

## PRÁCTICA-3: Modelado VHDL de una FSM CONTROL DE UN ASCENSOR

### 1. OBJETIVOS

- Realizar modelos de máquinas de estados finitos en estilo comportamiento
- Realizar modelos en estilo estructural de sistemas que incluyan subsistemas
- Realizar modelos de test
- Obtener e interpretar los resultados de simulación

### 2. ASPECTOS PREVIOS

PARA LA REALIZACIÓN DE ESTA PRÁCTICA ES NECESARIO:

- Conocer los estilos de descripción de FSM vistos en clase
- Saber manejar la herramienta ModelSim
- Haber realizado el diseño del problema
- Haber descargado de Moodle el fichero de test: [FSM\\_ascensor\\_tb.vhd](#)
- Haber realizado la práctica-1: Modelado VHDL de un codificador binario para la selección de un piso en un ascensor

### 3. PLAZOS Y FORMA DE ENTREGA - Las fechas de entrega serán las que indique el profesor

**Modelos VHDL:** Todos los ficheros deberán incluir una la cabecera

La entrega de los ficheros de los modelos VHDL deberá hacerse comprimida con ZIP o RAR. La entrega se llamará:

***FSM\_apellidos\_nombre\_vhd.zip***

y tendrá el contenido que se indica a continuación, entre paréntesis se indica el nombre de la entidad y arquitectura que debe haber en el fichero vhd:

- FSM_ascensor.vhd	(FSM_ascensor, FSM_arquitectura)
- codifica_boton.vhd	(codifica_boton, arquitectura_cod_boton)
- ascensor_completo.vhd	(ascensor_completo, arquitectura_ascensor_completo)
- ascensor_completo_tb.vhd	(ascensor_completo_test, ascensor_completo_arquitectura_test)

**Tareas:** La entrega de las cuestiones y capturas de pantalla, que se indican a lo largo del desarrollo de la práctica, ha de realizarse en un único fichero con formato pdf, y debe incluir el nombre y apellidos del alumno/s. Las tareas se deben contestar en el orden en el que se describen en este guión. Se llamará:

***FSM\_apellidos\_nombre\_Tareas.pdf***

y contendrá las tareas:

- Tarea-1:** Transformar el diagrama de estados de la Figura 2 a su diagrama ASM.  
**Tarea-2:** Realizar la tabla de transición entre estados.  
**Tarea-3:** Obtener las ecuaciones lógicas de las señales D1, D0, sube y baja.

- Tarea-4:** Implementar las ecuaciones lógicas de dichas señales con los componentes que se indica en el guión.
- Tarea-5:** Capturar los resultados de simulación de la FSM (comportamiento) e interpretar si con correctos.
- Tarea-6:** Captura de pantalla del contenido de la librería WORK, tras realizar la simulación en comportamiento de la FSM.
- Tarea-7:** Captura de los resultados de simulación del codificador del botón e interpretar si con correctos. **Tarea ya realizada en la Practica-1**
- Tarea-8:** Capturar los resultados de simulación del ascensor completo e interpretar si con correctos.
- Tarea-9:** Captura de pantalla del contenido de la librería WORK, tras realizar la simulación del ascensor completo.

Asimismo **se deberá indicar en este documento**, el número de horas dedicadas a terminar el trabajo fuera de las horas de clase, los problemas y dificultades que le han ido surgiendo durante su desarrollo, y cualquier comentario que desee hacer.

## 4. DESARROLLO

### FUNDAMENTOS TEORICOS

El problema a resolver consiste en realizar una FSM que controle el funcionamiento simplificado de un ascensor, generando las señales que activen el motor para que suba o baje, que proporcione información visual en el panel situado dentro del ascensor del movimiento que está llevando a cabo, y del piso donde se encuentra. El problema no considera otras acciones que están presentes en un sistema real, tales como la apertura o cierre de puertas, la detección de obstáculos entre las puertas, la secuencia del tránsito entre pisos, la solicitud de ayuda en caso de emergencia, etc.

El ascensor da servicio a una casa de 3 pisos, los componentes del panel se muestran en la Figura 1. Consta de tres botones con los que se selecciona el piso donde se quiere ir, una señal que se enciende cuando el ascensor está subiendo, otra cuando está bajando y un indicador numérico del piso donde está el ascensor. La estructura a realizar está formada por un subsistema que codifica el piso donde va, y otro que, es una FSM que controla el funcionamiento del ascensor. La funcionalidad de ambos subsistemas se especifica a continuación. El subsistema `codifica_boton`, fue descrito y realizado en VHDL en la práctica *Modelado VHDL de un codificador binario para la selección de un piso en un ascensor*, y se reutilizará en esta práctica. No obstante por facilidad, recordaremos sus especificaciones y funcionalidad a continuación.

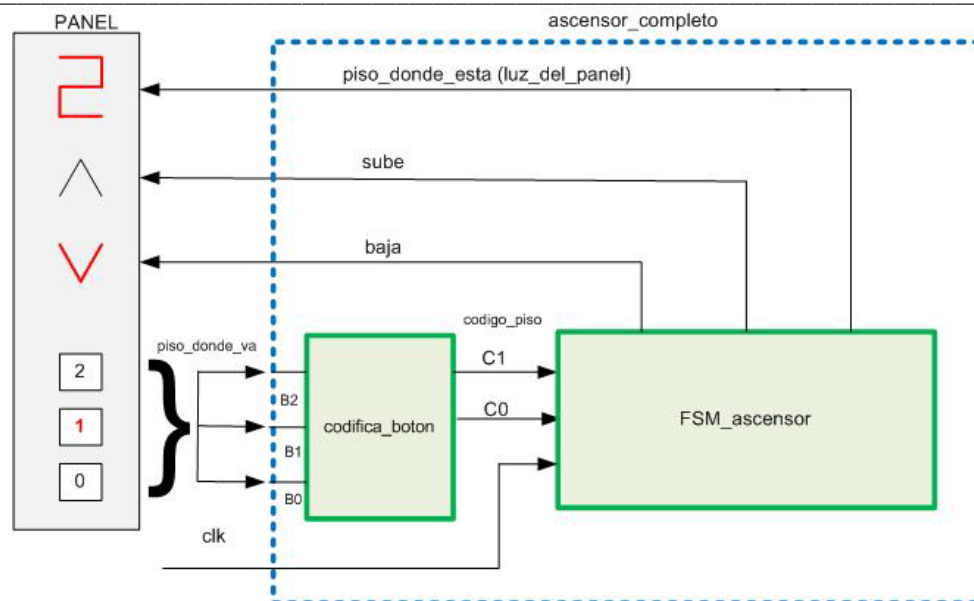


Figura 1. Esquema de bloques del control del ascensor

El codificador del botón, transformará un código de 3 bits, que se genera cuando se aprieta cualquier botón, a un código de 2 bits (C1 y C0), según se indica en la Tabla 1, el cual será la entrada a la FSM.

Botón activado	piso_donde_va código generado por el botón B2 B1 B0	código_piso (C1 C0)
0	0 0 1	0 0
1	0 1 0	0 1
2	1 0 0	1 0
Ninguno o más de uno a la vez	0 0 0/X 1 1	1 1

Tabla 1. Códigos binarios de los botones (señales de entrada a la FSM)

La FSM tendrá 3 estados: Piso\_0, Piso\_1 y Piso\_2, que serán codificados según la siguiente tabla:

Estado	Q1	Q0
Piso_0	0	0
Piso_1	0	1
Piso_2	1	0

Tabla 2. Codificación de los estados

El funcionamiento de la FSM será como sigue:

- El indicador numérico del panel de la Figura 1, representará el piso donde está el ascensor (irá asociado al estado).
- Permanecerá en el mismo piso cuando no se active ningún botón, se active más de uno a la vez, o se active el mismo piso donde está parado. Las señales de control del motor sube y baja permanecerán inactivas (valor '0').

- Transitará al estado del piso que se seleccione con el botón, cuando no se cumplan ninguna de las condiciones del párrafo anterior. Las salidas irán asociadas a la transición. Cuando al piso al que se vaya esté por encima del piso de donde se encuentra parado, la señal de subida estará activa (sube = '1') y de la bajada inactiva (baja = '0'). Cuando al piso al que se vaya esté por debajo del que está parado, las salidas se activarán de forma contraria (sube = '0', baja = '1').

- El estado de inicio será el piso\_0.

El diagrama de estados de la FSM que contempla las especificaciones anteriores, se indica a continuación:

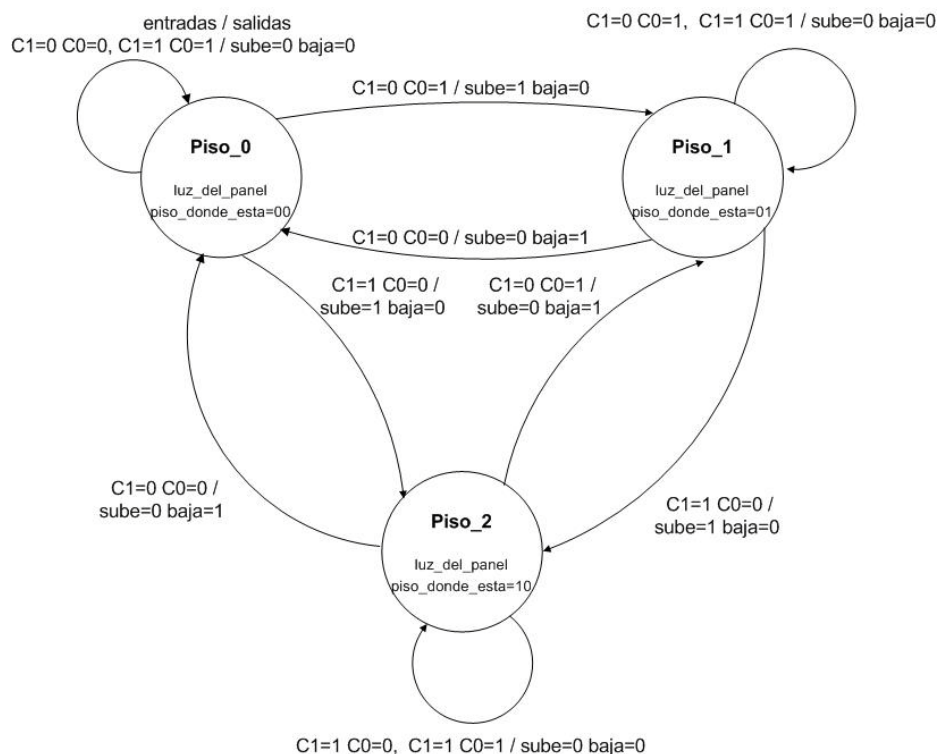


Figura 2. Diagrama de estados de la FSM

## APARTADOS A REALIZAR:

### PARTE I. DISEÑO DE LA MÁQUINA

**Tarea-1:** Realizar el diagrama ASM del diagrama de estados de la Figura 2.

**Tarea-2:** Realizar la tabla de transición entre estados. Según el modelo adjunto:

Estado actual (t)		Entradas(t)		Estado futuro (t+1)		Salidas			
Nombre del estado	codificación	codigo_piso						piso_donde_esta	luz panel
	Q <sub>1</sub> (t) Q <sub>0</sub> (t)	C <sub>1</sub>	C <sub>0</sub>	Q <sub>1</sub> (t+1)	Q <sub>0</sub> (t+1)	sube	baja	Q <sub>1</sub> (t) Q <sub>0</sub> (t)	
<b>Piso_0</b>	0 0	0	0	.....		...	...	0 0	

**Tarea-3:** Obtener las ecuaciones lógicas de las señales D1, D0, sube y baja.

**Tarea-4:** Implementar las ecuaciones lógicas de D1 y D2 con MUX y biestables tipo D activos por flanco de subida, las de sube y baja con un decoder, y las puertas auxiliares que se necesiten.

## PARTE II. Modelado VHDL de la FSM

1) Todos los modelos de las unidades de diseño que se realizarán a lo largo de la práctica, deberán ir documentados con una cabecera que contenga información sobre el diseño y el autor/es. Un modelo de la misma se indica a continuación, **el código también deberá ir comentado**.

```
-----
--© Este diseño es propietario el autor y puede ser utilizado con fines de estudio
-----
-- Proyecto:                Práctica de Sistemas Digitales
--
-- Diseño:                  Máquinas de estados finitos (FSM)
-- Nombre del fichero:      FSM_ascensor.vhd
-- Autor:                   Poner aquí el nombre del alumnos/o/a
-- Fecha:                   19/5/2016
-- Versión:                 1.0
-- Resumen:                 Contiene una entidad y una arquitectura
--                           en estilo de comportamiento de un modelo de ascensor de 3 pisos
--                           Actúa por flanco de subida de reloj
--                           Se utilizan el tipo de datos STD_LOGIC para todas las señales
--                           Excepto para la señal piso_donde_va que será INTEGER
--
-- Modificaciones:
--
-- Fecha                    Autor                    Versión            Descripción del cambio
-----
-- =====
```

2) Editar un fichero que contendrá la entidad de la FSM del ascensor (**FSM\_ascensor**) y la arquitectura en estilo de comportamiento (**FSM\_arquitectura**). El nombre del fichero será igual que el de la entidad, **FSM\_ascensor.vhd**.

La entidad de la FSM, que especifica el nombre, el modo I/O y el tipo de datos de las señales se indica a continuación, los nombres indicados son de uso obligatorio, así como el tipo de datos. Si no se respetan los nombres indicados, no podrá usarse el modelo de test proporcionado.

ENTITY FSM_ascensor IS			
PORT(clk	: IN	STD_LOGIC;	-- activo por flanco de subida
codigo_piso	: IN	STD_LOGIC_VECTOR(1 DOWNTO 0);	-- código binario del N° de piso pulsado
sube, baja	: OUT	STD_LOGIC;	-- dirección movimiento del motor
piso_donde_esta	: OUT	STD_LOGIC_VECTOR(1 DOWNTO 0)	-- código binario del piso donde se encuentra el ascensor
);			
END FSM_ascensor;			

La arquitectura deberá realizarse **OBLIGATORIAMENTE** utilizando la sentencia concurrente **PROCESS** y dentro de ella la sentencia secuencial **CASE**, y otras sentencias secuenciales que se consideren necesarias. Se modelará mediante dos procesos, que tendrán la siguiente funcionalidad:

- Proceso-1: Generar el estado futuro y las salidas.
- Proceso-2: Actualizar el estado.

3) Realizar la simulación con el test bajado de Moodle: **FSM\_ascensor\_tb.vhd** durante 200 ns, visualizando las formas de onda de las señales, **OBLIGATORIAMENTE** en el siguiente orden: clk, codigo\_piso, piso\_donde\_esta e\_actual, e\_futuro, sube, baja. Los nombres de las señales deberán visualizarse completos, como se indica en la figura, se valorara negativamente las capturas donde figuren los nombres cortados.

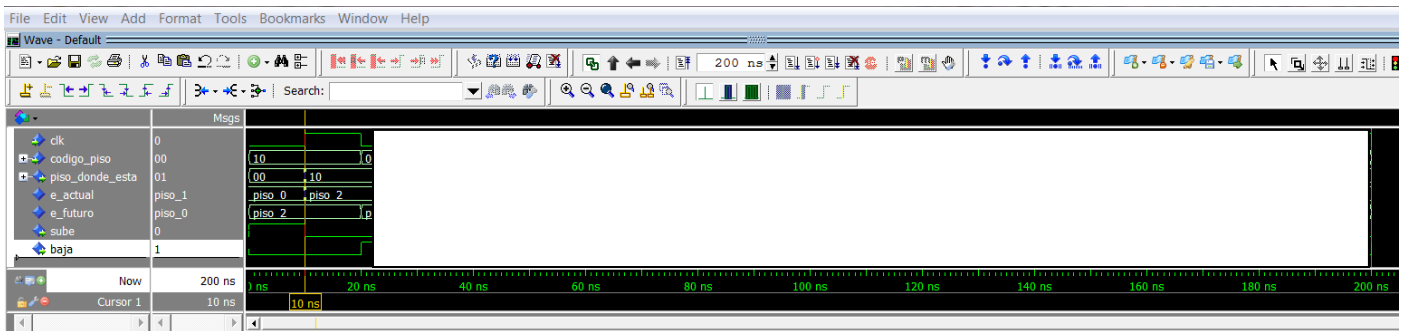


Figura 3. Orden de las señales para la simulación de la FSM

**Tarea-5:** Capturar los resultados de simulación e interpretar si son correctos. Sobre ellos, explicar comentando los valores que tienen las señales, y que es lo que está ocurriendo cuando el reloj toma los valores de 25 y 105 ns.

**Tarea-6:** Captura de pantalla del contenido de la librería WORK, tras realizar la simulación.

### PARTE III. Modelado VHDL del codificador del piso activado

Esta tarea, ya se realizó en la práctica: *Modelado VHDL de un codificador binario para la elección del piso de un ascensor*. Se reutilizará el código realizado en ella, el cual implementaba la funcionalidad de la Tabla 2. El fichero se denominaba **codifica\_boton.vhd** y contenía la entidad (**codifica\_boton**) y la arquitectura (**arquitectura\_cod\_boton**) realizada en estilo de comportamiento.

4) Para garantizar que este componente funciona correctamente antes de integrarlo en la estructura completa, se realizará la simulación del codificador con el fichero **codifica\_boton\_tb.vhd** disponible en Moodle. Realizar la simulación durante 200 ns visualizando las formas de onda de las señales.

**Tarea-7:** Capturar los resultados de simulación e interpretar si son correctos. Los nombres de las señales deberán visualizarse completos. Se valorará negativamente las capturas donde figuren los nombres cortados.

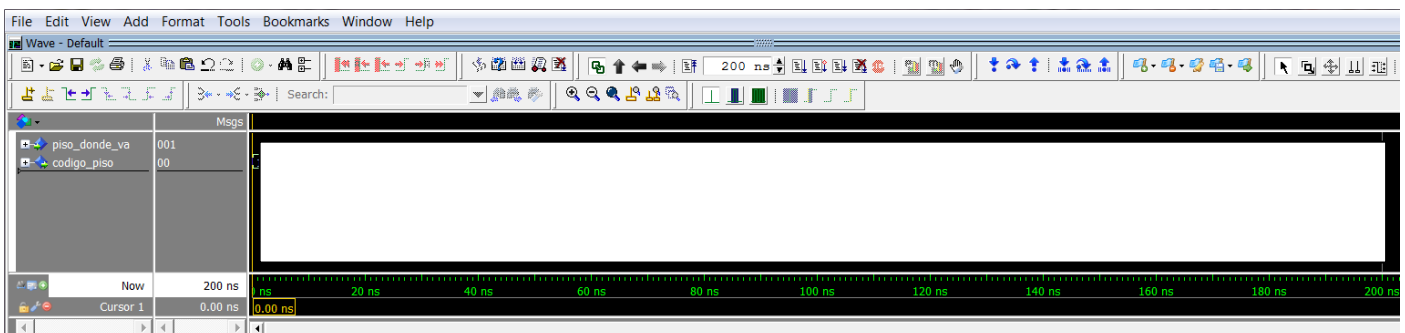


Figura 4. Orden de las señales para la simulación del codificador del botón

## PARTE IV. Modelado VHDL del sistema completo

En este apartado se modelará la funcionalidad del sistema completo, conectando los dos subsistemas realizados en los apartados anteriores, tal como se indica en la Figura. 1.

5) Editar un fichero que contendrá la entidad (**ascensor\_completo**) y la arquitectura (**arquitectura\_ascensor\_completo**). El nombre del fichero será igual que el de la entidad, **ascensor\_completo.vhd**. El modelo de arquitectura deberá ser realizado en estilo estructural, conectando los dos componentes de que consta el sistema: **codifica\_boton** y **FSM\_ascensor**.

6) Editar un nuevo fichero que contendrá la entidad para comprobar la funcionalidad del ascensor (**ascensor\_completo\_test**) y una arquitectura en estilo estructural (**ascensor\_completo\_arquitectura\_test**). El nombre del fichero será **ascensor\_completo\_test\_tb.vhd**. La señal de reloj se definirá de forma algorítmica y tendrá un periodo de 20 ns, será igual que la que se definió en el test de la FSM, y las entradas tendrán los mismos drivers que las incluidas en fichero **codifica\_boton\_tb.vhd**, **ambos** disponibles en Moodle.

7) Realizar la simulación durante 200 ns visualizando las formas de onda de las señales, **OBLIGATORIAMENTE**, en el siguiente orden: clk, piso\_donde\_va, código\_piso, piso\_donde\_esta, e\_actual, e\_futuro, sube, baja. Los nombres de las señales se deben visualizar completos.

En la siguiente figura, se muestra en la parte de la izquierda la jerarquía del diseño y sus componentes, las unidades de diseño y sus tipos. Y en la de la derecha las señales (objetos) del componente que se ha seleccionado. Para poder visualizar las señales que se indican en el párrafo anterior, se tendrá que seleccionar previamente, la unidad de diseño donde se encuentren. La parte superior de la figura muestra las señales de la unidad de diseño ascensor\_completo\_test, mientras que la parte inferior, muestra las del componente FSM.

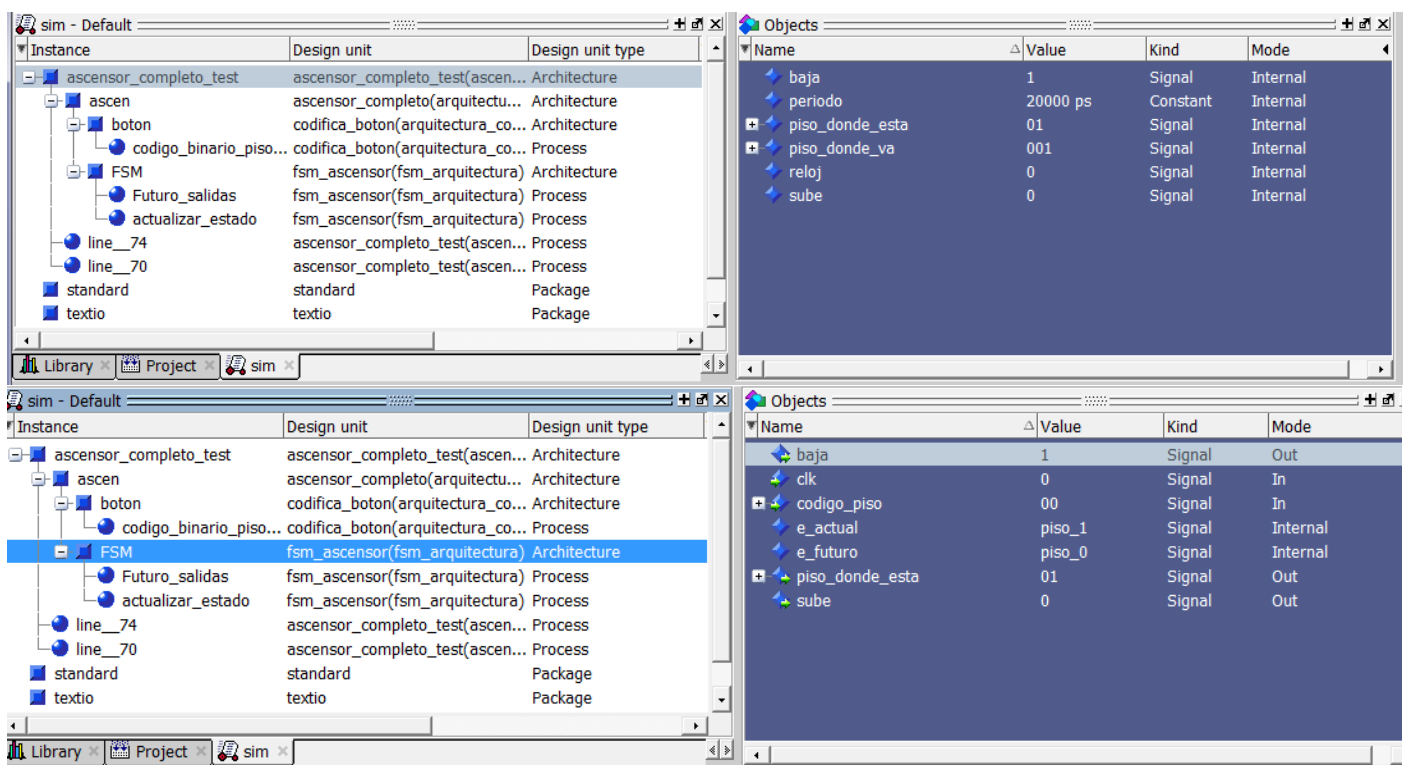


Figura 5. Jerarquía del diseño y señales visibles de la unidad de diseño seleccionada



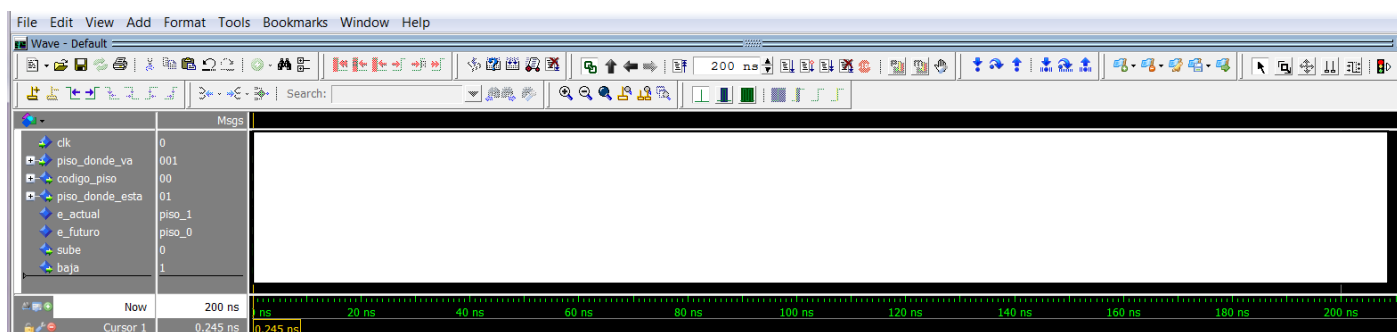


Figura 6. Orden de las señales para la simulación del ascensor completo

**Tarea-8:** Capturar los resultados de simulación e interpretar si son correctos.

**Tarea-9:** Captura de pantalla del contenido de la librería WORK, tras realizar la simulación completa