

**Trabajo de Electrónica Digital:  
*Chinchimoni* Digital**



**RESUMEN**

Este documento contiene las normas y la especificación del trabajo de la asignatura de Electrónica Digital del curso 2025/2026.

**NORMAS DEL TRABAJO DE CLASE**

El presente trabajo es obligatorio y representa un 30% de la nota final de la asignatura. El trabajo debe presentarse para su evaluación antes del examen final, ya sea la convocatoria que sea. Si la asignatura se suspende en febrero, la nota del trabajo se guarda para julio del mismo año, pero no es posible aprobar la asignatura sin volver a examinarse (la nota de cada examen final no se guarda para la siguiente convocatoria, por lo que no es posible aprobar presentando un trabajo mejor después del examen o presentándose sólo a la parte de VHDL). Es necesario obtener un mínimo de 4 (sobre 10) en el examen final (la parte de diseño) para aprobar la asignatura, ya sea en febrero o en julio.

El trabajo se realizará en los grupos ya definidos y comunicados en Moodle, y la entrega se hará mediante una entrevista personal, en la que se demostrará el circuito configurado en una FPGA del *laboratorio*. Todos los alumnos deberán estar presentes durante la entrega, en la que el profesor encargado realizará, además, una entrevista con todos los miembros del grupo. Posteriormente, deberán entregarse los ficheros fundamentales del diseño (ficheros de descripción y bancos de prueba con las simulaciones realizadas) a través de Moodle.

### OBJETIVO DEL TRABAJO

El objetivo del trabajo es el diseño de un circuito mediante la herramienta Xilinx Vivado, utilizando el lenguaje VHDL como lenguaje de entrada. El circuito deberá ser simulado sobre la herramienta Vivado, y posteriormente probado sobre las placas Pynq-Z2 disponibles en el *laboratorio*, con la *FPGA Zynq-7020* y la placa de expansión de entradas/salidas con LEDS, pulsadores, botones y *displays* de siete segmentos. En la Figura 1 se muestra la placa de desarrollo que se va a emplear y los elementos de la placa de expansión que se necesitan para este trabajo.

La especificación del circuito consta de una funcionalidad mínima, requerida para todos los trabajos, pero luego se podrá optar por implementar opciones avanzadas del juego, con las que se podrá mejorar la nota del trabajo. De cualquier forma, la decisión de implementar una opción o no se basa en el número de horas, medidas con detalle, que se hayan utilizado. Se calcula que el trabajo no debería llevar más allá de veinte horas de trabajo real por persona, por lo que, si llegado este tiempo no se han completado las fases opcionales, es mejor no hacerlas. En la entrevista, a cada grupo se le preguntará por el número de horas que cada alumno ha empleado.

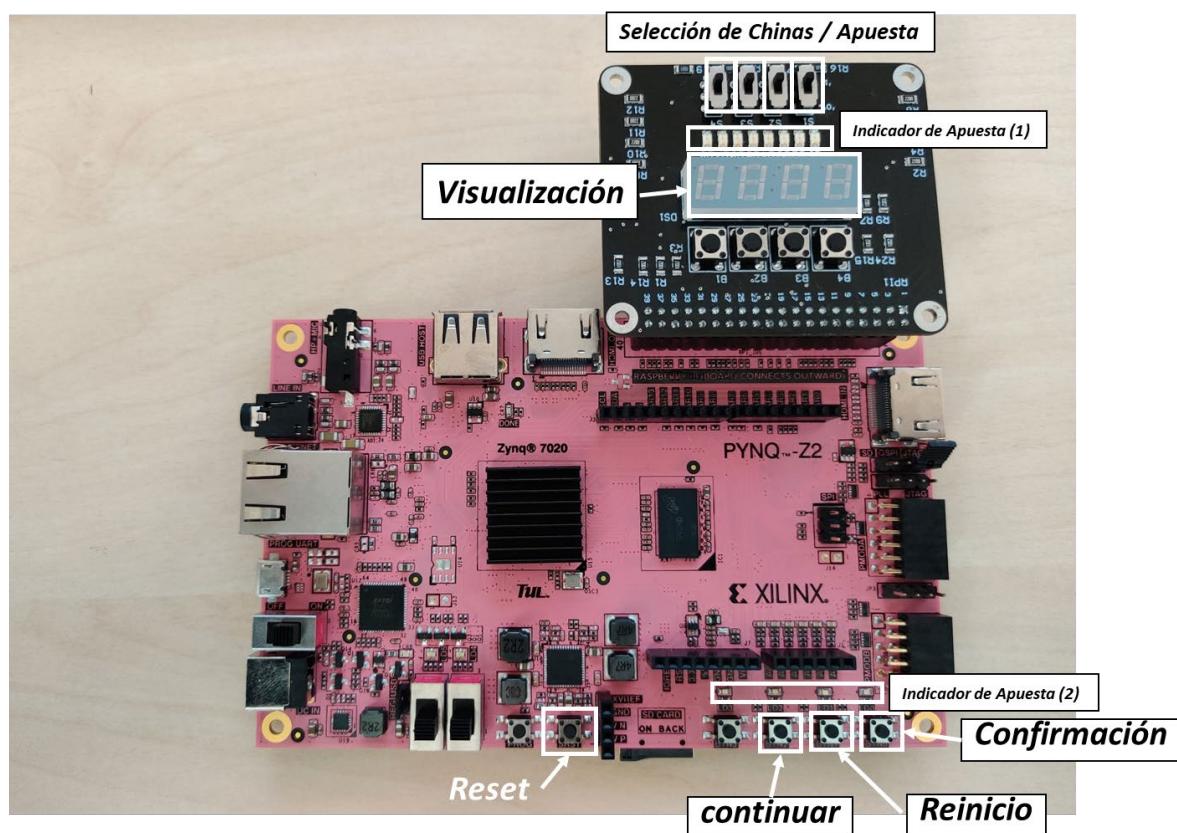


Figura 1: Placa de Desarrollo y Placa de Expansión con dispositivos de E/S que se van a emplear en el trabajo

### **ANTECEDENTES HISTÓRICOS**

(Extracto del artículo del Diario de Leon: [El chichimoni, de origen leonés](#))

Era el año 1746 cuando un pastor, natural de Bercianos del Camino, motivado por sus largas horas sin distracción por el campo, decidió coger tres piedras y echar a suertes con unos compañeros a ver quién adivinaba cuantas «piedrecitas» tenía en la mano. Surgió así el llamado [...] *chichimoni*, como se le llama en la provincia. La trashumancia del pastoreo favoreció la difusión del juego por toda la península.

[...]

El *chichimoni* es un juego de azar, astucia y psicología, para niños y mayores, que consiste en intentar adivinar el número total de monedas que varios jugadores (al menos 2) guardan en su mano. Se utilizan 3 monedas por jugador y se pueden sacar 0, 1, 2 ó 3 monedas. El juego se usa típicamente como fórmula de echar a suertes quién paga la consumición en un bar.

[...]

En cada ronda, cada jugador guarda a escondidas, entre ninguna y 3 monedas en su mano, que a continuación muestra cerrada al resto de jugadores, con el brazo estirado delante de sí. Entonces cada jugador por turno dice una cifra, intentando adivinar cuantas monedas suman todas las manos. No se puede repetir una cifra ya dicha por otro. A continuación, se abren las manos y se determina quién ha acertado, y se pasa a la siguiente ronda. Lo habitual es rotar los turnos en sentido contrario a las agujas del reloj. En la primera ronda no se permiten «blancas» (es decir, ninguna moneda) y no se puede mentir (p. ej., sacar 3 monedas y decir dos).

### **FUNCIONALIDAD BÁSICA**

El circuito debe implementar la funcionalidad del tradicional juego del *chinchimoni*, en versión electrónica, en la que un número variable de uno a tres jugadores humanos deberán jugar contra la máquina, que se comportará como un jugador más. Cada jugador tendrá 3 monedas, de las cuales podrá sacar 0, 1, 2 o 3 en cada turno. Los jugadores decidirán, en primer lugar, cuántas monedas quieren sacar. Posteriormente, y por orden, deberán adivinar el número total de monedas que se mostrarán. Con dos jugadores, el número máximo estará entre 0 y 6; con tres, entre 0 y 9; y con cuatro, entre 0 y 12. Una vez que el último jugador ha realizado su apuesta, el sistema comprobará el número total de monedas extraídas por todos los jugadores y, si este valor coincide con la apuesta previa de alguno de ellos. Si se da ese caso, ese jugador será el ganador de la ronda. El ganador del juego será el primero en alcanzar tres rondas ganadas.

### **El juego**

Todas las partidas del *chinchimoni* digital seguirán el mismo procedimiento. Se comenzará eligiendo el número de jugadores mediante los cuatro switches de la placa de expansión. Se aceptará cualquier número de jugadores entre 2 y 4. La propia

máquina será siempre el jugador número 1. La selección de los switches se confirmará mediante una pulsación en el botón de confirmación (ver figura 1), como se detalla en el apartado “Selección de número de jugadores”. **El valor introducido mediante los switches se considerará siempre en formato binario.**

Una vez introducido el número de jugadores, se pasará a seleccionar cuántas piedrecitas desea sacar cada jugador. Esta selección siempre se hace en el mismo orden, del jugador 1 en adelante. El modo de seleccionar el número de piedrecitas se describe en el apartado “Extracción de piedrecitas”. Una vez que cada jugador ha decidido el número de piedrecitas que sacará, se pasará a intentar adivinar el total de piedrecitas que sacarán todos los jugadores. Para ello, cada jugador apostará el valor que estime oportuno, sabiendo cuántas piedrecitas ha decidido extraer. Este procedimiento se detalla en el apartado “Introducción de la apuesta”. En este punto serán fundamentales la astucia de cada jugador y el conocimiento del adversario. Una vez que cada jugador hace su apuesta, se pasa a resolver la ronda, tal y como se indica en el apartado “Resolución de la ronda”. Tras ello, si alguno de los jugadores acierta con el número total de piedrecitas introducido, se incrementará su contador de rondas ganadas. Tan pronto como alguno de los jugadores alcance un total de tres rondas ganadas, se declarará el vencedor del juego.

Cada pantalla se muestra por defecto durante 5 segundos, como se detalla a continuación. Sin embargo, si en cualquier momento el jugador pulsa el botón de **continuar**, se pasa directamente a la siguiente pantalla del *display*, sin esperar este tiempo.

A continuación, se presentan más detalles sobre cada una de las etapas del juego.

### Selección del número de jugadores

Antes de empezar a jugar, se deberá elegir el número de jugadores mediante los cuatro switches de la placa de expansión. Para ello, al arranque del juego, se mostrará en los siete segmentos, de la forma que se indica JUG\_, como se muestra en la Figura 2 (derecha):

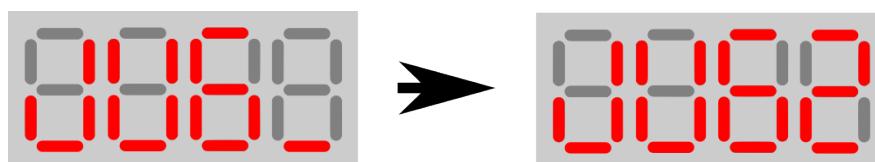


Figura 2 – Indicación de la selección del número de jugadores

Se aceptará cualquier número de jugadores entre 2 y 4. La propia máquina será siempre el jugador número 1. La selección de los switches se confirmará al pulsar el botón de confirmación. El valor debe introducirse, codificado en binario, mediante dichos switches. En ese momento, el número de jugadores seleccionados se mostrará en el *display* de la derecha, como se muestra en la Figura 2 (derecha), para el caso de dos jugadores seleccionados. Esta selección se mostrará en el *display* durante 5 segundos y, después, se pasará al turno de introducir la apuesta del jugador 1.

En caso de seleccionar un número menor que dos o mayor que cuatro jugadores, el valor no se confirmará, y el *display* mostrará el modo de selección (Figura 2, izquierda) hasta que se introduzca un número de jugadores válido.

### Extracción de piedrecitas

La primera parte de cada juego consiste en que los jugadores, por orden, deben indicar cuántas “piedrecitas” quieren sacar en esta ronda. Será un valor entre 0 y 3, sin permitir 0 en la primera ronda del juego.

Para mostrar el turno en el que cada jugador debe introducir su número de piedrecitas, en los cuatro *displays* se mostrará el código ‘ch’, seguido del número del jugador que debe realizar su selección. Como se ha indicado, el jugador 1 es la máquina. En la figura 3(a) se muestra un ejemplo de cómo solicitarle al jugador 2 que meta el número de piedrecitas que quiere sacar.

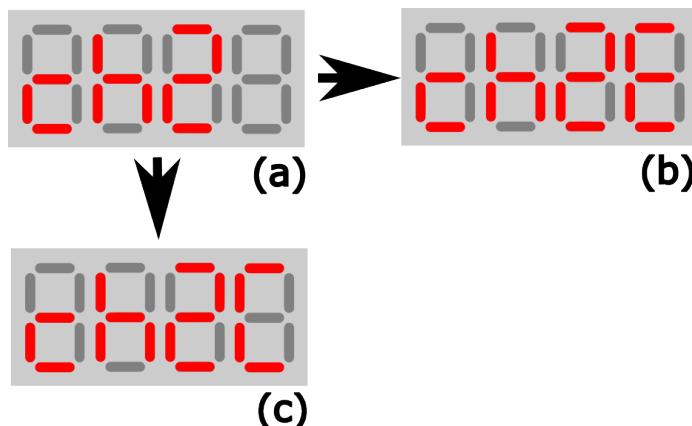


Figura 3 – Indicación de la selección del número de piedrecitas a extraer por cada jugador

En el caso de jugadores humanos, realizarán su selección ingresando el valor válido de piedrecitas, codificado en binario, mediante los cuatro switches de la placa. La selección se confirmará al pulsar el botón de confirmación. En caso de realizar una selección errónea, se mostrará durante 5 segundos el código de error ('E'), como se muestra en la figura 3(b). Durante esos 5 segundos, el sistema no responderá a una nueva selección del usuario. Pasado este tiempo, se volverá a mostrar el mensaje de que se requiere introducir un valor (figura 3(a)). Se repetirá esta secuencia hasta que la selección sea válida. En caso de realizar una selección válida, se mostrará durante otros 5 segundos el mensaje de confirmación ('C'), tal y como se muestra en la Figura 3(c). Despues se pasará a la selección del siguiente jugador o a la introducción de la apuesta, según corresponda.

Como ya se ha indicado, el jugador 1 es la máquina. También tendrá su turno para ingresar el valor. Para que el resto lo visualice, se forzará a mostrar el mensaje “ch1” durante 5 segundos, seguidos de otros 5 segundos para indicar que la máquina ha realizado la selección correcta (“ch1C”). Se garantizará que la máquina seleccione un valor aleatorio correcto en cada jugada.

En esta fase del juego no se muestra en el *display* el número de piedrecitas que se van a sacar, a propósito, para evitar dar esa información al resto de jugadores. En caso de

que haya más de un jugador humano, no se debe permitir que el resto de los jugadores vea cuántas piedrecitas sacarán los demás.

### Introducción de la apuesta

La segunda parte de cada ronda está dedicada a que los jugadores introduzcan sus apuestas. En este caso, el orden variará, ya que el último en introducir su apuesta tendrá ventaja. En la primera ronda, comenzará el jugador 1, seguido por los demás hasta que todos hayan hecho su apuesta. En la segunda ronda, el turno inicial pasará al jugador 2 y el jugador 1 apostará al final. Este patrón continuará de forma sucesiva, aplicando un desplazamiento circular en el orden de los jugadores en cada ronda.

El turno de apuesta del jugador correspondiente debe mostrarse con el código 'AP', seguido del número del jugador, como se muestra en la figura 4. El jugador que deba introducir su apuesta lo hará seleccionando el valor con los cuatro switches, seguido de pulsar el botón de confirmación. En este caso, la apuesta deberá ser un valor entre 0 y el número de jugadores multiplicado por 3, ya que este sería el máximo de piedrecitas que podrían sacarse entre todos. Al igual que en la fase de extracción de monedas, el procedimiento de introducción de apuesta irá acompañado de cambios en los *displays*. La selección correcta se mostrará al añadir el código 'C' a continuación del código 'AP' básico de selección de apuesta (figura 4 (c)), mientras que una selección errónea se marcará al añadir una 'E' al final del mensaje, como se muestra en (b).

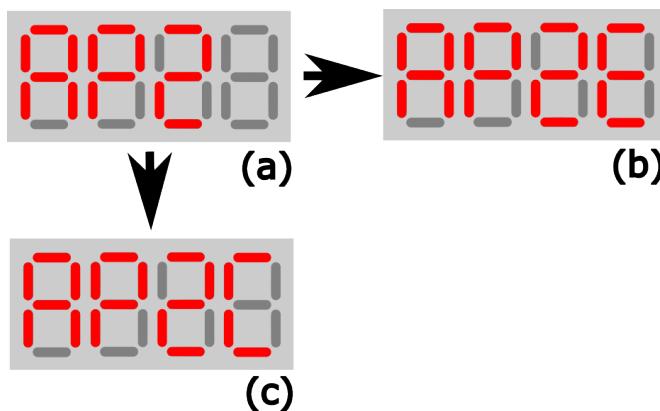


Figura 4 – Indicación de la apuesta de cada jugador

Los mensajes de *Correcto* o *Erróneo* se mantendrán en los *displays* durante 5 segundos, una vez se pulsa el botón de confirmación. En caso de apuesta correcta, se pasará al siguiente jugador o a la resolución de la ronda, según corresponda. En caso de introducir una apuesta incorrecta, se repetirá el procedimiento de introducción de apuesta tantas veces como sea necesario. No se permite repetir la apuesta realizada por un jugador anterior en el mismo juego. Es decir, sólo un jugador puede apostar un valor determinado en cada ronda.

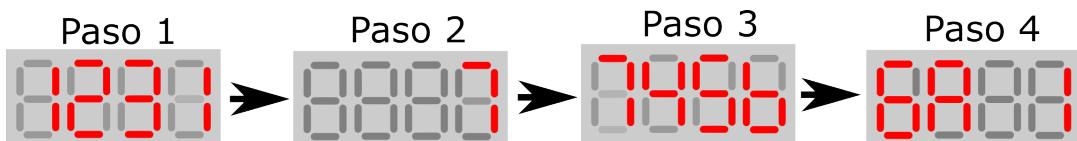
El jugador 1 repetirá este mismo procedimiento como si fuera humano, manteniendo el mensaje AP1 durante 5 segundos, y después el mensaje de AP1C, durante otros 5 segundos. Se garantizará que este jugador automático siempre introduzca una apuesta

correcta y sin mentir, es decir, siempre superior al valor de piedrecitas seleccionado en cada caso.

En este caso, sí es importante que el resto de los jugadores conozca la apuesta de los demás. Por lo tanto, se marcará el valor seleccionado con los 12 leds de la placa (4 de la *Pynq* + 8 de la placa de expansión). El valor se mostrará decodificado, es decir, encendiendo tantos *LEDs* como correspondan al valor de la apuesta introducido mediante los switches. Este valor se mostrará durante los mismos 5 segundos que se muestran el mensaje de confirmación de la apuesta correcta.

### Resolución de la ronda

Una vez que todos los jugadores han realizado su apuesta, se pasa a resolver el juego. Para ello, se mostrará en los cuatro *displays* el número de piedrecitas sacado por cada jugador (paso 1), después se mostrará el valor total de piedrecitas, resultante de la suma de piedrecitas sacadas por cada uno de ellos (paso 2); Despues se mostrará la apuesta de cada jugador (paso 3), y por último se mostrará el mensaje que indica el jugador ganador (paso 4), con el código de ‘GA’, seguido del número de jugador vencedor. Cada uno de estos mensajes se mostrará durante 5 segundos en el *display*.



**Figura 5 – Indicación de los pasos a seguir para la resolución de la ronda**

Se considerará ganador al jugador que acierte la apuesta. Si ninguno lo acierta, se indicará que no hay ganador mediante el código ‘GA 0’.

En caso de haber un ganador, se deberá incrementar su número de rondas ganadas. Para ello, se mostrará en los *displays* el número de rondas de cada jugador, empezando por el primero en el primer *display*, desde la izquierda. Esta pantalla se mostrará durante 5 segundos al finalizar la fase de resolución de la apuesta. Se trataría del paso 5 y se mostraría después de los indicados en la figura de arriba.



**Figura 6 – Indicación de las rondas ganadas por cada jugador**

### Fin del Juego

El jugador que llegue a tres rondas ganadas será el ganador final del juego. Esta situación debe comprobarse tras mostrar el número de rondas ganadas por cada jugador. En caso de que exista un ganador, se deberá mostrar con el código ‘Fin’, seguido del número del ganador. Este mensaje desaparecerá solo cuando se inicie un nuevo juego. Esto se indicará al pulsar el botón de **Reinicio**.

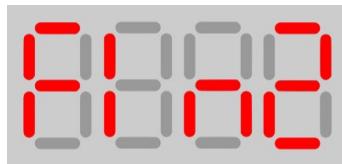


Figura 7 – Indicación de fin de juego y muestra de un jugador.

### Generación de números aleatorios

La generación de números aleatorios se hará mediante contadores que correrán libremente a 125 MHz, desde el instante en que se inicializa la FPGA, generando valores pseudo-aleatorios de forma continua. La ‘aleatoriedad’ se determina por el instante en que el jugador pulse el reset o alguno de los botones, con el objeto de iniciar la partida. Estos contadores no deben inicializarse en ningún momento y seguirán contando permanentemente. La generación de números aleatorios no puede ser sesgada, es decir, todas las posibilidades tendrán la misma probabilidad de ser escogidas.

### Anti-rebotes en las entradas

Todas las entradas procedentes de botones de la placa deben contar con el correspondiente circuito anti-rebotes, tal y como se ha explicado en la práctica 4 de la asignatura.

### OPCIONES AVANZADAS

#### Ganador Aproximado

Se decidirá que, en caso de que ningún jugador acierte la apuesta, se declare ganador a aquel que se aproxime más al valor final.

#### Timeout para realización de apuesta

Para agilizar el juego, se instaurará un *timeout* de 5 segundos para que el jugador realice una apuesta correcta. En caso de no realizarla dentro de ese tiempo, será descalificado de dicha ronda y no podrá sumar puntos hasta la siguiente ronda.

#### Selección del jugador electrónico

En esta opción, el jugador podrá elegir al inicio del juego si el jugador 1 es humano, o es la propia FPGA, tal y como sucede en la descripción del juego básico.

#### Opción libre (peso variable)

Queda a la imaginación y creatividad de los alumnos pensar en implementar, usando los recursos disponibles en la placa de prácticas, otras opciones que hagan más atractivo el juego. Queda a criterio del profesor la evaluación de esta parte extra dependiendo de la originalidad y la complejidad asociada.

### DESCRIPCIÓN DE ENTRADAS Y SALIDAS

El circuito dispondrá de las siguientes entradas y salidas

In	<b>clk</b>	Señal de reloj de 125 MHz.
In	<b>reset</b>	Señal de <i>inicialización</i> , procedente de un pulsador, y que se usa para iniciar un juego nuevo (botón B0 de la PYNQ).
In	<b>B1...B3</b>	Señales procedentes de los pulsadores, y que se emplearán para el control del juego. Es decir, serán las señales <i>continuar</i> , <i>reinicio</i> y <i>confirmación</i> .
Out	<b>segments(7:0)</b>	8 señales para los displays de 7 segmentos (señales a, ..., g) y el punto decimal. Esta señal es activa por nivel bajo.
Out	<b>selector(3:0)</b>	Señales para selección de displays (4 dígitos). El display más a la izquierda es el 3, y el más a la derecha, el 0. El selector es activo por nivel alto.
Out	<b>led(7:0)</b>	Barra con 8 LEDs que muestra los puntos acumulados por cada jugador.
In	<b>mode(1:0)</b>	Señal procedente de dos interruptores de la placa Pynq, y que permitirá seleccionar entre el modo básico, el temporizado, el autónomo, o el modo con las mejoras libres.

**Nota extremadamente importante:** Es requisito de diseño que el circuito sea completamente síncrono, es decir, absolutamente **todos** los biestables que se usen deberán ser síncronos y trabajar con el único reloj de 125 MHz. Es fundamental que se empleen descripciones secuenciales correctas, especialmente en lo que se refiere a los procesos síncronos. **El no cumplir con las reglas básicas de diseño VHDL restará puntos en la nota final de forma significativa, aunque el circuito cumpla la funcionalidad especificada.**

Se aconseja una velocidad de refresco de los displays superior a 100 Hz, pero nunca muy elevado (con frecuencias de MHz los LEDs no llegan a emitir luz).

#### DISEÑO DEL CIRCUITO Y ENTREGA

El circuito se diseñará mediante VHDL. El fichero principal consistirá en una descripción **puramente estructural** en el que la funcionalidad se separa en bloques funcionalmente consistentes. Queda a criterio del grupo la partición interna del circuito en los bloques que se consideren oportunos. En cualquier caso, la interfaz externa del circuito completo será la anteriormente expuesta y se entregarán todos los ficheros .vhd que conformen el diseño completo, así como los ficheros de *test bench* con las simulaciones de todos los bloques.

Aparte, en un documento (Word o PDF) se hará una breve explicación (1 ó 2 páginas, pero no más) de la división en bloques realizada y la funcionalidad de cada señal interna de conexión entre bloques. Si algún bloque se ha diseñado mediante un diagrama de estados, se deberá adjuntar éste en el documento también.

La entrega se realizará por Moodle como un único fichero .zip.

#### VERIFICACIÓN DEL FUNCIONAMIENTO

Para comprobar el correcto funcionamiento del circuito diseñado, se realizará la simulación de todos los bloques fundamentales y del circuito completo. **La realización**

**de dichas simulaciones es obligatoria y serán revisadas sobre el propio simulador por el profesor encargado, durante la entrega.**

Posteriormente, el diseño se probará en las placas FPGA del laboratorio. Cuando los grupos consideren, se citarán con el profesor para la entrevista final, empleando un formulario que se habilitará en Moodle. Esto sucederá siempre, antes de la fecha del examen de la convocatoria correspondiente. Salvo casos muy excepcionales que considere el profesor, sólo hay una oportunidad para realizar la entrega final.

#### **CRITERIOS DE PUNTUACIÓN**

Además de evaluar con una nota entre 0 y 10 la funcionalidad alcanzada por el trabajo, se considerarán otros aspectos:

- El tamaño (*LUTs* y *Flip-Flops*) del circuito resultante (que se mira en la tabla-resumen, tras la síntesis). En este caso, se realizará una comparativa de todos los alumnos de la clase, para cada uno de los modos. Se otorgará una puntuación entre +1 y -1 en este apartado, a sumar directamente sobre la nota alcanzada en la funcionalidad.
- Organización del proyecto y claridad del código. Se realizará un análisis del código proporcionado por los alumnos, así como de la estructura de bloques propuesta. Se otorgará una puntuación entre +2 y -2 en este apartado, a sumar directamente sobre la nota alcanzada en la funcionalidad.
- La ausencia de test-bench así como el incumplimiento de las reglas básicas de diseño serán fuertemente penalizadas (hasta -5 puntos sobre la nota final).
- Se debe tener en cuenta que la entrega del trabajo es a todos los efectos como un examen de la asignatura, por lo que el profesor podrá penalizar o premiar a alguno de los componentes del grupo, en función del conocimiento que demuestren del código realizado, y las respuestas que proporcionen a una serie de cuestiones sobre VHDL y el propio trabajo que irá realizando el profesor.

Es muy importante **contabilizar** las horas de **esfuerzo real** que usa cada alumno en la realización del trabajo. La complejidad que se ha estimado es de 20 horas por alumno. Si al cubrir este número de horas sólo se ha abordado la funcionalidad básica, es conveniente no hacer otras, para no extender el tiempo dedicado al trabajo. El profesor también preguntará cuántas horas ha dedicado cada alumno por separado, y que cada uno le explique el bloque que él decida.

#### **DOCUMENTACIÓN ADICIONAL**

Ésta es la primera versión del documento de especificaciones, aunque conviene mirar periódicamente en Moodle para posibles actualizaciones del documento.

A parte del documento de especificaciones, es conveniente mirar los guiones de prácticas de la asignatura, también disponibles en Moodle.