

# **Entregable reloj digital. Diseño y síntesis de hardware.**

**Jesús Manuel Muñoz Mazuecos.  
José María Gutierrez Gutierrez.**

## Introducción.

Se ha diseñado un circuito integrado para el control de un reloj digital en el lenguaje de descripción de hardware *vhdl*. La solución consta de un fichero de código y una serie de bancos de prueba para comprobar el correcto funcionamiento del reloj. A lo largo de este documento se explicará la estructura de la solución, algunos fundamentos de diseño, así como problemas encontrados durante el desarrollo.

## Estructura de la solución.

El diseño del circuito integrado se ha especificado en un fichero de código llamado *reloj.vhd*, en donde se especifican todas las entradas, salidas, procesos y señales involucrados que implementan la lógica del reloj digital. A continuación se van a detallar todos los procesos implementados, indicando que señales, entradas o salidas manejan.

- **Entradas del circuito.**

- **clk:** Señal de reloj.
- **rst:** Señal de reset.
- **Botón b1:** Entrada del tipo *std\_logic* correspondiente con el botón b1.
- **Botón b2:** Entrada del tipo *std\_logic* correspondiente con el botón b2.
- **Botón mode:** Entrada del tipo *std\_logic* correspondiente con el botón de mode.

- **Salidas del circuito.**

- **Modo:** vector de *std\_logic* de dos bits que indica el modo en el que se encuentra el reloj.
- **Horas:** vector de *std\_logic* cinco bits que implementa el vector de salida para las horas. El valor de esta salida se verá afectado por el modo en el que se encuentre el reloj.
- **Minutos:** vector de *std\_logic* cinco bits que implementa el vector de salida para los minutos. El valor de esta salida se verá afectado por el modo en el que se encuentre el reloj.
- **Segundos:** vector de *std\_logic* cinco bits que implementa el vector de salida para los segundos. El valor de esta salida se verá afectado por el modo en el que se encuentre el reloj.
- **Sigalarm:** salida del tipo *std\_logic* que indica si la alarma está sonando o no.

- **Señales internas.**

- **Estado:** representa el modo o estado del reloj. Este puede ser: ver hora, cronometro, configurar alarma o configurar hora.
- **Estado siguiente:** representa la siguiente transición en el modo o estado del reloj.
- **Modo hora:** señal de tipo enumeración que se usa para especificar el formato de hora la del reloj (24h o 12h).
- **Modo hora anterior:** señal del mismo tipo que la anterior que se usa para detectar cambios en el formato de la hora.
- **Estado cronómetro:** representa el estado actual del cronómetro. Este puede ser: parado, en marcha, o reinicio.
- **Estado alarma:** representa si la alarma está encendida o apagada.
- **Am:** con esta señal de tipo *std\_logic* sabemos si estamos de madrugada o por la tarde. Esta señal es realmente importante a la hora de cambiar el formato de la hora (24h a 12h y viceversa) o para controlar la activación de la alarma.
- **Am alarma:** esta señal de tipo *std\_logic* nos indica si la alarma tiene que saltar por la tarde o por la mañana. Si hubiésemos implementado sólo el modo 24h, esta señal sería redundante. Sin embargo el hecho de tener dos modos de hora obliga a implementar esta señal.
- **Hora actual:** representa la hora del día en una escala de 0 a 24.
- **Minutos actuales:** representa los minutos transcurridos dentro de cada hora del día en una escala de 0 a 59.
- **Segundos actuales:** representa los segundos transcurridos dentro de cada minuto del día en una escala de 0 a 59.
- **Horas cronómetro:** representa las horas transcurridas desde la última puesta en marcha en una escala de 0 a 24.
- **Minutos cronómetro:** representa los minutos transcurridos desde la última puesta en marcha en una escala de 0 a 59.
- **Segundos cronómetro:** representa los segundos transcurridos desde la última puesta en marcha en una escala de 0 a 59.
- **Hora alarma:** Representa la hora para la que está configurada la activación de la alarma en una escala de 0 a 24.

- **Minuto alarma:** representa los minutos dentro de la hora para la que está configurada la activación de la alarma en una escala de 0 a 59.
- **Segundo alarma:** representa los minutos dentro de la hora para la que está configurada la activación de la alarma en una escala de 0 a 59. Por defecto los segundos de la alarma no se pueden configurar, así que se dejan siempre a cero.

A continuación se va a detallar todos los procesos que se han implementado, especificando la lista de sensibilidad de cada uno, qué señales manejan, cuál es su propósito principal, y si se encargan de activar alguna de las salidas requeridas por la especificación.

- **Proceso *secHora*:** Proceso secuencial que controla la hora actual del reloj
  - Lista de sensibilidad:
    - **clk:** reloj del sistema
    - **rst:** señal de reset.
    - **Modo hora:** señal que indica el formato de la hora.
    - **b1:** botón b1. Utilizado para incrementar las horas en caso de que nos encontremos en modo configuración de hora.
    - **b2:** botón b1: Utilizado para incrementar los minutos en caso de que nos encontremos en modo configuración de hora.
  - Señales actualizadas por el proceso:
    - **Am.**
    - **Estado.**
    - **Horas actuales.**
    - **Minutos actuales.**
    - **Segundos actuales.**

El proceso *secHora* se encarga de llevar el conteo de horas, minutos y segundos, además es sensible a cambios como el formato de visualización la hora, o cambios en la hora en caso de que nos encontremos en modo configuración de hora. En una primera aproximación se tuvo problemas con este proceso debido a que había casos en los que se cambiaba el formato de la hora y se seguía con el conteo de horas respecto al formato anterior dando lugar a incoherencias graves en el valor de las señales. También se optó por diseñar otro proceso que manejara las mismas señales, sin embargo al final nos decidimos por la unificación así cada proceso maneja sus propias señales. De esta manera este proceso puede sólo puede seguir uno de los tres caminos (secuenciales) disjuntos en cada pasada:

- Cambiar el formato de la hora (12h o 24h).
- Seguir contando horas, minutos y segundos.
- Configurar horas. Sólo responde si se pulsan los botones b1 o b2 para incrementar minutos o segundos. Si no se pulsa ningún botón se sigue contando la hora a pesar de estar en modo configuración de hora (igual que en los relojes de Casio).

Si se cambia de estado se sigue con el respectivo conteo de horas minutos y segundos siempre teniendo en cuenta las transiciones entre am o pm. Por último, destacar que este proceso secuencial se encarga de haber las transiciones al siguiente estado.

- **Proceso *secCrono*:** Proceso que controla el paso por distintos estados de cronómetro, así como el conteo de horas, minutos y segundos del cronómetro.
  - Lista de sensibilidad:
    - **clk:** reloj del sistema
    - **rst:** señal de reset.
    - **Estado crono:** es sensible a cambios en el estado del cronómetro. Estos cambios de estado son registrados por un proceso combinacional que se detallará más adelante. El estado puede ser:
      - Parado (por defecto).
      - En marcha.
      - Reinicio.
  - Señales actualizadas por el proceso:
    - **Horas cronómetro.**
    - **Minutos cronómetro.**
    - **Segundos cronómetro.**

Dependiendo del estado del cronómetro se muestran unos valores u otros. Si se hace una transición a otro estado del reloj, los valores de las señales anteriores se mantienen o se van modificando según el estado en que se haya dejado cronómetro.

- **Proceso *secAlarma*:** Proceso secuencial para controlar el salto de la alarma, su activación o su configuración.
  - Lista de sensibilidad:
    - **clk:** reloj del sistema
    - **rst:** señal de reset.
    - **Modo hora:** señal que indica el formato de la hora. Si se cambia el formato de la hora, también debemos el formato de visualización de la alarma. Así se es más coherente con el estado del reloj, además de ser más fácil detectar el salto de la alarma
    - **b1:** botón b1. Utilizado para incrementar las horas en caso de que nos encontremos en modo de configuración de alarma.
    - **b2:** botón b1: Utilizado para incrementar los minutos en caso de que nos encontremos en modo configuración de alarma.
  - Señales actualizadas por el proceso:
    - **Am alarma.**
    - **Sigalarm (Salida del sistema, se activa si la hora de alarma y la actual coinciden).**
    - **Hora alarma.**
    - **Minuto alarma.**
    - **Segundos alarma.**

El proceso *secAlarma* se encarga de hacer saltar la alarma, activar o desactivar la alarma, configurar la alarma mediante la pulsación de los botones b1 o b2, cambiar el formato de visualización de la alarma (12h o 24h). Este proceso alarma es sensible a la señal de reloj, así podemos hacer saltar la alarma cuando la hora de la alarma y la actual coincidan.

En un principio se optó por meter todo dentro de un proceso combinacional todo lo relativo a la configuración, activación o desactivación de la alarma, esta forma hacía el tratamiento de la alarma bastante engorroso, dado que ese proceso combinacional era sensible a las entradas del sistema y compartía código con otros estados. Esta fundamento de diseño hacía un proceso combinacional excesivamente engorroso que se ejecutaba en multitud de ocasiones debido a la gran lista de sensibilidad y en muchas ocasiones sin resultados aparentes debido al estado actual. De esta forma, y como se verá más adelante, los procesos combinacionales se han implementado para manejar cambios de estado tanto del cronómetro como del reloj, escritura en las salidas o notificar cambios en el formato de visualización de la hora.

Por último, cabe destacar que se ha llevado un control en el am o pm de la propia alarma. De esta forma si se cambia el formato de la hora, se cambiará el formato de la alarma automáticamente.

- **Proceso *combEntradas*:** este proceso es sensible a las entradas del sistema y a los cambios de estado. Dependiendo del estado actual, ante la activación de las entradas se modificarán determinadas señales como el formato de visualización de la hora, el estado del cronómetro, o hacer una transición a otro estado.
  - Lista de sensibilidad:
    - **b1:** botón b1.
    - **b2:** botón b2.
    - **Mode:** botón mode.
    - **Estado:** señal estado del reloj.
  - Señales actualizadas por el proceso:
    - **Estado siguiente.**
    - **Modo hora.**
    - **Estado cronómetro.**

Como ya se ha mencionado antes, una de las señales de la lista de sensibilidad es el estado. A continuación se van a detallar las acciones según el estado y las entradas:

- **Ver hora.**
  - **Si b1:** cambiar formato de visualización de hora (12h o 24h).
  - **Si b1 y b2:** pasar a modo configuración de hora.
  - **Si mode:** pasar a modo cronómetro.
- **Cronómetro.**
  - **Si b1:** cambiar estado del crono a reinicio.
  - **Si b2:** poner el crono en marcha o pararlo según el estado anterior del mismo cronómetro.
  - **Si mode:** pasar a configuración de alarma.
- **Configuración de alarma.**
  - **Si mode:** pasar a ver hora.
- **Configuración de hora.**
  - **Si b1 y b2:** Pasar a ver hora.

- **Proceso *combRefrescarDisplay*:** este proceso combinacional tiene como cometido actualizar las salidas, según el estado del reloj.
  - Lista de sensibilidad:
    - **Segundos actuales:** La hora siempre está corriendo. Entonces cada vez que transcurra un segundo debemos actualizar el display según el estado del reloj.
    - **Estado:** estado del reloj.
  - Señales actualizadas (En este caso son salidas del sistema):
    - **Horas.**
    - **Minutos.**
    - **Segundos.**
    - **Modo.**

En cada activación, dependiendo del estado en que se encuentre el reloj se muestran los valores de la hora actual, el cronómetro, los valores de configuración de la alarma o los valores de configuración de la hora.

Dependiendo del estado se codifica la salida modo de la siguiente manera:

- Ver hora: 00.
- Cronómetro: 01.
- Configuración alarma: 10.
- Configuración de hora: 11.

Para terminar con la estructura de la solución cabe destacar que en el código fuente de la aplicación se dispone de comentarios del programador que ayudan a entender el funcionamiento del sistema.



## Bancos de pruebas.

Para comprobar el correcto funcionamiento de la aplicación, se han implementado varios bancos de prueba para verificar el funcionamiento de cada uno de los casos de uso que se han implementado. A continuación se va a explicar con detalle la funcionalidad de estos test.

- **TbModoHora:** Este banco de pruebas simplemente cambia el formato de visualización de la hora (12h o 24h). El fin es comprobar que las conversiones de hora se hacen correctamente.
- **TbCrono:** Este banco de pruebas realiza la siguiente secuencia:
  - Repetir:
    - Cambiar a modo cronómetro.
    - Resetear cronómetro.
    - Encender cronómetro.
    - Ir a modo configuración de alarma.
    - Ir a modo configuración de hora.
    - Volver a modo cronómetro.
    - Parar cronómetro.
    - Reanudar cronómetro.
    - Resetear cronómetro.
    - Encender cronómetro.
    - Ir a modo configuración de alarma.
    - Ir a modo configuración de hora.
- **TbAlarma:** Este test configura una alarma y espera a que salte mientras rota por los distintos estados del reloj.
  - Ir a modo configuración de alarma.
  - Dar 8 tics al botón b1.
  - Dar 2 tics al botón b2.
  - Encender/Apagar alarma.
  - Cambiar a modo ver hora.
- **TbConfHora:** Este banco de pruebas cambia a modo configuración de hora y establece una nueva hora.
  - Cambiar a modo configuración de hora.
  - Dar 8 tics al botón b1.
  - Dar 2 tics al botón b2.
  - Cambiar a modo ver hora.

## Capturas de pantalla.

En esta sección vamos a adjuntar algunas capturas de pantalla para cada uno de los casos de prueba. De esta forma verificaremos el correcto funcionamiento del sistema. Para el modo de prueba se ha supuesto que las horas tienen 3 minutos y los minutos dos segundos.

- **TbModoHora:** en la siguiente captura podemos observar como se conmuta del modo 24h al modo 12h a las doce de la noche.
  - Archivo: capturas/tbModoHora.png
- **TbCrono:** en la siguiente captura podemos observar como ponemos el reloj en modo cronómetro, reiniciamos el mismo y lo ponemos en marcha. Se puede observar que mientras estamos en otro modo el cronómetro sigue con su cuenta.
  - Archivo: capturas/tbCrono1.png
  - Archivo: capturas/tbCrono2.png
  - Archivo: capturas/tbCrono3.png
- **TbAlarma:** en la siguiente captura se observa como se cambia a modo hora, se configura la alarma, se enciende la misma y se espera a que suene.
  - Archivo: capturas/tbAlarma1.png
  - Archivo: capturas/tbAlarma2.png
  - Archivo: capturas/tbAlarma3.png.
- **TbConfHora:** en la siguiente captura se observa como se conmuta a modo configuración de hora, se establece una nueva hora. y se vuelve al modo de visualización de hora.
  - Archivo: capturas/tbConfHora1.png
  - Archivo: capturas/tbConfHora2.png
  - Archivo: capturas/tbConfHora3.png.
  - Archivo: capturas/tbConfHora4.png.