

Implementación de un Transmisor de ISDB-T Abierto Bajo el Paradigma de Radio Definida por Software

Javier Hernández (javier.hernandez@fing.edu.uy), Santiago Castro (santiago.castro@fing.edu.uy)

Resumen—ISDB-T es el principal estándar adoptado en América Latina para la difusión de televisión digital terrestre. Hoy en día el estándar está consolidado en la región, los países de América Latina han realizado importantes inversiones para desplegar el nuevo sistema de televisión digital terrestre basado en este estándar. Esto nos lleva a investigar y desarrollar nuevas posibilidades en lo que respecta al sistema de televisión digital. En este artículo presentamos gr-isdbt-tx, un transmisor ISDB-T abierto, libre y gratuito totalmente basado en software, implementado en GNU Radio. Esta implementación es capaz de generar una señal de televisión digital compatible con cualquier televisor digital homologado para ISDB-T. También permite que estudiantes, profesionales e investigadores puedan conocer al detalle cómo es que funciona un transmisor de televisión digital, sin necesidad de costear equipos de gran valor económico. En conjunción con gr-isdbt: Un receptor ISDB-T fullseg abierto y libre implementado en GNU Radio, el usuario tiene una poderosa herramienta para correr un sistema de televisión digital de punta a punta, y así explorar y probar el comportamiento de un sistema de comunicación complejo sin ningún tipo de hardware. En este artículo presentamos el transmisor desarrollado, así como sus fundamentos y los problemas más relevantes que han tenido que ser resueltos para lograr una implementación exitosa. También presentamos algunas pruebas y sus resultados.

Términos—Integrated Services Digital Broadcasting, GNU Radio, Software Defined Radio.

I. INTRODUCCIÓN

Los antiguos sistemas de televisión analógica están siendo reemplazados por los nuevos sistemas digitales. Existe una innumerable cantidad de mejoras y ventajas respecto a los antiguos, como por ejemplo una mayor eficiencia espectral, mejor calidad de audio y video, la posibilidad de transmitir múltiples programas al mismo tiempo en el mismo canal, y la utilización de canales de datos. El enorme crecimiento que han tenido las comunicaciones inalámbricas en los últimos tiempos, ha hecho que el espectro radioeléctrico se convierta en un recurso natural finito extremadamente valioso. Varios países ya han transitado el camino apagón analógico, reemplazando definitivamente los sistemas de televisión analógica. Sin embargo hay muchos otros en los que aún conviven ambos sistemas de teledifusión. El despliegue de un nuevo sistema de televisión digital es una tarea mayúscula, tal como en otros campos de las comunicaciones inalámbricas surgen distintas soluciones para el mismo problema. Respecto a los sistemas de televisión digital, se pueden encontrar los siguientes estándares: ISDB (Integrated Services Digital Broadcasting) es el estándar japonés, DVB (Digital Video Broadcasting) es el estándar europeo, ATSC (Advanced Television Systems

Committee) es el estándar norteamericano y DTMB (Digital Terrestrial Multimedia Broadcast) es el estándar chino.

Este trabajo se enfoca en el estándar ISDB-T el cual ha sido ampliamente adoptado por los países de América Latina. Muchos países aún se encuentran desplegando sus sistemas de televisión digital, para lograr un despliegue exitoso es imprescindible contar con una comprensión profunda de la norma. Este grado de dominio no sería posible alcanzarlo si no se conoce en detalle todo el sistema, desde la etapa de transmisión hasta la etapa de recepción. Las compañías que fabrican esta clase de equipamiento son algunas pocas en el mundo y sus soluciones no suelen estar al alcance de todos.

Las Radios Definidas por Software (SDR por sus siglas en inglés) son una gran alternativa para desarrollar sistemas de comunicación. Se trata de un nuevo paradigma donde los sistemas de radiocomunicaciones, tradicionalmente implementados en hardware, son en cambio implementados por software. Esto típicamente se lleva a cabo a través de una computadora personal encargada de ejecutar el software, y un hardware genérico que se encarga de sintonizar, filtrar y muestrear la señal de radio a cierta tasa. De esta manera la señal puede ser procesada por la computadora. Hay una gran variedad de estos equipos que van desde unos pocos dólares hasta algunos cientos o incluso miles de dólares. El paradigma de la radio definida por software permite lograr una implementación completa de sistemas de radiocomunicaciones de manera económica y sencilla en cierto sentido.

En este trabajo en particular, utilizamos una PC de propósito general junto con equipos basados en SDR como lo son el USRP B100 (Universal Software Radio Peripheral de Ettus) [1] o el HackRF One (de Great Scott Gadgets) [2] que cumplen con la tasa de muestreo requerida por el estándar ISDB-T. Para el desarrollo utilizamos el toolkit de GNU Radio, se trata de un software ampliamente utilizado para el desarrollo de radios definidas por software. Provee de bloques de procesamiento de señal de propósito general como filtros, re-samplers, moduladores y demoduladores analógicos y digitales. Estos bloques pueden ser combinados para lograr diferentes sistemas de comunicación. También es posible la implementación de nuevos bloques personalizados por parte de los usuarios, de manera de contribuir con el código fuente de GNU Radio. Esto hace que la comunidad de GNU Radio sea muy activa y resulte en una contribución continua de los usuarios tanto en el desarrollo de nuevas radios definidas por software o el testeo de otras implementaciones.

En este trabajo presentamos gr-isdbt-tx

(<https://github.com/jhernandezbaraibar/gr-isdbt-Tx>), un transmisor de ISDB-T abierto bajo el paradigma de radio definida por software, que continúa y complementa la línea de trabajo comenzada con el receptor gr-isdbt.

Este transmisor ha sido implementado completamente en GNU Radio y puede tomar hasta tres flujos de entrada MPEG-2 tal como lo establece el estándar. Tiene la posibilidad de trabajar tanto en modo one-seg como en full-seg, y el usuario cuenta con la libertad de establecer los parámetros de configuración del sistema. El desarrollo sobre GNU Radio hace que, con solamente un PC, el usuario pueda correr un sistema de televisión digital de punta a punta integrando gr-isdbt-tx con gr-isdbt. Alternativamente con un hardware SDR, como el USRP B100, es posible transmitir la señal por el aire hacia cualquier televisor de uso doméstico compatible con ISDB-T.

En lo que sigue se presenta una descripción del estándar ISDB-T — Los detalles de programación y los algoritmos utilizados pueden ser consultados en detalle en el repositorio del proyecto. Aquí describiremos el funcionamiento del sistema

II. EL ESTÁNDAR ISDB-T

El estándar de televisión digital terrestre ISDB-T puede ser caracterizado en cuatro grandes etapas: la conformación del Broadcast Transport Stream (BTS), la codificación de canal, la formación del cuadro OFDM, y la puesta en el aire de la señal. En la figura ?? se presenta el esquema completo del sistema transmisor. El sistema admite como fuente de datos hasta tres flujos de transporte MPEG-2 [3]. Cada flujo está conformado por una secuencia de paquetes denominados *Transport Stream Packet* (TSP) de 188 bytes. Los flujos MPEG-2 son multiplexados por el bloque TS Remux en un único flujo llamado *Broadcast Transport Stream* (BTS); además de la multiplexación de los TSP el TS Remux tiene otras tareas como el agregado de cierta información jerárquica en los paquetes.

Una de las características destacables de ISDB-T es la posibilidad de la transmisión jerárquica de la información, esto es, que cada flujo MPEG-2 puede ser transmitido con una configuración propia con distintos grados de robustez. A la transmisión de un flujo MPEG-2 con su configuración jerárquica se lo denomina *capa jerárquica* y el estándar permite transmitir hasta tres capas en simultáneo. Cada capa tiene asignada un conjunto de portadoras en el espectro que más adelante se analizará su conformación.

Una vez conformado el BTS se comienza a robustecer la señal agregando redundancia y utilizando distintos tipos de entrelazamientos. En los sistemas de comunicaciones es común encontrar que se utilicen códigos correctores de errores concatenados, esto aumenta significativamente el desempeño de los mismos. En ISDB-T se utiliza como código exterior un Reed-Solomon (204,188) y un código convolucional (FEC) como código interior.

Como cada flujo debe ser procesado por separado es necesario que a cada uno se le aplique un proceso según la configuración jerárquica elegida. Es por esto que hay un

bloque **divisor jerárquico** encargado de separar el BTS en tres flujos y encaminarlos según lo que corresponda.

El robustecimiento de la señal se realiza con distintos objetivos, por ejemplo para evitar el fading en frecuencia, las ráfagas de errores, ..., etc. El bloque dispersor de energía tiene como objetivo eliminar posibles largas secuencias de unos o ceros que puedan haber en el flujo de transporte. De otra manera podrían haber errores de sincronismo y también habrían porciones del espectro donde se concentraría la energía, quedando así subutilizado este recurso. El dispersor de energía utiliza un circuito para generar una secuencia pseudo-aleatoria la cual es invertible aplicando el mismo circuito a la secuencia. Como resultado se obtiene una secuencia aleatoria sin largas secuencias de ceros y unos.

A continuación del dispersor de energía se encuentra el bloque de **entrelazamiento de bytes**. El entrelazamiento consiste en aplicar una serie de retardos a los distintos bytes que van arribando al bloque. Como el receptor conoce la manera en que se realizaron estos retardos es capaz de deshacerlos y volver a la secuencia original. Esta estrategia de entrelazamiento es ampliamente utilizada luego de un código de bloque como el Reed-Solomon, este último es capaz de corregir errores de hasta ocho bytes. Si durante la transmisión se corrompiera una cantidad mayor de bytes consecutivos el código se volvería inútil, sin embargo al estar los bytes entrelazados el receptor ve estos errores como puntuales y el código es capaz de corregirlos.

Como *inner code* se utiliza un código convolucional con *puncturing* cuya tasa madre es 1/2. La técnica del *puncturing* permite lograr **tasas de código** de 1/2, 2/3, 3/4, 5/6, 7/8.

La siguiente etapa consiste en el **entrelazamiento de bits y mapeo**. Este proceso es similar al entrelazamiento de bytes; en *gr-isdbt-tx* los bits que ingresan a este bloque son empujados en distintas colas de distintos tamaños, luego se extrae el bit que se encuentra al otro extremo de cada cola. Con esto se logra tener a la salida un flujo de bits entrelazados. Es importante tener en cuenta que, tanto el bloque de entrelazamiento de byte como el de bit, generan atrasos que deben ser compensados. En *gr-isdbt-tx* el mapeo de bits en símbolos complejos se realiza conjuntamente con el entrelazamiento de bits. ISDB-T admite las modulaciones DQPSK, QPSK, 16QAM y 64QAM. Este proceso simplemente consiste en agrupar los bits en símbolos complejos según tablas que pueden consultarse en [4]. Hasta este punto los flujos de datos de las distintas capas vienen siendo procesados de acuerdo a los parámetros de cada una de ellas, ahora resta volver a combinar los tres flujos para lograr un único flujo de transporte. El bloque que se encarga de esta tarea es el **combinador jerárquico**. Luego de armado el flujo único con las tres capas jerárquicas, se realiza un **entrelazamiento temporal** de los símbolos complejos. Consiste en distribuir los símbolos complejos en el dominio del tiempo. Esta técnica actúa como mecanismo de protección frente a ruidos impulsivos que típicamente se caracterizan por una corta duración en el tiempo. La **profundidad** o largo del entrelazamiento temporal es el parámetro que rige este proceso de entrelazado y puede ser seteado de manera independiente para cada capa. Está íntimamente ligado a la dispersión temporal que se realiza en los símbolos; a mayor profundidad,

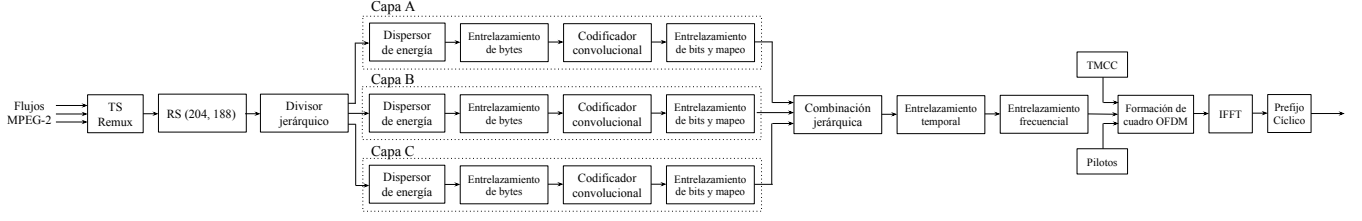


Fig. 1: Esquema del sistema transmisor ISDB-T.

los símbolos de una misma portadora son retardados un tiempo mayor.

Debido a la característica de ISDB-T de utilizar un gran número de portadoras en el espectro, ocurre que frente a canales selectivos en frecuencia se pueden ver afectadas una cantidad importante de portadoras consecutivas. Una estrategia para mitigar esto es el **entrelazamiento frecuencial**. El entrelazamiento que realiza este estándar consiste en dos etapas; el entrelazamiento inter-segmentos y el entrelazamiento intra-segmentos. En la primera etapa los símbolos que comparten la misma modulación son intercalados, mientras que en la segunda etapa los símbolos de un mismo segmento son rotados de acuerdo a cierta expresión que establece el estándar. Posteriormente son aleatorizados según ciertas *Look Up Tables* [4].

Para colocar la señal en el aire, ISDB-T utiliza un esquema de **modulación OFDM** [5] (Orthogonal Frequency-Division Multiplexing). Este esquema consiste en el uso de portadoras ortogonales, lo que ofrece una mejor eficiencia espectral respecto a los clásicos esquemas de multiplexación en frecuencia. La idea de esta tecnología consiste en modular pulsos rectangulares con los símbolos complejos provenientes del sistema, que como ya se mencionó, pueden ser DQPSK, QPSK, 16QAM y 64QAM. Luego de esta modulación se agrega un **prefijo cíclico** que consiste en una copia del símbolo OFDM y es una fracción del símbolo activo. Puede tomar los valores de 1/4, 1/8, 1/16, 1/32. Esta copia consiste en tomar la última porción del símbolo activo OFDM y copiarlo al comienzo. Es posible ver que si la señal tiene un ancho de banda W , y N es la cantidad total de portadoras, entonces un símbolo activo OFDM tiene una duración $T_s = N/W$ [5].

En cuanto al número de portadoras utilizadas, ISDB-T admite la utilización de 2^{11} , 2^{12} o 2^{13} portadoras según si el **modo de transmisión** es 1, 2 o 3 respectivamente. Sin embargo no todas estas portadoras son utilizadas ya que algunas son reservadas para mantener un intervalo de guarda. Las portadoras son divididas en 14 conjuntos denominados **segmentos**, cada uno de ellos con igual número de portadoras. De ellos 13 son utilizados para transmitir datos y el restante es utilizado como guarda a ambos lados del espectro.

Es usual encontrar que en los sistemas de comunicación digitales se delimiten los datos transmitidos dándoles forma de cuadros con una duración fija. Esto vuelve más sencillas las tareas de sincronización dado que en recepción se sabe en qué momento comienza un cuadro. Particularmente en los sistemas OFDM estos cuadros se denominan precisamente **cuadros OFDM**. El cuadro OFDM es una estructura de datos

Parámetro	Valor
Ancho de banda del canal	6 MHz
Cantidad de segmentos	13
Ancho de banda de cada segmento	$6000/14 \approx 428.57 kHz$
Cantidad de portadoras activas por segmento	96 de datos y 12 pilotos (Modo 1) 192 de datos y 24 pilotos (Modo 2) 384 de datos y 48 pilotos (Modo 3)
Duración de símbolo activo	$252\mu s$ (Modo 1) $504\mu s$ (Modo 2) $1008\mu s$ (Modo 3)
Duración del prefijo cíclico	1/4, 1/8, 1/16, 1/32 (fracción del símbolo activo)
Tasa de código convolucional	1/2, 2/3, 3/4, 5/6, 7/8
Tasa de código Reed-Solomon	(188, 204)
Profundidad del entrelazamiento temporal	0, 1, 2, 4 (Modo 1) 0, 2, 4, 8 (Modo 2) 0, 4, 8, 16 (Modo 3)
Esquemas de modulación	DQPSK, QPSK, 16QAM, 64QAM
Frecuencia de muestreo (f_{FFT})	$512/63 \approx 8.127 MHz$

Tabla I: Parámetros relevantes en el estándar ISDB-T.

que agrupa los datos de carga útil, portadoras piloto y señales de control. Tiene todo lo necesario para que un receptor compatible con ISDB-T sea capaz de decodificarlo en flujos MPEG-2 y reproducir la información.

Se trata de un conjunto de 204 símbolos OFDM, cada uno de ellos formados por $13 \times 108 \times 2^{modo-1}$ símbolos complejos. El bloque que se encarga de conformar el cuadro OFDM también tiene a cargo las tareas de insertar ciertas señales piloto como son la **Transmission and Multiplexing Configuration Control** (TMCC), las **Scattered Pilot** (SP) y las **Auxiliary Channel** (AC).

La TMCC contiene información de sincronismo y control para que el receptor pueda saber qué cantidad de capas jerárquicas se está utilizando, qué segmentos están asignados para cada una de ellas, la modulación utilizada, el código convolucional y entrelazamiento temporal entre otra información útil.

Las SP o portadoras dispersas son señales BPSK que se insertan a lo largo de los símbolos OFDM y van rotando símbolo a símbolo. Estas portadoras se generan a partir de un circuito que evoluciona según el índice de la SP dentro del símbolo OFDM. El objetivo de estas portadoras es la estimación del canal y por lo tanto son rotadas para evitar que caigan siempre en un lugar que presente fade en frecuencia.

También existe la posibilidad de transmitir información adicional en las portadoras AC, consiste en portadoras que utilizan modulación DBPSK pensadas para oficiar de canal auxiliar. En caso de no ser utilizadas se rellenan con unos.

Con el propósito de facilitar el sincronismo en recepción, se utiliza un **piloto continuo** modulado en BPSK a la derecha del espectro. En la tabla I se presenta en detalle los parámetros más relevantes en el estándar.

Una vez armado el cuadro OFDM lo que sigue es la modulación de los símbolos en el esquema OFDM. Para ello se aplica el algoritmo de la **Inverse Fast Fourier Transform (IFFT)** y a continuación se agrega el prefijo cíclico.

III. GNU RADIO Y LAS RADIOS DEFINIDAS POR SOFTWARE

GNU Radio es un entorno de desarrollo orientado a procesamiento de señales, gratuito y de código abierto. Mediante la interconexión de bloques de procesamiento, puede usarse en conjunto con antenas de RF para desarrollar SDRs, o en una versión local en modo de simulación de sistemas. La aplicación por defecto trae una amplia gama de bloques funcionales para trabajar, desde herramientas simples como filtros y ecualizadores, hasta estructuras complicadas, como lo es un transmisor TVD completo. Existe además una gran comunidad, muy activa, que está permanentemente publicando nuevos contenidos, para ampliar la gama de herramientas existentes en la actualidad. Esto se da porque al ser de código abierto, es relativamente sencillo crear bloques nuevos. El entorno soporta desarrollos de código en Python y en C++, siendo este último el lenguaje con el que hemos implementado nuestro transmisor *gr-isdbt-tx*.

IV. EL BROADCAST TRANSPORT STREAM

El estándar MPEG no está pensado para la transmisión en capas jerárquicas. Para lograr la característica de la transmisión jerárquica y la recepción parcial, ISDB-T cuenta con una solución propia denominada Broadcast Transport Stream. Esta solución consiste en el uso de un flujo de transporte propio, formado a partir de tres flujos MPEG multiplexados, lo que supone un procesamiento no tan sencillo. El nuevo flujo deberá incluir cierta información jerárquica, que no provee MPEG, que permita identificar los flujos con sus correspondientes capas jerárquicas. Es importante que el sistema funcione a una velocidad de reloj constante y determinada, por este motivo el flujo BTS deberá tener una tasa perfectamente definida y constante independientemente de los parámetros de cada capa.

La conformación del BTS se lleva a cabo en el bloque TS Remux. Es el encargado de agregar 16 bytes al final de cada paquete TS con la ISDB-T information y la paridad; posicionar los TSP dentro del BTS de acuerdo a cierto patrón de ordenamiento que se detallará, para posibilitar la transmisión jerárquica; insertar la cantidad necesaria de paquetes nulos para asegurar una tasa constante

$$r_{BTS} = 4 \times f_{IFFT} \approx 32.508 \text{ Mbps} \quad (1)$$

La obtención de la fórmula anterior puede encontrarse en [?]. Durante la división jerárquica la ISDB-T information de los TSP es removida por lo cual en las siguientes fases de procesamiento ya no se cuenta con la información jerárquica en los flujos de transporte. Es por eso, que resulta necesario definir un orden dentro del BTS de manera que el receptor

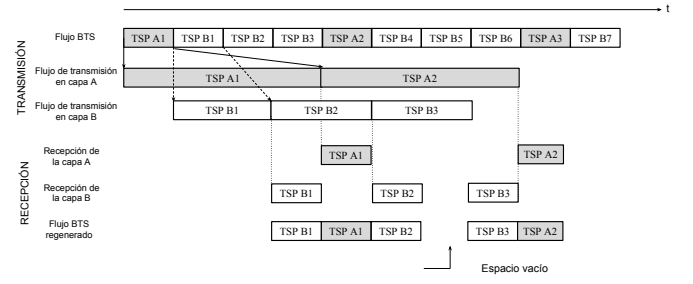


Fig. 2: Patrón de ordenamiento incorrecto del BTS.

pueda reconstruirlo perfectamente sin necesidad de la ISDB-T information. Supongamos que tenemos un BTS como el de la figura 2.4 en el cual hay dos capas jerárquicas A y B, cada una con su respectiva tasa tal que $r_{BTS} > r_B > r_A$. Las capas con tasas altas implican un tiempo de transmisión de paquetes mucho más corto que las capas con tasas bajas, y viceversa. Observando el diagrama de la figura 2 puede verse que en recepción el paquete B1 es procesado antes de que termine de ser transmitido el paquete A1, cuando en realidad el orden original era A1 y luego B1. Además se generan espacios de tiempo en los cuales no se entregan paquetes al sistema porque aún se están procesando paquetes que van arribando. Esos intervalos de tiempo deben ser rellenos con paquetes nulos a fin de mantener la tasa constante. Esto resulta en un desordenamiento del patrón original del BTS y de la pérdida total de la capacidad de reordenar los datos en recepción.

Para formar los patrones de ordenamiento en transmisión de manera tal que el receptor sea capaz de recuperarlos perfectamente, se agregan los denominados ajustes de atraso. En el ejemplo de la figura 3 se agrega un ajuste de atraso de 2 TSP y se introduce un paquete nulo en el BTS original. Como resultado, el flujo BTS reconstruido por el receptor resulta ser exactamente igual al BTS original. Dada una configuración jerárquica del sistema, existe un único patrón de ordenamiento del BTS.

El transmisor implementado en este trabajo toma como flujo de entrada un BTS ya conformado, y con la información jerárquica de los TSP es que logra procesar cada capa por separado. De ahí en adelante las capas son entrelazadas y moduladas cada una de acuerdo a su propia configuración.

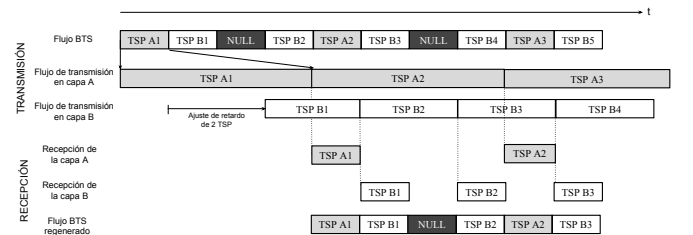


Fig. 3: Patrón de ordenamiento correcto del BTS.

V. LA ESTRUCTURA DE CUADRO OFDM

VI. LOS AJUSTES DE ATRASO Y EL SINCRONISMO

Los bloques de entrelazamiento introducen atrasos. Estos atrasos son producto de los retardos variables a los que se

someten los bytes, bits o símbolos complejos. Para que el sistema funcione correctamente el atraso total introducido por cada bloque debe ser un múltiplo de cuadro OFDM. Esto es un detalle importante que se tuvo en cuenta en el desarrollo de *gr-isdbt-tx*, por lo que entendemos necesario desarrollar el concepto del ajuste de atraso.

A. Ajuste de atraso en el entrelazamiento de byte

B. Ajuste de atraso en el entrelazamiento de bits

En este entrelazamiento se utilizan 2, 4 ó 6 colas dependiendo si la modulación utilizada es QPSK, 16QAM, o 64QAM respectivamente. El tamaño de las colas, q_i , viene definido por el estándar. Por ejemplo para 64QAM los tamaños son $q_0 = 0$, $q_1 = 40$, $q_2 = 80$, $q_3 = 120$. Cualquiera sea la modulación el retardo máximo siempre es de 120 bits. Sin embargo vamos a dejarlo paramétrico en q_i a fin de calcular el ajuste de atraso que es necesario introducir a cada cola.

El retardo agregado por el proceso corresponde al camino más largo que deben atravesar los bits, es decir un retardo de 120 símbolos. Según la modulación que utilice la capa, estos 120 símbolos se traducen en un mayor o menor retardo en el tiempo; para QPSK corresponde a esperar que ingresen 120×2 bits en la entrada, mientras que para 64QAM deben pasar 120×6 bits.

Como los retardos deben ser siempre múltiplos de símbolos OFDM, el tamaño de un símbolo OFDM corresponde a:

$$N_{OFDM} = m_L \times N_L \times 96 \times 2^{modo-1} \text{ bits} \quad (2)$$

N_{OFDM} depende de la cantidad de segmentos utilizados N_L , del modo de transmisión y de la modulación de la capa $m_L \in \{2, 4, 6\}$.

La solución implementada en este bloque es distinta a la del entrelazador de bytes. Aquí, el retardo total se obtiene incrementando los tamaños de las colas de manera que el retardo total sea de dos símbolos OFDM para cada capa. Por lo tanto para tener el retardo deseado los tamaños de las colas deben ser:

$$Q_i = q_i + \frac{2 \times N_{OFDM} - 120 \times m_L}{m_L} \quad (3)$$

$$Q_i = q_i + N_L \times 96 \times 2^{modo-1} - 120 \quad (4)$$

Una vez que son empujados los bits en cada cola, desde el otro extremo se extrae un bit por cola y se mapean en un símbolo complejo de acuerdo a lo establecido en la norma.

C. Ajuste de atraso en el entrelazamiento temporal

En el caso del entrelazamiento temporal, los símbolos complejos que arriban al bloque son encolados de manera secuencial en búfers de tamaño definido por la siguiente expresión:

$$q_L(i) = I_L \times ((i \times 5) \bmod 96) \quad (5)$$

donde $i \in [0, 96 \times 2^{modo-1}]$ representa el número de portadora e I_L es la profundidad del entrelazamiento y puede tomar los valores de la tabla I.

Este proceso genera un atraso de $95 \times I_L \bmod 204$ símbolos, entonces para completar un cuadro OFDM hacen falta:

$$d_I = 204 - (95 \times I_L \bmod 204) \quad (6)$$

símbolos OFDM. Por lo tanto el retardo total en cuadros OFDM introducido por el proceso será:

$$N_L = (95 \times I_L + d_I) \bmod 204 \quad (7)$$

La implementación en el transmisor de este ajuste de atraso consiste en empujar los símbolos que arriban al bloque a cada una de las colas de tamaño $I_L \times (i \times 5) \bmod 96 + d_I$. Del otro extremo de las colas se toman los símbolos secuencialmente y se tiene la secuencia entrelazada. Este mecanismo, al igual que otros entrelazamientos, hace que en el arranque del sistema se tengan datos espurios en los registros.

VII. PRUEBAS

A. Subsection Heading Here

Subsection text here.

1) Subsubsection Heading Here: Subsubsection text here.

VIII. CONCLUSIONES

The conclusion goes here.

APÉNDICE A

PROOF OF THE FIRST ZONKLAR EQUATION

Appendix one text goes here.

APÉNDICE B

Appendix two text goes here.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

- [1] E. Research, <https://kb.ettus.com/B100>, 2010, [Web; accedido el 01-10-2018].
- [2] G. S. Gadgets, 2018, [Web; accedido el 01-10-2018]. [Online]. Available: URL <https://greatscottgadgets.com>
- [3] MPEG, "Moving Picture Experts Group," <https://mpeg.chiariglione.org/>, 1995, [Web; accedido el 06-11-2018].
- [4] ARIB, "Transmission System for Digital Terrestrial Television Broadcasting," https://www.arib.or.jp/english/html/overview/doc/6-STD-B31v1_6-E2.pdf, 2001.
- [5] R. W. Chang, "Synthesis of band-limited orthogonal signals for multi-channel data transmission," *Bell System Technical Journal*, vol. 45, no. 10, pp. 1775–1796, 1966.