

DESARROLLO DE HERRAMIENTA SOFTWARE PARA LA MODIFICACIÓN DE LOS SUBTÍTULOS DVB DE LA TV DIGITAL

Manuel Galán Valdés¹, Beatriz Alba Toledo²

¹e-mail: mgalanv@tesla.cujae.edu.cu ² <mailto:beatriz.alba@etecsa.cu>

¹CUJAE, Marianao, ²ETECSA, División Territorial de Mayabeque

RESUMEN

Desde hace unos años, nuestro país comenzó el despliegue de la Televisión Digital en el territorio nacional. En este proceso se hace imprescindible la evaluación técnica de todos los receptores de la Televisión Digital (Set-Top Box, y televisores receptores de televisión digital, iDTV) que entran al país para su comercialización. Fue durante este proceso de evaluación técnica donde surgió la necesidad de evaluar todos los aspectos relacionados con los subtítulos DVB (Digital Video Broadcasting, norma europea para la televisión digital), de ahí la necesidad de contar con una herramienta que permita generar tramas de transporte que posibiliten explotar todas las funcionalidades de los mismos. El presente trabajo hace un recuento de los aspectos técnicos de los subtítulos DVB y de cómo esos son utilizados en el sistema actual de transmisión. Además, se realiza un estudio de algunas herramientas que existen actualmente para generarlos e insertarlos en las tramas de transporte. Finalmente se desarrolla una herramienta que posibilita realizar ciertas modificaciones a las funcionalidades de los subtítulos presentes en las tramas de transporte. Con esto se sientan las bases para lograr implementar una herramienta capaz de realizar todas las modificaciones posibles.

PALABRAS CLAVE: Televisión Digital, receptor digital, iDTV, Trama de transporte, subtítulos DVB.

SOFTWARE TOOL DEVELOPMENT TO MODIFY DVB SUBTITLES ON DIGITAL TELEVISION SYSTEMS

ABSTRACT

Since a few years ago, our country began to broadcast Digital Television services around national territory. In this process, it is of major importance to asses technical evaluations to each acquired Digital Television box receiver (STB and iDTV) that arrives to our country. Based on this technical evaluation process began to evaluate main elements related to DVB subtitles. That's why it is needed a tool capable of generate transport streams which exploit its functionalities. This research explains technical aspects of DVB subtitles and how they are used on current transmission systems. Besides, this research addresses a study regarding some available tools which allow to generate them. It is developed a

software tool to perform some modifications to functionalities of the subtitles on transport streams. Additionally, a first step is taken to develop for modifying all functionalities of transport streams regarding DVB subtitles.

KEYWORDS: Digital Television, Set-top box, iDTV, Transport Stream, DVB subtitles.

INTRODUCCIÓN

Actualmente Cuba continúa en el proceso de despliegue de la Televisión Digital Terrestre (TDT). Gran parte del territorio nacional cuenta con cobertura de la señal digital haciendo posible que gran cantidad de usuarios pueda disfrutar de las ventajas de este servicio. Entre otras bondades, esta tecnología ofrece a los usuarios una calidad uniforme en la recepción de todos los programas digitales. Además, permite el acceso a nuevos servicios entre los que destacan la Guía Electrónica de Programa, el Servicio de Radiodifusión de Datos, así como la posibilidad de grabar un programa mientras se visualiza otro.

Las transmisiones digitales posibilitaron además la multiplexación de programas de radio junto a los programas de televisión ya existentes, así como la posibilidad de utilizar una opción de subtítulo más flexible y atractiva a los usuarios. La norma europea (DVB) para los subtítulos, hace posible esa flexibilidad toda vez que aporta novedosas modificaciones con respecto a las anteriores soluciones para subtítulos.

La introducción en el país de equipos receptores de Televisión Digital (TVD) conteniendo estas nuevas funcionalidades, hizo necesario contar con un proceso de evaluación técnica de los mismos. LACETEL, Instituto de Investigación y Desarrollo de Telecomunicaciones, es la entidad encargada de llevar a cabo este proceso de evaluación. Fue durante este proceso de evaluación donde se comenzaron a detectar problemas relacionados a los subtítulos DVB. Sin embargo, no se cuenta con un procedimiento que permita evaluar el soporte de todas sus funcionalidades en los receptores. Dada la situación problemática explicada anteriormente se hace necesario contar con una herramienta o procedimiento que posibilite analizar y modificar los subtítulos presentes en las tramas de transporte (TS). Luego posibilitar su empleo durante el proceso de evaluación técnica de los receptores digitales.

Para la resolución del problema se propone el diagrama en bloques de una herramienta software que conduzca el proceso de diseño del sistema, enfocándose en la implementación de los módulos Sincronismo, Estructura PSI1 y Extracción de PES2 de Subtítulos. Para la concepción de dicha herramienta se utiliza como ambiente de desarrollo el programa Visual Studio 2008, y se programa en lenguaje C.

DESARROLLO

La herramienta propuesta se compone de cuatro módulos principales: Sincronismo, Estructura PSI, Extracción PES de subtítulo y Modificación PES de subtítulo. A continuación, se describe funcionalmente cada uno de los módulos anteriormente mencionados.

¹ PSI: *Program Specific Information*. Información Específica de Programa.

² PES: *Packetized Elementary Stream*. Trama Elemental Paquetizada.

Módulo “Sincronismo”

Antes de realizar cualquier operación de lectura, procesamiento o extracción de datos de una Trama de Transporte MPEG-23, resulta de vital importancia sincronizarse con la estructura de paquetes de la misma. El primer paso a realizar es buscar el byte de sincronismo (`sync_byte`) entre los paquetes que conforman el flujo MPEG, el cual tiene valor `0x47` (0100 0111) y siempre es el primer byte del paquete, el cual está formado en su totalidad de 188 bytes. No obstante, el valor `0x47` puede aparecer en cualquier otra posición del paquete, en cuyo caso constituiría un dato del mismo [1].

La constante de valor `0x47` y la constante de espaciamento de 188 bytes, son elementos esenciales del módulo Sincronismo. Si un byte con el valor `0x47` es encontrado, debe ser examinado el byte que se encuentra posicionado 188 bytes después y verificar que coincida con el valor `0x47`. Si en alguna lectura, el valor obtenido no es un `sync_byte`, entonces los bytes de las consultas anteriores eran palabras de código que asumieron este valor. El sincronismo con la trama de transportes se completa después de encontrar satisfactoriamente el `sync_byte` en 5 paquetes consecutivos. Una vez establecido el sincronismo, el mismo se pierde cuando tres paquetes sucesivos no contienen el `sync_byte` [2].

La Figura 1 muestra el algoritmo implementado durante el diseño de este módulo. Cada vez que la herramienta es iniciada, se lleva a cero el contador de bytes de sincronismo (dado por la variable `i`) y se comienza a comparar cada byte que se lee con la constante `0x47`. Cuando sea verdadero se incrementa `i` y se pregunta si la misma alcanzó el valor 5. Si $i < 5$ se leen 187 bytes y se vuelve a comparar el siguiente byte con el valor `0x47`, si esta última comparación es falsa se vuelve al inicio y el contador es reiniciado. Si $i = 5$ se considera el sistema como sincronizado [2].

En el flujo de datos MPEG pueden ocurrir errores, lo cual provocar que los bytes de sincronismo de valor `0x47` dejen de ocurrir en las posiciones esperadas y esto provoque que se pierda el sincronismo con la estructura de paquete. El estado de sincronización se pierde cuando no son recibidos de manera consecutiva 3 paquetes conteniendo el `sync_byte` en la posición esperada [2]. En caso de perderse el sincronismo, el programa procederá a re-sincronizarse.

En el flujo de datos MPEG pueden ocurrir errores, lo cual da lugar a que los bytes de sincronismo de valor `0x47` dejen de ocurrir en las posiciones esperadas y esto provoque que se pierda el sincronismo con la estructura de paquete. El estado de sincronización se pierde cuando no son recibidos de manera consecutiva 3 paquetes conteniendo el `sync_byte` en la posición esperada [2]. En caso de perderse el sincronismo, el programa procederá a re-sincronizarse.

³ MPEG: *Moving Picture Experts Group*. Grupo de Expertos de Imágenes en Movimiento. Es un estándar de compresión

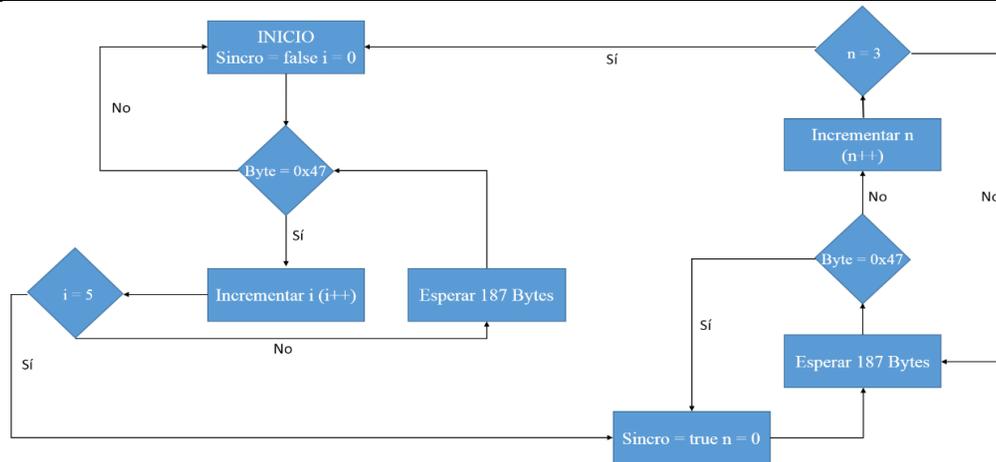


Figura 1: Diagrama de flujo de algoritmo para sincronizarse con la trama de transporte.

Estructura del paquete TS

La trama de transporte MPEG se conforma multiplexando varios programas de TV. Cada uno de estos programas se compone de información de video y audio, aunque en ocasiones también suelen contener datos. Todos ellos son encapsulados en flujos PES, los que a su vez se estructuran en paquetes de 188 bytes (ver Figura 2), conformando de esta forma la trama de transporte MPEG [3].

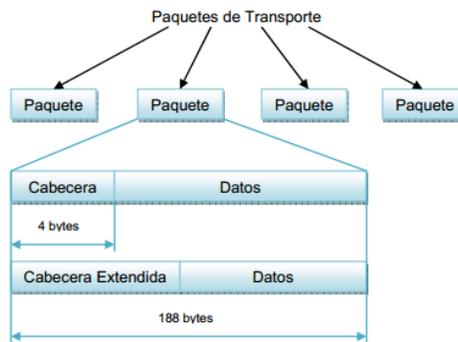


Figura 2: Estructura del TS

Un paquete TS puede ser transmitido conteniendo dos tipos de cabecera como muestra la Figura 2. Usualmente se utiliza la cabecera de longitud corta (32bits), pero en ocasiones es utilizada la cabecera extendida, la cual posee una longitud variable, siendo su valor 188 bytes – x bytes de datos. La cabecera extendida es usada generalmente para enviar otras informaciones. En la carga útil de ser necesario se utilizan bytes de relleno para mantener constante el tamaño de los paquetes TS, manteniendo de esta forma la eficiencia del sistema [4].

La Figura 3 muestra la estructura de la cabecera (sea extendida o no) de un paquete de transporte. La misma siempre comienza con el byte de sincronismo (sync_byte), al cual siguen los bits transport_error_indicator, payload_unit_start_indicator y transport_priority, los que indican

respectivamente, la presencia de errores después del proceso de demodulación, el comienzo en este paquete TS de un paquete PES y prioridad de transporte para este paquete sobre los de igual PID⁴.

El campo que sigue a estos bits es el PID (13 bits), el cual permite identificar un paquete de transporte de los restantes. Al campo PID le sigue el campo `transport_scrambling_control` (2 bits), este indica si los datos que transporta el paquete están codificados. Inmediatamente después se encuentra el campo `adaptation_field_control`, indicando si el campo adaptación (cabecera extendida) está presente en el paquete (ver Tabla 1) [5].

Tabla 1: Significado del valor del campo `adaptation_field_control`.

Valor	Descripción
00	Reservado para uso futuro por ISO/IEC
01	Sin campo de adaptación, solo carga útil
10	Solo campo de adaptación, sin carga útil
11	Campo de adaptación seguido por carga útil

El último campo presente en la cabecera de un paquete de transporte es el `continuity_counter` (4 bits). Este campo permite organizar los paquetes de igual PID una vez recibidos en el receptor, pues el valor de este campo es el orden del paquete.

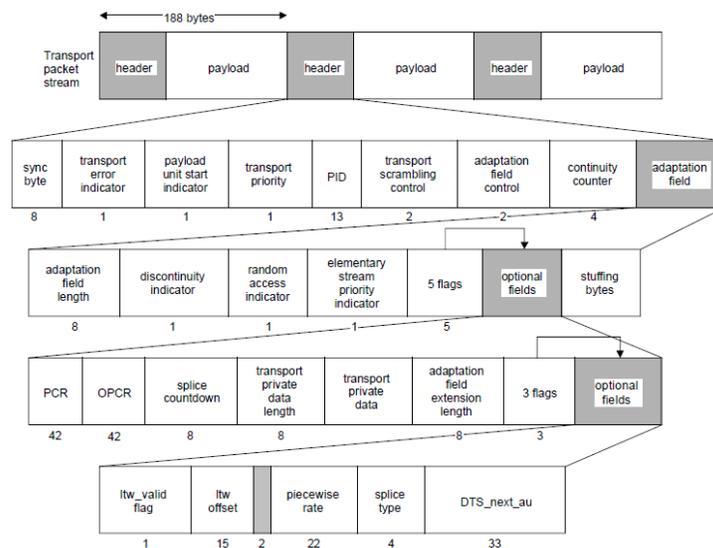


Figura 3: Diagrama de sintaxis de trama de transporte

⁴ PID: *Packet Identifier*. Identificador de Paquete.

Módulo “Estructura PSI”

El módulo Estructura PSI es el encargado de conocer cómo la trama de transporte se encuentra estructurada, es decir, conocer los distintos elementos que componen a los programas presentes en la misma. Para tal fin, este módulo debe ser capaz de buscar y registrar la estructura de tablas PSI presentes en el flujo MPEG, específicamente las tablas PAT⁵ y PMT⁶. Este módulo además brinda al usuario la información relativa a los programas que contienen subtítulos DVB, especificando el idioma de los mismos.

Estructura de las tablas PAT y PMT

La tabla PAT transporta los valores de PID en los que se encuentra el mapa de información de programa, es decir, contiene los PID de las tablas PMT de cada programa presente en la trama de transporte. También contiene el valor de PID donde está siendo transmitida la tabla NIT. Esta transmite la información relativa a la organización física del multiplexor y características de la propia red. La tabla PAT es transmitida en paquetes con PID = 0x0000 y con valor 0x00 en el campo `table_id`.

Por su parte, la tabla PMT transporta información relativa a los distintos flujos (video, audio o datos) que componen un determinado programa, indicando el valor de PID en los cuales estos son transmitidos. Los PID de las tablas PMT son indicados en la tabla PAT, pero cada una de ellas debe tener 0x02 como valor en el campo `table_id`.

Cuando en un paquete TS comienza una sección PAT o PMT (`payload_unit_start_indicator = 1`), deberá ser leído el campo `pointer_field`, el cual indica donde esta nueva sección comienza dentro del paquete. Si el paquete no contiene campo de adaptación, el `pointer_field` será el quinto byte del paquete. Si el paquete no contiene un remanente de una sección anterior, el valor del `pointer_field` será 0x00, indicando que la nueva sección comienza inmediatamente después de este campo [5].

Las figuras 4 y 5 muestran las estructuras de las secciones de las tablas PAT y PMT respectivamente. Ambas secciones cuentan con una cabecera compuesta por campos similares: `table_id`, que identifica el tipo de sección que transporta el paquete; `section_syntax_indicator`, campo que siempre tiene valor “1” cuando se transporta una sección; `section_length`, especifica el tamaño de la sección (bytes que siguen después de este campo); `version_number`, indica el número de versión de la sección; `current_next_indicator`, especifica si la sección es la actualmente aplicable (“1”) o será la próxima en ser aplicada (“0”); `section_number`, indica el número de la sección que está siendo procesada; `last_section_number`, especifica el número de la última sección que tendrá que ser recibida para completar la tabla PAT; CRC32, es un código de redundancia cíclica para chequear la integridad de los datos de la sección. Los datos de la tabla PAT siguen la estructura Programa 0 – PID de la NIT, Programa i – PID de su PMT [5].

⁵ PAT: *Program Association Table*. Tabla de Asociación de Programa.

⁶ PMT: *Program Map Table*. Tabla de Mapa de Programa.

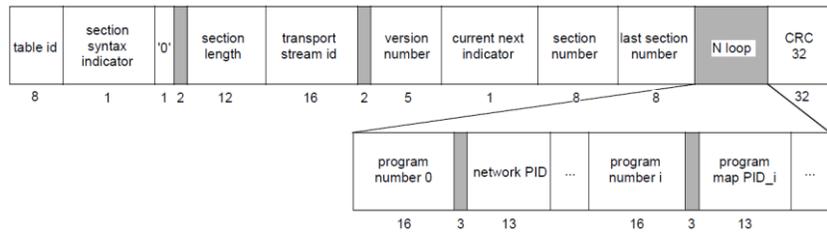


Figura 4: Estructura de la PAT.

En el caso de la tabla PAT, la cabecera incluye el campo `transport_stream_id`, el cual especifica que trama de transporte esta tabla describe. Por su parte, la tabla PMT incluye los campos `program_number`, que especifica el número de programa al que pertenece esta sección; `PCR_PID`, campo que de estar presente indicará el PID de los paquetes que transportan la referencia de reloj de este programa; `program_info_length`, especifica el tamaño en bytes del conjunto de descriptores que siguen a este campo [5]. Los datos de una sección PMT comienza con el campo `stream_type`, que especifica el tipo de elemento presente en el programa (video, audio o datos privados); `elementary_PID`, indica el PID dónde ese elemento es transportado; `ES_info_length`, especifica el tamaño en bytes de los descriptores de este elemento que siguen a este campo.

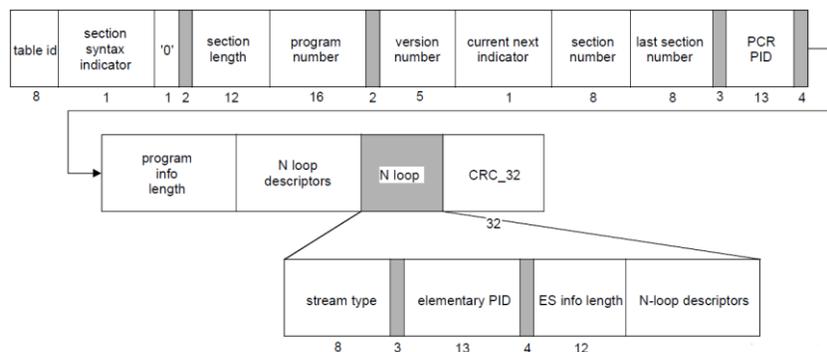


Figura 5: Estructura de la PMT

El algoritmo seguido durante el desarrollo de este módulo es mostrado en la Figura 6. Una vez establecido el sincronismo con la trama TS (bandera Sincro activada), el software procede a filtrar los paquetes con PID 0x0000 (tabla PAT) y que tengan valor "1" en el campo `payload_unit_start_indicator` (indicando comienzo de sección). Una vez encontrado un paquete que cumpla el criterio anterior, la herramienta analiza dónde comienza la cabecera de la sección PAT, teniendo en cuenta para ello, si el paquete contiene campo de adaptación y el valor del campo `pointer_field` (ver Figura 6) [5].

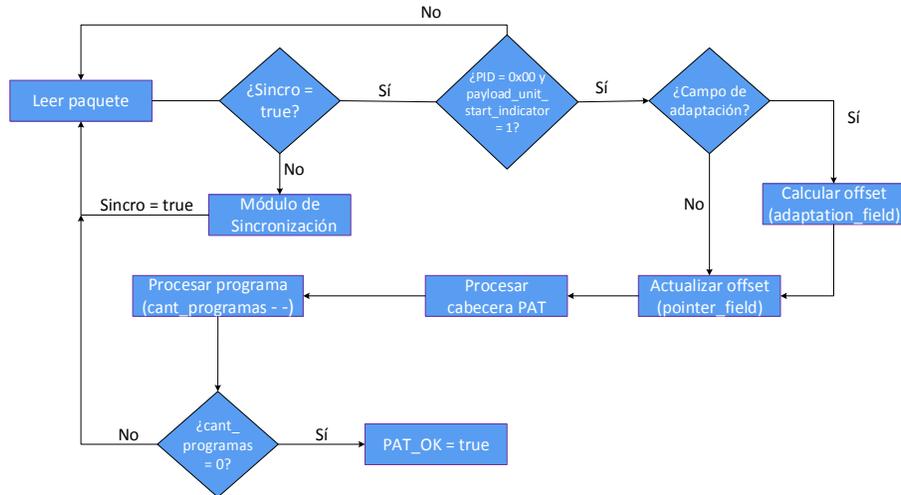


Figura 6: Procedimiento para procesar la PAT

Una vez conocido dónde la cabecera comienza, el software procede a procesarla y determinar cuántas PMT de programas son transmitidas en su carga útil, para posteriormente registrar cada par program_number - PID. Los dos bytes que conforman el campo program_number se encuentran siempre en las posiciones (offset + 1 + 8 bytes cabecera PAT + (4 * i)) y (offset + 9 bytes cabecera PAT + 1 + (4 * i)) respectivamente, donde “i” es un entero que recorre los valores entre 0 y cantidad de programas menos 1, dado que la obtención de estos campos es un proceso iterativo. El campo PID se obtiene de los dos bytes siguientes a cada program_number. Al concluir el procesamiento de la sección PAT, la herramienta activa una bandera (PAT_OK) indicando el correcto procesamiento de esta tabla.

El análisis de las distintas PMT comienza con la activación de la bandera PAT_OK. Como se observa en el algoritmo implementado para el desarrollado de esta parte del módulo (ver Figura 7), el software comienza un proceso de filtrado de paquetes, utilizando para este fin los PID registrados durante el procesamiento de la tabla PAT. Al encontrar uno de los paquetes a procesar, la herramienta procede a verificar si en el mismo comienza la sección PMT (payload_unit_start_indicator = “1”), pues en caso contrario (payload_unit_start_indicator = “0”), el paquete es descartado.

Una vez encontrado el paquete dónde comienza la sección PMT, la herramienta calcula dónde inicia su cabecera, aplicando para ello el método descrito en la Figura 7. Al concluir el procesamiento de la cabecera, el software comienza el análisis de los diferentes flujos elementales que conforman el programa. Para tal fin, se calcula el tamaño en bytes que contiene la carga útil de la sección (section_length – 9 bytes de cabecera PMT – valor del program_info_length – 4 bytes CRC). Conociendo el tamaño de la carga útil, se procesan los campos stream_type, elementary_PID y ES_info_length (4 bytes en total) de cada elemento que conforma el programa.

Durante el procesamiento de cada elemento se indica su tipo, y es registrado el PID donde es transmitido, para en caso de corresponder a subtítulos, poder ser filtrado en el módulo “Extracción PES de subtítulo”. Los elementos que corresponden a subtítulos tiene valor 0x06 en el campo stream_type y los bytes que siguen al ES_info_length corresponden al descriptor de subtítulos, permitiendo identificar

el idioma del subtítulo asociado. Este descriptor está presente cuando el campo descriptor_tag tiene valor 0x59; el idioma del subtítulo es transportado en el campo ISO_639_language_code, campo constituido de 3 bytes, donde cada uno de ellos representa una letra de acuerdo al estándar ISO 639-2.

El procesamiento de una sección PMT concluye cuando la carga útil – [4 + valor ES_info_length] = 0, pues un valor distinto de cero después de esta operación indica que quedan por procesar bytes de la carga útil, es decir, que el programa cuenta con al menos otro flujo elemental. El procesamiento de todas las secciones PMT transportadas en los PID registrados durante el análisis de la sección PAT, es indicado mediante la activación de la bandera PMT_OK.

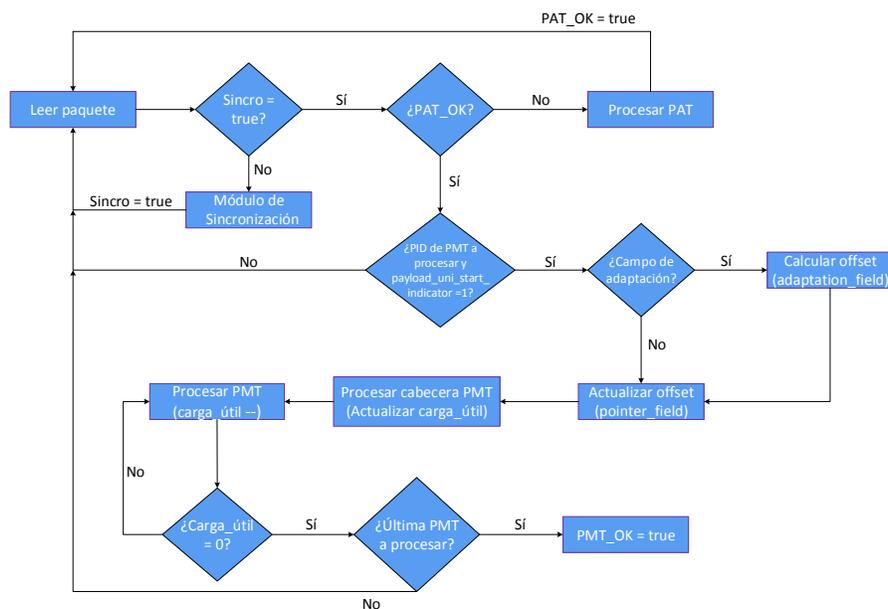


Figura 7: Procedimiento para procesar la PMT

Módulo “Extracción PES de subtítulos”

El módulo “Extracción PES de subtítulos” es el encargado de, una vez conocido los distintos programas que contienen subtítulos DVB, poder extraerlos para su posterior procesamiento. Con tal fin, el módulo brindará al usuario la posibilidad de elegir el programa del cual quiere procesar los subtítulos. Una vez que el usuario haya seleccionado el subtítulo a procesar, el módulo realiza una búsqueda de todos los paquetes que contienen el subtítulo seleccionado, registrando el flujo PES del mismo para su posterior procesamiento.

Los paquetes PES se forman a partir de las tramas elementales (ES) de video, audio y datos que componen un determinado programa. Su estructura es mostrada en la Figura 8. El paquete PES puede tener una longitud máxima de 64 KBytes y comienza con una cabecera de 6 bytes, la cual indica su inicio mediante el campo packet_start_code_prefix (3 bytes), cuyo valor siempre es “00 00 01”. A continuación, le sigue el campo stream_id, identificando el tipo de dato que contiene el mismo (video,

audio o datos). Le siguen dos bytes (campo PES_packet_length) indicando la cantidad de bytes que siguen a este campo.

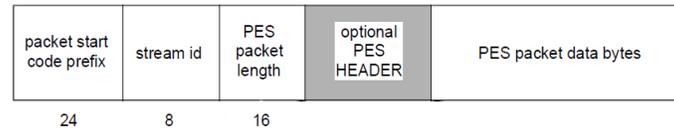


Figura 8: Estructura de un paquete PES

Dentro de la cabecera opcional, son transmitidas las estampillas PTS Y DTS, que por lo general no se encuentran en todos los paquetes. Estos campos transportan la información necesaria para realizar una correcta sincronización entre los flujos de video y audio, así como de los subtítulos. La estampilla PTS es una referencia temporal del orden de presentación de los fotogramas, mientras que la DTS es una referencia del orden de codificación, es decir, la PTS determina cuándo la imagen asociada debe ser presentada en la pantalla mientras que la DTS indica cuándo debe ser decodificada [6]. Los valores de PTS y de DTS no requieren enviarse en cada paquete PES, pero si con un período de tiempo no inferior a 0.1s. El campo data_aligment_indicator, presente también dentro de la cabecera opcional, indica la manera en que los datos son alineados dentro del paquete PES [7]. Teniendo en cuenta la estructura del paquete PES de subtítulos, así como las restricciones aplicadas a estos, el desarrollo de este módulo debe seguir el algoritmo descrito a continuación.

Una vez concluido el procesamiento de las secciones PMT (bandera PMT_OK activada), la herramienta se queda en espera de que el usuario seleccione el subtítulo a modificar. Entonces, seleccionado el subtítulo que se desea modificar, se realiza un filtrado de paquetes buscando aquellos del PID seleccionado y dónde se inicie el paquete PES de subtítulo (payload_unit_start_indicator = "1"), dado que en este caso se transporta el subtitling_segment, dentro del cual se encuentra es campo segment_type, utilizado para definir el tipo de segmento. Este paso resulta de vital importancia para el procedimiento en el último módulo. Posteriormente es comprobado si el paquete contiene campo de adaptación, para de esta forma conocer dónde comienza la cabecera del paquete PES. Una vez procesada la misma, se calcula el tamaño total en bytes del paquete PES, lo que posibilita conocer cuándo han sido registrados todos los bytes que conforman el mismo (PES_length = 0), en cuyo caso es activada la bandera PES_OK.

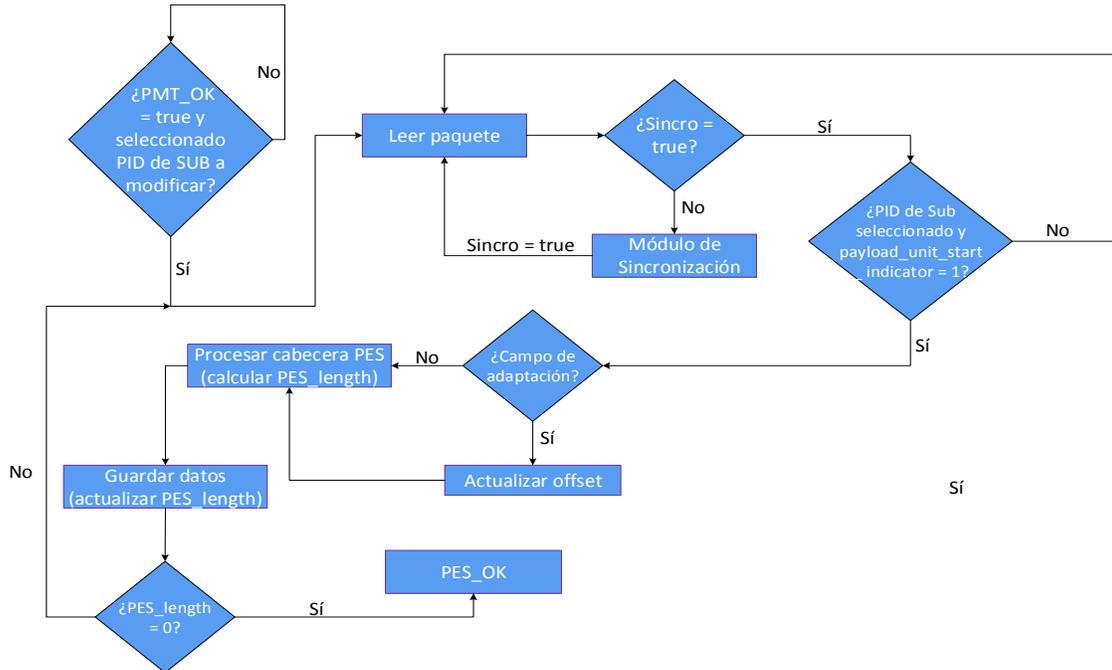


Figura 9: Procedimiento para extraer de PES de subtítulos

Módulo “Modificación PES de subtítulo”

El módulo “Modificación PES de subtítulo” es el encargado de realizar las modificaciones elegidas por el usuario. Estas pueden ser cambiar la posición de los mismos, el color del texto o del fondo, pues la fuente y el tamaño de los subtítulos son definidos por los equipos receptores. Todos los cambios son realizados al flujo PES extraído en el módulo “Extracción PES de subtítulos” y, una vez realizados, este módulo es el encargado de reinsertarlo en la trama TS original. A continuación, se describen los procedimientos generales para realizar ambas modificaciones.

La información referente a la posición de los subtítulos es transportada dentro del segmento de definición de presentación (DDS). La ausencia del segmento DDS implica que la trama se codifica en concordancia con EN 300 743 (v1.2.1), asumiendo la presentación en pantalla como 720 píxeles de ancho por 576 líneas de altura [5]. En el caso de los TS que no contienen segmentos DDS7, la posición en que se muestran los subtítulos depende del segmento de composición de página PCS8. Para realizar una correcta modificación de la posición de los subtítulos una vez conocida la estructura del PCS, basta con modificar los campos `region_vertical_address` y `region_horizontal_address`, de dos bytes cada uno. El primero define la posición vertical en pantalla del bit superior, mientras que el campo `region_horizontal_address` define el bit superior más a la izquierda de la región. El algoritmo propuesto para implementar tal funcionalidad es mostrado en la Figura 10.

⁷ DDS: *Display Definition Segment*. Segmento de Definición de Presentación.

⁸ PCS: *Page Composition Segment*. Segmento de Composición de Página.

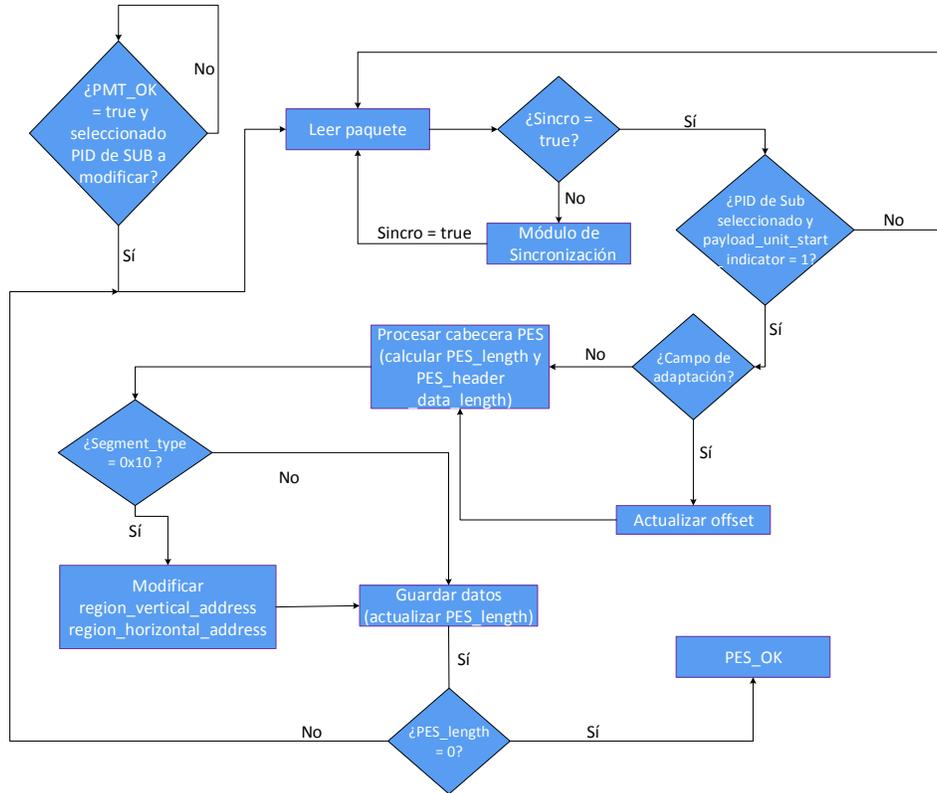


Figura 10: Procedimiento para modificar subtítulos modificando el PCS

El color con que se muestran los subtítulos depende del segmento CLUT⁹ asociado a los mismos. Los colores que se aplican a la familia de CLUTs son transportados en el campo CLUT_definition_segment. En la Figura 11 se muestra el procedimiento para realizar las modificaciones concernientes al color de los subtítulos (Este módulo no está implementado en el software hasta el momento). Partiendo de que en el RCS se encuentra 1 byte de CLUT-id correspondiente a los subtítulos, se comienza por identificar el CLUT-id de los subtítulos a los que se le desea modificar el color. Para conocer a qué está asociado determinado CLUT, se verifica el campo (de dos bits) object_type. El mismo, cuando tiene valor 0x01 o 0x02 indica que se refiere a un carácter o a una cadena de caracteres, respectivamente. En tales casos, se deben modificar los campos foreground_pixel_code y background_pixel_code, los cuales indican los colores de primer plano y de fondo, respectivamente. En caso de que el object_type sea 0x00 (0x03 está reservado) quiere decir que se refiere a mapa de bits, por lo que el color se especifica directamente con los campos Y_value, Cr_value, Cb_value y T_value en concordancia con la Recomendación BT.654-4 de la UIT-R.

⁹ CLUT: *Colour Look-Up Table*. Tabla de Búsqueda de Colores.

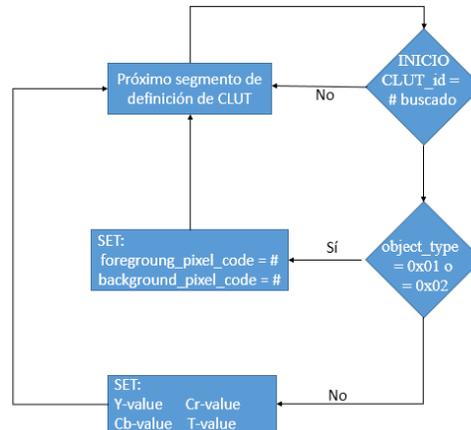


Figura 11: Procedimiento para modificar el color de los subtítulos

RESULTADOS

Los algoritmos propuestos en el capítulo anterior fueron implementados mediante el desarrollo de una herramienta en lenguaje C. El ambiente de diseño utilizado fue la plataforma Visual Studio 2008, debido a las potencialidades que este brinda. Además, esta plataforma permite generar un ejecutable, el cual posibilita realizar el procesamiento de un TS que se encuentre en la misma ubicación siempre y cuando el archivo tenga como nombre stream.ts. Si se desea realizar el análisis de algún TS, solo se debe ubicar un TS con nombre stream.ts junto al ejecutable, ejecutarlo, y será mostrado en consola la estructura de las tablas PAT y PMT, además de algunos campos de interés de los mismos. También se genera un archivo de texto en formato .txt con el nombre TABLAS, el cual contiene el análisis de las tablas PAT y PMT tal cual se muestra en consola, con el objetivo de facilitar el análisis de los resultados obtenidos una vez cerrado el software. Además, se exporta un instalador del software, el cual al instalarlo arroja como resultado un ejecutable portable, el cual realiza exactamente lo mismo que el proyecto en Visual Studio.

Con el objetivo de comprobar el correcto funcionamiento de los bloques desarrollados fueron realizadas varias simulaciones, y con la ayuda de los softwares MPEG TS Utils, 4T2 Content Analyzer, TSR y MPEG-2 TS Packet Analyser, fueron verificados los resultados obtenidos de las mismas. Se realizaron varias simulaciones con diferentes tramas. Se utilizó una trama de una obtenida de una transmisión real de la Televisión Cubana, la cual contiene 17 programas (8 de televisión y 9 de radio) y además contiene subtítulos en uno de ellos, en el programa Cubavisión. Se utilizó además una trama de una transmisión real de la empresa televisiva británica BBC. Dicho TS contiene 6 programas, de los cuales 4 contienen subtítulos.

Como resultado de la simulación con la última trama mencionada se genera un TS que contiene la misma información que el TS sin modificar, salvo que la posición de los subtítulos que viajan en el PID 0x26F. El TS generado (con el nombre SUB_PES) fue utilizado para verificar si los receptores digitales mostraban correctamente la posición de los subtítulos, lo que posibilita comprobar si el procesamiento relacionado a esta característica es correcto o no.

El TS generado se probó en cada uno de los equipos receptores de la TV Digital comercializados en el país hasta la fecha. Los resultados se resumen en la Tabla 2. La Figura 12 muestra una de las modificaciones realizadas con el software desarrollado.



Figura 12: Trama obtenida mediante una simulación con la posición de los subtítulos modificada.

Se comprobó que los datos mostrados como resultado del análisis realizado de dos tramas de transporte por la herramienta creada son correctos, pues el resultado obtenido con estas es igual al mostrado por otras herramientas softwares profesionales. Además, se logró modificar la posición de los subtítulos de un TS modificando los campos `region_vertical_address` y `region_horizontal_address` dentro del PCS, posibilitando evaluar si los distintos receptores digitales comercializados en el país cumplían con la posición indicada en el flujo de subtítulo.

Tabla 2: Comprobación de TS obtenido en equipos comercializados en el país.

Tipo de receptor	Equipo Receptor	Modelo	Muestra correctamente los subtítulos
S	KONKA	KSDT863	Sí
	SOYEA	SDP160	Sí
T	GELECT	SD-HL215	Sí
	RUNCH	DTT1513	Sí
T	MICO	DT46-N03	Sí
	MICO	DT25-2080	Sí
P	HAIER	HDMB-2000T/M	Sí
	KONKA	KHDT875-A	Sí
B	Real TV	HMA1	Sí
	RUNCH	DTT1900	Sí

X	SOYEA	HDP160	Sí
	KONKA	KHDT875-CE	Sí
	GELECT	HD-AA1604	Sí
	GELECT	HDHL1209	Sí
T V	ATEC	32L14D	Sí
	KONKA	KDL32KT627	Sí
	SOYEA	LKK32CMB	Sí
	DAYTRON	DT32DUHB	Sí

CONCLUSIONES

Este trabajo culminó con la validación mediante simulación del diseño propuesto de los bloques de Sincronismo, Estructura PSI y Extracción PES de subtítulos en lenguaje C, logrando durante el proceso una herramienta software que permite analizar las tablas PAT y PMT de los TS e indicar cuáles de los programas transportados cuenta con subtítulos DVB. Además, se implementó el cuarto módulo propuesto, Modificación PES de subtítulo (para el caso en que contengan PCS), logrando modificar la posición de los subtítulos del programa elegido por el usuario, lo que posibilitó sentar las bases para crear TS que permitan explotar todas las funcionalidades de los subtítulos DVB.

Con este propósito se profundizó en los estándares ISO/IEC 13818 o ITU-T REC.H.222.0 Parte 1, ETSI EN 300 743 o Digital Video Broadcasting (DVB) Subtitling systems, y ETSI EN 300 468 o Digital Video Broadcasting (DVB) Specification for Service Information in DVB systems, que sirvieron de punto de partida para dominar la estructura de los TS contenedores de subtítulos DVB. Se logró la definición del sistema a partir de la descripción de los módulos principales que forman parte del diagrama en bloque propuesto, sentando las bases para el desarrollo del proyecto de obtención. Esto se logra mediante una herramienta software de TS que permitan explotar las funcionalidades de los subtítulos DVB. Además, se implementó para un caso particular el módulo Modificación de PES de Subtítulos, lo cual constituye un resultado útil en el proceso de evaluación de los receptores digitales. La herramienta software creada, incluso cuando pudiera mejorarse, resulta muy útil para el trabajo con tramas de transporte toda vez que realiza un análisis exhaustivo de las mismas, y recopila los datos en un archivo de texto, lo cual facilita su posterior utilización.

RECONOCIMIENTOS

Se desea agradecer al tutor del presente trabajo Ing. Jorge Rodríguez Rodríguez, pieza clave para alcanzar el resultado obtenido en el presente trabajo, sin su guía y apoyo no hubiese sido posible.

REFERENCIAS

1. A. Delgado “Flujos de Programa y de Transporte MPEG”. *Electromagnetismo y Teoría de Circuitos*, 2001.
2. E. Cabrera, J. Rodríguez “Desarrollo de Módulos para un Re-multiplexor de Tramas de Transporte MPEG-2 sobre FPGA”. Tesis de diploma, Instituto Superior Politécnico José Antonio Echeverría, La Habana, 2012.
3. D. Huayhualla “Operación de Plataforma de Televisión Digital Satelital Regional”. Tesis de Diploma, Universidad Ricardo Palma, Lima, 2013.
4. A. Lozano “Sistema para el alineamiento de subtítulos y audio en escenarios de rehablado en televisión”. Tesis de diploma, Universidad de Madrid, España, 2013.
5. ETSI. “Digital Video Broadcasting (DVB); Subtitle System v:1.5.1”. 2014.
6. C. Mollat “Normas ISO de Codificación de Contenidos Audiovisuales”. 2013
7. P. A. Campos “Estudio del estándar de televisión digital DTMB y propuesta de reglamento para la prestación del servicio de televisión digital terrestre en el Ecuador”. Quito, Ecuador, 2010.