

INGENIERÍA INVERSA AL BLOQUE IFFT DE UN MODULADOR CONFORME A LA NORMA DTMB

Ing. Edith Annette Cabrera Hernández¹, Ing. Anaima Robert Nápoles², Ing. Yosmany Hernández Sánchez³, Ing. Ernesto Fontes Pupo⁴

ISPJAE, calle 114 No. 11901 e/ Ciclovía y Rotonda Marianao, La Habana, Cuba

^{1,2}dide0703@gmail.com, ³yosmany@lacetel.cu
⁴e-mail: fontes@lacetel.cu

RESUMEN

Como parte del despliegue de la Televisión Digital, en Cuba se ha adoptado el estándar chino DTMB (por sus siglas en inglés, Digital Television Terrestrial Multimedia Broadcasting), basado en la utilización de la modulación TDS-OFDM (por sus siglas en inglés, Time Domain Synchronous – Orthogonal Frequency Division Multiplexing) implementada mediante una IFFT (por sus siglas en inglés, Inverse Fast Fourier Transform) de 3780 portadoras. Esta investigación se enfoca en realizar un proceso de ingeniería inversa al bloque IFFT para describir su estructura y funcionamiento. Se cuenta con un diseño implementado en un FPGA (por sus siglas en inglés, Field Programmable Gate Array) de la familia Cyclone IV del fabricante Altera, desarrollado en el lenguaje de descripción de hardware Verilog. Como parte de los resultados se obtienen diagramas en bloque y máquinas de estado que permiten representar el comportamiento del módulo IFFT. Para la obtención y comprobación de algunos de los resultados se realizan simulaciones empleando el software ModelSim. Se arriba a conclusiones sobre la arquitectura del bloque, que podrán ser empleadas como base para futuras investigaciones.

PALABRAS CLAVES: Ingeniería inversa, modulación TDS-OFDM, Verilog, DTMB, IFFT.

ABSTRACT

As part of the Digital Television deployment, in Cuba has been adopted the Chinese DTT Standard DTMB. This standard is based on the use of a TDS-OFDM modulation scheme, implemented through an IFFT of 3780 carriers during transmission. This investigation is focused on the reverse engineering process performed to the IFFT block in order to describe its structure and working principles. Is available a practical design, targeting a Cyclone IV FPGA device from Altera's company, developed using Verilog Hardware Description Language. Among the obtained results are block diagrams and state machines representing the layout and behavior of the IFFT block. To obtain and verify these results were performed functional simulations with ModelSim software. Finally, are provided conclusions about the architecture of this block, useful as the basis for future research.

KEYWORDS: Reverse engineering, TDS-OFDM modulation, Verilog, DTMB, IFFT.

INTRODUCCIÓN

El estándar DTMB ha sido adoptado en Cuba para el despliegue de la Televisión Digital Terrestre (TDT). Este estándar emplea una modulación basada en la Multiplexación Por División de Frecuencias Ortogonales (OFDM), específicamente emplea modulación TDS-OFDM[1].

La idea básica de OFDM consiste en dividir la información a transmitir en varias subportadoras espaciadas en la frecuencia y ortogonales entre sí. Dos señales se considerarán ortogonales cuando el resultado del producto punto entre ellas sea cero y por tanto estas señales serán perpendiculares entre sí. Al ser ortogonales las subportadoras, no es necesaria la banda guarda requerida en los sistemas FDM; mientras se mantenga la condición de ortogonalidad pueden solaparse los espectros de las subportadoras y aún es posible recuperar la información contenida en cada una[2].

La correlación entre dichas subportadoras será cero dado a su ortogonalidad. El proceso de modulación puede analizarse, entonces, como la correlación entre la señal de entrada y cada una de las subportadoras ortogonales no correlacionadas. Por lo tanto la correlación realizada entre la señal y una subportadora determinada sólo será una medida de la energía para dicha portadora, no existirá contribución energética de las demás subportadoras puesto que no están correlacionadas entre ellas [2].

Entonces si la señal de entrada tiene alguna componente espectral a determinada frecuencia, existirá un pico en la correlación de la señal de entrada y la réplica de la subportadora que corresponde a esa frecuencia. Debido a este método de separación de la energía individual de cada componente de la señal es posible que los espectros de las subportadoras OFDM se solapen sin producirse interferencia [2].

En la figura 1 se observan cuatro subportadoras del espectro de una señal OFDM. Se evidencia cómo es posible recuperar la información contenida en cada una de las subportadoras por separado, aun cuando existe solapamiento entre sus componentes espectrales, pues, todos los cruces por cero corresponden a picos de las subportadoras continuas, o sea, en el punto de máxima amplitud de cada una de estas no existe aporte de energía de las otras[2].

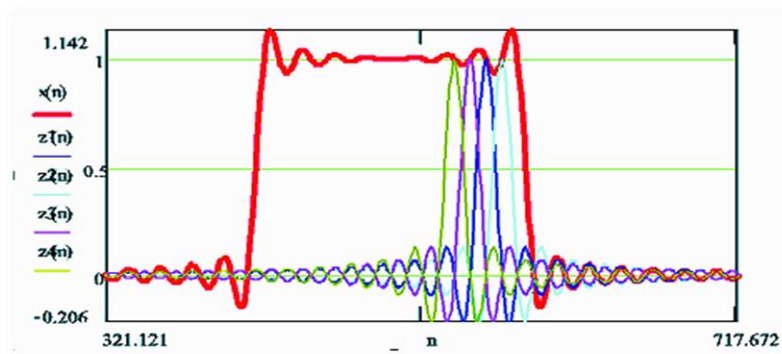


Figura 1: Espectro de una señal OFDM mostrando 4 subportadoras.

IMPLEMENTACIÓN DE LA MODULACIÓN OFDM MEDIANTE DFT.

Teóricamente puede implementarse un generador OFDM a partir de un número N de sintetizadores (equivalente al número de muestras a procesar) y una lógica de control. Para los sistemas de TDT esto es extremadamente difícil pues los valores de N son elevados variando en un rango de 2K, 4K y 8K. Además lograr que las frecuencias de cada sintetizador sean ortogonales entre sí no es sólo complejo, sino que los costos para su implementación ascenderían a valores considerables[3].

La esencia se encuentra en lograr que incluso para valores elevados de N se cumpla la ortogonalidad. Para ello es necesario que cada subportadora tenga una frecuencia diferente y que se cumpla que durante un período de símbolo ocurra un número de ciclos enteros para cada subportadora. Esto se ilustra en la figura 2.

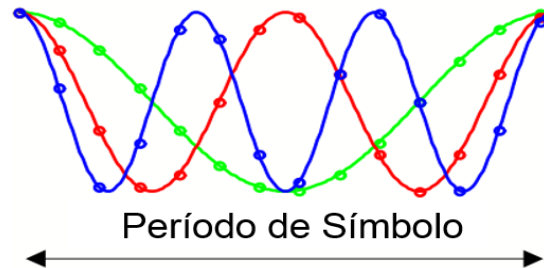


Figura 2: Números enteros de ciclo durante un período de símbolo para tres subportadoras.

La Transformada Discreta de Fourier o DFT (por sus siglas en inglés, Discrete Fourier Transform) plantea que sus sinusoides forman un conjunto de bases ortogonales que cumplen las condiciones anteriores lo que justifica su empleo para implementar la modulación OFDM.

El proceso de cálculo empleando DFT involucra un elevado número de operaciones aproximadamente de N^2 , donde N sería el número de subportadoras empleadas, para el caso de DTMB serían 3 780, lo que

arrojaría un aproximado de 14 millones de operaciones requeridas. En la práctica se emplean algoritmos más eficientes para realizar estos cálculos conocidos como Transformada Rápida de Fourier o FFT (por sus siglas en inglés, Fast Fourier Transform). Estos métodos permiten reducir las operaciones requeridas a $N\log(N)$ que para las 3780 subportados de DTMB serían aproximadamente 13 mil operaciones.

Tanto DFT como FFT tienen la característica de convertir expresiones que describen señales en el dominio del tiempo en sus equivalentes en el dominio de la frecuencia. Sin embargo para realizar la operación contraria y llevar señales del dominio de la frecuencia al del tiempo se utilizarán sus respectivas operaciones inversas IDFT (por sus siglas en inglés InverseDiscrete Fourier Transform) e IFFT (por sus siglas en inglés InverseFast Fourier Transform).

En los sistemas de TDT es empleada la IFFT en la etapa de modulación del transmisor y la FFT en la demodulación en el receptor de manera que antes de la etapa de modulación y posterior a la

demodulación la señal se encuentra en el dominio de la frecuencia y en los procesos comprendidos luego de la modulación y antes de la demodulación, la señal se encuentra en el dominio del tiempo. Esto se ilustra en la figura 3.

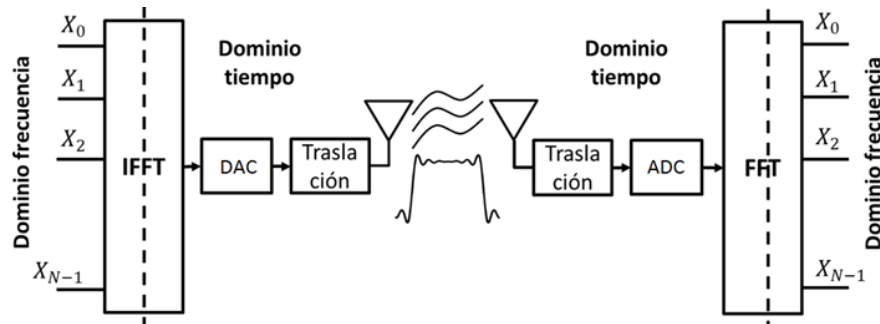


Figura 3: Esquema de modulación y demodulación OFDM

FUNDAMENTOS MATEMÁTICOS DE LA FFT Y SU INVERSA.

Descomponer un problema complejo en problemas más pequeños es una práctica común; la idea detrás de la FFT es la de “divide y vencerás”, para lograr el mismo resultado con un número reducido de operaciones (recursos).

Una forma de simplificar el cálculo de la DFT es realizando un cambio o reordenamiento en el índice de la secuencia de entrada, para transformar el problema original de una dimensión en un problema multidimensional y lograr una implementación más eficiente.

La idea que permite esta optimización es la descomposición de la transformada a tratar en otras más simples de forma iterativa hasta llegar a transformadas de pocos elementos. Una vez resueltas las transformadas más simples hay que agruparlas en otras de nivel superior que deben resolverse de nuevo y así sucesivamente hasta llegar al nivel más alto. Al final de este proceso, los resultados obtenidos deben reordenarse.

Los algoritmos FFT se basan en transformar la ecuación (1) de la DFT en una representación multidimensional, descomponiendo el tamaño N en i dimensiones, a través de factorización. En general este problema suele verse como de dos factores que se descomponen iterativamente, para lograr más de dos dimensiones. De manera que puede expresarse el tamaño de la transformada como $N=N_1 \cdot N_2$ [4].

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j2\pi n k \frac{1}{N}}, \quad k = 0; 1; \dots (N-1) \quad (1)$$

En (1) N , n y k son enteros y $j = \sqrt{-1}$. Nótese que la definición de la transformada lleva implícito el tratamiento de señales complejas, con componentes I (In-Phase) y Q (Quadrature).

Dado que la transformada discreta de Fourier inversa es análoga a la transformada discreta de Fourier, con distinto signo en el exponente y un factor $1/N$ como se observa en (2), cualquier algoritmo FFT puede ser fácilmente adaptado para el cálculo de la transformada inversa[4].

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi nk \frac{1}{N}}, \quad n = 0; 1; \dots (N-1) \quad (2)$$

Es común que las transformadas, tanto directa como inversa, queden expresadas en función del factor “twiddle” W , definido como $W = e^{-j(2\pi)/N}$, donde $e^{jx} = \cos x + j \sin x$ siendo la parte real I y la imaginaria Q . Este factor es generado como una matriz que contiene las bases sinusoidales ortogonales. W es de longitud N por lo que se puede expresar como W_N . [3]

Un elemento de necesaria consideración en los algoritmos de FFT multidimensionales es si N_1 y N_2 tienen un factor común (mayor común divisor, *mcd*) mayor que 1 o no (cuando los números son co-primos)[3].

Aquellos algoritmos con *mcd* mayor que 1 son conocidos como Algoritmos de Factor Común (CFA, Common Factor Algorithms) y aquellos con *mcd* igual a 1 son conocidos como Algoritmos de Factor Primo (PFA, Prime Factor Algorithms). Es importante recalcar el hecho de que los de tipo CFA pueden tener cualquier factor, incluyendo los co-primos a los cuales se restringen los PFA, por lo que los CFA son más amplios y extendidos[3].

RESULTADOS OBTENIDOS A PARTIR DE LA REALIZACIÓN DE INGENIERÍA INVERSA AL BLOQUE IFFT.

Análisis de la jerarquía seguida en el diseño apoyado en su esquema general.

El diseño de hardware, en el cual se basa esta investigación, fue implementado y en un FPGA (por sus siglas en inglés, Field Programmable GateArray) de la familia Cyclone IV del fabricante Altera y desarrollado en el lenguaje de descripción de hardware Verilog de forma jerárquica a partir de módulos. El módulo define las interfaces del bloque (entradas o salidas), su comportamiento y estructura interna. La estructura del circuito electrónico se forma realizando instancias de módulos y primitivas dentro de otro módulo e interconectándolas a través de cables o buses. El comportamiento es descrito utilizando construcciones como *Initial*, *Always* y asignaciones continuas[5].

El nivel más alto en la jerarquía lo ostenta *tx_top_new*. Esta es la instancia donde se encuentra configurado el diseño general. Este bloque se conecta directamente a los pines del FPGA y contiene las jerarquías inferiores inmediatas: *asi2spi* y *tx_top*.

En el módulo *asi2spi* ocurre el proceso de conversión de la señal en formato serie ASI (Asynchronous Serial Interface) al formato paralelo SPI (Synchronous Parallel Interface).

Los datos presentes en la entrada ASI arriban como un flujo de transporte conformado según el estándar MPEG-2 (ISO_IEC_13818-1) y son procesados antes de entrar al FPGA por el dispositivo CY7B933, que realiza la recepción y acondicionamiento, la decodificación 8B/10B y la extracción del reloj de la señal (27 MHz). El FPGA incluye una interfaz de entrada SPI que no requiere procesamiento previo y pasa directamente al bloque de modulación.

El bloque conversor *asi2spi* se inserta en el diseño para brindar la posibilidad de recibir datos tanto en formato ASI como SPI. Internamente el bloque modulador incluye un multiplexor que selecciona la entrada deseada.

Por su parte el *tx_topse* corresponde con la porción de la descripción que se encarga de las distintas fases de la modulación. En él encontramos contenidos los módulos U1, U2 y el de IFFT. Estos módulos se encuentran por tanto, en un nivel jerárquico inferior al del *tx_top*, pero entre ellos se encuentran al mismo nivel. La figura 4 representa el esquema general del diseño y la jerarquía establecida en la descripción de hardware.

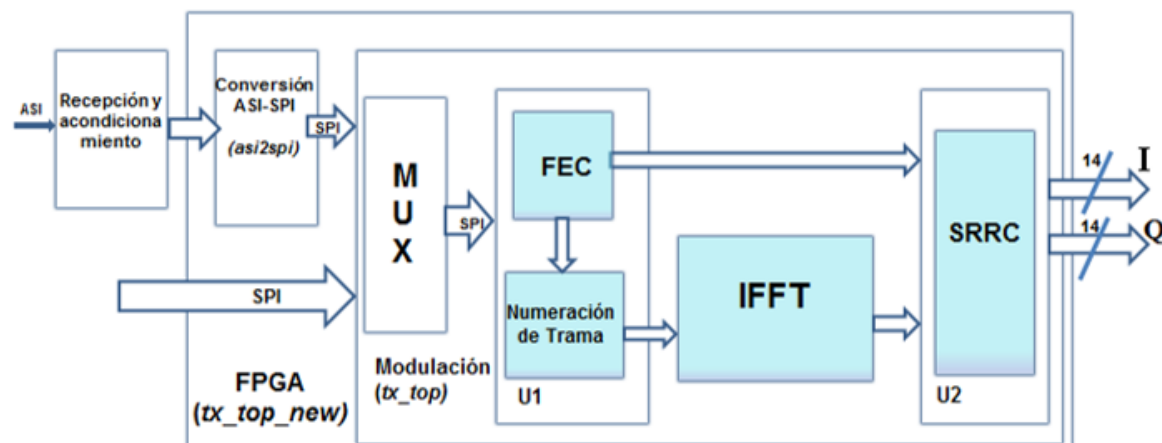


Figura 4: Esquema general del diseño.

El módulo U1 ejecuta las etapas de la adaptación de tasas de transmisión, codificación de canal, entrelazado en tiempo, mapeo de símbolos, aplica corrección de errores e inserción de la información de sistema.

El módulo U2 estructura la trama, inserta la secuencia Pseudoaleatoria como intervalo de guarda, procesa la señal en banda base a través del filtro SRRC (del inglés Square Root Raised Cosine) e inserta las portadoras piloto para conformar el espectro de transmisión.

El bloque IFFT es el objeto central de la investigación y es el encargado de la modulación TDS-OFDM.

Los bloques destacados en azul tanto en U1 como en U2 se corresponden con aquellos que envían y reciben, respectivamente, información (señales) directamente del módulo IFFT.

Etapa de entrada a la IFFT

La etapa de entrada a la IFFT está compuesta por los módulos *fec-top* representado por el bloque FEC (del inglés, Forward Error Correction) de la figura 5, encargado de la corrección de errores y el *frm_num2u2* representado por el bloque Numeración de Trama de la figura 5, este bloque se utiliza para el funcionamiento interno del sistema permitiendo que los bloques posteriores reciban las tramas en correcto orden, en caso de transmisión usando múltiples portadoras. Para portadora única la información proveniente del bloque FEC pasa directamente al bloque U2.

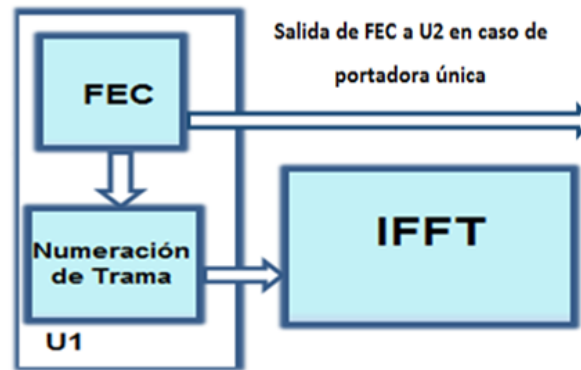


Figura 5: Etapa de entrada a la IFFT.

En el bloque FEC se inserta información adicional que permita la detección y corrección de posibles errores en el receptor. Este método permite la corrección sin retransmisión de la información original por lo que es ampliamente utilizado en sistemas sin retorno o sistemas en tiempo real, donde no se puede esperar a la retransmisión para mostrar los datos, como es el caso de TDT.

Las señales salidas del bloque FEC, que en el caso de utilizar una portadora se envían directo a U2 son: *fec2srrc_l*, *fec2srrc_Q*, *fec2srrc_syn*. Estas se corresponden respectivamente con las señales de datos I, Q y la de sincronismo.

En el caso de utilizar múltiples portadoras se envían al módulo Numeración de Trama las: *frm_l_out*, *frm_Q_out*, *frm_en_out*, *frm_str_out* y *frame_num*. Tanto *frm_l_out* como *frm_Q_out* son de 16 bits, mientras que *frame_num* es de 8 bits y *frm_en_out* y *frm_str_out* son de 1 bit cada una. En el módulo *frm_num2u2* las señales pasan a llamarse: *frm_l_in*, *frm_Q_in*, *frm_en_in*, *frm_str_in* y *frm_num_in* respectivamente. Con ellas se conforma la señal de 42 bits *to_ifft_din* de la siguiente manera *to_ifft_din* = {*frm_num_in*, *frm_en_in*, *frm_str_in*, *frm_l_in*, *frm_Q_in*}.

La señal *to_ifft_din* es pasada como entrada del búfer de datos *ifft_datbuf* del bloque *frm_num2u2*. En dicho búfer se implementa una línea de demora FIFO mediante SRAM cuya función consiste en ajustar las diferencias en el tiempo de procesamiento de etapas anteriores.

Los datos de salida del búfer serán llamados *to_iff_t_dout*. Además se crean los registros *to_iff_t_dout_dly* y *to_iff_t_dout_dly_st*. El registro *to_iff_t_dout_dly* será donde se guarden los datos obtenidos del búfer según la lógica de control representada en la figura 6, la cual es un multiplexor con dos bits de selección conformados por *to_iff_t_dout_dly_st* y *sym_in_en*.

Con *to_iff_t_dout_dly_st* se informa que los datos que se colocan en la entrada del multiplexor provienen del búfer cuando toma valor "1", de lo contrario no hay flujo de información proviniendo del búfer. La señal *sym_in_en* (activa en 1) proveniente del bloque FEC es la habilitación para el envío de símbolos desde *fec-top* hacia *frm_num2u2*. La señal *frame_num_init* se coloca como cabecera de trama con los bits correspondientes a los datos en cero con el fin de alertar de que los paquetes próximos a enviarse si contendrán información. Como se observa, esto último sucede cuando *to_iff_t_dout_dly_st* es cero y *sym_in_en* es uno, lo que implica que aunque los datos no han pasado por el búfer ya han sido enviados al bloque que lo contiene.

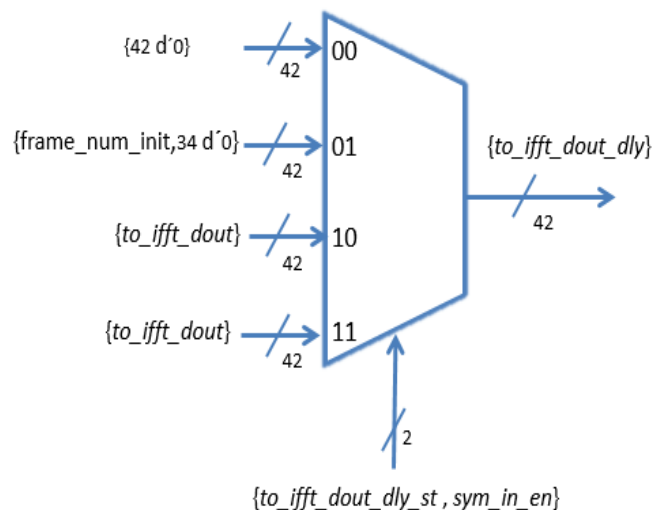


Figura 6: Multiplexor de *to_iff_t_dout_dly*

Los datos de salida de esta lógica, llamados *to_iff_t_dout_dly*, se enviarán a un bloque conformado por dos multiplexores donde la señal se dividirá en I y Q. En el multiplexor de la figura 7 se trabaja con los bits del 31 al 16 donde se almacenan los datos In-Phase (I) de la señal compleja y en el multiplexor de la figura 8 será con los bits del 15 al 0 donde se almacenan los datos In-Quadrature (Q).

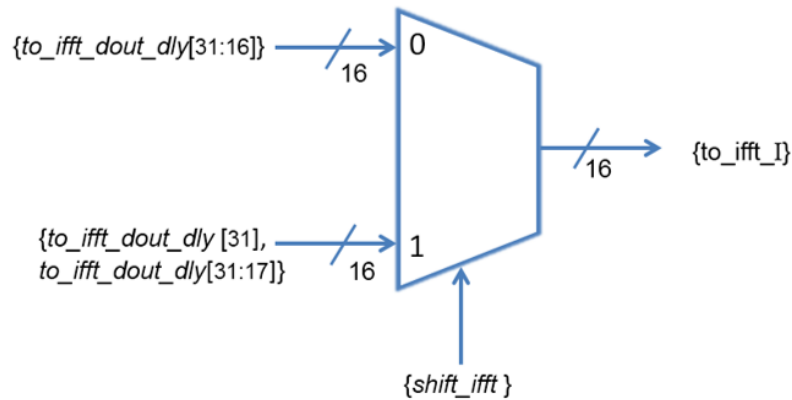


Figura 7: Multiplexor para datos In-Phase.

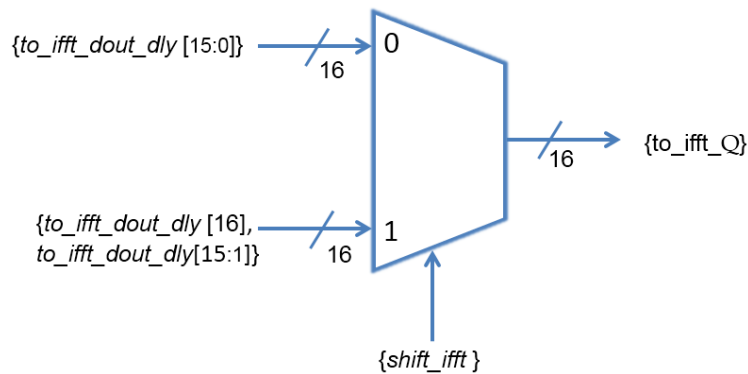


Figura 8: Multiplexor para datos In-Quadrature.

La señal de selección *shift_iftt* es generada por el bloque *reg_top*, donde se generan las señales de control dentro del módulo U1.

Finalmente *to_iftt_I*, *to_iftt_Q* y *to_iftt_sync* se conectan a las entradas *iftt_in_I*, *iftt_in_Q* e *iftt_in_sync* del bloque *iftt_top*.

El bloque *iftt_top* también presenta la entrada *reset_b*, la cual es activa a nivel bajo y su generación puede ser mediante un pin de entrada al FPGA o mediante la escritura a uno de los registros de configuración.

Las señales *clk_7p56* y *clk_30p24* constituyen las entradas de reloj del sistema. Estas poseen una frecuencia de 5.67 MHz y 22.68 MHz respectivamente.

Se incluye una entrada *cnt_clk_ctr* de 2 bits que establece el tipo de red de distribución en que opera el modulador. Para el valor 01 el modulador opera en modo MFN y para el valor 00 lo hace en modo SFN.

Árbol de la IFFT y arquitectura Pipeline.

Para la implementación de la modulación OFDM mediante IFFT primero se determina la descomposición del tamaño de muestras ($N=3780$) en los factores que se continuarán descomponiendo iterativamente. La figura 9 lo esquematiza en una estructura de árbol que facilita su explicación y comprensión.

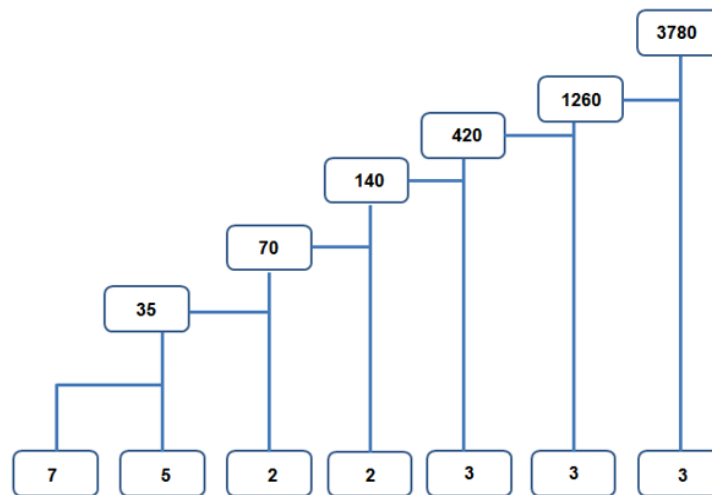


Figura 9: Esquema de la descomposición factorial de la IFFT.

En la base del árbol encontramos la longitud de cada una de las transformadas básicas que se deberán resolver. De manera ascendente se encuentran los valores de las transformadas de nivel superior en cada caso.

Una manera sencilla de verlo sería que en la primera etapa (*stage*) se resuelven 5 transformadas de longitud 7, en la segunda, 7 transformadas de longitud 5 y en su combinación se obtiene una transformada de longitud 35. Con los resultados anteriores se da paso a la tercera etapa donde se combinan 35 transformadas de longitud 2, con 2 de longitud 35 para lograr una de tamaño 70. Resolviendo 2 transformadas de longitud 70 y 70 de longitud 2, se obtiene una de 140 que sería el resultado de la cuarta etapa. En la quinta se resuelven tres de 140 y 140 de tres, y el resultado es una de 420. En la sexta el proceso es similar y se resuelven y combinan 3 de 420 y 420 de 3 para obtener la transformada de longitud 1 260. En la séptima y última etapa se resuelven 1 260 de 3 y 3 de 1 260 para obtener la transformada deseada de longitud 3 780.

Para que este proceso de cálculo ocurra de manera secuencial el bloque IFFT fue diseñado siguiendo una arquitectura Pipeline. Esta se basa en colocar registros equitativamente en diferentes porciones de la descripción de hardware para segmentar el proceso de cálculo y de esta forma reducir la ruta crítica. La ruta crítica normalmente se define como la sección con mayor cantidad de niveles de lógica combinacional (tiempo entre dos registros). Al intercalar registros en distintas secciones de la ruta

crítica, esta disminuye; aumentando la frecuencia máxima de trabajo, lo que manifiesta una relación inversamente proporcional entre ellas.

Al introducir registros, se obtienen latencias en la salida de los datos. Cuando estas latencias son equivalentes al número de segmentaciones se dice que el pipe está lleno y los resultados de cada segmento comienzan a venir continuos y sin latencia en cada flanco de reloj.

Características generales del bloque IFFT

El bloque IFFT, está compuesto de forma general, por siete sub-módulos funcionales denominados *stage*. En términos jerárquicos estos están comprendidos dentro del sub-módulo *ifft_core*. El sub-módulo *ifft_core* interconecta cada uno de los *stages*, estableciendo el flujo de la información a procesar. Las señales que lo componen se muestran en la figura 10.

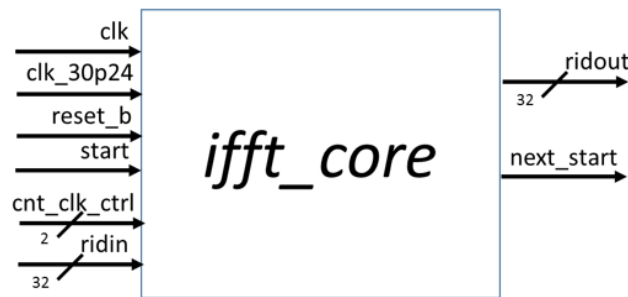


Figura 10: Entradas y salidas de *ifft_core*.

La señal *start* se utiliza para garantizar el sincronismo y proviene de *ifft_out_sync*. La señal *next_start*, proveniente de la señal de sincronismo a la salida del *stage 7*, permite la sincronización con el próximo módulo funcional U2.

A través de la señal de entrada *ridin* se obtiene la información útil que será procesada mediante la IFFT, proveniente de la señal *ridin_core*, que concatena los datos de *ifft_in_I* e *ifft_in_Q*.

Mediante la señal de salida *ridout* se adquiere la información final, como consecuencia de todo este proceso, es decir, símbolos OFDM.

Arquitectura interna de los sub-módulos que componen el bloque IFFT

La arquitectura IFFT Pipeline empleada en el diseño bajo análisis, divide en siete sub-módulos denominados *stages* el cálculo de la transformada inversa de 3 780 símbolos como se explicó anteriormente. Cada uno está compuesto por tres bloques principales: conmutador o de acondicionamiento, operadores IDFT básicos (*butterfly*) y multiplicador complejo. (Figura 11)

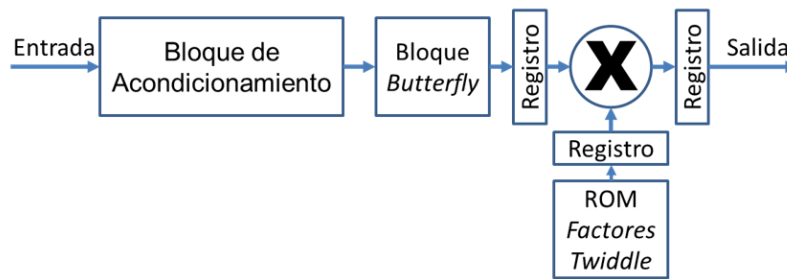


Figura 11: Arquitectura IFFT Pipeline empleada en cada etapa.

El bloque de acondicionamiento permite adaptar la secuencia de los datos de entrada de cada *stage*, al reordenamiento que precisan los mismos para la realización de la IDFT según sea el caso.

El bloque *butterfly* realiza las IDFTs básicas en cada *stage* mediante operaciones de suma y resta complejas y multiplicaciones por valores complejos constantes.

El multiplicador complejo combina las muestras que componen las transformadas de mayor nivel que se quieren lograr, con sus factores *twiddle* correspondientes.

Las operaciones complejas de suma y resta realizadas en el cálculo de las IDFTs y la multiplicación por los factores *twiddle*, tendrán más términos mientras mayor sea el orden de la transformada. La representación del resultado de cada operación de suma-resta, potencialmente incrementa en uno, el número de bits respecto a la representación del mayor sumando. Es por eso que se hace necesario incluir etapas de saturación, dentro de cada uno de los bloques *butterfly*, multiplicadores complejos, módulo *ifft_corey* finalmente en la etapa de salida.

Etapas de salida de la IFFT

Una vez que los datos se encuentran en las salidas de *ifft_top* se encuentran listos para ser enviados a U2. El bloque contenido en U2 encargado de recibir estos datos es el *srrc_top*. Este bloque corresponde con la función de filtro SRRC (del inglés Square Root Raised Cosine). Estos filtros son comúnmente utilizados en sistemas de comunicaciones de ancho de banda limitado para el filtrado digital en banda base.

La señal transmitida a través de estos sistemas pasa por un filtro en forma de pulso que la limita al ancho de banda deseado cumpliendo con el criterio de Nyquist.

Se ha definido la etapa de salida de la IFFT como el proceso más inmediato que sufren las señales provenientes del bloque *ifft_top*, al llegar al bloque *srrc_top* de U2. Dado que el interés fundamental de este trabajo es el comportamiento interno del bloque IFFT, sólo se abordará lo que se consideró esencial en relación con las salidas.

Las señales de salida *ifft_out_I* e *ifft_out_Q* llegan al bloque *srrc_top*. Una vez en este pasan a una etapa de saturación que se muestra en la figura 12 la cual es idéntica para *ifft_out_I* e *ifft_out_Q*.

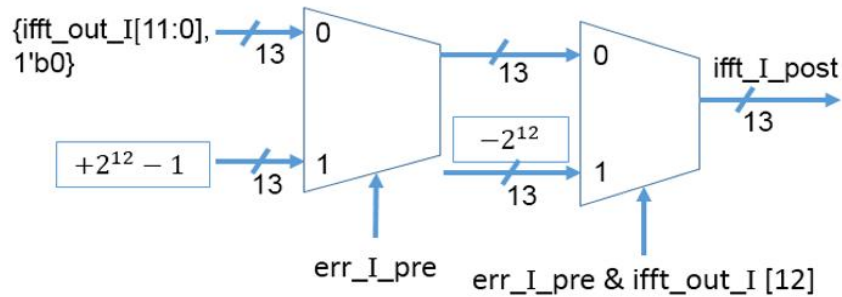


Figura 12: Etapa de saturación.

La señal *err_I_pre* informa si ocurrió desbordamiento sin incluir información de signo, esta información la brinda el segundo multiplexor dado que si el **and** de selección da cero implica que el bit 12 es cero y el número es positivo de lo contrario será negativo.

Al salir de esta etapa las señales son multiplexadas, como se indica en la figura 13, donde se selecciona si las señales con las que trabajará el bloque U2 provienen de *fec_top* o de *ifft_top*. Al igual que con la etapa de saturación esta figura será idéntica para las señales que contienen la información In-Phase e In-Quadrature.

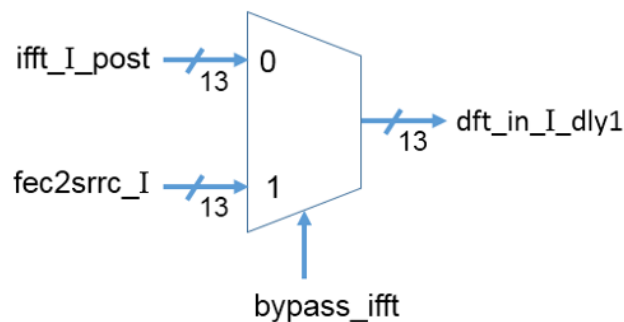


Figura 13: Multiplexación de los datos de entrada de *srrc_top*.

Las señales de sincronismo tanto para lo que proviene del *fec_top* como para lo que proviene de *ifft_top*, no requieren pasar por la etapa de saturación puesto que son de un bit. Por tanto las señales

ifft_out_sync y *fec2src_sync* son multiplexadas de acuerdo a la misma lógica de los datos, ilustrada en la figura 13.

Hasta este punto se realiza el seguimiento de las señales provenientes del bloque IFFT, en el contexto de interés de esta investigación.

CONCLUSIONES

En la presente investigación se estudiaron los principios fundamentales de la modulación OFDM y se argumentaron las razones por las cuales es requerido su empleo para transmitir, en múltiples portadoras.

Se logró mediante la aplicación de ingeniería inversa, la caracterización del bloque IFFT que implementa la modulación OFDM según los requisitos del estándar DTMB.

Se identificó el tipo de arquitectura empleada como IFFT Pipeline la cual permite el procesamiento en un flujo continuo de datos, que es muy utilizado en sistemas en tiempo real.

Se corroboró que la implementación de métodos IFFT garantiza el cálculo de transformadas de elevado valor con el mínimo empleo de recursos.

Se elaboraron diagramas en bloque que describen de forma ilustrativa la estructura seguida en el diseño.

Finalmente se obtuvo un cúmulo de información relevante para el entendimiento y asimilación de las tecnologías asociadas a la norma de TDT conocida como DTMB, lo que representa un paso significativo en el contexto actual de Cuba.

REFERENCIAS

1. TSINGHUA, U., *DTMB(GB20600-2006) Modulator FPGA Code DT-1008-A1*. 2012.
2. LOUIS LITWIN, M.P., *The principles of OFDM*. 2001.
3. RODRÍGUEZ, N.G., *Diseño de la Transformada de Fourier para el estándar Digital Television Multimedia Broadcast 2012*, IPSJAE: La Habana, Cuba.
4. BURRUS, C. Sidney, e.o., *Fast Fourier Transforms*. 2008.
5. B. BALA, Tripura Sundari, T.R.P., *Design Through Verilog HDL*. 2009

6. CORTÉS, Ainhoa; VÉLEZ, Igone; SEVILLANO, Juan F.; IRIZAR, Andoni, *"An Approach to Simplify the Design of IFFT/FFT Cores for OFDM Systems,"* IEEE Transactions on Consumer Electronics, Vol. 52, No. 1, FEBRUARY 2006, pp. 26-32 [11]
7. CHI, J.C.; S. Chen, *"An efficient FFT twiddle factor generator,"* in Proc. European Signal Process. Conf., 2004, pp. 1533–1536.
8. ZHOU, B.; HWANG, D. *"Implementations and optimizations of pipeline FFTs on Xilinx FPGAs,"* in Proceedings of the International Conference on Reconfigurable Computing and FPGAs (ReConFig '08), pp. 325–330, 2008.