

DISEÑO DE PLATAFORMA DE TELEINGENIERÍA PARA PRÁCTICAS DE LABORATORIO A DISTANCIA: CONTROL DE PROCESO INDUSTRIAL DE TEMPERATURA

Juan José Guevara Vásquez

Técnico en Ingeniería Electrónica, Docente Investigador de Escuela de Ingeniería Eléctrica y Electrónica. Escuela Especializada en Ingeniería ITCA-FEPADE, Sede Central Santa Tecla. Correo electrónico: juan.guevara@itca.edu.sv

Juan José Cáceres Chiquillo

Ingeniero en Electrónica, Docente Coinvestigador de Escuela de Ingeniería Eléctrica y Electrónica. Escuela Especializada en Ingeniería ITCA-FEPADE, Sede Central Santa Tecla. Correo electrónico: jcaceres@itca.edu.sv

Recibido: 25/04/2022 - Aceptado: 1/07/2021

Resumen

Este proyecto de investigación fue ejecutado por la Escuela de Ingeniería Eléctrica y Electrónica de ITCA-FEPADE en 2021 y está enmarcado dentro de la mejora de la educación e innovación, a través de tecnologías electrónicas e informáticas que buscan proveer a estudiantes y docentes de carreras técnicas y de ingeniería, el desarrollo de prácticas de laboratorio a distancia, no simuladas y en tiempo real, en el área de control de procesos industriales de temperatura de fluidos, a través de la plataforma de Teleingeniería desarrollada. Se realizó un proceso de reingeniería a un entrenador de Control de Procesos de Fluidos, FPC, para que éste sea comandado de forma remota en una arquitectura cliente-servidor; de forma paralela, se diseñó una aplicación Web para el control, registro y monitoreo de las prácticas de laboratorio que funciona como cliente del controlador electrónico del entrenador. Entre los resultados se destacan el diseño y arquitectura del FPC y el diseño del protocolo de comunicación entre la máquina y la aplicación. La Plataforma de Teleingeniería utiliza herramientas de software y hardware de libre distribución, reduciendo los costos de implementación y mantenimiento. La Plataforma de Teleingeniería integra diversos sistemas y tecnologías por lo que habrá que realizarle ajustes y adecuaciones que mejoren la experiencia de los estudiantes y expandan los procesos a controlar con la adición de otros entrenadores FPC. Este proyecto establece un nuevo estado de la técnica, ya que en El Salvador no existe una plataforma similar desarrollada en un proyecto de una Institución de Educación Superior.

Palabras clave

Control de procesos, control de temperatura, sensor, microcontroladores, sistemas de control automático.

DESIGN OF A TELEENGINEERING PLATFORM FOR REMOTE LABORATORY PRACTICES: TEMPERATURE INDUSTRIAL PROCESS CONTROL

Abstract

This research project was executed by the Escuela de Ingeniería Eléctrica y Electrónica de ITCA-FEPADE in 2021 and is framed within the improvement of education and innovation, through electronic and computer technologies that seek to provide students and teachers of technical and engineering careers, the development of remote laboratory practices, not simulated and in real time, in the area of industrial process control of fluid temperature, through the Teleengineering platform developed. A Fluid Process Control (FPC) trainer was reengineered to be remotely controlled in a client-server architecture; at the same time, a Web application was designed to control, register and monitor the laboratory practices that works as a client of the trainer's electronic controller. The results include the design and architecture of the FPC and the design of the communication protocol between the machine and the application. The Teleengineering Platform uses freely distributed software and hardware tools, reducing implementation and maintenance costs. The Teleengineering Platform integrates several systems and technologies, so it will be necessary to make adjustments and adaptations to

improve the students' experience and expand the processes to be controlled with the addition of other FPC trainers. This project establishes a new state of the art, since in El Salvador there is no similar platform developed in a Higher Education Institution project.

Keyword

Process control, temperature control, sensor, microcontrollers, automatic control systems.

Introducción

En el control de procesos electrónicos industriales es muy importante que tanto docentes como estudiantes dispongan de equipos de entrenamiento, que bajo un entorno controlado puedan realizar prácticas que permiten aplicar los conceptos estudiados y solidificar las competencias.

ITCA-FEPADE dispone de entrenadores para el control de distintos procesos en cátedras de nivel técnico y de ingeniería, con la limitante de un entrenador para cada proceso industrial específico; se suman los efectos de la pandemia por Covid-19 que ha reducido la presencialidad de los estudiantes y docentes por motivos de bioseguridad, limitando el acceso a los entrenadores.

Por esta razón se realizó un proceso de reingeniería a uno de los entrenadores de Control de Procesos de Fluidos FPC con los que cuenta la Escuela; el objetivo fue controlar el equipo de forma remota a través de una aplicación que se accede por medio de un navegador Web moderno, ya sea desde una computadora de escritorio o dispositivo móvil. Los docentes y estudiantes podrán realizar las prácticas de laboratorio de forma programada y segura sin estar frente al equipo.

El sistema o plataforma como tal, es una combinación de dispositivos electrónicos discretos, de potencia y control, que junto a una aplicación desarrollada utilizando lenguajes de programación orientados a la Web, logra acercar a los usuarios a una experiencia real de trabajo de laboratorio sin importar la distancia.

Desarrollo

I. METODOLOGÍA

A. Análisis del Entrenador de Control de Procesos de Temperatura

Inicialmente se verificó el estado de cada una de las partes electrónicas y electromecánicas que forman parte del entrenador, con el objetivo de determinar si alguna

presentaba fallas que impidieran el correcto funcionamiento de la máquina.

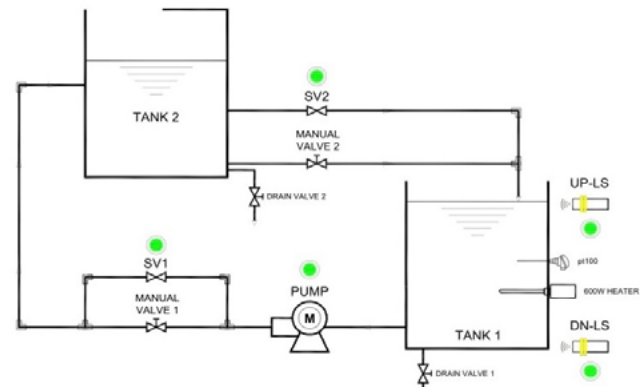


Fig. 1. Diagrama del entrenador de Control de Procesos Industriales de Temperatura.

En la figura 1 se observa el diagrama del proceso y presencia de sensores y actuadores. Se dispone de tres sensores: un sensor RTD PT100 fabricado con platino con una resistencia eléctrica de 100 ohmios a una temperatura de 0 °C que es el tipo más común de sensor RTD [1] que se utiliza en la industria, siendo su función medir constantemente la temperatura del fluido; dos sensores capacitivos de proximidad IP65 con led de estado (UPLS, DNLS), que tienen como función detectar la presencia de fluido por lo que operan como sensores de nivel bajo y alto para el tanque 1; estos sensores están especialmente diseñados para operar en contenedores no metálicos[2]. El entrenador cuenta además con actuadores, dos de ellos son electroválvulas NC (SV1, SV2), cuya función es permitir o bloquear el paso del fluido entre los tanques y una resistencia calentadora de 600W (Heater) que se utiliza para calentar el fluido que se encuentra en el tanque 2 y que recirculará en el sistema por medio de una bomba de agua (PUMP). Las electroválvulas, sensores capacitivos y la bomba operan con niveles de tensión de 24V ampliamente utilizada en sistemas de control industrial.

El control nativo de este entrenador es de tipo local, esto significa que cuenta con un panel de control compuesto por interruptores de START, STOP, RESET con selección de modo

de operación y un monitor de temperatura Autonics T3NI.

El controlador electrónico original del entrenador estaba basado en un sistema de WAGO, compuesto por un controlador Profibus y módulos de expansión para el acople de niveles de voltaje y corrientes de sensores y actuadores.

Tabla 1.
Definición de entradas y salidas del controlador del entrenador de control de procesos de temperatura

CONTROLADOR FPC WAGO 750-342		
ENTRADAS		
NOMBRE	TIPO	DESCRIPCIÓN
UP-LS	DIGITAL	Activo en H
DN-LS	DIGITAL	Activo en H
MODE 1	DIGITAL	Activo en H
MODE 2	DIGITAL	Activo en H
START	DIGITAL	Activo en H
STOP	DIGITAL	Activo en H
RESET	DIGITAL	Activo en H
EMERGENCY	DIGITAL	Activo en L
RTD	ANALOGICO	0V - 5V DC
SALIDAS		
NOMBRE	TIPO	DESCRIPCIÓN
SOL VALVE 1	DIGITAL	Activo en H
SOL VALVE 2	DIGITAL	Activo en H
PUMP MOTOR	DIGITAL	Activo en H
HEATER	PWM	1V - 5V

La revisión del entrenador se completó realizando un análisis técnico de las entradas y salidas del controlador WAGO que se pueden apreciar en la tabla I.

B. Premisas Generales de Diseño

El trabajo aquí descrito integra diversas tecnologías y áreas de aplicación, fue necesario crear un equipo multidisciplinario de investigadores y estudiantes que trabajaron de forma paralela para alcanzar los resultados. En general, se consideraron las siguientes premisas de diseño:

1) Bajo costo y mantenimiento

Debido a limitaciones económicas la solución a diseñar debía ser de bajo costo, sin reducir la calidad del producto

y servicios. El mismo criterio aplicó para el diseño de la aplicación Web de gestión de prácticas a distancia. Se consideró que todos los componentes de hardware fuesen adquiridos fácilmente en el mercado local.

2) Utilizar hardware y software libre

Tanto el controlador electrónico como la aplicación Web para la gestión y control de las prácticas de laboratorio a distancia fueron diseñadas desde un inicio utilizando tecnología y herramientas de hardware y software libres.

C. Diseño del Controlador Electrónico

Con la información de la Tabla I, así como con las características eléctricas de actuadores y sensores, se hizo un diseño modular del controlador electrónico que sustituiría al controlador WAGO.

El diseño del nuevo controlador está esencialmente compuesto por dos subsistemas de control.

a) Sistema de control

Este circuito tiene como principal función el correcto establecimiento de la comunicación con la aplicación web de control, motivo por el cual tiene la capacidad de conectarse a una red de datos Ethernet. Además, gestiona el funcionamiento del circuito convertidor del sensor de temperatura PT100 y del microcontrolador gestor de entradas y salidas.

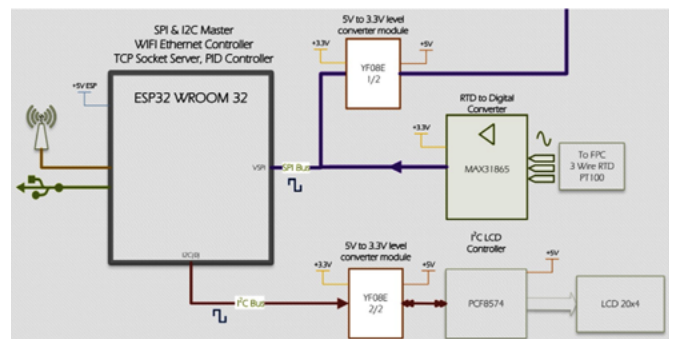


Fig. 2. Diagrama Modular del Sistema de Control.

Como microcontrolador principal se eligió el ESP32 WROOM32 de Expressif, que viene embebido en una placa de desarrollo ESP32 Devkit V1 de 30 terminales. Este dispositivo cumple con todos los requerimientos de diseño.

Al ser un procesador con niveles de tensión de hasta 3.3V se dotó al circuito de un acople de niveles de tensión basado en el circuito YF08E. El diseño modular resultante se muestra en la figura 2.

b) Sistema de control de entradas y salidas

Este es un sistema de control robusto que opera como esclavo SPI del sistema de control principal. El requerimiento de robustez implica un mayor manejo de corrientes de entrada y salida, así como la generación y control de una señal PWM. Se eligió el microcontrolador PIC18F4550 de Microchip que está ampliamente documentado y puede programarse utilizando lenguaje XC8 que es ANSI C.

Considerando que el entrenador posee dispositivos industriales que trabajan con niveles de tensión de 24V, el circuito controlador debe cumplir con un requisito indispensable que es el aislamiento galvánico entre los sensores e interruptores de entrada y el sistema de control de entradas y salidas, esto se logró con optoacopladores PC817. Adicionalmente, el circuito de salida debe amplificar las señales para activar los actuadores correspondientes; para lograrlo se utilizó el circuito integrado ULN2003A que es compatible con señales TTL de entrada y sus salidas pueden manejar niveles de voltaje de 24V. El diagrama modular se muestra en la figura 3.

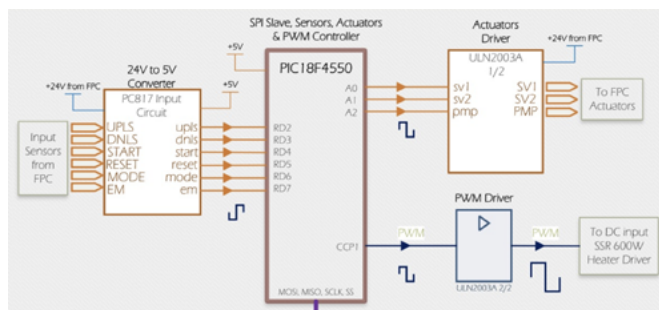


Fig. 3. Diagrama Modular del Sistema de Control de Entradas y Salidas.

D. Diseño de las Condiciones Generales de Funcionamiento de la Aplicación.

Una vez finalizado el diseño general de bloques del controlador electrónico y habiendo seleccionado el dispositivo controlador de comunicación ESP32 WROOM 32, se procedió a crear las condiciones de funcionamiento de la aplicación de software que forma parte de la Plataforma de Teleingeniería.

En esta etapa, el trabajo estuvo orientado a definir dos elementos:

1) Módulo de gestión; control de acceso y pantalla de control del entrenador

- Módulo de Control de acceso seguro para que docentes y alumnos puedan hacer uso de la plataforma

validándose por medio de correo electrónico y contraseña de acceso.

- Módulo de Administración de usuarios mediante la cual el docente puede registrar y realizar modificaciones en los datos de los estudiantes.
- Módulo de Administración de horarios de prácticas a distancia en donde los estudiantes pueden programar y reservar una práctica de laboratorio semanal la cual debe a su vez ser aprobada por el docente.
- Pantalla de Control del entrenador en donde se tendrá acceso a los controles del entrenador, cajas de texto para la introducción y modificación de parámetros, así como gráficas e indicadores que se utilizarán para el monitoreo del estado del proceso que se está ejecutando.

2) Protocolo de comunicación entre la aplicación y el controlador electrónico del entrenador FPC

Una de las etapas más críticas del proyecto consistió en el diseño y mejoramiento del protocolo de comunicación, el cual describe la forma en que la aplicación detecta lo que está haciendo el entrenador en un momento determinado, así como el estado del proceso, también sirve para que la aplicación gire órdenes al entrenador a través del controlador electrónico.

La definición del protocolo de comunicación está directamente relacionada con la estrategia de comunicación a utilizar y la red de datos. Por la naturaleza del proceso a controlar se requiere de un nivel de respuesta y control en tiempo real, por lo que se decidió utilizar el estándar de comunicación Ethernet, ya que los usuarios usarán una computadora y un navegador Web para conectarse y realizar sus prácticas de laboratorio.

Una vez definido el estándar de comunicación se pasó a determinar que protocolo de comunicación compatible con Ethernet sería el más adecuado; se evaluaron varias alternativas desde el uso de MQTT[3] hasta el diseño de una Rest API[4]. Sin embargo, debido a la necesidad de mantener una comunicación y actualización de la información en tiempo real se decidió utilizar Sockets basados en TCP[5] como método de transporte de los comandos y datos entre la aplicación y el controlador electrónico, que a su vez forman parte de un protocolo de comunicación hecho a medida. La figura 4 muestra las capas del modelo de comunicación utilizado.

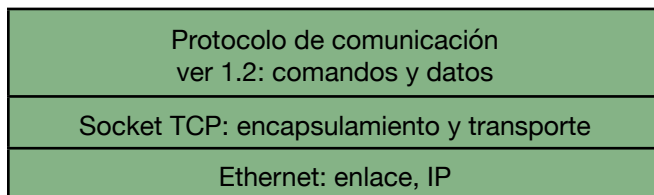


Fig. 4. Capas del Modelo de Comunicación.

E. Diseño del Firmware del Controlador Electrónico.

En el controlador electrónico hay dos microcontroladores: ESP32 y PIC18F4550. Se decidió iniciar el proceso de diseño del firmware del PIC. El lenguaje de programación seleccionado es el nativo del fabricante Microchip XC8 en su versión 2.0, en conjunto con el IDE MPLAB X y programador Pickit 3.

1) Diseño de firmware del controlador de entradas y salidas (PIC18F4550)

Se establecieron las siguientes premisas:

- Todas las acciones de control deben ser sincronizadas por un temporizador interno de 10ms.
- Para detectar el cambio de estado de los sensores de entrada tipo push button se utilizará la estrategia de lectura basada en estados de máquina[6] por medio de interrupciones internas, lo cual constituye la manera más efectiva de efectuar un debouncing por software[7].
- Se debe evitar en todo momento diseñar estructuras de programación con potenciales riesgos de bloqueo del procesador y tiempos muertos que impiden que otras tareas se lleven a cabo.

```
void __interrupt() high_isr(void){
    INTCONbits.GIEH = 0;

    //Check for 10ms timer flag
    if(INTCONbits.TMR0IF) {
        //Check if it's time to read and update SPST sensors
        Sensors_Update_SPST();

        /*Check if it's time to read ADC (every 1 second)
        * Argument(0) means that AN0 is selected.
        */
        ADC_Read(0);

        //Check if a button sensor is pressed
        Sensors_Update_Buttons();

        //Load count value
        Timer_Load_Count(tmr0lValue, tmr0hValue);

        INTCONbits.TMR0IF = 0; //Initialize Timer0 interrupt
    }
}
```

Fig. 5. Segmento de código que muestra parte de las rutinas de interrupción que a su vez son gobernadas por el temporizador de 10ms.

En la figura 5 se muestra un segmento de código que evita el bloqueo de los distintos procesos que el microcontrolador

debe manejar. Dado que todas las funciones se ejecutan a distintos intervalos de tiempo no se interfieren entre ellas produciendo una sensación efectiva de multitarea. Esta metodología de programación es la que se utiliza en los Sistemas Operativos de Tiempo Real (RTOS)[8] que son muy utilizados en sistemas embebidos.

- Para garantizar legibilidad y rápida depuración, cada dispositivo sensor, actuador, señal, etc. que esté sujeto a configuración y/o programación cuenta con su fichero de código correspondiente.
- El método main() del código se limita a invocar funciones de inicialización, implementar el bucle infinito y realizar el llamado de las funciones correspondientes cuando la bandera de interrupción apropiada sea activada.

2) Diseño de firmware del sistema de control

El controlador principal del SYSCON ESP32 fue seleccionado para garantizar la comunicación entre el servidor de aplicaciones en donde se encuentra el software de gestión y el control Web del entrenador FPC. Además, este dispositivo debe comunicarse de forma efectiva con el controlador de entradas y salidas por medio del bus SPI. En síntesis, es el elemento más importante del circuito controlador del entrenador FPC.

Se evaluaron distintos frameworks de programación, buscando principalmente un lenguaje que sea abierto, multiplataforma, de propósito general y que tuviese una versión para microcontroladores como el ESP32 WROOM 32.

El único lenguaje de programación que cumple con estas condiciones es Python, ya que además de ser un lenguaje de prototipado rápido [9], cuenta con MicroPython que es una implementación de Python 3.4 con todas las prestaciones de las versiones más recientes[10] y dispone de un firmware específicamente diseñado para la familia de controladores ESP32. El firmware es cargado en el microcontrolador utilizando la herramienta esptool de Python por medio de comandos específicos de consola sin necesidad de contar con herramientas especiales de programación (Como por ejemplo el programador Pickit 3 para el PIC18F4550) facilitando el proceso de desarrollo y depuración[11].

Todas las tareas asíncronas se definieron con un tiempo de actualización cuyo valor es experimental, esto significa que se pueden ajustar hasta obtener los mejores resultados.

Debe poseer funciones y variables que inicialicen y protejan al entrenador y controlador.

Si bien el entrenador será accedido remotamente, siempre será necesario el monitoreo de un operario o responsable del laboratorio que revise antes del inicio de las prácticas de laboratorio el estado del entrenador y garantice las condiciones de operación mecánicas y eléctricas.

F. Integración de Aplicación Web de la Plataforma y Controlador Electrónico.

Inicialmente, se realizó el montaje de todos los dispositivos de hardware que en su conjunto conforman la Plataforma de Teleingeniería. Para realizar este trabajo se rediseñó el diagrama de control y potencia del entrenador cuyo diseño final se muestra en la figura 6.

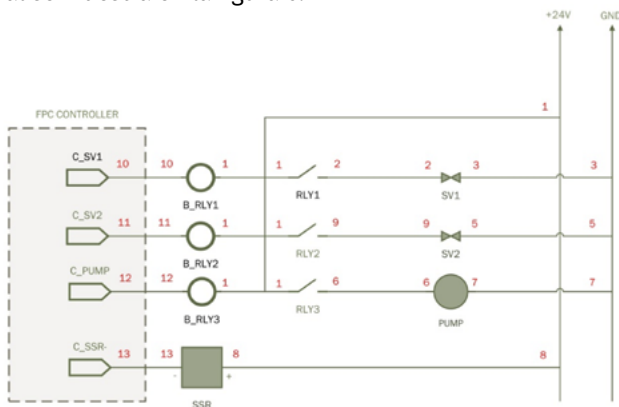


Fig. 6. Diagrama de control y potencia.

A partir de esta disposición se realizaron las conexiones en el tablero de control tal como se muestra en la figura 7.

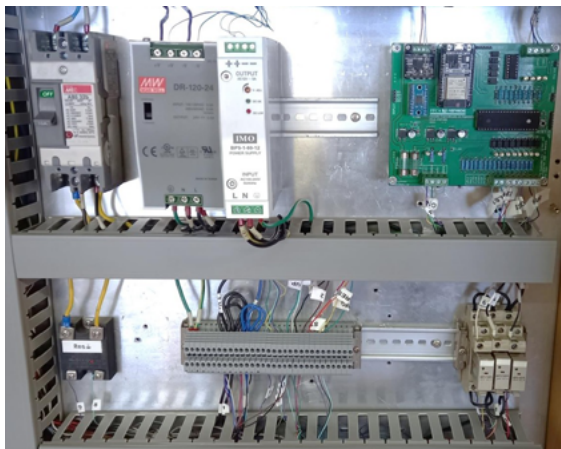


Fig. 7. Conexiones de control y potencia en el entrenador. En la esquina superior derecha puede observarse el circuito controlador electrónico diseñado.

Como puede apreciarse en la figura 7, el resultado es una instalación minimalista pero funcional para el control del entrenador. El diseño del controlador electrónico, en el que se procuró integrar etapas de entrada-salida con aislamiento galvánico, circuito de acondicionamiento de la RTD PT100 y el driver para los relés de los actuadores basado en ULN2003A, permitió ahorrar espacio y reducir el cableado para disminuir errores y facilitar el mantenimiento.

El hardware del servidor lo constituye una minicomputadora Raspberry Pi 4 de 4GB de memoria RAM y 32GB de almacenamiento en memoria SD, para su montaje se diseñó un case de protección. El entrenador cuenta con dos cámaras instaladas en brazos de soporte ajustables.

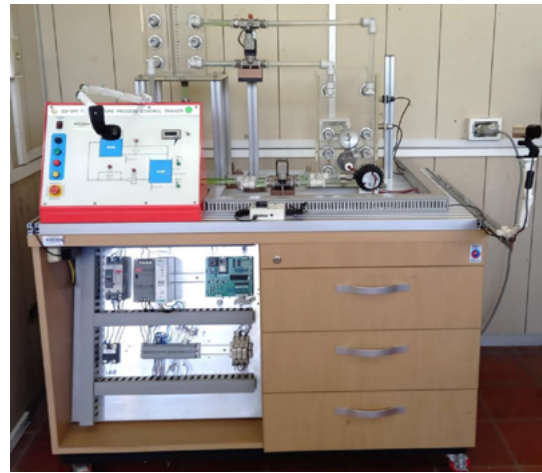


Fig. 8. Vista general del montaje final del entrenador FPC.

En la figura 8, se muestra el montaje final del entrenador. Para que los usuarios del módulo de prácticas tuvieran una visión clara de que las órdenes que se proporcionen de forma remota se estén reflejando en el entrenador, una de las cámaras muestra el tablero de control local, donde se observa la activación o desactivación de los componentes, válvulas solenoides de las tuberías y la bomba de agua del entrenador, así como el estado de los sensores capacitivos que detectan el nivel del fluido en el tanque 2. La segunda cámara muestra una vista general de las tuberías y tanques del entrenador en donde se pueda apreciar el flujo del fluido.

Una vez finalizada la instalación de los dispositivos del entrenador FPC se procedió a realizar la configuración de las direcciones IP del controlador electrónico y servidor Raspberry Pi 4, para ambos casos las direcciones son asignadas de forma estática por reserva de IP mediante DHCP.

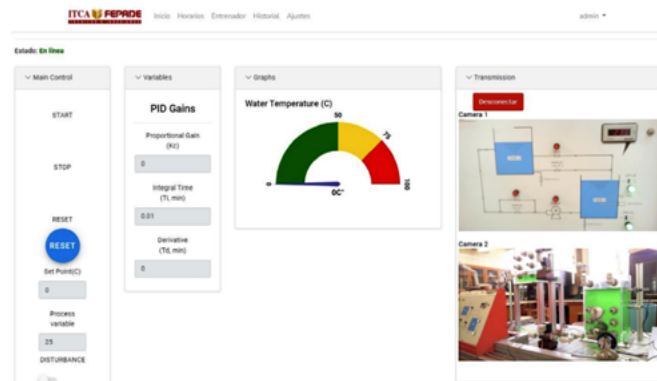


Fig. 9. Módulo de prácticas de la aplicación Web de la plataforma de Teleingeniería.

En la figura 9, se muestra la pantalla del módulo de prácticas de laboratorio de la aplicación Web. La columna de la izquierda Main Control, muestra el estado de máquina del entrenador; la columna a la derecha Variables, permite la asignación y modificación de las variables de control del proceso de forma dinámica; la columna Graphs muestra las gráficas de estado del proceso y finalmente la columna Transmission muestra las cámaras de video.

Para la puesta a punto de la plataforma, se diseñaron dos documentos de referencia: Protocolo de Comunicación y Diagramas de Estado de Máquina y de Flujo-estado, estos sirven para regir la comunicación entre el controlador electrónico del entrenador FPC y la aplicación Web.

En la figura 10 se muestra el Diagrama de Estado de Máquina del entrenador FPC; en este documento que es utilizado en aplicaciones de control industriales, se muestran los tres estados del entrenador que son RESET, START y STOP.

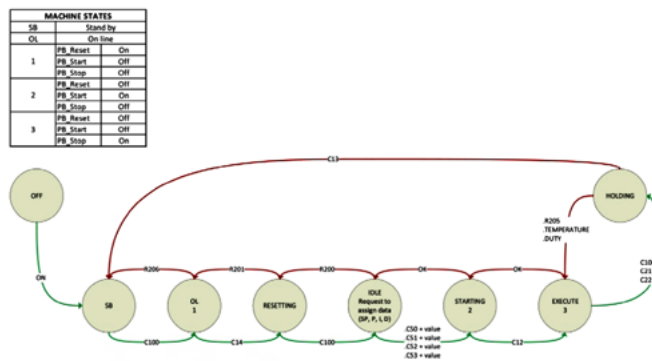


Fig. 10. Diagrama de Estado de Máquina del entrenador FPC.

Finalmente se realizaron pruebas generales de funcionamiento para detectar potenciales errores, sobre todo de comunicación entre la aplicación Web y el controlador electrónico. También tuvo como objetivo mejorar el firmware de los microcontroladores del controlador en un proceso de mejora continua.

Esta comprobación se realizó mediante el desarrollo de prácticas de laboratorio remotas, ya sea dentro de la red de datos de ITCA-FEPADE, como fuera del campus. Entre otras cosas, este proceso permitió ajustar los tiempos de refresco de lectura de la temperatura de la RTD PT100, actualizar el controlador PID para reducir lecturas erróneas que afecten el control del proceso. Permitieron mejorar las rutinas de protección del entrenador FPC para reducir los potenciales de daños por pérdidas de conexión o comandos no recibidos desde la aplicación.

Resultados

El resultado general de este proyecto lo constituye la Plataforma de Teleingeniería compuesta entre otros por:

- Diagrama Modular del Controlador Electrónico**, que puede y debe servir como referencia para la creación de otros controladores para el control de procesos de tipo industrial.
- Diseño del Controlador Electrónico** que integra etapas de entrada-salida con aislamiento galvánico, circuito de acondicionamiento de la RTD PT100 y el driver para los relés de los actuadores basado en ULN2003A.
- Software con diseño del Protocolo de Comunicación**, que es una referencia de comunicación asíncrona entre dos plataformas, aplicaciones Web y circuitos electrónicos, que utilizan tecnologías diferentes y que con la implementación de la industria 4.0 y las clases a distancia cada vez están más unidas.
- Diagrama Esquemático del Controlador Electrónico** del entrenador FPC que define un estado del arte como punto de partida para el diseño de otros circuitos controladores que manejen procesos industriales similares.
- Diseño de Circuito Impreso** de doble cara del controlador electrónico con software CAD gratuito que demuestra que se puede hacer un diseño profesional incluyendo vistas 3D del resultado final sin incurrir en adquisición de software con licenciamiento elevado.
- Diagramas de Estado de Máquina y de Flujo-estado** para plataformas que servirá como referencia de un modelo de comunicación entre un circuito de control electrónico industrial y una aplicación Web que se ejecuta en un servidor remoto.

Forman parte complementaria de los resultados:

- El código fuente de la aplicación Web para la gestión y desarrollo de prácticas de laboratorio.
- El código fuente del firmware del controlador SYSCON ESP32 y del controlador ES PIC18F4550.
- Manual de servicio de la aplicación Web de la plataforma de Teleingeniería.
- Guías de práctica de laboratorio para nivel básico, intermedio y avanzado.

Conclusiones

- Es posible diseñar un controlador electrónico de bajo costo pero que cumpla con los requerimientos de diseño de una máquina destinada a realizar un control de proceso industrial. Una solución de hardware como la diseñada en este proyecto de investigación puede llegar a tener un costo hasta cinco veces superior al utilizado con la desventaja que por lo general se trabaja con una plataforma cerrada y el agregar más prestaciones a la solución requiere siempre de una mayor inversión.
- Es posible conectar dinámicamente una aplicación que se ejecuta en un servidor remoto de forma eficiente y efectiva con un controlador electrónico siempre que se implemente un enfoque de comunicación y programación asíncrono.
- Los sockets TCP son especialmente útiles cuando se diseña una solución en la cual se debe intercambiar información en tiempo real y la conexión debe mantenerse activa por mucho tiempo.
- Se pueden realizar prácticas de laboratorio a distancia en entrenadores que tradicionalmente se manipulan únicamente de forma presencial. La plataforma de Teleingeniería diseñada es única en el país ya que no es un simulador, sino que todas las acciones de control que realiza el usuario se hacen en el entrenador físico, pero a distancia. Además, la incorporación del video en tiempo real mejora la experiencia del usuario ya que puede observar de forma efectiva que todas sus acciones se ejecutan directamente en el entrenador como si estuviera en una práctica presencial.
- El entrenador será accedido remotamente, pero siempre será necesario el monitoreo de un operario o responsable del laboratorio para que revise antes del inicio de las prácticas el estado del entrenador y garantice las condiciones de operación mecánicas y eléctricas.

Agradecimientos

Para el desarrollo de la aplicación Web de la plataforma se contó con el apoyo de E. Martínez y R. López, ingenieros docentes de la Escuela de Ingeniería en Computación (correos: emmartinez@itca.edu.sv; rina.lopez@itca.edu.sv) así como un equipo de estudiantes de la carrera de ingeniería en desarrollo de software.

Para el montaje de dispositivos de potencia y diseño del diagrama de estados de la máquina se contó con la colaboración de G. Meléndez docente de la Escuela de

Ingeniería Eléctrica y Electrónica (correo: carlos.melendez@itca.edu.sv)

Referencias

- [1] “¿Qué es una Pt100 y cómo funciona? | Termorresistencia pt100”. [En línea]. Disponible en: <https://es.omega.com/prodinfo/pt100.html>. [Accedido: 22-abr-2022]
- [2] “Capacitive Proximity Sensors | AutomationDirect”. [Online]. Available: https://www.automationdirect.com/adc/overview/catalog/sensors_-z-_encoders/capacitive_proximity_sensors. [Accessed: 22-Abr- 2022]
- [3] “MQTT - The Standard for IoT Messaging”. [Online]. Available: <https://mqtt.org/> [Accessed: 22-Abr- 2022]
- [4] I. Hübschmann, “A Complete Guide to REST APIs in IoT”, Nabto. [Online]. Available: <https://www.nabto.com/rest-api-iot-guide>. [Accessed: 22-Abr- 2022]
- [5] “Transmission Control Protocol”, Internet Engineering Task Force, Request for Comments RFC 793, sep. 1981. doi: 10.17487/RFC0793.
- [6] T. (Tommy) Gartlan, “Debouncing Push-Buttons Using a State Machine Approach”, EEWeb, el 6 de junio de 2018. [Online]. Available: <https://www.eeweb.com/debouncing-push-buttons-using-a-state-machine-approach/> [Accessed: 24-Feb- 2022]
- [7] B. says, “Ultimate Guide to Switch Debounce (Part 8)”, EEJournal, el 16 de abril de 2020. [Online]. Available <https://www.eejournal.com/article/ultimate-guide-to-switch-debounce-part-8/>[Accessed: 24-Feb- 2022]
- [8] “Why RTOS and What is RTOS?”, FreeRTOS. [Online]. Available: <https://www.freertos.org/about-RTOS.html> [Accessed: 26-Feb- 2022]
- [9] “An Overview of Packaging for Python — Python Packaging User Guide”. [Online]. Available: <https://packaging.python.org/en/latest/overview/> [Accessed: 27-Feb- 2022]
- [10] “MicroPython language and implementation — MicroPython 1.18 documentation”. [Online]. Available: <https://docs.micropython.org/en/latest/reference/index.html> [Accessed: 27-Feb- 2022]
- [11] “1. Getting started with MicroPython on the ESP32 — MicroPython 1.18 documentation”. [Online]. Available <https://docs.micropython.org/en/latest/esp32/tutorial/intro.html#esp32-intro> [Accessed: 27-Feb- 2022]