

# LABORATORIO DE CONTROL DE PROCESOS DE FLUJO Y NIVEL DE FLUIDOS FPC CONTROLADO A DISTANCIA EN TIEMPO REAL APLICANDO LA TELEINGENIERÍA

**Juan José Guevara Vásquez**

Ingeniero en Electrónica. Docente Investigador de la Escuela de Ingeniería Eléctrica y Electrónica, ITCA-FEPADE Sede Central.

Correo electrónico: [juan.guevara@itca.edu.sv](mailto:juan.guevara@itca.edu.sv)

**Juan José Cáceres Chiquillo**

Ingeniero en Electrónica. Docente Coinvestigador de la Escuela de Ingeniería Eléctrica y Electrónica, ITCA-FEPADE Sede Central.

Correo electrónico: [jcaceres@itca.edu.sv](mailto:jcaceres@itca.edu.sv)

**Carlos Geovanny Meléndez Molina**

Técnico en Ingeniería Eléctrica. Docente Coinvestigador de la Escuela de Ingeniería Eléctrica y Electrónica, ITCA-FEPADE Sede Central.

Correo electrónico: [carlos.melendez@itca.edu.sv](mailto:carlos.melendez@itca.edu.sv)

Recibido: 31/05/2024 - Aceptado: 17/07/2024

## Resumen

En este artículo, se describe el proyecto de investigación y los resultados obtenidos en el diseño y construcción de un laboratorio de experimentación para prácticas de control de procesos industriales de flujo y nivel, el cual fue desarrollado por la Escuela de Ingeniería Eléctrica y Electrónica de ITCA-FEPADE. El objetivo de la investigación fue mejorar la experiencia de aprendizaje y experimentación de estudiantes y profesores en el campo de control de procesos industriales. Se diseñó el hardware y firmware de un circuito controlador electrónico para automatizar y controlar el laboratorio a distancia, utilizando el Internet Industrial de las Cosas IIoT; se interconectó el controlador electrónico a un gateway industrial por medio de Modbus y un Protocolo de Control de Transmisión TCP para el intercambio de información; se creó un dashboard de control en una plataforma de Red Virtual VNET en la nube para monitorizar los datos del proceso y alterar a distancia las condiciones operativas de un controlador Proporcional Integral Derivativo PID. Al dashboard se accede utilizando un navegador web, cuya administración se encuentra centralizada en una plataforma de teleingeniería. Se diseñó un "Manual de Instalación" y un "Manual de Prácticas" de diferentes niveles para que el entrenador pueda ser utilizado para prácticas remotas programadas por estudiantes de carreras técnicas en el área de Ingeniería Electrónica.

## Palabras clave

Control de procesos, control de procesos industriales, control PID, microcontroladores, control de flujo, semiconductores, teleingeniería, Modbus TCP, circuitos electrónicos.

## REAL-TIME REMOTE CONTROLLED FLUID FLOW AND LEVEL PROCESS CONTROL LABORATORY APPLYING TELEENGINEERING

## Abstract

This article describes the research project, and the results obtained in designing and constructing an experimental laboratory for flow and level industrial process control practices, which the Escuela de Ingeniería Eléctrica y Electrónica of ITCA-FEPADE developed. The research objective was to improve students' and professors' learning and experimentation experience in industrial process control. The hardware and firmware of an electronic controller circuit were designed to automate and control the laboratory remotely, using the Industrial Internet of Things IIoT; the electronic controller was interconnected to an industrial gateway via Modbus and a Transmission Control Protocol TCP for information exchange; a control dashboard was created on a cloud-based VNET Virtual Network platform to monitor process data and remotely alter the operating conditions of a Proportional Integral Derivative PID controller. The dashboard is accessed using a web browser, the administration of which is centralized on a teleengineering platform. An "Installation Manual" and a "Practices Manual" of various levels were designed so the trainer can be used for remote practices scheduled by students of technology degree programs in Electronics Engineering.

## Keyword

Process control, industrial process control, PID control, microcontrollers, flow control, semiconductors, teleengineering, Modbus TCP, electronic circuits.

## Introducción

En Educación Superior, el acceso a laboratorios de experimentación suele estar limitado por el alto costo de adquisición de la maquinaria y equipo. El campo del control de procesos industriales no es una excepción, por lo general se dispone únicamente de un entrenador para un control de procesos específico, lo cual reduce el tiempo de acceso para experimentación a estudiantes y profesores. Este proyecto se enmarca en la búsqueda de soluciones que permitan optimizar el tiempo de acceso a los equipos para las prácticas de control de procesos industriales, registrar las actividades que los estudiantes lleven a cabo, generar métricas, que permitan al docente identificar aprendizajes para tomar acciones pedagógicas y alcanzar las competencias de una manera efectiva para el control de procesos industriales aplicando Teleingeniería.

Esta investigación se fundamenta en el trabajo realizado previamente referente al diseño de un laboratorio de experimentación de un proceso industrial de temperatura [1] así como el diseño, análisis y modelado matemático del proceso industrial de flujo y nivel, [2] que tuvo entre sus principales objetivos la elaboración del diagrama esquemático del circuito controlador electrónico y un circuito driver para válvulas proporcionales de control de paso de fluidos. Dentro de la metodología de la investigación se detalla la protección del circuito, la elaboración del circuito impreso PCB, los problemas detectados y las mejoras aplicadas, el diseño del firmware, diseño del dashboard para el control de procesos, así como el montaje del entrenador para pruebas de funcionamiento de las prácticas correspondientes.

## Desarrollo

### METODOLOGÍA DE INVESTIGACIÓN

A partir del diseño del diagrama electrónico del controlador existente se realizó el siguiente proceso de revisión y rediseño:

#### A. Protecciones del circuito

##### 1) Entradas y salidas

Se listaron los riesgos de seguridad del proceso para personal operario y los dispositivos que forman parte del entrenador. Se hicieron modificaciones en los circuitos del controlador electrónico para que sus entradas y salidas estuviesen protegidas por sobretensiones. Se incorporaron dispositivos de protección ESD que han demostrado ser ideales para proteger entradas y salidas [3], [4], [5]. En la Fig. 1, se aprecia una entrada analógica del controlador que está protegida por medio del diodo ESD D15.

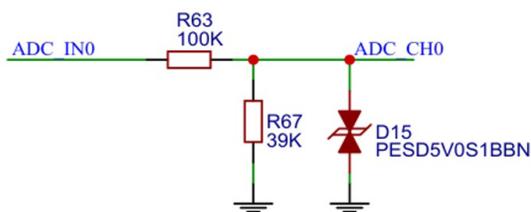


Fig. 1. Entrada analógica para estándar de industria de 0V a 10V con protección ESD por medio de D5V0S1.

##### 2) Inversión de polaridad

Es posible que se inviertan por error las líneas de alimentación del controlador. Por esta razón, resulta conveniente dotar al circuito de alimentación de una etapa previa que lo proteja. Existen diversas y efectivas maneras de proteger circuitos contra la inversión de polaridad, estas van desde el uso de fusibles, hasta semiconductores como diodos rectificadores y Schottky [6], [7]. Sin embargo, el circuito que ha mostrado ser más eficiente y efectivo está formado por un transistor de efecto de campo MOSFET canal P [8], [9], [10]. En la Fig. 2, se muestra el circuito diseñado, que fue sometido a pruebas de inversión mostrando resultados satisfactorios.

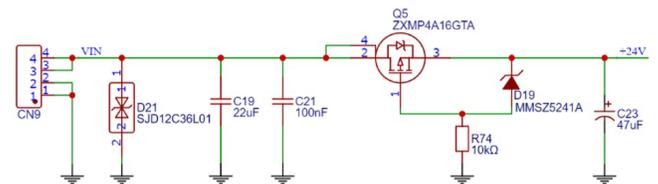


Fig. 2. Circuito de protección contra inversión de polaridad con MOSFET canal P. El circuito también cuenta con protección ESD en la Entrada de Voltaje VIN.

##### 3) Luz de estado

Es importante indicar a los técnicos y demás personal que se encuentra cerca del entrenador, el estado operativo de éste, ya que existe la posibilidad de que esté siendo utilizado y que las personas que se encuentren en el laboratorio no se enteren de esto. Para solventar este requerimiento, se dotó al equipo de una luz de estado que mediante secuencias de apagado/encendido o blinks, indique su estado operativo. Se utilizó una luz táctil de grado industrial con activación PNP, conectada a una salida digital del circuito controlador electrónico, cuya codificación de blinks y su relación con el estado del equipo se indica en la Tabla I.

TABLA I. Patrón de blinks de la luz de estado del entrenador.

PATRÓN DE BLINKS	ESTADO
1	Paro
2	Reiniciando
3	Esperando
4	Iniciando
Apagado	Marcha
Encendido	No energizado

#### B. Elaboración del Circuito Impreso (PCB)

El PCB se diseñó utilizando el programa EasyEDA que es gratuito. La fabricación del PCB así como el montaje de los componentes de montaje superficial (SMD) fue realizada por encargo. Todos los dispositivos electrónicos se eligieron de las librerías de la empresa JLCPCB quien fue la elegida para fabricar el PCB y soldar los componentes SMD. El PCB es multicapa, cuenta con cuatro capas

de cobre. Para su diseño se siguieron las reglas y recomendaciones correspondientes [11, p. 12], [12], [13], [14].

La capa Top que se aprecia en la Fig. 3, se utilizó para conectar líneas de alimentación y de comunicación. El área destinada a la fuente de alimentación posee las pistas de cobre más anchas debido a que por ellas circula una mayor cantidad de corriente.

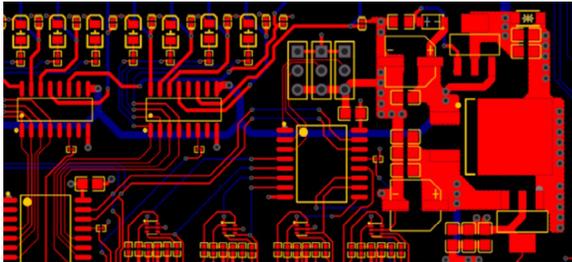


Fig. 3. Capa Top del PCB del controlador electrónico. Entre los diversos circuitos integrados se observan pistas mucho más delgadas (de hasta 0.127 mm) que se han destinado a las líneas de comunicación en las cuales circulan corrientes muy pequeñas.

Bottom, es la capa que conecta las entradas y salidas. La parte inferior del PCB corresponde a las ocho entradas analógicas de 0V a 10V y ocho digitales de +24V tipo NPN. Inner 1, es la tercera capa que se utilizó para distribuir la tensión de +3.3V a todos los dispositivos que se alimentan con este nivel de tensión. Inner 2, es la cuarta capa y corresponde a la referencia a tierra. El estándar extendido Gerber es el más utilizado en la industria manufacturera de PCB [15], por lo que a partir del diseño original de las capas, se procedió a crear los ficheros en este formato, los cuales se enviaron al fabricante de PCB junto con la lista de partes de componentes SMD para el montaje directamente en la fábrica, el resultado se aprecia en la Fig. 4.

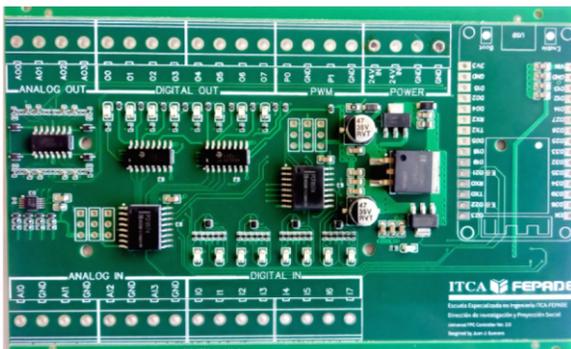


Fig. 4. PCB del controlador electrónico. Los componentes electrónicos SMD fueron soldados de fábrica.

Después de recibir el PCB del fabricante, se soldaron los dispositivos THT que corresponden a los terminales de sujeción con tornillo para las entradas y salidas, así como los conectores pin header macho y hembras. Las pruebas eléctricas consistieron en realizar mediciones de continuidad para verificar la existencia de corto circuitos y aislamiento de las capas. Posteriormente, se energizó el circuito con una fuente de +24V por 5A y se invirtieron las polaridades para

verificar el funcionamiento de las protecciones obteniendo resultados satisfactorios.

### C. Problemas detectados y mejoras aplicadas

Al realizar pruebas de funcionamiento, se determinó que el circuito integrado expensor de puertos PCF8574T presentaba conflictos con los datos en sus entradas y corrientes de excitación insuficientes en sus salidas, lo cual producía un funcionamiento no deseado en las entradas y salidas de 24V. La solución a estos problemas fue sustituir los expansores de entradas y salidas colocando en su lugar un microcontrolador.

Una investigación exhaustiva determinó que el microcontrolador RP2040 puede utilizarse en este tipo de aplicaciones, siendo además compatible con el procesador principal [16], [17], [18]. El RP2040 viene embebido en una placa de desarrollo Raspberry Pi Pico con suficientes entradas y salidas digitales, así como puertos de comunicación configurables y compatibles con el bus I<sup>2</sup>C.

Las funciones principales del RP2040 son comunicarse con el microcontrolador principal ESP32 WROOM32 bajo demanda, cuando se detecte un cambio de estado en las entradas y recibir los cambios de estados del microcontrolador principal para aplicarlos a las salidas digitales. El circuito se montó sobre el PCB del controlador principal como se aprecia en la Fig. 5. Posteriormente se realizaron pruebas eléctricas de compatibilidad siendo estas satisfactorias.

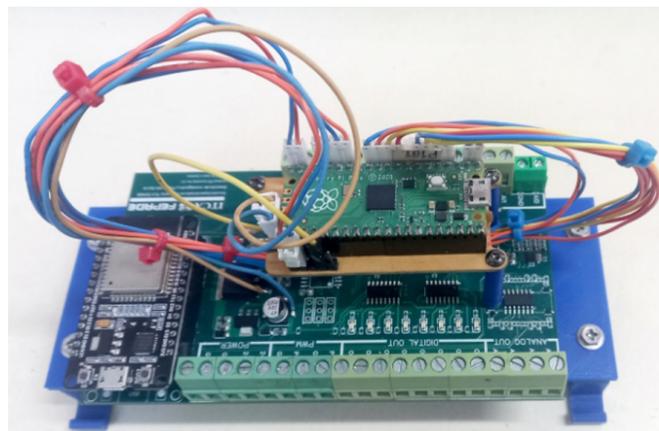


Fig. 5. Montaje del controlador de entradas y salidas en el circuito principal.

### D. Diseño del Firmware

El diseño del firmware o programa del controlador se dividió en dos etapas:

#### 1) Diseño del firmware del controlador de entradas y salidas digitales

El programa se creó utilizando el framework SDK de Raspberry Pi Pico que está basado en el lenguaje C. La plataforma está diseñada para trabajar con MicroPython para crear el firmware de microcontroladores [19], [20]. En este proyecto se utilizó

lenguaje C ya que es más flexible para crear aplicaciones con microcontroladores en donde se manejan múltiples procesos de forma asíncrona y que requieren intercambio de información a través de redes Ethernet [21]. Además, existe una alta disponibilidad de librerías de código abierto.

El firmware de este controlador tiene como función realizar tres tareas:

• *Monitorear el estado de las entradas digitales*

En donde una función temporizada que se ejecuta cada 10 ms, se encarga de monitorear constantemente el estado de las entradas digitales. Cuando se detecta un cambio de estado, se activa una interrupción por hardware que avisa al controlador principal que se ha producido un cambio en las entradas y que este debe responder preguntando por medio del bus I<sup>2</sup>C cuál de las entradas cambió de estado.

• *Cambiar el estado de las salidas digitales*

Una función se encarga de cambiar el estado de las salidas cuando el controlador principal ha enviado un comando con la orden de cambio. Esta función determina cuál es la salida afectada y realiza los cambios de estado correspondientes.

• *Mantener comunicación por el bus I<sup>2</sup>C con el controlador principal*

Para esto se creó una función manejadora de eventos (handler) que se encarga de detectar cuando se produce un evento de comunicación que viene desde el máster del bus I<sup>2</sup>C (el microcontrolador principal), este controlador de eventos es el encargado de determinar si la acción a realizar está relacionada con la lectura del estado de las entradas con algún cambio en las salidas. Es una función asíncrona por lo tanto se ejecuta bajo demanda en el microcontrolador principal. El diagrama de estados del firmware del controlador de entradas y salidas se muestra en la Fig. 6.

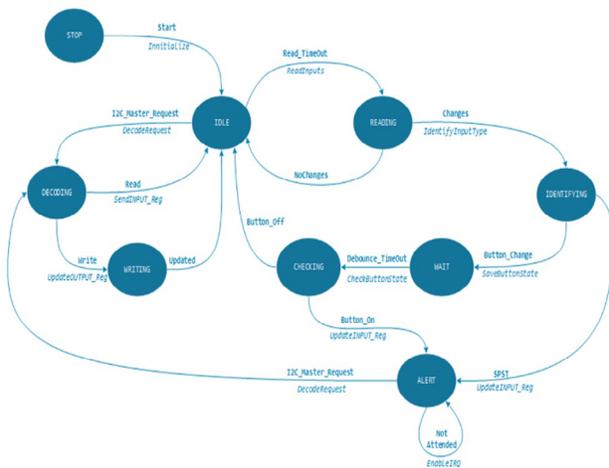


Fig. 6. Diagrama de estados del firmware del controlador de entradas y salidas.

2) *Diseño del firmware del controlador principal*

El objetivo principal de este programa es garantizar que las tareas se realizaran bajo demanda, de forma asíncrona para acciones que requieren acceso a redes de datos o síncronas lanzadas por medio de un temporizador con un período base de 100 ms. Cada vez que se activa el temporizador se lanza una función que activa las banderas que controlan la ejecución de las tareas que se muestran en la Tabla II. Cada tarea tiene a su vez un registro que controla el intervalo de décimas de segundo que deben transcurrir para que esta se pueda ejecutar. Estos valores fueron ajustados experimentalmente de acuerdo con su importancia en el proceso.

TABLA II. Tareas del programa principal del controlador ESP32.

TAREA	ACCIÓN
update_outputs	Actualizar el estado de las salidas digitales y analógicas.
read_adc_value	Leer los canales analógicos activos.
compute_pid	Calcular el valor de respuesta del controlador PID.
check_states	Verificar el estado de máquina del proceso.
refresh_states	Cambiar el estado de máquina del proceso.
modbus_request	Actualizar registros modbus.

Dentro de las tareas del Programa principal están las siguientes:

1) *Manejar la comunicación con el controlador de entradas y salidas*

La comunicación se establece a partir de dos condiciones. La primera es cuando el controlador de entradas y salidas detecta un cambio de estado en sus entradas digitales. En este caso, se activa una interrupción por hardware que pone en alto a PIO26 del ESP32 que corresponde a la interrupción PCF01\_INT. Esto hace que se active una bandera para indicar que en la siguiente transacción I<sup>2</sup>C se debe determinar que entrada ha sido alterada.

La segunda condición se produce cuando se cambia el estado de las salidas como parte del proceso que se está controlando ya sea de forma automatizada (establecido en los estados de máquina del proceso) o por acción directa del usuario. Para tal efecto, se programó la acción de cambio utilizando registros de estado actual y anterior. Ambos registros se comparan cada 100 ms, en caso de existir diferencias, se comunica vía I<sup>2</sup>C al controlador de entradas y salidas que realice un cambio de estado.

2) *Leer las entradas analógicas*

El controlador fue diseñado para manejar hasta cuatro entradas analógicas con una resolución de 16 bits lo que equivale a 65536 valores discretos. En la Fig. 7 se muestra el circuito utilizado que está basado en un convertidor analógico a digital ADS1115 de baja potencia tipo SAR, que ha mostrado ser efectivo para este tipo de aplicaciones [22], [23].

Para controlar al circuito conversor, se utilizó la librería ADS1X15 que es de acceso y uso libre [24]. Una bandera se utiliza para indicar el momento de leer las entradas analógicas. Cada entrada se activa y configura durante el arranque, lo cual depende de cuantos circuitos estén conectados al controlador. La lectura de los canales analógicos se realiza cuando el tiempo transcurrido es igual al establecido en una constante cuyo valor por defecto es de diez, lo que equivale a una lectura cada un segundo. Al primer canal analógico se conectó un sensor ultrasónico de grado industrial Pepperl Fuchs 3RG6343 que envía información sobre el nivel del agua en el tanque 2 (Fig. 8) donde se realiza la acción de control. Mientras que en el segundo canal se conectó un sensor de flujo SMC PF2W720 que provee información sobre el caudal en litros por segundo del agua que circula por el sistema.

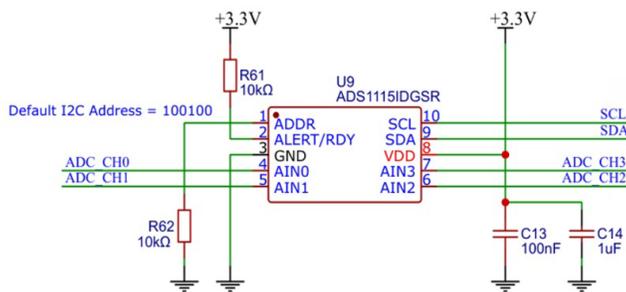


Fig. 7. Circuito convertidor de analógico a digital de 16 bits con ADS1115.

3) *Controlar las salidas analógicas y de alta velocidad PWM*

El controlador posee dos salidas de alta velocidad PWM y cuatro analógicas. Las salidas analógicas proveen señales de 0V a 10V. A una de las salidas se conectó la válvula proporcional de flujo con control Burkert type 1094 para realizar el proceso de regulación del caudal. Para aumentar o disminuir el voltaje de salida analógico, se varía el ciclo de trabajo de la señal PWM por medio de una función que también define la frecuencia de la señal.

4) *Mantener comunicación con el gateway industrial Vbox*

Se utilizó Modbus TCP para establecer comunicación con el gateway industrial Vbox ya que se dispone de la librería Modbus-esp8266 [25], es un protocolo bien documentado [26] y ampliamente utilizado en aplicaciones industriales [27], [28], [29].

Modbus TCP funciona por sobre TCP/IP con registros de intercambio, en este proyecto se utilizaron registros tipo bobina (Coils) para lanzar o detener acciones y actuadores todo o nada, registros variables (IREG) para el intercambio de valores enteros de hasta 16 bits como son el flujo y el nivel que cambian en función del tiempo y, registros de retención (HREG) para valores enteros de 16 bits que no cambian constantemente como las variables que controlan el proceso. En la Tabla III se muestran los registros utilizados, funcionalidad y direcciones programadas.

5) *Realizar cálculos del controlador PID*

En el control de procesos continuos se utilizó el controlador PID. Este controlador se implementó con una librería de Arduino creada y distribuida bajo licencia de MIT [30] que implementa la fórmula 1, la cual es una ecuación diferencial ordinaria lineal con lazo cerrado en la señal de error [31], [32].

Fórmula 1:

$$Output = K_p e(t) + K_i \int e(t)dt + K_d \frac{d}{dt} e(t)$$

De acuerdo con la fórmula, para realizar un control básico, el valor de Kp debe ser de un valor superior a cero, pero tanto Ki como Kd si pueden ser cero, con esto se logra un control proporcional siempre y cuando el Set Point sea un valor superior a cero, y se encuentre dentro de los límites del tanque de agua cuyo nivel no debe exceder los 250 mm. El límite de la salida se estableció en 100, que equivale a la máxima apertura de la válvula proporcional. Una constante que posee un valor por defecto de 10 permite que los cálculos se realicen cada segundo.

TABLA III. Registros MODBUS TCP utilizados para la gestión del proceso industrial.

VARIABLE	DESCRIPCIÓN	DIRECCIÓN DE REGISTRO MODBUS	TIPO DE REGISTRO MODBUS
SP_M	SP Controller Memory	200	IREG
PV	Real Time Level	210	IREG
P_M	P Controller Memory	220	IREG
I_M	I Controller Memory	230	IREG
D_M	D Controller Memory	240	IREG
FLOW	Proportional Valve Real Time Flow in l/min	250	IREG
PUMP	Water Pump Power in %	260	IREG
SV1_M	Solenoid Valve 1 Status	270	IREG
SV2_M	Solenoid Valve 2 Status	280	IREG
UP_LS	Up Water Level Sensor	290	COIL
DN_LS	Down Water Level Sensor	300	COIL
START_M	Plant Start Status	320	COIL
STOP_M	Plant Stop Status	330	COIL
RESET_M	Plant Reset Status	340	COIL
EMRG	Plant Emergency Flag	350	COIL

VARIABLE	DESCRIPCIÓN	DIRECCIÓN DE REGISTRO MODBUS	TIPO DE REGISTRO MODBUS
SV1	Solenoid Valve 1 ON/OFF Command	360	I REG
SV2	Solenoid Valve 2 ON/OFF Command	370	I REG
MODE_L	Plant Operation Mode Level	380	I REG
MODE_F	Plant Operation Mode Flow	390	
START	Plant Start Command	400	COIL
STOP	Plant Stop Command	410	COIL
RESET	Plant Reset Command	420	COIL
SP	Desired Water Level in mm	430	H REG
P	Proportional Value for PID	440	H REG
I	Integral Value for PID	450	H REG
D	Derivative Value for PID	460	H REG

Una bandera de estado indica al procesador que el controlador PID debe funcionar solo cuando el estado del proceso es "ejecutando". En este caso el procesador debe empezar a realizar los cálculos a partir de los valores de las variables SP, Kp, Ki, y Kd.

SP se actualiza cada segundo con el nivel actual en el tanque en donde se está realizando la acción de control.

6) Manejar los estados de máquina del proceso

En la Fig. 8 se muestra el esquema general del proceso del entrenador de flujo y nivel.

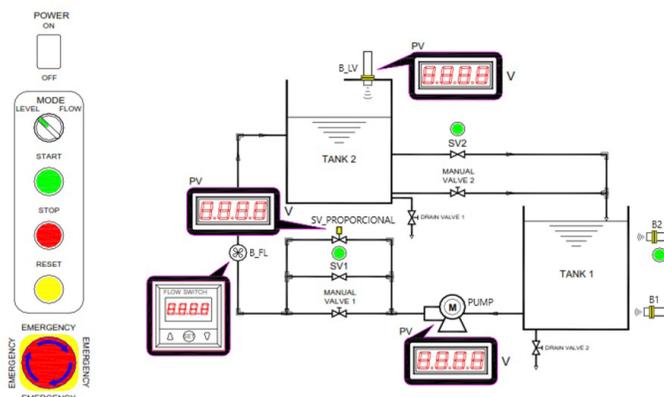


Fig. 8. Esquema del Proceso Industrial de Flujo y Nivel FPC.

A partir del esquema del proceso FPC, se definieron cinco estados:

• INICIO O PARO

Es el estado al que entra la máquina (entrenador) cuando es energizada por primera vez o cuando se activa el interruptor de paro o STOP. En esta condición, el indicador luminoso de estado hace blink de forma continua. Se puede pasar a estado de inicio en cualquiera de los otros cuatro estados de máquina.

• REINICIANDO

Si la máquina se encuentra en estado de inicio se puede activar el estado reiniciando. En este estado, se descarga el líquido del tanque 2 y se detiene; se pasa al siguiente estado hasta que los sensores capacitivos B1 (nivel inferior) y B2 (nivel superior) del tanque 1, se encuentren activados simultáneamente. Para descargar el líquido del tanque 2, se abre la electroválvula SV2 lo cual es indicado por la intermitencia o blinks del indicador de estado en el tablero principal. En el estado reiniciando, el indicador luminoso de la máquina hace dos blinks.

• ESPERA

Después del proceso de reinicio, se espera a que el operario establezca los parámetros mínimos de control, que son el establecimiento del nivel deseado SP y el control proporcional P. Mientras esta condición no se cumpla solamente se puede pasar a estado de inicio. El indicador de estado hace tres blinks.

• INICIANDO

Es un estado temporal con transición automática al siguiente y su finalidad es indicar al operario que la máquina ya se encuentra lista para entrar en estado de ejecución o marcha. El indicador de estado hace cuatro blinks.

• EJECUTANDO O MARCHA

En este estado, se ejecuta el proceso industrial tratando de hacer llegar el nivel del líquido en el tanque 2 al nivel establecido en el SP. El controlador PID funciona y su salida de control se pasa a la salida analógica correspondiente para controlar el grado de apertura de la válvula proporcional de flujo. El operario puede actualizar en cualquier momento los parámetros de control o variables SP, P, I y D, para poner a prueba el sistema. El indicador de estado se encuentra encendido.

E. Diseño de dashboard para el control del proceso

El diseño del dashboard se hizo utilizando las herramientas que proporciona la plataforma VNET que es utilizada por el gateway Vbox para intercambiar los datos del proceso. Primero, se crearon todas las etiquetas o tags en VNET que corresponden a los registros Modbus de la Tabla III. A continuación, se verificó que todos los tags tuviesen conectividad activa desde VNET hacia el controlador electrónico.

Posteriormente, se diseñó un dashboard de pruebas, cuyo objetivo fue comprobar el funcionamiento de cada uno de los módulos del firmware del controlador a medida que estos fueron incorporándose.

El uso del dashboard de pruebas detectó que era necesario crear variables de memoria que retroalimenten al usuario remoto sobre los datos almacenados en las variables. Esto permite al estar seguro de que el controlador ha recibido los parámetros que se han enviado desde VNET.

Finalmente, se elaboró el dashboard del control del proceso que se muestra en la Fig. 9, que incorpora todos los elementos de control correspondientes.

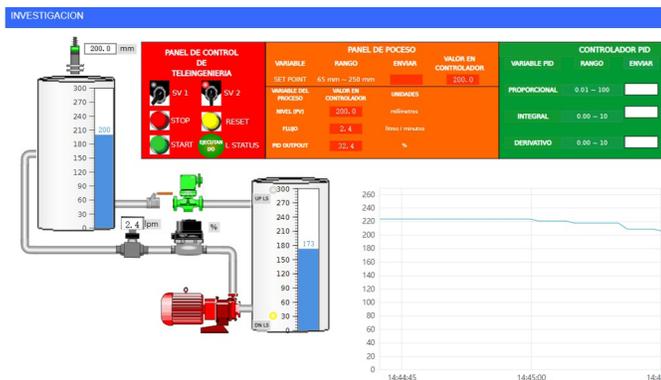


Fig. 9. Dashboard del proceso industrial de flujo y nivel creado en la plataforma industrial VNET.

## F. Montaje del entrenador

Se diseñó una base compatible con riel DIN para montar la tarjeta electrónica del controlador y se instaló en el tablero de control y potencia del entrenador. Se creó una estructura metálica para montar las cámaras IP, los dispositivos de la red de datos y la instalación eléctrica como se muestran en la Fig. 10. Con la puesta en marcha del laboratorio de experimentación FPC, se procedió a realizar ajustes en el funcionamiento.



Fig. 10. Estructura para montaje de cámaras y dispositivos de red de datos del entrenador.

## Resultados

- Circuito Controlador Electrónico Principal de propósito general, que permite la gestión a distancia y en tiempo real del entrenador FPC de flujo y nivel.

- Panel de Control Web o Dashboard que permite monitorear y realizar ajustes a distancia de un control de procesos industriales de flujo y nivel.
- Manual de Prácticas para el laboratorio de control de procesos industriales de flujo y nivel utilizando Teleingeniería.
- Manual de Instalación para el laboratorio de control de procesos industriales de flujo y nivel.

## Conclusiones

1. Se puede diseñar un controlador electrónico a la medida de los requerimientos, cumpliendo requisitos de control, de señales de entrada-salida y demás características que poseen los controladores industriales existentes en el mercado.
2. Es posible monitorear y manipular las variables de proceso de forma remota, usando el protocolo de comunicación industrial Modbus TCP, implementándolo en un controlador electrónico utilizando librerías de código abierto.
3. La plataforma en la nube VNET, facilita el trabajo de diseño de los Dashboard de control de procesos.
4. Se pueden monitorear y manipular las variables de proceso de forma remota utilizando el protocolo de comunicación industrial MODBUS TCP implementándolo en un controlador electrónico con librerías de código abierto.
5. La plataforma con ejecución basada en la nube VNET facilita el trabajo de diseño de los Dashboard de control de procesos, permitiendo a los ingenieros concentrarse en los mecanismos de comunicación y actualización de los datos.
6. Es posible interactuar directamente con el entorno virtual, ajustar parámetros y observar las respuestas en tiempo real, lo que fomenta la experimentación. Esto facilita la retención del conocimiento y mejora las habilidades de resolución de problemas.

## Referencias

- [1] J. J. Guevara, J. J. Cáceres, "Diseño de plataforma de teleingeniería para prácticas de laboratorio a distancia: control de proceso industrial de temperatura", Revista Tecnológica ITCA-FEPADE, no. 15, pp. 19-22. Ene./Dic. 2022.
- [2] J. J. Guevara, et al, "Diseño matemático y modelado de componentes de control de proceso de flujo y nivel FPC para integrar un simulador virtual con realidad aumentada y control a distancia por teleingeniería", Revista Tecnológica, no. 16, pp. 12-16, ene. 2023, Accedido: el 15 de mayo de 2024. [En línea]. Disponible en: <http://redicces.org.sv/jspui/handle/10972/5134>
- [3] E. Rosenbaum, et al, "Compact Models for Simulation of On-Chip ESD Protection Networks", IEEE Trans. Electron Devices, vol. 71, núm. 1, pp. 151-166, ene. 2024, doi: 10.1109/TED.2023.3320093. [En línea]. Disponible en: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10292739>

- [4] Texas Instruments, "System-Level ESD Protection Guide", 2022. [En línea]. Disponible en: <https://www.ti.com/lit/sg/sszb130d/sszb130d.pdf?ts=1716272366714>
- [5] Solutions Cubed, "Protecting Inputs in Digital Electronics", Digikey. Consultado: el 15 de mayo de 2024. Abr. 2012. Disponible en: <https://www.digikey.com/en/articles/protecting-inputs-in-digital-electronics>
- [6] Cadence Design Systems, Inc. "Simple Solutions for Reverse Polarity Protection", Cadence. Consultado: el 23 de mayo de 2024. [En línea]. Disponible en: <https://resources.pcb.cadence.com/blog/simple-solutions-for-reverse-polarity-protection>
- [7] R. Keim, "Reverse Polarity Protection: How to Protect Your Circuits Using Only a Diode, All about circuits. Consultado: el 23 de mayo de 2024. 28 Jun. 2018 [En línea]. Disponible en: <https://www.allaboutcircuits.com/technical-articles/how-to-protect-your-circuits-using-only-a-diode/>
- [8] Components. "Design Guide - PMOS MOSFET for Reverse Voltage Polarity Protection", Components101. Consultado: el 12 de septiembre de 2023. [En línea]. Disponible en: <https://components101.com/articles/design-guide-pmos-mosfet-for-reverse-voltage-polarity-protection>
- [9] Ly-Minh-Duy, "Analytical Expressions of Time-Domain Responses of Protection Circuits to ISO Reverse Transients in Automotive Applications", IEEE Xplore, vol.10, pp. 1-9, May. 2022, doi: 10.1109/ACCESS.2022.3176639. [En línea]. Disponible en: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9779151>
- [10] "Designing a Reverse Polarity Protection Circuit (Part I) | Article | MPS", Por MPS Proprietary Information (07 Jul. 2022). Article #W045 Rev. 1.0 Consultado: el 23 de mayo de 2024. [En línea]. Disponible en: <https://www.monolithicpower.com/designing-a-reverse-polarity-protection-circuit-part-i>
- [11] H. Zumbahlen, "Chapter 12: Printed Circuit Board Design Issues", en Linear Circuit Design Handbook, H. Zumbahlen, 1a ed., Estados Unidos: Analog Devices, 2007. no.12., pp. 1-97.
- [12] JLCPCB, "Advancing Your Circuit Design Skills: Tips and Techniques for Improving Performance". JLCPCB. Consultado: el 23 de mayo de 2024. 26 Jul. 2023 [En línea]. Disponible en: <https://secure.livechatinc.com/>
- [13] D. Knight. "Practical PCB Layout Tips Every Designer Needs to Know - Technical Articles". All about circuits. Consultado: el 23 de mayo de 2024. 13 no. 2015. [En línea]. Disponible en: <https://www.allaboutcircuits.com/technical-articles/practical-pcb-layout-tips/>
- [14] Cadence Design Systems, Inc. "PCB Layers Explained: Multilayer Boards and Stackup Rules". Cadence. Consultado: el 23 de mayo de 2024. [En línea]. Disponible en: <https://resources.pcb.cadence.com/blog/2019-pcb-layers-explained-multilayer-boards-and-stackup-rules>
- [15] The gerber Layer Format Specification, UCAMCO, 2021. pp.1-203. [En línea]. Disponible en: [https://www.ucamco.com/files/downloads/file\\_en/416/the-gerber-layer-format-specification-revision-2021-02\\_en.pdf](https://www.ucamco.com/files/downloads/file_en/416/the-gerber-layer-format-specification-revision-2021-02_en.pdf)
- [16] Z. C. M. Davidson, S. Dang, y X. Vasilakos, "Blended Laboratory Design Using Raspberry Pi Pico for Digital Circuits and Systems", IEEE Trans. Learn. Technol., vol. 17, pp. 1170-1183, 7 Feb. 2024, Consultado: 23 de mayo de 2024. doi: 10.1109/TLT.2024.3363230. [En línea]. Disponible en: <https://ieeexplore.ieee.org/document/10423829>
- [17] Raspberry Pi documentation. "Raspberry Pi Documentation - Raspberry Pi Pico and Pico W". Raspberry Pi. Consultado: el 22 de febrero de 2024. 2012-2024. [En línea]. Disponible en: <https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html>
- [18] S. Yamanoor; S. Yamanoor, "Raspberry Pi Pico DIY Workshop: Build exciting projects in home automation, personal health, gardening, and citizen science", IEEE Xplore, 2022. [En línea]. Disponible en: <https://ieeexplore.ieee.org/document/10163107>
- [19] G. Gaspar, P. Fabo, M. Kuba, J. Flochova, J. Dudak and Z. Florkova "Development of IoT applications based on the MicroPython platform for Industry 4.0 implementation", en 2020 19th International Conference on Mechatronics - Mechatronika (ME), Prague, Czech Republic dic. 2020, pp. 1-7. doi: 10.1109/ME49197.2020.9286455
- [20] M. Popovic, M. Djukic, I. Basicovic, y M. Popovic, "DPSTM-PS Based Edge Network with MicroPython Based IoTs", en 2023 31st Telecommunications Forum (TELFOR), Belgrade, Serbia, nov. 2023 - ene 2024, pp. 1-4. doi: 10.1109/TELFOR59449.2023.10372679
- [21] V. M. Ionescu y F. M. Enescu, "Investigating the performance of MicroPython and C on ESP32 and STM32 microcontrollers", en 2020 IEEE 26th International Symposium for Design and Technology in Electronic Packaging (SIITME), Pitesti, Romania, Oct./Dic 2020, pp. 234-237. doi: 10.1109/SIITME50350.2020.9292199.
- [22] T. Li, Y. Zhang, y Y. Ni, "An Ultra-fast ADC Linearity Test and Calibration Method", en 2019 IEEE 13th International Conference on Anti-counterfeiting, Security, and Identification (ASID), Xiamen, China, oct. - dic. 2019, pp. 282-285. doi: 10.1109/ICASID.2019.8925271.
- [23] N. Yan, C. Kang, G. Mu, S. Chen, M. Wang, y H. Min, "A 10-bit 16-MS/s Ultra Low Power SAR ADC for IoT Applications", en 2018 14th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), Qingdao, China, Oct./Dic. 2018, pp. 1-3. doi: 10.1109/ICSICT.2018.8564872.
- [24] R. Tillaart, "RobTillaart/ADS1X15". RobTillaart. Consultado: el 22 de febrero de 2024. [En línea]. Disponible en: <https://github.com/RobTillaart/ADS1X15>
- [25] A. Emelianov. "modbus-esp8266/examples/TCP-ESP at master · emelianov/modbus-esp8266", GitHub. Consultado: el 22 de febrero de 2024. [En línea]. Disponible en: <https://github.com/emelianov/modbus-esp8266/tree/master/examples/TCP-ESP>
- [26] Modbus Organization, "Modbus Messaging on TCP/IP Implementation Guide V1.0b". Modbus Organization, 24 oct. 2006. [En línea]. Disponible en: [https://modbus.org/docs/Modbus\\_Messaging\\_Implementation\\_Guide\\_V1\\_0b.pdf](https://modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf)
- [27] M. Dwiyaniti, Y. E. S. Situmorang, Y. D. Handoyo, H. Setiana, y A. K. Wardhany, "Implementation of IoT on PLC-Based Induction Motor Speed Control", en 2023 International Conference on Converging Technology in Electrical and Information Engineering (ICCTEIE), Bandar Lampung, Indonesia, oct.-dic 2023, pp. 118-123. doi: 10.1109/ICCTEIE60099.2023.10366716.
- [28] P. Nguyen-Hoang y P. Vo-Tan, "Development An Open-Source Industrial IoT Gateway", en 2019 19th International Symposium on Communications and Information Technologies (ISCIT), Ho Chi Minh City, Vietnam, sep.-nov. 2019, pp. 201-204. doi: 10.1109/ISCIT.2019.8905157.
- [29] N. A. Reyes y H. I. Cerrato, "Modbus TCP Bridging for Interconnecting Non-Compatible Devices in the Energy Sector Using Node-RED and Edge Computing", en 2023 IEEE 41st Central America and Panama Convention (CONCAPAN XLI), Tegucigalpa, Honduras, nov. 2023, pp. 1-4. doi: 10.1109/CONCAPANXLI59599.2023.10517535.
- [30] br3ttb, "br3ttb/Arduino-PID-Library". el 26 de febrero de 2024. Consultado: el 27 de febrero de 2024. [En línea]. Disponible en: <https://github.com/br3ttb/Arduino-PID-Library>
- [31] L. Wang, "Basics of PID Control", en PID Control System Design and Automatic Tuning using MATLAB/Simulink, IEEE, 2020, pp. 1-30. doi: 10.1002/9781119469414.ch1.
- [32] F. Zayas Gato, et al, Diseño de Controladores PID, 1a ed. España: Universidad da Coruña, 2020. [En línea]. Disponible en: [https://ruc.udc.es/dspace/bitstream/handle/2183/25824/Zayas\\_Gato\\_2020\\_Dise%C3%B1o\\_de\\_controladores\\_PID.pdf](https://ruc.udc.es/dspace/bitstream/handle/2183/25824/Zayas_Gato_2020_Dise%C3%B1o_de_controladores_PID.pdf)