

IMPLEMENTACIÓN DE ENTRELAZADOR EN EL DOMINIO DEL TIEMPO PARA MODULADOR DTMB.

Alejandro C. González Urquiza¹, Manuel Navarrete Hernández², Reinier Díaz Hernández³

LACETEL, Instituto de Investigación y Desarrollo de Telecomunicaciones, Ave. Independencia, Km. 14½, No. 34818, Boyeros, La Habana.

¹e-mail: alejandroc@lacetel.cu

²e-mail: manuel@lacetel.cu

³e-mail: reinier@lacetel.cu

RESUMEN

En *LACETEL*, Instituto de Investigación y Desarrollo de Telecomunicaciones, se han diseñado diferentes bloques de la cadena de transmisión-recepción de televisión digital terrestre del estándar DTMB. En este trabajo se describe el diseño y la implementación del entrelazador en el dominio del tiempo para un modulador del estándar DTMB. Este bloque mejora el rendimiento del sistema de televisión digital, ya que contribuye a evitar el efecto desfavorable de las ráfagas de errores a la entrada del decodificador de canal. El diseño se describió en lenguaje de descripción de hardware VHDL, utilizando el entorno de desarrollo ISE WebPACK Design y se implementó en el FPGA Spartan-6 LX9 de Xilinx, empleando memoria externa. El entrelazador se estructuró de forma modular, dividido en bloques funcionales. Estos bloques se validaron de forma independiente mediante simulaciones en ISim y verificaciones en hardware con ChipScope. Se utilizó como modelo de referencia el bloque de entrelazador en el dominio del tiempo de Simulink. Este modelo fue utilizado en conjunto con ChipScope para depurar y validar el entrelazador implementado.

PALABRAS CLAVES: Entrelazador, modulador, televisión digital terrestre.

ABSTRACT

LACETEL, Research and Development Telecommunications Institute, has designed different blocks of transmission-reception chain of digital terrestrial television of DTMB standard. This paper describes the design and implementation of time interleaver for DTMB standard modulator. This block enhances the performance of digital television system, because it contributes to avoid adverse effects of bursts error at the channel decoder input. The design was described in hardware description language VHDL, using ISE WebPACK Design development environment and was implemented in Xilinx Spartan-6 LX9 FPGA using external memory. The interleaver was modularly structured, divided into functional blocks. The blocks were independently validated by simulations in ISim and were checked in hardware with ChipScope. Simulink time interleaver block was used as reference model. This model was used in conjunction with ChipScope to debug and validate the implemented interleaver.

KEYWORDS: Interleaver, modulator, digital terrestrial television.

INTRODUCCIÓN

La televisión digital ha revolucionado el concepto que hasta ahora se tiene de la televisión, ya que ha supuesto un cambio significativo tanto en el ámbito tecnológico como en lo que respecta a la producción de programas y servicios que se ofrecen al espectador.

La transición de la televisión analógica a la digital es un cambio inevitable que cada país tiene que enfrentar y que se enmarca en el proceso de digitalización que ya ha tenido lugar en muchas tecnologías, como la telefonía fija y móvil, el almacenamiento y la transmisión de datos, Internet, la prensa, el cine, las transmisiones satelitales, y las comunicaciones en general.

Para determinar la norma más conveniente para Cuba, fue necesario tomar en cuenta numerosos elementos de índole político, económico y tecnológico. Después de varios años de estudio, análisis y evaluación de la información técnica y de pruebas de campo realizadas se determinó la tecnología china como la más ventajosa para Cuba. Finalmente, en junio de 2011 se hace oficial la adopción de la norma de televisión digital terrestre DTMB (*Digital Terrestrial Multimedia Broadcasting*) y del estándar de compresión digital de audio y video AVS (*Audio Video Standard*), ambos desarrollados por la República Popular China.

El Instituto de Investigación y Desarrollo de Telecomunicaciones (**LACETEL**), ha sido asignado como el centro, que dentro de este proyecto de despliegue de la televisión digital, está concentrado en la asimilación y absorción de esta tecnología para su posterior introducción, difusión e innovación en Cuba; así como desarrollar soluciones propias que permitan alcanzar la independencia tecnológica.

Con este fin, en **LACETEL** se han desarrollado diferentes bloques de la cadena de transmisión-recepción de este estándar. Como resultado, se cuenta con modelos funcionales, pero no implementables en FPGAs (*Field Programmable Gate Array*) de entrelazador en el dominio del tiempo para un modulador acorde al estándar DTMB, debido a que el entrelazador exige grandes demandas de memoria y que los FPGAs disponibles tienen recursos limitados.

Este trabajo se limitará a diseñar e implementar los subsistemas que componen el entrelazador que pueden ser implementadas en el FPGA Spartan-6 LX9 de Xilinx predeterminado (ver figura 1).

El estándar DTMB incluye dos modos de entrelazado de tiempo. Para reducir la complejidad del entrelazador, la implementación es, por tanto, concebida para un modo particular: B = 52, M = 240.

TRANSMISOR DEL ESTÁNDAR DTMB

El sistema DTMB se utiliza para convertir el flujo de datos de entrada MPEG - TS (*Moving Picture Experts Group – Transport Stream*) a una señal de salida de RF (radiofrecuencia). El siguiente procesamiento en banda base se aplica al flujo de datos de entrada:

- Aleatorización
- FEC (*Forward Error Correction*)
- Mapeo QAM (*Quadrature Amplitude Modulation*)
- Entrelazado
- Multiplexación de bloque de datos básicos e información de sistema.
- Combinación del Cuerpo de la Trama y de la Cabecera de la Trama formando la Trama Señal.

Después de este procesamiento en banda base, el flujo de datos de entrada es convertido a señales de RF dentro de un ancho de banda de 6/7/8 MHz en la banda de UHF (*Ultra High Frequency*) o VHF (*Very High Frequency*).

El diagrama del sistema de transmisión se muestra en la figura 1. [1]

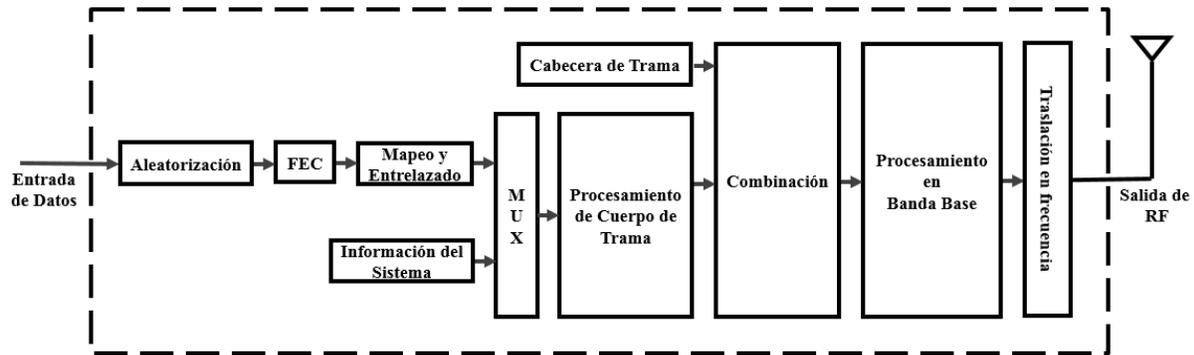


Figura 1: Diagrama del sistema de transmisión del estándar DTMB

ENTRELAZADOR EN EL DOMINIO DEL TIEMPO

Aunque la codificación del bloque FEC ofrece una protección robusta frente a errores y puede manejar errores múltiples en cada palabra de datos, hay un límite en el número de fallas que un código de corrección puede corregir y detectar, dados ciertos parámetros de código. En algunos medios de transmisión, como es el caso de la transmisión terrestre, la señal recibida se ve afectada por desvanecimientos profundos, efectos multi-trayecto severos, etcétera. En estas condiciones, la capacidad correctora del decodificador de canal se ve significativamente reducida, ya que la recepción de una serie consecutiva de bits erróneos dentro de una misma palabra no permite aprovechar la redundancia para corregir los errores si estos superan la capacidad correctora del código. [2]

La técnica de entrelazado de tiempo pretende evitar el efecto desfavorable de las ráfagas de errores a la entrada del decodificador de canal. Así se consigue explotar al máximo la capacidad correctora del código y por tanto se mejora la calidad de la transmisión digital sobre el canal de radio. Esto se logra alterando el orden de símbolos sucesivos después de ser codificados en diferentes intervalos de tiempo, de forma tal que los símbolos no se transmiten en su secuencia original, sino que se esparcen entre muchas tramas diferentes.

En otras palabras, el proceso de entrelazado esparce los errores a través de diferentes tramas y dentro de las tramas y reduce el número de errores dentro de las palabras de código debido a fallas del canal, ya que el número de errores dentro de una palabra de código recibida se reduce por debajo del umbral dado por los parámetros del código. [1,2]

El receptor conoce la secuencia de alteración y recupera el orden original antes de la decodificación. Cuando el receptor entrega los bits demodulados, estos pasan por el bloque de desentrelazado y a

continuación los bits pasan al bloque de decodificación de canal, donde se procede a intentar corregir los errores. [1]

Sin embargo, el entrelazado mejora el rendimiento de los sistemas de radio digitales a costa de aumentar los requisitos de espacio de memoria, la complejidad del sistema, y el tiempo de retardo. El aumento de espacio de memoria es normalmente tolerable. La cuestión de complejidad es relativa y siempre está disminuyendo debido a los avances en la tecnología. Sin embargo, el aumento de tiempo de retraso puede hacer que el entrelazado sea poco práctico en ciertas aplicaciones (por ejemplo, comunicaciones de voz).

El entrelazado en el dominio del tiempo se realiza de acuerdo con el algoritmo de entrelazado convolucional, por lo que en la próxima sección la atención se centra en el mismo, su realización, ventajas y rendimiento. [1]

ENTRELAZADOR CONVOLUCIONAL

El entrelazado en el dominio del tiempo se realiza por un algoritmo conocido como entrelazado convolucional, tal y como se muestra en la figura 2.

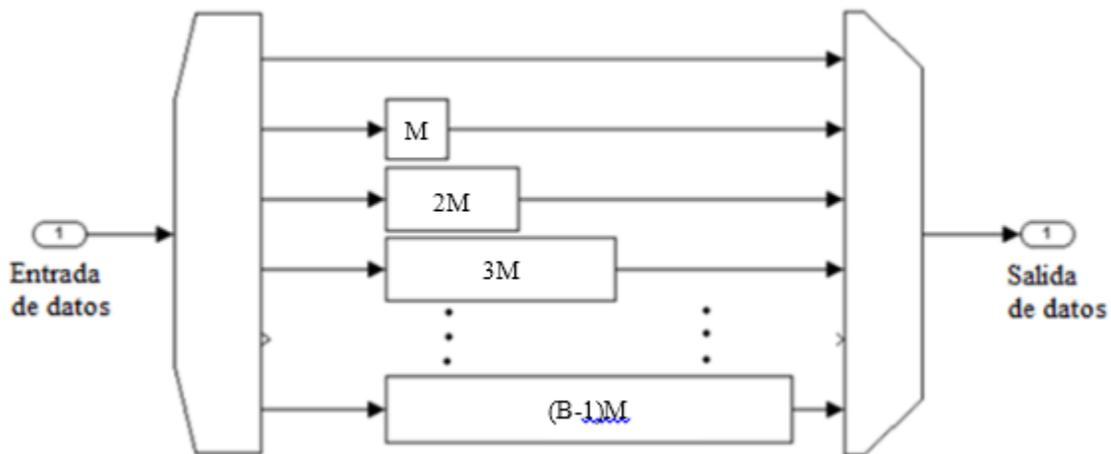


Figura 2: Realización del entrelazador convolucional

Cada rama es un registro de desplazamiento con un retardo de $n \cdot M$ donde $n = 0, 1, 2, \dots, B-1$ que corresponde con la rama seleccionada. B es el número de ramas y M es el largo del paso de retardo.

En primer lugar, un valor es escrito en la entrada del registro de desplazamiento de la rama actual, luego uno es leído de la salida. Cuando se llevan a cabo la escritura y lectura, se selecciona la siguiente rama. Esto se realiza continuamente para cada símbolo, de forma que el entrelazador introduce una latencia en el sistema (mayor que otros bloques del modulador), correspondiente a la profundidad de entrelazado. La profundidad de entrelazado es la cantidad de tramas entre las cuales los símbolos se esparcen. [2]

En el estándar DTMB, el parámetro B es igual a 52 y M puede ser igual a 240 ó a 720. [1]

Requerimientos de memoria

La demanda de memoria de los modos del entrelazador se calcula según la ecuación (1).

$$\sum_{k=0}^{51} k(M + 1) = 1326(M) \text{ símbolos} \quad (1)$$

Por lo que, para el modo $M = 240$ y 8 bits por símbolo, la memoria total consumida se calcula según la ecuación (2).

$$\text{mem}(M = 240) = 1326(M + 1) * 8 \approx 2.44 \text{Mbits} \quad (2)$$

SELECCIÓN DE HARDWARE Y HERRAMIENTAS

En esta sección se presentan el hardware y las herramientas utilizadas en la implementación del entrelazador.

Hardware

Los FPGAs son una opción viable para esta aplicación, de tratamiento de señales, debido a su alta frecuencia de trabajo, a su capacidad de procesamiento en paralelo, y a su bajo precio en comparación con los Circuitos Integrados de Aplicación Específica (ASICs, por sus siglas en inglés) cuando su producción es de bajo volumen, como es el caso de este tipo de diseño, el cual tiene una aplicación comercial muy particular.

El FPGA utilizado para la implementación fue el Spartan-6 LX9 de Xilinx (XC6SLX9-2CSG324) y el hardware usado para las pruebas y verificación fue el kit de desarrollo S6LX9 de Avnet, por su disponibilidad en **LACETEL**. Para configurar el FPGA, la lógica fue descrita en VHDL (*VHSIC Hardware Description Language*, a su vez VHSIC es el acrónimo de *Very High Speed Integrated Circuit*). [3,4]

Herramientas

Como líder del sector Xilinx ofrece soluciones integrales que consisten en, aparte de dispositivos FPGA, software avanzado y configurable, núcleos de Propiedad Intelectual (IP, por sus siglas en inglés) listos para uso de mercado y para aplicaciones tales como la que se implementa en este trabajo.

Se utilizó el entorno de desarrollo ISE WebPACK Design Software de Xilinx para generar y sintetizar el código VHDL. También se emplearon otras herramientas de software para diseños avanzados, tales como Xilinx CORE Generator, ISim y ChipScope, entre otras, las cuales están integradas dentro del entorno de desarrollo ISE y contribuyen a agilizar el proceso y a mejorar la calidad del diseño. [5]

Se utilizó Simulink como herramienta de simulación de sistemas para crear un modelo de referencia del entrelazador en el dominio del tiempo.

MODELO DE REFERENCIA DEL SISTEMA

El propósito del modelo de referencia es verificar la función y medir el rendimiento del entrelazador implementado. El modelo también da una estimación de la complejidad del sistema.

El modelo de referencia del sistema constituye un bloque independiente y se implementó a partir del bloque de entrelazado en el dominio del tiempo de Simulink. Este modelo fue utilizado en conjunción con ChipScope para depurar y validar el entrelazador implementado.

CONSIDERACIONES PREVIAS A LA IMPLEMENTACIÓN

En general, el diseñador debe seguir ciertas pautas para que el modelo pueda sintetizarse. Esto es también importante para lograr implementaciones óptimas del sistema que se está diseñando.

No todas las construcciones posibles en VHDL pueden sintetizarse y transformarse en circuitos. Esto puede deberse a las limitaciones de las herramientas utilizadas o a que el circuito descrito por el modelo VHDL no puede implementarse físicamente.

Implementación usando registros de desplazamiento

Debido a las muy grandes demandas de memoria del entrelazador (2.44 Mbits), las exigencias de RAM (*Random-Access Memory*) distribuida para implementar los registros de desplazamiento son muy grandes en comparación con los recursos con que cuentan los FPGAs disponibles (el FPGA destinado tiene 90 kbits de RAM distribuida). Por lo tanto, no es posible implementar el entrelazador usando registros de desplazamiento. Sin embargo, los registros de desplazamiento se pueden simular usando memoria. [6,7]

Implementación usando memoria

Igualmente, debido a las muy grandes demandas de memoria del entrelazador, las exigencias de RAM dedicadas para implementar los registros de desplazamiento usando memoria son muy grandes en comparación con los recursos con que cuentan los FPGAs disponibles (el FPGA destinado tiene 576 kbits de bloques de RAM dedicado). Por lo tanto, el sistema se implementó usando memoria externa, como es común en prácticas con plataformas hardware basadas en FPGAs. Esto hace posible utilizar el controlador de memoria pre-construido en el FPGA. Existen dos controladores de memoria cableados dentro del FPGA que se pueden utilizar en la plataforma de hardware previsto. [6]

INTERFAZ DE TRANSFERENCIA DE DATOS

Para permitir el desarrollo independiente de los diferentes subsistemas que componen el modulador, se especifica una interfaz de transferencia de datos que se utiliza entre los diferentes subsistemas. La interfaz consta de dos señales de estado y un puerto de transferencia de datos tanto en la entrada como en la salida. La interfaz de datos se ilustra en la figura 3.

La señal **data_valid_i** es activa siempre y cuando existe un dato válido en el puerto de entrada. El sistema activa **busy_o** cada ciclo de reloj durante el cual se procesa el dato, indicando que está ocupado, que no le pueden enviar otro dato hasta tanto no haya terminado con el presente. El sistema activa **busy_o** también cuando el subsistema siguiente activa **busy_i**, indicando, este último, que está ocupado.

Esta interfaz facilita diferentes velocidades de datos en diferentes módulos y permite que las señales de interfaz sean creadas a partir de, por ejemplo, banderas vacías o llenas de bloques de FIFOs (*First In, First Out*) pre-construidos.

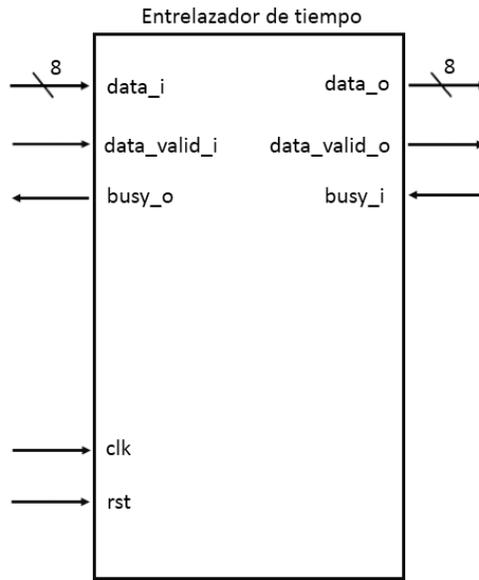


Figura 3: Interfaz de transferencia de datos del entrelazador en el dominio del tiempo.

IMPLEMENTACIÓN DEL SISTEMA

El sistema se implementó sistemáticamente como una serie de subsistemas interconectados, para lo cual se dividió en un conjunto de bloques funcionales, comunes para todo sistema digital que hace uso intensivo de transferencias de datos con la memoria. La figura 4 muestra el diagrama en bloques del sistema.

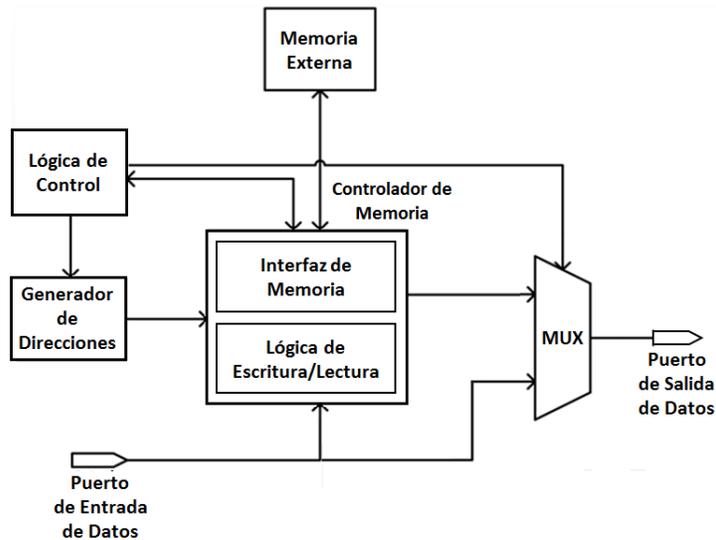


Figura 4: Diagrama en bloques del entrelazador en el dominio del tiempo

Controlador de Memoria

El Controlador de Memoria controla el flujo de datos que va y viene desde la memoria. Siguiendo el mismo esquema de trabajo, se dividió en dos bloques funcionales:

- Interfaz de Memoria: está basada en el Bloque Controlador de Memoria (MCB, por sus siglas en inglés) dentro del FPGA, el cual simplifica enormemente la tarea del dispositivo Spartan-6 LX9 de interactuar con la memoria y se encarga de refrescarla automáticamente, evitando que se pierdan los datos escritos en el dispositivo. [8]
- Lógica de Escritura / Lectura: permite la ejecución de las transacciones de escritura y de lectura con la memoria.

Generador de direcciones

Para implementar registros de desplazamiento usando RAM, dos direcciones son necesarias por cada símbolo: una dirección de escritura y una de lectura. Un segmento de memoria está asociado a las direcciones, con un límite inferior y uno superior. El número de filas en cada segmento de memoria es el tamaño de cada registro de desplazamiento más uno respectivamente.

Las direcciones de escritura y de lectura siempre están asociadas a espacios de datos consecutivos dentro del segmento de memoria. Ambas direcciones están enmarcados por el límite inferior y el límite superior del segmento de memoria. Si una alcanza el límite superior, entonces es *reseteada* al límite inferior.

La escritura y la lectura son siempre realizadas en conjunto y no importa el orden en que son realizadas con esta propuesta. Cuando un ciclo de escritura-lectura ha sido realizado, el siguiente segmento es seleccionado.

Estas direcciones son generadas por el Generador de Direcciones. Como estas son pasadas al Controlador de Memoria, el cual realiza la escritura y la lectura a la memoria externa, deben respetar el esquema de direccionamiento de la memoria. El tamaño del puerto usado en este diseño es de 32 bits, lo cual impone el requerimiento de que cada dirección de fila es un múltiplo de 4. [8]

Lógica de control

La Lógica de Control controla las operaciones del sistema. Esta genera las señales de control necesarias y establece el mecanismo de secuenciación de estas señales, para asegurar que las operaciones sucedan en el momento adecuado y en el orden correcto. Este bloque se implementó de forma tal que hasta que una escritura a la memoria no ha sido completada no se efectuará una lectura.

VERIFICACIONES

Hay varias razones para probar un circuito lógico. Cuando un circuito se desarrolla por primera vez, es necesario verificar que el circuito diseñado se ajusta a las especificaciones funcionales y de tiempo requeridas.

Verificación de los subsistemas

El sistema digital que se describe en este trabajo es relativamente grande, por lo que fue diseñado como un conjunto de subsistemas más pequeños. Para ahorrar tiempo, estos subsistemas se aislaron y se verificaron a través de simulaciones en ISim y verificaciones en hardware con ChipScope de forma independiente mediante pequeños circuitos para pruebas diseñados específicamente para este propósito.

Validación del sistema

El modelo de referencia de entrelazador convolucional utilizado para validar el diseño fue un modelo de Simulink. Este modelo fue utilizado en conjunto con ChipScope para depurar y verificar el entrelazador implementado.

La misma secuencia de entrada fue aplicada tanto en la implementación en VHDL, como en el modelo de Simulink. Entonces se extrajeron secciones de las secuencias de salida del modelo y de la implementación y se compararon en Excel. Los resultados coincidieron, indicando que la secuencia correcta se obtuvo en la salida de la implementación.

Una simulación en ISim del entrelazador convolucional no se realizó debido a tres razones. En primer lugar, la simulación tomaría demasiado tiempo para completarse, ya que la profundidad del entrelazado es de 170 tramas de 3780 símbolos cada una. En segundo lugar, no existía ningún modelo de simulación para la memoria en VHDL. Por último, para verificar completamente el controlador de memoria, se consideró mejor probarlo en hardware para confirmar su funcionalidad. [2]

En la figura 5 se muestran las señales capturadas con el ChipScope, las cuales demuestran el correcto funcionamiento del entrelazador implementado.

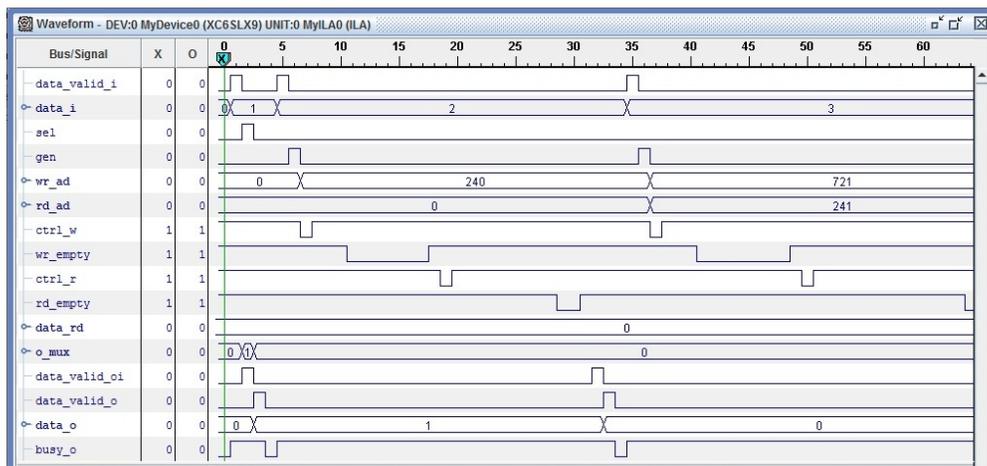


Figura 5: Mecanismo de secuenciación de las principales señales internas y de E/S del entrelazador en el dominio del tiempo

RECURSOS UTILIZADOS DEL FPGA

En la tabla 1 se muestran los recursos utilizados del FPGA predeterminado. Estos también se especifican como un porcentaje de los recursos disponibles del dispositivo para permitir una mejor comprensión de qué parte de la capacidad del FPGA es empleada. Como se puede apreciar en la tabla 1, el entrelazador se puede implementar físicamente en el FPGA.

Tabla 1: Recursos de hardware utilizados en el FPGA por el entrelazador en el dominio del tiempo

	Slices	Flip flops	LUTs	Slices DSP48A1	Bloques de RAM de 18kb	MCBs
Utilizada	944	1,675	1,913	0	0	1
Disponibles	1,430	11,440	5,720	16	32	2
Porcentaje de Utilización	66%	14%	33%	0%	0%	50%

CONCLUSIONES.

Se determinó como método factible para la implementación práctica del entrelazador en el dominio del tiempo, la simulación de los registros de desplazamiento mediante memoria externa.

Se logró implementar el entrelazador en el dominio del tiempo para un modulador del estándar DTMB usando los recursos internos del FPGA Spartan 6 LX9 de Xilinx y la memoria RAM externa del kit de desarrollo S6LX9 de Avnet.

Se verificó cada subsistema que compone el entrelazador de forma independiente a través de simulaciones en ISim y verificaciones en hardware con ChipScope.

Se utilizó como modelo de referencia un bloque de Simulink en conjunto con ChipScope para depurar y validar el entrelazador implementado.

REFERENCIAS.

1. *"Framing Structure, Channel Coding and Modulation For Digital Television Terrestrial Broadcasting System"*. 2006. p. 132.
2. ABRAHAMSSON, Sebastian, RÅBE, Markus. *An FPGA implementation of a modulator for digital terrestrial television according to the DTMB standard, Department of Electrical Engineering, 2010, Linköpings University: Linköping. p. 99.*
3. PARNELL, K; BRYNER, R. *"Comparing and Contrasting FPGA and Microprocessor System Design and Development"*. 2004. p. 32.
4. *FPGA vs. ASIC*. 2004. p. 15
5. *Xilinx Design Tools: Release Notes Guide. Vivado Design Suite and ISE Design Suite Design Suite*, Xilinx, 2012. p. 50
6. *Spartan-6 Family Overview*, Xilinx, 2011. p. 11
7. REY DOMÍNGUEZ, Addis; RAYMOND RODRIGUEZ, Luis Giraldo. *"Diseño del Aleatorizador, la Codificación de Canal, Mapeo y Entrelazado de un modulador DTMB"*. Dpto. de Telecomunicaciones y Telemática, Instituto Superior Politécnico José A. Echeverría.: La Habana, 2011.
8. *Spartan-6 FPGA Memory Controller. User Guide*, Xilinx, 2010. p. 66