

# Design and Implementation of a PID Control System for a Temperature Model Plant

## Diseño e Implementación de un Control PID para una Planta Modelo de Temperatura

Pedro Mendoza Arnaez

Engineering Physics Program  
College of Sciences  
National University of Engineering

### ABSTRACT

This paper presents the design, implementation, control and testing of a temperature model plant used to verify the performance of diverse control strategies

The plant temperature is in the range 30°C - 70°C and the control strategies are programmed in an Arduino microcontroller. The plant has been modeled and the control strategies tested using Simulink before their implementation in real time control. For data acquisition and control a serial communication channel was established between the computer, microcontroller and plant.

El proyecto se divide: en la primera sección se da una breve introducción al trabajo, en la segunda sección mostramos el problema, en la tercera sección presentamos los objetivos, en la cuarta sección damos una descripción de la solución al problema presentado, en la quinta sección mostramos un ejemplo de control utilizando la planta modelo de temperatura y por último en la sexta sección mostramos nuestras conclusiones.

### INTRODUCCIÓN

Este trabajo presenta las distintas etapas del desarrollo de una planta modelo de temperatura y el desarrollo de un control PID utilizando el software MATLAB como ejemplo.

El diseño del instrumento está basado en una estructura con el microcontrolador ATMEGA328, que viene incorporado en el módulo ARDUINO UNO el cual fue elegido por tener una conexión vía serial en sus características.

Esta planta modelo de temperatura brinda muchas facilidades al momento de implementar distintos tipos de control ya que fue desarrollada en forma modular y estas partes pueden ser intercambiadas o modificadas.

### PRESENTACION DEL PROBLEMA

En la actualidad para poder implementar algún sistema de control en el laboratorio se requieren de plantas modelos que por su elevado costo no pueden ser adquiridas por las instituciones, además de ello estas plantas modelo requieren de software de alto costo que es necesario para su correcto funcionamiento.

Por ese motivo se diseñó una planta modelo de temperatura de bajo costo utilizando un módulo ARDUINO y etapas independientes que pueden ser modificadas para diferentes tipos de experiencias que se requieran en el laboratorio.

### OBJETIVOS

El objetivo del presente trabajo es poder realizar un sistema de control de bajo costo

utilizando microcontroladores para complementar el curso de Control.

Otro objetivo es el de desarrollar e implementar un sistema de control PID con microcontroladores utilizando esta planta de temperatura a modo de ejemplo.

## DESCRIPCIÓN DE LA SOLUCIÓN

### a. Diseño del módulo temperatura

La planta modelo de temperatura se diseñó se realizó en etapas como se muestra en la fig.1. y se procederá a describir cada una.

El sensor utilizado en este trabajo es un PT100 que está conectada a un circuito acondicionador que nos da de salida un rango de corriente de 4 a 20 mA. Esta señal pasa a la etapa de acondicionamiento donde la señal del sensor en miliamperios se convierte en una señal de 0 a 5 V con esta señal modificada se puede pasar a la etapa de conversión análoga digital.

La siguiente etapa es la del conversor analógico/digital, donde las señales amplificadas que entregan los acondicionadores de señal (0v a 5v), esta etapa es realizada directamente por el módulo ARDUINO UNO (fig. 2), en este caso utilizamos el pin A0 que es un conversor análogo/digital. El conversor del módulo ARDUINO UNO es de 10 bits.



Fig. 1 Etapas de diseño del módulo de temperatura

La señal digitalizada es procesada en el microcontrolador. En este proyecto se utilizó el microcontrolador ATMEGA328, característico del módulo ARDUINO UNO.

El micro controlador toma la señal del conversor A/D y lo relaciona directamente con un valor de temperatura previamente calculado. Este valor de temperatura se utilizara para controlar la resistencia eléctrica del tanque.

La resistencia eléctrica es utilizada para calentar el tanque a la temperatura deseada dada por el operario, en este caso esa temperatura debe variar en el rango de 30°C a 70°C.

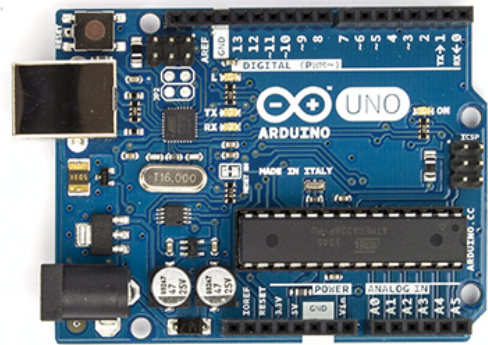


Fig. 2 Módulo ARDUINO UNO

Esta planta modelo de temperatura puede comunicarse con la PC utilizando los protocolos de comunicación serial ya que esta es una característica del módulo ARDUINO UNO.

### b. Construcción de la planta modelo de temperatura

La estructura de la planta modelos de temperatura está realizada en madera ya que con esto aislamos la temperatura de los tanques (fig. 3).

Ambos tanques están hechos de fierro galvanizado que es muy económico y fácil de maniobrar, se selló todas las juntas con silicona de calor para prevenir alguna filtración.

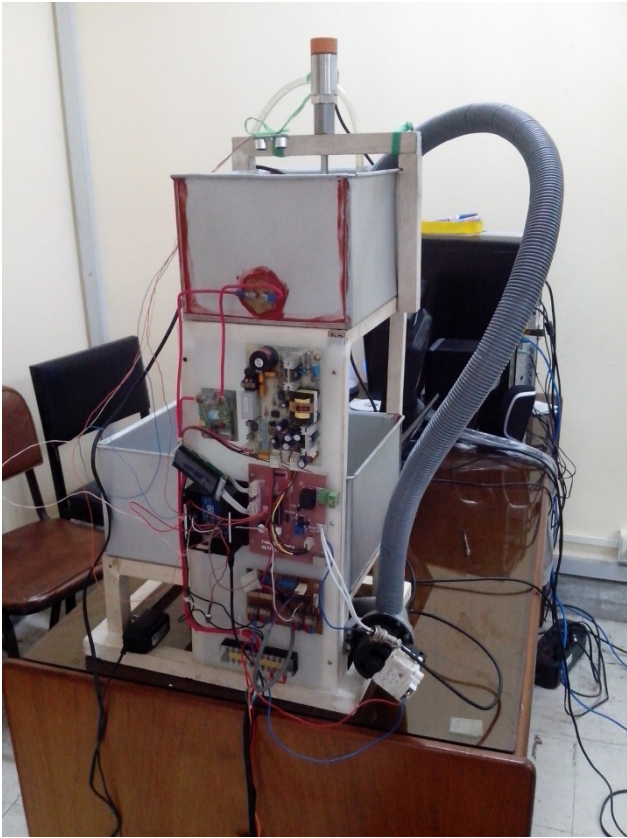


Fig. 3a Planta modelo de temperatura. Vista Frontal.

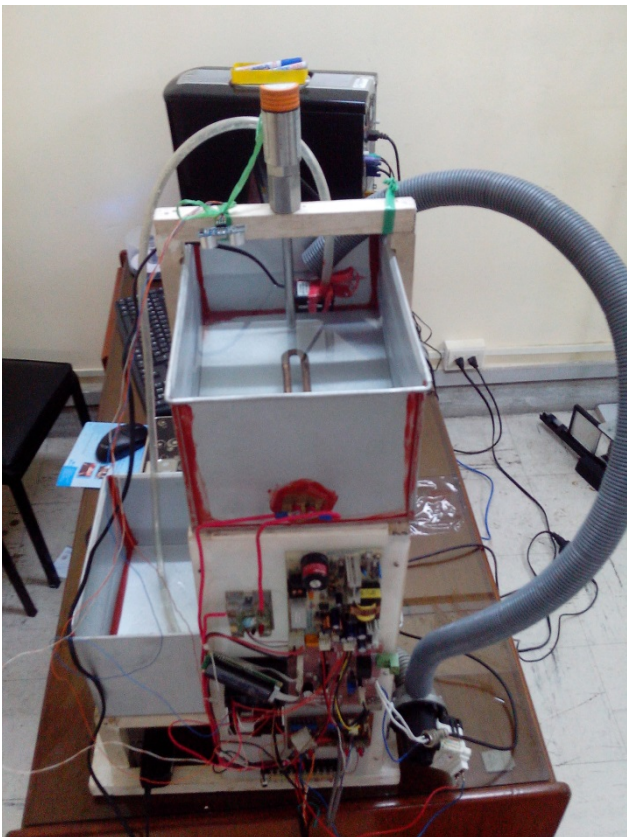


Fig. 3b Planta modelo de temperatura. Vista Superior.

## CONTROL PID UTILIZANDO LA PLANTA MODELO

Una vez construido la planta modelo de temperatura se realizó un experimento utilizando el método de control PID y la interface gráfica SIMULINK del software MATLAB.

A continuación la parte de código, tanto de ARDUINO como de Simulink/Matlab.

### a. ARDUINO: lectura y escritura de variables físicas

En primer lugar, el código del Arduino debe ser capaz de (1) leer la temperatura del sensor PT100, (2) enviarlo como dato a través del puerto serie, (3) recibir el dato de potencia necesaria a la resistencia y (4) escribir dicho dato en la salida analógica como señal PWM. Hay que tener en cuenta que el conversor devuelve un valor entre 0 y 1024, que mapeamos a 0 y 255 para poder enviarlo como un solo bit unsigned.

El código utilizado para esta experiencia es el siguiente:

```
int out = 0;
byte in = 0;
byte pinOut = 10;

void setup() {
  // inicializar puerto serie
  Serial.begin(9600);
  // preparar output
  pinMode(pinOut, OUTPUT);
}

void loop() {
  // leer del pin A0 como
  out = analogRead(A0);
  // escalar para obtener formato
  uint8
  out = map(out, 0, 1023, 0, 255);
  // enviar en base 10 en ASCII
  Serial.write(out);
  // leer del serie si hay datos
  if(Serial.available()){
    in = Serial.read();
    // escribir en el pin 10
    analogWrite(pinOut, in);
  }
  // esperar para estabilizar el
  conversor
  delay(20);
}
```

## b. Simulink: referencia y bucle de control

Por otra parte, nuestro programa de Simulink debe recibir este dato de a través del puerto serie, compararlo con una referencia que controlaremos nosotros con un slider en la propia interfaz gráfica de Simulink, y, mediante un controlador PID, determinar la señal de control (potencia a la resistencia) necesaria. Después debe enviarla a través del puerto serie en formato uint8, unsigned integer de 8 bits, que toma valores entre 0 y 255, ideales para la función analogWrite() de Arduino.

Dentro del bloque PID, se pueden editar los parámetros P, I y D, siendo los últimos que hemos empleado  $P = 0.26$ ,  $I = 0.9$ ,  $D = 0.04$  y una discretización de 10 ms. Otro dato a tener en cuenta es que la transmisión serie se hace en formato uint8, pero las operaciones se hacen en formato double, de ahí los conversores.

El esquema realizado en simulink se muestra en la fig. 4.

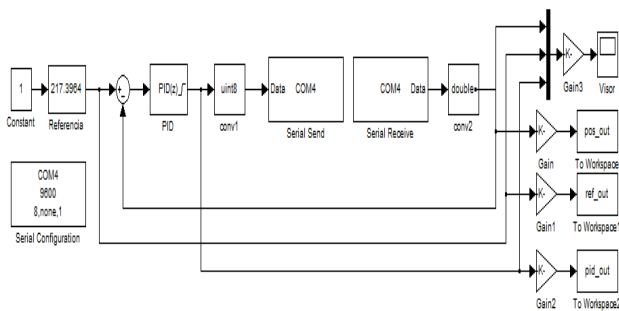


Fig. 4. Esquema realizado en simulink para el control PID.

## c. Resultados de la experiencia.

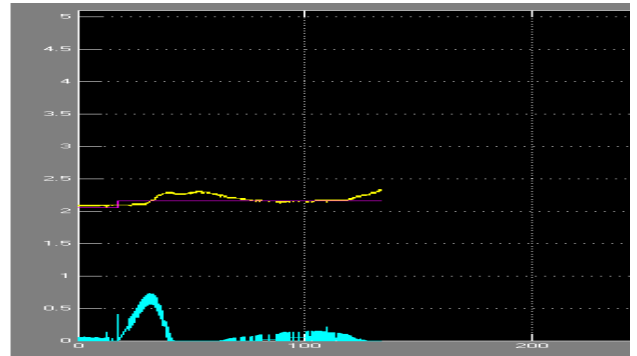
La figura 5 muestra los resultados obtenidos del control utilizando el entorno gráfico del software MATLAB.

## CONCLUSIONES

Se desarrolló un módulo de control de temperatura de bajo costo para complementar las clases de control que puede trabajar en un rango de temperatura entre 30°C y 70°C.

Se implementó un control PID utilizando la planta modelo de temperatura.

Se realizó una comunicación serial entre la PC y la planta modelo de temperatura para la adquisición de datos y realización del control.



## REFERENCIAS

- [1] ADZ NAGANO website. [Online] <http://www.adz.de/>
- [2] "MCP3202 data sheet" 2.7V Dual Channel 12-Bit A/D Converter with SPI Serial Interface. Microchip.
- [3] "PIC18F2559 data sheet" 28/40/44-Pin, High-Performance, Enhanced Flash, USB Microcontrollers with nanoWatt Technology. Microchip.
- [4] "Procedimiento de Calibración de Manómetros, Vacuómetros y Manovacúómetros de deformación elástica", 2da. Edición, Junio - 2012, SNM-INDECOPI.
- [5] "LM35 data sheet" LM35 Precision Centigrade Temperature Sensors. Texas Instruments.
- [6] "HD44780U data sheet" HD44780U (LCD-II) (Dot Matrix Liquid Crystal Display Controller/Driver), HITACHI.