

Práctica 4

Sistemas de control dinámico I

Esta práctica es la primera de dos prácticas dedicadas a la implementación de un sistema de control de la temperatura de un horno. Su objetivo es implementar dos programas para medir las características dinámicas de un horno que se modela con la ecuación:

$$C_t \frac{dT(t)}{dt} = P(t - L) - C_p [T(t) - T_e] \quad (4.1)$$

donde,

$T(t)$ (C) es la temperatura, en función del tiempo,

$P(t)$ (W) es la potencia aplicada, en función del tiempo,

C_t (J/K) es la capacidad térmica,

L (s) es el retardo en la respuesta,

C_p (W/K) es el coeficiente de pérdidas y

T_e (C) es la temperatura del exterior del horno o temperatura ambiente.

El comportamiento del horno se simula con un paquete de nombre **Horno** que exporta las operaciones **Leer** y **Escribir** que aparecen en la especificación siguiente:

Fichero 4.1: Especificación del paquete Horno (Horno.ads)

```
1 package Horno is
2   type Temperaturas is new Float;
3   type Potencias is new Float;
4
5   procedure Escribir (la_Potencia: Potencias);
6   procedure Leer (la_Temperatura: out Temperaturas);
7 end Horno;
```

El cuerpo de este paquete encierra los detalles del simulador y el objeto de esta práctica es calcular los parámetros con los que ha sido diseñado el simulador. En concreto, hay que calcular los parámetros: T_e , L , C_p y C_t .

4.1 Tareas a realizar

1. Escribir un paquete de nombre **Sensor** que se ajuste a la especificación siguiente:

Fichero 4.2: Especificación del paquete Sensor (Sensor.ads)

```
1 package Sensor is
2   type Temperaturas is new Float range -25.0..500.0;
3
```

```

4  procedure Leer (la_Temperatura: out Temperaturas);
5  end Sensor;

```

En el cuerpo de este paquete se implementará la operación **Leer**, encargada de leer la temperatura actual del horno. Esta implementación se hará usando el paquete **Horno**.

2. Escribir un paquete de nombre **Calefactor** que se ajuste a la especificación siguiente:

Fichero 4.3: Especificación del paquete **Calefactor** (**Calefact.ads**)

```

1  package Calefactor is
2    type Potencias is new Float range 0.0 .. 10_000.0;
3
4    procedure Escribir (la_Potencia: Potencias);
5  end Calefactor;

```

En el cuerpo de este paquete se implementará la operación **Escribir**, encargada de escribir la potencia que se le suministra al horno. Esta implementación se hará usando el paquete **Horno**.

3. Escribir un programa de nombre **Medir1** que mida el valor de los parámetros: T_e , L y C_p . La medida de T_e es directa y se corresponde con la temperatura del horno en reposo. La medida de L se obtiene viendo el tiempo que tarda el horno en responder cuando le aplicamos una potencia cualquiera, por ejemplo 1000 W. La medida de C_p se puede hacer cuando el horno alcanza el régimen permanente después de aplicarle una potencia cualquiera. El régimen permanente se corresponde con $t \rightarrow \infty$ y se verifica que $dT/dt = 0$. En estas condiciones $0 = P - C_p(T - T_e)$.
4. Dibujar el diagrama de componentes del programa **Medir1**.
5. Escribir un programa de nombre **Medir2** que utilice los valores obtenidos con el programa anterior y que mida el parámetro C_t . La medida de C_t se debe hacer durante el régimen transitorio. Para ello hay que excitar el horno con una potencia y ver cómo evoluciona la temperatura. Si aproximamos la derivada por un cociente de incrementos tenemos:

$$C_t \frac{\Delta T}{\Delta t} = P - C_p(T - T_e) \quad (4.2)$$

de donde se obtiene

$$C_t = \frac{P - C_p(T - T_e)}{\Delta T} \Delta t \quad (4.3)$$

Ambos programas utilizan el simulador del horno. El simulador se compone de varias tareas concurrentes y para que los programas **Medir1** y **Medir2** terminen correctamente y devuelvan el control es necesario apagar el horno. Esto se consigue escribiendo la potencia 0.0 en el horno al final de los programas **Medir1** y **Medir2**.

Una vez realizadas todas la medidas se completará la tabla siguiente:

Parámetro	Valor y unidades
T_e	
L	
C_p	
C_t	