalización de las industrias y la transición hacia una Industria 4.0 en busca de conseguir entornos de producción más eficientes, flexibles y conectados.

Finalmente, la investigación ha dado lugar a un sistema de control secuencial y supervisión diseñado para una plataforma experimental de embotellado. Este sistema se ajusta al paradigma actual de las tendencias en manufactura, integrando un control estandarizado y respaldado por modernas tecnologías industriales. Como resultado, se proporciona una herramienta valiosa para la formación y capacitación de profesionales en el control y automatización industrial.

REFERENCIAS

- Castellanos, J. and Solaque, L. (2010). Modelado con redes de Petri e implementación con GRAFCET de un sistema de manufactura flexible con procesos concurrentes y recursos compartidos. In *Universidad Militar Nueva Granada*.
- García Moreno, E. (2014). Automatización de procesos industriales. In *2014 Universitat Politècnica de València*.
- Hacienda, S. (2023). El nearshoring es la oportunidad de impulsar el crecimiento económico de México, señala Ramírez de la O. In *2023 Secretaría de Hacienda y Crédito Público*.
- Hassan Reza, M.N., Agamudai Nambi Malarvizhi, C., Jayashree, S., and Mohiuddin, M. (2021). Industry 4.0 Technological Revolution and Sustainable Firm Performance. In *2021 Emerging Trends in Industry 4.0 (ETI 4.0)*, 1-6. 10.1109/ETI4.051663.2021.9619363.
- IEC, I.E.C. (2003). IEC 61131-3 Programmable controllers - Part 3: Programming languages. In *International Standard*.
- IEC, I.E.C. (2013). IEC 60848 GRAFCET specification language for sequential function charts. In *International Standard*.
- Quiroz Cuenca, S. (2021). Condiciones del nearshoring en México 2023. In *2021 Universidad Autónoma del Estado de México*.
- Ramanathan, R. (2014). The IEC 61131-3 programming languages features for industrial control systems. In *2014 World Automation Congress (WAC)*, 598-603. 10.1109/WAC.2014.6936062.
- Sánchez-Chávez, I.Y. and Martell-Chávez, F. (2019). Automatización multinivel de celda de manufactura robotizada utilizando máquinas de estados finitos. In *Ingeniería Investigación y Tecnología*.