

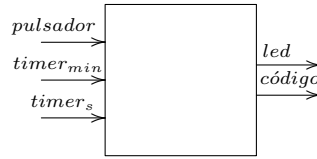
# ISEL 2020 : Ejercicio 12

Miguel Taibo

Abril 2020

## 1 Especificaciones

Tenemos un sistema con un pulsador, una luz y un código de 3 dígitos.



input		variables		output	
pulsador	p={0,1}	contador	q={0,1...10}	led	l={0,1}
timer <sub>min</sub>	T={0,1}	indice	i={0,1...5}	alarma	a={0,1}
timer <sub>s</sub>	t={0,1}	códigos	c <sub>1,2,3,4,5,6</sub> = {0,1...10}		

### 1.1 Control de luces

1. El pulsador enciende la luz.
2. Si no se pulsa el pulsador la luz se queda apagada (no se enciende).
3. Dura encendida 1 minuto desde que la luz se ha dejado de pulsar.

1.  $G(p \Rightarrow F(l))$
2.  $G(\neg l \wedge \neg p \Rightarrow F(\neg l))$
3.  $G(l \Rightarrow l \text{ U } T)$

### 1.2 Alarma de código

1. Si la alarma esta desactivada y se pulsa el pulsador se empieza a introducir un código.
2. Si la alarma esta activada y se detecta con el detector infrarrojo salta la alarma.
3. El estado no cambia mientras no salte el timer.

1.  $G(state == idle \wedge p \Rightarrow F(state == introducir))$
2.  $G(state == activada \wedge PIR \Rightarrow F(a))$
3. Para  $P = \{state == idle, state == introducir, state == activada\}$  se cumple:  
 $G(P \wedge \neg t \wedge F(t) \Rightarrow (PUt))$

## 2 Modelado

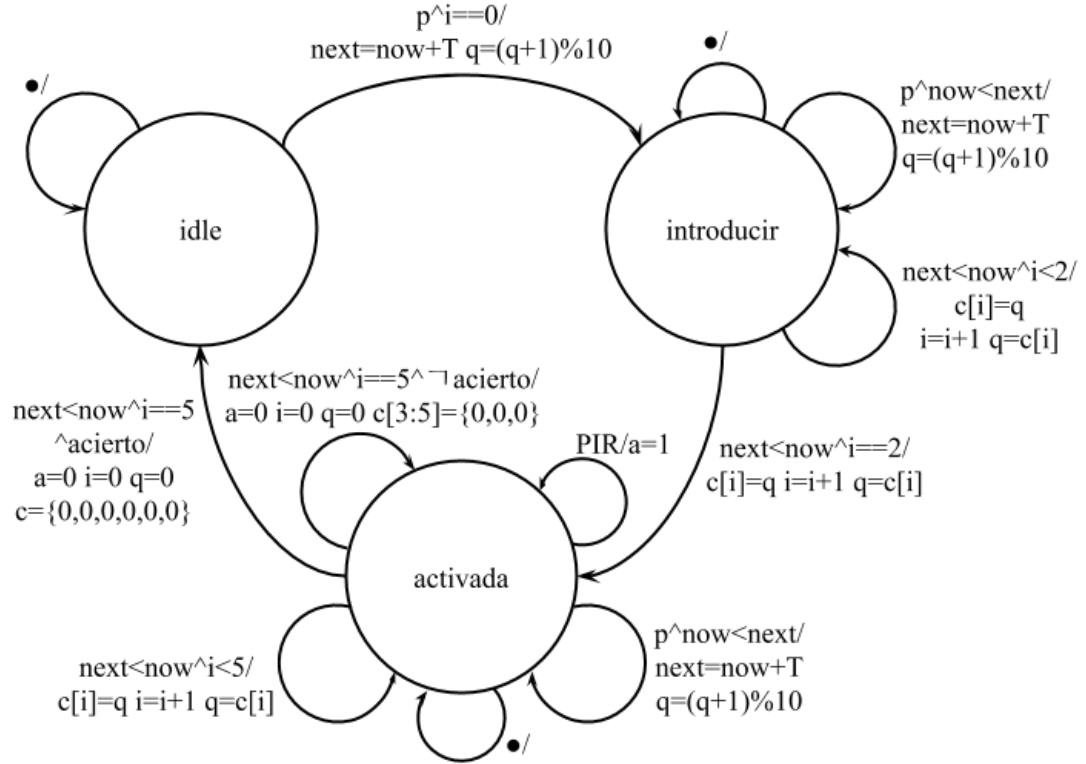


Figure 1: Máquina de estados finita del control de la alarma de código.

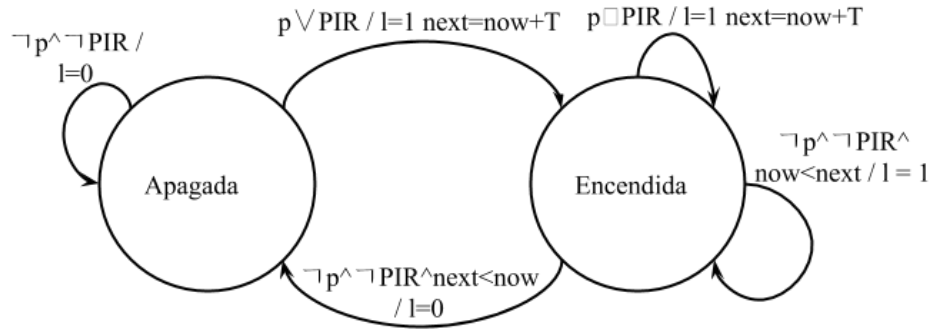


Figure 2: Máquina de estados finita del control de luz.

### 3 Análisis

Para calcular los tiempo de ejecución máximos medimos el tiempo justo antes y justo después de la tarea y comparamos la diferencia con la diferencia máxima obtenida anteriormente. Guardando el tiempo mayor, al salir imprimimos este valor por pantalla. Para no obtener datos irreales repetimos varios ciclos varias veces y de formas diferentes, incluyendo formas muy aleatorias. En este caso es algo más simple de lo normal al tener un sistema con una sola entrada.

Elegimos un valor del tiempo de ejecución ligeramente superior al obtenido sabiendo que a lo largo de las distintas pruebas el tiempo a penas varió, lo que nos hace pensar que esos tiempo son suficientemente pesimistas.

Elegimos el tiempo de activación para asegurar la velocidad a la que se deben ejecutar las tareas para que el sistema funcione correctamente. Tomamos el plazo de respuesta igual al tiempo de activación.

- La alarma de código debe funcionar más rápido de lo que se puede pulsar el botón, por lo que elegimos una frecuencia de 100Hz. Pulsa mas de 100 veces un botón en un segundo es imposible.
- El procesado de tecla tiene que tener la misma velocidad que la alarma, no tiene sentido que sea menor ni mayor, ya que la más restrictiva será la velocidad real.
- El refresco de pantalla lo fijamos a 25Hz, que es aproximadamente lo que tenemos en las películas, y una frecuencia suficiente para que se vea continuo por el ojo humano.
- El control de la luz no tiene una gran importancia: debe esta encendida 1 minuto entero por lo que el tiempo de decenas de milisegundos es más que suficientemente pequeño.

Tarea	Nombre	$C_{exp}(\mu s)$	$C(\mu s)$	$T(ms)$
1	Control de luces	4.500	6	40
2	Alarma de código	2.1193	3	10
3	Procesado de tecla	4.6784	6	10
4	Refresco de pantalla	6.2817	8	40

Table 1: Tareas del sistema

De esta forma tenemos un hiperperiodo de:

$$T_M = mcm\{40, 10, 10, 40\} = 40 \text{ (ms)}$$

Y un ciclo secundario de:

$$T_S = mcd\{40, 10, 10, 40\} = 10 \text{ (ms)}$$

Luego necesitamos 4 frames y un periodo de 10 ms. No se considera necesario realizar la planificación completa, ya que, seleccionando los plazos iguales a los periodos ( $D = T$ ):

$$\sum_{i=1}^4 C_i = 23\mu s \ll T_j; \quad j = 1, 2, 3, 4 \quad (1)$$