

CONTROLADOR PID DE NIVEL DE AGUA

Kevin Erney Certuche Campo
Kevin.certuche.c@uniautonomo.edu.co
Uniautonomo del cauca [05/04/2024]

I. INTRODUCCION

La ingeniería de control desempeña un papel crucial en el diseño y funcionamiento de una amplia gama de sistemas, desde procesos industriales hasta dispositivos domésticos. En este contexto, el presente estudio se centra en la implementación de un controlador de nivel de agua, un problema clásico en el campo de los sistemas de control, que tiene aplicaciones significativas en diversas industrias y áreas de la vida cotidiana.

El control preciso del nivel de agua es fundamental en numerosos escenarios, desde el control de tanques de almacenamiento hasta la regulación de sistemas de riego agrícola. Sin embargo, este proceso presenta desafíos únicos, como la selección adecuada de sensores, la sintonización óptima del controlador y la consideración de la dinámica inherente del sistema.

En este informe, se explorarán las diversas etapas involucradas en el diseño e implementación de un controlador de nivel de agua. Se abordarán aspectos clave, como la descripción del sistema, la selección de la instrumentación apropiada y la aplicación de técnicas de control para lograr un desempeño eficiente y estable.

A través de este análisis, se busca no solo resolver los desafíos específicos asociados con el control de nivel de agua, sino también proporcionar una comprensión más amplia de los principios fundamentales de los sistemas de control y su aplicación en el mundo real.

II. DESCRIPCIÓN DEL SISTEMA

a) Funcionamiento de la Planta

1. *Descripción general del sistema:* La planta consta de dos tanques, cada uno equipado con dos mini bombas sumergibles. Uno de los tanques actúa como el tanque controlado, mientras que el otro puede considerarse como un tanque de suministro. En el tanque controlado, se encuentra un sensor que mide el nivel de agua presente. El propósito de la planta es mantener el nivel de agua en el tanque controlado lo más cerca posible del valor deseado, conocido como el Setpoint.

2. Fotografía de la planta

3. Identificación y descripción de:

Variable controlada: El nivel de agua en el tanque controlado es la variable que se está controlando y monitoreando continuamente.

Variable de referencia: El Setpoint, proporcionado por el potenciómetro, establece el nivel deseado al cual se quiere que el sistema mantenga el nivel de agua.

Variable manipulada (esfuerzo de control): El flujo de agua, controlado por las mini bombas sumergibles, es la variable manipulada. Se ajusta para mantener el nivel de agua en el tanque controlado cerca del Setpoint.

Señal de error: La señal de error se calcula comparando la variable de referencia (Setpoint) con la variable controlada (nivel medido por el sensor). Esta diferencia representa la discrepancia entre el nivel deseado y el nivel real del agua en el tanque controlado.

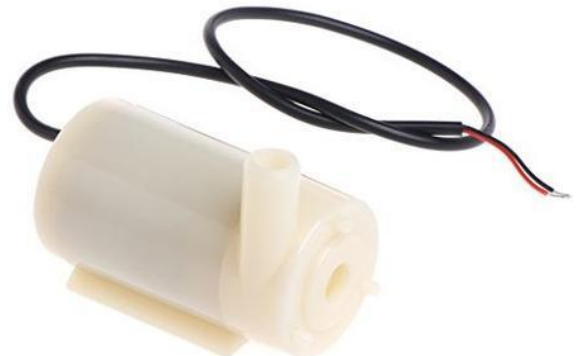
b) Instrumentación utilizada

1. Sensor: HC-SR04



El sensor HC-SR04 es un módulo compacto y de bajo costo que permite medir distancias de forma precisa mediante ultrasonidos. Este sensor funciona emitiendo un pulso de sonido de alta frecuencia y luego midiendo el tiempo que tarda el eco en regresar al sensor. La distancia al objeto se calcula utilizando la velocidad del sonido. [1]

2. Actuadores: Mini bomba de agua sumergible



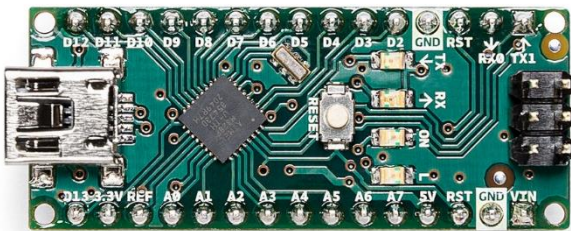
La mini bomba de agua sumergible es un dispositivo compacto y versátil diseñado para bombear agua. Funciona mediante un motor eléctrico que impulsa un impulsor, generando presión para mover el agua a través de un conducto de salida sumergido. [1]

3. Transistor: TIP31C NPN



El Tip 31 es un transistor de potencia que se utiliza comúnmente para controlar cargas de alta corriente mediante una señal PWM (modulación de ancho de pulso, por sus siglas en inglés). Este componente es ideal para aplicaciones donde se requiere regular la velocidad de motores (para este proyecto las bombas de agua). El Tip 31 es un transistor bipolar de tipo NPN que actúa como un interruptor electrónico. Cuando se aplica un voltaje adecuado a su base, permite el paso de corriente desde el colector hasta el emisor, actuando como un interruptor cerrado. El control PWM se logra al aplicar una señal de entrada de pulso digital a la base del transistor. Al variar el ancho de los pulsos (es decir, el tiempo durante el cual la señal está en alto), se puede controlar la proporción de tiempo durante el cual el transistor está activado, lo que a su vez controla la cantidad de corriente que pasa a través de la carga conectada al colector. [1]

4. Microcontrolador utilizado: Arduino NANO



Arduino Nano es una placa pequeña, completa y compatible con placas de pruebas basada en ATmega328 (Arduino Nano 3.x). Tiene más o menos la misma funcionalidad que Arduino Duemilanove, pero en un paquete diferente. Sólo carece de un conector de alimentación de CC y funciona con un cable USB Mini-B en lugar de uno estándar. [2]

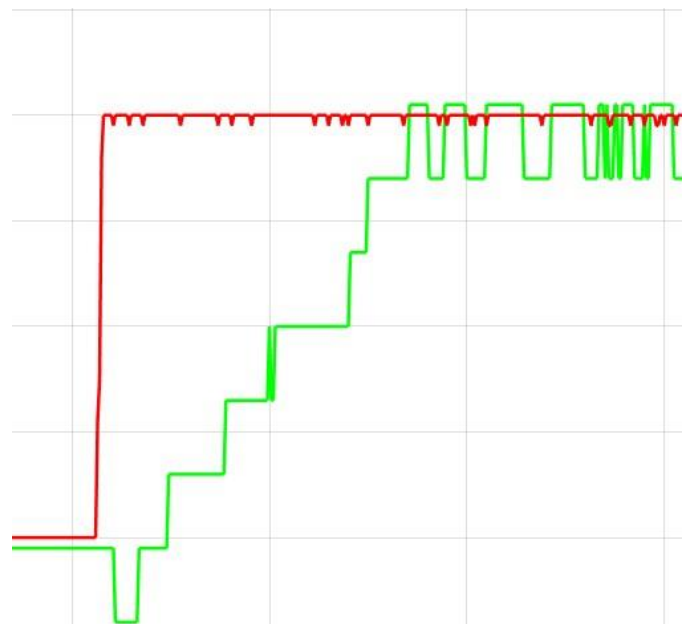
c) Software del Sistema

1. Interfaz Hombre-Máquina en MATLAB: La interfaz hombre-máquina en MATLAB proporciona una representación visual integral del sistema de control PID de nivel. A través de esta interfaz, se muestran dos gráficas fundamentales: una que refleja el nivel actual del agua en el tanque controlado y otra que exhibe el Setpoint actual, el cual es suministrado por un Potenciometro. Estas gráficas son esenciales para brindar una visión clara y en tiempo real del estado del sistema y del objetivo deseado.
2. Comunicación Serial Microcontrolador/MATLAB: La comunicación serial entre el microcontrolador y MATLAB es esencial para la transferencia de datos críticos que sustentan el funcionamiento del sistema. Para lograr esta comunicación de manera efectiva, se establece un protocolo simple pero robusto. Los datos se envían en forma de cadenas de texto con un formato específico, donde el nivel actual y el Setpoint están separados por un delimitador ",". Por ejemplo, una cadena de datos podría tener el formato "nivel, Setpoint" (por ejemplo, "10,15" indica un nivel actual de 10 unidades y un Setpoint de 15 unidades). MATLAB, a su vez, utiliza funciones de lectura para recibir los datos de retroalimentación. Una vez que MATLAB recibe los datos del microcontrolador, procesa la información para actualizar las gráficas de la interfaz hombre-máquina en tiempo real.

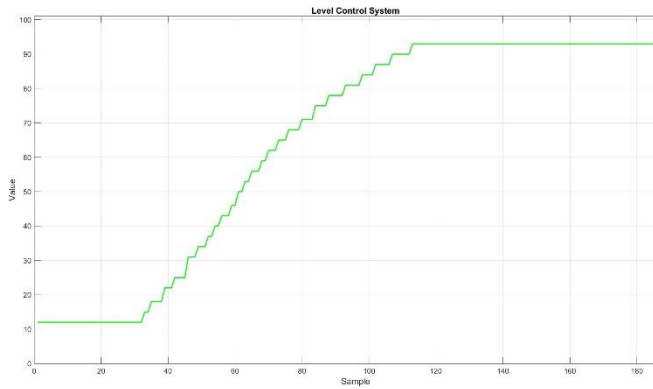
III. MODELO MATEMÁTICO DEL SISTEMA

- a) *Gráfica y descripción de la curva real de respuesta en lazo abierto con una señal escalón de entrada.*

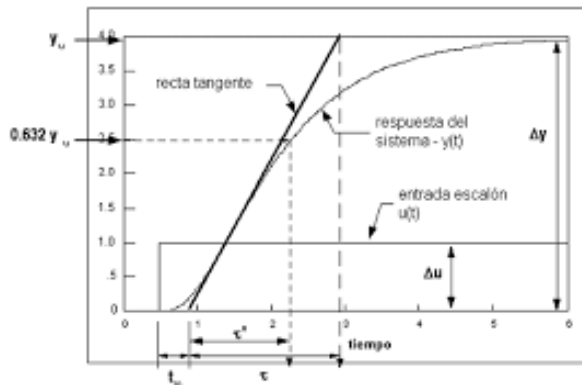
Curva real obtenida



Curva con modificación en sus valores



b) Cálculo de $G(s)$



$$G_p(s) = \frac{K e^{-\theta s}}{\tau s + 1}$$

$$G_p(s) = \frac{3.09 e^{-4.2s}}{21.2s + 1}$$

IV. ESQUEMA DE CONTROL PROPUESTO

El esquema de control realimentado propuesto para el sistema de control PID de nivel se basa en el principio de ajuste continuo de la salida del sistema para mantener el nivel de agua en el tanque controlado lo más cerca posible del valor deseado, o Setpoint:

a) *Retroalimentación del Sensor de Nivel*

El esquema de control comienza con la retroalimentación del sensor de nivel, que proporciona información en tiempo real sobre el nivel actual de agua en el tanque controlado. Esta retroalimentación es esencial para monitorear el estado del sistema y determinar si el nivel de agua está en línea con el Setpoint deseado.

b) *Comparación con el Setpoint*

La señal de nivel medida por el sensor se compara con el Setpoint proporcionado por el usuario. La diferencia entre el nivel medido y el Setpoint deseado se conoce como el error.

c) *Controlador PID*

El error calculado se introduce en el controlador PID (Proporcional-Integral-Derivativo), que es el núcleo del sistema de control. El controlador PID utiliza el error para calcular una señal de control que ajusta el flujo de agua a través de las mini bombas sumergibles. El controlador PID ajusta la salida del sistema de manera proporcional al error (P), integral del error acumulado en el tiempo (I) y la derivada del error con respecto al tiempo (D), lo que permite una respuesta precisa y rápida a las variaciones en el nivel de agua.

d) *Ajuste del Flujo de Agua*

La señal de control generada por el controlador PID se utiliza para ajustar el flujo de agua a través de las mini bombas sumergibles en el tanque controlado. Al aumentar o disminuir el flujo de agua, el sistema busca minimizar el error y mantener el nivel de agua lo más cercano posible al Setpoint.

e) *Ciclo de Retroalimentación Continuo*

Este proceso de comparación, cálculo y ajuste se repite continuamente en un ciclo de retroalimentación cerrado, lo que permite al sistema adaptarse dinámicamente a cambios en el nivel de agua y mantener una operación estable y precisa en todo momento.

V. CONTROLADOR PID

I. *Descripción del Controlador PID*

Para implementar el controlador PID en el código implementado, primero definimos las acciones proporcionales, integrales y derivativas de manera individual y luego combinamos estas acciones para calcular la señal de control

1. *Acción Proporcional (P):*

```
// En la función Compute():
Output = kp * error + ITerm - kd * dInput;
```

2. *Acción Integral (I):*

```
// En la función Compute():
ITerm += (ki * error);
```

3. *Acción Derivativa (D):*

```
// En la función Compute():
double dInput = (Input - lastInput);
Output = kp * error + ITerm - kd * dInput;
```

4. Discretización del Controlador PID: La discretización se maneja mediante el tiempo de muestreo (SampleTime) definido en el código. Los cálculos se realizan en función del tiempo de muestreo para asegurar que el controlador opere en intervalos discretos.

5. *Ajuste de Parámetros del Controlador PID:* Los parámetros del controlador PID, como K_p , K_i y K_d , se pueden ajustar utilizando las funciones SetTunings()

para establecer las ganancias del controlador. La función `SetSampleTime()` se utiliza para ajustar el tiempo de muestreo del controlador.

6. *Modo de Operación del Controlador PID:* El controlador puede funcionar en modo automático o manual, dependiendo de la variable `inAuto`. La función `SetMode()` se utiliza para cambiar entre los modos de operación, y la función `Initialize()` se llama cuando se cambia de modo manual a automático para inicializar los parámetros del controlador.

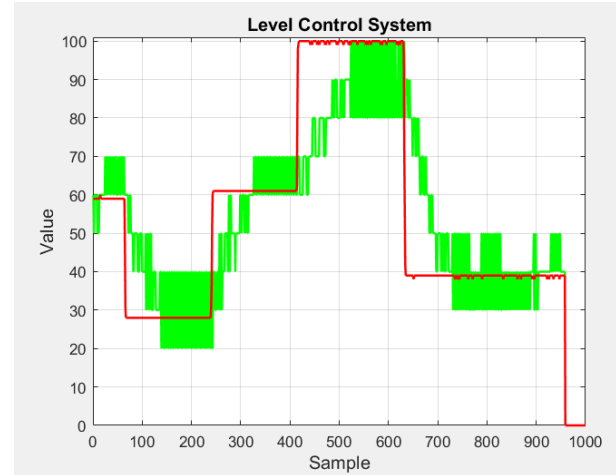
II. Procedimiento de Sintonización

El proceso de sintonización del controlador PID utilizando el método de curva de reacción implica una serie de pasos diseñados para ajustar los parámetros del controlador y optimizar el rendimiento del sistema. Inicialmente, se establecen los objetivos de rendimiento deseados, como tiempos de respuesta rápidos y mínimos sobrepasos. Luego, se inicia un experimento donde se aplica un cambio en el Setpoint del sistema o se introduce una perturbación controlada. Durante este experimento, se observa y registra la respuesta del sistema, incluidas variables como el tiempo de respuesta y la estabilidad. Los datos recopilados se analizan para comprender la dinámica del sistema y se identifican posibles áreas de mejora. A partir de este análisis, se ajustan los parámetros del controlador PID, como las ganancias proporcionales, integrales y derivativas. Este ajuste se realiza iterativamente, refinando los parámetros del controlador en cada paso para mejorar gradualmente el rendimiento del sistema. Finalmente, se valida y verifica el rendimiento del sistema mediante pruebas adicionales para asegurar que se cumplan los objetivos de rendimiento establecidos. Este proceso de sintonización permite lograr un ajuste óptimo del controlador PID y garantizar una respuesta estable y precisa del sistema en diversas condiciones de operación.

Ajustes de Ziegler-Nichols para el controlador ($\theta = \theta' + T/2$)

Controlador	K_c	τ_i	τ_d
P	$\frac{\tau}{K\theta}$	—	—
PI	$\frac{0.9\tau}{K\theta}$	3.33θ	—
PID	$\frac{1.2\tau}{K\theta}$	2θ	0.5θ

VI. RESULTADOS



VII. CONCLUSIONES

Durante la implementación del controlador de nivel de agua, se enfrentaron diversas dificultades que pusieron de manifiesto la importancia crítica de la selección adecuada de los componentes del sistema y el proceso de sintonización del controlador. Una de las principales limitaciones encontradas radicó en la elección del sensor de nivel de agua. Aunque se optó por un sensor disponible y económico, se evidenció su inadecuación para el propósito deseado. El sensor exhibió una alta susceptibilidad al ruido, lo que se tradujo en lecturas fluctuantes y poco confiables. La lentitud en la detección de cambios en el nivel de agua resultó ser un factor determinante que afectó significativamente la capacidad del controlador para mantener un control preciso y estable del sistema.

La problemática asociada al sensor se vio agravada durante el proceso de sintonización del controlador. La falta de respuesta rápida y consistente por parte del sensor dificultó considerablemente la obtención de una curva de reacción bien definida y estable. Si bien se intentaron ajustes manuales en los parámetros del controlador para compensar estas deficiencias, esta aproximación resultó ser más una medida paliativa que una solución definitiva. Los valores de sintonización modificados lograron una adaptación del controlador a las condiciones específicas del sistema, pero a expensas de la estabilidad y la robustez del control.

Como resultado de estas limitaciones, el sistema exhibió un comportamiento subóptimo durante las pruebas experimentales. Las fluctuaciones impredecibles en el nivel de agua y el comportamiento oscilatorio alrededor del punto de referencia fueron evidencia directa de las deficiencias en la implementación del controlador. Aunque se logró una funcionalidad básica del sistema, la falta de precisión y estabilidad comprometió su rendimiento y aplicabilidad en entornos prácticos.

En conclusión, los desafíos encontrados durante la implementación del controlador de nivel de agua destacan la importancia de una selección cuidadosa de los componentes del sistema y la adopción de métodos de sintonización rigurosos. Para futuras investigaciones o aplicaciones prácticas, se recomienda explorar alternativas de sensores más adecuadas y seguir enfoques de sintonización más sistemáticos y profesionales para garantizar un control efectivo y confiable del sistema.

VIII. REFERENCIAS

- [1] SIGMA ELECTRONICA LTDA., «Sigma Electrónica,» 09 Junio 2017. [En línea]. Available: www.sigmaelectronica.net. [Último acceso: 05 Abril 2024].

- [2] Arduino, «arduino.cc,» 25 Noviembre 2020. [En línea]. Available: www.arduino.cc. [Último acceso: 5 Abril 2024].