

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y  
ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA  
Y  
TELECOMUNICACIONES

PROYECTO DE GRADO PARA LA OBTENCIÓN  
DEL  
TÍTULO DE INGENIERÍA

ANÁLISIS DEL TRANSPORT STREAM PARA  
EL ESTÁNDAR DE TELEVISIÓN DIGITAL  
ISDB-TB

NELSON RICARDO GRANJA TOLEDO

SANGOLQUÍ - ECUADOR

2011

## CERTIFICACIÓN

Certificamos que el presente proyecto de grado titulado: “ANÁLISIS DEL TRANSPORT STREAM PARA EL ESTÁNDAR DE TELEVISIÓN DIGITAL ISDB-TB”, ha sido desarrollado en su totalidad por el señor NELSON RUCARDO GRANJA TOLEDO con CI: 172139244-5, bajo nuestra dirección.

Atentamente

---

Ing. Gonzalo Olmedo  
DIRECTOR

---

Ing. Freddy Acosta  
CODIRECTOR

## Resumen

En el presente proyecto de grado se presenta un análisis sobre la estructura que debe tener el Transport Stream para la transmisión de datos de acuerdo al estándar de Televisión Digital ISDB-Tb, además se describen todos los servicios, como interactividad, Guías de programación electrónicas, closed caption, etc., que se pueden agregar en el proceso de transmisión de las señales.

Inicialmente se recopilan las características y ventajas que presenta la Televisión Digital, los estándares existentes a nivel mundial y bajo que proceso el Ecuador decidió adoptar el estándar ISDB-Tb.

Posteriormente se analiza la estructura general que posee el estándar ISDB-Tb y las diferencias y similitudes que tiene comparado con el estándar japonés. Dentro de este análisis se estudia el modelo de transmisión y codificación de datos que presenta el estándar. Además se analiza las características que deben tener el video, el audio y los datos para que sean compatibles con el protocolo de multiplexación de las señales. Esta información debe ser codificada, empaquetada y dentro de este proceso se agregan cabeceras que permiten la generación de un flujo único de transporte que debe ser interpretado correctamente por el decodificador.

Finalmente se genera un Transport Stream que contenga un servicio de Televisión Digital, mediante el software *OpenCaster* y se detallan todas las configuraciones que se deben realizar para cumplir con este proceso.

## DEDICATORIA

*A mi amigo y mejor maestro, que me dijo  
que nunca crea en la suerte. Por enseñarme  
a encontrar la comunión entre la ciencia y  
el arte. Por construir mi puño lleno de todo  
tu optimismo, porque no querías que sea  
solo un Ingeniero, tú querías un alegre  
caballero. A ti, por dejarme tu espíritu en  
mi corazón.*

*A la mujer maravillosa que me dio la vida y  
me enseñó que la verdadera sabiduría no se  
encuentra en las aulas. Por enseñarme a  
creer en mí y que el amor es la ciencia que  
construye la felicidad.*

## AGRADECIMIENTO

Agradezco a Dios por haberme dado la vida y la salud necesaria para alcanzar esta meta, poniendo en mi camino a gente maravillosa que me ha ayudado a lo largo de estos años. Gracias por permitir que la Fe me acompañe hasta en los momentos más duros de mi vida.

Gracias a mi Padre, por ser el brazo firme que me guió a lo largo de todos mis estudios, por ser mi ejemplo, mi inspiración, mi mejor maestro. Ahora desde el cielo, eres la esencia que mueve todos mis sueños.

Gracias a mi Mamá, por ser ese apoyo incondicional a todas mis locuras, por escucharme siempre que lo necesito y por cada uno de tus consejos. Por la alegría y el consuelo que solo expresa el amor de una madre.

Gracias a mis hermanas, por ese cariño casi maternal que me han tenido siempre, por ser mis amigas, mi ejemplo y por cultivar ese espíritu inagotable de mantenernos unidos como mosqueteros, como una gran familia.

Gracias a mi tía, por ser ese arroyito de letras que consiente mis ideas y entrega más de lo que puede por sacarme una sonrisa.

Gracias a Sharon, Andrés y Luis, por años de una amistad incondicional y sincera, por apoyarme y cuidarme siempre. Gracias, por ser mis hermanos.

Gracias a todos mis amigos, por tantos años que compartimos llenos de momentos tristes, felices, con sueño, de todo un poco. Por toda su ayuda, enteramente Gracias y recuerden que cuentan conmigo para siempre.

Gracias a todos los buenos profesores que entregaron su tiempo para compartir su conocimiento y su experiencia al servicio de la enseñanza, porque con su esfuerzo crean un mejor Ecuador. Pero recuerden que tienen una gran responsabilidad en sus manos.

Una etapa hermosa de mi vida ha terminado y ahora es tiempo de seguir adelante y demostrar todo lo que he aprendido en estos años. Me llevo conmigo no solo el conocimiento ni un título que preceda mi nombre, realmente eso es lo de menos, sino valores bien definidos, un carácter de lucha y esfuerzo, horizontes de largo alcance y lazos de amistad inquebrantables. Ahora sí, estamos listos para conquistar el mundo.

## PRÓLOGO

La digitalización como concepto ha abarcado todo el amplio abanico de la tecnología y la televisión no ha estado exenta de ello. La Televisión Digital Terrestre (TDT) combina la televisión digital con la transmisión terrestre, emitiendo por los canales de la TV convencional analógica. La novedad de la TDT es su difusión a través de ondas hertzianas terrestres y, además, es gratuita, en el sentido que no hay que abonar ninguna cuota mensual para acceder a su oferta.

La TDT consigue aumentar la oferta de programas, mejorar la calidad de imagen y sonido, facilitar la recepción, ampliar la interactividad, ofrecer aplicaciones multimedia y permitir programas con diferentes coberturas: nacional, autonómica y local. Uno de sus principales atractivos es la interactividad, con la cual se facilita la interacción de los usuarios con la programación y además ofrece servicios y aplicaciones útiles para el comercio, la información, el aprendizaje entre otros.

Con este antecedente se observa que la TDT comprende una innovación tecnológica muy importante pero que aún presenta muchos retos e investigaciones que hay que desarrollar para que este proyecto pueda volverse una realidad. Como parte de este proceso, encontramos la etapa de multiplexación de las señales. El estándar de ISDB-Tb utiliza MPEG-2 para la generación del flujo de transporte que transporta las señales de video, audio y datos. Este flujo, conocido mejor como “Transport Stream”, es un protocolo de comunicación para audio, vídeo y datos que se comprimen en cada programa independientemente formando cada uno de ellos una “corriente elemental”. Cada una de estas corrientes elementales se estructura en forma de paquetes llamados PES (Packetized Elementary Stream). Estos paquetes de video y audio, así como de otros datos de un mismo programa, pasan posteriormente a un multiplexor donde se conforma un solo tren binario.

En este proceso de transmisión de datos se debe tener claramente establecido cómo deben ir empaquetadas y ordenadas las tramas de los datos, tomando

en cuenta que este estándar utiliza un nuevo protocolo de codificación de datos y un nuevo middleware para desarrollo de aplicaciones interactivas. Caso contrario el desarrollo de software para televisión digital y los proyectos en los que se viene trabajando al respecto no podrían verse integrados y aprovechados, quedando únicamente como pruebas aisladas y carentes de la posibilidad de una implementación real dentro de la red entera que encierra la Televisión Digital. El análisis del Transport Stream comprende una etapa fundamental ya que es el nexo que integra las etapas de codificación, desarrollo de aplicaciones y la transmisión de las señales.

## ÍNDICE DE CONTENIDOS

### CAPÍTULO 1

<b>LA TELEVISIÓN DIGITAL .....</b>	<b>1</b>
1.1 GENERALIDADES .....	1
1.1.1 Antecedentes .....	1
1.1.2 Definición .....	2
1.1.3 Características generales de TDT .....	2
1.2 RESUMEN HISTÓRICO .....	4
1.2.1 Antecedentes Internacionales .....	4
1.2.2 Televisión Digital Terrestre .....	5
1.3 ESTÁNDARES DE TV DIGITAL .....	6
1.3.1 Sistema Americano ATSC .....	6
1.3.2 Sistema Europeo DVB .....	7
1.3.3 Sistema ISDBT .....	8
1.3.4 Sistema DTMB .....	9
1.4 LA TV DIGITAL EN EL ECUADOR .....	10
1.4.1 Definición y antecedentes .....	10
1.4.2 Pruebas y Mediciones .....	12
1.4.2.1 Pruebas de evaluación subjetiva .....	12
1.4.2.2 Pruebas de cobertura .....	14
1.4.2.3 Ejecución e inicio de pruebas objetivas .....	15
1.4.2.4 Ejecución e inicio de pruebas subjetivas .....	17
1.4.2.5 Pruebas de cobertura .....	19
1.4.2.6 Pruebas de recepción portátil, peatonal y personal .....	19
1.4.2.7 Metodología de evaluación .....	20
1.4.2.8 Criterios de ponderación de las mediciones y resultados .....	20
1.4.3 Selección del estándar .....	24
1.5 SITUACIÓN ACTUAL .....	25
1.6 ESTÁNDAR ISDB-Tb .....	27
1.6.1 Estructura general .....	27
1.6.2 Organización del Espectro Radioeléctrico .....	30
1.6.3 Sub-Sistema de Modulación .....	31
1.6.3.1 Transmisión Jerárquica .....	32



1.6.3.2	Modos de Transmisión y Segmentos de datos .....	33
1.6.3.3	Tasa útil de transmisión.....	35
1.6.4	Formato de compresión de audio y video.....	38
1.6.4.1	Sistema de compresión de audio.....	38
1.6.4.2	Sistema de compresión de video.....	39
1.6.5	Middleware y capa física .....	40
1.6.5.1	Definición de Middleware.....	40
1.6.5.2	Middleware GINGA para ISDB-TB.....	41
1.6.5.3	Características de GINGA .....	41
1.6.5.4	Capa Física.....	43
1.6.5.5	Ginga-NCL .....	43
1.6.5.6	Ginga-J.....	44
1.6.5.7	Ginga-CC .....	45

## CAPÍTULO 2

<b>TRANSPORT STREAM .....</b>	<b>46</b>
2.1 TRANSPORT STREAM PARA ISDB-T.....	46
2.1.1 Parámetros básicos de transmisión para ISDB-T .....	47
2.2 TRANSPORT STREAM PARA ISDB-Tb.....	48
2.2.1 Multiplexación y formación del TRANSPORT STREAM.....	48
2.2.1.1 Packetized elementary Stream .....	49
2.2.1.2 Program Stream .....	51
2.2.1.3 Program Stream y Transport Stream.....	51
2.2.2 Definición del Transport Stream .....	52
2.2.3 Sincronización del TRANSPORT STREAM .....	54
2.2.4 Remultiplexación y generación del MPEG-2 Transport Stream .....	54
2.3 ESTRUCTURA DEL TRANSPORT STREAM .....	55
2.3.1 Estructura del Transport Stream para el contenido de video.....	55
2.3.1.1 Características de la señal de video.....	55
2.3.1.2 Parámetros de la señal de video.....	57
2.3.1.3 Codificación de la señal de video .....	59
2.3.2 Estructura del Transport Stream para el contenido de audio.....	66
2.3.2.1 Características de la señal de audio.....	66
Parámetros de la señal de audio .....	67

2.3.2.2	Compresión y codificación de la señal de audio .....	68
2.3.2.3	Transporte y multiplexación de la señal de audio .....	71
2.3.3	Estructura del Transport Stream para el contenido de datos.....	73
2.3.3.1	Multiplexación de datos .....	73
2.3.4	Tipos de especificación de transmisión de datos.....	76
2.3.4.1	PES independiente.....	77
2.3.4.2	Carrusel de datos /objetos.....	78
2.3.4.3	Mensajes de eventos .....	80
2.3.4.4	Data Piping.....	81
2.3.4.5	Encapsulado Multiprotocolo .....	81
2.4	ESTRUCTURA GENERAL DEL TRANSPORT STREAM .....	82
2.4.1	Estructura de las Señales Codificadas .....	82
2.4.1.1	Sección.....	82
2.4.1.2	PES.....	84
2.4.1.3	Señales de control de transmisión.....	86
2.4.1.3.1	Tabla de asociación de Programa (PAT) .....	87
2.4.1.3.2	Tabla de mapeo de programa (PMT) .....	89
2.4.1.3.3	Tabla de acceso condicional (CAT).....	92
2.4.1.3.4	Tabla de información de red (NIT) .....	92
2.4.1.3.5	Servicios de Información (SI).- .....	94
2.4.1.4	Paquete Transport Stream .....	95
2.4.2	Multiplexación de Servicios.....	99

## CAPÍTULO 3

<b>GENERACIÓN DE TRANSPORT STREAM.....</b>	<b>101</b>
3.1 SOFTWARE DE GENERACIÓN DE TRASPORT STREAM.....	101
3.1.1 Características de OpenCaster.....	101
3.1.2 Instalación de OpenCaster.....	102
3.1.3 Configuración de los parámetros para la generación del Transport Stream	104
3.1.3.1 Configuración de los parámetros y funciones útiles para la señal de video	104
3.1.3.2 Configuración de los parámetros y funciones útiles para la señal de audio	107
3.2 GENERACIÓN DE TRANSPORT STREAM.....	109

3.2.1	Generación de un Transport Stream con un servicio de TV Digital .....	110
3.2.1.1	Conversión del video original .....	110
3.2.1.2	Obtención de la información del video original .....	110
3.2.1.3	Creación de los archivos .ts de audio y video.....	112
3.2.1.4	Creación de las tablas PSI/SI.....	115
3.2.1.5	Compilación de las tablas y Remultiplexación de los flujos de transporte de audio y video .....	122

## **CAPÍTULO 4**

4.1	CONCLUSIONES .....	127
4.2	RECOMENDACIONES .....	129

## ÍNDICE DE TABLAS

Tabla 1. 1 Evaluación subjetiva de imagen y sonido .....	13
Tabla 1. 2 Cronograma de pruebas de la SUPERTEL .....	16
Tabla 1. 3 Parámetros de configuración de cada estándar de TV digital .....	16
Tabla 1. 4 Canal utilizado por cada estándar .....	17
Tabla 1. 5 Criterios de ponderaciones de las mediciones .....	20
Tabla 1. 6 Resultados de las mediciones y evaluaciones de cada estándar....	22
Tabla 1. 7 Resultados de las mediciones y evaluaciones de los estándares con compresión MPEG-4 .....	23
Tabla 1. 8 Grado de importancia de los aspectos de evaluación .....	24
Tabla 1. 9 Resultados de las pruebas de cada estándar.....	25
Tabla 1. 10 Principales parámetros del sub sistema de transmisión del estándar ISDB-Tb.....	34
Tabla 1. 11 Tasa útil de datos de un segmento.....	37
Tabla 2. 1 Ecuaciones de las señales analógicas y de los complementos de color.....	56
Tabla 2. 2 Parámetros de la señal de video .....	58
Tabla 2. 3 Parámetros de la señal de video .....	58
Tabla 2. 4 Modos de Audio permitidos para el estándar ISDB-Tb.....	70
Tabla 2. 5 Principales parámetros LATM a ser configurados. ....	72
Tabla 2. 6 Table id.....	83
Tabla 2. 7 Stream ID .....	85
Tabla 2. 8 Ejemplo de una Tabla PAT .....	89
Tabla 2. 9 Ejemplo de una Tabla PMT .....	91
Tabla 2. 10 Ejemplo de tabla CAT.....	94
Tabla 2. 11 Identificadores y sus funciones.....	95
Tabla 2. 12 Atribución del PID.....	97
Tabla 2. 13 Valores de control de campo adaptable .....	98
Tabla 3. 1 Configuración de los flujos elementales	124
Tabla 3. 2 Configuración de los Flujos Elementales Paquetizados .....	125
Tabla 3. 3 Configuración de los Flujos de Transporte .....	125
Tabla 3. 4 Configuración de las tablas de PSI/SI .....	126

## ÍNDICE DE FIGURAS

Figura 1. 1 Sistema DVB .....	7
Figura 1. 2 Sistema ISDB .....	9
Figura 1. 3 Ubicación de los emplazamientos para pruebas de cobertura .....	14
Figura 1. 4 Disposición del equipo para las pruebas de campo .....	15
Figura 1. 5 Diagrama del sistema de Transmisión .....	17
Figura 1. 6 Estructura de la presentación de una sesión de prueba .....	18
Figura 1. 7 Resultados finales de las pruebas técnicas con estándares que utilizaron compresión MPEG-2 .....	22
Figura 1. 8 Resultados finales de las pruebas técnicas con estándares que utilizaron compresión MPEG-4 .....	23
Figura 1. 9 Estructura del sistema ISDB-TB .....	27
Figura 1. 10 Sistema General de transmisión de ISDB-Tb .....	29
Figura 1. 11 Sistema básico ISDB-TB .....	30
Figura 1.12 Organización del canal en segmentos .....	31
Figura 1. 13 Formación de las capas .....	32
Figura 1. 14 Orden de los segmentos en la banda de 6MHz .....	32
Figura 1. 15 Distribución de los segmentos en las capas .....	33
Figura 1. 16 Estándares de referencia utilizados por ISDB-Tb .....	38
Figura 1. 19 Subsistema de audio dentro de un sistema de televisión digital ..	39
Figura 1. 20 Equivalencia del Middleware de los estándares ATSC, ISDB-T y DVB-T con Ginga de ISDB-Tb .....	41
Figura 1. 21 Arquitectura Middleware Ginga .....	43
Figura 1. 22 Subsistema Ginga-NCL .....	43
Figura 1. 23 Middleware de estándares de TDT a nivel mundial .....	44
Figura 1. 24 Estructura del Middleware GINGA e identificación del Ginga CC	45
Figura 2. 1 Formato Multiplexado en el sistema ISDB-T .....	46
Figura 2. 2 Sistema ISDB-T .....	47
Figura 2. 3 Estructura del paquete PES .....	50
Figura 2. 4 Sistema de transporte MPEG-2 .....	51
Figura 2. 5 Paquete de TS .....	53
Figura 2. 6 Creación del Transport Stream .....	53

Figura 2. 7 Remultiplexación de paquetes TS.....	55
Figura 2. 8 Encadenamiento de las imágenes I, P y B.....	61
Figura 2. 9 Codificación inter-frame .....	62
Figura 2. 10 Codificación intra-frame .....	62
Figura 2. 11 Estructura de la trama de video.....	66
Figura 2. 12 Proceso de Compresión y codificación del audio .....	68
Figura 2. 13 Procedimiento de codificación de audio .....	69
Figura 2. 14 Transporte de datos sincrónicos .....	74
Figura 2. 15 Ejemplo de servicio de transporte sincrónico .....	74
Figura 2. 16 Sistema de transporte sincronizado .....	75
Figura 2. 17 Ejemplo de servicio de transporte sincronizado .....	75
Figura 2. 18 Sistema de transporte asíncrono.....	76
Figura 2. 19 Ejemplo servicio de transporte asíncrono .....	76
Figura 2. 20 Formación del Carrusel de datos dentro del Transport Stream....	78
Figura 2. 21 Formato extendido de las secciones .....	82
Figura 2. 22 Estructura del paquete PES .....	84
Figura 2. 23 Funcionamiento de la PAT .....	87
Figura 2. 24 Tabla de asociación de programa (PAT).....	87
Figura 2. 25 Estructura de la sección de la PAT .....	88
Figura 2. 26 Tabla de mapeo de programa .....	89
Figura 2. 27 Tabla de acceso condicional .....	92
Figura 2. 28 Tabla de información de red.....	92
Figura 2. 29 Modelo de Capas para la formación del TS .....	95
Figura 2. 30 Tren de paquetes de transporte .....	96
Figura 2. 31 Estructura del encabezamiento de un paquete TS.....	96
Figura 2. 32 Estructura del Campo de adaptación .....	98
Figura 2. 33 Multiplexado de flujo de datos.....	100
Figura. 3. 1 Proceso de Generación del Transport Stream	109
Figura. 3. 2 Resultado mostrado por el comando IDVID .....	111

## GLOSARIO

**AAC:** (*Advanced Audio Coding*). Codificación Avanzada de Audio.

**ABNT NBR:** Asociación Brasileira de Normas Técnicas

**ATSC:** (*Advanced Television System Committee*). Comité de Sistema de Television Avanzado.

**AU:** (*Access Unit*). Unidad de Acceso.

**CAT:** (*Conditional Access Table*). Tabla de Acceso Condicional.

**COFDM:** (*Coded Orthogonal Frequency Division Multiplexing*) Multiplexación por División de Frecuencia Ortogonal Codificada.

**DCT:** (*Discrete Cosine Transform*). Transformada de Coseno Discreto.

**DDB:** (*Download Data Block*). Download de Bloque de Datos

**DII:** (*Download Information Indicator Message*). Mensaje Indicador de Información de Download

**DSM-CC:** (*Digital Storage Media Command and Control*). Medio de Almacenamiento Digital para Control y Comando.

**DTS:** (*Decoding Time Stamp*). Marca de Tiempo de Decodificación

**DVB-T:** (*Digital Video Broadcasting – Terrestrial*). Difusión Digital de Video Terrestre.

**ECM:** (*Entitlement Control Message*). Mensaje de Control de Derecho.

**EMM:** (*Entitlement Management Message*). Mensaje de Administración de Derechos.

**ES:** (*Elementary Stream*). Flujo Elemental.

**FPS:** (*Frames per Second*). Cuadros por Segundo.

**FTP:** (*File Transfer Protocol*). Protocolo de Transferencia de Archivos.

**GINGA-CC:** (*Ginga- Common Core*). Ginga de Núcleo Común.

**GINGA-J:** Ginga Java

**GINGA-NCL:** (*Ginga – Nested Context Language*). Ginga – Lenguaje de Contextos Anidados.

**GOP:** (*Group of Pictures*). Grupo de Imágenes.

**HDTV:** (*High Definition Television*). Televisión de Alta Definición

**IDR:** (*Instantaneous Decoding Refresh*). Actualización de Decodificación Instantánea.

**ISDB-Tb:** (*International System for Digital Broadcast, Terrestrial, Brazilian version*). Transmisión Digital de Servicios Integrados, versión Brasileña.

**ISO/IEC:** (*International Standard Organization / International Electrotechnical Commission*). Organización Internacional de Normalización / Comisión Electrotécnica Internacional.

**LATM/LOAS:** (*Low overhead audio multiplex and transport mechanism*).

**LDTV:** (*Low Definition Television*). Televisión de Baja Definición.

**LIFIA:** Laboratorio de Investigación y Formación en Informática Avanzada.

**MHP:** (*Multimedia Home Platform*). Plataforma de Hogar Multimedia

**MPEG:** (*Moving Picture Experts Group*). Grupo de Expertos en Imágenes Móviles.

**NCL:** (*Nested Context Language*). Lenguaje de Contextos Anidados.

**NIT:** (*Network Information Table*). Tabla de Información de Red

**OFDM:** (*Orthogonal frequency-division multiplexing*). Multiplexación por División de Frecuencias Ortogonales.

**PAT:** (*Program Association Table*). Tabla de Asociación de Programa.

**PCR:** (*Program Clock Reference*). Referencia de Reloj de Programa.



**PES:** (*Packetized Elementary Stream*). Flujo Elemental Paquetizado.

**PID:** (*Program Identifier*). Identificador de Programa.

**PID:** (*Packet Identifier*). Identificador de Paquetes.

**PMT:** (*Program Map Table*). Tabla de Mapa de Programa.

**PS:** (*Program Stream*). Flujo de Programa.

**PTS:** (*Presentation Time Stamp*). Marca de Tiempo de Presentación.

**QAM:** (*Quadrature Amplitude Modulation*). Modulación de Amplitud en Cuadratura.

**QPSK:** (*Quadrature Phase Shift Keying*). Modulación por Desplazamiento de Fase en Cuadratura.

**SATVD:** Sistema Argentino de Televisión Digital.

**SBTVD:** Sistema Brasileiro de Televisión Digital.

**SDT:** (*Service Descriptor Table*). Tabla de Descriptor de Servicio.

**SDTV:** (*Standard Definition Television*). Televisión de Definición Estándar.

**SI / PSI:** (*Service Information / Program Specific Information*). Información de Servicio / Información Específica de Programa.

**STB:** Set top Box

**STC:** (*System Time Clock*). Reloj de Referencia del Sistema.

**TDI:** Televisión Digital Terrestre

**TMCC:** (*Transmission Multiplexing Configuration Control*). Control de Configuración de Transmisión de Multiplexación

**TS:** (*Transport Stream*). Flujo de Transporte.

**VBV:** (*Video Buffer Verifier*). Verificador del Buffer de Video.

## **CAPÍTULO I**

### **LA TELEVISIÓN DIGITAL**

#### **1.1 GENERALIDADES**

##### **1.1.1 Antecedentes**

La digitalización como concepto ha abarcado todo el amplio abanico de la tecnología y la televisión no ha estado exenta de ello.

Este proceso en la televisión comenzó con la digitalización de las transmisiones por satélite y el cable, que ahora se ha volcado hacia el servicio de libre de recepción. Todo esto tuvo sus inicios en el año 1982 cuando se desarrolló el estándar CCIR-601 (actualmente ITU-601), el cual además de ser el primer estándar internacional de Codificación de Televisión Digital para Estudio, anticipa la codificación de la señal análoga a digital por componentes a 8 bits de resolución de cuantificación. Desde entonces la tecnología no ha parado de avanzar de una manera muy acelerada, creando así canales de televisión con estudios totalmente digitales, de la misma manera algunos países desarrollaron sus propios estándares de tv digital como lo son el ATSC (Advanced Television Systems Committee) originado en Estados Unidos, DVB (Digital Video Broadcasting) de la Comunidad Europea e ISDB (Integrated Services Digital Broadcasting) de origen japonés, DTMB (Digital Terrestrial / Television Multimedia Broadcasting) de origen chino.

Asimismo, entre estos estándares de televisión digital su operación es incompatible. En la práctica, esto se traduce que si queremos que la televisión abierta sea recibida por todos, tanto el canal de televisión como los televidentes deben contar con el mismo estándar de televisión digital, tanto en la transmisión como en la recepción

Con esta digitalización de la televisión se presentan varias ventajas, entre ellas se puede mencionar la mejora de la calidad de la imagen y del sonido, el

aumento de la oferta de canales de televisión, alta definición y otras características como guías de programación e interactividad.

### **1.1.2 Definición**

La Televisión Digital Terrestre o TDT es la aplicación de las nuevas tecnologías del medio digital a la transmisión de contenidos a través de una antena convencional (aérea), combinando la televisión digital con la transmisión terrestre, emitiendo por los canales de la TV convencional analógica. En este sistema la señal es transmitida ya sea por aire, cable o satélite con una alta resolución. La novedad de la TDT es su difusión a través de ondas hertzianas terrestres y, además, es un servicio público y por consiguiente, gratuito.

Aplicando dichas tecnologías digitales se consigue un mejor uso del espectro disponible, lo que puede utilizarse para proveer un mayor número de canales, mejor calidad de imagen o imagen en alta definición y mejor calidad de sonido, facilitar la recepción, ampliar la interactividad, ofrecer aplicaciones multimedia y permitir programas con diferentes coberturas: nacional, autonómica y local. Uno de sus principales atractivos es la interactividad, con la cual se facilita la interacción de los usuarios con la programación y además ofrece servicios y aplicaciones útiles para el comercio, la información, el aprendizaje entre otros.

Por esta razón se considera a la TDT como un verdadero acontecimiento tecnológico que va a revolucionar este servicio que ha permanecido aparentemente aletargado y poco desarrollado en las últimas décadas.

### **1.1.3 Características generales de TDT**

Básicamente la TDT consiste en una nueva técnica de radiodifusión de señales que en la cual se transforma a los datos (audio, video, datos) en información digital representada por ceros y unos, contrario a lo que significaba la televisión analógica que consistía en ondas que necesitaban grandes anchos de banda y muy sensibles a interferencias que disminuían la calidad de la información. Con la información digitalizada se aprovechan recursos de procesamiento como lo son la compresión, multiplexación, codificación y

modulación digital de las señales tanto de audio, vídeo y datos, con el único fin de optimizar la transmisión de las señales de televisión.

Como se dijo anteriormente en TDT tenemos un incremento sustancial de la calidad de la imagen y el sonido, los cuales se dividen en tres categorías:

- La SD o SDTV es la televisión normal ANALÓGICA transmitida en formato DIGITAL 576i (calidad de 576 líneas).
- La HDTV o Televisión de alta definición con formato de alta 'definición' o 'resolución' que en la actualidad llega a 1080p líneas (con otros formatos intermedios).
- La LD (Low Definition), televisión que se transmite con una baja resolución para que pueda ser receptada por dispositivos móviles como celulares, PDAs, etc.

En la transmisión terrestre de televisión la señal se ve afectada por dispersión de energía, zonas de sombra y reflexiones que provocan ecos. En transmisión analógica esos problemas se manifiestan como nieve, ruido en la imagen, doubles imágenes (llamados fantasmas), colores deficientes y sonido de baja calidad. En transmisión digital, al estar la señal codificada, recibimos una imagen siempre íntegra, pero que tiene otras fallas como pixelación, congelamiento de la imagen (completa) y que incluso se acaba llegando al denominado “abismo digital”, cuando la señal no es suficiente para los circuitos decodificadores se pierde completamente la recepción. Una recepción óptima suele necesitar menor potencia de señal que una transmisión analógica de calidad normal, la imagen, sonido y datos asociados a una emisión de televisión se codifican digitalmente en formato MPEG-2 (Moving Picture Experts Group 2).

Otra característica importante de TDT es la transmisión múltiple o multicasting que permite a las estaciones de televisión transmitir múltiples canales tanto HD como SD de manera simultánea, con esto lo que se logra es que se aproveche de mejor manera el espectro y que el usuario pueda recibir, de una misma estación, varios canales al mismo tiempo y por la misma interfaz aire.

Como se mencionó en el punto anterior, una de las características fundamentales que va a revolucionar el concepto de televisión es la capacidad de que no solo se pueda enviar audio y video sino también información a través de la misma interfaz, lo que permitirá la interactividad con el usuario, dando paso a un mercado muy interesante donde el usuario pueda gozar de varios servicios que van desde los relacionados con la banca, noticias, clima, teleducación, etc.

## **1.2 RESUMEN HISTÓRICO**

### **1.2.1 Antecedentes Internacionales**

La televisión empezó a dar sus primeros pasos a finales del siglo XIX hasta 1935. En un inicio existían dos modelos: la televisión mecánica ideada por John Baird y la electrónica creada por el investigador ruso-norteamericano Vladimir Zworikyn. Rápidamente la tecnología electrónica superó a la mecánica desplazándola completamente del mercado y en muy poco tiempo pasó de ser un simple invento de laboratorio a convertirse en un medio al alcance del público. En 1931 la RCA colocó una antena emisora en la terraza del Empire State Building, el edificio más alto de Nueva York, y comenzó con sus emisiones de pruebas. Por otro lado en Inglaterra, la empresa EMI se lanzó a trabajar en la televisión electrónica. Los ingenieros de EMI realizaron una demostración a la BBC sobre su sistema televisivo.

Ante la existencia de estos dos modelos televisivos, el Gobierno británico decidió nombrar una comisión investigadora para definir la posición del Estado en materia televisiva. En enero de 1935, dicha comisión optó por la televisión electrónica debido a la superior calidad respecto al otro sistema y el 2 de noviembre de 1936, la BBC comenzó sus transmisiones desde los estudios londinenses de Alexandra Palace.

Hacia mediados de la década de 1930 las transmisiones tienden a regularizarse y a crecer en las principales urbes (Londres, Berlín, París, Nueva York).

En 1936 Alemania hicieron todos los esfuerzos por llegar a transmitir los juegos olímpicos en Berlín, logrando un gran éxito. Lamentablemente en la

segunda Guerra Mundial el desarrollo de la televisión se vio interrumpido quedando este mercado muy al margen.

Finalizada la guerra empieza el renacimiento de la televisión, extendiéndose las transmisiones por toda Europa y ganando una cantidad muy numerosa de adeptos a este nuevo servicio.

En los años 50 la televisión da un gran salto con la llegada de la televisión a color y se empiezan a realizar las primeras transmisiones en Latinoamérica, mientras que Estados Unidos se consolidaba como una poderosa industria televisiva con cerca de 108 emisoras que daban servicio a unos 21 millones de televisores, mientras que en países europeos el número de televisores estaba muy distante de esa cifra.

Décadas más tarde empieza la expansión mundial con la internacionalización de los programas ayudándose por medio de los satélites que se habían empezado a lanzar al espacio.

Debido a que existían lugares donde la señal de radio de la televisión no llegaba por cuestiones de ensombrecimiento de la señal, nace el concepto de llevar la señal de la televisión por cable en los años 40. En la década de los años 70 nace el concepto de la televisión por cable como un nuevo modelo de negocio que brinda un servicio de valor agregado al consumidor [1].

### **1.2.2 Televisión Digital Terrestre**

La generalización de los satélites, unida a la digitalización de la señales, ha producido la multiplicación exponencial de canales, la fragmentación de audiencias y el surgimiento de nuevas formas de financiación: el abono mensual y el pago por visión. A este nuevo escenario se vino a añadir, desde la segunda mitad de los años 90, el debate acerca de los cambios que traería aparejados la TDT.

Numerosos investigadores coinciden en señalar que la televisión se encuentra ante el proceso de transición tecnológica más importante de su historia, puesto que a partir de la digitalización de los sistemas de televisión (producción, emisión y recepción de programas) emerge una serie de posibilidades que puede llegar a terminar con la televisión hasta hoy conocida.

La digitalización permite, entre otras muchas cosas, incrementar notablemente el número de programas, servicios y señales actualmente disponibles y mejorar la calidad de las imágenes y del sonido. Si bien la mayor parte de los planes gubernamentales sitúa como fecha límite para el “apagón analógico” entre los años 2006 y 2012, un importante grupo de analistas califica a estas estimaciones demasiado optimistas y considera que el “apagón” definitivo se producirá en un plazo mínimo de entre 15 y 20 años.

### **1.3 ESTÁNDARES DE TV DIGITAL**

Alrededor del mundo se han desarrollado varios estándares de televisión digital, los cuales son:

- Estándar Americano, ATSC (*Advanced Television Systems Committee*).
- Estándar Europeo, DVB (*Digital Video Broadcasting*).
- Estándar Japonés, ISDB-T (*Integrated Services Digital Broadcasting*).
- Estándar Chino, DTMB (*Digital Terrestrial / Television Multimedia Broadcasting*).

#### **1.3.1 Sistema Americano ATSC**

ATSC (comité avanzado de los sistemas de la televisión) es el reemplazo digital para el estándar analógico anterior, NTSC. El estándar ATSC fue creado por el comité de dirección avanzado de la televisión. Fue desarrollado para la transmisión de señales HDTV, SDTV a una velocidad de 19.39Mbps, la difusión de los datos, el audio de varios canales y la difusión basada en los satélites. Define el contenido de la secuencia de bits, su transporte y transmisión digital en un ancho de banda de 6 MHz conservando el ancho de banda de NTSC. El sistema ATSC utiliza múltiples formatos de transmisión, compresión de audio y video digital, empaquetamiento de datos y nuevas técnicas de modulación de señales RF. El empaquetamiento permite separar video, audio y datos auxiliares en unidades de tamaño determinado con corrección de errores.

En ATSC se tienen tres etapas bien definidas que son: Compresión, multiplexación y modulación. La señal de video es una señal de alta definición se comprime con MPEG-2, la señal de audio se comprime con Audio Code 3

AC-3 (Código de audio 3, se compone de 6 canales discretos), existen también servicios interactivos y otras de altas presentaciones denominadas DASE (Digital TV Applications Software Environment) [2].

### 1.3.2 Sistema Europeo DVB

Digital Video Broadcasting (DVB) es una organización europea que promueve estándares aceptados internacionalmente de televisión digital, en especial para HDTV y televisión vía satélite, así como para comunicaciones de datos vía satélite (unidireccionales, denominado SDVB-IP, y bidireccionales, llamados DVB-RCS).

Su uso principal es la transmisión de programas múltiples en un solo canal. Fue desarrollado después de ATSC, con la intención de ampliar la competitividad.

La norma DVB se ocupa de adecuar la señal al medio de difusión, define el código común de corrección de errores, elige el tipo de modulación y codificación adecuada a cada canal, marca el método de cifrado además fija la interfaz común de acceso condicional. DVB utiliza para el transporte de flujo de datos la plataforma MPEG-2, ISO/IEC 13.818-1 de muy amplia difusión en diversas aplicaciones, para el audio utiliza MPEG-2 ISO/IEC 13.818-2 la modulación utilizada es COFDM [2].

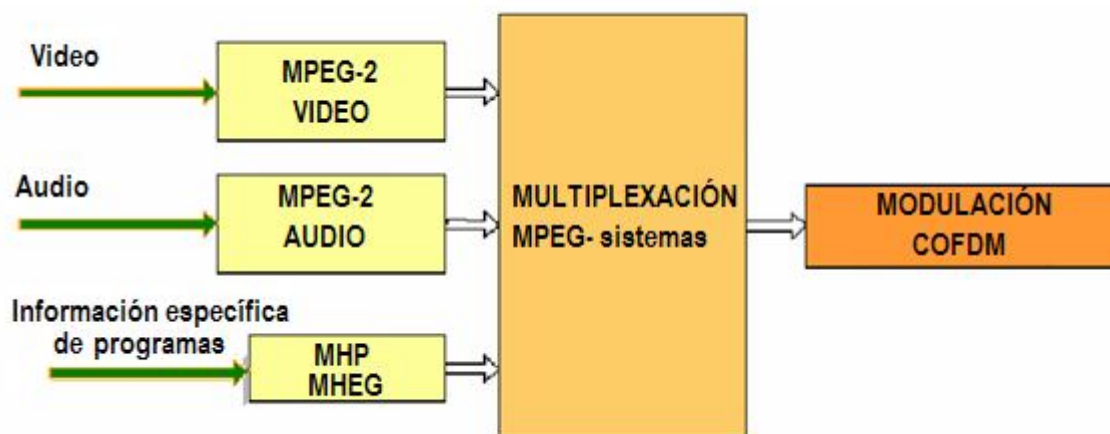


Figura 1. 1 Sistema DVB

DVB-T sacrifica una parte de su capacidad de transporte de datos para lograr mejor inmunidad al multitrayecto. Se debe tener muy en cuenta ya que es muy importante una vez adoptada una forma de transmisión, la plataforma sólo es eficiente para esa forma de servicio es por ello que se transmiten 5 o



más programas de SDTV en un solo canal de 8MHz para el servicio fijo, para el servicio de televisión móvil, se lo hace alojando solamente una señal o máximo 2 señales de SDTV en el canal de 8MHz.

DVB tiene muchos estándares, los más utilizados en televisión son los siguientes:

- DVB-S Estándar para transmisión de señales DVB por satélite.
- DVB-C Estándar para transmisión de señal por cable.
- DVB-H Estándar para televisión digital terrestre para dispositivos móviles
- DVB-T Estándar para televisión digital terrestre.

### **1.3.3 Sistema ISDBT**

El estándar ISDB (Integrated Services Digital Broadcasting), se ha ido desarrollado en Japón por el grupo DIEBEG (Digital Broadcasting Experts Group), es un estándar semejante al DVB-T con ciertas modificaciones como lo veremos a continuación.

El sistema ISDB-T es el estándar internacional de televisión digital terrestre más avanzado que existe actualmente. ISDB tiene tres tipos de sistemas ISDB-S (Satélite), ISDB-T (Terrestre) e ISDB-C (cable), todos ellos desarrollados en Japón con la finalidad de ofrecer flexibilidad, capacidad de expansión y difusión de los servicios de transmisión de multimedia, cada uno empleando las redes respectivas.

Este sistema está diseñado para transmitir una señal móvil y fija simultáneamente en 6MHz de ancho de banda, ya que en Japón viajan varias horas hasta llegar a sus trabajos, lo que más tomaron en cuenta es la movilidad.

ISDB utiliza uno o más flujos de transporte TPS definidos en el estándar MPEG-2 que son remultiplexados para crear un único TS, el mismo que luego es sometido a múltiples procesos en el canal de codificación, de acuerdo a la naturaleza del servicio y se envía finalmente como una única señal OFDM. Además ISDB difiere principalmente en las modulaciones, debido a los requisitos de las diferentes bandas de frecuencia. La compresión del audio y el video es igual que en DVB-T es decir, se utilizan para algoritmos MPEG-2.

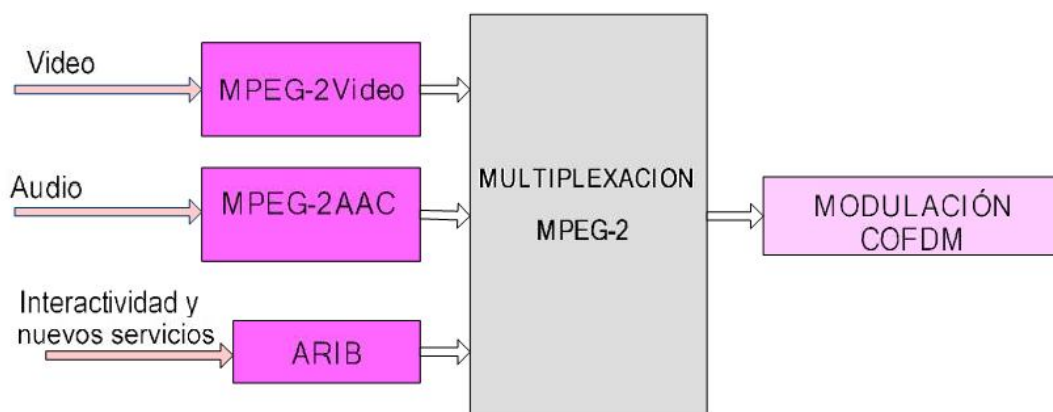


Figura 1. 2 Sistema ISDB

### 1.3.4 Sistema DTMB

*Digital Terrestrial Television Multimedia Broadcasting* (DTMB) fue desarrollada en la República Popular China, aprobado en agosto de 2007, posee características diferentes a los otros estándares tanto en el sistema de modulación como en la codificación de canal. Existen dos estándares previos también desarrollados en China, ADTB-T, similar al ATSC y el desarrollado en la Universidad de Jiaotong de Shanghai y el DMB-T desarrollado en la Universidad Tsinghua de Beijing.

DTMB ofrece alta definición de la imagen, movilidad plena (gracias a su estándar DTMB-H), no existe reporte de errores que afecten la calidad de la imagen, utiliza *time interleaving* pero su desarrollo e investigación es costosa, además es un estándar aceptado sólo en China, Hong Kong y Macao.

*Características principales del estándar:*

- Opera en un ancho de banda de 8 MHz
- Utiliza intervalos de guarda de 1/9, 1/7 ó 1/4.
- Tasa de Datos: Mínima de 4.81 Mbps pero puede llegar a alcanzar los 21.96 Mbps
- Carga útil: 20 Mbps más para HDTV y 5Mbps más para SDTV.
- Cobertura SFN con propagación máxima de alrededor de 16 a 35 km.
- Ofrece transmisión de HDTV, SDTV, audio, datos, diversos servicios multimedia e interactividad.

Con respecto a la calidad de servicio ofrecido por el estándar chino de DTTB y sus características, comparado con los otros estándares internacionales (DVB-T, ATSC, ISDB-T), DTMB ha implementado algunas tecnologías nuevas referentes al campo de la comunicación, con el fin de optimizar el desempeño y rendimiento del sistema.

Además el estándar DTMB garantiza protección contra el multitrayecto y el ruido impulsivo, tiene altas velocidades en movilidad, posee ventajas en cantidad de Mb/s comparado con el sistema ATSC Y DVB.

Otra ventaja significativa de este estándar es que emplea TDS-OFDM (*Time Domain Synchronuous- Orthogonal frequency-division multiplexing*), lo cual permite transmitir HDTV de calidad aceptable en vehículos en movimiento a velocidades de hasta 200 Km/h. Además, tiene soporte para dispositivos móviles y reproductores multimedia.

A pesar de las ventajas y beneficios que brinda este estándar, todavía no tiene definido sus estándares en cuanto a codificación de vídeo se refiere, por lo cual sus costos de desarrollo e investigación son elevados.

## **1.4 LA TV DIGITAL EN EL ECUADOR**

### **1.4.1 Definición y antecedentes**

La introducción de la Televisión Digital Terrestre en el Ecuador puede ser parte de un rutinario y simple cambio tecnológico en el devenir de cualquier industria o una verdadera oportunidad para transformar notablemente el modo en que vemos, sentimos y pensamos de la televisión.

El desarrollo Tecnológico experimentado en los tiempos actuales en los diferentes ámbitos incluidos el de la televisión, trae consigo la creación de nuevos modelos de negocios, con características incluyentes y solidarias, que responden a las necesidades de un público cada vez más exigente, contribuyendo al desarrollo de los pueblos.

Los gobiernos de muchos países alrededor del mundo, a fin de facilitar la inclusión de sus pueblos a esta nueva forma de ver televisión, han definido

según su propio análisis el estándar de televisión digital que implementarán en sus países. Ecuador no ha sido la excepción, y algunos países de la región han tomado su definición y se encuentran en la fase de implementación de la Televisión Digital Terrestre, es por esto que Ecuador empezó a trabajar en esta oportunidad histórica para revolucionar la televisión, y poner la tecnología y sus ventajas, al alcance de todos. Otra perspectiva de análisis hace relación con los alcances que tendrá la llegada de la Televisión Digital Terrestre:

Para los consumidores, la televisión digital teóricamente puede dar acceso a un gran número de contenidos, con mayor calidad de imagen y sonido, podrán acceder a nuevos servicios que desarrollen los futuros operadores, además de las variadas aplicaciones interactivas.

Para los operadores de televisión digital, les dará la posibilidad de crear nuevos modelos de negocios basados en la variedad de programación que podrían ofrecer y en el inmenso atributo de la interactividad.

Para poder elegir el estándar adecuado para el Ecuador, la Superintendencia de Telecomunicaciones “SUPERTEL” fue el organismo que se encargó de realizar una investigación profunda sobre el tema, realizando pruebas técnicas, analizando el impacto socioeconómico, la cooperación internacional, aspectos políticos de integración regional y regulatorios. Además la “SUPERTEL” especificó los criterios que empleó para la evaluación de estos distintos aspectos según el grado de importancia subjetiva dada a cada uno de ellos.

En cuanto a las pruebas técnicas antes que nada se definió el método o protocolo de pruebas de evaluación del desempeño técnico y los resultados obtenidos para cada uno de los estándares. Para analizar el impacto socioeconómico se realizó una investigación de uso, hábitos y preferencias de la televisión en el Ecuador, que sirvieron de base en conjunto con otros factores sociales y económicos para la visualización del impacto que tendría la implementación del estándar de televisión digital en el Ecuador; considerando la oferta y demanda del servicio.

Por el aspecto de cooperación internacional y estratégica, la SUPERTEL analizó den detalle cada una de las ofertas de los proveedores de los estándares como respuesta a requerimientos concretos en los diferentes tópicos de cooperación. Finalmente se analizaron las consideraciones que deberán tenerse en cuenta al momento de regular la prestación y oferta de servicios.

#### **1.4.2 Pruebas y Mediciones**

El principal objetivo planteado era determinar el estándar que se adapte, desde el punto de vista técnico, a todas las características de transmisión y recepción del medio, teniendo en cuenta que dicho estándar en un principio debería coexistir con la televisión analógica durante el tiempo de migración entre tecnologías y no debería haber problemas en el comportamiento de estos estándares en canales adyacentes a los de televisión analógica. Desde un punto de vista más técnico, se deberá evaluar los estándares de Televisión Digital Terrestre, bajo parámetros similares de transmisión y recepción, teniendo en cuenta la recomendación UIT-R BT.2035-11.

Es muy importante determinar la calidad y disponibilidad de la señal de Televisión Digital Terrestre en recepción: fija, portátil, peatonal, móvil y personal.

Finalmente se debe identificar las fortalezas, debilidades, niveles de cobertura y eficiencia de cada estándar a prueba.

##### **1.4.2.1 Pruebas de evaluación subjetiva**

Antes de realizar estas pruebas primero se definió los diferentes modos de recepción necesarios que son:

**Recepción fija:** Pueden ser exteriores o interiores y se realiza con equipos y antenas sin movimiento.

**Recepción portátil:** Se realiza con equipos que se pueden mover de un lugar a otro y poseen antenas incorporadas. Estos equipos permanecen inmóviles durante las pruebas.

**Recepción peatonal:** La recepción se realiza con un equipo que se desplaza a una velocidad menor o igual a 5 km/h, sujeto a ligeros movimientos ocasionales y frecuentes.

**Recepción móvil:** La recepción se realiza con un equipo que se desplaza a una velocidad superior a 5 km/h. Para el efecto se utiliza un vehículo en movimiento.

**Recepción Personal (Movilidad con portabilidad):** La recepción se realiza con un dispositivo portátil, con antena de baja ganancia, ubicado al interior de un vehículo en movimiento.

Todas las pruebas de campo para cada estándar se realizan en condiciones similares de transmisión. La evaluación abarca aspectos relacionados con la propagación, cobertura, disponibilidad del servicio y robustez.

Para realizar esta evaluación subjetiva de imagen y sonido se consideró los 5 niveles establecidos en la recomendación UIT-R BT-500 que se muestran en el cuadro 1.1 [4].

**Tabla 1. 1 Evaluación subjetiva de imagen y sonido**

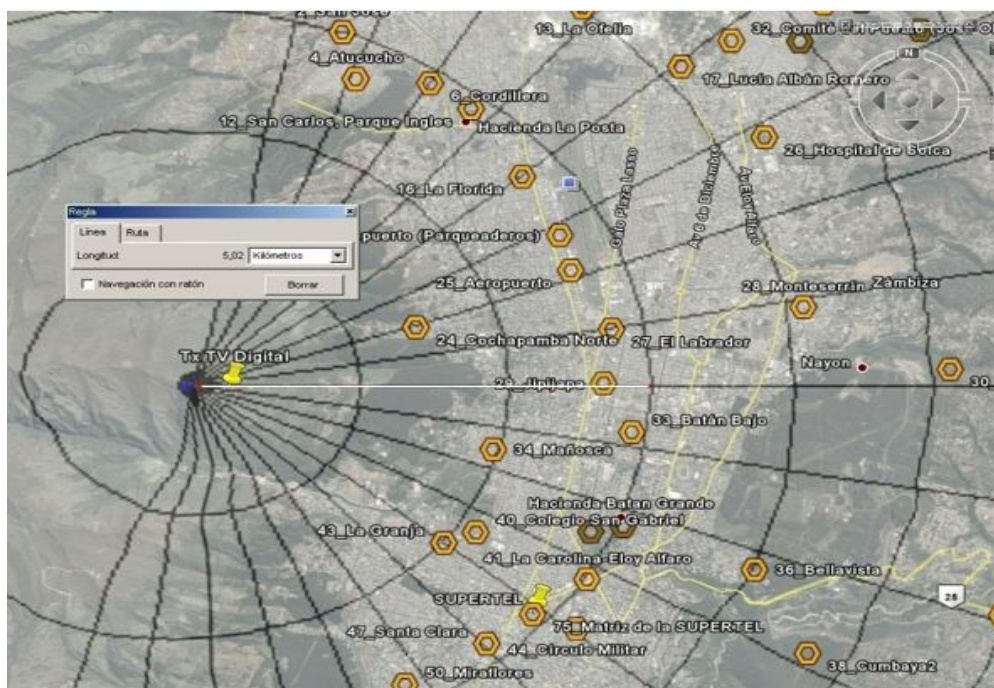
<b>Grado</b>	<b>Calidad</b>	<b>Deterioro</b>	<b>Definición</b>
5	Excelente	Imperceptible	Recepción sin fallas
4	Bueno	Perceptible, pero no molesto	Recepción con fallas que no motivan al evaluador cambiar de canal.
3	Regular	Ligeramente molesto	Recepción con fallas, por lo cual el evaluador no está conforme, pero considera que se puede esperar un intervalo de tiempo sin cambiar de canal
2	Pobre	Molesto	Recepción con fallas que motiva al evaluador cambiar de canal
1	Malo	Muy molesto	No hay señal

Fuente: Supertel

Definidos los niveles de evaluación, se procedió a coordinar con universidades para contar con veedores durante las pruebas y tener el punto de vista de un posible futuro televidente y de esta manera poder contar con evaluaciones subjetivas ya que no es posible caracterizar totalmente la calidad de funcionamiento de manera objetiva.

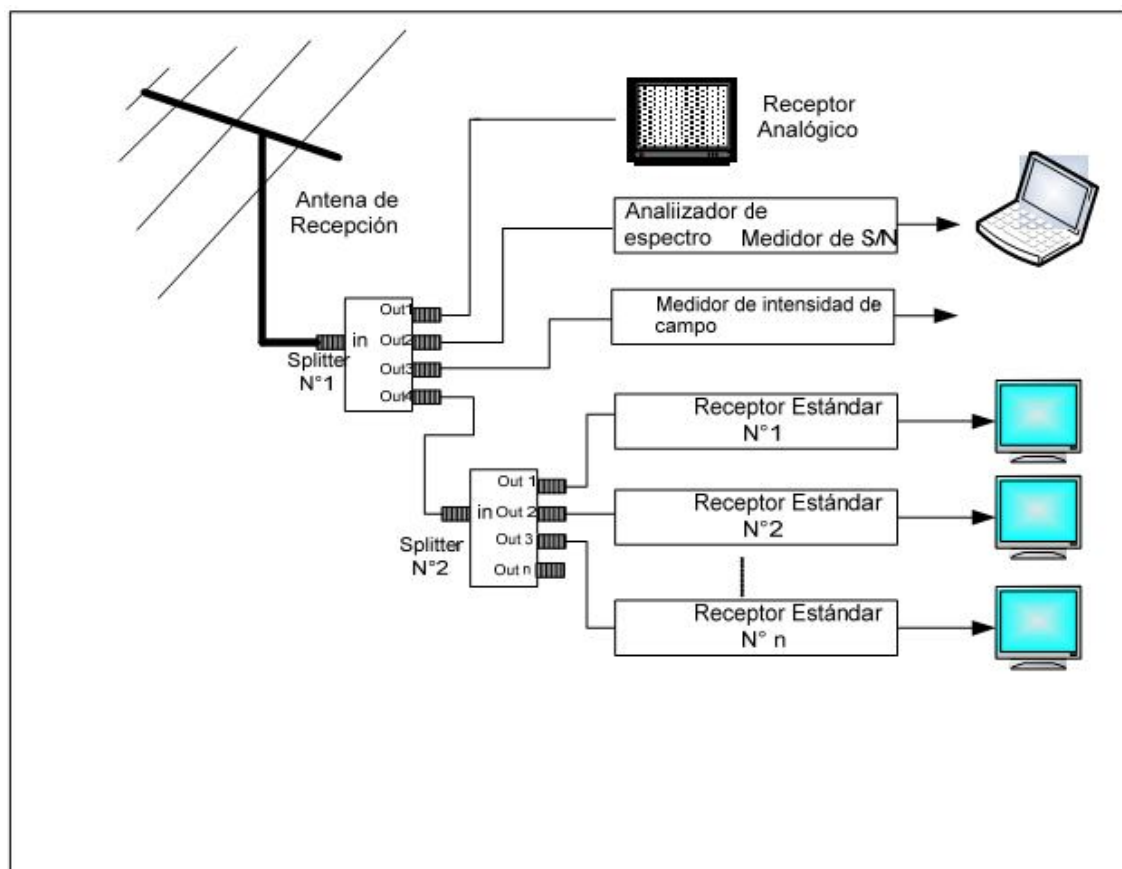
#### 1.4.2.2 Pruebas de cobertura

Para estas pruebas se considera la Recomendación UIT-R P.845-3 (Medición de la intensidad de campo en ondas decamétricas) y la Recomendación UIT-R BT.2035-1 (Directrices y técnicas para la evaluación de sistemas de radiodifusión de televisión digital terrenal). La cobertura se establece con la determinación de la intensidad de campo real medida para una facilidad de transmisión dada, que permite obtener una determinada calidad de recepción, dependiendo de la sensibilidad del receptor, del rendimiento de la antena y del ruido natural o artificial. Las pruebas de cobertura se realizan siguiendo una metodología basada en mediciones a lo largo de radiales y arcos. Para estas mediciones se definen 85 emplazamientos. En la figura 1.3 se puede ver la ubicación de los emplazamientos en exteriores [5].



**Figura 1. 3 Ubicación de los emplazamientos para pruebas de cobertura**  
Fuente: Supertel

Para poder realizar las pruebas se tomó como base la recomendación UIT-R BT. 2035.1, “Directrices y técnicas para la evaluación de sistemas de radiodifusión de televisión digital terrenal”, cuyo objetivo es evaluar la calidad de funcionamiento del sistema o sistemas disponibles con diversas configuraciones de transmisión y recepción.



**Figura 1. 4 Disposición del equipo para las pruebas de campo**

#### 1.4.2.3 Ejecución e inicio de pruebas objetivas

Según información proporcionada por la SUPERTEL, el 20 de febrero de 2009 se da inicio a las pruebas de Televisión Digital Terrestre en el Ecuador y se continúan de acuerdo a la tabla 1.2.



**Tabla 1. 2 Cronograma de pruebas de la SUPERTEL**

ESTÁNDAR	PERIODO DE PRUEBAS	
	INICIO	FIN
	20/02/2009	13/03/2009
ISDB-T	20/02/2009	13/03/2009
SBTV D	29/06/2009	10/07/2009
DTMB	29/06/2009	10/07/2009

Fuente: Supertel

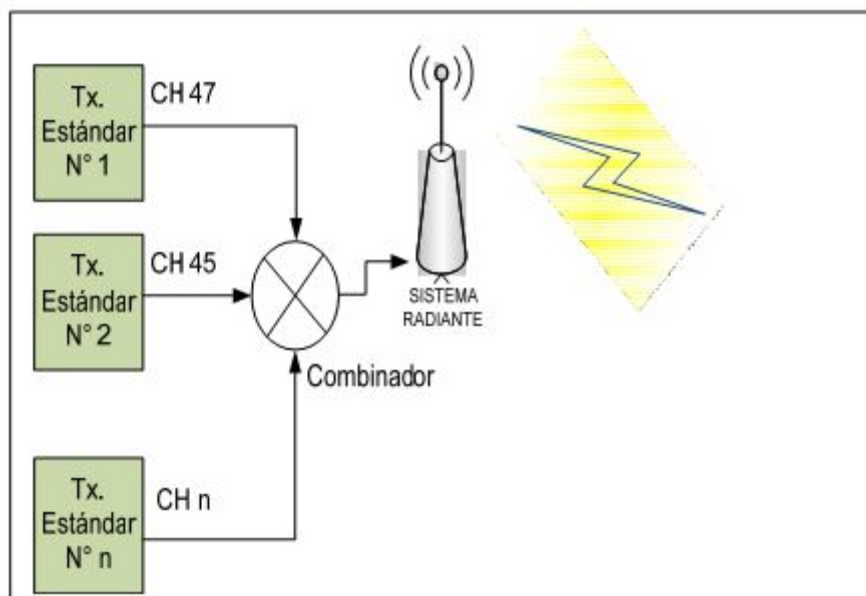
Para la ejecución de las pruebas de cada estándar, se emplean transmisores con una potencia de 500 W y un ancho de banda de 6 MHz. Los parámetros de configuración de cada estándar se muestran en la tabla 1.3.

**Tabla 1. 3 Parámetros de configuración de cada estándar de TV digital**

ESTÁNDAR	DVB-T	ISDB-T		SBTV D		DTMB
Tipo de señales	1 HD	1 HD /One Seg		2 HD/One Seg		2HD/SD
No. de portadoras	8K	8k		8k		8k
Intervalos de guarda	1/16	1/16		1/8		1/9
Compresión	MPEG-2	MPEG-2		MPEG 4		MPEG4
Esquema modulación	64QAM	QPSK	64QAM	QPSK	64QAM	64QAM
FEC	$\frac{3}{4}$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{3}{4}$
Time interleave	----	0,4 seg.	0.2 seg.	0,4 seg.	0.2 seg.	----
Tasa de transmision	17,56 Mbps	440 kbps	17,8 Mbps	440 kbps	17,8 Mbps	8Mbps
Potencia operacion	500W	500 W				500W
Marca TX	BTESA	TOSHIBA				BBEF

Fuente: Supertel

Las pruebas se realizan utilizando la siguiente configuración de transmisión:



**Figura 1. 5 Diagrama del sistema de Transmisión**

Para estas pruebas, se cuenta con los canales 43, 45 y 47 de la banda UHF, asignados por el Consejo Nacional de Radiodifusión y Televisión, CONARTEL.

**Tabla 1. 4 Canal utilizado por cada estándar**

Estándar	Canal utilizado
DVB-T	45
ISDB-T	47
SBTVD	47
DTMB	45
ANALÓGICO (Adyacentes operativos)	46 y 48

#### 1.4.2.4 Ejecución e inicio de pruebas subjetivas

Los métodos utilizados en este tipo de pruebas fueron:

- De estímulo único:* Se presenta una secuencia de imágenes a las que el evaluador otorga una calificación subjetiva

- b) *De apreciación por categoría de adjetivos*: Los observadores asignan a una imagen o secuencia de imágenes una categoría elegida, entre un conjunto de categorías, que se definen en términos semánticos. Las categorías reflejan apreciaciones o atributos.

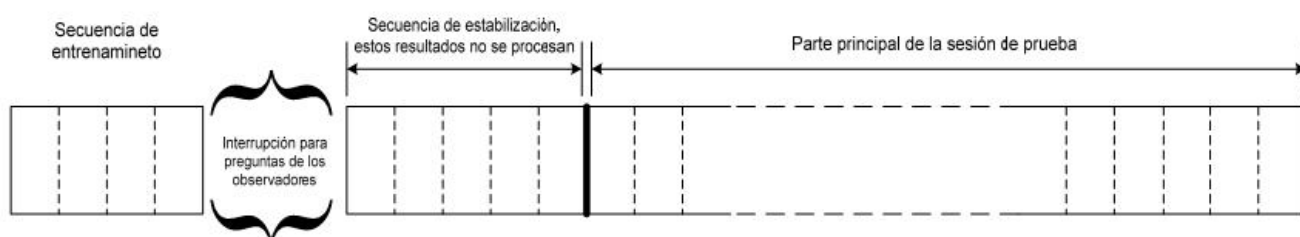
Cabe recalcar que las escalas de categorías que avalúan la calidad y degradación de imagen son las que se mostraban en el cuadro 1.1 de acuerdo a la recomendación definida por la UIT-R.

Con esta prueba se quería evaluar la calidad de la cadena de transmisión de televisión percibida por el usuario. En estas pruebas se incluyen material variado con poco y mucho movimiento que simulen horas normales de programación.

#### 1.4.2.4.1 Definición de evaluadores y sesión de evaluación

Se seleccionaron evaluadores de diversas universidades del país y con diferentes especializaciones, sin que necesariamente sean expertos en calidad de imagen.

Previo a la realización de las pruebas, se orientó a los evaluadores sobre el proceso, los tipos de degradaciones que probablemente se produzcan, la escala de apreciaciones, la secuencia y la temporización. La sesión de evaluación se indica en la figura 1.6, en donde la sesión de prueba tiene una duración de 3 minutos.



**Figura 1. 6 Estructura de la presentación de una sesión de prueba**

#### **1.4.2.5 Pruebas de cobertura**

Para esta prueba se ubican diferentes puntos para la recepción en modo fijo en la ciudad de Quito y sus alrededores, evaluando en cada uno de ellos la calidad de la señal de audio y video de los estándares DVB-T, ISDB-T, SBTVD, DTMB y de la señal analógica, bajo las siguientes condiciones:

- Ubicación de 85 puntos exteriores y 20 interiores
- Recepción fija con antena yagi de uso convencional, a 10 metros sobre el nivel del suelo para los puntos exteriores y antena de lazo convencional generando ruido impulsivo con una licuadora para los puntos interiores
- TV LCD de 22" para ambos casos

Posteriormente se procede a configurar el sistema de recepción que conlleva la antena UHF, el cable coaxial y el montaje sobre un mástil a 10 metros del nivel del suelo. A continuación se configura el sistema de televisión conectándolos a los diferentes decodificadores de cada uno de los estándares en cuestión, escogiendo los canales adecuados para las pruebas.

Finalmente Las mediciones efectuadas en cada uno de los puntos se registran en los respectivos formularios con la firma de respaldo de cada uno de los observadores. Las mediciones de carácter técnico se registran en formato electrónico y el proceso en general en formato audiovisual a través de una cámara de video. Los evaluadores observan la señal durante un tiempo promedio de 3 minutos, luego del cual proceden a asignar la calificación respectiva con base en el cuadro de evaluación subjetiva incluido en cada formulario.

#### **1.4.2.6 Pruebas de recepción portátil, peatonal y personal**

Estas pruebas se las realizó en 20 puntos de la ciudad (recepción portátil y peatonal) y en una unidad móvil (recepción personal) bajo las siguientes condiciones:

Recepción portátil fija, por un periodo de 3 minutos, sentado o de pie en un determinado lugar (se las realiza para los estándares ISDB-T, SBTVD y DTMB).

Recepción peatonal, por un período de 3 minutos caminando en un área aproximada 20 m<sup>2</sup> (se las realiza para los estándares ISDB-T, SBTVD y DTMB). Recepción portátil dentro de un vehículo en movimiento (se las realiza para los estándares ISDB-T, SBTVD y DTMB).

#### 1.4.2.7 Metodología de evaluación

Para la recepción portátil y peatonal, se proporcionó a cada uno de los evaluadores un terminal portátil, que se sitúa a una altura de 1.5 metros sobre el nivel del suelo.

Para la recepción personal se proporcionó a cada uno de los evaluadores un terminal portátil a ser utilizado en los circuitos destinados para pruebas de recepción móvil. El desarrollo de la medición se registró en formato audiovisual a través de una cámara de video.

#### 1.4.2.8 Criterios de ponderación de las mediciones y resultados

Para obtener los resultados de cada una de las mediciones primero se estableció un cuadro de ponderaciones de cada uno de los resultados obtenidos sobre la base de los criterios que se muestran en la Tabla 1.5 [6].

**Tabla 1. 5 Criterios de ponderaciones de las mediciones**

Pruebas	Parámetros	Recepción	No de mediciones a cada estándar	Ponderación	%
<i>Cobertura</i>	Intensidad de campo [dBμV]	Exteriores	85	30%	60%
<i>Disponibilidad</i>	Voltaje en el receptor [dBμV]	Exteriores, interiores	105 exteriores y 20 interiores	10%	
	Ancho de	Exteriores,	105 exteriores y	10%	

	banda [MHz]	interiores	20 interiores		
	Señal a ruido [S/N]	Exteriores, interiores	105 exteriores y 20 interiores	10%	
<i>Calidad</i>	Evaluación subjetiva	Exteriores, interiores	375*	20%	40%
		Móvil	24 tramos **	10%	
		Portátil, peatonal y personal	123 tramos ***	10%	
TOTAL					100%

\* 3 veedores por cada punto de medición; 85 puntos exteriores, 20 puntos en interiores (con y sin ruido impulsivo).

\*\* 3 veedores por cada circuito; 3 circuitos, separados por 8 tramos para pruebas de movilidad

\*\*\* 3 veedores para un circuito de 4 tramos para pruebas de recepción personal, 20 mediciones con dispositivos portátiles y 20 mediciones con recepción peatonal.

Fuente: Supertel

Con estos criterios pasamos a visualizar los resultados que se obtuvieron en las pruebas.

#### 1.4.2.8.1 Resultados globales del desempeño técnico

Los resultados globales obtenidos de la evaluación técnica realizada a cada estándar se incluyen en la tabla 1.6 [7].

Tabla 1. 6 Resultados de las mediciones y evaluaciones de cada estándar

Pruebas	Objetivo	Parámetros	Resultados		
			Estándar	Medidos (promedio)	Ponderación al 30 %
Técnicas	Cobertura	Intensidad de campo [dBμV/m]	DVB-T	57.49	29.86 %
			ISDB-T	57.75	30.00 %
	Disponibilidad	Voltaje en el receptor [dBμV]	Estándar	Medidos (promedio)	Ponderación al 10 %
			DVB-T	26.49	9.90 %
			ISDB-T	26.76	10.00 %
		Ancho de banda	Estándar	Medidos	Ponderación al 10 %
			DVB-T	< 6 MHz	10.00 %
			ISDB-T	< 6 MHz	10.00 %
		Señal a ruido	Estándar	Medidos (promedio)	Ponderación al 10 %
			DVB-T	18.77	9.37 %
Subjetivas	Calidad	Exteriores, Interiores,	Estándar	Medidos	Ponderación al 20 %
			DVB-T	115	10.55 %
			ISDB-T	218	20.00 %
		Móvil	Estándar	Medidos	Ponderación al 10 %
			DVB-T	3	6.00 %
			ISDB-T	5	10.00 %
		Portátil, Peatonal y Personal	Estándar	Medidos	Ponderación al 10 %
			DVB-T	--	---
			ISDB-T	82	10 %

Fuente: Supertel

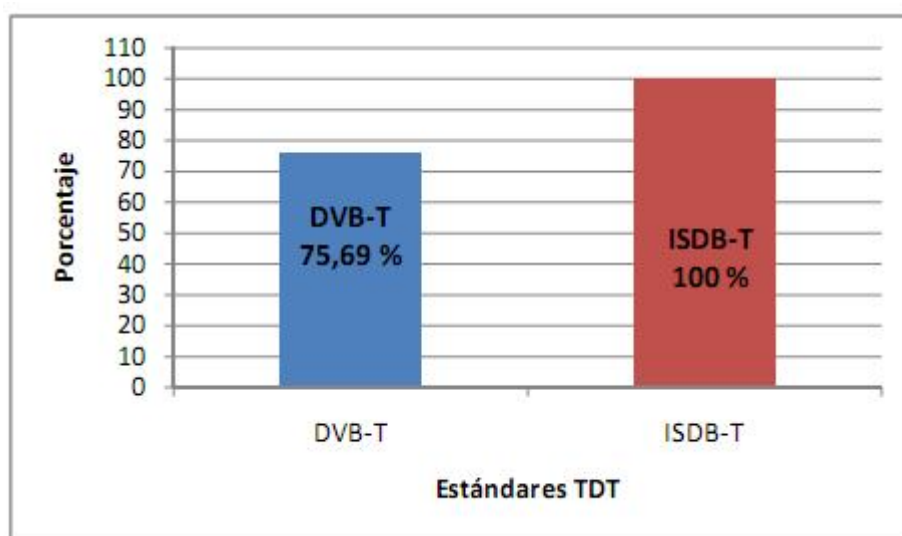
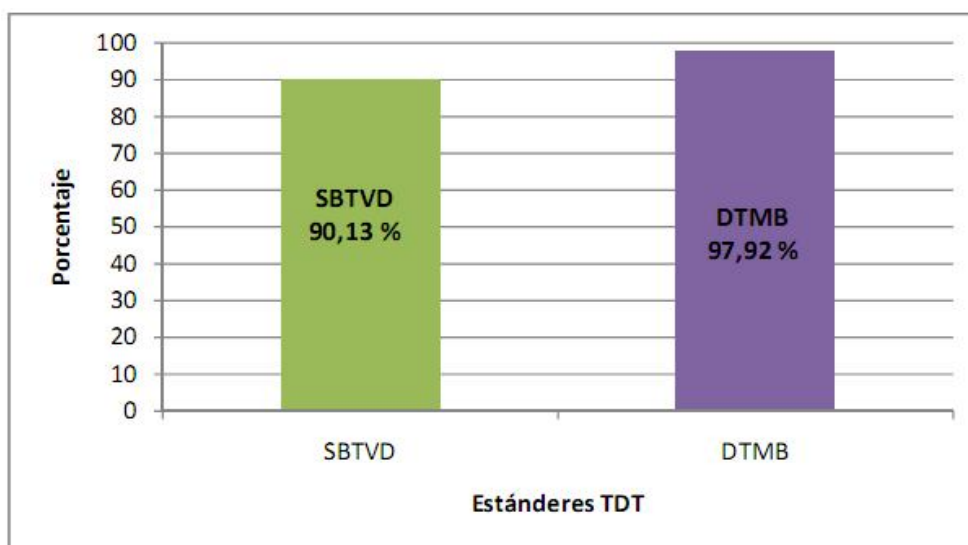


Figura 1. 7 Resultados finales de las pruebas técnicas con estándares que utilizaron compresión MPEG-2

**Tabla 1. 7 Resultados de las mediciones y evaluaciones de los estándares con compresión MPEG-4**

Pruebas	Objetivo	Parámetros	Resultados		
			Estándar	Medidos (promedio)	Ponderación al 30%
Técnicas	Cobertura	Intensidad de campo [dBμV/m]	SBTVD	56.17	28.90 %
			DTMB	58.30	30.00 %
	Disponibilidad	Voltaje en el receptor [dBμV]	Estándar	Medidos (promedio)	Ponderación al 10%
			SBTVD	23.87	8.69 %
			DTMB	27.47	10,00 %
		Ancho de banda	Estándar	Medidos	Ponderación al 10%
			SBTVD	< 6 MHz	10.00 %
			DTMB	< 6 MHz	10.00 %
		Señal a ruido	Estándar	Medidos (promedio)	Ponderación al 10%
			SBTVD	19.31	9.66 %
			DTMB	19.98	10.00 %
Subjetivas	Calidad	Exteriores, Interiores,	Estándar	Medidos	Ponderación al 20%
			SBTVD	256	19.54 %
			DTMB	262	20.00 %
		Móvil	Estándar	Medidos	Ponderación al 10%
			SBTVD	1	3.33 %
			DTMB	6	10.00 %
		Portátil, Peatonal y Personal	Estándar	Medidos	Ponderación al 10%
			SBTVD	101	10.00 %
			DTMB	80	2.38 %

Fuente: Supertel



**Figura 1. 8 Resultados finales de las pruebas técnicas con estándares que utilizaron compresión MPEG-4**

Fuente: Supertel



### 1.4.3 Selección del estándar

Luego de realizar todas las pruebas y análisis respectivos, para que las autoridades competentes puedan tomar una decisión sobre el estándar que debía adoptar el Ecuador se tomó en cuenta cuatro aspectos principales:

*Estudio y Pruebas técnicas:* Incluye la información anteriormente mostrada sobre la evaluación de desempeño de cada estándar, en los diferentes parámetros considerados en el protocolo de pruebas.

*Impacto Socioeconómico:* Este aspecto es evaluado considerando la oferta y la demanda del servicio. En el primer caso, se evaluó el impacto de las inversiones en las que incurrirán los operadores de televisión; en el segundo, se hace el mismo ejercicio tomando como punto de partida las repercusiones económicas en la población, con la adopción de cada uno de los estándares

*Cooperación Internacional:* Los beneficios de cooperación que obtendría el país con la decisión evaluado de manera cualitativa.

*Despliegue:* Se evaluó la penetración mundial del estándar, el potencial de población a servir, despliegue efectivo actual y los años de existencia del estándar.

El grado de importancia de los aspectos a considerar en la toma de decisión, se observa en la tabla 1.8.

**Tabla 1. 8 Grado de importancia de los aspectos de evaluación**

PARÁMETROS	IMPORTANCIA
Estudio Socioeconómico	Muy Alta
Cooperación Internacional	Muy Alta
Pruebas Técnicas	Alta
Despliegue	Media

Finalmente se obtuvieron los siguientes resultados de la evaluación en base a los aspectos mencionados anteriormente:

Tabla 1. 9 Resultados de las pruebas de cada estándar.

RESULTADOS DE LA EVALUACION					
ASPECTOS	IMPORTANCIA	Estándar Americano ATSC	Estándar Japonés-Brasileño ISDB-T/SBTVD	Estándar Europeo DVB-T	Estándar Chino DTMB
ESTUDIO SOCIOECONÓMICO	MUY ALTA	4º	1º	3º	2º
COOPERACIÓN INTERNACIONAL	MUY ALTA	4º	2º	1º	3º
PRUEBAS TÉCNICAS	ALTA	4º	2º	3º	1º
DESPLIEGUE	MEDIA	3º	2º	1º	4º

Fuente: Supertel

Finalmente en marzo de 2010 el Ecuador eligió el Estándar Japonés Brasileiro ISDB-T / SBTVD, volviéndolo el estándar más popular de la región ya que ha sido adoptado por Brasil, Perú, Argentina, Chile, Venezuela, Ecuador, Costa Rica, Paraguay, Bolivia, Nicaragua y Guatemala.

### 1.5 SITUACIÓN ACTUAL

Muy pronto la población, empezará a acoger el lenguaje audiovisual como el modo de comunicación interpersonal. Resultado de la convergencia de las telecomunicaciones que se está implementando, por ejemplo, a través de los teléfonos y de la transmisión de imágenes por estos, de Internet, de la Televisión Digital y de otros medios que seguramente están por venir.

Cuando se habla de Televisión Digital, la calidad, movilidad, interactividad, internet, son algunos de los conceptos que repiten los expertos cuando analizan el presente y el futuro del medio televisivo, sus predicciones parten de una realidad indiscutible, “El usuario va a decidir qué quiere ver, cuando quiere ver y a través de que quiere ver”

En cuanto a la migración de a esta nueva tecnología, el tiempo que durará este proceso viene a ser una de las principales interrogantes que se extienden sobre la mesa.

Según autoridades de la SUPERTEL el tiempo en que se implemente la televisión digital en el Ecuador depende de varios factores, uno de ellos son los ciudadanos y su capacidad económica para adquirir tanto los decodificadores o también conocidos como set-top boxes o cambiar los televisores a unos que acepten el estándar ISDBT internacional.

Este es un proceso largo que va a tomar algunos años y depende además de las operadoras y su capacidad de inversión para digitalizar sus transmisiones, aunque según estudios realizados por la SUPERTEL, las operadoras estarían en capacidad de pasar a la tecnología digital en un lapso de cuatro años, por lo cual por el lado de la oferta al menos estaría solucionado este inconveniente en este tiempo [8].

Por otro lado en este mismo estudio se dice que los usuarios estarían en capacidad de pasar a esta tecnología en la mayor parte de su totalidad en aproximadamente siete años.

Además se está desarrollando una nueva industria que es la creación de contenidos, que deberán aportar a la interactividad de la televisión. Estas investigaciones y desarrollos tecnológicos se encuentran a cargo de algunas universidades del país.

Por el lado legal, en este momento el Estado ecuatoriano se encuentra organizando la parte normativa ya que un servicio no se puede prestar si no existe un marco adecuado por lo cual se está elaborando la norma técnica y el régimen de concesión, que establecen bajo qué condiciones se van a prestar los servicios, durante qué tiempo, cuáles serían las obligaciones y otro tipos de detalles importantes que deberán conocer los futuros inversionistas de este sector.

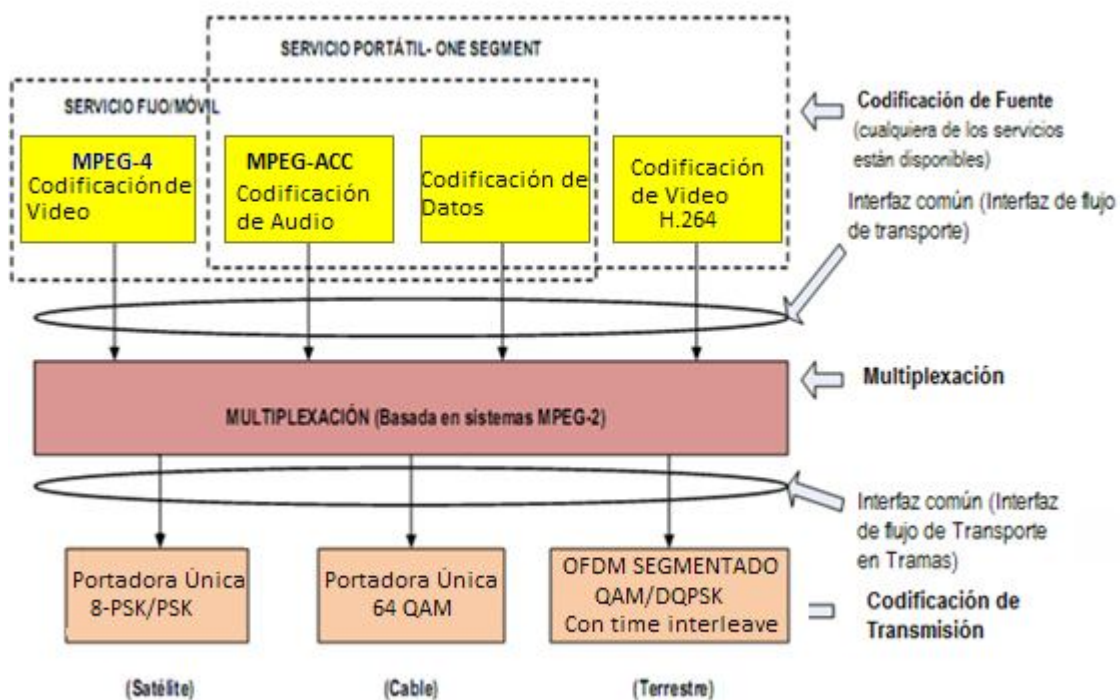
## 1.6 ESTÁNDAR ISDB-Tb

### 1.6.1 Estructura general

Un sistema de televisión transmisión digital está compuesto por tres bloques funcionales:

- Bloque de código fuente
- Bloque Multiplexación
- Bloque de transmisión de código.

En el diseño de un sistema de transmisión digital, se consideran los temas de servicio, configuración para el servicio de transmisión, es decir recepción fija, móvil y portable. Además la estructura tecnológica para la transmisión [9].



**Figura 1. 9 Estructura del sistema ISDB-TB**

Este sistema brasileiro está basado en el estándar japonés ISDB-T y sus diferencias fundamentales se encuentran en el uso de la utilización de tecnologías de compresión de audio y video mucho más avanzadas como lo son H.264/HE-AAC contra las usadas en Japón que son MPEG-2/MPEG L2.

Otra gran innovación es el middleware desarrollado en Brasil y la parte de protección de contenido. Es importante mencionar que la modulación en ambos sistemas es idéntica al igual que la parte del transporte que se realiza en base al estándar MPEG-2.

Una diferenciación que presenta ISDB-Tb en cuanto a los estándares de codificación de video y audio utilizados en las transmisiones móviles es que estos no son idénticos a los empleados en el sistema japonés. Por un lado el sistema Japonés utiliza Video H264 a 15 fps (frames per second) y Audio HE-AAC v.1 de baja complejidad, el sistema Brasileiro usa Video H264 a 30 fps, Audio HE-AAC v.2 de baja complejidad.

En la parte de multiplexación, este estándar es igual al ISDB-T ya que utiliza el tecnología MPEG-2, todos los contenidos transmitidos de audio, video y datos son multiplexados en un paquete de flujo de transporte llamado Transport Stream.

A pesar de que cualquier tipo de contenido/servicio puede ser multiplexado, los contenidos de audio, video y datos son convertidos al formato PES (Packet elementary stream) y finalmente convertidos al TS y multiplexados. Sin embargo los contenidos que no son del tipo de flujos de datos, son convertidos al formato de Sección y finalmente convertidos al formato TS y multiplexados. Ese TS debe ser sometido a la etapa de codificación de canal múltiple y debe ser obligatoriamente enviado como una señal OFDM común.

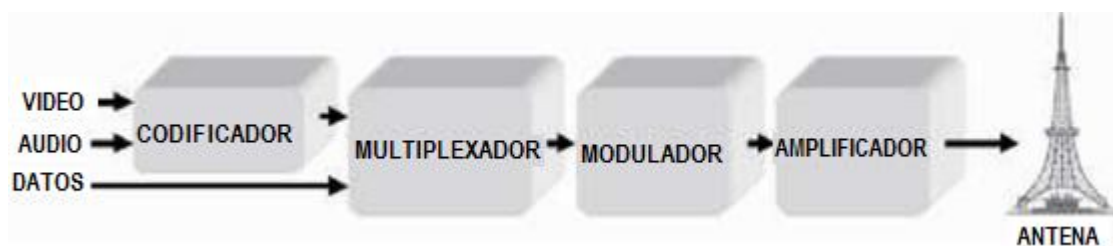
El Formato de TS nos permite multiplexar diferente clases de datos que a su vez se convierten en servicios que presenta entre sus características la TV Digital. Entre estos servicios se encuentran:

- Transmisión de Canales en SD.
- Transmisión de Canales en HD.
- Transmisión de Canales en One-Seg (dispositivos móviles).
- Servidor de EPG (Electronic Program Guide)
- Servidor de servicios de Closed Caption

- Servidor de Datos para Interactividad(Carrusel de Datos y Objetos)

En cuanto a la transmisión digital terrestre, se debe utilizar de manera obligatoria el time interleaving para poder proveer una codificación con una tasa de errores mínima para la recepción móvil. En cuanto al espectro de la radiodifusión de televisión digital, este debe consistir en 13 bloques OFDM sucesivos, con cada segmento ocupando el 1/14 del ancho de banda de canal de televisión.

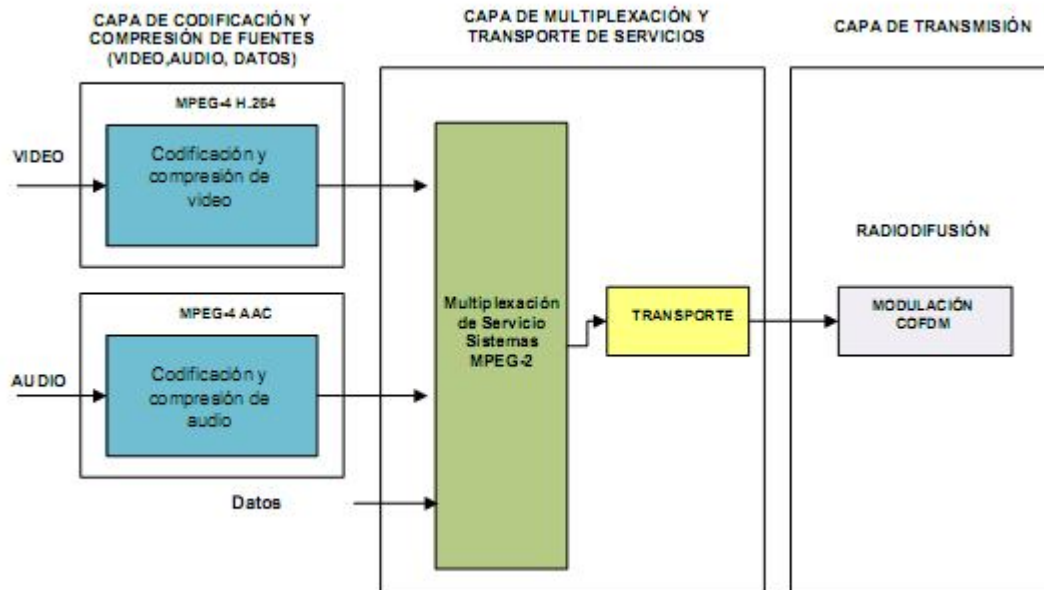
Un segmento OFDM debe obligatoriamente tener una configuración que permita la conexión de múltiples segmentos para abastecer un ancho de transmisión que atienda la necesidad del medio [10].



**Figura 1. 10 Sistema General de transmisión de ISDB-Tb**

El interleaving consiste en mejorar la calidad de transmisiones en medios sujetos a ruidos impulsivos ya que en el entrelazamiento temporal, los símbolos son transmitidos con redundancia y se puede corregir errores que se produzcan en determinadas partes de una palabra. El estándar ISDB-Tb utiliza el interleaving convolucional.

ISDB-Tb utiliza el concepto de OFDM, de multiplexación en frecuencia con múltiples portadoras ortogonales. Es llamado BST-OFDM porque utiliza también transmisión segmentada por bandas (transmisión jerárquica). La transmisión segmentada por bandas (BST) permite al sistema proporcionar tres tipos de servicios: recepción fija, móvil y portátil. El estándar brasileño utiliza COFDM, con un código Reed Solomon [11].



**Figura 1. 11 Sistema básico ISDB-TB**

### 1.6.2 Organización del Espectro Radioeléctrico

Primero se parte de la premisa que un canal de TV Digital va a utilizar un canal de 6MHz, el mismo ancho de banda que utilizan los canales de TV Analógica. Se tomó esta decisión para evitar problemas al momento de dividir nuevamente el espectro radioeléctrico.

Por otro lado, si ya empezamos hablar sobre la división del ancho de banda en varios canales se tiene que tomar en cuenta que no es posible utilizar todo el ancho de banda asignado para la transmisión, por lo que es necesario dejar ciertos márgenes, que se conocen como bandas de guarda, por encima y por debajo de los límites del canal para evitar posibles interferencias con los canales adyacentes.

Entonces si el canal tiene un ancho de banda de 6MHz se debe poner dos bandas de guarda de 200kHz. En total tenemos un ancho de banda de 400kHz que no podrá ser utilizado para la transmisión de datos. A partir de ahí se decide dividir al canal en segmentos de unos 400kHz aproximadamente, por lo cual si dividimos los 6MHz para 400kHz se obtiene que tenemos 15 segmentos disponibles de 400kHz.

Entonces como 400kHz van a ser utilizados para las bandas de guarda, se tienen 14 segmentos disponibles de 400kHz para el servicio. Una de las características fundamentales del estándar ISDB-Tb es la posibilidad de ofrecer un servicio de banda angosta que utiliza un solo segmento (conocido como servicio one-seg) y que debe ocupar el centro de la banda para que el receptor pueda sintonizarlo fácilmente y exista el menor consumo de energía posible [12].

Entonces si deseamos que exista un segmento en la mitad de la banda debería existir la misma cantidad de segmentos a la derecha y a la izquierda del mismo, por lo cual se decidió dividir a la banda en 14 segmentos en lugar de 15, donde uno de ellos estaría reservado para las bandas de guarda. Entonces se tiene que el ancho de banda de cada segmento sería:

$$Bw_s = \frac{Bw_c}{14} = \frac{6000kHz}{14} = 428,57kHz$$

Finalmente el estándar ISDB-Tb nos ofrece una banda con 13 segmentos de 428,57kHz para los diferentes tipos de servicios que se pueden ofrecer.

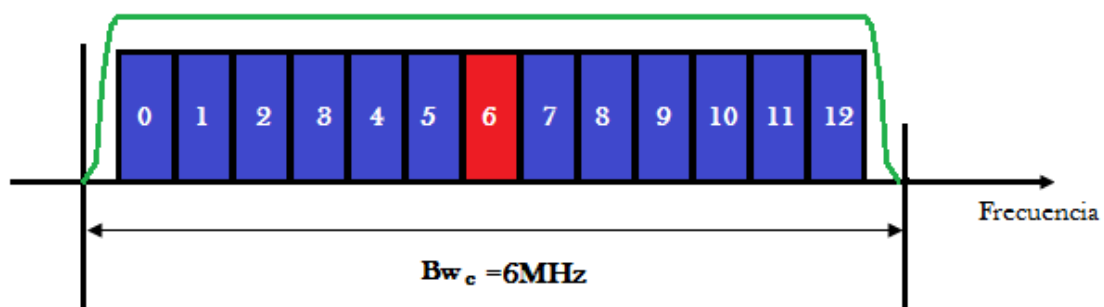


Figura 1.12 Organización del canal en segmentos

### 1.6.3 Sub-Sistema de modulación

El sistema ISDB-TB utiliza el sistema OFDM, el cual consiste en que los datos digitales son divididos en multiportadoras y enviados y como resultado, la longitud del símbolo de transmisión tiene mayor longitud que en un sistema de transmisión de una sola portadora.

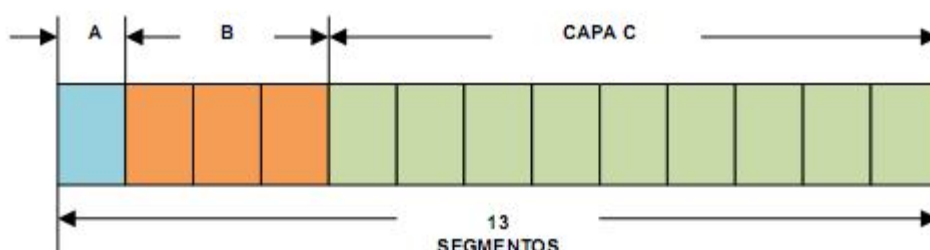
En el caso de ISDB-Tb se utiliza la transmisión segmentada OFDM, que es el único sistema de transmisión que es capaz de transmitir diferentes



parámetros de señal en el mismo ancho de banda. A este sistema se lo conoce como “transmisión en modo jerárquico”.

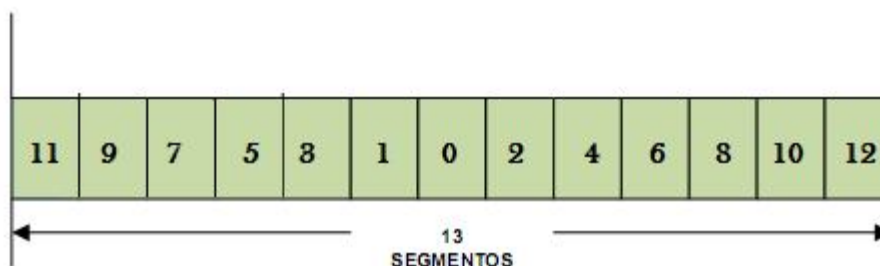
### 1.6.3.1 Transmisión Jerárquica

La “Transmisión jerárquica” se refiere a que los tres elementos que conforman la codificación de canal, es decir, el sistema de modulación, la tasa de código interno, y la duración del time interleave o intervalo de guarda, pueden ser establecidos de forma independiente y permiten transmitir flujos de datos para recepción fija y móvil de manera simultánea. Cada capa jerárquica consiste de uno o más segmentos OFDM. Para receptores fijos se puede transmitir canales en HDTV o varios programas en SDTV. Mientras más segmentos se utilicen, el ancho de banda será mayor y por lo tanto la velocidad con la que se transporta la información también aumentará. Finalmente dentro de los 13 segmentos podremos transportar distintos servicios como HDTV, SDTV y LDTV en un máximo de 3 capas [12].



**Figura 1. 13 Formación de las capas**

En primer lugar se debe aclarar que los 13 segmentos no están ordenados de manera clásica, es decir, ascendente de izquierda a derecha sino que se parte del segmento central 0 y luego hacia la izquierda y a la derecha de manera intercalada.



**Figura 1. 14 Orden de los segmentos en la banda de 6MHz**

Como se indicó en la sección anterior, el segmento central puede ser utilizado para recepción parcial y se lo considera una capa jerárquica (Capa A), este servicio, denominado “one-seg”, se lo considera de baja definición (LDTV) y está destinado para teléfonos móviles. La razón por la cual se maneja el servicio *one-seg* en el segmento central es debido a que cuando se sintoniza una banda de frecuencia, se parte de la frecuencia central y al hablar de teléfonos móviles se tiene que su principal limitación es el consumo de energía, entonces al estar ubicado el segmento en la frecuencia central, el dispositivo no gastará energía desplazándose hacia los segmentos contiguos sino que sintonizará la señal de manera más fácil y con un menor consumo de energía.

Además en ISDB-Tb los símbolos transportados por cada portadora pueden ser modulados utilizando demodulación diferencial o coherente. En caso de que alguna capa utilice la modulación diferencial, esta capa debe estar ubicada antes de la capa que utiliza modulación coherente. Es decir la Capa A será utilizada para recepción móvil, la capa B para modulación diferencial como DQPSK (servicios con definición estándar SD) y la capa C para modulación coherente como QPSK, 16QAM y 64QAM (servicios con alta definición HD)

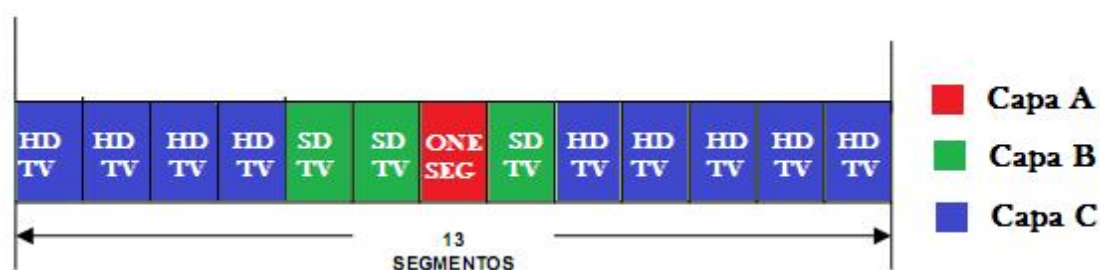


Figura 1. 15 Distribución de los segmentos en las capas

### 1.6.3.2 Modos de Transmisión y Segmentos de datos

El esquema de modulación OFDM puede ser configurado para operar en tres modos de transmisión con diferentes cantidades de portadoras que ocupan la misma banda. Es importante recalcar que la tasa de transmisión no depende de estos modos de operación, es decir, no depende del número de portadoras que se utilice. Esta tasa se ve afectada por los parámetros de codificador convolucional del canal, el intervalo de guarda y el tipo de modulación utilizada. Por otro lado la tasa útil de transmisión de cada segmento OFDM, puede variar

de 280,85kbps, cuando un segmento OFDM opera con una mayor inserción de redundancia, hasta 1787,28kbps, cuando un segmento opera con una menor cantidad de redundancia. Es por esto que el canal de 6Mhz solo puede transmitir hasta una tasa máxima de transmisión de 23234,64kbps utilizando sus 13 segmentos.

En la tabla 1.10 se encuentran los principales parámetros del subsistema de transmisión, a partir de los cuales es posible calcular todos los demás parámetros. Los parámetros fundamentales que se deben definir son el modo de operación y el intervalo de guarda que se aplican a todos los segmentos OFDM. Cada capa jerárquica debe definir cuantos segmentos utiliza, el tipo de modulación, la tasa de código interno y la profundidad del entrelazador temporal.

**Tabla 1. 10 Principales parámetros del sub sistema de transmisión del estándar ISDB-Tb**

Parámetros	Valores				
Número de Capas jerárquicas	Hasta 3				
Número de Segmentos OFDM	13				
Banda de Segmento OFDM	428,57kHz				
Compresión del cuadro OFDM	204 símbolos OFDM				
Frecuencia de muestreo IFFT	$(512/63) \times 10^6 = 8,12698\text{MHz}$				
Número de portadoras por segmento OFDM ( $\beta$ )	108 (modo 1) 216 (modo 2) 432 (modo 3)				
Número de muestras	2048 (modo 1) 4096 (modo 2) 8192 (modo 3)				
Intervalo de Guarda ( $I_G$ )	1/4	1/8	1/16	1/32	
Métodos de Modulación	DQPSK	QPSK	16 QAM	64 QAM	
Tasa de código interno	1/2	2/3	3/4	5/6	7/8
Profundidad del entrelazador de tiempo	0	4	8	16	modo 1
	0	2	4	8	modo 2
	0	1	2	4	Modo 3

### 1.6.3.3 Tasa útil de transmisión

Para minimizar los efectos de la interferencia entre símbolos, ocasionada por las diferentes señales que pueden aparecer en el canal, un intervalo de guarda es insertado entre dos símbolos OFDM. La inserción de un intervalo de guarda evita la ISI siempre y cuando la dispersión temporal del canal sea menor que el intervalo de guarda insertado. Un intervalo de guarda consiste en una parte de fin de símbolo OFDM que se copia en su inicio. Esa parte,  $I_G$ , es una fracción de tiempo del símbolo. De esa manera, la duración del intervalo de guarda está dada por:

$$T_{IG} = I_G T_U \quad (1.1)$$

A partir de ahí se puede concluir que la duración total del símbolo sería:

$$T_O = T_U + T_{IG} = T_U(1 + I_G) \quad (1.2)$$

Posteriormente, en cada intervalo  $T_O$  se transmite un nuevo símbolo OFDM, resultando una tasa de símbolos OFDM dada por:

$$R_O = \frac{1}{T_O} \quad (1.3)$$

Los segmentos de datos, que corresponden a un conjunto de portadoras que transportan datos en un segmento OFDM, es formado por 92, 192 o 384 portadoras para los modos de operación 1, 2 o 3 respectivamente. Cada portadora lleva un símbolo serial proveniente de la modulación utilizada. De esta manera la tasa útil de símbolos por segmentos estaría dada por:

$$R_{S\alpha} = \alpha R_O = \frac{\alpha}{(1 + I_G) N t_s} \quad (1.4)$$

Donde  $\alpha$  es el número de portadoras por segmento de datos. La tasa de cada segmento no se altera con la variación de  $\alpha$ , ya que la relación del número de portadoras por segmento de datos y el número de portadoras utilizadas para el cálculo de la IFFT, no depende del modo de operación, siendo:

$$N = \frac{64}{3} \alpha \quad (1.5)$$

De esa forma, la tasa de símbolos seriales por segmento de datos podría ser escrita como:

$$R_{S\alpha} = \frac{3}{64(1 + I_G)t_s} \quad (1.6)$$

Los parámetros representados hasta este momento se aplican a todos los segmentos que componen el símbolo OFDM. Los siguientes parámetros deben ser calculados para cada capa jerárquica utilizada. La tasa total de bits transmitidos por segmentos de datos estaría dada por:

$$R_{B\alpha} = \log_2 (M_x) R_{S\alpha} \quad (1.7)$$

Donde  $M_x$  representa el orden de modulación de la capa X. La tasa de código interno son dos códigos utilizados para proteger la información a través de la inserción de bits de redundancia. Al insertar dos bits de redundancia disminuye el flujo útil del sistema. Entonces la tasa de bits a la entrada del codificador convolucional estaría dada por:

$$R_{BCx} = r(x) R_{B\alpha} \quad (1.8)$$

Donde,  $r(x)$  es la tasa de código interno de la capa X.

Por otro lado el codificador *Reed Solomon*, llamado de código externo, inserta bytes de redundancia con el mismo objetivo de proteger la información. En un sistema con códigos en cascada, el decodificador de canal externo es utilizado para eliminar los errores residuales que no fueron corregidos por el decodificador de canal interno. Este codificador *Reed Solomon* recibe paquetes de datos de 188 bytes e inserta 16 bytes de redundancia, generando así un paquete de 204 bytes. Entonces la tasa útil de bits en un segmento de datos de una capa X estaría dada por:

$$R_{Bx} = \frac{188}{204} R_{BCx} \quad (1.9)$$

Relacionando las ecuaciones (1.6), (1.7), (1.8) y (1.9), se tiene que:

$$R_{Bx} = \frac{47 r(x) \log_2 (M_x)}{1088(1 + I_G)t_s} \quad (1.10)$$

Lo que permite determinar la tasa útil de datos en un segmento para cualquier configuración de subsistema de transmisión para el estándar ISDB-Tb. Estas tasas están representadas en la tabla 1.11:

**Tabla 1. 11 Tasa útil de datos de un segmento**

Modulación	Tasa de Código Interno	Tasa de Datos (Kbps)			
		Intervalo de Guarda 1/4	Intervalo de Guarda 1/8	Intervalo de Guarda 1/16	Intervalo de Guarda 1/32
<b>DQPSK QPSK</b>	<b>1/2</b>	280,85	312,06	330,42	340,43
	<b>2/3</b>	375,47	416,08	440,56	453,91
	<b>3/4</b>	421,28	468,09	495,63	510,65
	<b>5/6</b>	468,09	520,10	550,70	567,39
	<b>7/8</b>	491,50	546,11	578,23	595,76
<b>16-QAM</b>	<b>1/2</b>	561,71	624,13	660,84	680,87
	<b>2/3</b>	748,95	832,17	881,12	907,82
	<b>3/4</b>	842,57	936,19	991,26	1021,30
	<b>5/6</b>	936,19	1040,21	1101,40	1134,78
	<b>7/8</b>	983,00	1092,22	1156,47	1191,52
<b>64-QAM</b>	<b>1/2</b>	842,57	936,19	991,26	1021,30
	<b>2/3</b>	1123,43	1248,26	1321,68	1361,74
	<b>3/4</b>	1263,86	1404,29	1486,90	1531,95
	<b>5/6</b>	1404,29	1560,32	1652,11	1702,17
	<b>7/8</b>	1474,50	1638,34	1734,71	1787,28

### 1.6.4 Formato de compresión de audio y video

En el sistema ISDB-Tb al igual que en los otros estándares (ATSC, DVB-T e ISDB-T) existen tres tipos de fuentes: fuente de audio, fuente de video y fuente de datos, como se muestra en la Figura 1.18, estas fuentes tienen que ser comprimidas antes de ingresar al multiplexor de servicios, para que puedan ser comprimidas y codificadas estas dichas fuentes ISDB-Tb utiliza el estándar MPEG-4 H.264 para la fuente de video, y MPEG-4 AAC (Advanced Audio Coding) para la fuente de audio [13].

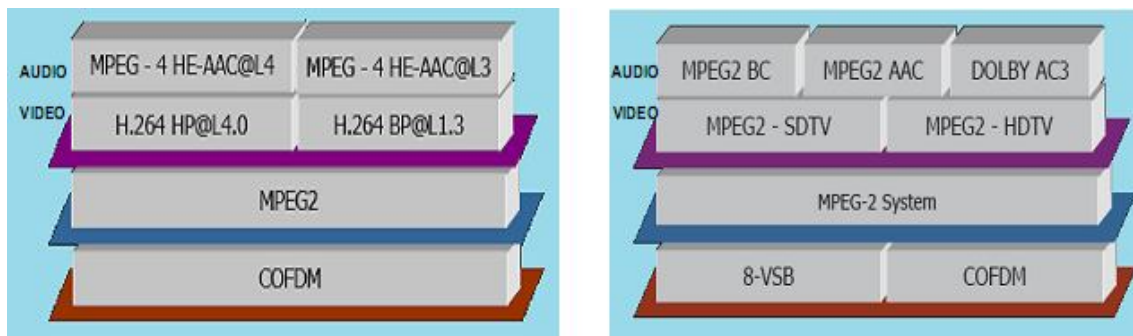
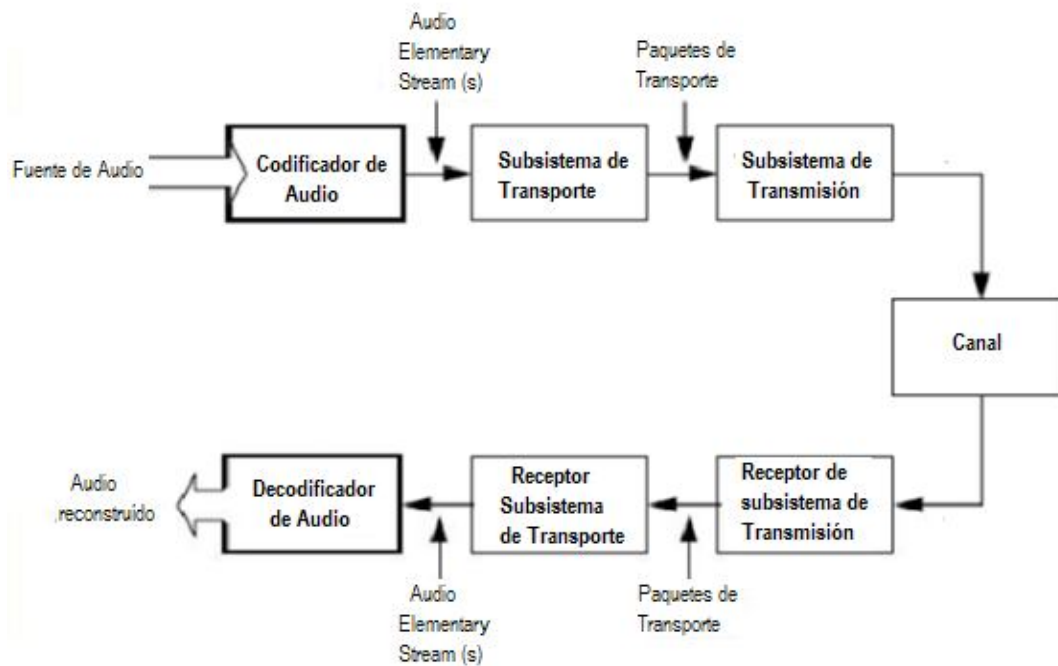


Figura 1. 16 Estándares de referencia utilizados por ISDB-Tb

#### 1.6.4.1 Sistema de compresión de audio

El sistema de compresión de audio, consta de un decodificador y un codificador de audio, el cual acepta señales de entrada analógica o digital y el bloque del subsistema de audio está antes del subsistema de transporte, el bloque del codificador recibe las señales del audio en dominio del tiempo y las convierte en el dominio de la frecuencia con el propósito de convertir la salida en una cadena de bits, para así llegar al subsistema de Transporte, el cual empaqueta los datos de audio y se pasa al subsistema de Transmisión el cual cambia los paquetes en una señal modulada en RF para luego ser recibida por el subsistema de Recepción y luego demodularla desempaquetar y decuantificar los flujos o tramas con los datos como se aprecia en la figura 1.19 [13].



**Figura 1. 17 Subsistema de audio dentro de un sistema de televisión digital**

#### 1.6.4.2 Sistema de compresión de video

A diferencia del estándar japonés que utiliza MPEG-2, el estándar brasileiro utiliza MPEG-4, que permite transmitir en el mismo canal un programa con calidad de alta definición (HDTV), informaciones de interactividad y programas adicionales con calidad de definición (SDTV). Además estas herramientas de compresión deben estar de acuerdo de manera obligatoria con la recomendación H.264 de la ITU-T.

La intención del proyecto H.264/AVC fue la de crear un estándar capaz de proporcionar una buena calidad de imagen con tasas binarias notablemente inferiores a los estándares previos (MPEG-2, H.263 o MPEG-4 parte 2), además de no incrementar la complejidad de su diseño.

La diferencia más considerable de MPEG-4 en base a las nuevas funcionalidades con otros estándares audiovisuales, es su modelo de representación audiovisual que se basa en objetos. Una escena basada en objetos es construida mediante objetos individuales que tienen relación tanto espacial como temporal, ofreciendo numerosas ventajas.



Estos objetos pueden existir de manera independiente o también pueden agruparse entre algunos de ellos y formar enlaces audiovisuales. A esta agrupación se la conoce como composición y el resultado es una escena MPEG-4. La ventaja de realizar esto es que tanto el audio como el video pueden ser fácilmente manipulados.

#### **1.6.4.2.1 Tipos de imágenes**

Se puede encontrar las mismas imágenes que en las normas precedentes (MPEG-2) (Imágenes I, P y B) y dos nuevas; la SP (conmutada P) y la SI (conmutada I) que sirven para codificar la transición entre dos flujos de vídeo. Utilizando predicción temporal o espacial permiten, pasar de un vídeo a otro sin enviar imágenes entrelazadas que gastan tiempo de procesamiento, tiene una ventaja permite la reconstrucción de valores específicos exactos de la muestra aunque se utilicen imágenes de referencia diferentes o un número diferente de imágenes de referencia en el proceso de predicción.

#### **1.6.4.2.2 Códec de Video**

El códec MPEG-4 tiene muchas mejoras con respecto a MPEG-2: incluye mejor estimación de movimiento y filtraje de desbloqueo y se pueden hacer composiciones de video sobre un fondo en tiempo real. Esto puede ser usado para una segmentación, ya que es posible separar internamente el fondo de la imagen sobre una escena. Su calidad y flujo de datos (20 kbps hasta 1000 kbps) es enormemente mejor que en el MPEG-1 y, generalmente, más competitivo que otras soluciones Web.

### **1.6.5 Middleware y capa física**

#### **1.6.5.1 Definición de Middleware**

El middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas. Funciona como una capa de abstracción de software distribuida, que se sitúa entre las capas de aplicaciones y las capas inferiores (sistema operativo y red). El middleware abstrae de la complejidad y heterogeneidad de las redes de comunicaciones subyacentes, así como de los

sistemas operativos y lenguajes de programación, proporcionando una API para la fácil programación y manejo de aplicaciones distribuidas. Dependiendo del problema a resolver y de las funciones necesarias, serán útiles diferentes tipo de servicios de middleware.

### 1.6.5.2 Middleware GINGA para ISDB-TB

Todos los estándares de tv digital utilizan un middleware para el desarrollo de las aplicaciones de interactividad. El estándar europeo DVB-T utiliza MHP (Multimedia Home Platform), el cual está basado en tecnología Java TV. El estándar americano utiliza el DASE (Advanced Common Application Platform) mientras que el estándar Japonés ISDB-T utiliza el ARIB B.23 (Association of Radioindustries and Businesses).

En el caso del estándar ISDB-TB se desarrolló el middleware llamado GINGA, el cual se encarga o permite el desarrollo de aplicaciones interactivas para televisión digital, en forma independiente a la plataforma de hardware de los fabricantes de terminales de acceso (set-top boxes). Ginga es el fruto del desarrollo de proyectos de investigación coordinados por los laboratorios Telemidia de la PUC-Rio y LAViD de la UFPB.

En la figura 1.20 se muestra una comparación de los diferentes middleware de tv digital.

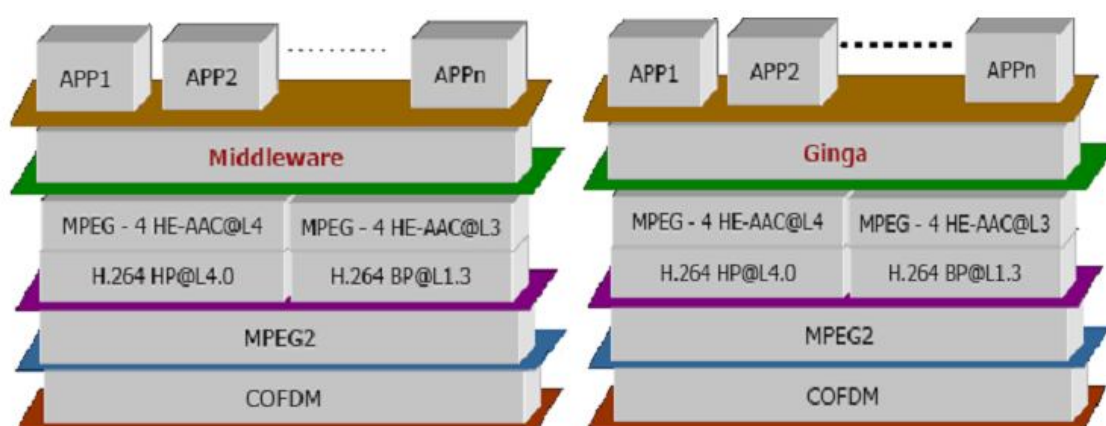


Figura 1. 18 Equivalencia del Middleware de los estándares ATSC, ISDB-T y DVB-T con Ginga de ISDB-Tb

### 1.6.5.3 Características de GINGA

El middleware abierto Ginga está dividido en tres módulos, que permiten el desarrollo de aplicaciones siguiendo dos paradigmas de programación diferentes. Dependiendo de las funcionalidades requeridas en cada proyecto de aplicación, un paradigma será más adecuado que otro. Estos módulos son Ginga-CC (Common Core, Núcleo Común), Ginga-J (para aplicaciones procedurales Java) y Ginga-NCL (para aplicaciones declarativas NCL) [14].

En lo relacionado al lenguaje NCL, estas son las siglas de Nested Context Language. Se trata de un lenguaje declarativo que provee facilidades para especificar aspectos de interactividad, sincronismos espacial/temporal entre objetos de multimedia, adaptabilidad y soporte para múltiples dispositivos, es decir construir aplicaciones.

Estas aplicaciones se pueden ejecutar de dos maneras en el STB. En principio las aplicaciones a ejecutar en los STBs vienen dentro de la señal que reciben, en un canal de datos, dedicado para las aplicaciones. Este canal, recibe el nombre de carrusel de datos, dado que se va iterando sobre los datos de manera circular enviándolos constantemente. Esto es necesario porque cuando se sintoniza un canal de televisión, el STB debe poder descargar el contenido completo del carrusel. Luego ya sea por voluntad del usuario, o por eventos emitidos desde el broadcaster la aplicación se ejecuta.

La otra manera para ejecutar aplicaciones es cargarlas a través de un dispositivo USB y ejecutarlas como hoy en día en un dvd-player podemos mirar fotos o escuchar música

Una característica importante de este innovador middleware es que permite la interoperabilidad de los sistemas de ISDBT, DVB-T y ATSC. Además permite su uso tanto en el modelo estándar como también en alta definición y también permite la posibilidad de transmitir aplicaciones interactivas en distintos niveles.

Ginga además acepta TV, celulares, PDAs o TV Pagada, entre otros, lo cual hace que los contenidos de TV digital sean exhibidos sin importar el sistema de recepción.

Una de las principales ventajas de GINGA es que ofrece código abierto, una interfaz gráfica y una conexión a internet.

#### 1.6.5.4 Capa Física

Como se mencionó anteriormente GINGA está dividida en tres módulos: GINGA-NCL, GINGA-J y GINGA-CC. A continuación se muestra la arquitectura del software para el middleware GINGA con sus respectivos módulos:



Figura 1. 19 Arquitectura Middleware GINGA

#### 1.6.5.5 GINGA-NCL

GINGA-NCL provee una infraestructura de presentación para aplicaciones declarativas escritas en el lenguaje NCL (Nested Context Language). Este lenguaje con facilidades para los aspectos de interactividad, sincronismo, adaptabilidad, soporte a múltiples dispositivos y soporte a la producción de programas interactivos en vivo no-lineales. La estructura de este lenguaje separa el contenido de la estructura de un aplicativo permitiendo definir objetos de media estructurados y relacionados tanto en tiempo y espacio.

Los componentes de este subsistema son:



Figura 1. 20 Subsistema GINGA-NCL

Todo documento NCL posee la siguiente estructura:

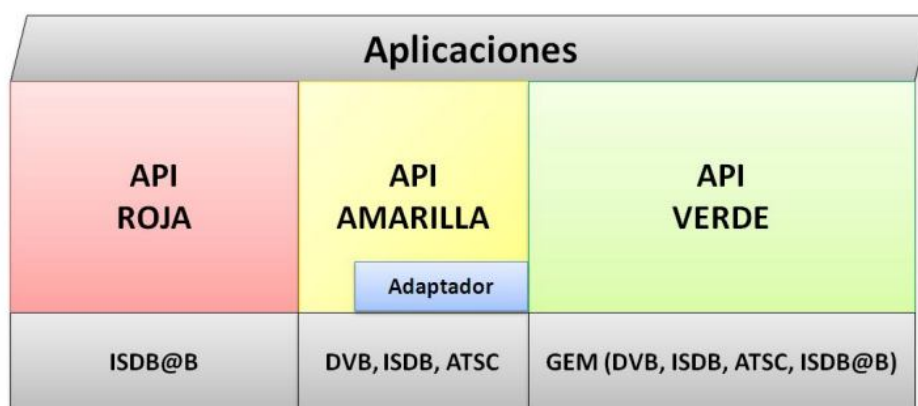
- Una cabecera de archivo NCL
- Una sección de cabecera del programa, donde se define las regiones, los descriptores, los conectores y las reglas utilizadas por el programa.
- El cuerpo del programa donde se definen los contextos, los nodos de media, y los otros elementos que definen el contenido de la estructura del programa.
- Por lo menos una parte que indique por donde el programa comienza a ser exhibido.

#### 1.6.5.6 Ginga-J

Este módulo provee una infraestructura de ejecución de aplicaciones basadas en lenguaje Java, llamadas Xlet con facilidades específicamente creadas para TV Digital.

Ginga-J es un subsistema lógico del Sistema Ginga que procesa aplicaciones procedimentales (Xlets Java). Un componente clave del ambiente de aplicaciones procedurales es el mecanismo de ejecución de contenido procedimental, que tiene como base la máquina virtual de Java.

Ginga-J está basado en tres grupos de API's llamados Verde, Amarillo y Azul. En la figura 1.23 se muestra los conjuntos de APIs del Ginga-J.



**Figura 1. 21 Middleware de estándares de TDT a nivel mundial**

Los API's Verdes son compatibles con los API's GEM. Los API's Amarillos son extensiones propuestas para cumplir con los requisitos específicos de Brasil que se pueden implementar mediante el uso de un

adaptador de software utilizando los API's Verdes. Los API's Azules solo pueden ser ejecutadas en el entorno del middleware Ginga

#### 1.6.5.7 Ginga-CC

Ginga Common Core, es el subsistema lógico que provee toda funcionalidad común al soporte de los ambientes de programación declarativos, GINGA-NCL, e imperativo, GINGA-J. En la figura 1.24 se ilustra la arquitectura del Ginga-CC.



Figura 1. 22 Estructura del Middleware GINGA e identificación del Ginga CC

La arquitectura del sistema permite que únicamente el módulo Ginga-CC deba ser adaptado a la plataforma donde será implementado. Ginga-CC provee un nivel de abstracción de la plataforma de hardware y del sistema operativo, accesible a través de las APIs.

## CAPÍTULO II

### TRANSPORT STREAM

#### 2.1 TRANSPORT STREAM PARA ISDB-T

En ISDB-T como en ATSC y DVB-T se adoptó el estándar MPEG-2 como tecnología de multiplexación para cualquier tipo de contenido o servicio. En este estándar todos los contenidos de audio, video y datos son multiplexados y empaquetados en un flujo de transporte conocido como **Transport Stream**.

Estas señales de audio o video ya comprimidas resultan de un flujo de bits denominado Elementary Stream, luego se divide a estos flujos en paquetes elementales PES con información elemental de cabecera. A partir de estos paquetes se crea el flujo de transporte TS (transport stream) y multiplexados, por otro lado, los flujos de datos son convertidos al formato de Sección y después al formato TS y multiplexados.

En la figura 2.1 se puede ver claramente cómo se van empaquetando la información hasta llegar al TS. [1]

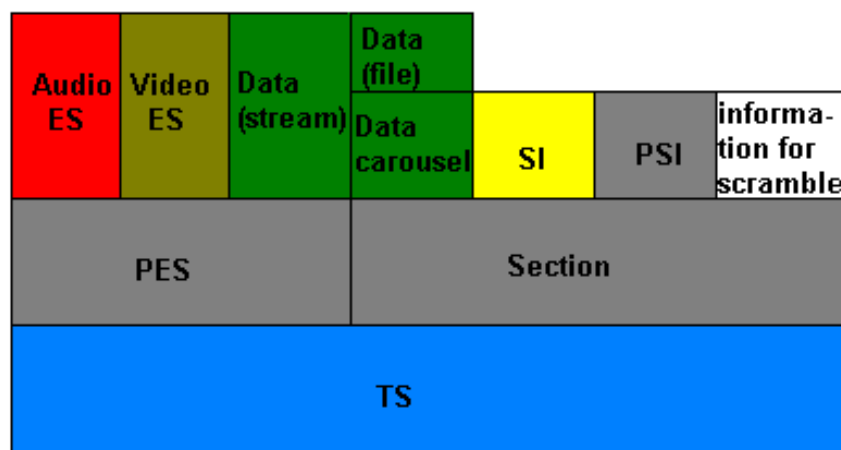


Figura 2. 1 Formato Multiplexado en el sistema ISDB-T

Como se mencionó en el capítulo anterior, luego de que se genera el flujo de transporte, ISDB-T utiliza una modulación OFDM, interleaving en el dominio del tiempo y de la frecuencia y códigos de corrección de errores. El esquema de modulación utilizado es “Band Segment Transmission-OFDM” y se

compone de 13 segmentos OFDM. Este tipo de modulación soporta tres clases de transmisiones jerárquicas de hasta tres capas (A, B y C), donde los parámetros de transmisión pueden ser modificados en cualquiera de las tres capas. Cada segmento de transmisión tiene un ancho de banda de 6/14 MHz y sus parámetros de transmisión pueden ser establecidos de manera individual para lograr una composición flexible del canal.

Ya que el objetivo es obtener una interfaz entre múltiples flujos de transporte (TS) MPEG-2 y el sistema de transmisión BST-OFDM, estos flujos de transporte son nuevamente multiplexados en un único TS. Es importante mencionar que la información acerca del control de transmisión, configuración del segmento del canal, parámetros de transmisión, etc..., son enviados hacia el receptor en forma de una señal TMCC (transmission multiplexing configuration control) [2].

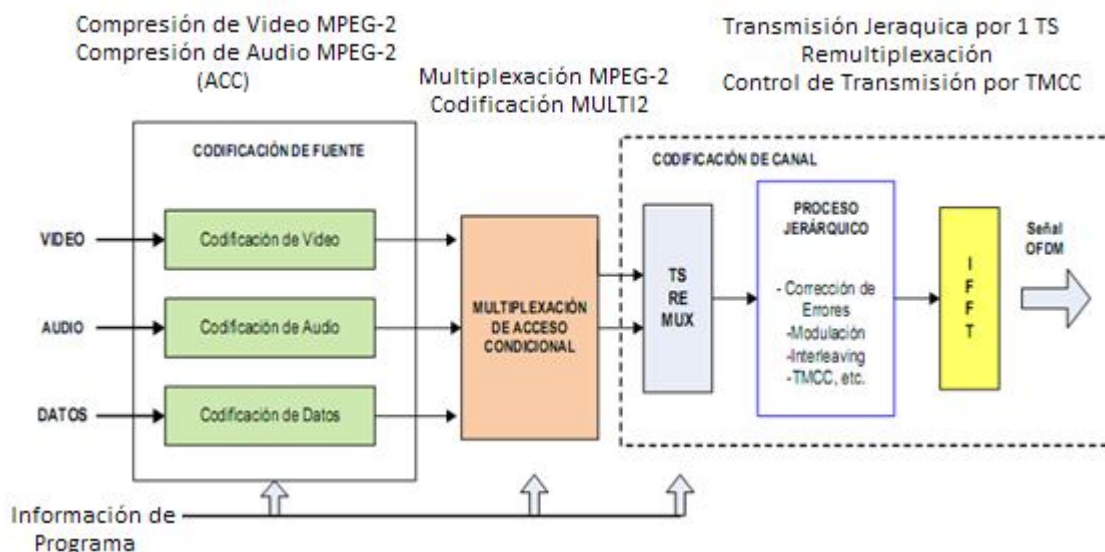


Figura 2. 2 Sistema ISDB-T

### 2.1.1 Parámetros básicos de transmisión para ISDB-T

ISDB-T cuenta con tres modos de transmisión con diferentes intervalos de portadora con el fin de hacerle frente a la variedad de condiciones que se presentan, como los intervalos de guarda variables y el desplazamiento Doppler que ocurre en la recepción móvil.



Un segmento OFDM corresponde a un espectro de frecuencia que tiene un ancho de banda de 6/14 MHz (430 kHz aproximadamente). En el Modo 1, un segmento está conformado de 108 portadoras, mientras que el Modo 2 y Modo 3 tienen dos y cuatro veces el número de portadoras, respectivamente. La difusión de televisión emplea 13 segmentos con un ancho de banda de alrededor de 5.6 MHz, para la transmisión. En cambio, la radiodifusión terrestre digital de audio emplea solamente uno o tres segmentos.

Una señal digital es transmitida en forma de conjuntos de símbolos. Un símbolo está formado de 2 bits en QPSK y DQPSK, 4b en 16 QAM, y 6 bits en 64QAM. De esta manera se evita la interferencia de portadoras. Una trama OFDM está compuesta de 204 símbolos con los diferentes tiempos de intervalo de guarda dependiendo del modo de transmisión. La duración del time interleave en tiempo real depende de los parámetros establecidos en la etapa de digitalización de la señal y de la duración del intervalo de guarda.

## **2.2 TRANSPORT STREAM PARA ISDB-Tb**

La diferencia fundamental entre el estándar japonés con el estándar brasilero es la utilización de tecnologías de compresión más avanzadas y su middleware GINGA. En cuanto a la modulación y el transporte, ambos sistemas lo realizan en base al estándar MPEG-2 que explicaremos en mayor detalle en esta sección.

### **2.2.1 Multiplexación y formación del TRANSPORT STREAM**

Como se mencionó en la sección 2.1, luego de que la información de audio, video y datos ha sido comprimida bajo las diferentes tecnologías, las señales se tienen que multiplexar. MPEG-2 es el estándar utilizado para realizar esta operación y sincronizar los streams individuales de video, audio y datos para que puedan ser reproducidos correctamente.

Cada señal de video o audio comprimida resulta en los antes mencionados Elementary Streams. Estos flujos elementales forman paquetes de longitud variable llamados PES.

### **2.2.1.1 Packetized elementary Stream**

Los PES son el nivel básico del sistema de capas de MPEG-2, son los encargados de transportar los distintos datos de una aplicación, es decir datos de audio o video. Estos paquetes contienen información de cabecera, que sirve para identificar el tipo de datos que transportan, y carga útil. Todos los PES que formen parte de un mismo programa comparten una misma base de tiempos y en cuanto a la carga útil, esta contiene un solo frame de video o audio que tiene una longitud de 64 Kbyte o puede ser mayor en algunos casos especiales.

En cuanto a la cabecera, esta contiene un prefijo de código de comienzo (Start Code Prefix) de 24 bits que siempre está formado por el código 00 00 01 que sirve para identificar el comienzo de un paquete PES, seguido por una cadena de ID (Stream identifier) que identifica el tipo de contenido del paquete, es decir, indica si el paquete contiene información de video, audio o datos.

El PES cuenta con una cabecera opcional que es adaptada a los requerimientos del elementary stream que se está transmitiendo. Esta cabecera está formada por 12 bits que constituyen 11 banderas que indican cuales componentes están presentes actualmente en los “campos opcionales” en la cabecera opcional PES y cuáles no lo están. Los más importantes son el PTS (Presentation Time Stamps) y el DTS (Decoding Time Stamps) que sirve para la correcta sincronización del audio y el video. Con esta información de sincronización en la cabecera, el decodificador sabe cuando decodificar y presentar los frames adecuadamente. A continuación se muestra la estructura de los PES [3].

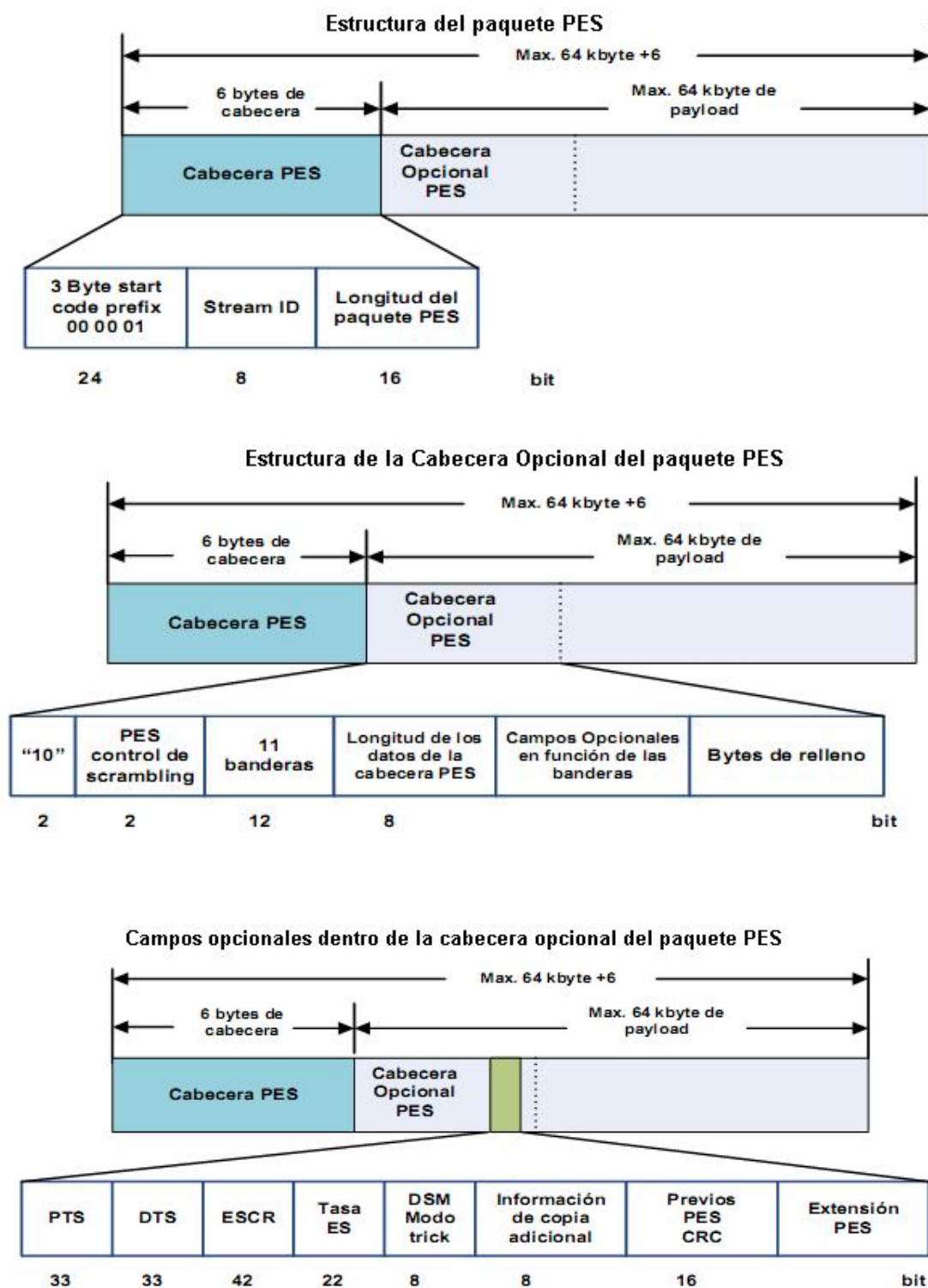
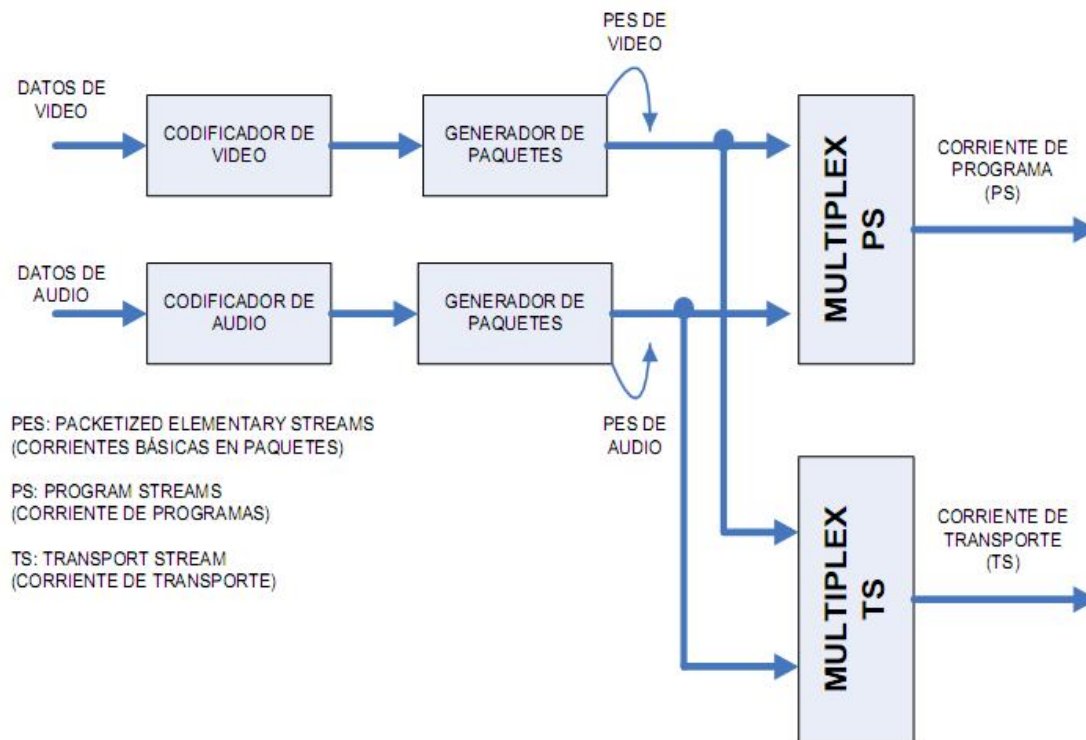


Figura 2. 3 Estructura del paquete PES

Con los paquetes PES se crean dos tipos de multiplex: el flujo de programa (PS) y el flujo de transporte (TS) como se muestra en la figura 2.4.



**Figura 2. 4 Sistema de transporte MPEG-2**

### 2.2.1.2 Program Stream

Son el resultado de la combinación de uno o más PES de longitud variable y relativamente grande en un único stream con una base de tiempo en común ya que los distintos PES que forman parte del PS se codifican usando un STC (System Time Clock). Este stream puede ser un stream de video con su correspondiente stream de audio. Antes de ser multiplexados, los ES constituyen paquetes para formar Video PES y Audio PES.

Cada PS tiene sus propias cabeceras y puede contener hasta 32 streams de audio, 16 streams de video, 16 streams de datos y varios streams con información variada para manejo interno. Es usada para aplicaciones de video playback y algunas aplicaciones de red.

### 2.2.1.3 Program Stream y Transport Stream

Al hablar de Program Stream y Transport Stream, el objetivo es como MPEG-2 envía la información y no como está codificada. Entonces lo primero que hay que tener en cuenta que si un Video es un Program Stream o un

Transport Stream no tiene nada que ver con la calidad de codificación del mismo.

El Program Stream es generalmente usado en aplicaciones de almacenamiento, en cambio el Transport Stream es usado para difusión (broadcasting) [4].

Se trata de un problema de formatos y hay que diferenciar que si uno tiene que transmitir el contenido o programa de un canal específico, no implica que necesariamente se tenga que transportar por medio del Program Stream.

Un program Stream contiene uno y solo un contenido de un canal determinado. Es como un carro con espacio para un único pasajero que no está preparado para colisión alguna y se desliza por una carretera muy tranquila.

Un transport stream en cambio puede contener contenido de múltiples canales simultáneamente. Se lo puede ver como un gran bus que puede llevar uno o más pasajeros y posee una buena suspensión y mecanismos anticollisiones que le permiten soportar una carretera que puede provocar accidentes y contratiempos.

### **2.2.2 Definición del Transport Stream**

El objetivo de MPEG es poder agrupar de 6 hasta 20 programas para formar una señal común de datos multiplexada que es transmitida vía satélite o cable o cualquier tipo de enlace terrestre. Para que esto sea posible, se dividen los paquetes PES en paquetes más pequeños de longitud constante de 188 bytes cada uno, con 4 bytes de cabecera y 184 bytes de payload. La carga útil o payload contiene el audio, video y datos mientras que la cabecera contiene información muy importante para la transmisión y sincronización de los paquetes. Por ejemplo, el primer byte de la cabecera se lo denomina “sync byte” y tiene un valor de 0x47.

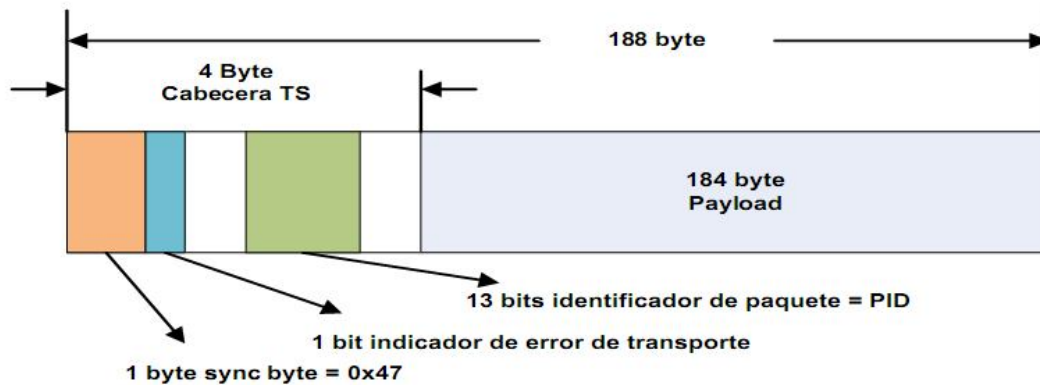


Figura 2. 5 Paquete de TS

Cuando los flujos de audio y video se encuentran divididos en paquetes de transporte, estos son multiplexados o combinados con contenidos similares para otros servicios. Al resultado de esta multiplexación compuesta por uno o más servicios se la conoce como flujo de transporte TS (transport stream) [5].

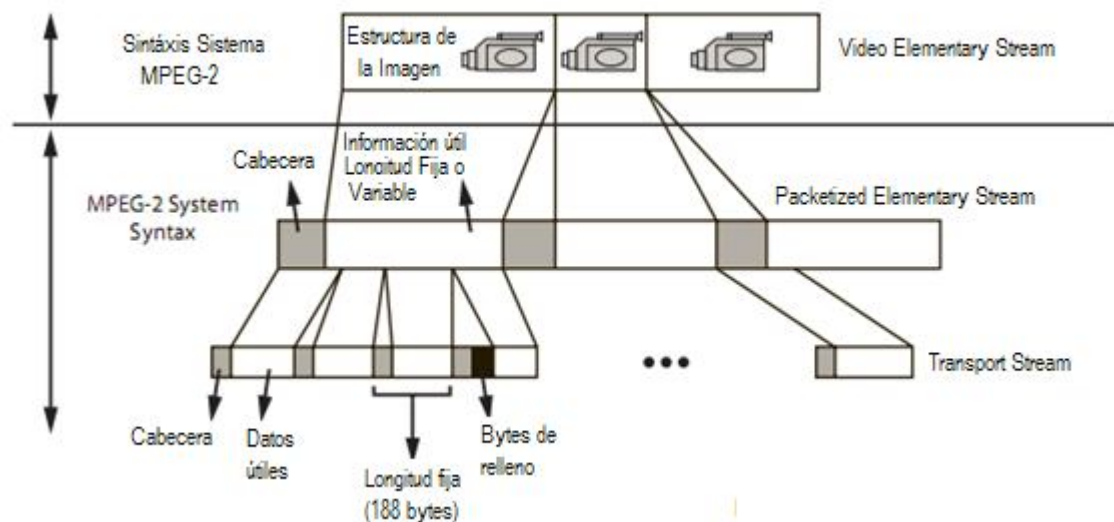


Figura 2. 6 Creación del Transport Stream

El flujo de transporte TS es un formato que se especifica en MPEG-2, es el resultado de combinar uno o más programas con una o más bases de tiempos independientes, en un solo flujo. Su objetivo es permitir multiplexar los distintos trenes de datos y sincronizar el resultado y se define como unidad fundamental de información de la imagen y está representada por un bloque de coeficientes de DCT (Transformada de coseno discreto), que tiene un tamaño de 8x8 pixeles los cuales representan datos, de los cuales el coeficiente que tenga mayor valor es enviado primero ya que tiene mayor precisión para

representar la información del bloque. A continuación de este se envían el resto de coeficientes.

Cada paquete en el Transport Stream puede contener audio, video, tablas o información y es identificado por medio de un número denominado PID (packet identifier). Este número le permite clasificar al decodificador todos los paquetes del Transport Stream. Se puede comparar el mecanismo de transporte de MPEG-2 con el protocolo de transporte de Internet IP ya que MPEG-2 contiene información que ha sido dividida en paquetes de transporte cada uno con su header y un payload. El proceso consiste en transformar flujos de datos, audio y video análogo en un solo flujo de transporte

### **2.2.3 Sincronización del TRANSPORT STREAM**

Para esta tarea se utiliza el *sync byte* mencionado anteriormente además la sincronización del Transport Stream está basada en el STC de 27 MHz del codificador. Para poder garantizar una adecuada sincronización en el proceso de decodificación, el reloj del decodificador debe estar bloqueado al STC del codificador. Para poder lograr esto, el codificador debe insertar en el Transport Stream una marca de tiempo (time stamp) a 27 MHz para cada programa. Esta marca de tiempo se la conoce como la referencia de reloj del programa (PCR) y por medio de la misma, el decodificador genera una señal de reloj local de 27 MHz que está igualada a la señal del codificador.

En MPEG, la sincronización del decodificador ocurre solo cuando se han recibido 5 paquetes del Transport Stream. Como se mencionó en la sección anterior en la cabecera existe un sync byte que tiene un valor de 0x47 que permite al decodificador identificar el inicio de cada paquete. Para esto el decodificador debe desplazarse hacia delante y hacia atrás en busca de otro byte con un valor de 0x47 con un espacio de 188 bytes para identificar que se trata de un sync byte y no de un byte que contenga el dato 0x47 [5].

### **2.2.4 Remultiplexación y generación del MPEG-2 Transport Stream**

En la etapa anterior se generaron los paquetes de transport stream en el multiplexor MPEG-2, luego estos paquetes son entregados a un remultiplexor TS para formar un nuevo stream de datos que contiene todos los programas que se lo denomina "MPEG-2 transport stream".

El MPEG-2 Transport Stream contiene en un solo flujo, todos los paquetes de Transport Stream de 188 bytes cada uno que contienen las señales de audio, video y datos correspondientes a los diferentes programas transmitidos. Las tasas de transmisión de estos flujos de transporte pueden llegar a ser hasta de 40Mbps ya que en estos flujos de transporte pueden existir hasta 20 programas. Pero estas tasas de transmisión no pueden ser constantes ya que la velocidad va a depender del tipo de contenido de cada paquete que puede ser de audio, video y datos [6].

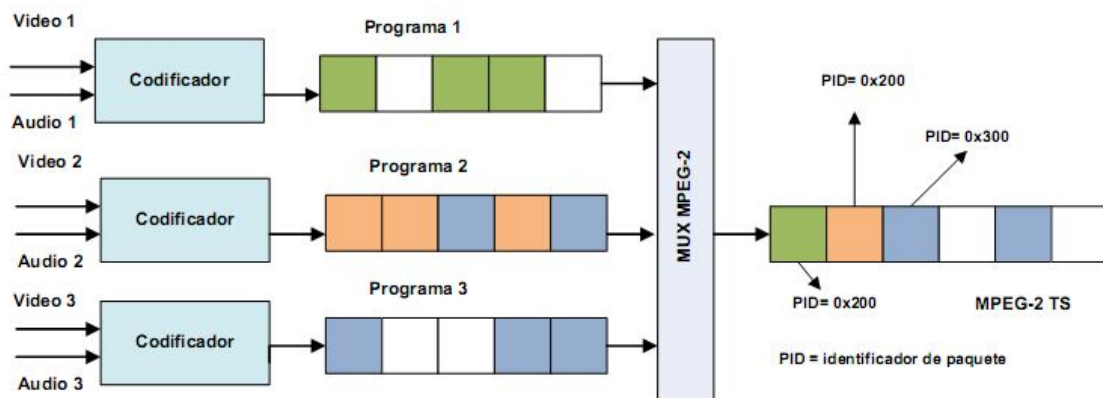


Figura 2. 7 Remultiplexación de paquetes TS

## 2.3 ESTRUCTURA DEL TRANSPORT STREAM

### 2.3.1 Estructura del Transport Stream para el contenido de video

Como se mencionó anteriormente, el TS contiene información correspondiente al video, al audio y a los datos que se transmiten en TV digital. En esta sección describiremos la configuración y codificación que debe tener la estructura del contenido de video para que pueda ser procesado dentro la multiplexación de las señales.

Estas configuraciones incluyen los parámetros para las señales en la entrada del codificador y las restricciones al proceso de codificación aplicables al sistema brasileño de televisión digital terrestre (SBTVD).

#### 2.3.1.1 Características de la señal de video

Las señales digitales de video que se manejan en este estándar son SD (Definición Estándar) o HD (Definición alta). Estas señales son compuestas por



tres tipos de señales que representan la luminancia de la escena, las características de crominancia y las características de saturación de la escena. Las señales de crominancia y saturación también se las denomina señales de complemento de color. Todas estas señales se las puede obtener a partir de las siguientes ecuaciones:

$$Y = INT[219 DE'_Y + 16D + 0,5]$$

$$C_R = INT[224DE'_R + 128D + 0,5]$$

$$C_B = INT[224DE'_B + 128D + 0,5]$$

Donde,

INT A, representa la parte entera de un número real A;

Y, es el valor numérico de la señal digital de luminancia;

$C_R$  y  $C_B$ , son los valores numéricos de las señales digitales de complemento de color, con relación al rojo y al azul, respectivamente.

D, es el valor numérico sustituido en las ecuaciones por "1" o "4" dependiendo de la cantidad de bits que se utilizan en la cuantización, 8 o 10 bits respectivamente.

$E'_Y$ ,  $E'_R$  y  $E'_B$ , son los valores numéricos de las señales analógicas de luminancia y de los complementos de color que se obtienen de las formulas de la tabla 2.1.

**Tabla 2. 1 Ecuaciones de las señales analógicas y de los complementos de color**

SD	HD
$E'_Y = 0,229 E'_R + 0,587E'_G + 0,114E'_B$	$E'_Y = 0,2126 E'_R + 0,7152E'_G + 0,0722E'_B$
$E'_{CR} = (E'_R - E'_Y)/1,402$	$E'_{CR} = (E'_R - E'_Y)/1,5748$
$E'_{CB} = (E'_B - E'_Y)/1,772$	$E'_{CB} = (E'_B - E'_Y)/1,8556$

Los valores  $E'_R$ ,  $E'_G$  y  $E'_B$  representan los niveles de tensión normalizados resultantes de la corrección gama.

Por ejemplo, en el caso de que queremos conocer el nivel de luminancia y crominancia del color amarillo para una señal digital de definición estándar. Entonces se parte de que el amarillo se forma con 1 voltio de tensión de rojo, 1 voltio de tensión de verde y 0 voltios de azul. Entonces tenemos que las señales analógicas de luminancia y complemento de color se obtienen a partir de las siguientes expresiones:

$$E'_Y = 0,229(1) + 0,587(1) + 0,114(0)$$

$$E'_Y = 0,816V$$

$$E'_{CR} = \frac{1 - 0,816}{1,402} = 0,13V$$

$$E'_{CB} = \frac{0 - 0,816}{1,772} = 0,46V$$

Posteriormente se calculan los valores numéricos de las señales digitales, suponiendo que el cuantizador utilizado es de 8 bits:

$$Y = INT[219(1)(0,816) + 16 + 0,5]$$

$$Y = 195$$

$$C_R = INT[224(0,13) + 128 + 0,5]$$

$$C_R = 157$$

$$C_B = INT[224(0,46) + 128 + 0,5]$$

$$C_B = 231$$

### 2.3.1.2 Parámetros de la señal de video

Por otro lado, es importante recalcar que las señales de video deben ser obligatoriamente cuantizadas por 8 bits o 10 bits. La dirección de exploración de las figuras que componen un video debe ser de izquierda a derecha y de arriba hacia abajo.

Las señales digitales de video se componen de varios parámetros como número de líneas, número de líneas activas, sistema de exploración, frecuencia de cuadro, frecuencia de campo, relación de aspecto, frecuencia de línea, frecuencias de muestreo (para señales de luminancia y de complemento de color), número de muestras por línea activa (para señales de luminancia y de complemento de color). Estos parámetros deben estar ajustados a los que se indican en las tablas 2.2 y 2.3 [7].

**Tabla 2. 2 Parámetros de la señal de video**

No. de líneas	No. de líneas activas	Sistema de barradura	Frecuencia de cuadro	Frecuencia de campo	Relación de aspecto	Frecuencia de línea
525 (SD)	483	Entrelazado	30/1,001 Hz	60/1,001 Hz	16:9 o 4:3	15,750/1,001 kHz
525 (SD)	483	Entrelazado	60/1,001 Hz		16:9	31,500 /1,001kHz
750 (HD)	720	Programado	60/1,001 Hz		16:9	45,500 /1,001kHz
1125 (HD)	1080	Entrelazado	30/1,001 Hz	60/1,001 Hz	16:9	33,750 /1,001kHz

**Tabla 2. 3 Parámetros de la señal de video**

Frecuencia de muestreo		Número de muestras por línea		Número de muestras activas por línea	
<i>Señal de luminancia</i>	<i>Señal de complemento de color</i>	<i>Señal de luminancia</i>	<i>Señal de complemento de color</i>	<i>Señal de luminancia</i>	<i>Señal de complemento de color</i>
13,5MHz	6,75 MHz	858	429	720	360
27MHz	13,5MHz	858	429	720	360

74,25/1,001 MHz	37,125/1,001 MHz	1 650	825	1 280	640
74,25/1,001 MHz	74,25/1,001 MHz	2 200	1 100	1 920	960

### 2.3.1.3 Codificación de la señal de video

Antes que nada hay que considerar que la señal de video debe usar obligatoriamente herramientas de compresión que estén de acuerdo con la ITU-T Recomendación H.264.

El objetivo principal de la compresión de video es representar con un menor número de bits los datos originales para aumentar las tasas de transmisión por lo que se aprovecha la redundancia que existe entre cuadros y pixeles sucesivos entre sí respectivamente, es decir, que el codificador envía la diferencia existente entre la imagen previa y la actual. La tecnología utilizada para realizar esta compresión es la MPEG.

MPEG define la sintaxis de las señales de audio y video, describe su estructura, contenido y regula el funcionamiento de decodificadores estandarizados. Es importante mencionar que MPEG no define algoritmos de codificación de las señales.

#### 2.3.1.3.1 Tipos de Imágenes MPEG

En un inicio MPEG definió tres tipos de imágenes que son el soporte de la codificación diferencial y bidireccional. Estos tipos de imágenes son I (intra), P (previstas) y B (bidireccionales).

*Imagen tipo I (intra).-*

Imagen que no necesita información adicional para su codificación. Son codificadas sin referencia a otras imágenes, contiene todos los elementos básicos para su reconstrucción por el decodificador (inicia el lazo de predicción) y son por ello, el punto de entrada obligatorio para el acceso a una secuencia, las imágenes I se codifican como una única imagen utilizando solo la información de la imagen, en términos de datos transmitidos es la imagen más grande. Además son utilizadas para

facilitar la captura del canal cuando se apaga el decodificador o se cambia el canal.

*Imagen P (Previstas).-*

Son imágenes de predicción y se codifican con respecto a las imágenes I o P anteriores, gracias a las técnicas de predicción con compensación de movimiento. Su tasa de compresión es mayor que la de las imágenes I, las imágenes P requieren aproximadamente la mitad de los datos de las imágenes I.

*Imagen B (Bidireccionales).-*

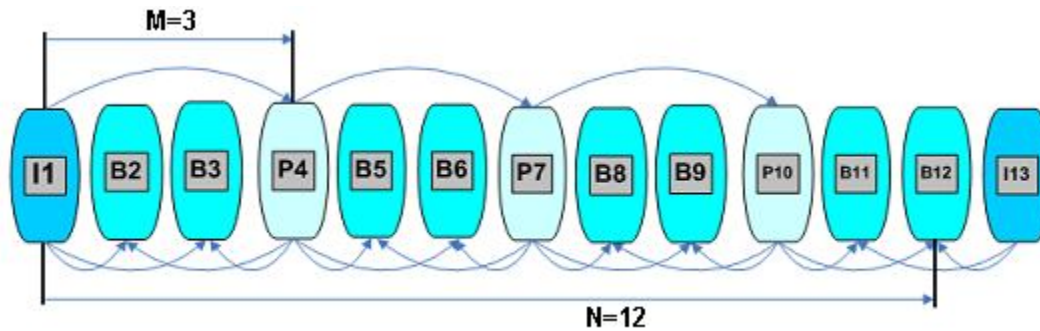
Son imágenes de predicción bidireccional. Para su formación, toman información tanto de una imagen futura (P) como de una imagen previa (I), se codifican por interpolación. Este tipo de imagen es el que ofrece el factor de compresión más alto, que generalmente es de una cuarta parte de los datos de las imágenes I, como no se utiliza para describir otras imágenes, las imágenes B no propagan los posibles errores de codificación.

El factor de compresión, la posibilidad de accesos aleatorio, el tiempo de codificación y la calidad dependen de la complejidad del codificador, es decir depende de si el codificador puede procesar imágenes tipo I, tipo I y P o tipo I,P y B ya que cada configuración obtendrá resultados diferentes.

Además existen parámetros como M y N que definen la manera en como las imágenes I, P y B se encadenan.

- M es la distancia en número de imágenes entre dos imágenes P sucesivas.
- N es la distancia entre dos imágenes I sucesivas.

Estos parámetros nos permiten asegurar una calidad satisfactoria, una resolución aceptable y una velocidad de transmisión adecuada. Por ejemplo para alcanzar una velocidad de transmisión de 1,5Mbps y una resolución <0.5s se utiliza parámetros M=3 y N=12



**Figura 2. 8 Encadenamiento de las imágenes I, P y B**

Si analizamos la figura del encadenamiento de las imágenes tenemos una secuencia de 12 imágenes compuesta por 1 imagen tipo I, 3 imágenes tipo P y 8 imágenes tipo B. En este caso el factor de compresión es mayor porque existe una mayor cantidad de imágenes tipo B.

Como sabemos el estándar ISDB-TB utiliza MPEG-4 para la compresión del video que se define en la Recomendación H.264 de la ITU. MPEG-4 define dos tipos más de imágenes que son SP (Switching P) y SI (switching I) [8].

*Imagen SP (Switching P).-*

Una imagen tipo SP se codifica de forma que permite la transición entre dos streams que representan una misma secuencia, con calidad diferente, o también permite avanzar o retroceder una imagen de un mismo stream, transmitiendo así menos información que una imagen I.

*Imagen SI (Switching I).-*

Una imagen tipo SI representa un punto de sincronismo para una transición entre dos streams, diferenciándose de las imágenes tipo SP por presentar todos los macrobloques codificados utilizando predicciones intra.

### 2.3.1.3.2 Técnicas de predicción o compresión

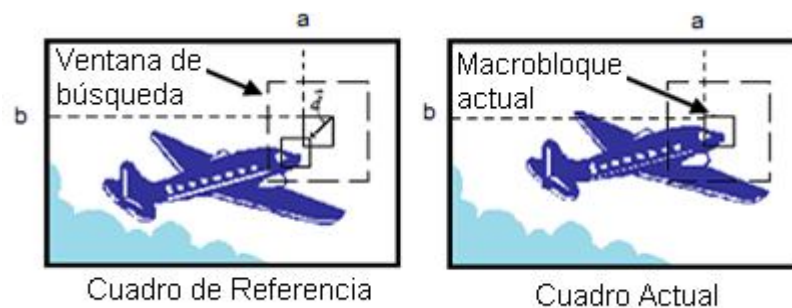
Para la compresión de video se exploran dos tipos de redundancias para reducir la representación del video original.

*Codificación Inter-frame o temporal.-*

La redundancia temporal consiste en el aprovechamiento de las similitudes existentes entre cuadros sucesivos que forman una imagen

dinámica con el objetivo de reducir considerablemente la cantidad de información necesaria para su transmisión o almacenamiento.

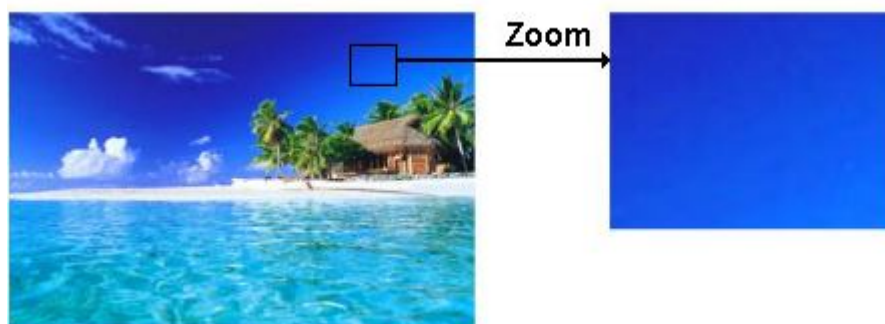
A esta codificación se la conoce como técnica de predicción de compensación de movimiento, consiste en la deducción de la mayoría de imágenes de una secuencia de video, comparando un determinado cuadro con su antecesor adicionando un mínimo de información, se aplica en ambas direcciones hacia adelante o causal (forward) y hacia atrás o no causal (backward), para soportar el acceso aleatorio al video almacenado se definen tres tipos de imágenes o cuadros: codificados internamente (I), predictivos (P) e interpolados bidireccionalmente (B). En la figura 2.9 se puede observar este proceso [9].



**Figura 2. 9 Codificación inter-frame**

#### *Codificación Intra-frame o espacial.-*

Consiste básicamente en la semejanza de dos pixeles adyacentes de una misma imagen. En la figura 2.10 se puede apreciar que existe una variación mínima entre los pixeles señalados.



**Figura 2. 10 Codificación intra-frame**

En esta codificación se utilizan técnicas de codificación basadas en la DCT. La DCT utiliza la transformada rápida de Fourier (FFT) que consiste en

transformar una serie de puntos del dominio espacial al dominio de la frecuencia

### 2.3.1.3.3 Perfiles

Un perfil establece el grado de complejidad de la codificación de video. Los perfiles definen un subconjunto de la sintaxis del flujo de bits que se adoptan buscando, por lo general, determinada clase de aplicaciones. MPEG-4 posee una flexibilidad de aplicaciones debido a que posee perfiles que define un stream de bits codificados. Los perfiles soportados son:

**Perfil Baseline:** Este perfil debe contener slices del tipo I o P. Soporta códigos de compresión variable basados en códigos Huffman, orden flexible de macrobloques. Este perfil es utilizado para aplicaciones conversacionales como video conferencias y video en estaciones móviles.

**Perfil Main:** Este perfil debe contener slices del tipo I, P o B solamente, además soporta código aritmético binario adaptativo (CABAC) y codificación de video entrelazado usando y codificación cuadro/campo adaptiva por imagen (PAFF) o codificación cuadro/campo adaptiva por macrobloque (MBAFF), y utilizado en aplicaciones de radiodifusión de televisión digital. Este perfil no permite orden arbitrario de slices.

**Perfil High:** Este perfil debe contener slices del tipo I, P o B solamente, además soporta herramientas del perfil Main, formato 4:2:0 con 8 bits por muestra, utiliza transformadas de 8 x 8 o 4 x 4, matrices de escalamiento para cuantización, presenta variaciones: High 10, High 4.2:2, High 4:4:4. El perfil High tampoco permite orden arbitrario de slices. Este perfil es utilizado para aplicaciones de radiodifusión de televisión digital.

**Extended:** Soporta herramientas del perfil Baseline, imágenes tipo B, SI y SP, codificación de video entrelazado. Tiene su aplicación para streaming de video.



#### 2.3.1.3.4 Niveles

El término nivel establece para cada perfil los valores permitidos de algunos parámetros de codificación, como el tamaño de la imagen, las tasas de bits y la resolución. Los niveles especifican el límite de desempeño requerido para realizar la decodificación, de esta manera garantizan, aún más que los perfiles, la interoperabilidad entre los diferentes decodificadores. El codificador debe ser configurado para operar en nivel especificado para cada perfil. Mediante la conjugación de los perfiles y los niveles se puede utilizar en diferentes aplicaciones y así se garantiza la conformidad e interoperabilidad adecuada.

#### 2.3.1.3.5 Configuración de la señal de video

A la salida de un codificador obtenemos una secuencia de video MPEG que contiene la información estrictamente necesaria para poder restablecer la imagen original. Estas señales de video son procesadas en varias capas que iremos describiendo a continuación:

**Secuencia de Video:** Una secuencia de video está compuesta por una o más unidades de acceso (AU). Posee un código inicial que corresponde a una AU del tipo IDR (Instantaneous Decoding Refresh) y se extiende hasta una AU que termine con una NAL del tipo end\_of\_seq. Cuando se tiene un conjunto de secuencias de video terminadas por un end\_of\_stream, se obtiene un flujo de video. Una secuencia de video debe tener al menos un cuadro del tipo I para disminuir la posibilidad de que existan errores en la transmisión y se refresque la transmisión cada cierto tiempo.

Una secuencia de video puede contener datos visuales pero también de contenidos auxiliares a los que se los conoce también como secuencia de soporte donde se especifica la configuración de parámetros como el tamaño horizontal y vertical de la imagen, la tasa de imágenes, dirección de barrido, tipo de barrido, velocidad de transferencia de bits, el perfil, el nivel y cuales matrices de cuantificación se utilizan para codificar imágenes espaciales y temporales. Esta secuencia de soporte es muy importante porque permite al decodificador entender el flujo de bits y decodificar la información e manera adecuada.

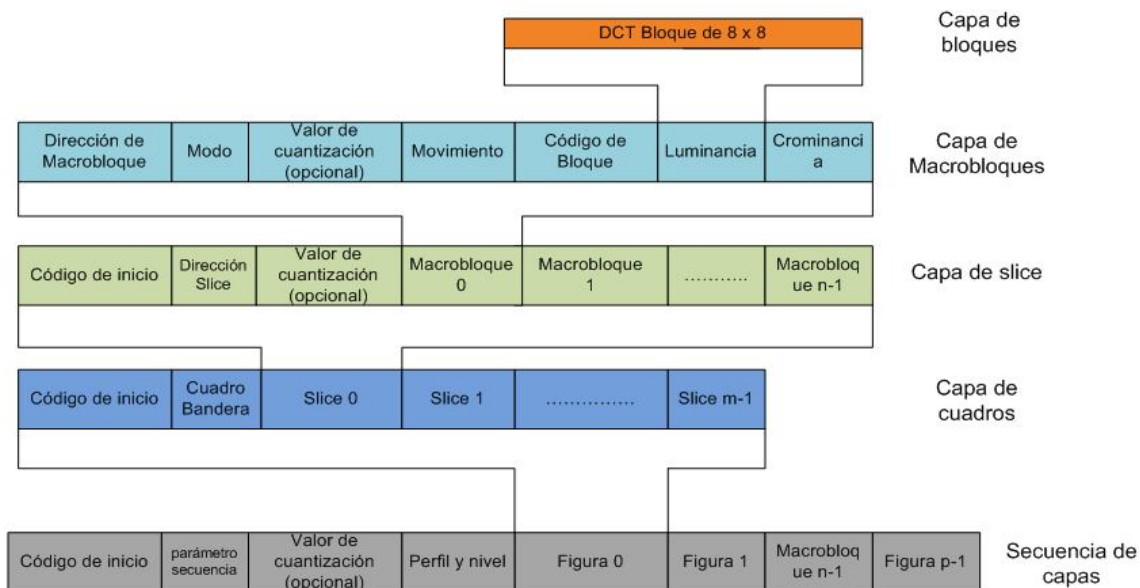
**Gop:** El acrónimo GOP significa grupo de imágenes (en inglés) y es la unidad fundamental de codificación temporal. La combinación de imágenes da como resultado un GOP. Su uso es opcional aunque en la práctica es realmente necesario, define las imágenes I, B, P definidas en el punto 2.3.2.1.

**Slice:** Es un conjunto de macrobloques que siempre deben representar una fila horizontal la cual se ordena de izquierda a derecha. Las Franjas son la unidad fundamental de sincronización para la codificación de la longitud variable y diferencial, los vectores iniciales en una Franja son enviados completamente, mientras que los demás vectores son transmitidos diferencialmente.

**Macrobloque:** Es la unidad fundamental de la imagen que además está compensada en movimiento. Cada macrobloque es un vector de desplazamiento en dos dimensiones situado en la parte superior de la secuencia. En el caso de una imagen B, el vector puede ser hacia adelante o hacia atrás. Por medio de estos vectores, el decodificador obtiene información acerca de las imágenes.

Un macrobloque consiste en 4 bloques de luminancia con un tamaño de 16 x 16 pixels y dos señales de crominancia cuya dimensión va a variar dependiendo del formato de codificador utilizado. Un macrobloque se puede partir en bloques de tamaño de 16 x 16, 16 x 8, 8 x 16 y 8 x 8 muestras. En el caso de que la partición sea de 8 x 8, cada bloque a su vez se puede partir en 8 x 8, 8 x 4, 4 x 8 y 4 x 4.

**Bloque:** Es la unidad fundamental de la información de la imagen, se representa por una matriz de coeficientes DCT de 8x8 pixeles, los mismos que representan datos Y (luminancia), Cr (crominancia) o Cb (saturación). En la recomendación H.264:2005 de la ITU-T define que los bloques luego deben dividirse en partes de 8 x 8 o 4 x 4 para la aplicación de la transformada DCT [10].



**Figura 2. 11 Estructura de la trama de video**

### 2.3.2 Estructura del Transport Stream para el contenido de audio

En esta sección describiremos la configuración y codificación que debe tener la estructura del contenido de audio para que pueda ser procesado dentro la multiplexación de las señales.

Estas configuraciones incluyen los parámetros para las señales en la entrada del codificador y las restricciones al proceso de codificación aplicables al sistema brasileño de televisión digital terrestre (SBTVD).

#### 2.3.2.1 Características de la señal de audio

Las condiciones generales del formato de entrada de audio deben ser obligatoriamente:

- Frecuencia de muestreo de la señal de audio: 32kHz, 44.1kHz ó 48kHz.
- Configuración de señales estereofónicas o multicanal.
- La tasa de muestreo para todas las señales debe ser obligatoriamente la misma
- La cuantización de las señales de entrada debe utilizar 16 o 20 bits.
- Un programa de audio debe tener al menos un canal de audio. El número máximo de canales en el programa estará limitado por el número máximo permitido en la ISO/IEC 14496-3. Además es recomendable que los

programas multicanales se preparen de conformidad con la ITU. Recomendación BS.775-1.

### **Parámetros de la señal de audio**

El formato de la señal de audio debe obligatoriamente admitir flujos de bits o archivos conteniendo audio digital no comprimido en formato PCM como WAVE o AIFF, estéreo y multicanal.

Entre las interfaces de entrada y salida digitales permitidas deben estar la AES3, SD, HD-SDI y HDMI.

El nivel de referencia para la intensidad o presión sonora debe ser obligatoriamente igual a 0 dB. La banda dinámica admisible de excursión debe estar limitada obligatoriamente a + 20 dB (headroom) y - 70 dB con respecto a la referencia, correspondiendo a una banda dinámica típica de 90 dB. Es conveniente que los niveles de audio medio estén a 0 dB, para permitir homogeneidad en el volumen entre canales distintos. La señal debe acomodar picos de por lo menos 4 veces su potencia media RMS

#### **2.3.2.1.1 Metadatos**

Existen datos auxiliares que contienen descripciones del contenido de los programas de audio, parámetros de configuración de los servicios de audio y parámetros de las señales de audio transmitidas en el flujo de bits.

Este tipo de datos auxiliares pueden ser la descripción del contenido de los programas de audio, es decir, la descripción de los objetos de audio mezclados en el contenido. Otro tipo de dato auxiliar es el modo multicanal y finalmente el volumen de referencia para operaciones de ecualización en la reproducción en el terminal de acceso.

Además los datos auxiliares y la descripción de contenidos de programas de audio pueden ser clasificadas en dos niveles. El primer nivel es obligatoriamente normativo, con datos esenciales para la decodificación y reproducción correcta del servicio de audio en el receptor. Estos datos pueden ser información de cantidad, modo de los canales, perfiles, niveles, etc.

El segundo nivel en cambio es informativo netamente y no debe afectar a la decodificación pero en cambio trae consigo información sobre los programas de audio asociados a cada PID.

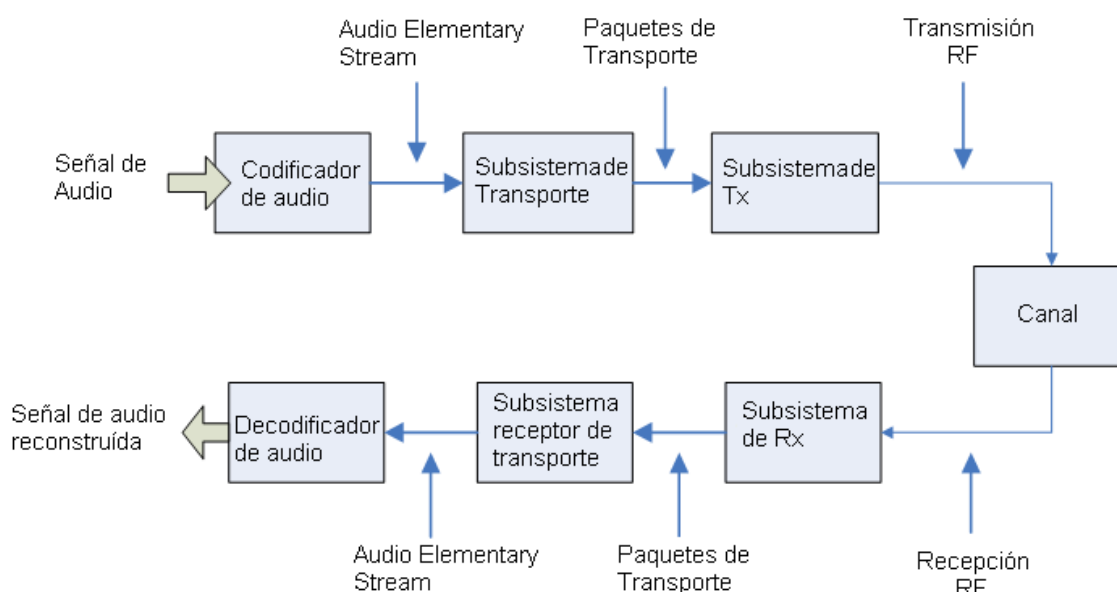
### 2.3.2.1.2 Servicios de audio y canales auxiliares

La transmisión de canales adicionales al programa principal se los considera como servicios de audio que pueden ser opcionales u obligatorios. Estos servicios se transmiten por canales auxiliares adicionales con un PID distinto o en un mismo flujo de bits (PID), dependiendo del tipo de servicio. Ejemplos de estos servicios son:

- Servicio de descripción de audio (DA), es un canal monoaural de voz y suministra una descripción de la escena.
- Servicio de programa de audio secundario (SAP)
- Transmisión de programas adicionales al principal y audio secundario proveniente de otras formas de sonido como pueden ser efectos.

### 2.3.2.2 Compresión y codificación de la señal de audio

La señal de audio en un sistema de televisión digital es sometida a un subsistema de audio donde se la comprime, se la codifica, se la transforma en paquetes de transporte para poder empezar la transmisión y la decodificación respectiva con el mismo proceso inverso.



**Figura 2. 12 Proceso de Compresión y codificación del audio**

Al igual que el video, el audio también es sometido a un proceso de compresión que está compuesto por un codificador y decodificador. Existe un bloque de subsistema de audio que precede al bloque de transporte. El codificador recibe las señales de audio en el dominio del tiempo y la transforma en el dominio de la frecuencia por medio de la DCT para convertirlas en una cadena de bits, reduciendo así la cantidad de información.

Existe otra herramienta de compresión que se emplea de manera adicional, como lo son las estimativas psicoacústicas de la señal donde se elimina información en bandas de frecuencia que no son perceptibles para el oído humano.

La compresión y todos los procedimientos deben ser obligatoriamente compatibles con la ISO/IEC 14496-3. El proceso para la generación del bitstream de la señal de audio empieza por el procesamiento psicoacústico que calculan la cantidad de enmascaramiento que permite el discernimiento de una señal de audio de otras señales. Luego estas señales entran a los bancos de filtros. Posteriormente las señales son cuantizadas en base al factor de enmascaramiento utilizado.

Finalmente se genera el flujo de bits que va a ser transmitido a la etapa de transporte.

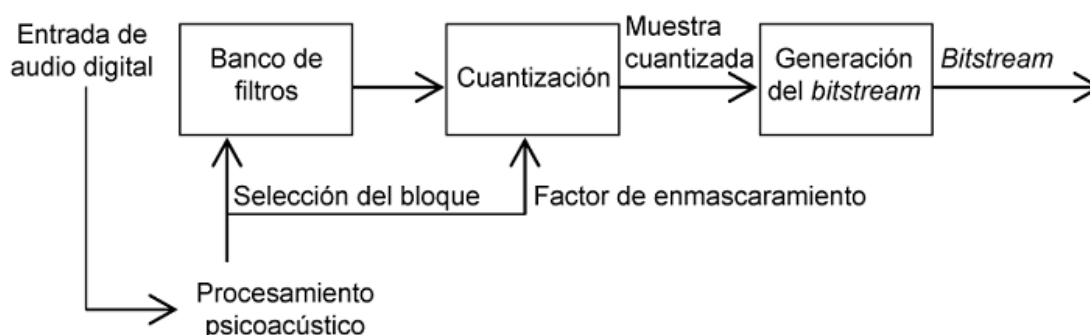


Figura 2. 13 Procedimiento de codificación de audio

### 2.3.2.2.1 Perfiles y Niveles

Dentro de la codificación del audio existen los siguientes perfiles con sus respectivos niveles para el estándar MPEG4-AAC que se deben permitir de manera obligatoria:

- LC –AAC (Baja complejidad), perfil básico del estándar AAC para niveles L2 y L4.
- HE –AAC (Alta eficiencia), perfil avanzado de alta eficiencia, combinando el perfil LC con el uso de la herramienta SBR (spectral band replication), niveles L2 y L4.
- HE –AAC, combinado con la herramienta PS (parametric stereo), nivel L2.

### 2.3.2.2.2 Restricciones en la codificación

#### 2.3.2.2.2.1 Modos de codificación

El modo de codificación determina el número de canales disponible en el servicio de audio. En la tabla 2.4 se describen los modos permitidos para el servicio full-seg como para one-seg que deben tener los codificadores y que deben ser capaces de procesar los decodificadores

**Tabla 2. 4 Modos de Audio permitidos para el estándar ISDB-Tb**

<b>Parámetro</b>	<b>Restricción Full-seg</b>	<b>Restricción One-seg</b>
<i>Modos de audio permitidos</i>	Monoaural (1/0), estéreo (2/0 y 2/0 + LFE)*, estéreo multicanal (3/0, 2/1, 3/1, 2/2, 3/2, 3/2 + LFE)*, dos señales de audio independientes (monoaural dual), multi-audio (tres o más señales de audio) y	Monoaural (1/0), estéreo (2/0)

	combinaciones de éstos	
<i>Modos de audio recomendados</i>	Estéreo (2/0), multicanal (3/2+LFE)	Estéreo (2/0)
<i>Downmix</i>	Para la configuración 5.0 y 5.1 se debe utilizar el parámetro matrix_downmix_idx.  En otras configuraciones multicanal no hay restricciones con tal que se mantenga la integridad del audio y nivel de inteligibilidad.	N/A
* Número de canales para altavoces delanteros/traseros.		

#### 2.3.2.2.2 Principales parámetros

Los principales parámetros del sistema de codificación de audio son:

- Mecanismos de transporte: Son LATM/LOAS (Low overhead audio multiplex and transport mechanism) definidos en la ISO/IEC 14496-3.
- Perfiles y niveles: Se debe cumplir con los perfiles y niveles descritos en la sección 2.3.2.2.3.
- Número de canales recomendados: Se debe cumplir con los descritos en la norma ABNT NBR 15602-2
- Tasa máxima de bits: Conforme ISO/IEC 14496-3

#### 2.3.2.3 Transporte y multiplexación de la señal de audio

La codificación y el empaquetamiento intermedio del audio deben ser compatibles con LATM/LOAS. El elementary stream debe ser encapsulado en el formato de transporte LATM y debe utilizar obligatoriamente el elemento de



multiplexación AudioMuxElement(). La capa de sincronización del transporte de audio (LOAS) debe utilizar el formato de transmisión AudioSyncStream()

Posteriormente la señal de audio que es transportada en el TS que utiliza LA sintaxis de transporte LATM/LOAS debe ser identificado obligatoriamente por `stream_type 0x11` de acuerdo con el `stream_type assignments` en la ISO/IEC 13818-1:2007.

En cuanto a la decodificación de la señal de audio, el receptor debe identificar el tipo, el perfil y el nivel transmitidos y ser capaz de extraer los payloads de los objetos de audio.

Como se transmite audio MPEG-4 sobre un flujo de transporte (TS) MPEG-2 se debe utilizar señalización explícita SBR (técnica de replicación de banda espectral, que sirve para mejorar los códecs de audio a bajos bit rate) sin alineamiento de PES.

En la tabla 2.5 se describe los campos de la sintaxis LATM/LOAS de transporte en StreamMuxCodig que deben estar obligatoriamente formateados para la identificación y recuperación de los payloads de audio [11].

**Tabla 2. 5 Principales parámetros LATM a ser configurados.**

<b>Parámetro LATM</b>	<b>Descripción</b>
<i>AudioMuxVersion</i>	Debe asumir obligatoriamente el valor "0"
<i>allStreamsSameTimeFraming</i>	Debe asumir obligatoriamente el valor "1"
<i>NumSubFrames</i>	Debe asumir obligatoriamente el valor "0" indicando un único PayloadMux() (Access unit) presente en el AudioMuxElement()
<i>NumProgram</i>	Debe asumir obligatoriamente el valor "0" indicando un programa por multiplex LATM
<i>NumLayer</i>	Debe asumir obligatoriamente el valor "0" indicando una sola capa.

<i>FrameLengthType</i>	Debe asumir obligatoriamente el valor "0" indicando que el tamaño de cuadro del payload puede tener extensión variable. La extensión del payload en bytes se especifica directamente en PayloadLengthInfo() con palabras de 8 bits
------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.3.3 Estructura del Transport Stream para el contenido de datos

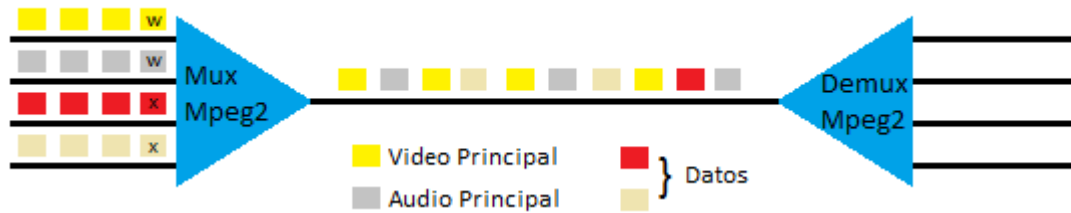
En esta sección se va a describir el modelo de referencia que permite la difusión de datos para televisión digital y la codificación que deben tener las señales de audio según su clasificación. Para el transporte de las señales por medio del Transport Stream, se deben multiplexar las señales de video, audio y datos en un único flujo de transporte. El sistema MPEG-2 agrega a los paquetes de audio y video, información necesaria para que el contenido pueda ser presentado de manera sincronizada. Esta sincronización se la realiza en base a un parámetro conocido como línea de tiempo (timeline) que aplica marcas de tiempo a conjuntos de muestras codificadas de audio y video que comparten una señal de reloj.

#### 2.3.3.1 Multiplexación de datos

Existen tres tipos de multiplexaciones que pueden transportar a los streams de datos que son a través de servicios sincrónicos (levemente acoplados), sincronizados (fuertemente acoplados) o asincrónicos [12].

##### 2.3.3.1.1 Servicio de transporte sincrónico

Este servicio asume que los flujos de datos son sincronizados entre sí en base al parámetro del timeline pero estos flujos de datos no están relacionados con la temporización de los flujos de audio y video, es decir, las marcas de tiempo asociadas a los flujos de datos son distintas de las asociadas a los flujos de audio y video principales.



**Figura 2. 14 Transporte de datos sincrónicos**

Estos casos se dan para el servicio de presentación de comerciales, sistemas de ayuda, estadísticas de deporte, etc.



**Figura 2. 15 Ejemplo de servicio de transporte sincrónico**

### 2.3.3.1.2 Servicio de transporte sincronizados

En este caso los flujos de datos están sincronizados en base a la línea de tiempo y contienen las mismas marcas de tiempo que los flujos de video y audio principales. En la figura se ve claramente como las marcas de tiempo de los flujos de video, audio y datos son las mismas. Un ejemplo de este servicio puede ser la transmisión de "closed caption".

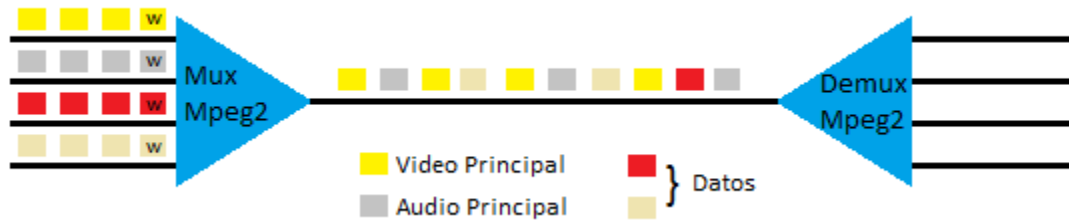


Figura 2. 16 Sistema de transporte sincronizado



Figura 2. 17 Ejemplo de servicio de transporte sincronizado

### 2.3.3.1.3 Servicio de transporte asincrónicos

En los casos anteriores, los datos van a ser transmitidos en base a una línea de tiempo determinística, entonces esto permite que los datos puedan ser transmitidos con sincronismo. Pero existe la posibilidad de transmitir datos donde la sincronización está dada en un instante de tiempo aleatorio que depende del usuario. Un claro ejemplo de este servicio son las aplicaciones interactivas, donde los datos son generados en tiempo real y la sucesión de evento depende totalmente del usuario. En estos casos los datos obligatoriamente deben transmitirse de manera asincrónica,

En esta clase de servicio los datos no tienen asociada ninguna marca de tiempo ya que no se utiliza el parámetro de la línea de tiempo sino el de causalidad/restricción. Sin embargo puede existir sincronización entre los

objetos transportados y entre esos objetos y los flujos de audio y video principales.

Además se envía una aplicación la cual, mediante una programación previa, especifica el comportamiento relativo de los datos, audio y video principales, en el tiempo y espacio. Este servicio es la única forma de sincronización de objetos con tiempo de sincronización indeterminado. Pero para esto se necesita un lenguaje que especifique el sincronismo y que el receptor sea capaz de interpretar. MPEG-2 no especifica el middleware (en este caso GINGA) empleado para la sincronización del servicio asincrónico.

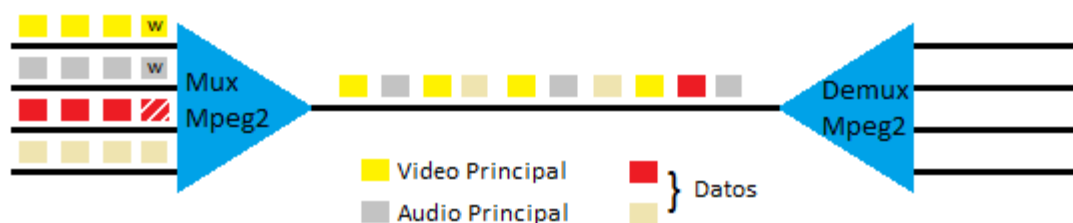


Figura 2. 18 Sistema de transporte asíncrono

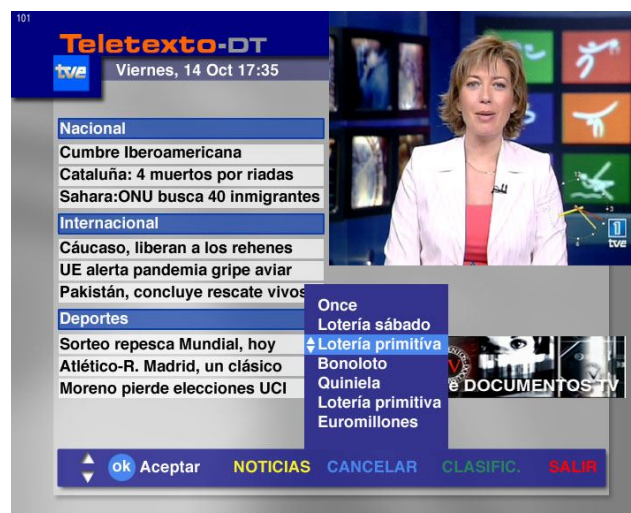


Figura 2. 19 Ejemplo servicio de transporte asíncrono

### 2.3.4 Tipos de especificación de transmisión de datos

En la sección anterior se detalló las clases de servicio que existen según el sincronismo que deben tener los datos que se transmite. Además se tiene otra clasificación que depende de la función o clase de stream que se transmite, es decir, existen varios tipos de especificaciones para la transmisión

de datos que dependen del tipo de stream. Esta identificación se la realiza en un PMT (Program map table) del que hablaremos más adelante. En esta clasificación veremos que existen especificaciones que pueden pertenecer a transmisiones asincrónicas o sincrónicas. A continuación se explica cada una de ellas:

#### **2.3.4.1 PES independiente**

Esta especificación es utilizada para streams de datos sincronizados y asíncronos para servicios de radiodifusión.

**Servicio de transmisión PES sincronizada:** Se utiliza cuando es necesario sincronizar datos en un stream con otros streams que pueden ser también de audio o video. En este tipo de servicio los datos se transmiten utilizando un paquete PES según se describe en la sección 2.2.1 según la ISO/IEC 13818-1 [13].

Además de la sintaxis y semántica especificada en la ISO/IEC 13818-1, la cabecera del paquete PES debe tener otras restricciones adicionales que son:

- Stream\_id: Debe estar configurado en "0xBD".
- Packet stream length: Campo de 16 bits que debe tener un valor diferente de cero.

En el Anexo A1 se puede encontrar la estructura de los datos PES sincronizados.

**Servicio de transmisión PES asíncrona:** Se utiliza cuando la sincronización no es necesaria, por ejemplo, para la transmisión de caracteres superpuestos.

En este tipo de servicio los datos se transmiten utilizando un paquete PES según se describe en la sección 2.2.1 según la ISO/IEC 13818-1.

Además de la sintaxis y semántica especificada en la ISO/IEC 13818-1, la cabecera del paquete PES debe tener otras restricciones adicionales que son:

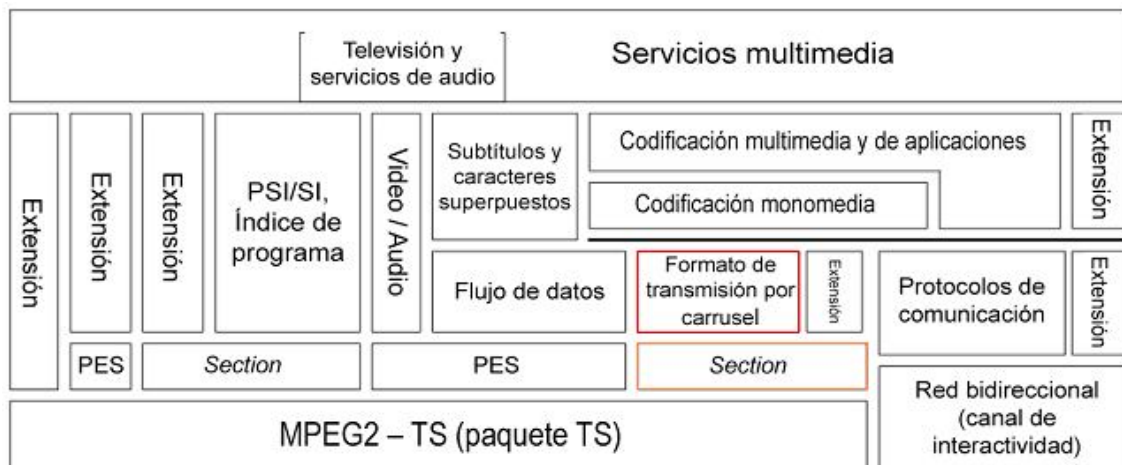
- Stream\_id: Debe estar configurado en “OXBf”.
- Packet stream length: Campo de 16 bits que debe tener un valor diferente de cero.

En el Anexo A2 se puede encontrar la estructura de los datos PES asíncronos.

### 2.3.4.2 Carrusel de datos /objetos

Cuando se tienen datos que se pueden transmitir en cualquier instante de tiempo, es decir que no tienen marcas de tiempo asociadas, pudiendo ser una transmisión sincronizada o asíncrona, entonces se debe realizar los envíos de dichos datos de manera cíclica, donde el tiempo de sintonización no tiene relación alguna con el tiempo en el que se reciben los datos.

Para poder realizar este envío cíclico se emplean los carruseles de datos u objetos, especificados en el estándar DSM-CC (Digital Storage Media – Command Control) y que son transportados en secciones privadas específicas MPEG-2 que pueden contener hasta 4096 bytes de datos y un encabezado que le indica al receptor cuantas secciones están siendo utilizadas para transportar un flujo de datos específicos y la manera como debe decodificarlas.



**Figura 2. 20 Formación del Carrusel de datos dentro del Transport Stream**

#### 2.3.4.2.1 Carrusel de datos

Esta especificación se destina para la transmisión de contenido como download de datos para una unidad receptora o transmisión de contenidos para

servicios de multimedia. Un carrusel de datos permite el envío de datos no estructurados ya que la estructuración queda a cargo del sistema de TV digital que se está utilizando.

La transmisión repetida de datos, como es definida en la especificación DSM-CC, permite a la unidad receptora obtener datos por demanda en cualquier momento durante un período de transmisión. Los datos se transmiten en una unidad modular formada por bloques donde todos los bloques, excepto los que están al final del módulo, tienen el mismo tamaño y cada bloque se transmite en secciones.

Como se mencionó anteriormente, en la transmisión del carrusel de datos se utiliza la sección DSM-CC, que puede transmitir dos tipos de mensajes de acuerdo al valor que tenga el parámetro `table_id`.

Estos tipos de mensajes son un mensaje download de bloque de datos (referida como mensaje DDB), donde se transmite el cuerpo de datos con cada módulo dividido entre los bloques y un mensaje de indicación de información download (referida como mensaje DII), que contiene información relacionada con el identificador del download e información acerca de los módulos.

El mensaje DDB se transmite por almacenamiento en el campo `downloadDataMessage ()` la estructura de datos por medio de bloques de datos. Cada bloque es representado con un número de bloque en el mensaje DDB para permitir que el receptor pueda reorganizar los bloques.

La estructura de datos de la sección DSM-CC, los mensajes DII y DDB se la puede encontrar en el Anexo A3, A4 y A5 [13].

#### **2.3.4.2.2 Carrusel de objetos**

El carrusel de objetos es un protocolo de transmisión cíclica de datos. El resultado es un stream elemental de datos que contiene el sistema de archivos transmitido de forma cíclica. Por lo tanto, si un receptor específico no recibió un bloque de datos en particular (debido a una falla en la transmisión o por haber sintonizado el canal después de la transmisión de ese bloque), basta con esperar la retransmisión correcta



Un carrusel de objetos permite el envío cíclico de un sistema de archivos, así al sintonizar un determinado canal, el receptor debe poseer la capacidad de decodificar los datos recibidos y colocarlos en un espacio de memoria para que puedan ser utilizados, manteniendo la estructura de archivos y directorios enviada.

Se puede realizar el envío de más de un carrusel de manera simultánea y un carrusel puede hacer uso de recursos (archivos, directorios y otros objetos) que están siendo transferidos en otros carruseles. Además las aplicaciones que se transfieren en archivos de carrusel pueden hacer referencia a recursos de otros carruseles, por ejemplo, una página HTML puede utilizar una imagen que está siendo transmitida en otro carrusel de objetos.

La transmisión de datos para carrusel de objetos se transmite de acuerdo con la DSM-CC de carrusel de objetos y especificación de carrusel de datos DSM-CC. Esta especificación DSM-CC de carrusel de objetos representa un dominio de servicio particular que consiste en la colección de objetos DSM-CC U-U dentro de una red de SBTVD. Este dominio de servicio tiene un puerto de servicio que presenta un gráfico de servicios y nombres de objetos para los receptores. En la DSM-CC se define un parámetro Network Service Access Point (NSAP), cuya sintaxis se define en la Norma ABNT NBR 15606-3, que se utiliza para referirse al carrusel de objetos desde otro dominio de servicio.

#### **2.3.4.3 Mensajes de eventos**

La especificación de la transmisión del mensaje del evento provee un medio para enviar mensajes sincronizados o asíncronos de información inmediatamente o en horas específicas para una aplicación operada en una unidad de receptor de una estación de difusión. Su identificación de *tipo de stream* es *0x0C*, *0x0D* [13].

La especificación de la transmisión del mensaje del evento es extendida para negociar los distintos tiempos apuntando métodos por la aplicación con base en la especificación del descriptor de *stream* y su especificación de la transmisión de sección DSM-CC especificada en la ISO/IEC 13818-6.

Los descriptores de stream que se definen son el descriptor de stream DSM-CC, descriptor de referencia NPT, descriptor de modo Stream, descriptores de evento de stream, descriptor de evento general.

El mensaje de eventos puede ser utilizado para transmisión de mensajes de emergencia a una población.

#### **2.3.4.4 Data Piping**

Protocolo que permite insertar datos de una red de radiodifusión directamente en el payload del paquete MPEG-2 TS por medio del servicio de transmisión de datos. Este servicio de transmisión de datos puede usar el campo `payload_unit_start_indicator` y el campo `transport_priority` de los paquetes de la stream de Transporte MPEG-2 en forma de servicio privado. Hay que considerar que el uso del campo *adaptation\_field* debe ser compatible con MPEG-2. Su identificación de *tipo de stream* es *0x7E*.

El servicio de transmisión de datos debe indicar el uso de un data pipe (canal de datos), para lo cual incluye uno o más descriptores de transmisión de datos (SI / PSI), que contienen información de servicio e información específica del programa para que el receptor pueda demultiplexar y decodificar los streams. Este servicio se ofrece en el estándar DBV y es por eso que la entrega de los bits en tiempo a través de un data pipe es un servicio privado y no se especifica en el estándar ISDB-Tb.

#### **2.3.4.5 Encapsulado Multiprotocolo**

Son datagramas que se encapsulan en *datagram\_sections* que son compatibles con el formato *DCMCC\_section* para datos privados. El mapeo de la sección dentro de los paquetes MPEG-2 de stream de transporte se define en sistemas MPEG-2. Su identificación de *tipo de stream* es *0x0A*. La sintaxis del *datagram\_section* se la puede encontrar en el Anexo A6.

El servicio de transmisión de datos debe indicar la transmisión de datagramas por la inclusión de uno o más descriptores de transmisión de datos en SI, donde cada descriptor debe ser asociado a un stream por medio de un

identificador *component\_tag* que debe ser idéntico al valor del campo *component\_tag* de un *stream\_identifier\_descriptor*.

Además existen otros descriptores que se deben definir como el descriptor de protocolo de transporte. Una aplicación de este tipo de datos es el envío de datagramas que contengan protocolos IP.

## 2.4 ESTRUCTURA GENERAL DEL TRANSPORT STREAM

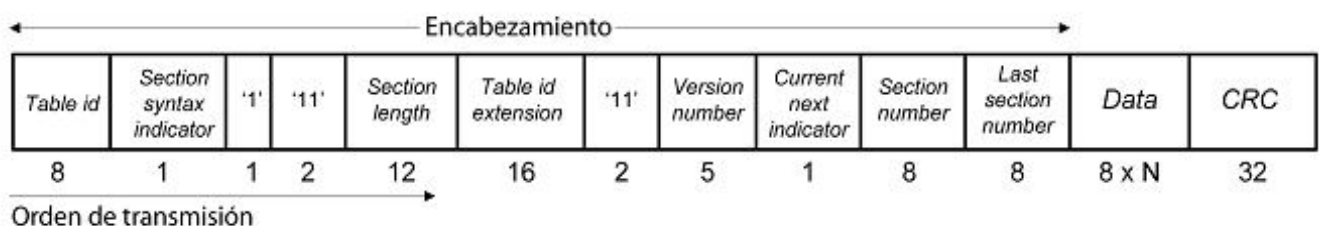
### 2.4.1 Estructura de las Señales Codificadas

Para explicar la estructura del transport stream adecuadamente vamos a partir nuevamente del concepto de que todas las señales deben estar codificadas y se deben multiplexar en paquetes.

Existen contenidos que no son del tipo de flujos de datos los cuales son convertidos al formato de Sección, el resto forman los antes mencionados PES (packetized elementary stream).

#### 2.4.1.1 Sección

Las secciones están compuestas por una cabecera, una carga útil y un código de detección de errores. En la siguiente figura se muestra la estructura de las secciones [13].



**Figura 2. 21 Formato extendido de las secciones**

La descripción de los ítems de una sección debe ser obligatoriamente como sigue:

- *Table\_id*: Debe indicar obligatoriamente a qué tabla pertenece la sección de acuerdo a la tabla 2.5;

- *Section\_syntax\_indicator*: Debe ser obligatoriamente un campo de 1 sólo bit que determina si se debe utilizar obligatoriamente el formato normal o extendido, respectivamente cuando sea '0' ó '1';
- *Section\_length*: Debe ser obligatoriamente un campo de 12 bits, que describe el número de bytes de datos que suceden a este campo. El valor de este campo puede variar de acuerdo con la tabla PSI o SI;
- *Table\_id\_extension*: Es el campo que debe servir obligatoriamente de extensión para el Table\_id;
- *Version number*: Debe describir obligatoriamente el número de la versión;
- *Current\_next\_indicator*: Debe ser obligatoriamente un campo de sólo 1 bit. Cuando definido con el valor '1' debe indicar obligatoriamente que la tabla de asociación de programas enviada es válida y aplicable en ese momento. Cuando este bit es definido como '0', debe indicar obligatoriamente que la tabla enviada no es aplicable y que el sistema debe obligatoriamente aguardar la próxima tabla válida;
- *Section\_number*: Debe ser obligatoriamente un campo de 8 bits e informar el número de la sección contenida en la tabla;
- *Last\_section\_number*: Debe ser obligatoriamente un campo de 8 bits y especificar el número de la última sección contenida en la tabla;
- *CRC\_32*: Debe ser obligatoriamente un campo de 32 bits y estar de acuerdo con la ISO/IEC 13818-1.

Tabla 2. 6 Table id

Valor	Descripción
0x00	program_association_section (Tabla PAT)
0x01	conditional_access_section ((Tabla CAT))
0x02	TS_program_map_section (Tabla PMT)
0x03	TS_description_section

0x04	ISO_IEC_14496_scene_description_section
0x05	ISO_IEC_14496_object_descriptor_section
0x06	Metadata_section
0x07	IPMP_Control_Information_section (definida en ISO/IEC 13818-11)
0x08-0x3F	ITU-T Rec. H.222.0   ISO/IEC 13818-1 reservada
0x04-0xFE	User private
0xFF	Prohibido

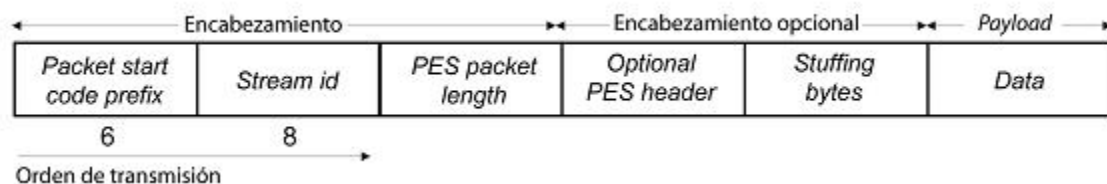
Por ejemplo en el caso de que la sección contiene una tabla PAT (Program Association Table), tendríamos que la cabecera de la sección estaría conformada de la siguiente manera:

Sección de la Tabla PAT										
0x00	1	-	Section length	0x073b	11	0	1	0	0	Payload

En cuanto a la carga útil de cada sección, esra variará dependiendo del tipo de información qu se transmita, es decir, si se trata de una tabla PAT, PMT, NIT, etc. Su especificación se explica en las secciones posteriores.

#### 2.4.1.2 PES

Los paquetes PES deben tener la estructura que se muestra en la figura 2.18 y la tabla 2.6 [14].



**Figura 2. 22 Estructura del paquete PES**

La descripción de los componentes del paquete PES debe ser obligatoriamente la siguiente:

- *Packet start code prefix*: Debe ser obligatoriamente un código que representa el comienzo del paquete de PES y se debe fijar obligatoriamente en 0x000001;
- *Stream id*: Se debe usar obligatoriamente para identificar el tipo y el número del elementary stream (señales codificadas; debe ser obligatoriamente válido para otras señales). El tipo y el número del elementary stream deben estar de acuerdo con la Tabla 2;
- *PES packet length*: Debe indicar obligatoriamente el número de bytes en el paquete PES después de este campo. El valor '0' indica que el tamaño del paquete PES no se debe especificar obligatoriamente y no debe tener límites obligatoriamente. El valor '0' solo se permite para paquetes PES cuando el payload esté compuesto por elementary streams de video;
- *Opcional PES header*: Debe estar de acuerdo obligatoriamente con la ISO/IEC 13818-1;
- *Stuffing bytes*: También conocido como bytes de relleno deben tener valor fijo en 0xFF y no deben exceder 32 bytes de longitud.

Tabla 2. 7 Stream ID

STREAM ID	
VALOR	FUNCIÓN
0xBC	Mapeador del stream de programa (program stream map)
0xBD	Stream privado 1 (private stream 1)
0xBE	Stream de rellenado (padding stream)
0xBF	Stream privado 2 (private stream 2)
'11 0x xxxx'	Número del stream de audio 'x xxxx' de acuerdo con la ISO/IEC 13818-3.
'1110 xxxx'	Número del stream de video 'xxxx' de acuerdo con la ISO/IEC 13818-3

0xF0	Stream ECM
0xF1	Stream EMM
0xF2	Stream DSM-CC de acuerdo con la Rec. H.222.0:2006,
0xF3	Stream de acuerdo con la ISO/IEC 13522-1
0xF4	De acuerdo con el tipo A descrito en la Rec. H.222.1 :1996
0xF5	De acuerdo con el tipo B descrito en la Rec. H.222.1 :1996
0xF6	De acuerdo con el tipo C descrito en la Rec. H.222.1 :1996
0xF7	De acuerdo con el tipo D descrito en la Rec. H.222.1 :1996
0xF8	De acuerdo con el tipo E descrito en la Rec. H.222.1 :1996
0xF9	Stream auxiliar (anciliary stream)
0xFA	Streams SL empaquetados (packetized stream) de acuerdo con la ISO/IEC 14496-1
0xFB	FlexMux streams de acuerdo con la ISO/IEC 14496-1
0xFC – 0xFE	Reservado para streams de datos
0xff	Directorio de stream de programa (program stream directory)
NOTA: Números entre ‘ ‘ representan números binarios. Los demás números son hexadecimales.	

#### 2.4.1.3 Señales de control de transmisión

Una señal de transport stream puede contener varios programas que contengan información de audio, datos o video de manera indistinta y combinada. Por esta razón es necesario incluir ciertas listas que describe la estructura que tiene el TS en ese instante. A estas listas se las denomina “información específica de programa” o PSI, las mismas que son tablas, ocasionalmente transmitidas en la parte de payload.

Las tablas PSI actúan como tablas de contenidos, proporcionando los datos necesarios para encontrar cada programa y presentarlo al espectador. Estas tablas le ayudan al decodificador a localizar el audio y video de cada programa, así como también en la verificación de los derechos de acceso condicional (CA).

Las tablas PSI se repiten frecuentemente para soportar el acceso aleatorio requerido por el decodificador cuando se sintoniza o se cambia de canal. Las tablas utilizadas son las siguientes:

#### 2.4.1.3.1 Tabla de asociación de Programa (PAT)

Es un directorio raíz para el transport stream, la tabla proporciona el valor PID para los paquetes que contengan el PMT asociado con cada programa, es decir, define la correspondencia entre el program number con el PID del paquete del TS que contiene las definiciones del programa.

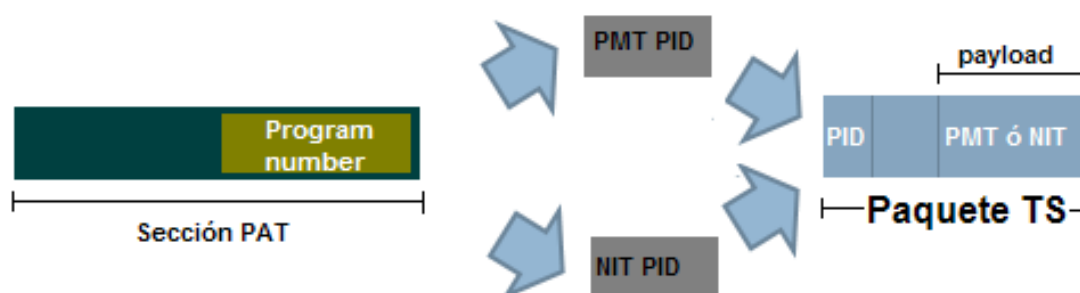


Figura 2. 23 Funcionamiento de la PAT

Dependiendo del valor del program number la sección contendrá el PID de la tabla PMT o NIT. En las figuras 2.24 y 2.25 se muestran cómo está compuesto el control de señales de la transmisión de la PAT [15].

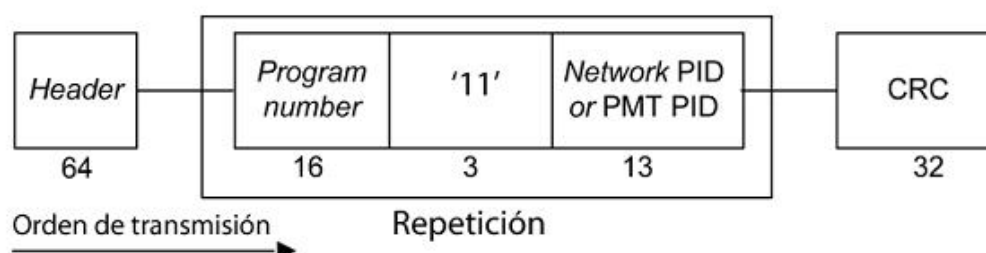
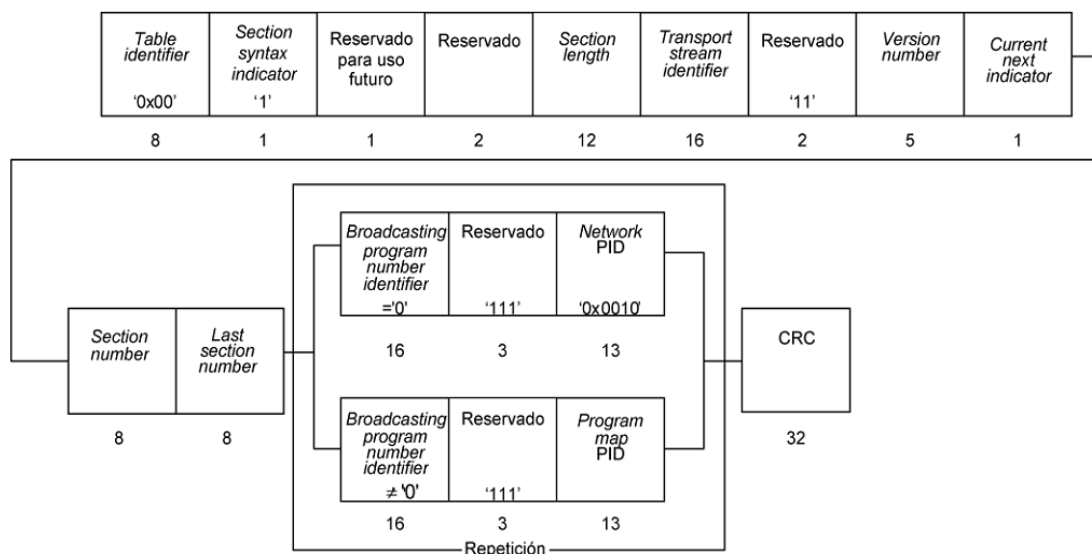


Figura 2. 24 Tabla de asociación de programa (PAT)



En cuanto a los ítems que se muestran en el diagrama de bloques, deben tener las siguientes descripciones:

- En cuanto al Header y el CRC, deben tener el mismo formato que la sección extendida. El contenido de bit que sigue al *Section syntax indicator* debe ser obligatoriamente 0. El valor del *Table id* obligatoriamente debe ser 0x00 para que represente a una tabla PAT. El *Table id extension* se debe usar obligatoriamente para transmitir el *Transport stream id*;
- El campo *Program number* se debe utilizar obligatoriamente para identificar el número del programa de radiodifusión. El valor '0' se debe utilizar obligatoriamente para la NIT.
- El campo *Network PID* o *PMT PID*, en el caso de que el *program number* sea '0', representará el PID de la NIT y debe tener un valor obligatorio de 0x0010. Si el *program number* es diferente de '0', entonces representará obligatoriamente al PID de la PMT.



**Figura 2. 25 Estructura de la sección de la PAT**

Por ejemplo en el caso que la sección transmita una tabla PAT que especifique dos servicios de TV Digital, es decir dos canales que se transmitan de manera simultánea, donde la primera tabla PMT tenga un PID igual a 1031 y la segunda 1032, debería contener 3 descriptores y su estructura debería contener los siguientes datos:

Tabla 2. 8 Ejemplo de una Tabla PAT

Parámetro	Valor	Observaciones
Header	El mismo formato que maneja una sección extendida	El table id debe ser 0x00
<b>Descriptor 1→NIT</b>		
Program number	0	Indica que se especificara el PID de una tabla NITT
NIT PID	16	PID del TS de la tabla NIT
<b>DESCRIPTOR 2→ PMT para el primer servicio</b>		
Program number	0x760	ID del servicio de TV dIGITAL
PMT PID	1031	PID del TS de la tabla PMT de este servicio
<b>DESCRIPTOR 2→ PMT para el primer servicio</b>		
Program number	0x761	ID del servicio de TV dIGITAL
PMT PID	1032	PID del TS de la tabla PMT de este servicio
CRC	El mismo formato que maneja una sección extendida	

#### 2.4.1.3.2 Tabla de mapeo de programa (PMT)

Se trata de una tabla que está localizada en la PAT y se encarga de listar los valores PID para los paquetes que contengan programación de video, audio, referencia de reloj, y componentes de datos, es decir, es la responsable de permitir el mapeo entre un número de programa y los elementos de programa que lo contienen. También realiza una lista de los valores PID para cada “Entitlement Control Message” (ECM). Los ECMs permiten decodificar el audio, video y datos para presentar un programa.

En la figura 2.26 se muestra el diagrama de bloques de esta tabla [14].

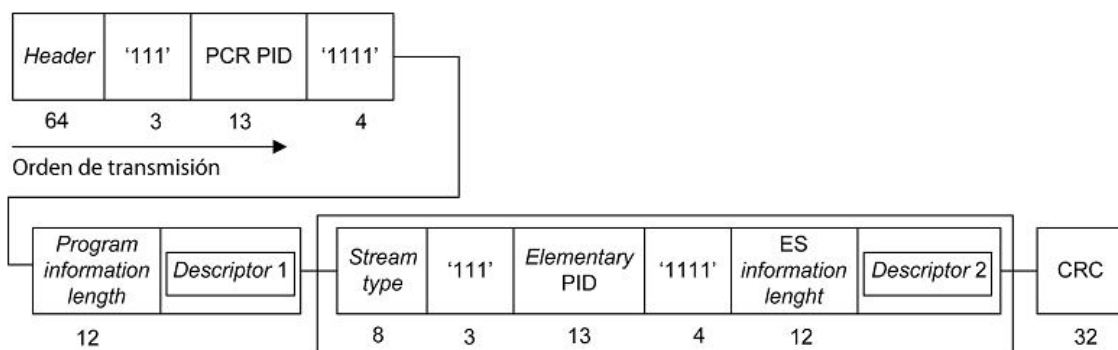


Figura 2. 26 Tabla de mapeo de programa

En cuanto a los ítems que se muestran en el diagrama de bloques, deben tener las siguientes descripciones:

- Los campos Header y CRC deben ser iguales a los descritos para la tabla PAT, con la diferencia que el *table id* debe ser 0x02 para representar la tabla PMT y el *Table id extensión* debe ser obligatoriamente usado para transmitir el número de programa.
- *PCR PID* debe representar obligatoriamente el PID del paquete de TS que transmite el campo de PCR válido para un programa de radiodifusión especificado por el *program number*.
- *Program information length* debe asumir obligatoriamente el valor '00' en los dos primeros bits. Los diez bits restantes deben indicar obligatoriamente el número de bytes en el descriptor que sigue la información del *Program information length*.
- *Descriptor 1* debe contener obligatoriamente el campo que contiene el descriptor relacionado al programa utilizado en la radiodifusión.
- *Descriptor 2* debe contener obligatoriamente el campo que contiene el descriptor relacionado al stream elemental utilizado.
- *Stream type* se debe utilizar obligatoriamente para identificar el tipo de elemento del programa de radiodifusión. Los tipos de elementos deben estar de acuerdo con lo descrito en el anexo A7.
- *Elementary PID* debe representar obligatoriamente el identificador para el paquete de TS que transmite el stream elemental asociado al elemento del programa.
- *ES information length* debe asumir obligatoriamente el valor '00' en los dos primeros bits. Los 10 bits restantes deben representar obligatoriamente el número de bytes en el descriptor que sigue la información del *ES information length*.

Continuando con la misma estructura del ejemplo de la tabla PAT y ahora se desea formar una sección transmita una tabla PMT que especifique que los PID

de los paquetes de TS correspondientes al PCR, señal de audio y video para dos servicios de TV Digital son 2064, 2065, 2068, 2075, 2064 y 2065 respectivamente, entonces su estructura debería contener los siguientes datos:

**Tabla 2. 9 Ejemplo de una Tabla PMT**

Parámetro	Valor	Observaciones
Header	El mismo formato que maneja una sección extendida.	El table id debe ser 0x02.
Program information length	Indica obligatoriamente el número de bytes en el descriptor que sigue la información del Program information length	
PMT del primer servicio de TV Digital		
Program number	0x760	Debe ser el mismo que se definió en la tabla PAT para el primer servicio
PCR PID	2064	Indica el PID del TS que contiene el PCR para el primer servicio.
Stream type	0x02	Indica que se trata de un stream de video MPEG-2
Elementary PID	2064	Indica el PID del TS de video
Es information length	Representar obligatoriamente el número de bytes en el descriptor que sigue la información del ES information length.	
Stream type	0x03	Indica que se trata de un stream de audio MPEG-2
Elementary PID	2068	Indica el PID del TS de audio
PMT del segundo servicio de TV Digital		
Program number	0x761	Debe ser el mismo que se definió en la tabla PAT para el primer servicio
PCR PID	2065	Indica el PID del TS que contiene el PCR para el primer servicio.
Stream type	0x02	Indica que se trata de un stream de video MPEG-2
Elementary PID	2065	Indica el PID del TS de video
Es information length	Representar obligatoriamente el número de bytes en el descriptor que sigue la información del ES information length.	
Stream type	0x03	Indica que se trata de un stream de audio MPEG-2
Elementary PID	2075	Indica el PID del TS de audio
CRC	El mismo formato que maneja una sección extendida	

### 2.4.1.3.3 Tabla de acceso condicional (CAT)

Esta tabla proporciona el valor PID para los paquetes que contengan el “Entitlement Management Message (EMM). Los EMMs actualizan las opciones de suscripción o derechos de pay-per-view para cada subscriber.

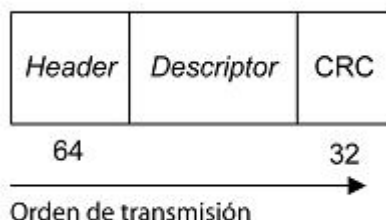


Figura 2. 27 Tabla de acceso condicional

*Header* y *CRC* deben ser los mismos que los del formato de la sección extendida. El contenido del bit que sigue al *Section syntax indicator* debe ser obligatoriamente igual a '0'; el valor de *Table id* debe ser obligatoriamente 0x01 para representar la Tabla CAT. El *Table id extension* no se debe usar, pues está reservado para futura extensión.

### 2.4.1.3.4 Tabla de información de red (NIT)

La NIT debe informar obligatoriamente la organización física del agrupamiento de transport streams (TS) existente en una misma red y sus características, así como los datos relevantes sobre la sintonía de los servicios existentes. En la figura 2.28 se muestra el diagrama de bloques de esta tabla [14].

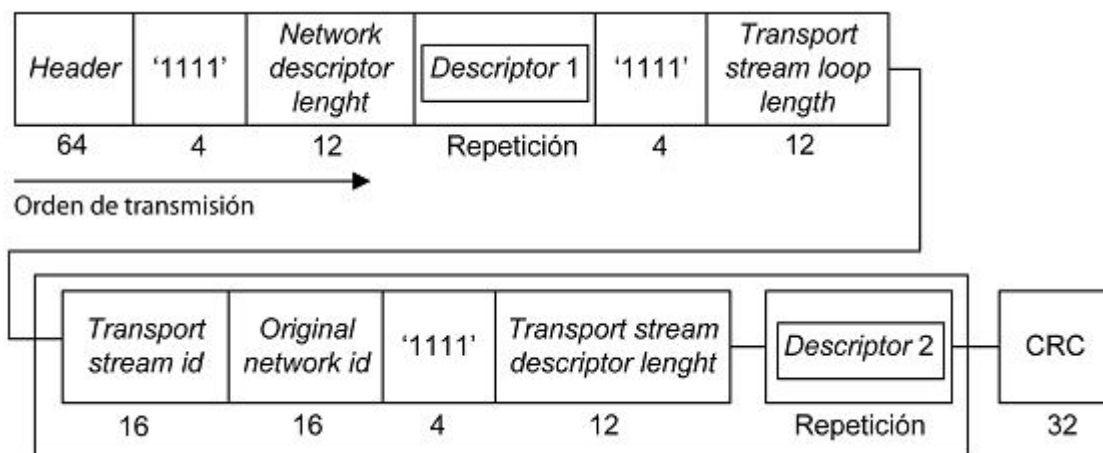


Figura 2. 28 Tabla de información de red

En cuanto a los ítems que se muestran en el diagrama de bloques, deben tener las siguientes descripciones:

- *Header y CRC* deben ser iguales al formato de la sección extendida. El valor del *table id* debe ser obligatoriamente 0x40 para red verdadera y 0x41 para cualquier otra red. El *Table id extension* se debe utilizar obligatoriamente para transmitir el *network id*. El campo *network id* debe identificar obligatoriamente el número de la red.
- *Network description length* debe asumir obligatoriamente el valor '00' en los dos primeros bits. Los diez bits restantes deben obligatoriamente describir el número de bytes en el descriptor que sigue al *Network description length*.
- *Descriptor 1 y descriptor 2* deben obligatoriamente contener los descriptors.
- *Transport stream loop length* debe asumir obligatoriamente el valor '00' en los dos primeros bits. Los diez bits restantes deben describir obligatoriamente el número de bytes en el descriptor que sigue al *Transport stream loop length*.
- *Transport stream id* debe representar obligatoriamente el número de identificación del *transport stream* utilizado.
- *Original network id* debe representar obligatoriamente el número de identificación de la red original del *transport stream* utilizado.
- *Transport stream descriptor length* debe representar obligatoriamente el número de bytes en todos los descriptors del *transport stream* utilizado que viene inmediatamente después de este campo. El valor de los dos primeros bits debe ser obligatoriamente '00'.

Por ejemplo si continuamos con la misma estructura de los ejemplos anteriores y ahora se quiere crear una tabla NIT que contenga dos servicio de TV digital debería contener descriptor de red, descriptor de lista de servicios, descriptor de gestión de sistema, descriptor de sistema de transmisión terrestre y el

descriptor de recepción parcial con las especificaciones que se indican en la tabla 2.10.

**Tabla 2. 10 Ejemplo de tabla CAT**

Parámetro	Valor	Observaciones
Original Network id	0x73b	Identificador de red regional
Transport stream id	0x73b	Identificador de red
<b>Descriptor de Red</b>		
Network name	ESPE	Campo tipo <i>char</i> que indica el nombre de la red
<b>Descriptor de gestión del sistema</b>		
Broadcasting flag	0	Define que el tipo de transmisión es abierta
Broadcasting identifier	3	Indica que el estándar de transmisión es ISDBT
Additional broadcasting identification	0x01	Identificación de los proveedores del servicio
<b>Descriptor de lista de servicios</b>		
Service id	0x760	Identificador de un solo servicio dentro del TS. Debe ser igual al program number.
Service type	1	Especifica que el tipo de servicio es de TV Digital
<b>Descriptor de sistema de transmisión terrestre</b>		
Area code	1341	Código de área del servicio
Guard Interval	0x01	Intervalo de guarda de 1/16.
Transmission mode	0x02	Modo de información #3
Frequency	533 MHz	Frecuencia central de transmisión.
<b>Descriptor de información de TS</b>		
Remote control key id	0x05	Número de canal virtual asociado al TS.
TS name	"ESPE"	Nombre del TS
Transmission type info	0x0F	Información del descriptor según el estándar ISDBT
Service id 1	0x760	Identificador del primer servicio dentro del TS. Debe ser igual al que se definió en el program number en la tabla PAT
Service id 2	0x761	Identificador del primer servicio dentro del TS. Debe ser igual al que se definió en el program number en la tabla PAT

#### 2.4.1.3.5 Servicios de Información (SI).-

También conocida como información de arreglo de programas, se trata de un conjunto de tablas que deben indicar obligatoriamente el arreglo de la secuencia de programa en un canal de transmisión. Todas las tablas mencionadas anteriormente son señales de control de transmisión que deben cumplir con un formato de sección aplicable para procedimientos de transmisión de PES, formatos de sección y paquetes TS, según lo indicado en la tabla 2.11 [14].

Tabla 2. 11 Identificadores y sus funciones

Identificador	Función
Table id	Identifica los tipos de sección
Descriptor tag	Identifica los tipos de descriptores
Stream type	Identifica los tipos de señales codificadas
Service type	Identifica el tipo de servicio
Program number	Identifica el número del programa de radiodifusión
Service id	Identifica el número del servicio de radiodifusión
Network id	Identifica la red
Transport stream id	Identifica el transport stream
CA System id	Identifica el sistema de acceso condicional para la recepción
System management id	Identifica si una transmisión es del tipo abierta o no abierta

#### 2.4.1.4 Paquete Transport Stream

Hasta ahora se han descrito algunos componentes que son los que conforman la estructura del Transport Stream. Esto se lo va a poder apreciar de una manera más clara en el siguiente diagrama de bloques.

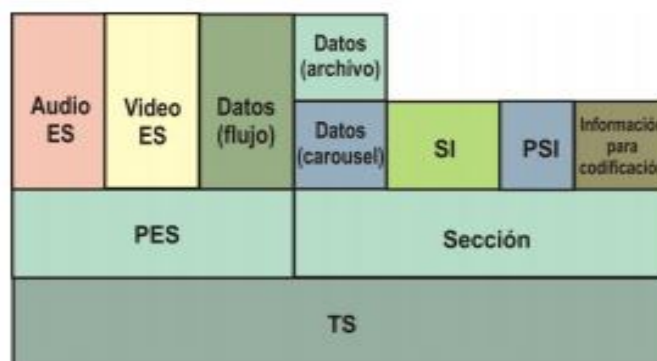
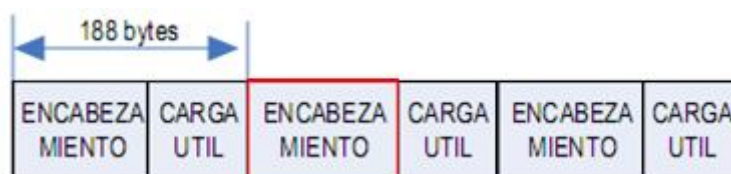


Figura 2. 29 Modelo de Capas para la formación del TS

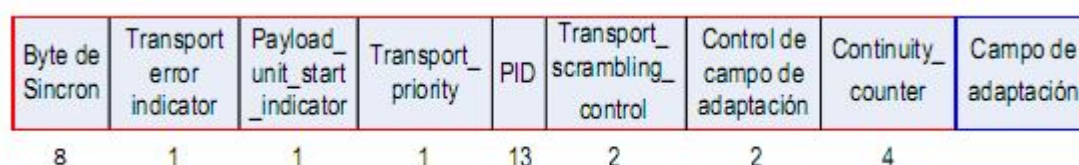


A partir de aquí, luego de que se tienen empaquetados los PES o las secciones, estos se agrupan para formar el Transport Stream, que está conformado por 184 bytes de payload y 4 bytes de cabecera. A continuación se muestra el diagrama de bloques de la estructura que constituye un tren de paquetes de *Transport Stream*.



**Figura 2. 30 Tren de paquetes de transporte**

Al proceso que se observa en la figura 2.30 se lo conoce como remultiplexación del Transport Stream, es decir, se juntan todos los paquetes de TS para conformar un solo flujo o tren de transporte. A partir de aquí empezamos a analizar el contenido de la cabecera de un paquete de TS en la figura 2.31 [16].



**Figura 2. 31 Estructura del encabezamiento de un paquete TS**

En cuanto a los ítems que se muestran en el diagrama de bloques, deben tener las siguientes descripciones:

- *Sync byte (byte de sincronismo)*. - Debe ser obligatoriamente 0x47 para la sincronización de paquetes,
- *Transport error indicator* (indicador de error de transporte). - Debe ser obligatoriamente un flag indicador de la presencia de cualquier error de bit en el paquete TS (TS packet). Si esta señalización contiene el valor '1', indica que hay, al menos, un bit erróneo no corregido dentro del paquete.
- *Payload unit start indicator* (indicador de inicio). - Los paquetes PES se extienden, normalmente, a lo largo de varios paquetes de transporte. El

primer byte del comienzo de un paquete PES se coloca en el primer byte disponible de la carga de un paquete de transporte, hecho que se indica poniendo este bit a “1”.

- *Transport priority*.- Debe ser obligatoriamente un flag que si está en “1” indica que el paquete asociado de transporte tiene más prioridad que otros paquetes con el mismo *PID*.
- *PID*.- Identifica inequívocamente a todos los paquetes pertenecientes a un mismo tipo de dato o de PES. De los  $2^{13}$  valores posibles, hay 17 reservados para funciones especiales. Los tipos de datos del payload deben estar de acuerdo con la tabla 2.8.
- *Transport scrambling control* (control de codificación de transporte).- Debe ser obligatoriamente un campo que identifica el modo de codificación (scrambling mode) del payload para el paquete TS.
- *Adaptation field control* (control del campo de adaptación).- Indica si la cabecera va seguida por un campo de adaptación y/o carga útil.
- *Continuity index* (índice de continuidad).- Se incrementa con cada paquete del mismo PID que contenga carga útil, es decir, Cuando el control de campo de adaptación es 01 o 11. El valor de este campo debe obligatoriamente empezar con ‘0000’ y debe ser obligatoriamente incrementado en 1.
- *Adaptation field* (campo de adaptación).- Debe cumplir obligatoriamente la ISO/IEC 13818-1.

**Tabla 2. 12 Atribución del PID**

Valor	Descripción
0x0000	PAT
0x0001	CAT
0x0002-0x000F	Reservado
0x0010	NIT

0x0011 – 0x1FFE	Otras tablas, excluyéndose PAT, CAT, NIT y paquetes nulos
0x1FFF	Paquetes nulos

Tabla 2. 13 Valores de control de campo adaptable

Valor	Descripción
00	Reservado
01	Sin campo adaptable, solo payload
10	Solo campo adaptable, sin payload
11	Campo adaptable siguiendo el payload

Finalmente tenemos el campo de adaptación (Adaptation field) que se divide en otros bloques opcionales y banderas que se muestran en la figura 2.32 [17].

Adaption field length	Discontinuity Indicator	Random access indicator	ES priority indicator	5 banderas	Campos opcionales	Bytes de relleno
-----------------------	-------------------------	-------------------------	-----------------------	------------	-------------------	------------------

Figura 2. 32 Estructura del Campo de adaptación

En cuanto a los ítems que se muestran en el diagrama de bloques, deben tener las siguientes descripciones:

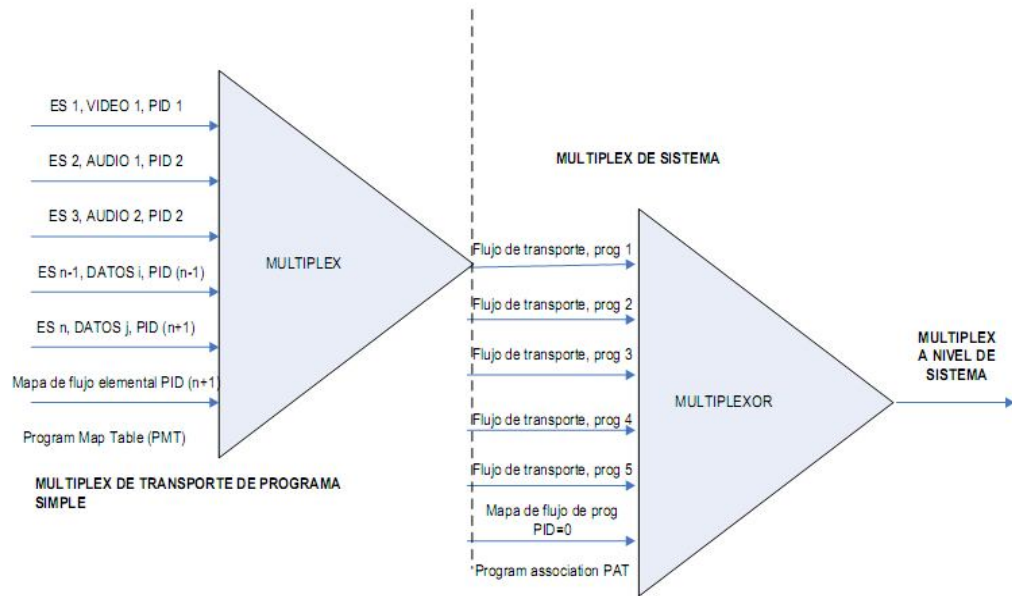
- *Adaptation field Length* (Longitud de campo de adaptación).- Es un campo de 8 bits que especifica el número de bytes del *adaptation field* que se encuentran inmediatamente a continuación de este parámetro.
- *Discontinuity indicator* (Indicador de discontinuidad).- Se trata de un campo de 1 bit que cuando tiene el valor de '1', estado de discontinuidad es verdadero para el paquete actual de *Transport Stream*. Al hablar de

discontinuidades se puede tratar de dos tipos: Discontinuidad del sistema de tiempo base y discontinuidad del contador.

- *Random Access indicator* (Indicador de acceso aleatorio).- Se trata de un campo de 1 bit que indica si el paquete actual Transport Stream y el posible paquete subsecuente con el mismo PID, contiene cierta información para ayudar el acceso aleatorio en este punto.
- *Elementary Stream priority indicator* (Indicador de prioridad de Flujo Elemental).- Es un campo de 1 bit, que entre paquetes con el mismo PID, indica la prioridad de la información del elementary stream que se encuentra en el payload de este paquete de Transport Stream.
- *Banderas*: Es un campo de 5 bits que contiene banderas relacionadas con el PCR, OPCR, Puntos de empalme, datos privados de transporte y extensiones del campo de adaptación.
- *Stuffing bytes* (Bytes de relleno).- Se trata de un campo de 8 bits fijos con un valor igual a '11111111' que pueden ser insertados por el codificador pero que son descartados por el decodificador.

#### **2.4.2 Multiplexación de Servicios**

Como se puede observar en la figura 2.33 la multiplexación de varios servicios elementales ES (Elementary Services) como son audio, video, datos, con su PID, así como el flujo de control llamado PMT que representa la Tabla del mapa de programa con sus PID's y tipos de flujo de video, audio o datos, forman un flujo de transporte de programa (PS, Program Stream) (Capa 1). El multiplexado de diferentes flujos de transporte de programa da como resultado el multiplex a nivel de sistema (Capa2), al multiplex del sistema ingresan los diferentes flujos de transporte de programas, con sus identificaciones correspondientes (PID's), además ingresa un flujo que constituye el mapa de flujo de programa con PID=0, este flujo lleva la Tabla de asociación de programas (PAT), todos los flujos deben tener la misma base de tiempo.



**Figura 2. 33 Multiplexado de flujo de datos**

El proceso de identificar un programa y su contenido se realiza en dos etapas. En la primera etapa, se utiliza la PAT en flujo e bits PID=0 para identificar el flujo de bits que lleva PMT para el programa. En la segunda etapa, se obtienen las identificaciones (PID's) de flujos elementales de bits que conforman el programa, consultando la PMT respectiva.

## **CAPITULO 3**

### **GENERACIÓN DE TRANSPORT STREAM**

#### **3.1 SOFTWARE DE GENERACIÓN DE TRASPORT STREAM**

Para la generación del flujo de transporte se utiliza un software de distribución libre llamado OpenCaster V. 2,4 que opera sobre un sistema Linux Debian estable, con un kernel personalizado y atributos web GUIs para la programación de eventos y videos.

Este software nos permite en primer lugar generar el Transport Stream para MPEG-2. Además incorpora funciones para procesar, multiplexar, presentar, analizar y transmitir el contenido relacionado con el Transport Stream de MPEG-2.

Las entradas para OpenCaster son de audio y video y pueden provenir de un servidor FTP (File Transfer Protocol), una señal de satélite o estar almacenadas en un disco duro. Las señales son convertidas al formato adecuado para luego empezar a generar los flujos elementales, los flujos de paquetes elementales y las tablas de información específica necesarias para la generación de un transport stream, que puede estar especificado a cualquier tasa de bit y con una marca PCR local, para luego pasar a la sincronización de los flujos de transporte tanto de audio como de video. Es muy importante recalcar que OpenCaster no es un codificador MPEG de audio y video.

Además se ofrece soporte para la transmisión avanzada de datos a través de DSM-CC, incorporación de aplicaciones GINGA por medio de la creación y transmisión de carruseles de objetos y datos.

La salida mediante DVB-T puede ser directamente recibida por Set-top boxes de bajo costo o televisores digitales con decodificadores de hardware y un sintonizador de DVB integrado.

##### **3.1.1 Características de OpenCaster**

OpenCaster V. 2.4 presenta las siguientes características dentro de su funcionamiento:

- Creación de tablas PSI/SI
- Remultiplexación de flujos de transporte UDP.
- Herramientas de conversión “m2ts”
- Optimización de Carrusel DSM-CC mejorada  
Soporte para TV interactiva a través de Ginga y HbbTV  
Remultiplexación de señales de entrada.
- Tiene un operador ip multicast para servicios de IPTV.
- Soporte para teletexto
- Emitir cualquier tipo de archivo de video, audio o datos, cuando esta correctamente codificado
- Manipular Transport Stream en tiempo real cuando está almacenado en un disco duro.
- Transformar los archivos de sistema en carruseles DSM-CC
- Generar otros tipos de archivos y transmitirlos sobre el Transport Stream.
- Multiplexación de Transport Streams con tasas de bits constantes (CBR)
- Permite agregar marcas de tiempo como PTS, PCR y DTS.
- Salida a DVB-ASI con PCI y tarjetas de hardware USB.
- Salida a DVB-S, DVB-T y DVB-C con tarjetas de hardware Dektec DTA110T.
- Permite múltiples carruseles por servicio [1].

### 3.1.2 Instalación de OpenCaster

OpenCaster es un software de libre distribución que corre bajo una plataforma Linux. Antes de empezar a instalar OpenCaster se debe verificar que se cumplan los siguientes requerimientos:

- Compilador de C de GCC con una versión 4.4.3 o superior
- Ambiente python versión 2.6.5

En ocasiones el ambiente python puede no encontrarse instalado inicialmente, si este es el caso se lo puede instalar mediante el siguiente comando:

```
sudo apt-get install python-dev
```

Además si se quiere comprobar la versión del paquete instalado se lo puede realizar de la siguiente manera:

```
python --version
```

```
gcc --version
```

A continuación procedemos a descargar las fuentes y el manual del programa en el sitio original de OpenCaster en la siguiente dirección:

```
http://www.avalpa.com/
```

También podemos descargar la versión 2.4 de OpenCaster de la página de LIFIA (Laboratorio de Investigación y Formación en Informática Avanzada), donde además encontramos un parche que sirve para aplicar modificaciones realizadas a la versión 2.4 y un archivo de lectura con la descripción de las modificaciones. Todos estos archivos los podemos descargar de la siguiente página:

```
ftp://tvd.lifia.info.unlp.edu.ar/OpenCaster2.4/
```

Es recomendable descargar la versión 2.4 de OpenCaster ya que se esta versión tiene asociado un parche que lo desarrollaron los investigadores del laboratorio LIFIA, para que el programa pueda ejecutarse sin problemas. Este parche se lo puede descargar en la misma dirección que se indicó anteriormente. Cuando tengamos la fuente comprimida de OpenCaster lo pegamos en cualquier ubicación. En este caso vamos a considerar que está guardado en:

```
/home//OpenCaster/
```

A continuación procedemos a descomprimir las fuentes:

```
$ cd /home//OpenCaster/
```

```
$ tar xzf OpenCaster3.1.tgz
```

El siguiente paso es parchar el programa:

```
$ cd OpenCaster2.4/
```



```
$ patch -p 1 < ../OpenCaster2.4-lifia-rev362.patch
```

Luego procedemos a instalar con privilegios de root:

```
sudo make install
```

Una vez que el software está instalado, ya se pueden generar transport streams. Para probar la instalación, ejecutar el siguiente comando:

```
$ python -c "from dvbobjects.PSI.PAT import *"
```

El comando debería ejecutar sin mostrar ningún error en la salida.

Finalmente instalamos la aplicación *ffmpeg* ingresando la siguiente línea de comando en la terminal:

```
$ sudo apt-get install ffmpeg
```

### **3.1.3 Configuración de los parámetros para la generación del Transport Stream**

En esta sección se explica los parámetros que se deben especificar para generar un Transport Stream de un video para que pueda ser transmitido por medio de una tarjeta moduladora, posteriormente a un set-top box, del estándar ISDB-TB, que será el encargado de decodificar la señal y finalmente presentarla en la televisión.

#### **3.1.3.1 Configuración de los parámetros y funciones útiles para la señal de video**

##### **3.1.3.1.1 ffmpeg**

Dentro de las librerías de OpenCaster, específicamente en libvacodec, está disponible una función llamada *ffmpeg*, que permite codificar las señales de audio y video. Esta función nos permite codificar una señal de video que se encuentra en un formato AVI y pasarla al formato mpeg-2 mediante códecs de compresión [3].

A continuación se muestra un ejemplo de cómo debe ser utilizada esta función:

```
ffmpeg -i input.txt -an -vcodec mpeg2video -f mpeg2video -s 720x576 -r
25 -aspect 4:3 -deinterlace -b 5000k -maxrate 5000k -minrate 5000k -bf
2 -bufsize 1835008 video.mp2
```

Donde,

- i: Archivo de entrada que puede ser el archivo de video soportado por el codificador *ffmpeg*
- an: No audio, ignora a la señal de audio si es que está presente en la entrada
- vcodec : Se trata del códec de video, en este caso se trata de mpeg2video que es el códec que necesitamos para que sea compatible con nuestra señal de Transport Stream
- f: Formato de salida, en este caso el formato que necesitamos es mpeg2video
- b: Es la tasa de bits de la señal codificada en kbps A continuación se utilizan los parámetros maxrate y minrate que son las tasas máximas y mínimas respectivamente.
- s: Es el formato del tamaño de presentación del video.
- Bf: Numero de cuadros del tipo b para cada GOP
- Bufsize: Es el tamaño del buffer en bits. Esta cantidad está directamente relacionada con el buffer VBV (video buffer verifier) mediante la siguiente expresión:
- $$vbv\_buffer\_size * 1024 * 16 = buffer\_size = 1835008$$
- aspect: Nos indica el formato de la presentación en la pantalla. Para poder visualizar todos los formatos disponibles se debe ingresar el comando “man ffmpeg” en la terminal

r: Es el número de frames por segundo. El normal es de 25 frames.

deinterlace: Define una opción para desentrelazar imágenes.

Existen otros parámetros adicionales que nos permiten configurar ciertas características especiales y para tener mayor detalle sobre ellos ingresamos el comando “man ffmpeg” en la terminal.

#### 3.1.3.1.2 *esvideo2pes*

La función *esvideo2pes* nos permite transformar el contenido de video que ha sido codificado por la función *ffmpeg* y empaquetarlo en una trama PES de la siguiente manera:

```
esvideo2pes fuente.m2v > salida.pes
```

Además se puede analizar la salida con el siguiente comando:

```
pesinfo salida.pes
```

Con este comando podremos ver un reporte de informaciones sobre los flujos elementales del programa, mostrando datos como el PTS y el DTS y la longitud de la cabecera de cada stream.

#### 3.1.3.1.3 *pesvideo2ts*

Esta función permite encapsular el video en una trama de tipo Transport Stream a partir de una trama PES. A continuación se muestra un ejemplo de cómo debe ser utilizada esta función:

```
pesvideo2ts PID fps vbv ab 0 salida.pes > video.ts
```

donde,

PID: Es el número del PID del flujo de video.

Fps: Es el número de cuadros (frames) por segundo. El valor por default es de 25.

Ab: Es el ancho de banda de la señal de transport stream. Es muy importante que el ancho de banda sea más grande que la tasa de bits de la señal de video definida en la función *ffmpeg*. Es recomendable manejar un 15% más grande.

Además existen otros parámetros como *yuvdenoise* que pueden especificarse en este comando para mejorar la calidad de compresión.

Para tener mayor detalle sobre ellos ingresamos el comando “man *pesvideo2ts*” en la terminal.

### **3.1.3.2 Configuración de los parámetros y funciones útiles para la señal de audio**

#### **3.1.3.2.1 *ffmpeg***

En la sección anterior explicamos la función *ffmpeg* pero orientada a la codificación de una señal de video. Esta misma función nos permite codificar la señal de audio, separarla de la señal original y transformarla a un determinado formato (mp2) para poder transformarla en una señal de Transport Stream..

A continuación se muestra un ejemplo de cómo debe ser utilizada esta función:

```
ffmpeg -i input.avi -vn -ac 2 -acodec mp2 -f mp2 -ab 128000 -ar 48000  
audio.mp2
```

Donde,

- i : Archivo de entrada que puede ser el archivo de audio soportado por el codificador *ffmpeg*
- vn: No video, ignora a la señal de video si es que está presente en la entrada
- ac: Define el número de canales. Por defecto es igual a 1.

- acodec : Se trata del códec de audio, en este caso se trata de mp2 que es el códec que necesitamos para que sea compatible con nuestra señal de Transport Stream
- f: Formato de salida, en este caso el formato que necesitamos es mp2.
- ab: Es la tasa de bits de la señal de audio en bits/s. En este ejemplo la tasa de bits es de 128kbps.
- ar: Define la frecuencia de muestreo de la señal de audio. La frecuencia por defecto es de 44100 Hz.

Existen otros parámetros adicionales que nos permiten configurar ciertas características especiales y para tener mayor detalle sobre ellos ingresamos el comando “man ffmpeg” en la terminal.

#### 3.1.3.2.2 *esaudio2pes*

La función *esaudio2pes* nos permite transformar el contenido de audio que ha sido codificado por la función *ffmpeg* y empaquetarlo en una trama PES de la siguiente manera:

```
esaudio2pes audio.mp2 spf ar fs 3600 > audio.pes
```

donde,

- spf: Es la cantidad de muestras por frame. Un valor usualmente utilizado para este parámetro es 1152.
- ar: Define la frecuencia de muestreo de la señal de audio.
- fs: Define el tamaño del frame. Un valor usualmente utilizado para este parámetro es 384.
- 3600: Define el desplazamiento del PTS para sincronizar con el video.

Además se puede analizar la salida con el siguiente comando:

```
pesinfo salida.pes
```

Con este comando podremos ver un reporte de informaciones sobre los flujos elementales del programa, mostrando datos como el PTS y el DTS y la longitud de la cabecera de cada stream.

### 3.1.3.2.3 pesaudio2ts

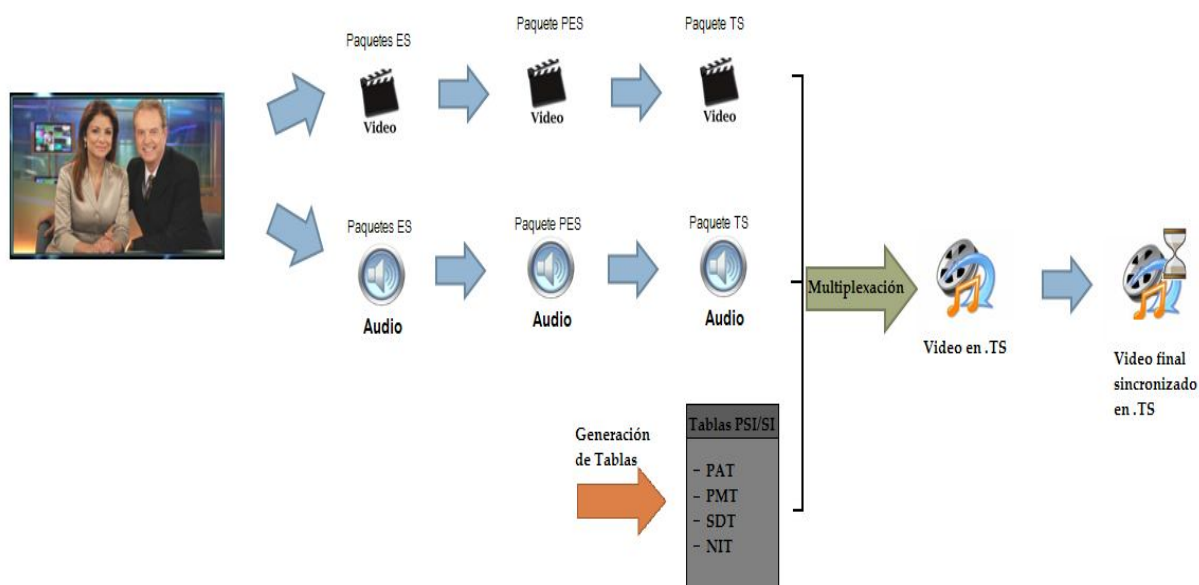
Esta función permite encapsular el audio en una trama de tipo Transport Stream a partir de una trama PES. A continuación se muestra un ejemplo de cómo debe ser utilizada esta función:

```
pesaudio2ts PID spf ar fs 0 audio.pes > audio.ts
```

En esta función se repiten la mayoría de los parámetros definidos en el comando anterior con excepción de PID que se refiere al número correspondiente de PID de la señal de audio.

## 3.2 GENERACIÓN DE TRANSPORT STREAM

En esta sección se va a detallar el proceso completo para la generación de un Transport Stream de un video que contenga un servicio de televisión digital y posteriormente como generar un transport stream que contenga una aplicación Ginga NCL en el servicio. En la figura 3. Se puede observar el proceso de manera global que va a ser detallado en las secciones a continuación.



**Figura. 3. 1 Proceso de Generación del Transport Stream**

### 3.2.1 Generación de un Transport Stream con un servicio de TV Digital

En primer lugar el video que va a ser convertido debe contener extensión .avi para una mejor compatibilidad con el software de generación.

#### 3.2.1.1 Conversión del video original

Si se tiene problemas con esta conversión y se necesita transformar un video en formato .mpg a .avi, se puede utilizar una herramienta que nos facilita Ubuntu.

Se necesita ingresar el siguiente comando:

```
$ mencoder video.mpg -ofps 25 -ovc xvid -oac mp3lame -lameopts abr:br=192 -  
srate 48000 -xvidencopts fixed_quant=4 -o video2.avi
```

Donde,

video.mpg: Video a convertir  
ofps 25: 25 cuadros por segundo  
video2.avi: Video convertido.

#### 3.2.1.2 Obtención de la información del video original

En esta sección se necesita conocer la información con la que ha sido codificado el video y el audio original, especialmente nuestro interés es conocer la tasas de bits utilizadas en la codificación para poder utilizar la misma tasa en el proceso de generación del transport stream y de esta manera no se pierda información ni se produzcan errores de sincronización.

Para realizar esto utilizamos la herramienta “idvid” que se ejecuta en desde la consola de Ubuntu. Si no se tienes instalada este paquete se lo puede realizar con el siguiente comando desde la consola:

```
$ sudo apt get-install tovid
```

Cuando ya se tiene instalada esta herramienta desde la consola nos dirigimos al directorio donde se encuentra nuestro video de interés e ingresamos el siguiente comando:

*\$ idvid video2.avi*

Al ingresar este comando vamos a obtener un resultado similar al siguiente:

```
usubuntu@UB-110210-1744:~/Escritorio$ idvid video2.avi
-----
idvid
Video identification script
Part of the tovid suite, version 0.31
http://www.tovid.org
-----
Analyzing file: 'video2.avi'. This may take several minutes...
=====
          File: video2.avi
          Width: 720 pixels
          Height: 480 pixels
    Aspect ratio: 1.33:1
          Frames: 534
      Duration: 00:00:21 hours/mins/secs
    Framerate: 25.000 frames per second
    Video format: XVID
    Video bitrate: 4000000 bits per second
-----
Audio track 1 (Stream 0.1, AID 0):
-----
          Codec: mp3
          Bitrate: 0000 bits per second
    Sampling rate: 48000 Hz
=====
Audio is compliant with the following formats:
  Not compliant with (S)VCD or DVD
Video is compliant with the following formats:
  Not compliant with (S)VCD or DVD
This video does not seem to be compliant with (S)VCD or DVD
standards. If you burn it to a video disc, it may not work.
=====
```

**Figura. 3. 2 Resultado mostrado por el comando IDVID**

Como se puede observar en la imagen 3.2, nos muestra toda la información del video que deseamos convertir a formato .ts, para nuestro caso la información relevante va a ser las tasa de bit (bit rate) del video y la



frecuencia de muestreo del audio. Estos datos los vamos a tomar en cuenta para el siguiente paso del proceso de generación del Transport Stream.

### 3.2.1.3 Creación de los archivos .ts de audio y video

En esta sección se va a utilizar los comandos descritos en la sección 3.1.3.1 y 3.1.3.2 para generar los paquetes de audio y video .es (Elementary stream), .pes (packetized elementary stream) y finalmente .ts (transport stream).

Para realizar este proceso empezamos obteniendo un archivo que contenga únicamente el video (no audio). Nótese que en este comando se debe utilizar la tasa de bits que tenía el video original en la sección anterior. En este caso se debe utilizar esa tasa tanto para las tasas mínimas y máximas como se muestra en la siguiente línea de comando:

```
$ ffmpeg -i video2.avi -an -vcodec mpeg2video -f mpeg2video -s 720x576 -r 25  
-aspect 4:3 -deinterlace -b 4000k -minrate 4000k -maxrate 4000k -bf 2 -bufsize  
1835008 pruebavideo.m2v
```

La descripción de los parámetros utilizados se la puede encontrar en la sección 3.1.3.1. En cuanto al bufsize es un valor que no deberá variar y depende del parámetro vbv que se relaciona mediante la siguiente expresión:

$$Bufsize = vbv\_buffer\_size * 1024 * 16$$

El video buffer verifier no es más que un modelo teórico que simula un buffer de video que se utiliza para asegurar que un flujo codificado de video va a poder ser correctamente almacenada y reproducido en el decodificador.

El tamaño del buffer vbv utilizado en para mpeg-2 es de 112kbps. Otro tamaño es 224 kbps, esto depende más del tipo de decodificador que se vaya a utilizar. Con estos datos tenemos la siguiente expresión:

$$Bufsize = 112 * 1024 * 16 = 1835008$$

El siguiente paso es transformar el archivo de video generado en formato .m2v a un flujo de paquetes con extensión .pes. Para realizar esta operación se utiliza la siguiente línea de comando:

```
$ esvideo2pes pruebavideo.m2v > pruebavideo.pes
```

Posteriormente generamos el transport stream de la señal de video con la siguiente línea de comando:

```
$ pesvideo2ts 2065 25 112 4600000 0 pruebavideo.pes > pruebavideo.ts
```

Los parámetros utilizados en el paso anterior se explican en detalle en la sección 3.1.3.1. En cuanto al parámetro del ancho de banda, este debe ser un 15% mayor que la tasa de bits de la señal de video. En el ejemplo la tasa de bits es de 4Mbps por lo cual el ancho de banda que se utilice debe ser al menos de 4.6Mbps.

Es importante mencionar que la función “*pesvideo2ts*” genera un bucle infinito por lo que se debe detener su ejecución mediante el comando “CTRL + C” una vez que se compruebe que el archivo generado con extensión .ts ya no varíe su tamaño. Para poder comprobar esto se puede utilizar la siguiente función desde una consola diferente:

```
$ ls-l pruebaaudio-ts
```

Se debe ejecutar la función anterior en repetidas ocasiones y verificar que el tamaño del archivo *pruebaaudio.ts* no varíe su tamaño.

Hasta este punto se tiene generado el flujo de transporte únicamente de la señal de video. Ahora el siguiente paso es generar el flujo de transporte de la señal de audio. Este proceso es muy similar al anterior y el detalle de los comandos que se van a utilizar se explican en la sección 3.1.3.2.

El primer paso consiste en separar el audio del archivo original mediante la función *ffmpeg*. Los parámetros que se utilizan en esta función se explican en detalle en las secciones anteriores. En cuanto a la frecuencia de muestreo y la tasa de bits, se deben utilizar los valores obtenidos de la función “*idvid*” donde muestra la información de codificación del audio original. De esta manera se asegura que no se pierda información ni se produzcan errores.

El comando quedaría de la siguiente manera:

```
$ ffmpeg -i video2.avi -vn -ac 2 -acodec mp2 -f mp2 -ab 128000 -ar 48000  
pruebaaudio.mp2
```

Los parámetros utilizados en esta función se explican en mayor detalle en la sección 3.1.3.2. En cuanto a los valores de tasa de bits y frecuencia de muestreo se debe utilizar los mismos valores que se obtuvieron al utilizar la función “*idvid*” donde mostraba los datos de codificación del video original. En este caso obtuvimos 128kbps y 48kHz respectivamente.

El siguiente paso consiste en generar el flujo de paquetes con extensión .pes mediante la siguiente línea de comando:

```
$ esaudio2pes pruebaaudio.mp2 1152 48000 384 3600 > pruebaaudio.pes
```

Los valores de los parámetros de cantidad de muestras por frames, frame size y PTS son valores estándares utilizados en codificación de audio. En cuanto a la tasa frecuencia de muestreo, debe ser la misma que la utilizada en la función “*ffmpeg*”.

El paso final de esta sección del proceso consiste en generar el flujo de transporte de la señal de audio. Para lo cual se utiliza la siguiente línea de comando:

```
$ pesaudio2ts 2075 1152 48000 384 0 pruebaaudio.pes > pruebaaudio.ts
```

Esta función genera un bucle infinito por lo que se debe detener su ejecución mediante el comando “CTRL + C” una vez que se compruebe que el archivo generado con extensión .ts ya no varíe su tamaño. Para poder comprobar esto se puede utilizar la siguiente función desde una consola diferente:

*\$ ls-l pruebaaudio-ts*

Se debe ejecutar la función anterior en repetidas ocasiones y verificar que el tamaño del archivo *pruebaaudio.ts* no varíe su tamaño. En cuanto a la función “*pesaudio2ts*” se puede notar que se repiten varios de los parámetros utilizado en la función anterior. En cuanto al PID es importante tomar en cuenta que debe ser diferente al valor de PID utilizado para la generación del flujo de transporte de la señal de video y estos valores se los deberá utilizar en la formación de las tablas de identificación que se explicarán en la siguiente sección.

#### **3.2.1.4 Creación de las tablas PSI/SI**

Hasta este punto hemos generado los flujos de transporte de las señales de audio y video de manera independiente, el siguiente paso es remultiplexar ambas señales en un solo flujo de transporte.

Primero se deben generar las tablas PSI/SI, para esto vamos a generar un script en python en un archivo al que llamaremos *gtable.py* y lo guardamos en el mismo directorio donde se encuentran los otros archivos que generamos anteriormente.

##### **3.2.1.4.1 Generación del script que contiene las tablas PSI/SI**

###### *a) Encabezado del Script*

Primero se empieza escribiendo el encabezado del archivo *gtable.py* como se muestra:

```
#!/usr/bin/env python
import os
```

```
from dvbobjects.PSI.PAT import *
from dvbobjects.PSI.NIT import *
from dvbobjects.PSI.SDT import *
from dvbobjects.PSI.PMT import *
from dvbobjects.SBTVD.Descriptors import *

tvd_ts_id = 0x073b # ID de red.
tvd_orig_network_id = 0x073b # ID de red original.
ts_freq = 533 # Frecuencia de transmisión
ts_remote_control_key = 0x05 # Tecla de control remoto.
tvd_service_id_sd = 0xe760 # ID de servicio de TV Digital.
tvd_pmt_pid_sd = 1031 # PID de la PMT del servicio.
```

Además de los encabezados propios de cualquier script python, se incluye la librería OpenCaster y definen identificadores para el Transport stream. Se detallan:

- *tvd\_ts\_id*: Es el identificador del Transport stream.
- *tvd\_orig\_network\_id*: Es el identificador de red original.
- *ts\_freq*: es la frecuencia en que se transmite el Transport stream, en este caso 533 MHz
- *ts\_remote\_control\_key*: Es la tecla de control remoto virtual, sirve para poder usar el control remoto para elegir el canal más rápido.
- *tvd\_service\_id\_sd*: Es el identificador del servicio de TV digital.
- *tvd\_pmt\_pid\_sd*: Es el PID que se usará para transmitir la información que componen el servicio.

#### *b) Definición de la red: NIT*

En el siguiente listado se muestra el contenido de la información de red actual, mediante el uso de la tabla NIT.

```
nit = network_information_section(
    network_id = tvd_orig_network_id,
    network_descriptor_loop = [
        network_descriptor(network_name = "ESPE"),
```

```
        system_management_descriptor(  
            broadcasting_flag = 0,  
            broadcasting_identifier = 3,  
            additional_broadcasting_identification = 0x01,  
            additional_identification_bytes = [],  
        )  
    ],  
    transport_stream_loop = [  
        transport_stream_loop_item(  
            transport_stream_id = tvd_ts_id,  
            original_network_id = tvd_orig_network_id,  
            transport_descriptor_loop = [  
                service_list_descriptor(  
                    dvb_service_descriptor_loop = [  
                        service_descriptor_loop_item (  
                            service_ID = tvd_service_id_sd,  
                            service_type = 1,  
                        ),  
                    ],  
                ),  
            ],  
        ),  
        terrestrial_delivery_system_descriptor(  
            area_code = 1341,  
            guard_interval = 0x01,  
            transmission_mode = 0x02,  
            frequencies = [  
                tds_frequency_item( freq=ts_freq )  
            ],  
        ),  
        partial_reception_descriptor (  
            service_ids = []  
        ),  
        transport_stream_information_descriptor (  
            remote_control_key_id = ts_remote_control_key,  
            ts_name = "ESPE",  
            transmission_type_loop = [  
                transmission_type_loop_item(  
                    transmission_type_info = 0x0F,  

```

```

        service_id_loop = [
            service_id_loop_item(
                service_id=tv_d_service_id_sd
            ),
        ]
    ),
    transmission_type_loop_item(
        transmission_type_info = 0xAF,
        service_id_loop = [],
    ),
],
)
],
),
],
    version_number = 0,
    section_number = 0,
    last_section_number = 0,
)

```

Entre los parámetros específicos del SBTVD y SATVD están:

- *system\_management\_descriptor*: Define propiedades del sistema, definiendo que es el sistema ISDB-Tb, y que está transmitiendo ahora.
- *terrestrial\_delivery\_system\_descriptor*: Define propiedades de la modulación, como intervalos de guarda, frecuencia de transmisión, etc.
- *partial\_reception\_descriptor*: Define la lista de servicios de recepción parcial.  
Esta lista tendría que tener la lista de servicios 1-Seg.
- *transport\_stream\_information\_descriptor*: Define otras propiedades del Transport stream que se está creando, como la tecla de control remoto, el nombre del Transport stream, información de los tipos de servicios ofrecidos, etc.

c) *Definición de servicio: SDT*

En el siguiente listado se muestra el contenido de la tabla SDT que sirve para especificar los servicios disponibles en el Transport stream.

```
sdt = service_description_section(
    transport_stream_id = tvd_ts_id,
    original_network_id = tvd_orig_network_id,
    service_loop = [
        service_loop_item(
            service_ID = tvd_service_id_sd,
            EIT_schedule_flag = 0,
            EIT_present_following_flag = 0,
            running_status = 4,
            free_CA_mode = 7 0,
            service_descriptor_loop = [
                service_descriptor(
                    service_type = 1,
                    service_provider_name = "",
                    service_name = "ESPE",
                ),
            ],
        ),
    ],
    version_number = 0,
    section_number = 0,
    last_section_number = 0,
)
```

d) *Definición del mapa de programas: PAT*

En el siguiente listado se muestra el contenido de la tabla PAT. La tabla PAT tiene el mapa de servicios en PMTs. Las PMTs son las tablas que definen como se compone cada servicio y se detalla en la próxima sección.

```
pat = program_association_section(
    transport_stream_id = tvd_ts_id,
    program_loop = [
```



```
    program_loop_item(  
        # Programa especial para la tabla NIT  
        program_number = 0,  
        PID = 16,  
    ),  
    program_loop_item(  
        program_number = tvd_service_id_sd,  
        PID = tvd_pmt_pid_sd,  
    ),  
],  
version_number = 0,  
section_number = 0,  
last_section_number = 0,  
)
```

*e) Definición de los componentes del servicio: PMT*

En el siguiente listado se muestra el contenido de la tabla PMT. A diferencia de las tablas anteriores, tiene que haber una PMT por cada servicio que se transmita. En este caso solo hay una sola.

Esta tabla define que flujos de datos componen el servicio. En ella se detallan cual es el stream de audio, de video, etc. que componen el servicio. También se define en que PID se encuentra el PCR que es la marca de reloj del servicio.

En esta tabla PMT se define que el servicio se compone de un stream de video MPEG2 en el PID 2065, y de un stream de audio MPEG2 en el PID 2075. Además se detalla que el PID de PCR es el 2065, el mismo que el de video.

```
pmt_sd = program_map_section(  
    program_number = tvd_service_id_sd,  
    PCR_PID = 2064,  
    program_info_descriptor_loop = [],  
    stream_loop = [  
        stream_loop_item(  
            stream_type = 2, # mpeg2 video stream type  
            elementary_PID = 2064,  
            element_info_descriptor_loop = [  

```

```
        ]
    ),
    stream_loop_item(
        stream_type = 3, # mpeg2 audio stream type
        elementary_PID = 2068,
        element_info_descriptor_loop = []
    ),
],
version_number = 0,
section_number = 0,
last_section_number = 0,
)
```

f) Escribiendo las tablas a archivos

En el siguiente listado se muestra el código necesario para escribir las tablas anteriores en archivos. Estos archivos son parte de los fuentes necesarios para generar el multiplexado final del Transport stream.

```
out = open("./nit.sec", "wb")
out.write(nit.pack())
out.close
out = open("./nit.sec", "wb") # python flush bug
out.close
os.system("sec2ts 16 < ./nit.sec > ./nit.ts")
out = open("./pat.sec", "wb")
out.write(pat.pack())
out.close
out = open("./pat.sec", "wb") # python flush bug
out.close
os.system("sec2ts 0 < ./pat.sec > ./pat.ts")
out = open("./sdt.sec", "wb")
out.write(sdt.pack())
out.close
out = open("./sdt.sec", "wb") # python flush bug
out.close
os.system("sec2ts 17 < ./sdt.sec > ./sdt.ts")
```

```
out = open("./pmt_sd.sec", "wb")
out.write(pmt_sd.pack())
out.close
out = open("./pmt_sd.sec", "wb") # python flush bug
out.close
os.system("sec2ts " + str(tvd_pmt_pid_sd) + " < ./pmt_sd.sec > ./pmt_sd.ts")
```

### 3.2.1.5 Compilación de las tablas y Remultiplexación de los flujos de transporte de audio y video

En la sección anterior dejamos listo todo el código que debe tener el script que define las tablas SI/PSI al que llamamos “*gtable.py*”, el siguiente paso es ejecutar el código anterior para generar las secciones que componen las tablas.

Al ejecutar el script, se generan archivos .sec que son las secciones de las tablas con la información detallada, y archivos .ts con las secciones embebidas en paquetes de 188 bytes. Para poder ejecutar estas tablas se utiliza las siguientes líneas de comando, en el mismo directorio que el resto de archivos, desde la consola:

```
$ chmod u+x gtables.py
$ ./gtables.py
```

Finalmente en el directorio donde se ha venido trabajando se encuentran los archivos de flujo de transporte del video, audio y las tablas generadas en el paso anterior. Posteriormente creamos un archivo en blanco al que llamaremos “*null.ts*”.

El siguiente paso es ingresar la función que nos permite multiplexar el flujo de video con el flujo de audio. Para realizar esta operación se ingresa el siguiente comando:

```
$ tscbrmuxer 600000 b:15040 pat.ts b:15040 pmt_sd.ts b:3008 sdt.ts b:3008
nit.ts b:4600000 pruebavideo.ts b:128000 pruebaaudio.ts b:25209238 null.ts >
pruebafinal.ts
```

Donde,

**600000**: Es la cantidad de paquetes a multiplexar. El sistema ISDB-T transmite cerca de 20000 paquetes por segundo, con lo cual, si el archivo generado es de cerca de 30 segundos se va a necesitar 600000 paquetes.

**b15040**: Tanto la PAT como la PMT, se quiere enviarla al menos 10 veces por segundo. Sabiendo que cada una de las tablas entra en un solo paquete de 188 bytes, tomando en cuenta que se tiene que enviar 10 paquetes por segundo y cada paquete es de 188 bytes \* 8 = 1504 bytes. Un análisis similar se necesita para la NIT y SDT.

**b 4600000, b:128000**: Son el ancho de banda del audio y del video respectivamente. Más detalle se puede encontrar en el manual de OpenCaster.

**b 27434198**: Es el ancho de banda de paquetes nulos. El sistema ISDB-T tiene un ancho de banda fijo de 29.958.294 bps, entonces calculamos el número de paquetes que se está utilizando con la siguiente expresión:

$$15040+15040+3008+3008+2300000+ 188000 = 4749056$$

Entonces para saber el número de paquetes que hacen falta para completar el Transport Stream que deberá tener el paquete null se hace la resta correspondiente:

$$29958294 - 2524096 = 25209238 \text{ bps.}$$

Finalmente ahora se tiene un archivo *pruebafinal.ts* que tiene el primer multiplexado. Sin embargo, hay un problema ya que el PCR del sistema, que es la referencia del reloj, trabaja en 27 MHz, y en este caso, viaja en el stream de video. Como se cambió la posición de los paquetes del video, la llegada de los mismos al Set-Top Box es imprecisa. Para arreglar esto se utiliza la herramienta “*tsstamp*”.

`$ tsstamp pruebafinal.ts 29958294 > pruebafinal.fixed.ts`

### Recopilación de los parámetros necesarios para la generación de un Transport Stream con un servicio de TV Digital

En la sección anterior se generó un archivo de transport stream llamado *“Transport Stream pruebafinal.fixed.ts”* que puede ser utilizado para ser transmitido en un servicio de televisión digital. En la tabla 3.1 se muestra un resumen de las características del transport stream generado en el ejemplo de la sección anterior.

**Tabla 3. 1 Configuración de los flujos elementales**

<b>FLUJOS ELEMENTALES DE VIDEO Y AUDIO</b>		
<b>Parámetro</b>	<b>Valor</b>	<b>Observaciones</b>
<b>Elementary Sream de Video</b>		
Formato de la señal original	.avi	
Formato del video de salida	m2v	
Códec de video	Mpeg2	
Tamaño de la pantalla	720 x 576	Formato del tamaño de presentación del video
Aspecto	4:3	Formato de la presentación en la pantalla
Bitrate	4000kbps	Tasa de bits del video original.
<b>Elementary Sream de Audio</b>		
Formato de la señal original	.avi	
Formato del audio de salida	.mp2	
Códec de audio	Mp2	
Bitrate	128kbps	
Frecuencia de muestreo	48kHz	
Número de canales	2	Define una señal estéreo

Tabla 3. 2 Configuración de los Flujos Elementales Paquetizados

FLUJOS PAQUETIZADOS DE VIDEO Y AUDIO		
Parámetro	Valor	Observaciones
<b>PES de Video</b>		
Formato de la señal entrada	.m2v	
Formato de la señal de salida	.pes	
<b>PES de Audio</b>		
Formato de la señal entrada	.mp2	
Formato de la señal de salida	.pes	
Muestras por frame	1152	
Frecuencia de muestreo	48kHz	Debe ser la misma que se definió en el ES.
Tamaño del frame de audio	384	
Desplazamiento del PTS	3600	Se define para poder sincronizar con la señal de video.

Tabla 3. 3 Configuración de los Flujos de Transporte

TRANSPORT STREAMS DE VIDEO Y AUDIO		
Parámetro	Valor	Observaciones
<b>TS de Video</b>		
Formato de la señal entrada	.pes	
Formato de la señal de salida	.ts	
PID	2065	Identificador del TS de la señal de video
Frames por segundo	25	

Tamaño de verificador del buffer de video	112kbps	
Ancho de banda	4800kbps	15% Mayor que el bitrate del video original.
<b>TS de Audio</b>		
Formato de la señal entrada	.pes	
Formato de la señal de salida	.ts	
PID	2075	Identificador del TS de la señal de audio
Muestras por frame	1152	Debe ser la misma que se definió en el PES.
Frecuencia de muestreo	48kHz	Debe ser la misma que se definió en el PES.
Tamaño del frame de audio	384	Debe ser el misma que se definió en el PES.

Tabla 3. 4 Configuración de las tablas de PSI/SI

<b>Tablas PSI /SI</b>	
<b>Parámetro</b>	<b>Valor</b>
Identificador de red	0x073
Identificador de red original	0x073
Frecuencia de transmisión	533MHz
Identificador de servicio de TV Digital	0x760
PID de la Tabla PMT	1031
Nombre de la Red	ESPE
Nombre del TS	ESPE
PID de la Tabla NIT	16
PID del PCR	2064

## CUARTO CAPÍTULO

### 4.1 CONCLUSIONES

1. La configuración del número de segmentos para cada capa jerárquica dependerá de las necesidades específicas del transmisor, el cual deberá escoger los parámetros que mejor le convengan para las tasas de transmisión deseadas para los distintos tipos de servicio, considerando que la tasa máxima para el canal de 6MHz es de 23234,64 kbps.
2. El formato de capas y paquetes necesarios para la formación del Transport Stream puede ser muy parecido al modelo OSI, donde cada capa agrega información de cabecera importante para que pueda interpretarse de manera adecuada en el siguiente nivel y de esta manera pueda finalmente generarse los flujos de transporte necesarios para la remultiplexación de las señales en un flujo único de transporte.
3. El Transport Stream constituye una parte fundamental dentro de la estructura de TV Digital ya que representa el nexo necesario entre la codificación de video, audio y aplicaciones interactivas con la transmisión y visualización de la información en el televisor.
4. El alcance que puede tener la configuración de Transport Stream puede llegar a ser muy extenso, ya que nos permite definir configuraciones muy avanzadas que explotan al máximo las bondades que ofrece la TV Digital. Estas configuraciones nos permitirían enviar transmisiones en definición estándar, alta definición, servicios one seg o full seg, generación de EPG, generación de closed caption, generación de contenidos interactivos, entre otros.
5. OpenCaster constituye una herramienta de distribución libre bastante útil para poder empezar a generar el Transport Stream para la transmisión



de varios canales de de TV Digital de manera simultánea para propósitos de investigación y desarrollo. Sin embargo su manejo puede resultar un poco complejo ya que al no tratarse de una interfaz gráfica, todos los procesos deben realizarse de manera manual lo que lo vuelve poco útil para propósitos comerciales.

6. Para poder generar el Transport Stream de manera adecuada es necesario tomar en cuenta los parámetros originales de las señales de audio y video. Un parámetro muy sensible es la tasa de bits (bitrate) de la señal de video, debido a que si no se realiza la conversión del video, mediante el comando *ffmpeg*, con la tasa de bits del video original se va a obtener una señal de salida con errores, como una reproducción más lenta y desincronizada de la señal de video respecto a la señal de audio.
7. Las Tablas de información específica constituyen una parte muy importante en la generación de los flujos de transporte ya que poseen información imprescindible para que el decodificador pueda interpretar de manera adecuada los diferentes flujos de transporte que se remultiplexan. Por medio de estas tablas se puede incluir más de un canal en un solo flujo de transporte, aplicaciones de interactividad, información sobre los canales de servicio utilizados, frecuencias de transmisión, entre otros.
8. En cuanto a la etapa de multiplexación de las señales, el estándar brasileiro presenta una estructura casi idéntica a la del estándar japonés. Las únicas diferencias son que el estándar brasileiro nos permite convertir formatos de video H.264 al formato .ts. Aunque esta diferencia aún no se la puede experimentar ya que las herramientas de software desarrolladas en el mercado aún no presentan esta funcionalidad.
9. Otra variación que existe entre el estándar japonés con el estándar brasileiro es la estructura de las tablas PAT, PMT, NIT y SDT (conocida como TSDT en el estándar ISDB-Tb), que aunque son igualmente utilizadas en el estándar japonés, la estructura de datos que lleva cada

una tiene variaciones cuando se quiere generar un Transport Stream que contenga aplicaciones de interactividad en lenguaje NCL.

## 4.2 RECOMENDACIONES

Si se desea realizar un análisis más profundo para el desarrollo de aplicaciones que permitan la generación de transport stream con una mayor cantidad de servicios integrados, es recomendable partir de las normas que define el estándar brasileiro, especialmente aquellas que hablan acerca de la Multiplexación, los servicios de información, las especificaciones de la transmisiones de datos y la codificación de la información, es decir, la ABNT NBR 15603, 15606 y 15602 respectivamente.

Se recomienda familiarizarse con el lenguaje *python* que sirve para escribir el script que nos genera las tablas de información específica, muy necesarias para formar el transport stream.

Si se desea desarrollar herramientas con fines comerciales para la generación de transport stream, es recomendable partir de las utilidades que nos presenta OpenCaster e incluir un entorno gráfico, más amigable para el usuario, donde la conversión de formatos y el cálculo de parámetros como el bitrate o anchos de banda se realicen de manera automática.

Actualmente no es recomendable generar transport stream con videos en formatos H.264 ya que aún el software no tiene todo el soporte para dar este servicio. Además que los parches utilizados que han desarrollado el laboratorio de LIFIA son compatibles con la versión 2.5 y aún no se trabaja de manera adecuada con la versión 3.0 de OpenCaster.

Para la instalación de OpenCaster se tuvieron algunos inconvenientes con encontrar las librerías de GCC y PYTHON en sus versiones más recientes por lo que se recomienda utilizar el sistema operativo UBUNTU 9.10 en adelante para no tener inconvenientes.

## ANEXOS

### A1. Estructura de datos de la PES sincronizada

De acuerdo con la especificación de transmisión PES sincronizada, los datos se transmiten utilizando un paquete PES especificado en la ISO/IEC 13818-1. Cualquier mapeo de paquete PES para un stream de transporte MPEG-2 debe cumplir la ISO/IEC 13818-1.

En caso de que sea un stream del tipo sincronizado, el campo *stream\_id* se debe configurar en '0xBD'(private stream\_1) y el campo *PES\_packet\_length* debe tener un valor diferente de cero.

La estructura de datos de la PES sincronizada mostrada en la siguiente tabla se debe insertar en el campo PES\_packet\_data\_bytes.

Sintáxis	# bits	Descripción
Data_identifier	8	Campo de 8 bits que se debe configurar en '0x80'
Private_stream_id	8	No utilizado (0xFF)
Pes_data_packet_header_length	4	Campo de 4 bits indica la extensión en bytes del PES <i>data private data bytes</i>
PES_data_private_data_byte	8	Es una utilización más detallada de este campo y depende de un servicio. Una unidad receptora puede omitir este campo
synchronized_PES_data_byte	8	Campo de 8 bits que contiene los datos transmitidos.

### A2. Estructura de datos de la PES asincrónica

De acuerdo con la especificación de PS asincrónica, los datos se transmiten utilizando un paquete PS especificada en la ISO/IEC 13818-1. Cualquier mapeo de paquete PS para un stream de transporte MPEG-2 debe cumplir la ISO/IEC 13818-1.

De acuerdo con la especificación de transmisión asíncrona, un paquete PES con las siguientes restricciones se utiliza además de la sintaxis y semántica especificadas en la ISO/IEC 13818-1.

En caso de que sea un stream del tipo sincronizado, el campo *stream\_id* se debe configurar en '0xBF' (private stream\_2) y el campo *PES\_packet\_length* debe tener un valor diferente de cero.

La estructura de datos de la PES sincronizada mostrada en la siguiente tabla se debe insertar en el campo PES\_packet\_data\_bytes.

Sintaxis	# bits	Descripción
Data_identifier	8	Campo de 8 bits que se debe configurar en '0x80'
Private_stream_id	8	No utilizado (0xFF)
Pes_data_packet_header_length	4	Campo de 4 bits indica la extensión en bytes del PES <i>data private data bytes</i>
PES_data_private_data_byte	8	Es una utilización más detallada de este campo y depende de un servicio. Una unidad receptora puede omitir este campo
synchronized_PES_data_byte	8	Campo de 8 bits que contiene los datos transmitidos.

### A3. Estructura de datos del mensaje de indicación de información de download (DII)

Un mensaje DII forma parte de un mensaje de control DSM-CC. Así, el mensaje DII transmite el contenido del mensaje reteniéndolo en el `userNetworkMessage()` en la sección DSM-CC.

La estructura de los datos del mensaje DII descrito en la sección se presenta en la siguiente tabla:

Sintaxis	No. de bits	Descripción
dsmccMessageHeader		Encabezamiento del mensaje DSM-CC
DownloadId	32	Campo que sirve como un rótulo para la

(identificador de download)		identificación única del carrusel de datos.
windowSize	8	Campo que no se utiliza para transmisión del carrusel de datos y el valor debe ser ajustado en 0.
ackPeriod	8	Campo que no se utiliza en la transmisión del carrusel de datos y el valor debe ser ajustado en 0.
tCDownloadWindow	32	Campo que no se utiliza en la transmisión del carrusel de datos y el valor debe ser ajustado en 0.
tCDownloadScenario	32	Campo que indica el período de límite de tiempo en el que se presume que el download está completo en microsegundos
compatibilityDescriptor		Estructura del descriptor de compatibilidad compatibilityDescriptor()) que se especifica en la ISO/IEC 13818-6 y que debe ser configurada en este campo.
numberOfModules	16	Campo que indica el número de módulos descritos en el enlace siguiente en este mensaje DII.
ModuleId	16	Campo que indica la identificación del módulo descrito en los siguientes campos: ModuleSize, module Version y moduleInfoByte.
ModuleSize	32	Campo que indica la extensión byte del módulo. Cuando la extensión del byte del módulo no es conocida, debe ser configurada en 0.
ModuleVersion	8	Campo que indica la versión de este módulo.
ModuleInfoLength	8	Campo que indica la extensión byte del área de información del módulo.
moduleInfoByte	8	Campo que se puede usar para insertar

		descriptores relacionados al módulo. Ver ABNT-NBR 15606-3.
PrivateDataLength	16	Campo que indica la extensión byte del campo PrivateDataByte.
privateDataByte	8	campo que se puede usar para contener una estructura de datos en un formato de descriptor.

#### A4. Estructura de datos del Mensaje DownloadDataBlock (DDB)

El contenido de un mensaje DDB se transmite por almacenamiento en el campo downloadDataMessage() en la sección DSM-CC. Un mensaje DDB es la estructura de datos transmitiendo bloques de datos. Un módulo se puede dividir con extensión fijada para formar bloques. En ese caso, cada bloque es representado con un número de bloque en el mensaje DDB para permitir que una unidad receptora reorganice los bloques en el orden pretendido. La estructura de datos del mensaje DDB se muestra en la siguiente tabla:

Sintaxis	No. de bits	Descripción
moduleId	16	Campo que indica el número de identificación al cual este bloque pertenece.
moduleVersion	8	Campo que indica la versión del módulo al cual este bloque pertenece.
blockNumber	16	Campo que indica la posición de este bloque dentro del módulo. El primer bloque de un módulo debe ser representado por el bloque número 0.
blockDataByte	8	El tamaño de una serie del área de datos del bloque es igual al tamaño del bloque de la DII, es decir, el tamaño de los bloques divididos desde un módulo

### A5. Estructura de la sección DSM-CC

Los mensajes DII y DDB se transmiten usando las secciones DSM-CC, la cual tiene una estructura de datos como se muestra en la siguiente tabla:

Sintaxis	No. de bits	Descripción
Table_id	8	Campo que contiene el número de identificación del tipo de datos en la sección de payload DSM-CC
section_syntax_indicator	1	Campo que cuando es configurado en 1, indica que existe un CRC32 al final de la sección. Cuando es configurado en ***0, indica que existe una suma de verificación. Se debe configurar en 1 para la transmisión de los mensajes DII y DDB.
private_indicator	1	Campo que almacena el valor complementario del flag del section_syntax_indicator;
dsmcc_section_length	12	Campo que indica el número de bytes del área desde el inicio del campo, inmediatamente después de ese campo hasta el fin de la sección. El valor en este campo no debe exceder 4093 bytes
table_id_extension	16	Campo que es configurado de acuerdo con el campo table_id como se indica en tabla de table_id_extension.
version_number	5	Campo que es configurado tal como se muestra abajo, de acuerdo con el identificador de tabla
Value		Cuando el valor del campo table_id es igual a 0x3B, este campo se debe configurar en 02. Cuando el

		valor del campo table_id es igual a 0x3C, se debe configurar en los 5 bits menos significativos del campo versión del módulo
current_next_indicator	1	Campo que indica que la subtabla está activa cuando está en “1”. Cuando está en “0”, la subtabla enviada aún no fue aplicada y usada como la próxima subtabla. Cuando el valor del campo table_id es igual a un valor en la banda de 0x3A a 0x3C, este campo se debe configurar en “1”
section_number	8	Campo que indica el número de la sección de la primera sección en la subtabla. Cuando la sección contiene un mensaje DII, este campo se debe configurar en 0. Cuando esta sección contiene un mensaje DDB, este campo debe transportar una copia de los 8 bits menos significantes del número del bloque de la DDB
last_section_number	8	Campo que indica el número de la última sección (sección que tiene el número máximo de la sección) de la subtabla a la cual pertenece la sección.
userNetworkMessage()		El mensaje DII es almacenado.
downloadDataMessage()		El mensaje DDB es almacenado.

En cuanto al campo table\_id basado en el valor de este campo, se aplica una regla de codificación específica para el campo siguiente en la sección DSM-CC. La tabla de los valores de identificación se muestra en la siguiente tabla.



<b>TABLE ID</b>	
<b>Table_id</b>	<b>Tipo de sección DSM-CC</b>
0x3A	Reservado
0x3B	Mensaje DII
0x3C	Mensaje DDB
0x3D	Descriptor de Stream
0x3E	Datos privados
0x3F	Reservados

<b>Table_id_extension</b>	
Table_id =0x3B	El campo table_id_extension debe transportar una copia de los 2 bytes menos significativos del campo transaction_id.
Table_id =0x3C	El campo table_id_extension debe transportar una copia del campo module_id

#### **A6. Estructura de datos del Encapsulado multiprotocolo**

Los datagramas son encapsulados en las datagram\_sections que son compatibles con el formato DSMCC\_section para datos privados. El mapeo de la sección dentro de los paquetes MPEG-2 de stream de transporte se define en sistemas MPEG-2.

La sintaxis y la semántica del datagram\_section se definen en la siguiente tabla:

<b>Sintaxis</b>	<b>Descripción</b>
table_id	Campo de 8 bits que se debe configurar en 0x3E, secciones DSM-CC con datos privados (ver la ISO/IEC 13818-6:1998, Sección 5)
section_syntax_indicator	Campo que se debe configurar conforme definido en la ISO/IEC 13818-6:1998, sección 5
private_indicator	Campo que se debe configurar conforme definido en la ISO/IEC 13818-6:1998, Sección 5.

Reserved	Campo de 2 bits que se debe configurar en “11”
section_length	Campo que se debe configurar conforme definido en la ISO/IEC 13818-6:1998, sección 5.
MAC_address_[1.. 6]	Campo de 48 bits que contiene la dirección MAC del destino. La dirección MAC es fragmentada en 6 campos de 8 bits, rotulados como MAC_address_1 a MAC_address_6.
payload_scrambling_control	Campo de 2 bits que define el modo de codificación del payload de la sección de acuerdo a la tabla del payload_scrambling_control
address_scrambling_control	Campo de 2 bits que define el modo de dispersión de la dirección MAC en esta subsección. (Ver tabla de <i>address scrambling control</i> ). Este campo permite un cambio dinámico de las direcciones MAC.
LLC_SNAP_flag	Flag de 1 bit. Si el flag está configurado en “1”, la payload carga un datagrama siguiendo campo MAC_address_1. Si el flag está configurado en “0”, la sección debe contener un datagrama IP sin encapsulado LLC/SNAPP.
current_next_indicator	Campo de 1 bit que se debe configurar en el valor de “1”.
section_number	Campo de 8 bits. Si el datagrama es transportado en secciones múltiples, entonces este campo indica la posición de la sección dentro del proceso de fragmentación. En caso contrario será configurado en cero
last_section_number	Campo de 8 bits que debe indicar el número de la última sección usada para cargar el datagrama, es decir, el número de la última sección del proceso de fragmentación.
LLC_SNAP	Estructura que debe contener el datagrama de acuerdo con las especificaciones de la ISO/IEC

	8802-2 LLC (Control de Conexión Lógica) y de la ISO/IEC TR 8802-1 SNAP (Punto de Anexión de la Subredes).
IP_datagram_data_byte	Bytes que contienen los datos del datagrama. Si la payload de la sección está codificada, estos bytes deben estar codificados.
stuffing_byte	Campo opcional de 8 bits cuyo valor no se especifica. El número de stuffing_bytes que se utilizan debe adecuarse a las exigencias de alineamiento de los datos definidos en el data_broadcast_descriptor
checksum	Campo que se debe configurar conforme definido en la ISO/IEC 13818-6:1998, Sección 5. Es calculado sobre el datagram_section completo
CRC_32	Campo que se debe configurar conforme definido en la ISO/IEC 13818-6:1998, Sección 5. Es calculado sobre el datagram_section completo.

<b>Codificación del campo payload_scrambling_control</b>	
<b>Valor</b>	<b>Control de codificación del payload</b>
00	No codificado
01	Definido por el servicio
10	Definido por el servicio
11	Definido por el servicio

<b>Codificación del campo address_scrambling_control</b>	
<b>Valor</b>	<b>Control de codificación del payload</b>
00	No codificado
01	Definido por el servicio
10	Definido por el servicio
11	Definido por el servicio

## A7. Tipos de elementos definidos en el campo *Stream type* en la tabla PMT

El stream type es un parámetro definido en la estructura de la sección de una tabla PMT que identifica el tipo de elemento del programa de radiodifusión de acuerdo con la siguiente tabla:

<b>STREAM TYPE</b>	
<b>Valor</b>	<b>Descripción</b>
0x00	Reservado
0x01	Video conforme ISO/IEC 11172-2
0x02	Video conforme ITU Recommendation H.262
0x03	Audio conforme ISO/IEC 11172-3
0x04	Audio conforme ISO/IEC 13818-3
0x05	Sección
0x06	Paquete PES
0x07	MHEG conforme ISO/IEC 13522-1
0x08	Conforme ITU Recommendation H.222.0:2006, Anexo DSM-CC
0x09	Conforme ITU Recommendation H.222.1
0x0A	Conforme tipo A descrito en la ISO/IEC 13818-6
0x0B	Conforme tipo B descrita en la ISO/IEC 13818-6
0x0C	Conforme tipo C descrito en la ISO/IEC 13818-6
0x0D	Conforme tipo D descrito en la ISO/IEC 13818-6
0x0E	Datos auxiliares conforme ITU Recommendation H222.0
0x0F	Audio con sintaxis de transporte ADTS conforme ISO/IEC 13818-7
0x10	Video conforme ISO/IEC 14496-2
0x11	Audio conforme ISO/IEC 14496-3
0x12	Flujo de paquetes SL o flujo FlexMux transportada en los paquetes de PES conforme ISO/IEC 14496-1
0x13	Flujo de paquetes SL o flujo FlexMux transportada en los paquetes de PES conforme ISO/IEC 14496-1
0x14	Protocolo de sincronización de download conforme ISO/IEC

	13818-6
0x15	Metadatos transportados por un paquete PES
0x16	Metadatos transportados por una metadata_sections
0x17	Metadatos transportados por el carrusel de datos de conformidad con la ISO/IEC 13818-6
0x18	Metadatos transportados por el carrusel de objetos de conformidad con la ISO/IEC 13818-6
0x19	Metadatos transportados por un protocolo de download sincronizado ISO/IEC 13818-6
0x1A	IPMP stream de conformidad con la ISO/IEC 13818-11
0x1B	Video de acuerdo con la ITU Recommendation H.264 y ISO/IEC 14496-10
0x1C - 0x7D	Reservado
0x7E	Data pipe
0x7F	IPMP stream
0x80 - 0xff	Uso privado

## BIBLIOGRAFÍA

### CAPÍTULO I

- [1] *Historia de la Televisión*, Asociación Plaza del Castillo.  
<http://www.asociacionplazadelcastillo.org/Textosweb/HistoriadelaTelevision.pdf>.
- [2] Guillén, Esperanza, “Estudio y propuesta de la factibilidad técnica, social y económica del sistema SBTVD-T”, EPN, Ecuador, 2007, Capítulo 1.
- [3] Campos Paulo, “*Estudio del estándar de Televisión Digital Terrestre DTMB y propuesta de reglamento para la prestación del servicio de Televisión Digital Terrestre en el Ecuador*”, EPN, Ecuador, 2010, Capítulo 2.
- [4] *Informe para la definición e implementación de la Televisión Digital Terrestre en el Ecuador*, Ecuador, SUPERTEL, 2010, Pág. 21.
- [5] *Informe para la definición e implementación de la Televisión Digital Terrestre en el Ecuador*, Ecuador, SUPERTEL, 2010, Pág. 22.
- [6] *Informe para la definición e implementación de la Televisión Digital Terrestre en el Ecuador*, Ecuador, SUPERTEL, 2010, Pág. 39.
- [7] *Informe para la definición e implementación de la Televisión Digital Terrestre en el Ecuador*, Ecuador, SUPERTEL, 2010, Pág. 47 a 49.
- [8] *Revista Institucional*, SUPERTEL, Edición No. 4, Febrero 2009.
- [9] *ISDB-T Reporte Técnico, ANEXO-AA. Contenidos técnicos y estructura del sistema ISDB-T*, DIBEG, Japón, 2007.
- [10] Calero, Antonio; Villacrés, Carlos, “*Análisis y estudio de ingeniería para la selección del estándar de Televisión Digital más apropiado para Ecuador bajo la supervisión de la SUPERTEL*”, ESPOCH, Ecuador, 2009.
- [11] Morales, Amparito, “*Diseño de la red para interactividad en Televisión Digital Terrestre e IPTV en el campus ESPE Sangolquí*”, Espe, Ecuador, 2010.

[12] Pisciotta, Néstor Oscar, “*Sistema ISDB-Tb*”, Primera parte, Universidad Tecnológica Nacional ,2010.

[13] Sotomayor, Patricio, “*Análisis de los estándares de Televisión Digital Terrestre (TDT) y pruebas de campo utilizando los equipos de comprobación técnica de la superintendencia de telecomunicaciones*”, EPN, Ecuador, 2009.

[14] Torres, Javier “Diseño y desarrollo de una aplicación de contenidos interactivos para tv digital basada en el Middleware Ginga del sistema brasileño”, ESPE, Ecuador, 2010.

## **CAPÍTULO II**

[1] ISDB-T Reporte Técnico, ANEXO-AA. Contenidos técnicos y estructura del sistema ISDB-T, DIBEG, Japón, 2007.

[2] Helio Coelho Junior, “Sistema de Transmissão no Padrão Brasileiro de TV Digital”, Departamento de Ingeniería de Telecomunicaciones, Universidad Federal Fluminense.

[3] Fischer, Walter. Digital Video and Audio Broadcasting Technology: A practical engineering guide (Signals and Communication Technology), Second Edition, Springer, Alemania. Enero 2008, pp. 34-38.

[4] “MPEG-2 Transport vs. Program Stream”, VBrick, 2008.

[5] “MPEG-2, Reference Guide to Digital Video Technology, Testing and Monitoring”, JDSU, 2008.

[6] Morales, Amparito, “Diseño de la red para interactividad en Televisión Digital Terrestre e IPTV en el campus ESPE Sangolquí”, Espe, Ecuador, 2010.

[7] ABNT NBR 15602-1 (Norma Brasileña), Televisión digital terrestre — Codificación de video, audio y multiplexación, Parte 1: Codificación de video, 2007.

- [8] H.264 / MPEG-4 Part 10 White Paper, VODEX.  
[http://read.pudn.com/downloads67/ebook/241367/h.264%E7%99%BD%E7%9A%AE%E4%B9%A6/h264\\_SP.pdf](http://read.pudn.com/downloads67/ebook/241367/h.264%E7%99%BD%E7%9A%AE%E4%B9%A6/h264_SP.pdf).
- [9] Sotomayor, Patricio, "Análisis de los estándares de Televisión Digital Terrestre (TDT) y pruebas de campo utilizando los equipos de comprobación técnica de la superintendencia de telecomunicaciones", EPN, Ecuador, 2009.
- [10] Guillén, Esperanza, "Estudio y propuesta de la factibilidad técnica, social y económica del sistema SBTVD-T", EPN, Ecuador, 2007.
- [11] ABNT NBR 15602-2 (Norma Brasileña), Televisión digital terrestre — Codificación de video, audio y multiplexación, Parte 1: Codificación de audio, 2007
- [12] Gomes Soares, Luiz Fernando. Y Junqueira Barbosa, Simone Diniz. TV DIGITAL INTERATIVA NO BRASIL SE FAZ COM GINGA: Fundamentos, Padrões, Autoria Declarativa e Usabilidade, PUC-Rio, 2008.
- [13] ABNT NBR 15606-3 (Norma Brasileña), Televisión digital terrestre — Codificación de datos y especificaciones de transmisión para radiodifusión digital Parte 3: Especificación de transmisión de datos, 2007.
- [14] ABNT NBR 15602-3 (Norma Brasileña), Televisión digital terrestre — Codificación de video, audio y multiplexación, Parte 3: Sistemas de multiplexación de señales, 2007.
- [15] ABNT NBR 15603-1 (Norma Brasileña), Televisión digital terrestre — Multiplexación y servicios de información (SI), Parte 1, 2007.
- [16] H222.0 (Estándar ITU-T), Information technology – Generic coding of moving pictures and associated audio information: Systems, 2006.



### CAPÍTULO III

- [1] OpenCaster 2.4: The free digital tv software,  
<http://www.avalpa.com/the-key-values/15-free-software/33-opencaster>
- [2] *OpenCaster para SATVD*, LIFIA, 2010.  
<http://wiki.ginga.org.ar/lib/exe/fetch.php?media=lifia:guiaopencaster2.pdf>
- [3] *Avalpa Broadcast Server User Manual*, Versión 2.0, 2011.
- [4] ABNT NBR 15603-2 (Norma Brasileña), *Televisión digital terrestre - Multiplexación y servicios de información (SI), Parte 2: Estructura de datos y definiciones de la información básica de SI*, 2007.