

UNIVERSIDAD NACIONAL DE AVELLANEDA



MATERIA: Sistemas de Control Automático

Trabajo de Laboratorio N°2:

Sistema de control proporcional y PI en pava eléctrica

Integrantes:

Apellido y Nombres

E-mails

Uriel Alejo Sueldo

usueldo@gmail.com

Nicolas Lopez Vidueiros

nicolas.lopez.vidueiros@gmail.com

Juan Carlos Lorenzo

juanchilorenzo@gmail.com

Profesores: Ing. Guillermo Caporaletti, Ing. Juan Manuel Castellano

Índice

Índice	2
Introducción	3
Modelo sistema de control	3
Simulación	4
Controlador Proporcional	4
Controlador PI	5
Experiencia	7
Controlador Proporcional	7
Primera medición con 0,75L de agua	7
Controlador proporcional-integral	9
Primera medición con 0,75L de agua	9
Segunda medición con perturbación	12
Código	14
Conclusión	18

Introducción


Se controlará el sistema físico para eléctrica con un controlador proporcional. Primero se hallará la constante "Kp" que más se adecue a nuestro requerimiento de diseño. Luego se modelizará el sistema realizando gráfico xcos utilizando scilab y por último se hará la experiencia real para comprobar los resultados.

También, se controlará la para eléctrica con un controlador proporcional-integral. Al ser un controlador PI, se elegirá un valor de K y Ti para realizar la simulación. De esta manera podremos observar si hay sobrevalor o si el sistema es muy lento, es decir, identificar si el tiempo que tarda en establecerse es el deseado.

Luego se hará la experiencia de laboratorio con una para real utilizando controlador PI, donde con los valores de K y Ti elegidas en la simulación, analizaremos si el sistema real se comporta como lo preestablecido en la simulación.

Modelo sistema de control

Nuestro objetivo es establecer la temperatura del agua a 85°C. En el controlador proporcional nuestro requerimiento de diseño es utilizar un Kp que otorgue un error a estado estacionario de 2%. Utilizando el teorema de valor final hayamos Kp = 55 aproximadamente para que la temperatura del agua en estado estacionario sea de 83°C que se encuentra dentro del rango de 2%.

Para el controlador proporcional-integral, se eligió ganancia K = 40 para no tener sobrevalor en la señal de salida. Al elegir un K, notamos que a medida que aumentaba el mismo el sistema se volvía más rápido, pero generaba sobrevalor. Por otro lado, si elegimos K muy pequeño el sistema se vuelve muy lento. Por lo tanto, se definió K de 40 debido a que notamos que es el más estable. Luego se eligió un tiempo de integración de 50 minutos. Con estos parámetros logramos un sistema más estable y eficaz, tratando de que nuestro sistema no sea muy lento y no genere un sobrevalor muy alto. 

El modelo lineal de la Planta G(S) expresado en forma de polinomio de s es el siguiente:

$$G(s) = \frac{Tl(s)}{Pe(s)} = \frac{Rla}{\tau*s+1} = \frac{0.9 \frac{^{\circ}C}{W}}{4490s*s+1}$$

Simulación

Controlador Proporcional

Se realiza diagrama en xcos de controlador proporcional. Se indica ganancia K_p de 55 debido a que es el hallado para obtener un error a estado estacionario de 2%. Esto quiere decir que nuestra señal en estado estacionario será de 83°C aproximadamente. Para que nuestro modelo sea más real, se agregó un bloque de saturación que limita la potencia de la pava a 2200W. Se añadió bloque de retardo que forma parte de la planta. Por último se agregó bloque de ruido para simular el ruido del sensor que mide la temperatura del agua.

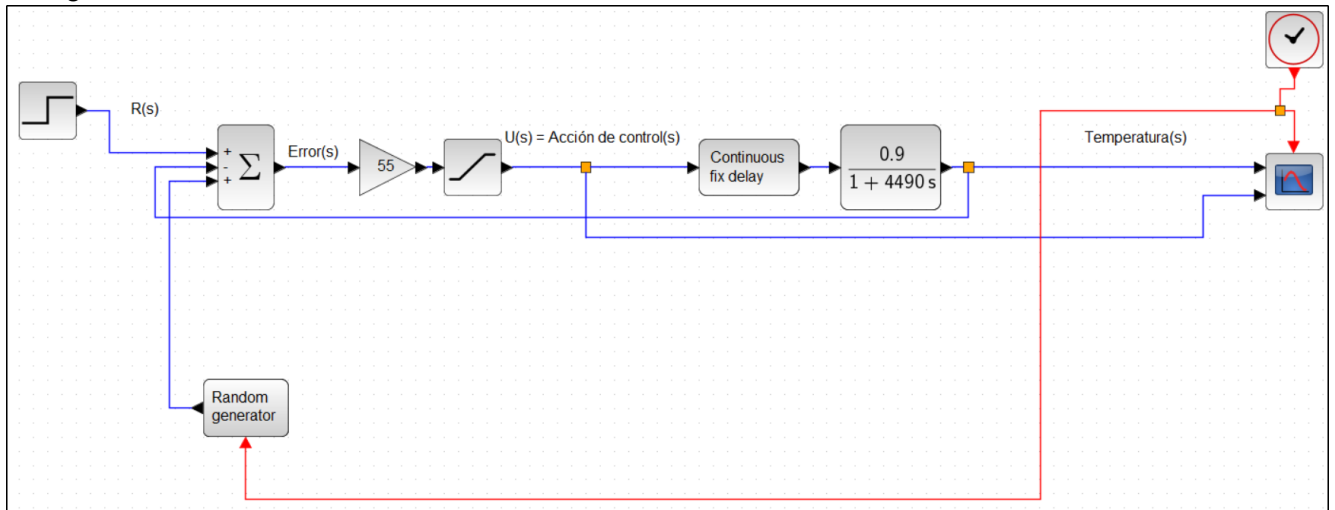


Figura 1

Simulación de la Señal de Salida

Nuestra señal de salida indica que el agua tarda en llegar a 83°C 527 segundos.

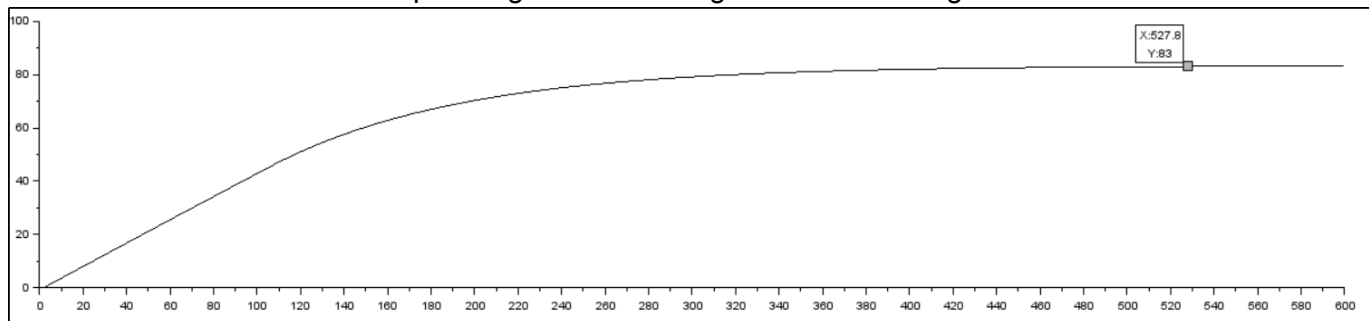


Figura 2

Simulación Señal de acción de control, potencia.

La potencia de la pava se ve saturada a 2200W. A medida que sube la temperatura del agua, la potencia va disminuyendo. También podemos notar el ruido del sensor.

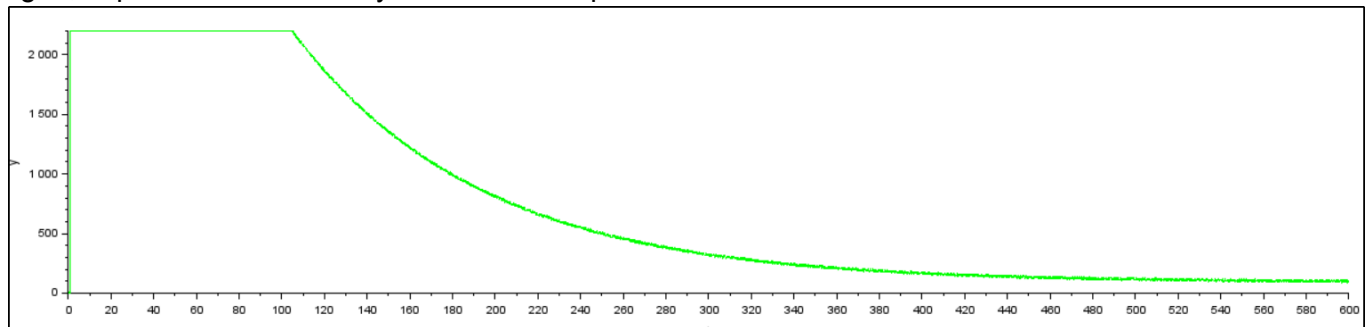


Figura 3

Controlador PI

Se realiza diagrama en xcos de controlador proporcional-integral. Se indica ganancia K de 40 y tiempo de integración T_i de 50 minutos. Se agregó bloque de saturación para simular un modelo más real donde la potencia es saturada a 2200W.

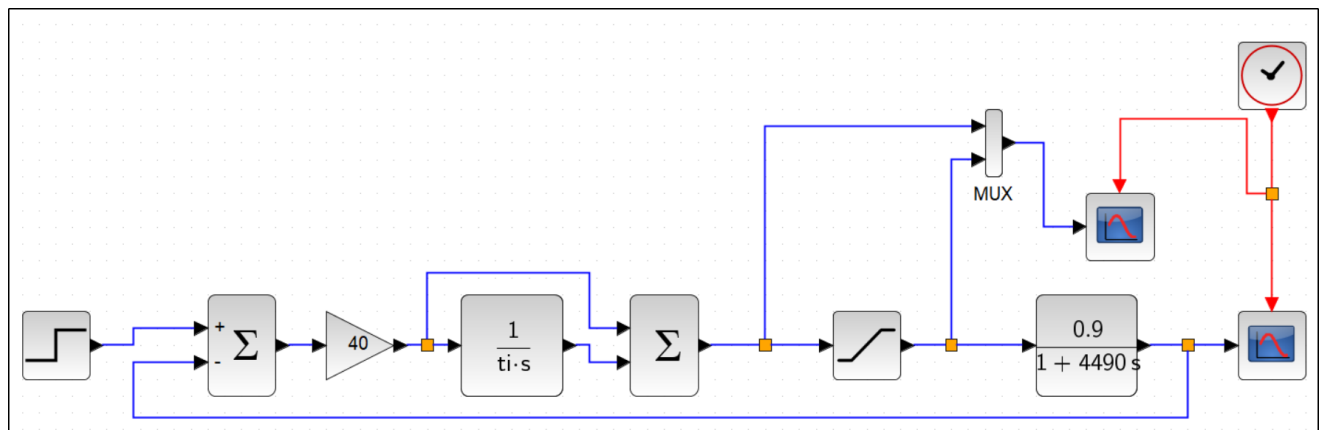


Figura 4

Luego se procede a realizar la simulación

Simulación de la Señal de Salida

En el caso del controlador PI, el tiempo de establecimiento de la señal de salida es de 423s. También podemos notar que, a diferencia del controlador proporcional, el controlador PI es capaz de tener error a estado estacionario nulo, donde se puede llegar a la temperatura del agua de 85°C a los 590s.

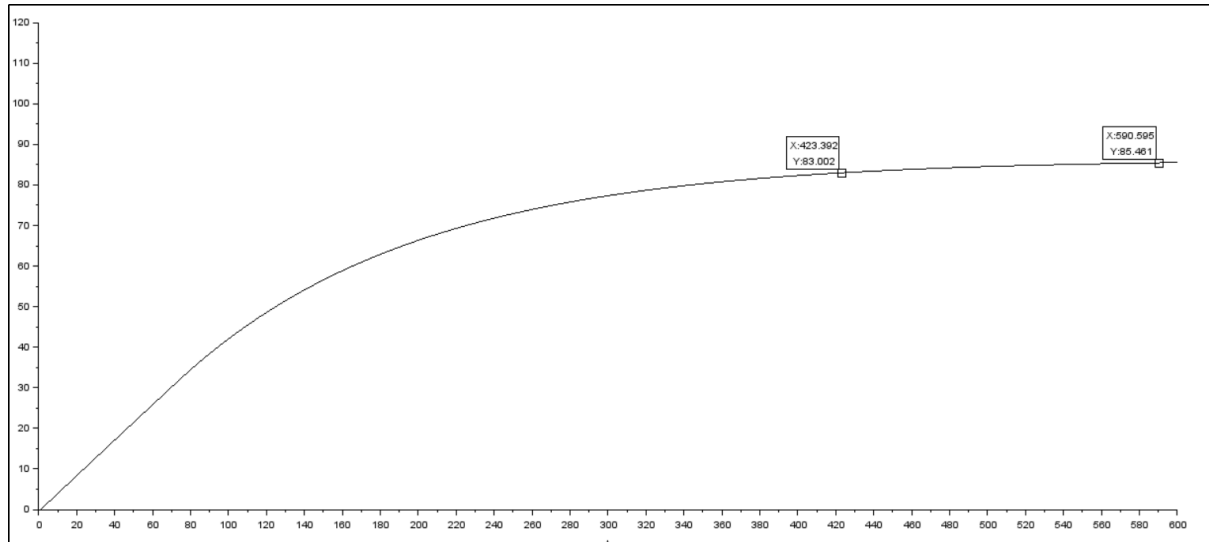


Figura 5

Simulación Señal de acción de control, potencia.

Al comienzo la potencia de la pava es saturada debido al bloque de saturación y luego se irá disminuyendo hasta que la temperatura del agua llegue a su temperatura deseada que es 85°C.

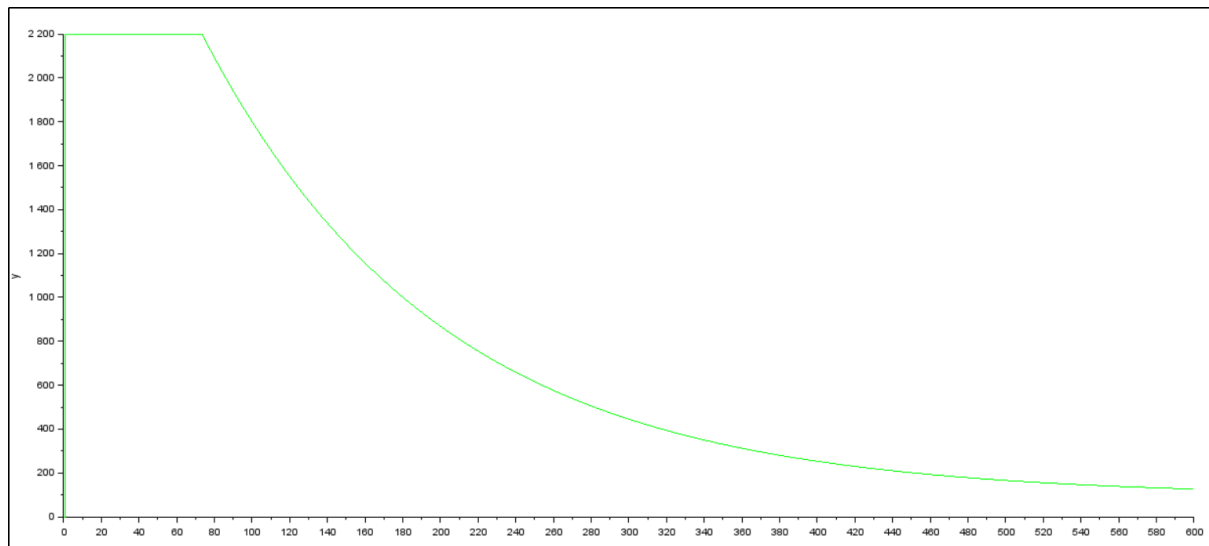


Figura 6

Experiencia

Controlador Proporcional

Primera medición con 0,75L de agua

Se mide la temperatura de 0,75 L de agua con sistema de control proporcional. El objetivo es llegar a una temperatura de 85°C con un error en estado estacionario de 2%. También se analizará cómo varía la potencia de la pava y la componente proporcional del sistema.

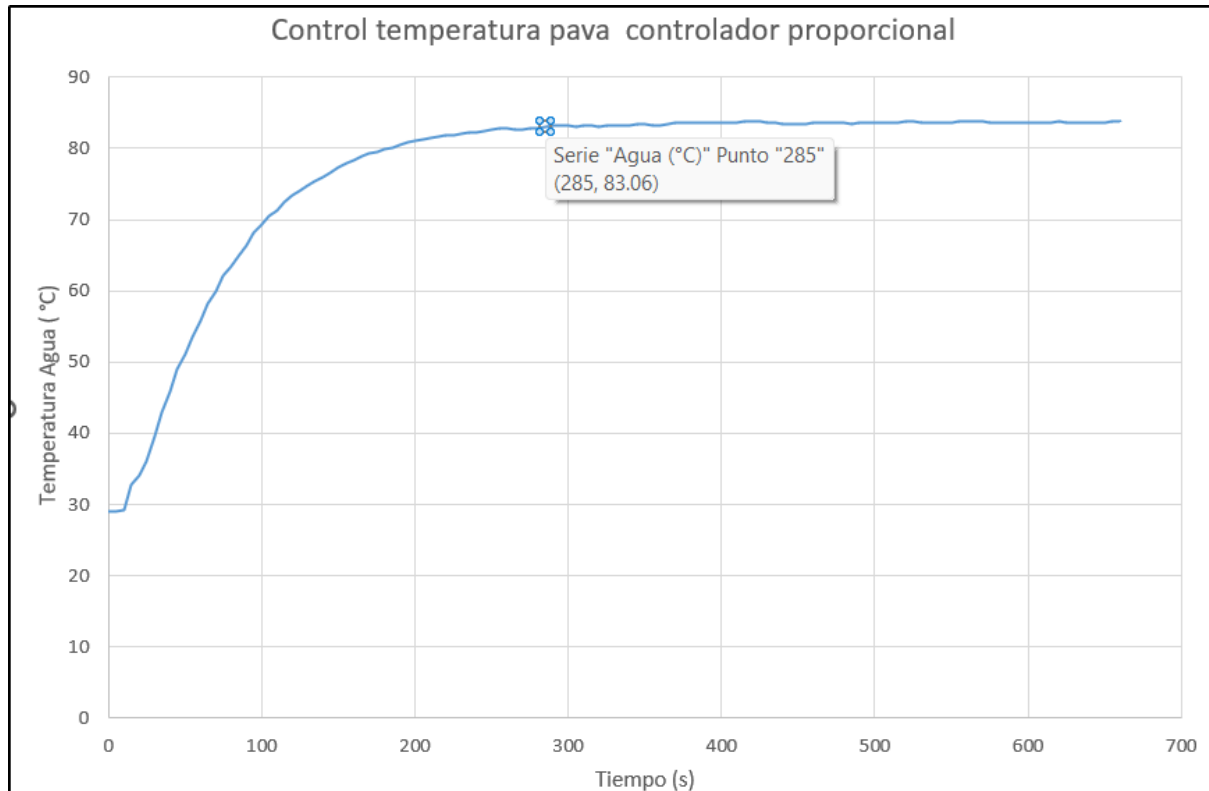


Figura 7

El agua tarda en llegar a los 83° en 285s, mucho menos tiempo que en la simulación debido a que se agregó un tiempo de retardo muy grande en la simulación.

Señal de acción de control, potencia

Al principio la potencia se ve saturada como en la simulación. La diferencia es que el voltaje máximo es de 1936W debido a que debemos tener en cuenta que el controlador del sensor de temperatura que usamos en arduino contiene delay bloqueante. Por lo tanto no podemos usar los 2200W.

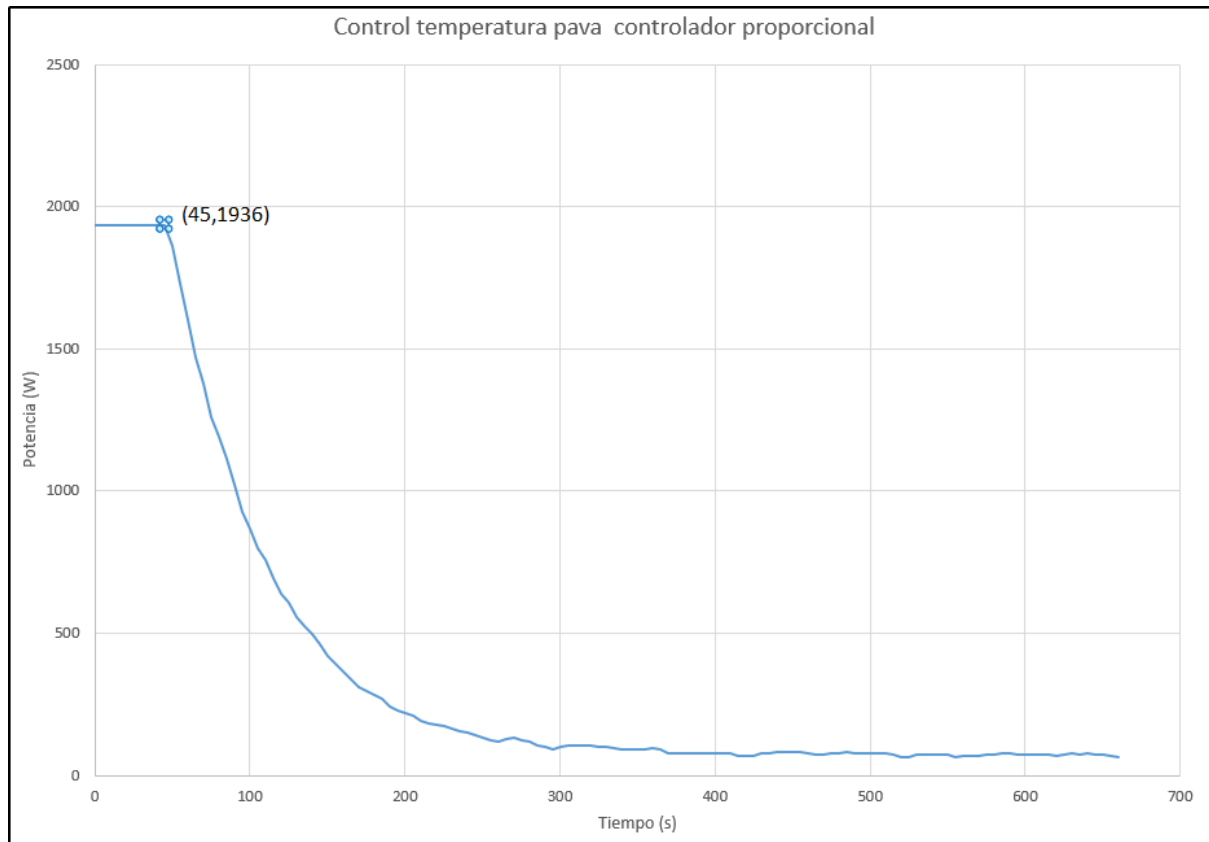


Figura 8

Señal del elemento actuador proporcional

El componente proporcional del sistema irá disminuyendo a medida que se llega a la temperatura objetivo del agua.

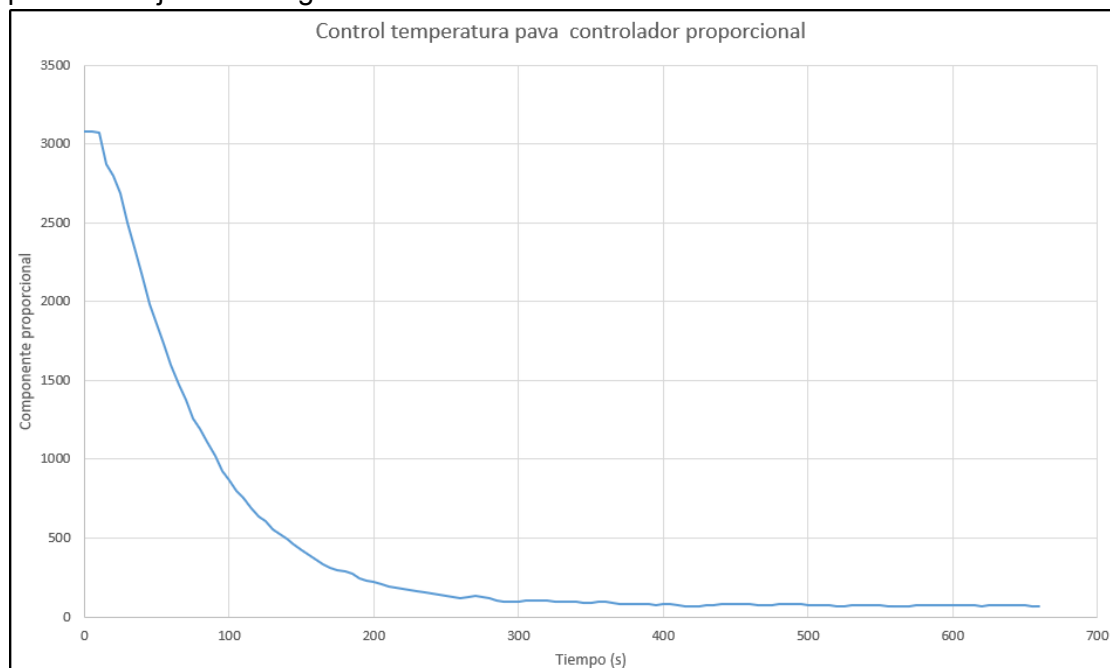


Figura 9

Controlador proporcional-integral

Primera medición con 0,75L de agua

Se mide la temperatura de 0,75 L de agua con sistema de control proporcional-integral. También se analizará cómo varía la potencia de la pava, la componente proporcional y la componente integral del sistema.

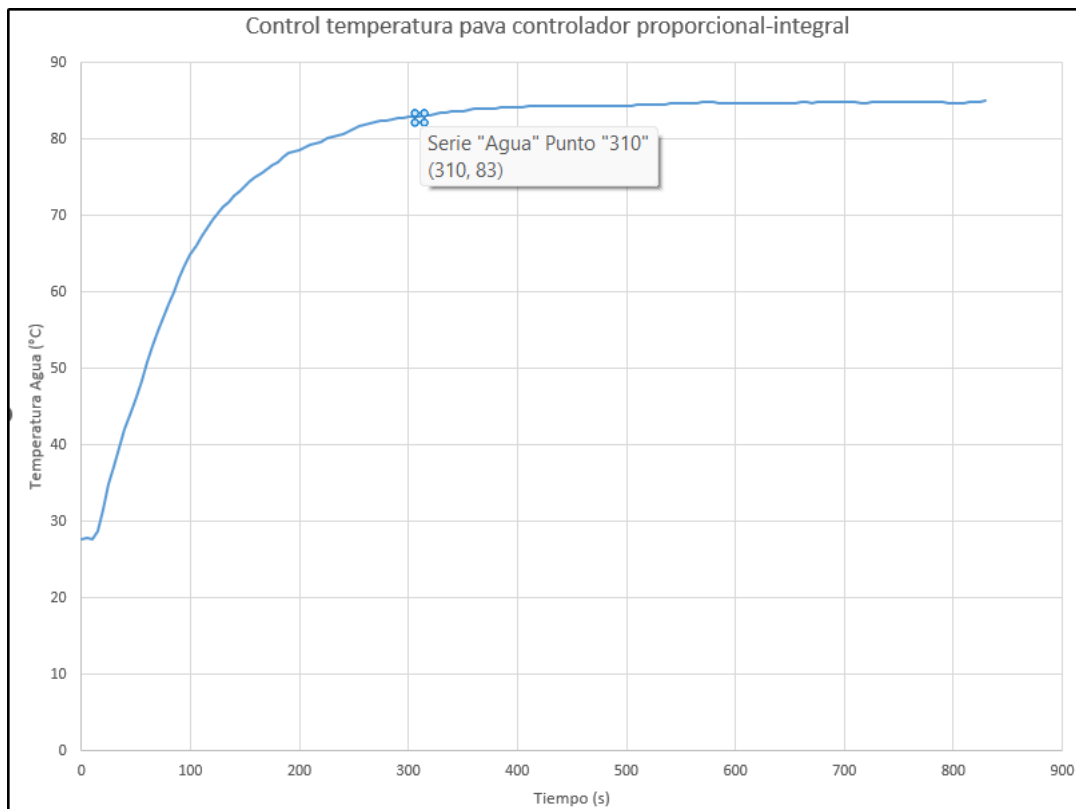


Figura 10

El agua tarda en llegar a los 83° en 310s, menos tiempo que en la simulación. Luego la temperatura del agua llega a los 85°C a los 830s.

Señal de acción de control, potencia

La señal de potencia fue saturada a 1936W para considerar el delay bloqueante del controlador del sensor de temperatura.

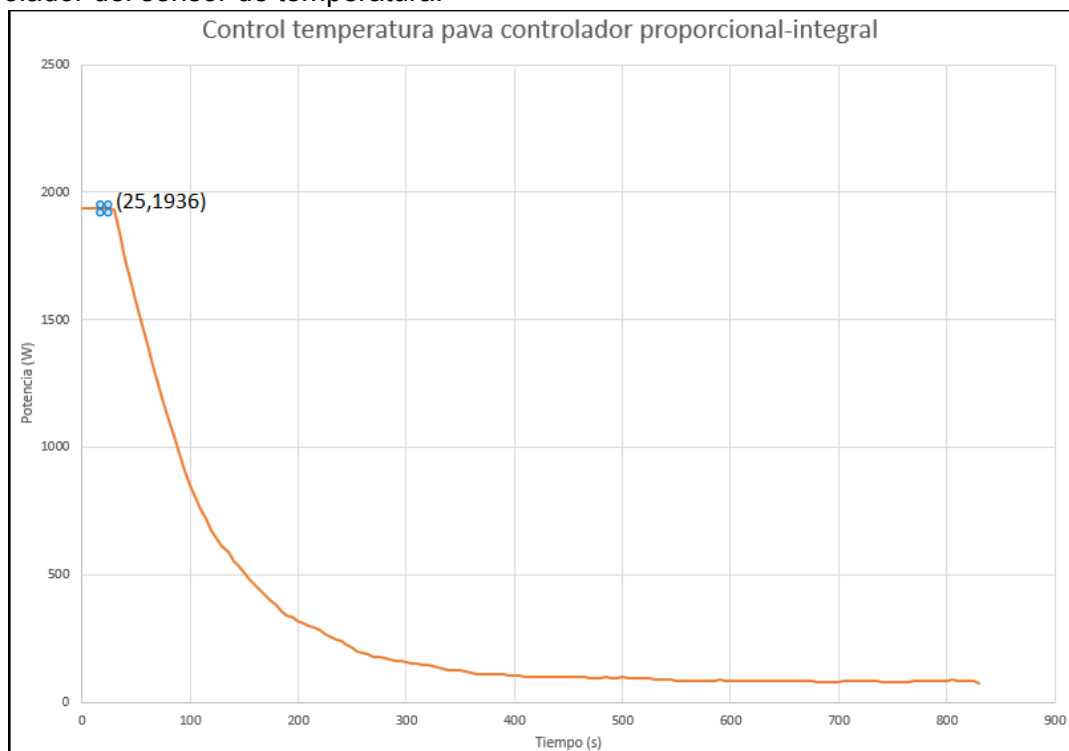


Figura 11

Señal del elemento actuador proporcional

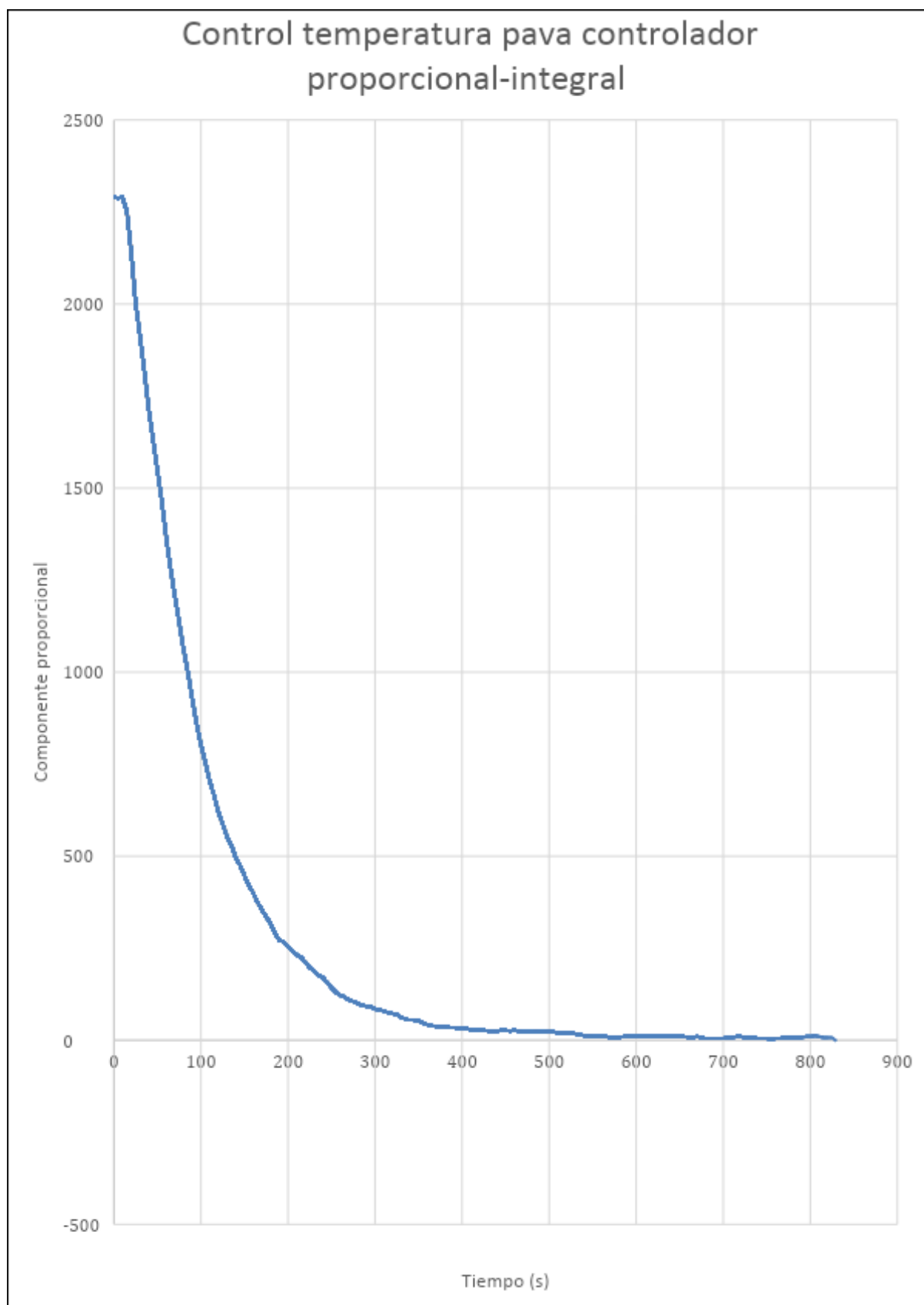


Figura 12

Señal del elemento actuador integral

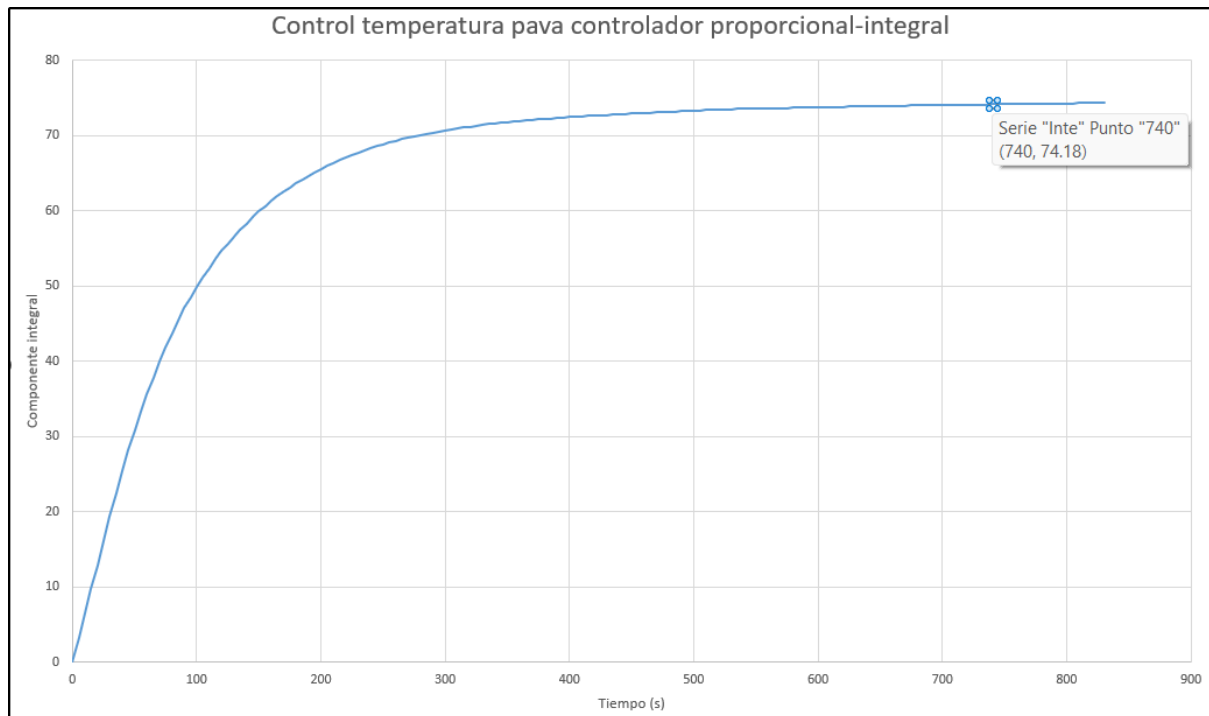


Figura 13

Segunda medición con perturbación

Se esperó a que la señal llegue a su estado estacionario. Una vez que el agua se mantuvo a los 85°C aproximadamente a los 830s, se cargó 0,3L de agua fría para generar una perturbación. Rápidamente podemos ver que la temperatura del agua disminuyó de 85°C a 68,94°C.

Luego de 405s la temperatura del agua se estableció aproximadamente a los 85°C en 1260s.

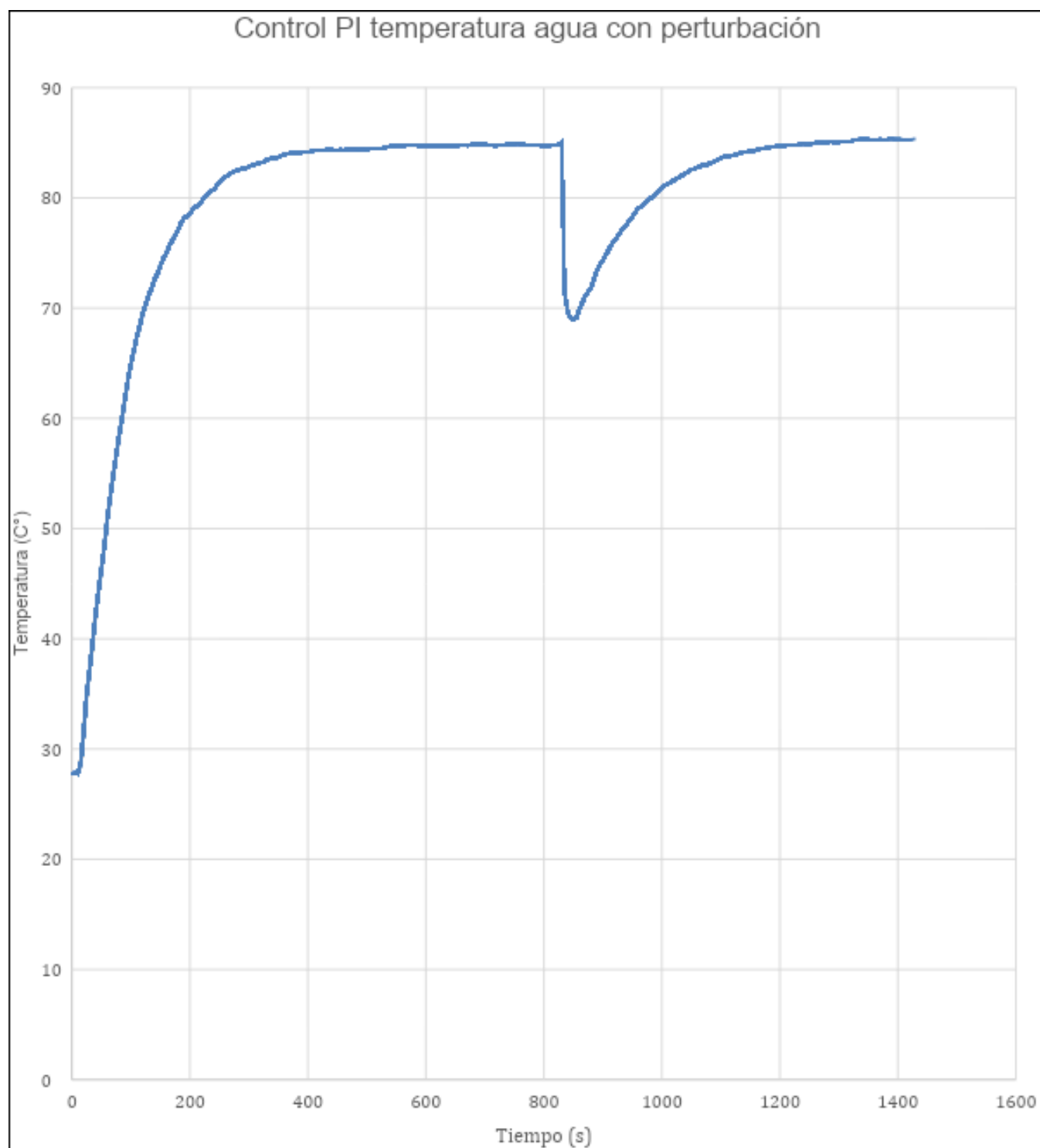


Figura 14

Código

```
/******  
* CONTROL PI DE UNA PAVA ELECTRICA  
* -----  
*  
* Sistemas de Control Automático UNDAV - septiembre de 2022  
*  
*****/  
  
#include <OneWire.h>  
#include <DallasTemperature.h>  
#include "pid2.h"  
  
const int sensorPin = 2;      //Pin del sensor de temperatura  
const int relePin = 7;        //Pin del Rele  
const int ledIntegradoPin = 13; // Pin del led que trae integrada la placa arduino  
const int botonParadaPin = 9;  //Pin para boton parada emergencia  
const float TempObj = 85;      // temperatura objetivo  
  
  
unsigned long TiempoAhora = 0; // tiempo en milisegundos  
unsigned long TiempoInicial = 0; // tiempo en que comienza la mprimer medición  
float UltTempMedida0;  
float UltTempMedida1;  
  
//float Kp = 55; //Proporcional  
//float Ti = 0; // Proporcional  
float Kp = 40; //proporcional-integral  
float Ti = 3000; // 3000 segundos son 50 minutos - proporcional-integral  
float Td = 0;  
float SalidaMin= 0;  
float SalidaMax= 2200;  
float SalidaSaturada=0;  
const bool limitarSalidaTrue = true; //  
const bool CompensaIntegralTrue = true; //denota si se esta compensando la integral  
controlPID PavaElectrica(Kp, Ti, Td);  
float accionControl = 0;  
float accionInt = 0;  
float accionProp = 0;  
float accionProplnt = 0;  
float compensacion = 0;  
  
const int periodoPWM = 5000; // 5 segundos  
unsigned long TH; //TH es le tiempo de encendido en el que se manda potencia a la pava  
unsigned long TL;  
  
OneWire ourWire(sensorPin); //Se establece el pin 2 como bus OneWire  
DallasTemperature sensors(&ourWire); //Se declara una variable u objeto para nuestro  
sensor  
  
void setup(){
```

```

        pinMode(botonParadaPin, INPUT);
        pinMode(ledIntegradoPin, OUTPUT);
        pinMode(relePin, OUTPUT);
        delay(1000);
        Serial.begin(9600);
        sensors.begin();           // Se inicia los sensores
        //Se calcula salida saturada teniendo en cuenta el delay del sensor de temperatura
        SalidaSaturada = ( (float) (periodoPWM - 600) / periodoPWM ) * SalidaMax;
        //Limito la salida del actuador y seteo sus valores min y maximos
        PavaElectrica.LimitarSalida(limitarSalidaTrue, SalidaMin, SalidaSaturada);
        PavaElectrica.CompensarIntegral(CompensarIntegralTrue);
        muestroEncabezadoProyecto();
        Tiempolnicial = millis();   // Este es el tiempo del primer muestreo
        TiempoAhora = Tiempolnicial;
    }

    void loop(void) {

        // *****
        // PASO 1 - MIDO
        // *****
        // Mido temperatura y actualizo variables
        sensors.requestTemperatures();
        UltTempMedida0 = sensors.getTempCByIndex(0);
        UltTempMedida1 = sensors.getTempCByIndex(1);

        // *****
        // PASO 2 - CONTROLAR PID
        // *****
        // Ajusta la tensión de la pava en base a la temperatura
        accionControl = PavaElectrica.Controlar(TempObj - UltTempMedida1);
        accionInt = PavaElectrica.ObtenerIntegral();
        accionProp = PavaElectrica.ObtenerProporcional();
        compensacion = PavaElectrica.ObtenerCompensacion();

        // *****
        // PASO 3 - MUESTRO VALORES
        // *****
        // Imprimo valores Sensados:

        imprimirValoresSensados(TiempoAhora-Tiempolnicial, UltTempMedida0, UltTempMedida1, acc
        ionControl, accionProp, accionInt, compensacion);

        //Logica PWM
        TH = (unsigned long) (accionControl / SalidaMax * periodoPWM);
        TL = millis() + TH;

        digitalWrite (relePin, HIGH);

        // *****

```

```

// PASO 4 - ESPERO ANTES DE LA PRÓXIMA MEDICIÓN
// *****
// delay de 5s con funcion millis()
do{
    // si se presionó el boton de emergencia --> paro el sistema
    if (digitalRead(botonParadaPin))
    {
        pararSistema();
    }
    if(millis() >= TL){
        digitalWrite (relePin, LOW);
    }

} while(millis() < TiempoAhora+periodoPWM);

TiempoAhora += periodoPWM;
digitalWrite (relePin, LOW);
}

void nuestroEncabezadoProyecto() {
    // Encabezado del proyecto
    Serial.println(F("-----"));
    Serial.println(F(" CONTROL DE TEMPERATURA DE PAVA ELECTRICA"));
    Serial.println(F("-----"));

    Serial.print("Constante Kp: ");
    Serial.print(Kp);
    Serial.println(" W/°C");
    Serial.print("Constante Ti: ");
    Serial.print(Ti);
    Serial.println(" ms");
    Serial.print("Periodo PWM: ");
    Serial.print(periodoPWM);
    Serial.println(" ms");
    Serial.println("");

    // Encabezado de la tabla
    Serial.print("Tiempo");
    Serial.print("\t");
    Serial.print("Amb");
    Serial.print("\t");
    Serial.print("Agua");
    Serial.print("\t");
    Serial.print("PI");
    Serial.print("\t");
    Serial.print("Prop");
    Serial.print("\t");
    Serial.print("Inte");
    Serial.print("\t");
    Serial.print("Comp");
    Serial.print("\t");
    Serial.println("");
}

```



```

void imprimirValoresSensados(unsigned long Tiempo,float Temp1, float Temp2, float
accionControl, float accionProp, float accionInt, float compensacion){
    // Imprimo valores Sensados:
    Serial.print(Tiempo);
    Serial.print("\t");
    Serial.print(Temp1);
    Serial.print("\t");
    Serial.print(Temp2);
    Serial.print("\t");
    Serial.print(accionControl);
    Serial.print("\t");
    Serial.print(accionProp);
    Serial.print("\t");
    Serial.print(accionInt);
    Serial.print("\t");
    Serial.print(compensacion);
    Serial.print("\t");
    Serial.println("");
}

void pararSistema()
{
    //Apagar pava
    PavaElectrica.Apagar();
    //Se apaga rele
    digitalWrite (relePin, LOW);

    //Se comunica parada de emergencia
    //Encendiendo y apagando el led del arduino
    //infinitamente
    Serial.print("PARADA DE EMERGENCIA");
    do
    {
        digitalWrite(ledIntegradoPin, HIGH);


        delay(1000);
        digitalWrite(ledIntegradoPin, LOW);
        delay(500);
    } while (true);
}

```

Conclusión

En la simulación podemos notar que utilizando el controlador proporcional el agua llega a la temperatura de los 83°C en 527s. En cambio, en el controlador proporcional-integral tarda 423s en llegar a dicha temperatura.

Por lo tanto, el control proporcional es más lento al llegar la temperatura del agua a 83°C en comparación al controlador proporcional-integral.

Además, el controlador proporcional-integral logra que la temperatura del agua se mantenga a los 85°C desde los 590s. 

En la experiencia, identificamos que el agua tarda 285s en llegar a los 83°C para el controlador proporcional. Tarda menos tiempo que en la simulación, debido a que en la simulación agregamos bloque de retardo que afecta en la velocidad del sistema.

En el controlador proporcional-integral se tarda 310s en llegar a los 83°C. Esto quiere decir que el controlador proporcional actuó más rápido que el proporcional-integral.

Sin embargo, con el controlador proporcional-integral logramos llegar a la temperatura objetivo de 85°C en 830s.

Si deseamos que el agua esté a la temperatura exacta que deseamos. Entonces es preferible usar el controlador proporcional-integral, pero debemos tener en cuenta de la experiencia que debemos esperar más tiempo para lograr dicha temperatura.

Si preferimos rapidez en vez de precisión, entonces elegimos el controlador proporcional ya que, si bien no llega a los 85°C, podemos utilizar el agua en una temperatura menor a la que deseamos, en este caso 83°C.