

## Trabajo de Electrónica Digital: Diez Mil Digital



### 1. RESUMEN

Este documento contiene las normas y la especificación del trabajo de la asignatura de Electrónica Digital del curso 2022/2023.

### 2. NORMAS DEL TRABAJO DE CLASE

El presente trabajo es obligatorio y cuenta un 30% de la nota final de la asignatura. El trabajo se debe presentar para su evaluación **antes del examen final**, 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 poder aprobar la asignatura, ya sea en febrero o en julio.

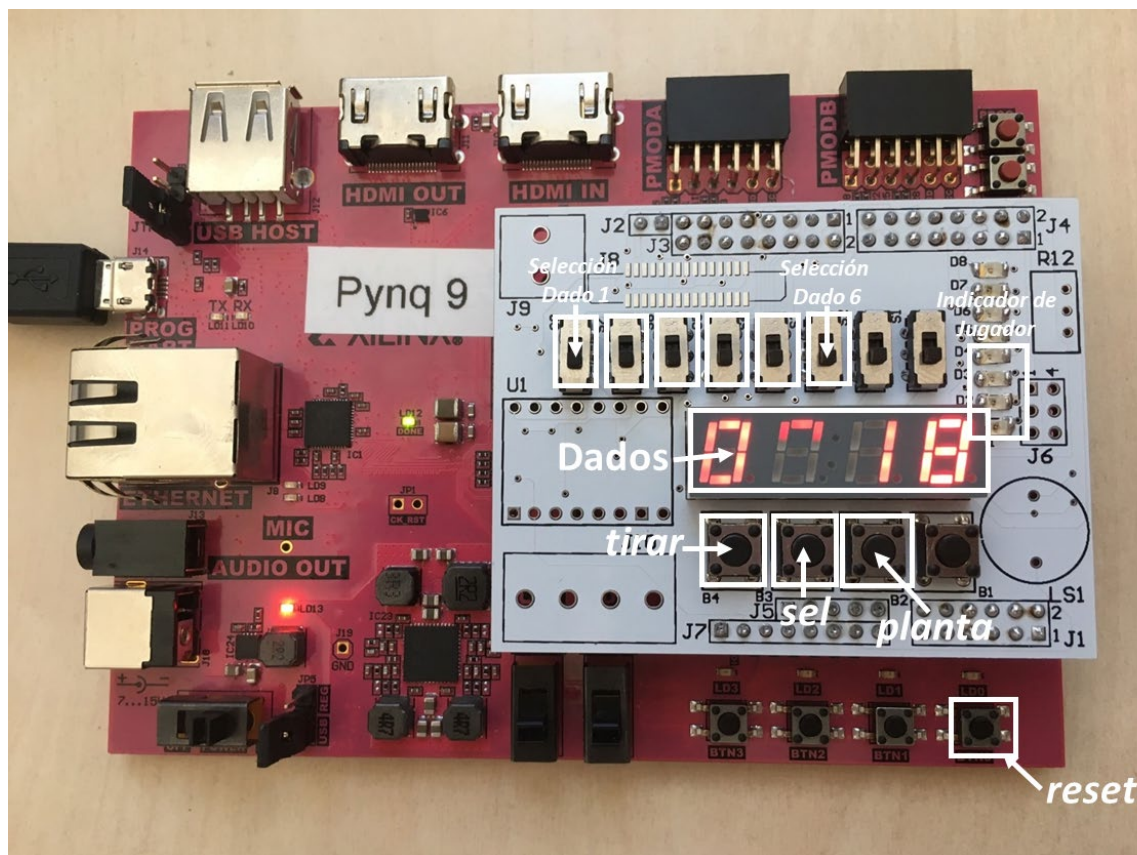
El trabajo se realizará en grupos de hasta cuatro personas, y la entrega del trabajo se hará mediante una entrevista en persona en el laboratorio, en la que se realizará una demostración del circuito configurado en una FPGA del *WebLab* remoto, así como en la propia FPGA del laboratorio. Todos los alumnos del grupo que está siendo evaluado deberán estar presentes durante la entrega, en la que el profesor encargado realizará además una entrevista con todos los miembros del grupo. **Antes de la revisión se**

**deberán entregar los ficheros fundamentales del diseño** (ficheros de descripción y bancos de prueba con las simulaciones que se hayan realizado) a través de Moodle.

### 3. OBJETIVO DEL TRABAJO

El objetivo del trabajo es el diseño de un circuito mediante la herramienta Xilinx Vivado, usando código VHDL como lenguaje de entrada. El circuito deberá ser simulado y probado sobre las placas *Pynq* disponibles en el *WebLab*, con la *FPGA Zynq-7020* y la placa de expansión de entradas/salidas con LEDs, *switches*, botones y *displays* de siete segmentos. Durante la semana anterior a la entrega del trabajo, y una vez que lo tengáis funcionando sobre el *WebLab*, debéis venir a los laboratorios de electrónica para pedirnos una placa *Pynq* física sobre la que deberéis descargar vuestro trabajo funcionando. 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 viene dada por 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 preguntará por el número de horas empleadas por cada alumno.



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

#### 4. FUNCIONALIDAD BÁSICA

El circuito debe implementar la funcionalidad del juego de dados conocido como **Diez Mil o Farkle**. Se trata de un juego de dados en el que un mínimo de dos jugadores, por turnos, van tirando seis dados, y acumulando puntos en función de las combinaciones que obtienen. Dentro de su turno, el jugador puede volver a tirar algunos de los dados para intentar acumular un mayor número de puntos. La clave del juego es decidir el momento óptimo en el que el jugador se debe plantar, valorando las probabilidades de obtener una mejor combinación de dados, frente al riesgo de perder todos los puntos ya acumulados en la jugada, cosa que sucedería en caso de no obtener ninguna combinación premiada en la nueva tirada. El objetivo del juego es llegar a **10.000 puntos**, de forma que el primer jugador que alcance o supere esa cantidad será el ganador.

##### 4.1 Mecánica del juego

En la versión básica del juego se enfrentarán siempre dos jugadores. La mecánica del juego es muy sencilla. Los jugadores, por turnos, tirarán los dados. Con el resultado de los dados se obtendrá una puntuación, calculada como sigue:

- *Uno*: cada dado con valor uno vale 100 puntos.
- *Cinco*: cada dado con valor cinco vale 50 puntos.
- *Tres dados iguales*: valen el número que se repite, multiplicado por 100. Por ejemplo, tres cuatros valdrán 400 puntos. Como excepción, si sale 1-1-1, la puntuación será de 1000.
- *Cuatro, cinco, o seis dados iguales*: Tendrán 1000, 2000 y 3000 puntos, respectivamente, independientemente del valor de los dados.

La tabla completa de puntuaciones es la siguiente<sup>1</sup>:





PUNTUACIONES FARKLE	
 = 50 pts.	 = 100 pts.
 = 1000 pts.	4 iguales = 1000 pts.
 = 200 pts.	5 iguales = 2000 pts.
 = 300 pts.	6 iguales = 3000 pts.
 = 400 pts.	Escalera 1-6 = 1500 pts.
 = 500 pts.	Tres parejas = 1500 pts.
 = 600 pts.	4 iguales y 1 pareja = 1500 pts.
	Dos tríos = 2500 pts.

Figura 2: Resumen de puntuaciones del juego Diez Mil.

<sup>1</sup> <https://www.ideasverdes.es/como-jugar-a-farkle-o-10-000-juego-con-solo-dados-y-papel/>

En la versión básica no se tendrá en cuenta el cálculo de la escalera, las tres parejas, los 4 iguales y pareja, ni de los dos tríos, que se muestran en la figura 2 rodeados por el cuadro de línea discontinua. Esas combinaciones se considerarán sólo en la versión extendida.

Tras tirar los dados, el jugador puede decidir plantarse y agregar todos los puntos de ese turno a su puntuación total, o tirar otra vez. En el caso de que decida seguir jugando, puede decidir reservar algunos de los dados (acumulando las puntuaciones asociadas a esos dados en un registro parcial del turno), volviendo a tirar sólo el resto. Los dados reservados por el jugador se retiran del tablero, no pudiendo ser empleados para realizar nuevas combinaciones.

Si un jugador hace puntos con todos los dados que quedan en el tablero, puede plantarse o volver a tirar los seis dados otra vez. El turno del jugador finalizará irremediabilmente cuando haga una tirada sin poder hacer ninguna combinación que supongan más puntos. Si esto sucede, además de finalizar su turno, el jugador perderá todos los puntos acumulados en la jugada.

#### 4.2 Versión digital del juego

Empezará el juego siempre por el primer jugador, pasando después al resto de jugadores, en orden creciente. Tras el turno del último jugador, se volverá al primero. Cada tirada consistirá en los pasos sobre la placa que se describen a continuación.

En primer lugar, **se mostrará el número de jugador al que le toca jugar** mediante los LEDs de la placa de expansión. Para el primer jugador se enciende el LED inferior, para el jugador dos se encienden los dos LEDs inferiores y, en caso de más jugadores, se haría de esta misma manera sobre el número de jugador al que le toca su turno.

Una vez indicado su turno, el jugador debe **tirar los dados** (mediante el botón *tirar*), y el resultado se visualizará en los *displays* de siete segmentos. El valor de cada dado se representará con los números del 1 al 6. El resultado de la tirada se mostrará mediante un *scroll* horizontal, tal y como se detalla en la sección 4.3.

Si tras la tirada se obtienen combinaciones de dados que den lugar a puntos, el jugador **podrá decidir si se planta o no**. Para plantarse deberá pulsar el botón *planta* de la placa. La opción de plantarse dejará de estar disponible si se realiza una selección de los dados que se mantienen en el tablero (mediante pulsación en el botón *se/*), como se verá a continuación. En caso de plantarse, se mostrarán los puntos acumulados en la jugada durante cinco segundos, y los puntos totales acumulados por el jugador, durante los cinco segundos siguientes. Pasado este tiempo se apagarán los *displays*, y se indicará el cambio de jugador con los LEDs.

Si no se planta, el jugador deberá **decidir que dados mantiene en el tablero**, y cuales retira. Esta selección se hará empleando los *switches* de la placa de expansión, tal y como se detalla en la sección 4.4. La selección se confirma con el botón *se/*. Una vez pulsado este botón, se mostrará en los *displays* **la puntuación parcial** que acumula con los dados **retirados del tablero**. En la sección 4.5 se ofrecen más detalles sobre el cálculo de puntuaciones. En este punto, el jugador pulsará de nuevo *tirar* para obtener una

nueva combinación de dados. En caso de volver a obtener puntos, el juego deberá proseguir como se indica en el párrafo anterior. Si un jugador hace puntos con todos los dados que quedan en el tablero, puede plantarse o volver a tirar los seis dados otra vez.

Si al finalizar el turno el jugador acumula 10 000 puntos o más en su casillero total, se mostrará ya como **vencedor**, dando por finalizado el juego.

Sin embargo, si la tirada finaliza sin alcanzar ninguna combinación, **el turno finalizará inmediatamente**. En este caso se deberán mostrar los valores de los dados durante un tiempo tal que se muestren al menos dos veces los dados que se han tirado. A continuación, se mostrará la puntuación de cero en los *displays* durante cinco segundos y la puntuación total durante cinco más, para pasar de nuevo el turno al siguiente jugador. Nótese que en este caso el paso de mostrar dados a mostrar puntuación parcial es automático, es decir, sin que el jugador deba pulsar ningún botón tras la tirada fallida (al contrario de lo que ocurre cuando el jugador se planta).

Se ofrecen a continuación más detalles las principales etapas del juego.

#### 4.3 Representación de los Dados

Tras pulsar el botón *tirar*, los dados se mostrarán usando para ello los cuatro *displays* de siete segmentos disponibles en la placa. Habiendo sólo cuatro *displays* y seis dados, el resultado de la tirada debe irse desplazando por los *displays* haciendo un *scroll* horizontal, siguiendo los siete pasos mostrados en la figura 3. En esta figura se muestra un ejemplo de tirada con resultado de 1-2-3-4-5-6. Cada uno de los pasos tendrá una duración de un segundo. Tras mostrar el valor del último dado (un seis, en este ejemplo) se debe mostrar el carácter separador formado por los tres leds horizontales. Como se ve en la figura, tras siete pasos, la secuencia vuelve a repetirse.

En caso de obtener una puntuación mayor que cero, la secuencia se interrumpirá cuando el jugador active la selección de los dados con los que repetir la tirada (botón *sel*), o decida plantarse (botón *planta*). Si la puntuación es igual a cero, la secuencia finalizará automáticamente pasado un tiempo tal que se muestren al menos dos veces los dados tirados.

En las sucesivas tiradas **se mostrarán sólo los dados no retirados del tablero**, desapareciendo los otros del *display*. Los dados que se vuelven a tirar se mostrarán en las primeras posiciones de los *displays*, independientemente de la posición que tuvieran en la tirada anterior. En este caso, la manera de mostrar los dados es la misma, pero eliminando los pasos del *scroll* que no sean necesarios. Se mantendrá el carácter separador formado por los tres LEDs horizontales antes de iniciar de nuevo la secuencia de los dados.



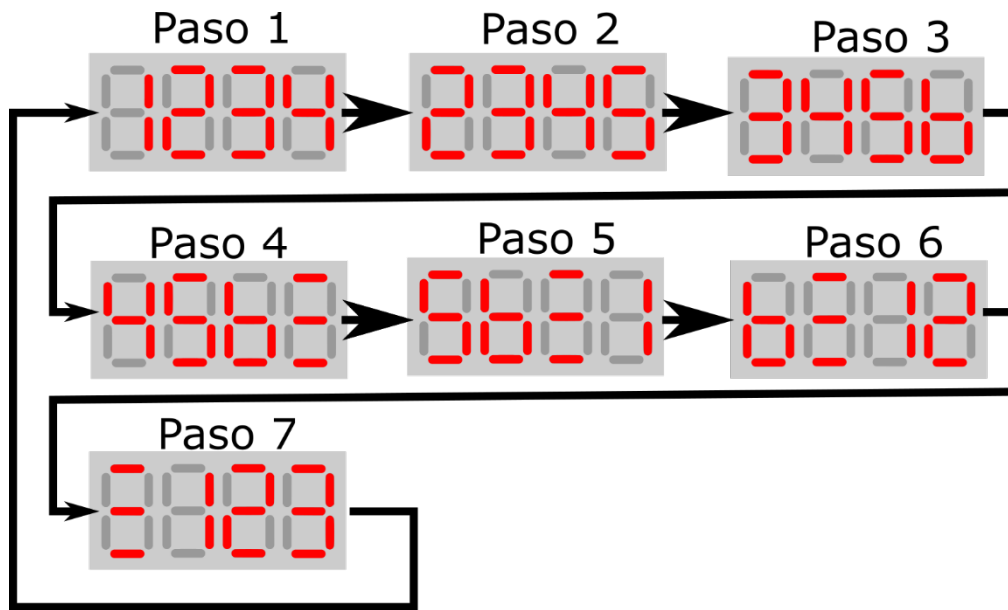


Figura 3: Scroll horizontal para mostrar la tirada de seis dados con resultado 1-2-3-4-5-6 en los cuatro displays

#### 4.4 Selección de los dados

Una vez el jugador ha visto los dados, puede decidir volver a tirar algunos de los dados del tablero, retirando el resto. Los dados retirados aportarán una puntuación que se sumará sobre un acumulador parcial del turno. Sólo se podrán retirar del tablero dados que formen parte de una combinación *premiada*, siempre que los dados que se retiren constituyen una combinación válida por sí mismos. Por ejemplo, si se extraen cuatro treses, el jugador podría retirar tres treses, o los cuatro treses. En el primer caso acumularía 300 puntos, y en el segundo, 1000. También podría decidir no retirar ninguno de los treses. Lo que no sería posible es retirar solo dos de ellos, pues no constituirían una combinación válida por sí mismos.

El jugador deberá poner a uno los *switches* correspondientes a los dados que desea retirar. Los *switches* que se mantienen a cero indican que el dado correspondiente se vuelve a tirar. El orden de selección de los dados se corresponde con el orden que se sigue para mostrar el resultado de la tirada en los *displays*. Es decir, para el ejemplo de la figura 3, el dado uno será el que ha resultado un uno y se controlará con el switch uno, el dado dos es el que ha resultado un dos y se controlará con el switch dos, y así sucesivamente, hasta llegar al dado 6. Es importante tener en cuenta que en tiradas sucesivas dentro de la misma jugada intervendrán menos de seis dados. Así, por ejemplo, si se tiran cuatro dados sólo serán tenidos en cuenta los primero cuatro *switches*. Tras poner los *switches* en la posición deseada, el jugador pulsará el botón *sel* para hacer efectiva su selección.

Puede darse el caso de que el jugador pida retirar dados sin puntuación. En ese caso se invalidará la selección, y se mostrará la letra E en los cuatro *displays* durante dos segundos, para mostrar la situación de error. Pasados esos dos segundos se volverá a mostrar el resultado de la tirada, permitiendo hacer una nueva selección.

En el caso de que se seleccionen para su retirada todos los dados aún en el tablero, el juego asumirá que el jugador quiere seguir jugando (y no plantarse), por lo que se volverán a lanzar los seis dados de nuevo al pulsar *tirar*.

#### 4.5 Detalle del cálculo de las puntuaciones

A la hora de calcular la puntuación, deben tenerse en cuenta una serie de situaciones especiales:

- Las puntuaciones se calculan siempre sobre los dados que el jugador decide retirar del tablero, salvo en caso de que se plante, situación en la que se tendrán en cuenta todos los dados que queden en el tablero.
- Hay combinaciones que pueden ser vistas de varias maneras. Por ejemplo, 5-5-5 podrían ser tres cincos (150 puntos), o un trio de cincos (500 puntos). En estos casos debe prevalecer siempre la opción que dé más puntos al usuario.
- Nunca se debe contar un dado como parte de dos combinaciones. Así, por ejemplo, si se obtiene 4-4-4-4, esta combinación contará siempre como cuatro iguales de cuatro (1000 puntos), y aun siendo también 3 iguales de cuatro, los 400 puntos de esta combinación no se sumarán a los 1000 anteriores.
- Los dados que se retiran del tablero dentro de una jugada no podrán formar parte de las combinaciones del resto de dados que continúan en juego. Por ese motivo se dejarán ya de mostrar en las siguientes tiradas.

#### 4.6 Fin del juego

Tras una tirada, puede darse el caso de que con los dados retirados el jugador acumule un total de puntos que igualen o superen los 10.000. En ese punto el jugador ganará el juego. Esta situación se mostrará de la siguiente manera. Tras pulsar el botón *planta* (si el jugador no se planta obtendrá cero puntos, por lo que será imposible que gane), se mostrarán en los **displays** la puntuación acumulada en la tirada durante cinco segundos. Pasado este tiempo los ocho LEDs de la placa empezarán a parpadear con un periodo de un segundo. Mientras, en los cuatro *displays* se mostrará el número identificador del jugador que ha resultado vencedor. Esta situación se mantendrá hasta que se presione de nuevo *reset* para empezar una nueva partida.

#### 4.7 Ejemplo de tirada

Se muestra un ejemplo de tirada en el que se plantean algunas de las situaciones vistas anteriormente: **1-1-1-5-2-3**.

En este caso el jugador decide retirar del tablero los tres unos (acumulando 1000 puntos), y tira los otros tres dados, obteniendo: **1-5-5**.

En este caso, obtendría  $100+50+50= 200$  puntos extra. En este punto, el jugador podría decidir plantarse, y acumular un total de 1200 puntos a su marcador. También podría decidir acumular los 1200 puntos y volver a tirar los seis dados de nuevo, puesto que ha hecho puntos con todos los dados. Supongamos que el jugador decide volver a tirar.

En la nueva tirada obtiene: **3-3-3-3-1-2**

En este caso decide quedarse con los dados 3-3-3-3-1. Al tener cuatro dados con valor de 3 (3-3-3-3), sumará 1000 puntos a la puntuación de la tirada. Por otro lado, el dado a 1 da 100 puntos extra. Es decir, obtiene 1100 puntos, que se suman a los 1200 anteriores, haciendo un total de 2300 puntos en esta tirada.

Llegados a este punto, el jugador podría bien plantarse y acumular los 2300 puntos sobre lo que tuviera hasta este momento en su casillero, o bien volver a tirar el dado en el que se obtuvo un 2, que aún no da puntos. Supongamos que decide esta segunda opción, obteniendo un **4** en el único dado que aún queda en el tablero.

En este caso, como no es un 1 ni un 5, no obtiene puntos con el dado que ha tirado, finalizando su turno y perdiendo los 2300 puntos parciales que tenía el jugador, y que, por lo tanto, no se acumularían a su casillero.

#### 4.8 Generación de números aleatorios

El resultado de tirar los dados debe ser completamente aleatorio. La generación de números aleatorios se podrá hacer 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*' viene determinada por el instante de tiempo en que el jugador pulse o bien el *reset* o bien alguno de los botones, con el objeto de empezar la partida. (NOTA: correr libremente significa que estos contadores no se deben inicializar en ningún momento y que estarán contando permanentemente). La generación de números aleatorios no puede ser sesgada, es decir, todas las posibilidades tendrán la misma probabilidad de ser escogidas. Como alternativa se propone la implementación de registros de tipo LFSR<sup>2</sup> con distintas semillas y/o estructuras, de forma que se garantice la independencia entre los distintos dados.

### 5. OPCIONES AVANZADAS

#### Tirada mínima para empezar (+0.5 puntos)

Con esta variante se requiere que el jugador obtenga un mínimo de 450 puntos en una jugada para empezar a acumular puntos sobre su casillero total.

#### Juego Temporizado (+0.5 puntos)

Se añaden una serie de contadores de *timeout* para que el jugador tenga un máximo de 10 segundos para hacer la selección de los dados. Pasado este tiempo perderá automáticamente.

#### Selección del número de jugadores (+1 punto)

---

<sup>2</sup> <https://es.wikipedia.org/wiki/LFSR>



Al inicio de la partida se podrá introducir mediante los botones el número de jugadores a participar, entre un mínimo de dos y un máximo de cuatro. Posteriormente, la dinámica del juego se modificará para soportar este número de jugadores.

#### **Puntuación con combinaciones avanzadas (+1.5 puntos)**

Se añadirá el cálculo del resto de opciones de la tabla de la figura 2 que no se consideran en la opción básica. Debe tenerse en cuenta que la combinación en escalera (1-2-3-4-5-6) puede salir en cualquier orden, es decir, por ejemplo, (1-3-5-4-2-6) también será considerada una escalera válida.

#### **Opción libre (A valorar por el profesor)**

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.

### **6. 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.
In	<b>tirar, sel, planta</b>	Señales procedentes de los pulsadores, y que se emplearán para el control del juego. <i>tirar, sel y planta. El cuarto botón no se usa, salvo que se requiera para alguna opción avanzada.</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 el jugador que tiene el turno en cada momento, así como el fin de juego.
In	<b>sw(5:0)</b>	Señal procedente de los switches que permite seleccionar los dados que se mantienen en el tablero, y cuales se retiran.
In	<b>mode(1:0)</b>	Señal procedente de los interruptores 7 y 8, y que permitirá seleccionar entre el modo básico y un modo que permita acelerar las simulaciones.

**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).

## **7. 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*.

## **8. 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 sobre la FPGA disponible en el WebLab remoto de la asignatura y, cuando los grupos consideren, se citarán con el profesor para la entrevista final, empleando para ello un formulario que se habilitará en Moodle. Salvo casos muy excepcionales que considere el profesor, sólo hay una oportunidad para realizar la entrega final. Para la entrega final se deberá contar también con una versión del juego probada sobre la placa.

## **9. CRITERIOS DE PUNTUACIÓN**

Además de evaluar con una nota entre 0 y 7 la funcionalidad básica (sección 4) alcanzada por el trabajo, se considerarán otros aspectos adicionales:

- 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).
- En el caso de implementar opciones avanzadas, estas se puntúan según se indica en el párrafo correspondiente.

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.

#### **10. DOCUMENTACIÓN ADICIONAL**

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

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