



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERIA

**Prácticas de Laboratorio de
Televisión Digital Terrestre
basadas en Radio Definido por
Software**

TESIS

PARA OBTENER EL GRADO DE:

INGENIERO EN TELECOMUNICACIONES

PRESENTA:

ÁNGEL DANIEL JUAN TORAL

DIRECCIÓN DE TESIS

DRA. FATIMA MOUMTADI



MÉXICO, D.F.

2013



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradezco antes que a nadie a mis padres por el apoyo y cariño que siempre me han brindado sin esperar nada a cambio. Además de siempre ser un excelente ejemplo a seguir. Sin todo esto no habría podido llegar a ser quien soy.

Por otra parte también agradezco a la UNAM y a los profesores que me ayudaron en mi desarrollo académico durante mi carrera en la Facultad de Ingeniería. En especial a mi tutora, la Dra. Fatima Moumtadi por su ayuda en la elaboración de esta tesis.

También agradezco al Dr. Ángel Lambert y al Dr. Carlos Delgado por su apoyo por facilitar el equipo en el que se basa este proyecto. Y a la Ing. Aniris Gisela Fonseca Hernández por su apoyo para la realización de la tesis.

“Por mi raza hablará mi espíritu”

ÍNDICE

Índice.....	i
Índice de figuras.....	v
Introducción.....	xi
Objetivo.....	xii
Planteamiento del problema.....	xiii
Método.....	xiv
Descripción de los capítulos.....	xv
1. Conceptos Generales de Televisión Digital.....	1
1.1 Proceso Exploración de la imagen.....	1
1.1.1 Tasa de imágenes.....	3
1.1.2 Resolución de la imagen.....	4
1.2 Sistemas Analógicos.....	4
1.2.1 NTSC.....	5
1.2.2 PAL.....	7
1.2.3 SECAM.....	9
1.2.4 Frecuencia de muestreo.....	10
1.3 Transmisor y Receptor de Televisión Digital.....	12
1.3.1 Ventajas de las Señales digitales.....	12

1.3.2	Sistemas de Televisión Digital	14
1.3.3	Diagrama de bloques del transmisor de Televisión Digital	14
1.3.4	Diagrama de bloques del receptor de Televisión Digital.....	18
2.	Codificación de Fuente	19
2.1	Codificación del Video	19
2.1.1	Herramientas para la compresión de vídeo.....	19
2.1.2	Estándares de Codificación de Video.....	24
2.3	Codificación de Audio.....	30
2.3.1	Principios de la Codificación de las señales de audio	30
2.3.2	Estándares de Compresión de Audio.....	34
3.	Codificación de Canal.	41
3.1	Corrección de Errores hacia Adelante	41
3.1.1	Codificación por bloques lineales	41
3.1.2	Códigos de Hamming	43
3.1.3	Códigos cíclicos	44
3.1.4	Códigos BCH	45
3.1.5	Códigos Reed-Solomon	46
3.1.6	Codificación Convolucional	47
3.1.7	Codificador de Trellis	51
3.1.8	interleaving	51
3.1.9	Puncturing.....	52
3.1.10	Códigos Concatenados.....	52

3.2 Modulaciones Digitales.....	53
3.2.1 Modulaciones de Amplitud.....	53
3.2.2 Modulaciones de Fase y Frecuencia.....	55
3.2.3 Modulaciones Multiportadora.....	56
4. Radio definido por Software.....	61
4.1 Comienzo de los radios definidos por Software.....	62
4.2 GNU Radio.....	63
4.2.1 Instalación de GNU Radio.....	64
4.2.2 Herramientas virtuales par alas simulaciones en GNU Radio.....	68
4.2.3 Modulación en amplitud mediante GNU Radio.....	72
4.2.4 Simulación del canal de comunicaciones mediante GNU Radio.....	74
4.2.5 Modulaciones digitales en GNU Radio.....	76
4.2.6 Codificación contra errores en GNU Radio.....	80
5. Prácticas.....	85
5.1 Introducción a GNU Radio.....	85
5.2 Simulación del Canal de comunicaciones con GNU Radio.....	87
5.3 Modulación en Frecuencia con GNU Radio.....	90
5.4 Modulaciones Digitales en Amplitud y en cuadratura.....	92
5.5 Modulaciones Digitales en Fase.....	98
5.6 Multiplexación por División de Frecuencias Ortogonales.....	100
5.7 Códigos lineales de bloques para protección contra errores.....	103
5.8 Trellis.....	104

5.9 Interleaving y Códigos Concatenados	105
5.10 Espectro de la señal codificada con NTSC y con ATSC	108
5.11 Resistencia a Errores del Video Codificado con H.264.....	109
Conclusiones	111
Bibliografía.....	112
Anexos	114
Anexo 1. Glosario.....	114

ÍNDICE DE FIGURAS

<i>Figura 1. 1 Señal Generada por una línea del proceso de exploración</i>	3
<i>Figura 1. 2 Estructura de un Televisión de video con la Codificación NTSC [1]</i>	6
<i>Figura 1. 3 Estructura de un canal de televisión con la codificación PAL [1]</i>	8
<i>Figura 1. 4 Estructura de un canal de televisión con la codificación SECAM [1]</i>	9
<i>Figura 1. 5 Estructura de la señal de video digitalizado</i>	12
<i>Figura 1. 6 Diagrama a bloques del transmisor de televisión digital terrestre</i>	15
<i>Figura 1. 7 Diagrama a bloques del transmisor de ATSC</i>	16
<i>Figura 1. 8 Diagrama a bloques del transmisor de DVB-T</i>	17
<i>Figura 1. 9 Diagrama a bloques del transmisor de ISDB-T</i>	17
<i>Figura 1. 10 Diagrama a bloques del receptor de televisión digital terrestre</i>	18
<i>Figura 2. 1 Exploración en Zig-Zag(A) y Alternada (b) de coeficientes DCT</i>	21
<i>Figura 2. 2 Diagrama a bloques de la codificación DPCM</i>	22
<i>Figura 2. 3 Capa de Secuencia de video</i>	26
<i>Figura 2. 4 Capa gop</i>	26
<i>Figura 2. 5 Capa imagen</i>	27
<i>Figura 2. 6 Capa de rebanada</i>	27
<i>Figura 2. 7 Capa de macrobloque</i>	28
<i>Figura 2. 8 Capa bloque</i>	28
<i>Figura 2. 9 Enmascaramiento frecuencial</i>	31
<i>Figura 2. 10 Enmascaramiento temporal</i>	32
<i>Figura 2. 11 Codificación de sub-bandas</i>	33
<i>Figura 2. 12 Decodificación de sub-bandas</i>	33

<i>Figura 2. 13 Trama AC-3.....</i>	<i>35</i>
<i>Figura 2. 14 BLOQUE DE AUDIO AC-3.....</i>	<i>35</i>
<i>Figura 2. 15 diagrama de bloques del codificador AC-3.....</i>	<i>36</i>
<i>Figura 2. 16 Trama de audio MPEG-2 capa 2.....</i>	<i>39</i>
<i>Figura 2. 17 Codificador de audio MPEG-2 Capa 2.....</i>	<i>39</i>
<i>Figura 3. 1 diagrama del sistema del codificador convolucional.....</i>	<i>49</i>
<i>Figura 3. 2 diagrama de estados.....</i>	<i>49</i>
<i>Figura 3. 3 diagrama de Árbol.....</i>	<i>50</i>
<i>Figura 3. 4 Proceso de Interleaving o Barajado.....</i>	<i>51</i>
<i>Figura 3. 5 Modulación PAM y 4-PAM.....</i>	<i>54</i>
<i>Figura 3. 6 Modulador FDM.....</i>	<i>56</i>
<i>Figura 3. 7 Gráfica de funciones base para N=16 y T=1.....</i>	<i>58</i>
<i>Figura 3. 8 Diagrama a bloques de una transmisión OFDM.....</i>	<i>58</i>
<i>Figura 4. 1 diagrama a bloques de un radio definido por software.....</i>	<i>61</i>
<i>Figura 4. 2 Bloque Fuente aleatoria y su configuración.....</i>	<i>68</i>
<i>Figura 4. 3 BLOQUE suma Y SU CONFIGURACIÓN.....</i>	<i>69</i>
<i>Figura 4. 4 Bloque Sumidero Osciloscopio y su configuración.....</i>	<i>69</i>
<i>Figura 4. 5 Bloque sumidero FFT y su configuración.....</i>	<i>70</i>
<i>Figura 4. 6 Bloque sumidero Waterfall y su configuración.....</i>	<i>71</i>
<i>Figura 4. 7 Bloque sumidero de Audio y su configuración.....</i>	<i>71</i>
<i>Figura 4. 8 Bloque Fuente de Archivo y su configuración.....</i>	<i>71</i>
<i>Figura 4. 9 Bloque Sumidero Archivo y su configuración.....</i>	<i>72</i>
<i>Figura 4. 10 Sumidero USRP y su Configuración.....</i>	<i>73</i>
<i>Figura 4. 11 Fuente USRP y Configuración.....</i>	<i>73</i>

<i>Figura 4. 12 SIMULACIÓN DEL CANAL DE COMUNICACIONES EN GNU RADIO COMPANION</i>	<i>74</i>
<i>Figura 4. 13 Fuente de Señal y su configuración en GNU Radio Companion.....</i>	<i>74</i>
<i>Figura 4. 14 Bloque FUENTE DE Ruido Y SU CONFIGURACIÓN EN GNU RADIO COMPANION.....</i>	<i>75</i>
<i>Figura 4. 15 Bloque Binary Symmetric Channel y su Configuración</i>	<i>76</i>
<i>Figura 4. 16 Bloque de Modelo de Canal y su configuración.....</i>	<i>76</i>
<i>Figura 4. 17 Modulación PSK en GNU Radio Companion.....</i>	<i>77</i>
<i>Figura 4. 18 Bloque Modulación PSK y configuración</i>	<i>77</i>
<i>Figura 4. 19 MODULACIÓN QAM EN GNU RADIO COMPANION</i>	<i>78</i>
<i>Figura 4. 20 Bloque Chunks to Symbols y su Configuración</i>	<i>78</i>
<i>Figura 4. 21 Bloque Decodificador de constelación y su configuración.....</i>	<i>79</i>
<i>Figura 4. 22 Bloque OFDM Mod y su configuración</i>	<i>79</i>
<i>Figura 4. 23 BLOQUE DE CODIFICADOR TRELIS Y SU CONFIGURACIÓN.....</i>	<i>80</i>
<i>Figura 4. 24 BLOQUE viterbi Y SU CONFIGURACIÓN</i>	<i>80</i>
<i>Figura 4. 25 Bloque métricas de trellis y su configuración</i>	<i>81</i>
<i>Figura 4. 26 BLOQUE DE CODIFICACIÓN REED-MULLER Y SU CONFIGURACIÓN</i>	<i>81</i>
<i>Figura 4. 27 BLOQUE DE DECODIFICACIÓN REED-MULLER Y SU CONFIGURACIÓN</i>	<i>81</i>
<i>Figura 4. 28 BLOQUE DE CODIFICACIÓN RMG Y SU CONFIGURACIÓN</i>	<i>82</i>
<i>Figura 4. 29 BLOQUE DE DecODIFICACIÓN RMG Y SU CONFIGURACIÓN</i>	<i>82</i>
<i>Figura 4. 30 Bloque interleaver y su configuración</i>	<i>82</i>
<i>Figura 4. 31 Bloque interleaver y su configuración</i>	<i>83</i>
<i>Figura 4. 32 Bloque puncture y su configuración</i>	<i>83</i>
<i>Figura 4. 33 Bloque depuncture y su configuración.</i>	<i>84</i>
<i>Figura 4. 34 Bloque contador BER y su configuración</i>	<i>84</i>
<i>Figura 4. 35 Bloque tasa de errores y su configuración.....</i>	<i>84</i>
<i>Figura 5. 1 Botones de simulación en GNU Radio</i>	<i>85</i>

<i>Figura 5. 2. Diagrama con bloques null en GNU Radio Companion</i>	<i>85</i>
<i>Figura 5. 3 Diagrama de análisis en frecuencia de las señales del generador de señales en GNU Radio Companion</i>	<i>86</i>
<i>Figura 5. 4. Diagrama de Suma de señales y sumideros de audio y osciloscopio en GNU Radio Companion ..</i>	<i>87</i>
<i>Figura 5. 5 Simulación del canal de comunicaciones mediante el bloque fuente de ruido en GNU Radio companion.....</i>	<i>88</i>
<i>Figura 5. 6 Simulación del canal de comunicaciones mediante el bloque modelo de canal en gnu radio companion.....</i>	<i>89</i>
<i>Figura 5. 7 Simulación de un Canal Simétrico Binario en GNU Radio Companion.....</i>	<i>89</i>
<i>Figura 5. 8 Diagrama de la simulación de una transmisión con la modulación FSK en GNU Radio Companion</i>	<i>90</i>
<i>Figura 5. 9 Espectro de una señal con modulación FSK.....</i>	<i>91</i>
<i>Figura 5. 10 Señal con modulación FSK en el osciloscopio</i>	<i>91</i>
<i>Figura 5. 11 Simulación de la modulación 4pam en gnu radio companion.....</i>	<i>93</i>
<i>Figura 5. 12 Constelación de una señal modulada con PAM.....</i>	<i>94</i>
<i>Figura 5. 13 Espectro de una señal modulada mediante PAM.....</i>	<i>94</i>
<i>Figura 5. 14 Constelación de una señal modulada con 4PAM.....</i>	<i>95</i>
<i>Figura 5. 15 Espectro de una señal modulada mediante 4PAM.....</i>	<i>95</i>
<i>Figura 5. 16 Constelación de una señal modulada con 8PAM.....</i>	<i>96</i>
<i>Figura 5. 17 Espectro de una señal modulada mediante 8PAM.....</i>	<i>96</i>
<i>Figura 5. 18 Simulación de la modulación 16-QAM en gnu radio companion</i>	<i>97</i>
<i>Figura 5. 19 Constelación de una señal modulada CON 16-QAM</i>	<i>97</i>
<i>Figura 5. 20 Espectro de una señal modulada mediante 16-QAM.....</i>	<i>98</i>
<i>Figura 5. 21 Constelación de modulación 8psk</i>	<i>99</i>
<i>Figura 5. 22 Constelación de modulación QPSK</i>	<i>99</i>
<i>Figura 5. 23 Resultados BER de la simulación BPSK</i>	<i>99</i>
<i>Figura 5. 24 Resultados BER de la simulación QPSK.....</i>	<i>100</i>

<i>Figura 5. 25 Resultados BER de la simulación 8PSK.....</i>	<i>100</i>
<i>Figura 5. 26 Espectro de la modulación OFDM en la simulación.....</i>	<i>101</i>
<i>Figura 5. 27 Modulación ofdm en la simulación.....</i>	<i>101</i>
<i>Figura 5. 28 Representación waterfall de la modulación ofdm en la simulación</i>	<i>102</i>
<i>Figura 5. 29 Simulación de la modulación OFDM en GNU Radio Companion</i>	<i>102</i>
<i>Figura 5. 30 Simulación de transmisión sobre un canal simetrico binario donde se aplica un código Reed- muller en GNU Radio Companion.....</i>	<i>103</i>
<i>Figura 5. 31 BER Resultado de la simulación de una transmisión sobre un canal simetrico binario donde se aplica un código reed-muller.....</i>	<i>103</i>
<i>Figura 5. 32 Simulación de la transmisión una señal codificada mediante un Trellis en GNU Radio companion</i>	<i>104</i>
<i>Figura 5. 33 Resultados ber de la simulación trellis qpsk</i>	<i>105</i>
<i>Figura 5. 34 simulación de la transmisión una señal codificada mediante Dos códigos concatenados en GNU Radio companion.....</i>	<i>106</i>
<i>Figura 5. 35 Resultado de la simulación de la transmisión una señal codificada mediante dos códigos concatendados</i>	<i>106</i>
<i>Figura 5. 36 Simulación de la transmisión una señal codificada mediante dos códigos concatenados mediante un interleaver en gnu radio companion</i>	<i>107</i>
<i>Figura 5. 37 Resultado de la simulación de la transmisión una señal codificada mediante dos códigos concatendados mediante un interleaver.....</i>	<i>108</i>
<i>Figura 5. 38 Espectro de señal codificada con NTSC.....</i>	<i>109</i>
<i>Figura 5. 39 Espectro de señal codificada con ATSC.....</i>	<i>109</i>
<i>Figura 5. 40 Diagrama de la simulación de la transmisión de un archivo de video sobre un canal simétrico binario</i>	<i>110</i>
<i>Figura 5. 41 Resultados de la simulación de la transmisión de un archivo de video sobre un canal simétrico binario con distintos BER</i>	<i>110</i>

INTRODUCCIÓN

En los últimos años la tendencia a las transmisiones digitales de información se ha acelerado, en algunos países se realizó el apagón analógico (se dejó de transmitir las señales de televisión con estándares analógicos), por ejemplo en la Unión Europea la fecha límite fue el año 2012 y en México está planeado para finalizarse en 2015. Por lo tanto es el tiempo adecuado para analizar, enseñar las tecnologías necesarias para implementar la televisión digital terrestre (TDT), además de las ventajas y desventajas que las transmisiones de televisión terrestre conllevan.

Al mismo tiempo que la tendencia a las transmisiones digitales se acelera se desarrollaron los radios definidos por software (SDR), un ejemplo son los dispositivos móviles que implementan diversos protocolos de comunicaciones, entre ellos UMTS, GSM, WiFi, Bluetooth etc. Además de ser muy utilizado en la industria también han surgido proyectos de software libres para implementar radios definidos por software como es el caso de GNU Radio y el software USRP. GNU Radio puede funcionar también sin el uso del USRP en forma de simulaciones o a través de la tarjeta de sonido de la computadora.

Por otra parte es importante ver la diferencia entre los diferentes estándares de televisión digital, pero hacer énfasis en el estándar que México ha adoptado, es decir, el estándar ATSC.

OBJETIVO

Desarrollar un conjunto de prácticas para el área de Radiocomunicaciones en general y particularmente la materia “Televisión Digital” con fin que el estudiante comprenda y analice las diferentes etapas de codificación, modulación, transmisión y recepción de la Televisión Digital Terrestre (TDT) mediante Radio Definido por Software (SDR).

PLANTEAMIENTO DEL PROBLEMA

La tendencia de la industria de las telecomunicaciones al desarrollo de sistemas digitales implementados mediante software y Procesadores de Señales Digitales (DSP) conlleva la necesidad de los ingenieros en telecomunicaciones de entender el funcionamiento de dicha tecnología y también de conocer la forma de implementarla.

Los estudios de ingeniería generalmente implican la necesidad de un balance entre la práctica y la teoría, de manera que el ingeniero no se convierta en un técnico que puede manejar el equipo pero no conoce a profundidad su funcionamiento, sino que el ingeniero conozca la forma de implementarlo y que al conocer su funcionamiento profundamente tenga la posibilidad de mejorarlo.

En el posgrado en Ingeniería Eléctrica: Telecomunicaciones se imparte la teoría sobre la Televisión Digital, sin embargo, no se cuenta con un laboratorio donde se prueben los conocimientos obtenidos en la clase de teoría y tampoco se enseña la forma de implementar transmisores, ni receptores de televisión digital.

El propósito del proyecto de tesis es diseñar un conjunto de prácticas con las que los alumnos de posgrado en Ingeniería Eléctrica: Telecomunicaciones complementen su conocimiento sobre las tecnologías de transmisión digital de televisión y conocer una forma de simular las tecnologías estudiadas en la clase de teoría mediante la plataforma de vanguardia GNU Radio. Al mismo tiempo se introduce a los alumnos a la rama de los radios definidos por software.

MÉTODO

El trabajo de esta tesis se divide en tres etapas:

Primera etapa: Se realiza una investigación sobre el funcionamiento de los diferentes sistemas de televisión digital terrestre que existen en la actualidad y del software GNU Radio.

Segunda etapa: Consiste en seleccionar y diseñar las prácticas de TDT basadas en SDR de manera que cubran las tecnologías que se utilizan en las transmisiones de Televisión Digital Terrestre.

Tercer Etapa: Comprobar y documentar el funcionamiento correcto de los experimentos que componen las practicas mediante la puesta en marcha de las simulaciones en el software GNU Radio 3.6.0.

DESCRIPCIÓN DE LOS CAPÍTULOS

En el Capítulo 1 se describen algunos conceptos generales sobre la televisión, como es el proceso de exploración de la imagen, además de contener un resumen de los estándares de televisión analógicos y de los actuales estándares de televisión digital terrestre.

El Capítulo 2 contiene un resumen de la investigación realizada sobre la codificación de las fuentes de audio y video de la televisión digital, es decir su codificación de fuente. Este capítulo se divide en dos secciones una para el audio y otra para el video. A su vez cada sección se divide en una parte que describe las técnicas para comprimir las señales y otra sección que describe los estándares de codificación que se utilizan en los sistemas de televisión digital terrestre.

En el Capítulo 3 se describen las técnicas de codificación de canal, es decir, los esquemas de corrección de errores y de modulación digital básicos y los que se utilizan en los sistemas de televisión digital terrestre.

El Capítulo 4 contiene un resumen de la investigación realizada sobre GNU Radio y su funcionamiento. Primero describe un breve resumen de la historia de los radios definidos por software, después describe el proceso de instalación de GNU Radio y en las otras secciones se describen varios bloques de procesamiento de señales utilizados en GNU Radio.

Por último en el Capítulo 5 se describen las prácticas diseñadas y se ilustran los resultados que el estudiante debe obtener en el desarrollo de las mismas.

1. CONCEPTOS GENERALES DE TELEVISIÓN DIGITAL

Antes de poder hablar de la televisión digital tenemos que hablar de los sistemas analógicos predecesores, ya que para que exista una señal digital antes debe haber una señal analógica. Además la señal analógica es una forma de transmitir una secuencia de imágenes pero la forma de convertir estas imágenes en una señal eléctrica es conocida como el proceso de exploración de la imagen, estas señales registran simplemente el nivel de color o de brillo de la imagen y basándose en la teoría del color se puede construir cualquier color a partir de los tres colores básicos. Este proceso se explica enseguida.

1.1 PROCESO EXPLORACIÓN DE LA IMAGEN

El proceso de exploración define la forma en que una cámara de video captura una escena de televisión, es decir, una secuencia de imágenes, para obtener una señal equivalente a sus valores de luminancia (intensidad del brillo en la imagen) y color, u otras señales equivalentes. De este proceso depende el número de líneas por cuadro y el número de cuadros por segundo.

En un proceso estándar de escaneo se siguen los siguientes pasos:

1. Se va capturando la imagen formando líneas paralelas horizontales de recorrido oblicuo, las líneas que se enfocan se traducen en señales eléctricas analógicas, de acuerdo a la intensidad del brillo o al grado de color de la imagen.
2. Cada que llega al fin de una línea regresa al otro extremo de la imagen donde comienza la siguiente línea, pero no registra ningún valor de luminancia o de intensidad de color, sino que registra un nivel de borrado, que se encuentra fuera del intervalo de las señales de luminancia, y agrega una señal de sincronización para que cuando se reciba la señal en el sistema que reproduce el video se salte a la siguiente línea. A este intervalo se le llama borrado horizontal o intervalo ciego horizontal. La duración de línea se considera como la suma del período de línea activa más el período de borrado horizontal.
3. Estas líneas se capturan hasta llegar a la parte más baja de la imagen. Cuando el proceso de exploración es progresivo este proceso termina en la inferior izquierda de la imagen, se salta a la parte donde inicio el proceso y se repite todo el proceso, al tiempo en que se salta de la parte más baja de la imagen a la parte superior se le conoce como intervalo de supresión vertical y en él se inserta la señal de sincronización vertical. En el caso de un proceso de exploración de la imagen entrelazado se da en dos partes el barrido de toda la imagen y las líneas de cada una de las dos partes se entrelazan para formar la imagen.

Para el video en color el proceso de exploración tiene como resultado tres señales con las que se puede reconstruir la imagen. Estas señales se conocen como señales RGB, es decir la señal de la intensidad de color rojo, la de color azul y la de color verde, y a partir de estas señales se pueden obtener las señales Y C_R C_B , que son una señal de luminancia y dos de diferencia de color, mediante una matriz de transformación. La señal que se genera para una línea de video se puede ver gráficamente en la figura 1.1 [1] [2].

El sistema puede cambiar su proceso de exploración de acuerdo a la resolución y a la tasa de imágenes por segundo que busque transmitir de acuerdo a las instrucciones que reciba de la estación terrena. El aumento de la tasa de imágenes por segundo implica un aumento del ancho de banda de la señal de video a transmitir.

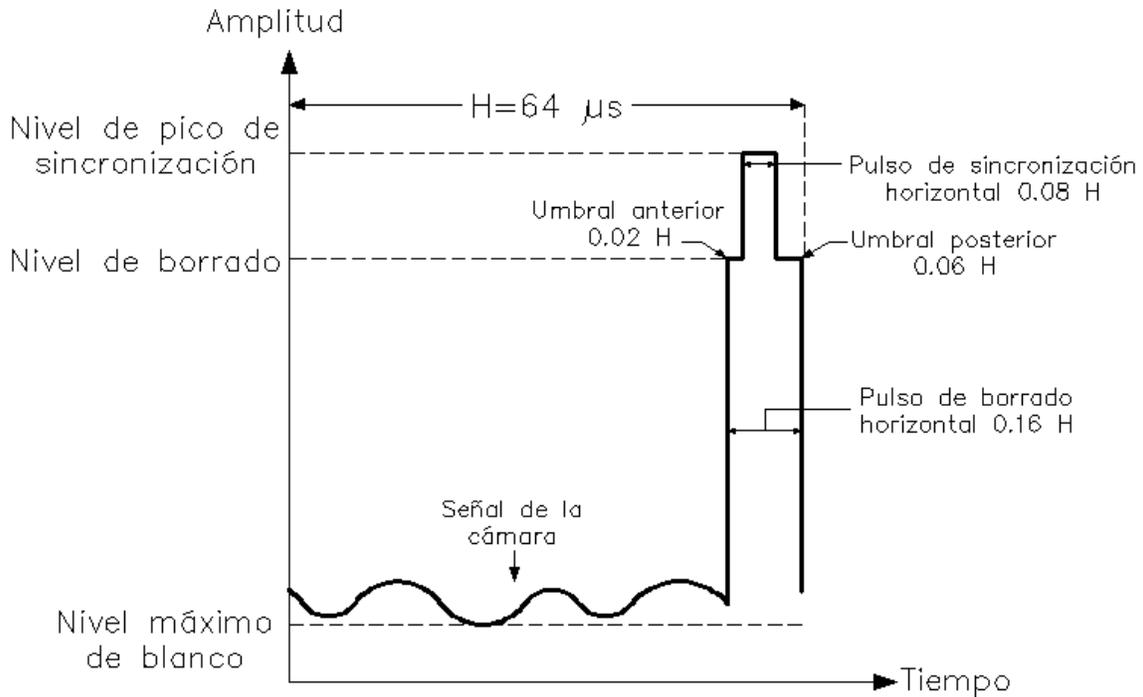


FIGURA 1. 1 SEÑAL GENERADA POR UNA LÍNEA DEL PROCESO DE EXPLORACIÓN

1.1.1 TASA DE IMÁGENES

La tasa de imágenes es el número de imágenes que la cámara de video captura por segundo. El número de imágenes por secuencia que se deben reproducir para que el ojo humano no perciba el parpadeo entre imágenes son 25 utilizando un proceso de exploración entrelazado. En las salas de cine se reproducen 24 imágenes por segundo pero el parpadeo entre imágenes no se percibe debido al ambiente oscuro, a este fenómeno se le conoce como enmascaramiento de luminancia.

1.1.2 RESOLUCIÓN DE LA IMAGEN

.La resolución de una imagen se toma en dos sentido, uno vertical que se refiere al número de líneas por la altura de la pantalla y uno horizontal que es el número de pixeles que se pueden mostrar en la pantalla horizontalmente y en los sistemas analógicos son las líneas verticales alternadas blancas y negras en tres cuartos del ancho de la pantalla.

La duración de una línea de televisión se relaciona directamente con la tasa de imágenes y la resolución vertical de la siguiente forma:

$$T_{línea} = \frac{\text{Líneas por campo}}{\text{Campos por segundo}} \quad (1)$$

En la ecuación (1) los campos por segundo son igual a la tasa de imágenes si el proceso de exploración es progresivo y es el doble cuando el proceso es entrelazado. Sin embargo, no toda la línea de televisión representa a la imagen sino que una parte se usa como intervalo de supresión o sincronización para que se salte del fin de una línea al inicio de la otra sin perder la sincronía. A la parte de la línea donde sí se envía información sobre la imagen se le llama línea activa [1].

1.2 Sistemas Analógicos

Los sistemas analógicos basaron su diseño en la agudeza visual, la persistencia de la visión y en la consideración de que la distancia del espectador al televisor sería 6 veces la altura de la pantalla. En las siguientes secciones se describen las características de los sistemas analógicos que fueron más utilizados y que aún se utilizan en algunos países que no han completado su migración de la televisión analógica a la televisión digital, proceso conocido como apagón analógico.

1.2.1 NTSC

El sistema analógico NTSC debe su nombre al Comité Nacional de Sistemas de Televisión (National Television Systems Committee) de Estados Unidos, que es el país donde surgió el estándar en 1940. Este comité fue establecido por la FCC (Federal Communications Comision) para resolver conflictos entre compañías de difusión de televisión. Este sistema también se implementó en gran parte continente americano menos en Brasil y Argentina.

Las características del sistema de exploración del sistema NTSC se describen en la tabla

1. Características del Sistema NTSC.

TABLA 1. CARACTERÍSTICAS DEL SISTEMA NTSC

Parámetro	Estándar 525/60
Líneas por Cuadro	525
Líneas por Campo	262.5
Tramas por segundo	29.97 ≈ 30
Campos por segundo [Hz]	59.94
Frecuencia de barrido horizontal [Hz]	15,734.25
Duración del borrado de campo [líneas]	20
Duración del borrado de trama [líneas]	40
Líneas activas por trama	485
Resolución vertical [LPH]	339
Duración de la línea [μs]	63.556
Duración del borrado o supresión horizontal [μs]	10.7 ± 0.1
Duración de la línea activa [μs]	52.856
Duración del ciclo del par de líneas [μs]	0.2338
Ancho de banda para resolución vertical y horizontal igual [MHz]	4.28
Factor de resolución horizontal [líneas/MHz]	79.20
Resolución horizontal [LPH]	333
Relación entre resolución vertical y horizontal H/V	0.98

En el sistema NTSC se utilizan canales de 6 MHz para transmitir cada canal de Televisión donde la señal de luminancia se modula con el esquema de banda lateral vestigial, de manera que la portadora de la señal de luminancia (que solo contiene la información de la imagen a blanco y negro) se transmite a 1.25 MHz del borde del canal de 6 MHz. Las señales de diferencia de color se modulan en cuadratura con una portadora de color 3.58 MHz mayor que la portadora de la señal de luminancia. Cada señal de diferencia de color tiene la mitad del ancho de banda de la señal de luminancia. Por último la señal de sonido que acompaña al video se modula en frecuencia con una portadora 4.5 MHz mayor que la portadora de la señal de luminancia [1].

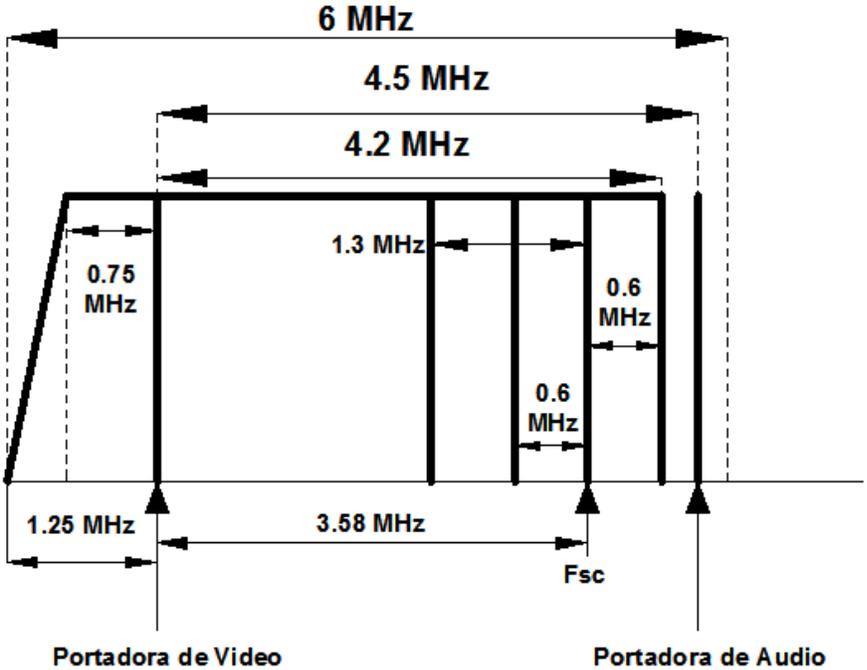


FIGURA 1. 2 ESTRUCTURA DE UN TELEVISIÓN DE VIDEO CON LA CODIFICACIÓN NTSC [1]

El sistema no ha permanecido estático desde su primera implementación hasta la fecha pero si se ha buscado que las innovaciones se mantengan compatibles con los sistemas anteriores. Uno de los servicios que se agregó a las transmisiones NTSC fue el SAP o Segundo Audio del Programa en el que además de transmitir el audio que se transmitía

regularmente se transmite el audio en un segundo idioma con calidad mono. SAP es parte del estándar de sonido multicanal de televisión o MTS (Multichannel Television Sound) desde 1984. La señal SAP no se modula con la portadora de audio sino con una subportadora a 78.670 kHz, que es 5 veces la señal piloto MTS. La señal piloto MTS debe estar en fase con la frecuencia de sincronía horizontal de la portadora de video y a diferencia de un canal regular de audio que ocupa una banda de 50 Hz a 15 kHz, la señal de SAP solo tiene una banda de 60 Hz a 12 kHz.

1.2.2 PAL

El sistema Línea de Fase Alternada o PAL (Phase Alternating Line) surgió en los laboratorios Telefunken en 1963 para reducir los defectos en los tonos del color y mejorar la calidad del sistema NTSC. En este sistema las señales de diferencia de color se invierten de fase en cada línea. Este sistema se utilizó en la mayoría de los países europeos menos en Francia, en varios países africanos, en Argentina, en Brasil y en algunos países del Sureste de Asia.

TABLA 2. CARACTERÍSTICAS DEL SISTEMA PAL

Parámetro	Estándar 625/50
Líneas por Cuadro	625
Líneas por Campo	312.5
Tramas por segundo	25
Campos por segundo [Hz]	50
Frecuencia de barrido horizontal [Hz]	15,625
Duración del borrado de campo [líneas]	25
Duración del borrado de trama [líneas]	50
Líneas activas por trama	575
Resolución vertical [LPH]	402

Duración de la línea [μs]	64
Duración del borrado o supresión horizontal [μs]	12 ± 0.3
Duración de la línea activa [μs]	52
Duración del ciclo del par de líneas [μs]	0.194
Ancho de banda para resolución vertical y horizontal igual [MHz]	5.15
Factor de resolución horizontal [líneas/MHz]	78
Resolución horizontal [LPH]	390
Relación entre resolución vertical y horizontal H/V	0.97

Las características del sistema de exploración del sistema PAL se describen en la tabla 2. Características del Sistema PAL.

Los canales para transmitir video codificado con el sistema PAL ocupan 8 MHz en el espectro radioeléctrico como se muestra en la figura 1.3. La portadora de las señales de color en la codificación PAL se encuentra a 4.43 MHz de la portadora de video y la portadora de audio tiene una separación de 6 MHz de la portadora de video. En la codificación PAL la banda lateral residual ocupa 1.25 MHz. [1]

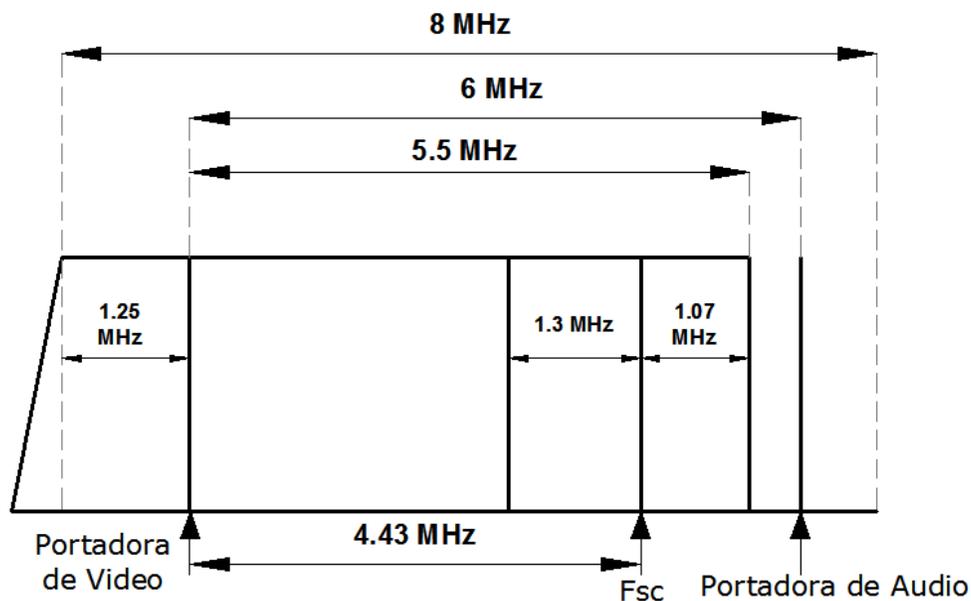


FIGURA 1. 3 ESTRUCTURA DE UN CANAL DE TELEVISIÓN CON LA CODIFICACIÓN PAL [1]

1.2.3 SECAM

El sistema SECAM (Secuencial Couleur Avec Memoire) o Sistema Secuencial de Transmisión de las señales de diferencia de color fue desarrollado en Francia por Henry France en 1958. Este sistema fue adoptado por Francia, por países bajo la influencia soviética y algunos países africanos.

A diferencia de los sistemas NTSC y PAL donde las señales de diferencia de color se transmiten simultáneamente, en el sistema SECAM en una línea se envía una señal de diferencia de color (C_{R-Y}) y en la siguiente línea se envía la otra señal de diferencia de color (C_{B-Y}). En cuanto al proceso de exploración de la imagen se utilizan parámetros idénticos a los del sistema PAL que se especifican en la tabla 2. Características del Sistema PAL. [1]

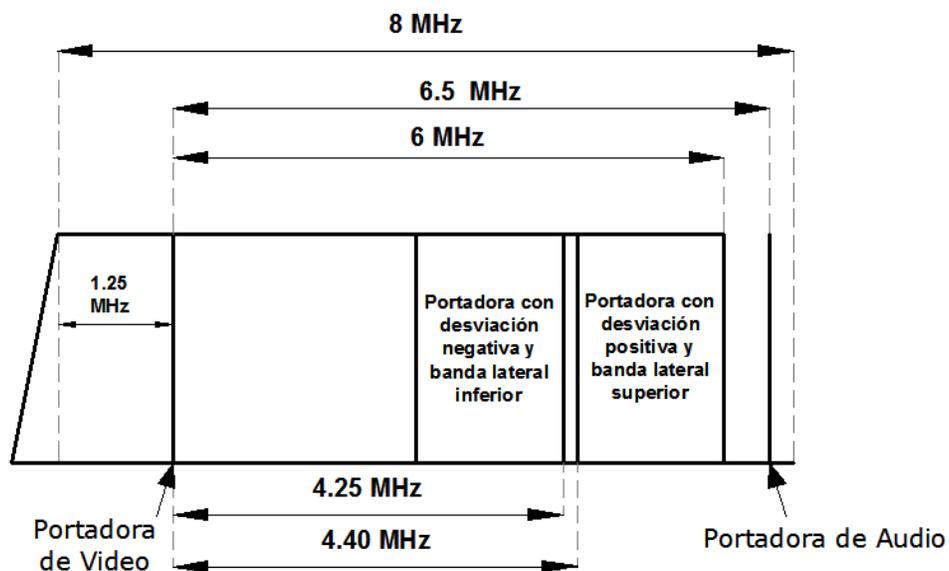


FIGURA 1. 4 ESTRUCTURA DE UN CANAL DE TELEVISIÓN CON LA CODIFICACIÓN SECAM [1]

Los canales codificados con codificación SECAM utilizan canales de 8 MHz en el espectro radioeléctrico y su banda lateral residual ocupa 1.25 MHz en el canal al igual que en la codificación SECAM, sin embargo, existe una portadora con desviación negativa para la

banda lateral inferior de la información de color a 4.25 MHz de la portadora de video y otra portadora con desviación positiva para la banda lateral superior de la información de color a 4.4 MHz de la portadora de video. La codificación SECAM también se diferencia de la PAL porque su portadora de audio se encuentra a 6.5 MHz de la portadora de video. La estructura de un canal de televisión codificado con SECAM se puede observar gráficamente en la figura 1.4.

1.2.4 FRECUENCIA DE MUESTREO

En los sistemas de televisión digital se toman las señales de video por componentes, es decir la señal de luminancia y las dos señales de diferencia de color se toman por separado para digitalizar cada una.

La UIT (Unión Internacional de Telecomunicaciones) estableció la recomendación ITU-R BT. 601-5 sobre la digitalización de la señal de video por componentes, para la compatibilidad entre los estándares de video. En esta recomendación se establecieron exactamente las siguientes señales:

$$E'_Y = 0.299E'_R + 0.587E'_G + 0.11E'_B + \text{sinc.H} + \text{sinc.V}$$

$$C_B = 0.564(E'_B - E'_Y)$$

$$C_R = 0.713(E'_R - E'_Y)$$

Donde E'_R es la componente de color rojo de la imagen, E'_G es la componente de color verde, E'_B es la componente de color azul sinc.H es la señal de sincronización horizontal y sinc.V es la componente de sincronización vertical.

Para lograr que las señales digitales de los sistemas de televisión NTSC y PAL fueran compatibles la ITU tomo las siguientes consideraciones:

1. El número de muestras de la señal de luminancia en una línea de TV debe ser entero. Si "M" es el número de muestras de luminancia en una línea de televisión, para hacer compatibles los estándares 625/50 y 525/50:

$$M_{625/50} = \frac{1}{625 \cdot 25 \cdot T_m} = 858$$

$$M_{525/60} = \frac{1}{525 \cdot 29.97 \cdot T_m} = 864$$

$$f_m = f_H \cdot M_{625/50} = 13.5 \text{ [MHz]}$$

$$f_m = f_H \cdot M_{525/60} = 13.5 \text{ [MHz]}$$

2. Frecuencia de muestreo de 13.5MHz de tal forma que en cualquiera de los dos estándares se cumpla el criterio de Shannon:

$$f_{625/50} = 2 \cdot f_{\text{máx}} = 2 \cdot 5.5 \text{ MHz} = 11 \text{ MHz} < f_m = 13.5 \text{ MHz}$$

$$f_{525/60} = 2 \cdot f_{\text{máx}} = 2 \cdot 4.5 \text{ MHz} = 9 \text{ MHz} < f_m = 13.5 \text{ MHz}$$

3. Que el número de muestras de luminancia en una línea de televisión activa fuera el mismo para los dos sistemas:

$$\text{Muestras por línea de televisión activa} = \frac{858 \cdot 52}{64} = \frac{864 \cdot 52}{63.55} = 702 \text{ muestras}$$

Pero se toman 720 muestras para que al inicio de cada línea de televisión activa agregando un nivel de negro de referencia al inicio y al final de cada línea de televisión activa. Esto provoca que línea de televisión activa dure 52 μ s. La señal de video digital tiene la estructura que se muestra en la figura 1.5 después del proceso de digitalización [2].

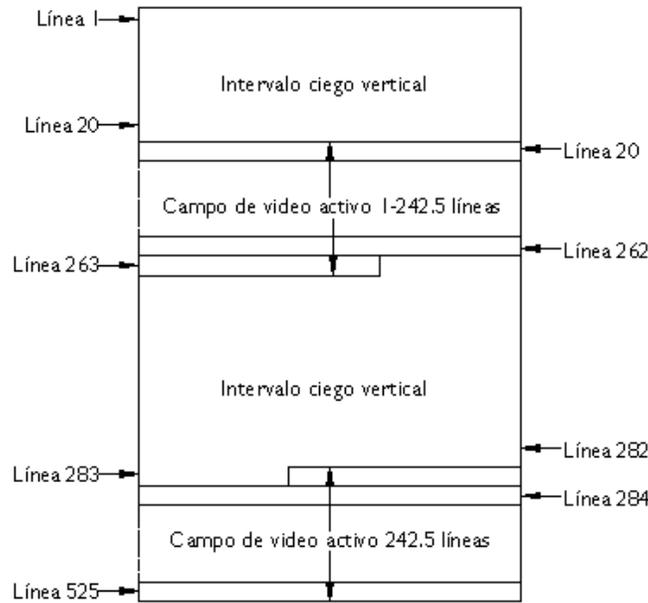


FIGURA 1. 5 ESTRUCTURA DE LA SEÑAL DE VIDEO DIGITALIZADO

1.3 TRANSMISOR Y RECEPTOR DE TELEVISIÓN DIGITAL

Tomando en cuenta la frecuencia de muestreo que se considera en la sección 1.2.4 (13.5 MHz) [2] y 8 bits por muestra (a pesar de que en los estudios se manejan 10 bits por muestra) se tiene la siguiente tasa bits para cada componente de video:

$$R = 13.5 \text{ [MHz]} \times 8 \text{ [bits]} = 108 \left[\frac{\text{Mb}}{\text{s}} \right]$$

Si se multiplexan los 3 flujos elementales se obtiene un flujo de 324 Mbps.

1.3.1 VENTAJAS DE LAS SEÑALES DIGITALES

Una tasa de 324 Mbps es muy alta y requiere de una modulación digital que tenga una constelación muy grande para enviarla en un canal de 6 o 7 MHz y al tener un gran

número de símbolos en su constelación su resistencia a errores disminuye, si se utilizará una modulación con una constelación de menos símbolos se necesitaría un ancho de banda mucho mayor. Pero ya que se trata de una señal digital se le pueden aplicar técnicas de compresión que no se pueden aplicar a las señales analógicas. En la sección de herramientas de compresión de video se describen estas técnicas, sin embargo, es importante resaltar que utilizando MPEG-4 se puede reducir la tasa de 324Mbps hasta 3Mbps. Esto implica un mejor aprovechamiento del espectro radioeléctrico al poder enviar más de una canal de televisión donde antes solo se podía transmitir uno.

Otra característica muy importante de las señales digitales es la codificación para protección contra errores que se les puede aplicar. En las transmisiones de televisión analógicas el ruido degrada la calidad de la imagen a pesar de que se continúa recibiendo la señal. En el caso de la transmisión de Televisión digital la imagen se recibe con la señal que se envió y los errores que provocó el ruido del canal de comunicaciones pueden detectarse y repararse hasta cierto límite de acuerdo a la codificación contra errores aplicada. Por otra parte una señal de televisión digital necesita una relación señal a ruido menor que una señal de televisión analógica para mantener la misma calidad que la imagen original.

Es muy importante mencionar la alta definición o HDTV que solo se puede transmitir por medio de sistemas de televisión digital, ya que esto implica una mejora en la calidad de la televisión. La resolución de la alta definición se estableció tomando en cuenta televisores más grandes o distancias menores entre el televisor y el espectador. Para la definición estándar o SDTV se considera la distancia entre el espectador y el televisor 6 veces la altura del televisor y para los sistemas de alta definición se considera la mitad, es decir, 3 veces la altura del televisor. Además en ambos casos se toma en cuenta la agudeza visual del sistema visual humano para determinar la resolución de la pantalla [1].

Por último la televisión digital ya funcionaba a nivel de estudio, es decir, en el estudio de televisión se digitalizaba la señal para modificarla y procesarla, sin embargo, para transmitirla se cambiaba a un formato PAL o NTSC.

1.3.2 SISTEMAS DE TELEVISIÓN DIGITAL

Los estándares sirven para mantener la interoperabilidad de equipos y aplicaciones. Existen cuatro estándares de transmisión Televisión Digital en la actualidad: El sistema creado en Estados Unidos, e implementado en Norteamérica y Corea del Sur, conocido con el nombre del comité que lo creó ATSC (Advance Television System Committee), fue publicado en 1995 y adoptado por Estados Unidos en 1996. El sistema creado en Japón por los Laboratorios de la corporación NHK, conocido como ISDB-T (Integrated Services Digital Broadcasting - Terrestrial). Ha sido implementado en Japón y en la Sudamérica. El sistema europeo llamado DVB-T (Digital Video Broadcast - Terrestrial) fue publicado por primera vez en 1997 y su primera transmisión fue hecha en el Reino Unido en 1998. El estándar DVB-T ha sido adoptado por todos los países de la Union Europea, los países asiáticos con excepción de China, Japón y Corea del Sur y por la mayoría de los países Africanos. Por último el sistema DTMB (Digital Terrestrial Multimedia Broadcast) fue creado en china y se utiliza en China, Hong Kong, Laos, Macau y otros países de forma experimental. [3]

1.3.3 DIAGRAMA DE BLOQUES DEL TRANSMISOR DE TELEVISIÓN DIGITAL

En la figura 1.6 se muestra el diagrama a bloques del transmisor de televisión digital y en la tabla 3. Tecnologías empleadas en la Televisión Digital Terrestre se especifica la tecnología que se utiliza en cada sistema de televisión digital. El diagrama de bloques de los sistemas de televisión digital terrestre ATSC se muestra en la figura 1.7, el de los

sistemas DVB-T se muestra en la figura 1.8 y el de los sistemas ISDB-t se muestra en la figura 1.9.

TABLA 3. TECNOLOGÍAS EMPLEADAS EN LA TELEVISIÓN DIGITAL TERRESTRE

Sistema de Televisión Digital	ATSC	DVB-T	ISDB-T	DTMB
Codificador de Video	MPEG-2 o H.264/MPEG-4 AVC	MPEG-2 o H.264/MPEG-4 AVC	MPEG-2 o H.264/MPEG-4 AVC	MPEG-2 o H.264/MPEG-4 AVC
Codificador de Audio	Dolby AC-3	MPEG-2 Capa 2 y Dolby AC-3	MPEG-2 AAC	MPEG-2 y AVS
Capa de Transporte	MPEG-TS	MPEG-TS	MPEG-TS	MPEG-TS
Codificación Contra Errores	Reed-Solomon y Trellis	Reed-Solomon y Código Convolutacional	Reed-Solomon y Código Convolutacional	LDPC y BCH
Modulación	8-VSB	COFDM	BST-OFDM	TDS-OFDM

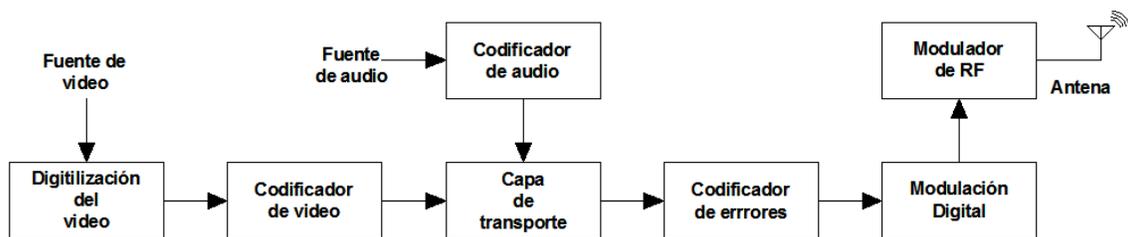


FIGURA 1. 6 DIAGRAMA A BLOQUES DEL TRANSMISOR DE TELEVISIÓN DIGITAL TERRESTRE

Como se muestra en el diagrama a bloques del transmisor de televisión digital terrestre a las señales digitales tanto de audio como de video se les aplica una codificación. La principal razón para realizar la codificación es disminuir la tasa de bits de las señales, aunque la codificación también agrega la información necesaria para decodificar la señal. Una ventaja de los sistemas digitales es la posibilidad de cambiar el codificador de video o actualizarlo mientras se implemente mediante software. Esta codificación corresponde a la codificación de fuente del sistema de comunicaciones, debido a que busca eliminar la redundancia de la señal original.

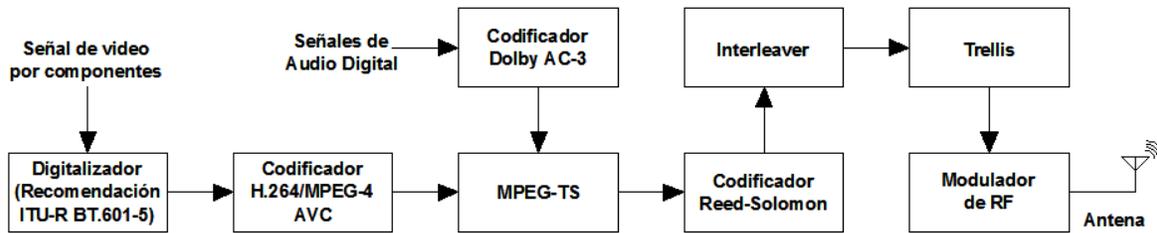


FIGURA 1. 7 DIAGRAMA A BLOQUES DEL TRANSMISOR DE ATSC

La capa de transporte en el transmisor sirve para multiplexar las señales de audio y video codificado y en la actualidad todos los sistemas de televisión digital terrestre utilizan el sistemas que se especifica en el estándar MPEG-2 también conocido como MPEG-TS (Transport Stream), incluso se pueden multiplexar más de una señal de audio por cada señal de video lo que equivale al SAP (Second Audio Program) en las transmisiones de televisión analógica donde se envía el audio en otro idioma que por lo general es el audio original de la película y no el doblaje. Al audio y video que corresponden al mismo canal de televisión se les llama flujo de programa y puede haber varios flujos de programa multiplexados en un flujo de transporte que es el flujo de bits que entra al codificador contra errores. Incluso a un solo flujo de video le pueden corresponder uno o más flujos de audio. Además de agregar la información necesaria para demultiplexar las señales de video y audio en esta parte del procesamiento de la señal también agrega información para sincronizar el video y el audio del programa de televisión.

El codificador contra errores del diagrama a bloques del transmisor de televisión digital corresponde a la codificación de canal en un sistema de comunicaciones, debido a que en esta parte se agrega la redundancia necesaria a la señal. En todos los sistemas de televisión digital terrestre se utilizan dos códigos de protección contra errores concatenados por un interleaver. Aunque los códigos pueden cambiar de un sistema a otro.

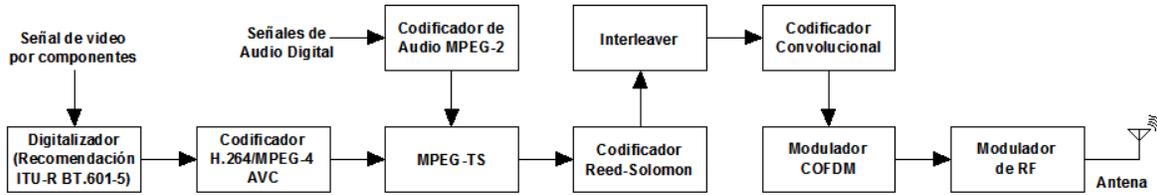


FIGURA 1. 8 DIAGRAMA A BLOQUES DEL TRANSMISOR DE DVB-T

Todos los sistemas de televisión digital terrestre utilizan el mismo codificador de video, el H.264/MPEG-4 AVC debido a los esfuerzos de la ISO y la ITU (International Telecommunications Union) por normalizar la codificación de video y a que ofrece el mejor desempeño. De la misma manera en todos los sistemas de televisión digital terrestre se aplica la recomendación ITU-R BT.601-5. Sin embargo, el codificador de video H.264/MPEG-4 AVC no está estandarizado, el estándar H.264/MPEG-4 AVC únicamente define el decodificador y establece un conjunto de herramientas que se pueden utilizar en el codificador. Al estandarizar únicamente el decodificador, los algoritmos de codificación pueden seguir mejorando pero el consumidor no necesita cambiar su decodificador y los codificadores se pueden implementar con software propietario de manera que se fomente la competencia en la industria. La compresión que se realiza en los codificadores MPEG se conoce como compresión asimétrica debido a que el codificador es mucho más complejo que el decodificador. Por su parte el decodificador únicamente realiza un conjunto de acciones fijas.

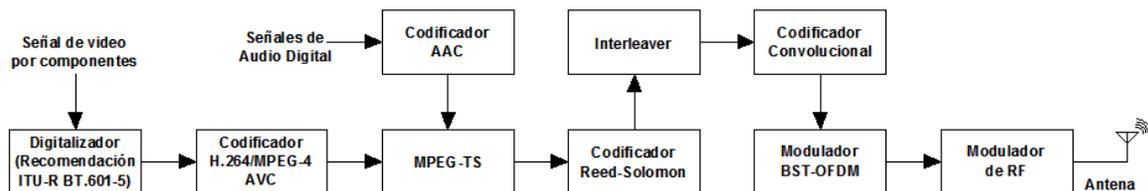


FIGURA 1. 9 DIAGRAMA A BLOQUES DEL TRANSMISOR DE ISDB-T

La modulación OFDM o sus variantes son utilizadas por todos los sistemas de televisión digital terrestre menos por el sistema ATSC debido a que cuando se definió el estándar antes de que las patentes de la modulación OFDM vencieran. La modulación OFDM ofrece la ventaja de que no presenta ICI (Interferencia entre canales), ni ISI (Interferencia entre Símbolos) y tiene buena eficiencia espectral.

1.3.4 DIAGRAMA DE BLOQUES DEL RECEPTOR DE TELEVISIÓN DIGITAL

En la figura 1.10 se muestra el diagrama a bloques del receptor de televisión que hacen el trabajo hacia atrás de los bloques del transmisor de tal forma que su resultado final sea una señal de video y una señal de audio que el televisor pueda mostrar en la pantalla.

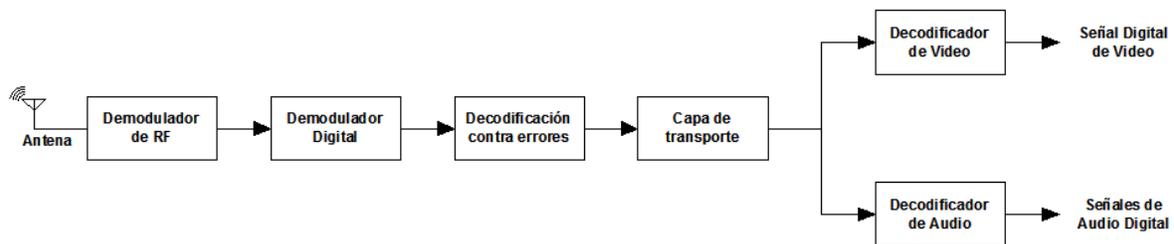


FIGURA 1. 10 DIAGRAMA A BLOQUES DEL RECEPTOR DE TELEVISIÓN DIGITAL TERRESTRE

Además la importancia del receptor reside en poder aplicar una ecualización que contrarreste los efectos del canal de comunicaciones y en recuperar la sincronía de la señal para realizar una demodulación digital eficiente y la sincronía de la señal de video y de audio de manera que el televidente no perciba ningún desfase entre la señal de audio y de video al momento de mirar el programa de televisión.

2. CODIFICACIÓN DE FUENTE

La codificación de fuente busca eliminar la redundancia de la señal digital de manera que solo se conserve la información únicamente. También son conocidas como compresión o codificación de video y audio. Se aplican herramientas semejantes con el video y el audio pero en conjunto se aplican de modo diferente.

2.1 CODIFICACIÓN DEL VIDEO

La codificación de video se refiere a un conjunto de técnicas que se aplican a las señales de video digital para reducir su tasa de bits, de manera que su transmisión sea adecuada para el ancho de banda del canal de comunicaciones. Por otro lado en la codificación del video también se agregan bits de sincronización y la información necesaria para que se realice la decodificación en el receptor.

2.1.1 HERRAMIENTAS PARA LA COMPRESIÓN DE VÍDEO

Para realizar la compresión de la señal de video se pueden utilizar dos estrategias, la codificación entrópica y la predictiva. En la codificación predictiva se utilizan algoritmos que predicen la señal de video y se resta la señal que se predice a la señal original para transmitir solo una señal de diferencia o señal de error. Cuando el algoritmo de predicción es eficiente la señal de diferencia o de error es un flujo de bits donde predominan los ceros, a esta señal de error se le aplica la codificación entrópica para reducir la tasa de bits. La codificación entrópica busca reducir la tasa de bits cambiando las secuencias que

más aparecen por unas de menor tamaño, esto es si una secuencia de 90 ceros aparece continuamente en la señal solo se envía un cero por cada vez que aparece esta secuencia de 90 ceros. Generalmente las señales de video digital no contienen cadenas largas de ceros por lo que se aplica una transformación y un proceso de cuantización a los coeficientes que resultan de esa transformación.

Transformada de Coseno Discreta (DCT).

La transformada de coseno discreta mejor conocida como DCT, por sus siglas en inglés (Discrete Cosine Transform), se usa para cambiar las señales en el dominio del tiempo al dominio de la frecuencia, partiendo del principio de que la mayor cantidad de información se concentra en la región de bajas frecuencias, es decir en pocos coeficientes. Estos coeficientes pueden ser cuantizados de forma que los coeficientes que no contienen mucha información reduzcan su tamaño, sin afectar la calidad de la imagen cuando se reconstruye utilizando los nuevos coeficientes. La DCT es un proceso de decorrelación de datos. DCT es una transformada similar a la transformada de Fourier discreta pero no utiliza números imaginarios.

Los bloques de $M \times N$ pixeles se convierten en conjuntos de coeficientes de $M \times N$ y el coeficiente $F(0,0)$ representa el valor de DC del bloque. El coeficiente de DC del bloque es un promedio de los valores de los pixeles de la señal. Los otros coeficientes se conocen como coeficientes de AC. Generalmente el coeficiente de DC tiene una amplitud muy grande comparada con los coeficientes de AC [1] [4].

Cuantización de los Coeficientes de DCT

Una vez que se obtienen los resultados de la transformada DCT se les multiplica por una matriz de cuantización ya que algunos son más importantes que otras debido a que el sistema visual humano es menos sensible a los coeficientes de alta frecuencia que a los

de baja, de esta forma se logra obtener una mejor codificación entrópica debido a que se crean cadenas de ceros más largas. En esta operación se pierde información sin embargo en general es información que el sistema visual humano no puede detectar.

Exploración de los Coeficientes de DCT

El proceso de exploración de los coeficientes de DCT consiste en convertir un bloque de coeficientes de DCT bidimensional de $M \times N$ en un flujo de bits serial, es decir los coeficientes uno tras otro. De manera que se obtenga un flujo serial de bits. Mientras se consigan cadenas más largas de ceros se mejora la compresión al aplicar VLC (Codificación de Longitud Variable), por esto se pueden aplicar dos tipos de exploración de coeficientes. La exploración en Zigzag, que se representa en la figura 2.1 (A) del lado izquierdo, y la exploración alternada que se representa en la misma figura pero del lado derecho. El patrón utilizado se debe indicar en el flujo de bits codificado.

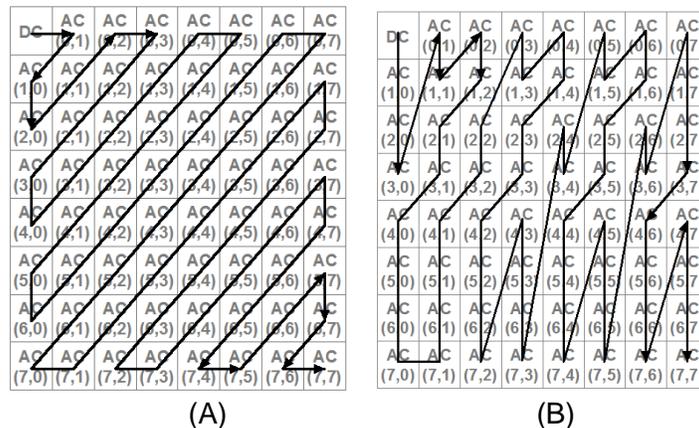


FIGURA 2. 1 EXPLORACIÓN EN ZIG-ZAG(A) Y ALTERNADA (B) DE COEFICIENTES DCT

Codificación de Longitud Variable (VLC)

Los códigos VLC siguen el criterio que la longitud de una palabra debe ser función de la frecuencia probabilidad de aparición de un valor, cuanto mayor sea la probabilidad de aparición menor hacer el número de bits que forman la palabra. Estos códigos también son llamados estadísticos y son una codificación entrópica. En el decodificador se debe

tener la misma tabla que se utilizó para codificar la señal. Este proceso corresponde a la codificación entrópica en MPEG-2 [5] [4].

Codificaciones Predictivas

Existen dos tipos de predicciones intraframe o espacial y interframe o temporal.

Predicciones intraframe: se dan cuando los elementos previos pertenecen a la misma imagen. Un ejemplo de este tipo de predicciones se da con la técnica DPCM (Diferencial Pulse Code Modulation) aplicada a los coeficientes de DC de cada bloque [2].

Predicciones interframe: se presenta cuando la predicción se da entre dos frames diferentes, siempre y cuando una imagen previa sea muy parecida. Este tipo de predicciones se utilizan en las técnicas de reducción de datos temporales, únicamente se pueden utilizar para comprimir el video ya que se aplican entre diferentes campos o cuadros [2].

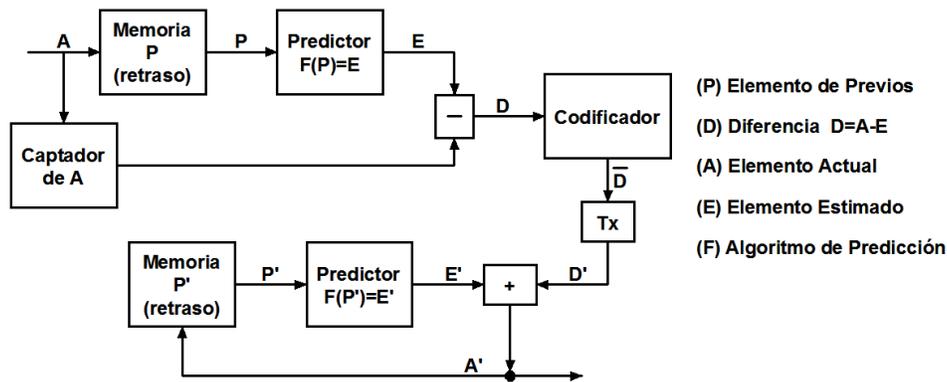


FIGURA 2. 2 DIAGRAMA A BLOQUES DE LA CODIFICACIÓN DPCM

Predicción PCM Diferencial (DPCM)

El DPCM es el algoritmo de predicción que se utiliza en la comprensión de la señal de video en las normas JPEG y MPEG, en éste se considera que el elemento estimado, es decir el que se predice, es elemento previo (ver figura 2.2). Se utiliza con los coeficientes

de DC ya que entre dos bloques juntos muchas veces varía poco o el coeficiente de DC es el mismo, por lo tanto utilizando para esta técnica únicamente se suma la diferencia en el decodificador, positiva o negativa al coeficiente de DC anterior para obtener el coeficiente de DC del siguiente bloque. Al aumentar el número de ceros en el flujo bits la compresión que se realiza con VLC son más eficientes [4] [5].

Predicción por Compensación de Movimiento

Para comprimir una secuencia de imágenes en el tiempo se podría mandar únicamente una imagen y pedir que la pantalla repita la imagen cierto número de veces mientras esta no cambie en la secuencia original. En la compresión de video digital se hace algo semejante, primero se codifica una imagen con DCT, y pasa por un proceso de cuantización, se decodifica la imagen con la transformada DCT inversa, se le aplica una compensación de movimiento a partir de un algoritmo y se le resta a la imagen siguiente de tal manera que si funciona el algoritmo el resultado será solo ceros. Después en el decodificador volverá a sumar la primera imagen a las demás para recuperar la secuencia de imágenes original. Se limitan a un grupo de imágenes para que los errores que se puedan cometer en la codificación no se transmitan indefinidamente de una imagen a otra [1].

Vector de Movimiento

Como los métodos de compresión se pueden utilizar para diferentes tipos de videos, no solo para los que son estáticos, se usan esquemas de predicción mejorados que detectan el movimiento entre imágenes sucesivas y se crean vectores de movimiento para indicar la nueva posición de los detalles de esa imagen. Los vectores de movimiento le indican al decodificador las coordenadas donde se deben replicar los bloques codificados de la primera imagen. Este esquema de compresión se denomina predicción por compensación

de movimientos interframe. Todos los macrobloques de referencia, es decir los de la primera imagen que se codifica sin restarle nada. Los vectores de movimiento se transmiten junto a los bloques de coeficientes de DCT de la diferencia, también conocida como señal de error.

En ocasiones no es utilizada la predicción de movimiento y solamente se aplica predicción temporal para terminar con la propagación de errores que ocurren en la transmisión.

2.1.2 ESTÁNDARES DE CODIFICACIÓN DE VIDEO

Para las transmisiones de televisión digital terrestre únicamente se han utilizado 2 estándares de codificación de video el MPEG-2 que fue el primero que se utilizó y el estándar MPEG-4 Parte 10-H.264 AVC.

MPEG-2

MPEG o el grupo de expertos de imágenes de video es un grupo de expertos conformado por la Organización Internacional de Normalización (ISO) y por la Comisión Electrotécnica Internacional (IEC) para establecer estándares para la transmisión de audio y video.

La compresión que proponen los estándares MPEG es llamada asimétrica ya que el codificador es más complejo que el decodificador, mientras que el codificador debe ser adaptativo mientras que el decodificador tiene una estructura fija. Sin embargo, los estándares MPEG no mencionan la estructura del codificador, sino la forma en la que el decodificador descomprime el flujo de bits para recuperar el video original. De esta forma los codificadores se pueden construir utilizando software propietario y se incentiva la competencia. Además se define la capa de transporte en los estándares, es decir la forma en la que se multiplexan diferentes flujos de vídeo, audio y datos. [6]

Tipos de Imágenes

En MPEG se definen 4 tipos de imágenes: Imágenes I, Imágenes P, Imágenes B e Imágenes D. Las Imágenes I y las Imágenes P se consideran cuadros clave y el espacio entre cuadros clave conoce como M, mientras que el espacio entre imágenes I se conoce como N. Un GOP está formado por una o más imágenes y no tiene una estructura periódica necesariamente. Enseguida se explica cada tipo de imágenes

- Imágenes I (Intracodificadas): este tipo de imágenes se codifica independientemente de las demás, por lo que no tiene referencias a otras imágenes. También evita la propagación de errores por su tipo de compresión. Este tipo de imágenes se componen únicamente de intrabloques.
- Imágenes P (Predictivas): estas imágenes son comprimidas utilizando la compensación por predicción de movimiento hacia adelante a partir de imágenes I o P anteriores, por lo tanto están más comprimidas que la imágenes I. Sirven como referencia para decodificar imágenes B o imágenes P futuras.
- Imágenes B (Predichas Bidireccionalmente): este tipo de imágenes permite que los macrobloques sean codificados utilizando compensación de movimiento en ambos sentidos (Predicción temporal bidireccional), es decir que se toma como referencia imágenes I y P anteriores y posteriores. En las escenas con cambios abruptos o movimiento multidireccional rápido se codifican menos imágenes B que en las escenas de video muy parecidas o que presentan poco movimiento. Además de introducir retraso es necesario almacenar dos imágenes para decodificar este tipo de imágenes.
- Imágenes D o DC: son imágenes de muy baja resolución que se obtienen al decodificar única-mente el coeficiente de DC de una imagen codificada con DCT. Este tipo de imágenes se define para permitir búsquedas rápidas en el almacenamiento.

Mientras más imágenes B existan más se reduce la correlación entre la imagen I o P anterior y la Imagen I o P siguiente. También causa un retraso de codificación y necesita un decodificador más complejo y con más memoria, debido a que necesita reordenar las imágenes pues primero decodifica las imágenes I, luego las P y al final las B [4].

Estructura del Video MPEG-2

Todos los estándares en los que MPEG ha participado tienen la siguiente estructura:

Capa de secuencia de video: está formada de una cabecera, uno o más GOPs y un código de fin de secuencia. La cabecera contiene información relacionada con el tamaño vertical y horizontal de la imagen, relación del aspecto de pixeles, la tasa de imágenes, la tasa de bits, el control para la carga de 64 octetos de bits para las tablas de intracuantización y no intracuantización.

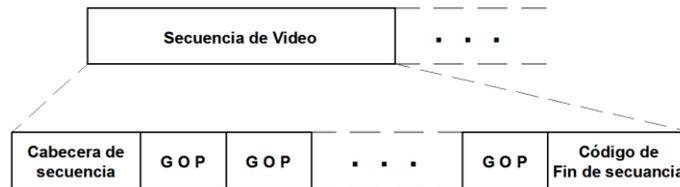


FIGURA 2. 3 CAPA DE SECUENCIA DE VIDEO

Capa GOP: se forma de un conjunto de imágenes de reproducción y una cabecera de GOP, pero siempre comienzan con una imagen I después de la cabecera. La cabecera contiene los siguientes parámetros: El código de tiempo de 25 bits, que contiene la hora, los minutos y los segundos donde inicia la secuencia y otros parámetros del GOP.

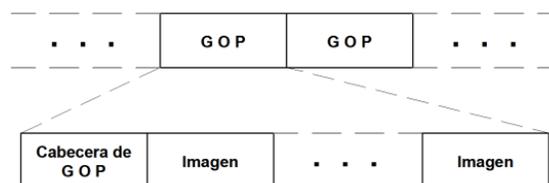


FIGURA 2. 4 CAPA GOP

Capa de Imagen: se utiliza como unidad principal de codificación en MPEG-2. Cada imagen contiene una cabecera de imagen y varias rebanadas. Su cabecera contiene los siguientes parámetros: la referencia temporal que sirve al decodificador para determinar el número de imagen en la secuencia y el orden de reproducción, el tipo de imagen, es decir si se trata de una Imagen D, I, P o B. El encabezado también contiene la resolución y el rango del vector de movimiento hacia adelante para las imágenes P y B, la resolución y el rango del vector de movimiento hacia atrás para las imágenes B, y por último la información de sincronización.

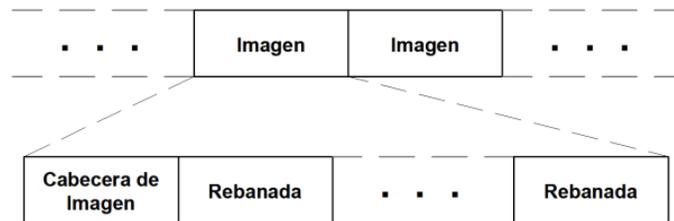


FIGURA 2. 5 CAPA IMAGEN

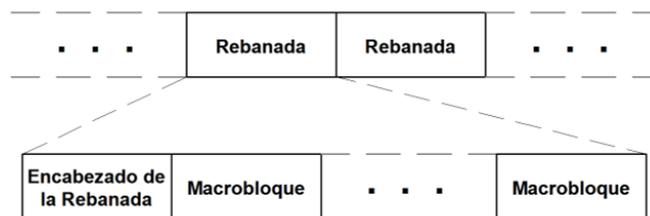


FIGURA 2. 6 CAPA DE REBANADA

Capa de rebanada: una rebanada es una secuencia de macrobloques contiguos en el orden de exploración, es decir de izquierda a derecha y de arriba abajo. Las rebanadas son muy importantes para evitar los errores, ya que los decodificadores pueden recuperar la sincronización al comienzo de la siguiente rebanada si en alguna se produce un error. Mientras más cabeceras haya, los errores se evitan pero como también existirán más cabeceras de rebanada se utilizarán más bits sin carga. Cada rebanada se forma de un encabezado de rebanada y de varios macrobloques, en cada encabezado de rebanada

está la posición vertical donde la rebanada comienza y esto permite utilizarla como una unidad de resincronización. Al inicio de cada rebanada se encuentra el coeficiente de DC que se usa como referencia en la codificación DPCM de los demás coeficientes de DC.

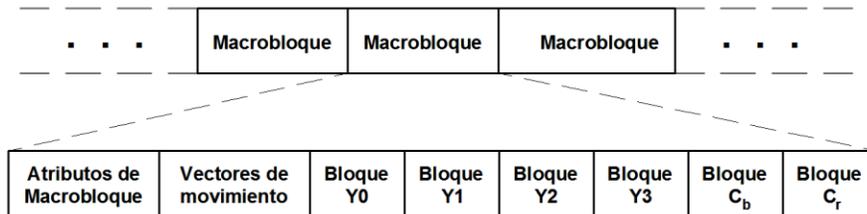


FIGURA 2. 7 CAPA DE MACROBLOQUE

Capa de macrobloque: se compone de un conjunto de 16x16 coeficientes de DCT de la señal de luminancia y de dos bloques de 8x8 de cada señal de crominancia



FIGURA 2. 8 CAPA BLOQUE

Capa de bloque: consiste de bloques de coeficientes de DCT de la señal de luminancia o de alguna de las señales de crominancia. Cuando un macrobloque que codifica como intramodo el coeficiente DC actual se predice a partir de los coeficientes DC de previos macrobloques.

MPEG-4: Parte 10 - H.264/AVC (Advanced Video Coding)

La parte 10 del estándar MPEG4 y el estándar H.264/AVC son equivalente, uno desarrollado por la ISO y IEC y el otro por la ITU. Dicho estándar se divide en dos capas, la capa de codificación de video o VCL (Video Coding Layer) y la capa de abstracción de red o NAL (Network Abstraction Layer) que provee la cabecera apropiada de acuerdo a la

red o al medio de almacenamiento. El estándar H.264/AVC es muy parecido a su predecesor MPEG-2, en especial en su estructura, sin embargo, las diferencias logran una mayor capacidad de compresión.

En el ámbito de la predicción espacial direccional la diferencia se encuentra en que en el estándar H.264/AVC se utiliza un modelo de compensación de movimiento avanzada utilizando una predicción de bloque de tamaño variable, una compensación de movimiento de precisión de un cuarto de muestra, predicción de imagen de múltiple referencia, predicción ponderada, un filtro de desbloqueo de lazo, una transformada entera de bloque de 8x8 o de 4x4 y dos modos de codificación entrópica de contexto adaptativo.

En cuanto a la resistencia a errores es más flexible H.264/AVC ya que la información de la cabecera de imagen y de secuencia de video se establece en estructuras llamadas conjuntos de parámetros, que pueden transmitirse de modo flexible para reducir la tasa de bits o para aumentar la resistencia a errores.

H.264/AVC permite un manejo más flexible de las rebanadas, en cuanto a la forma y cada rebanada se codifica independientemente de las otras rebanadas de la misma imagen para mejorar la resistencia a errores. Además se permite un orden arbitrario de las rebanadas.

La estructura del video codificado con H.264/AVC se diferencia en que en lugar de utilizar imágenes I, B y P, como los estándares predecesores, utiliza rebanadas I, B y P de forma que la predicción espacial se realiza a nivel de rebanadas en lugar de imágenes y en la misma imagen puede haber rebanadas de diferentes tipos.

En la parte de codificación de la entropía los codificadores H.264/AVC además de utilizar el sistema VLC que se utiliza en MPEG-2, también puede utilizar Codificación Aritmética

Binaria de Contexto Adaptativo mejor conocida como CABAC (Context-Adaptive Binary Arithmetic Coding). En CABAC cada símbolo se codifica en código binario y se codifica simétricamente, después para ajustarse a las estadísticas no estacionarias de los símbolos se utiliza un modelado de contexto.

2.3 CODIFICACIÓN DE AUDIO

Al igual que en el caso del video, para las señales de audio existen técnicas para reducir la tasa de bits. La tasa bits de una señal de audio no es muy grande, sin embargo, en los sistemas televisión digital se implementan sistemas de sonido multicanal (también conocidos como teatro en casa) y debido a que se trata de varias señales de audio con resolución de hasta 16 bits aumenta el ancho de banda y es necesario implementar una compresión de la señal.

2.3.1 PRINCIPIOS DE LA CODIFICACIÓN DE LAS SEÑALES DE AUDIO

La mayoría de los codificadores de audio se basan en la percepción humana, específicamente en los fenómenos de enmascaramiento de sonidos que se explican enseguida.

Enmascaramiento de Sonidos

Cuando dos tonos de frecuencias cercanas hacen vibrar zonas cercanas de la membrana basilar en el oído humano, se presenta un fenómeno de enmascaramiento frecuencial si uno de los dos tonos tiene una amplitud muy grande las dos en comparación con el otro, ya que las ondas vibraran a la frecuencia de ese tono y el otro tono no se podrá percibir.

Si el mismo tono de enmascaramiento frecuencial se detiene bruscamente y el otro tono de menor amplitud con otra frecuencia no se detiene, la zona que estaba vibrando de la membrana continúa vibrando a la frecuencia que se detuvo durante un tiempo menor a 200 ms, a este fenómeno se le llama enmascaramiento temporal.

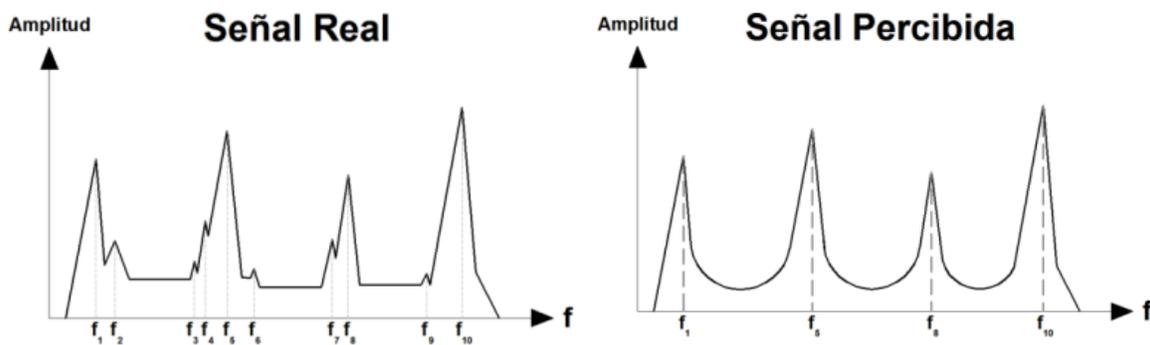


FIGURA 2. 9 ENMASCARAMIENTO FRECUENCIAL

El umbral de audibilidad varía con los enmascaramientos, los valores para cada tono aumentaran en las zonas próximas a los tonos enmascaradores, a este umbral se le llama umbral de audibilidad enmascarada y se utiliza para la compresión de la señal [2].

En esta compresión se pierde la información de las frecuencias enmascaradas que es información que el televidente no puede percibir. Cuando la compresión se basa en estos fenómenos se le llama compresión perceptual por su aspecto psicoacústico.

Otro aspecto que explota la compresión perceptual es umbral de audición, ya que el oído humano no percibe todas las señales con la misma intensidad, algunas frecuencias se perciben con mayor intensidad que otras, incluso puede cambiar de una persona a otra.

Otra técnica utilizada en la compresión de audio es la reducción de la redundancia temporal, por ejemplo existen partes donde se presentan silencios prologados y en ellos se puede reducir el tamaño del símbolo que representa los silencios para obtener una menor tasa de bits.

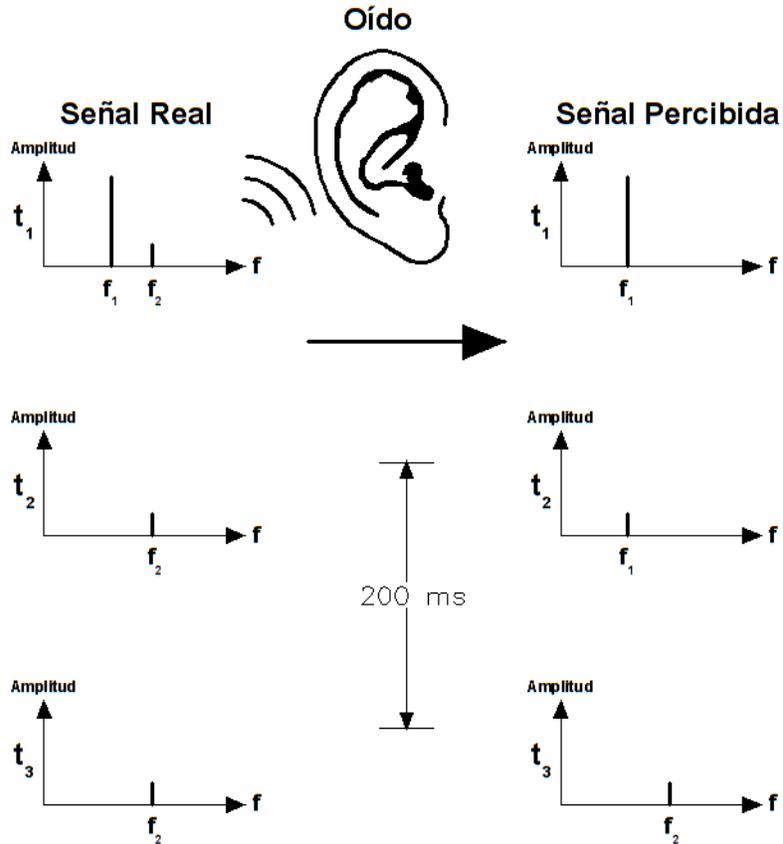


FIGURA 2. 10 ENMASCARAMIENTO TEMPORAL

Además se aplican algoritmos diferenciales, codificadores de entropía (de forma similar a la codificación de video) y sistemas de punto flotante de bloque que funcionan a partir de los principios de la percepción auditiva humana antes mencionados.

Codificación de Sub-bandas

Un método donde se utilizan las características del sistema auditivo humano es en la codificación de sub bandas, llamada así por que divide la señal de audio mediante un banco de filtros. Al pasar por el banco de filtros la señal se divide en un conjunto de sub-bandas de frecuencia que pasan por un cuantizador (un cuantizador por cada sub-banda), todos controlados por un bloque de FFT y por un bloque de modelo psicoacústico. Los

cuantizadores pueden suprimir la sub-banda o reducir el número de pasos de cuantización

Los intervalos de tiempo en los que se divide la señal se llaman tramas y tienen una duración múltiplo del período de muestreo de la señal: $T = k \cdot T_m$. Incluso el ruido que genera este proceso no es apreciable [7].

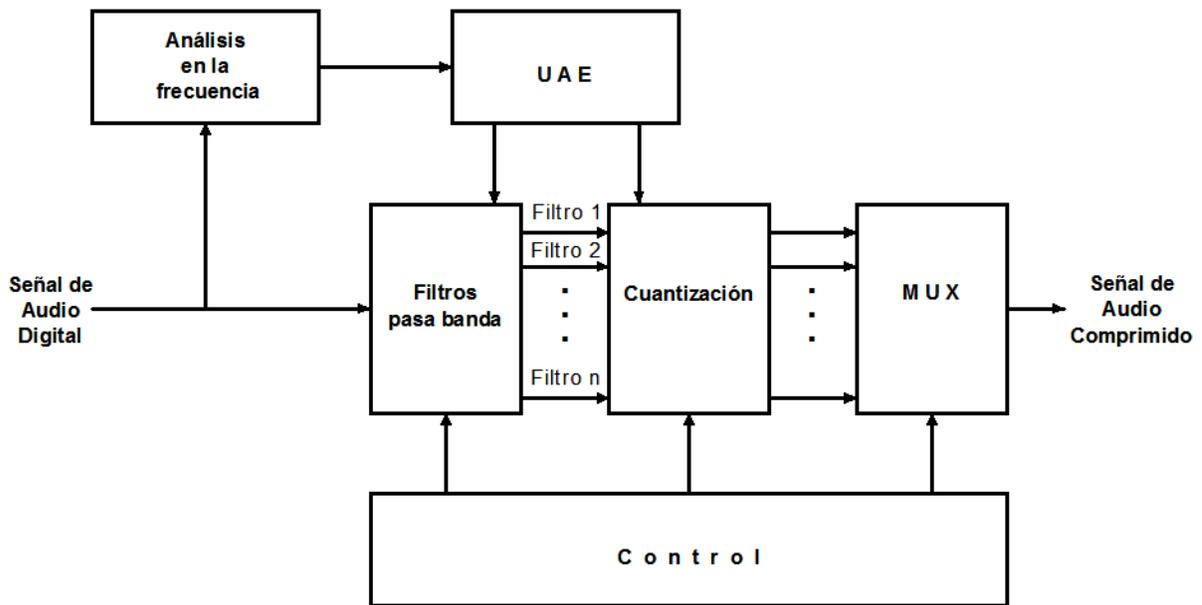


FIGURA 2. 11 CODIFICACIÓN DE SUB-BANDAS.

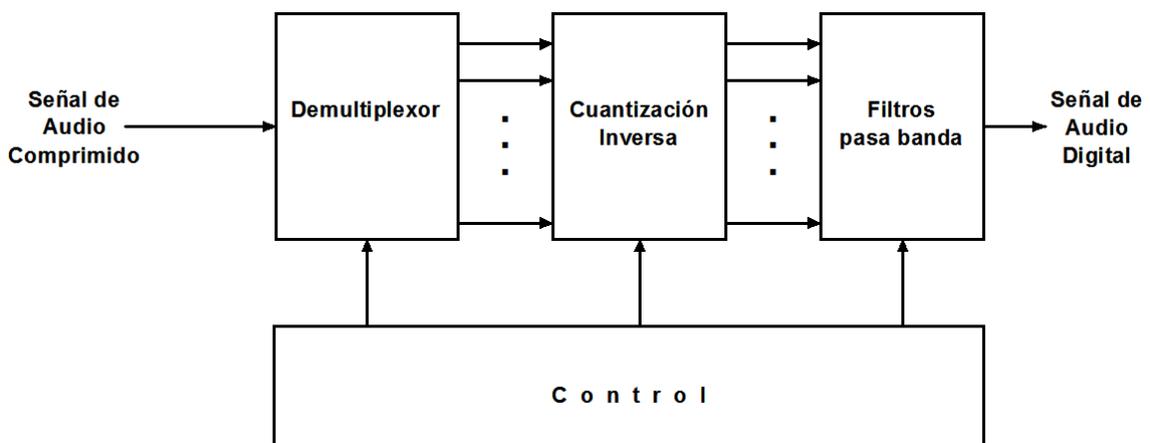


FIGURA 2. 12 DECODIFICACIÓN DE SUB-BANDAS

Codificación por Transformada

En la codificación por transformada no se utilizan bancos de filtros si no que se le aplica señal de audio una transformada para convertirla al dominio de la frecuencia, como son la DFT (Discrete Fourier Transform) o la MDCT (Modified Discrete Cosine Transform). Con esta transformación se obtienen 256 o 512 valores de potencia espectral. Los valores de potencia espectral se cuantizan de nuevo. El cuantizador es más fino, menos fino o suprime el valor de acuerdo a la modelo psicoacústico que lo controla. El bloque del modelo psicoacústico es alimentado por la seña transformada mediante FFT (Fast Fourier Transform). Este tipo de codificación se utiliza en el estándar Dolby Digital AC-3. La principal ventaja de la codificación por transformada es que proporciona una mayor resolución de frecuencia en el proceso de reducción de irrelevancia.

También se existe la codificación híbrida que mezcla ambos tipos de codificaciones como la que se utiliza en la capa III de MPEG [7].

2.3.2 ESTÁNDARES DE COMPRESIÓN DE AUDIO

Dolby Digital AC-3

En el estándar se indica que el sistema de compresión para el audio es el AC-3 con algunas restricciones. Este sistema codifica cinco canales de audio con ancho de banda completo (de 3Hz a 20KHz):

1. El canal izquierdo
2. El canal derecho
3. El canal del centro
4. El canal de envolvente izquierda
5. El canal de envolvente derecha

Y un canal de realce de bajas frecuencias (LFE, Low-frequency Enhancement) de ancho de banda reducido, ya que su ancho de banda es de la décima parte del ancho de banda de los otros canales (de 3Hz a 120Hz). Todos los canales se comprimen para formar un flujo de datos de 384kbps. También se le conoce como 5.1 o 5 + 1 por los canales que codifica.

Este estándar fue desarrollado por los laboratorios Dolby en los Estados Unidos al mismo tiempo que se desarrolló el estándar MPEG para audio. También es conocido como Dolby digital o Dolby D. Fue establecido en 1990 y la primera vez que se utilizó de manera que el público lo pudiera apreciar fue en diciembre de 1991 para el audio de una película. Además de utilizarse en los sistemas ATSC, también se utiliza en películas y DVDs. El estándar se basa en la transformada del coseno discreto modificada (MDCT).

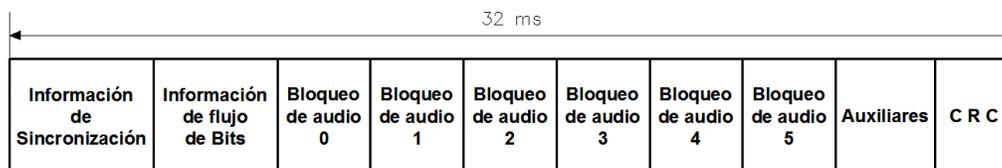


FIGURA 2. 13 TRAMA AC-3

Los 6 canales se muestrean 48KHz, usando 20 bits por muestra, lo que implica un flujo de 960 Kb/s por canal en señales PCM, junto con información de control, de referencia y códigos de tiempo, es decir que en total el flujo de entrada es de aproximadamente 6 Mb/s y con la compresión la tasa de bits de salida es de 32 a 640kb/s, pero en ATSC se limita a 384kb/s, esto es un reducción de 15:1. AC-3 también acepta tasas de muestreo de 32kHz y 47.1kHz.

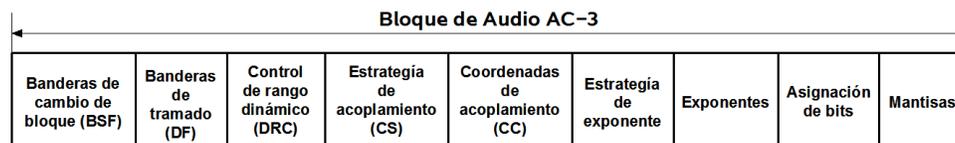


FIGURA 2. 14 BLOQUE DE AUDIO AC-3

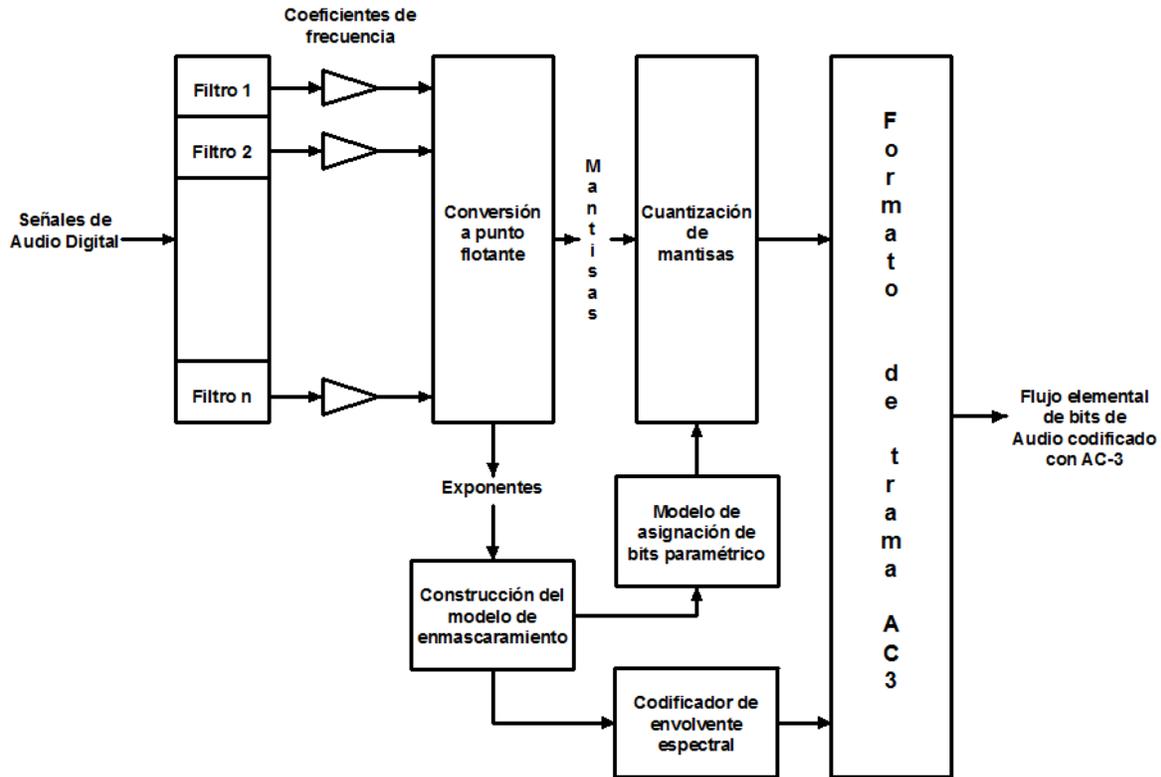


FIGURA 2. 15 DIAGRAMA DE BLOQUES DEL CODIFICADOR AC-3

El flujo de bits de entrada pasa a la primera etapa en la codificación AC-3 donde se transforman del dominio del tiempo al de la frecuencia usando un banco de filtros MDCT de 512 puntos, que tienen una resolución en la frecuencia de 93.75Hz para frecuencias de muestreo de 48kHz. La resolución adaptativa frecuencia a temporal se logra cambiando a una transformada de 356 puntos cuando existen cambios muy rápidos en la señal de entrada. En el siguiente bloque se realiza un proceso de punto flotante mediante el cual cambia los coeficientes de la compresión DCT a pares de exponentes y dígitos significativos, también llamados mantisa, es decir si el número flotante era 22.33 lo separa en un número significativo 2233 y un exponente 10^2 . Los números significativos se codifican con una cantidad de bits variable, utilizando un modelo paramétrico de asignación de bits, que se basa en los principios del enmascaramiento psicoacústico. Utilizando los principios del enmascaramiento psicoacústico determina el número de bits

para cada mantisa según la frecuencia. Los datos codificados de la envolvente espectral y los datos de la mantisa cuantizados correspondientes a 6 bloques de audio se incluyen en el formato de una trama AC-3 de sincronización. Se usa un algoritmo para que las informaciones idénticas en los canales del audio se codifiquen, es decir, que se extraen las similitudes de los canales, y se distribuya a lo largo de canales similares en la etapa de descompresión.

Los flujos de bits de AC-3 se componen de tramas de sincronización, y a su vez las tramas de sincronización se forman de seis bloques de audio que corresponden a los 6 canales de audio. Cada bloque de audio representa 256 muestras de audio y esto significa que cada trama contiene 1536 muestras de audio, además de la trama contiene al final un bloque auxiliar. Este bloque auxiliar en cada trama se reserva para información del control y del estado de la transmisión.

Las características más importantes de la compresión AC-3 son que se pueden utilizar 256 o 128 sub-bandas, que tiene un factor de escala de 4.5 bits por banda en un rango de 144 dB, tiene una asignación de bits dinámica tanto hacia adelante como hacia atrás, tiene un volumen uniforme de manera que puede reducir los cambios en volumen cuando se cambia de un canal de televisión a otro o entre programas y tiene un control del rango de dinámico.

Un bloque además de contener los exponentes, las asignaciones de bits y las mantisas resultantes de la transformación, contiene los siguientes bits y banderas con información de control:

- BSF: banderas para el intercambio de bloques
- DF: banderas de decisión.
- DRC: control del rango dinámico, permite controlar el rango del volumen y optimizar el rango dinámico. Se necesita debido a que las capacidades del

producto y las condiciones para oír la transmisión varían, también porque el decodificador necesita protección contra picos en especial al convertir la transmisión a modo estéreo o mono.

- CS: estrategia de acoplamiento.
- CC: coordenadas de acoplamiento.
- ES: estrategia de los exponentes.

Esta información le sirve al decodificador para saber cómo procesar la información de audio es conocida como metadatos. Aunque generalmente es estática, regularmente cambia con el formato del programa.

Otras características del sistema es que permite servicios auxiliares basados en texto, información de la guía de programación en la TV y un sistema de control de la información para permitir la interoperabilidad con la televisión restringida (cable, satelital, etc.). Características del transporte en el sistema después de la compresión MPEG-2 de video y AC-3 de audio, las señales de audio, video y señales auxiliares se multiplican en paquetes de longitud fija de 188 bytes. Este empaquetamiento se basa en la capa de transporte de MPEG-2, cada paquete de 188 bytes tiene una cabecera de 4 bytes, donde el primer es de sincronización, por lo que la carga real es de 184 bytes, sin embargo, dentro de esta carga puede haber una cabecera de adaptación.

MPEG-2 Capa 2

En el estándar de compresión de audio MPEG-2 capa 2, las señales de audio se analizan espectralmente cada cierto tiempo (por bloques) mediante la transformada FFT a 1024 muestras, pero no cubre completamente una trama de audio de 1152 muestras, con este análisis se realiza una curva del umbral de audibilidad enmascarada (UAE) de dicho sonido. Del UAE se determinan las frecuencias enmascaradores y se eliminan o reducen las demás mediante un escalador y cuantizador, para detectar estas frecuencias se utiliza un conjunto de 32 filtros que cubren de los 20 Hz a los 20 kHz, con sub-bandas del mismo

tamaño. El codificador de audio MPEG-2 capa 2 en cada canal utiliza tramas de longitud constante para facilitar la decodificación, específicamente de 1152 muestras.

Cabecera	C R C	Asignación de bits	Información de selección del factor de escala	Factores de Escala	Muestras	Datos Auxiliares
----------	-------	--------------------	---	--------------------	----------	------------------

FIGURA 2. 16 TRAMA DE AUDIO MPEG-2 CAPA 2

Mediante los codificadores de audio MPEG-2 capa 2 se obtienen tasas de bits que van de los 32 a los 384kbps. Se utilizan factores de escala por banda de 6 bits, diferentes para cada sub-banda. El formato de la trama de este esquema de codificación se muestra en la figura 2.16. Al igual que en la codificación AC-3, en la codificación de audio MPEG capa 2 se pueden codificar varias señales de audio

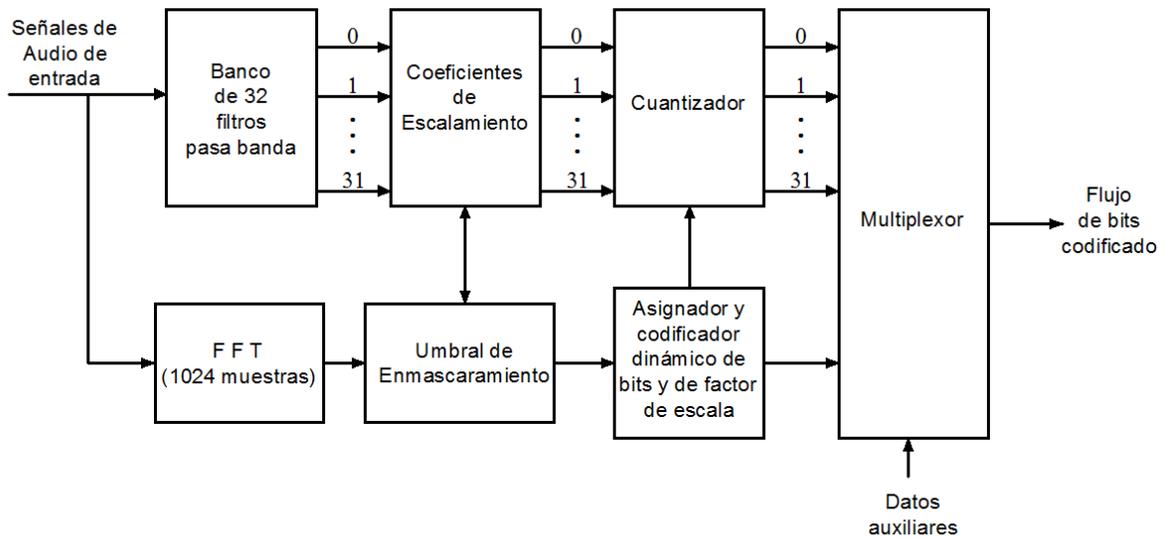


FIGURA 2. 17 CODIFICADOR DE AUDIO MPEG-2 CAPA 2

Codificación de Audio Avanzada

El esquema de Codificación de Audio Avanzada o AAC (Advance Audio Coding) fue publicado como la parte 7 del estándar MPEG-2. Los codificadores AAC tienen la misma

estructura que los de codificación de audio MPEG-2 capa 2, pero a diferencia de éstos aceptan señales con tasas de muestreo entre 8 y 96 kHz, puede codificar hasta 48 canales de audio, la longitud de las tramas que codifica es variable, usa bloques de 960 o de 1024 muestras, puede usar la función de ventana derivada de Kaiser-Bessel.

3. CODIFICACIÓN DE CANAL.

La codificación de canal consiste en agregar bits de redundancia al flujo de bits, para poder corregir los errores que presente debido al ruido que se le agrega en el canal de comunicaciones, y en modular la señal digital para su transmisión.

3.1 CORRECCIÓN DE ERRORES HACIA ADELANTE

Los tipos de corrección de errores donde la corrección se realiza en el receptor sin notificar ni pedir una retransmisión al emisor son conocidos como FEC (Forward Error Correction) o Corrección de Errores hacia Adelante. Este esquema es muy utilizado en las transmisiones inalámbricas omnidireccionales, como lo es la televisión terrestre, debido a que no hay forma de que el televidente pida una retransmisión. En seguida se describen algunos tipos de codificaciones FEC.

3.1.1 CODIFICACIÓN POR BLOQUES LINEALES

En los códigos correctores de errores a los bits de la señal se le considera como un cuerpo de Galois Binario o $GF(2)$ donde están definidas las operaciones de suma y de producto entero módulo 2. Para el caso de los códigos de bloque, un bloque de bits se considera un espacio vectorial sobre $GF(2)$ y se utilizan las operaciones con vectores y matrices.

Un código de corrección y detección de errores es un código de bloques lineal cuando cumple con las condiciones que se describen enseguida:

- El conjunto de las 2^k combinaciones corresponde a un subconjunto en las posibles 2^n combinaciones por lo que un bloque de k bits se convierte en un bloque de n bits después de ser codificado.
- Una de sus palabras de código tiene todos los bits 0.
- Cualquiera de las palabras pertenecientes al código es la combinación lineal de dos de las palabras de código.
- Las palabras del código poseen al menos otra palabra a una distancia mínima. La distancia entre dos palabras se calcula sumando dos palabras del código y obteniendo el peso del resultado.

MATRIZ GENERADORA

La matriz generadora (G) sirve para obtener las palabras de código (P) de un código por bloques lineal a partir de las palabras de datos (D):

$$[P] = [D] \cdot [G]$$

Donde la matriz generadora para un código sistemático debe cumplir con la siguiente relación:

$$[G] = \begin{bmatrix} 1 & 0 & \dots & 0 & g_{11} & g_{12} & \dots & g_{1(p-d)} \\ 0 & 1 & \dots & 0 & g_{21} & g_{22} & \dots & g_{2(p-d)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & g_{d1} & g_{d2} & \dots & g_{d(p-d)} \end{bmatrix} = [I_d \quad g]$$

MATRIZ DE COMPROBACIÓN DE PARIDAD

La matriz de comprobación de paridad [C] sirve para detectar si una palabra pertenece a un código por bloques lineal sistemático y tiene la siguiente forma:

$$[C] = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ g_{11} & g_{12} & \cdots & g_{1(p-d)} \\ g_{21} & g_{22} & \cdots & g_{2(p-d)} \\ \vdots & \vdots & \ddots & \vdots \\ g_{d1} & g_{d2} & \cdots & g_{d(p-d)} \end{bmatrix} = \begin{bmatrix} [d] \\ [g] \end{bmatrix}$$

SÍNDROME DE UNA PALABRA DE CÓDIGO

El síndrome de la palabra de código es el producto de la palabra (P) por la matriz de comprobación de paridad:

$$[S] = [P] \cdot [C]$$

Cuando la palabra no tiene errores el síndrome es la palabra nula, es decir que todos los bits del síndrome son 0.

Si la palabra contiene errores del síndrome es:

$$[S] = [P'] \cdot [C] = [E] \cdot [C]$$

Donde P' es la palabra con errores y E son los bits erróneos [2].

3.1.2 CÓDIGOS DE HAMMING

Si "r" es el número de bits de redundancia, los códigos de Hamming se caracterizan por que las filas de la matriz de comprobación son todas las combinaciones que pueden hacerse con r bits menos la palabra de solo ceros, por lo tanto:

$$p = 2^r - 1, \quad d = 2^r - 1 - r$$

Por lo tanto la tasa de código de Hamming es:

$$TC = \frac{d}{p} = \frac{2^r - 1 - r}{2^r - 1}$$

Y la de redundancia de código es:

$$RC = \frac{r}{d} = \frac{r}{2^r - 1 - r}$$

Los códigos de Hamming tienen una distancia mínima de tres por lo tanto sólo pueden corregir un error y detectar dos [2].

3.1.3 CÓDIGOS CÍCLICOS

Los códigos cíclicos son una clase de los códigos por bloques lineales, también conocidos como códigos de redundancia cíclica (CRC).

Pueden obtener las palabras de código de un código cíclico desplazando un lugar los bits de la palabra anterior, por lo que a partir de una palabra de P_1 de "p" bits se pueden formar las siguientes palabras de código:

$$\begin{array}{cccc} p_1 & p_2 & \cdots & p_p \\ p_p & p_1 & \cdots & p_{p-1} \\ \vdots & \vdots & \ddots & \vdots \\ p_2 & p_3 & \cdots & p_1 \end{array}$$

Los códigos cíclicos contienen un polinomio de grado "r" llamado polinomio generador $g(x)$, a partir del cual se pueden generar todas las palabras del código multiplicándolo por todos los polinomios $b(x)$ de grado máximo $k-1$ formado por las palabras de datos. El grado del polinomio generador es $r=n-k$ donde n es la longitud de las palabras codificadas y k es la longitud de las palabras de datos.

Para el caso de los códigos cíclicos sistemáticos, las palabras de código se definen como:

$$c(x) = b(x)x^{n-k} + d(x)$$

Donde $d(x)$ es igual a:

$$d(x) = - \left((b(x)x^{n-k}) \right)_{g(x)}$$

De igual forma que con la matriz generadora, para el polinomio generador existe un polinomio de comprobación de paridad $h(x)$ de tal forma que:

$$g(x)h(x) = x^n - 1$$

3.1.4 CÓDIGOS BCH

Para la construcción del polinomio generador con el menor grado posible y que garantiza una distancia mínima para un bloque se utiliza el teorema BCH (nombrado en honor a sus autores Bose, Chaudhuri y Hocquenghem), los códigos que generan estos polinomios se conocen como códigos BCH.

Para construir el polinomio generador se debe factorizar x^n-1 , como:

$$x^n - 1 = (x - \alpha_1) \dots (x - \alpha_n)$$

Y $g(x)$ es el resultado del producto de $n-k$ monomios $(x-\alpha_i)$ haciendo máxima la distancia mínima del código, con el resto de los monomios que factorizan x^n-1 se forma $h(x)$. En general las raíces son $\alpha^d = \alpha^{(d)}_{(2^m-1)}$ de forma que:

$$x^n - 1 = (x - \alpha^0)(x - \alpha^1) \dots (x - \alpha^{2^m-2})$$

Si se construye un polinomio que tenga como raíces los elementos de la clase conjugada se obtiene un polinomio mínimo de la clase conjugada. Para construir una clase

conjugada se crea una secuencia de cuadros consecutivos: $\alpha^d, \alpha^{2d}, \alpha^{4d}, \dots$, hasta que se repita α^d .

La cota BCH establece que si el polinomio generador contiene $\delta-1$ raíces consecutivas de α , es decir que $g(\alpha^b) = g(\alpha^{b+1}) = \dots = g(\alpha^{b+\delta-2}) = 0$ para algún b , entonces la distancia mínima del código es mayor o igual a δ . Cuando $b=1$ son llamados códigos BCH en sentido estricto.

Los códigos BCH son una generalización de los códigos de Hamming, donde:

$$p = 2^r - 1$$

Donde p podrá ser diferente de r a diferencia los códigos de Hamming, de tal manera que pueden corregir varios errores por palabra de código [2], [8].

3.1.5 CÓDIGOS REED-SOLOMON

Los códigos Reed-Solomon (R-S) son códigos no binarios que pertenecen a la familia de los BCH caracterizados por las siguientes relaciones:

$$p = 2^b - 1$$

$$H_m = r + 1 = p - d + 1$$

Las expresiones anteriores “ p ” es el número de símbolos por palabra de código, “ d ” es el número de símbolos por palabra de datos, “ b ” ese número de bits por símbolo, “ r ” es el número de símbolos redundantes y H_m es la distancia de Hamming mínima.

En estos códigos se pueden corregir:

$$EC_{RS} = \frac{H_m - 1}{2} = \frac{r}{2} \text{ símbolos erróneos}$$

Los códigos Reed-Solomon representan por: (p,d,b) [2]

La distancia de Hamming mínima de estos códigos es:

$$H_m = 2^b - d$$

Los códigos Reed Solomon se construyen sobre cuerpos de Galois de orden 2^m con $n=2^m-1$ y donde los polinomios generadores se forman únicamente a partir de raíces consecutivas. Para estos códigos la distancia mínima es igual a $n-k+1$.

3.1.6 CODIFICACIÓN CONVOLUCIONAL

A diferencia de los códigos de bloque el codificador de los códigos convolucionales es un sistema con memoria y su salida es el resultado de una convolución, de ahí su nombre.

Existen varias formas de representar los códigos convolucionales. Pueden representarse como un conjunto de ecuaciones, por ejemplo un código con una tasa de $\frac{1}{2}$ se representa con las siguientes ecuaciones:

$$C_0[l] = B_0[l] + B_0[l - 1] + B_0[l - 3]$$

$$C_1[l] = B_0[l] + B_0[l - 1] + B_0[l - 2] + B_0[l - 3]$$

Aplicando a estas ecuaciones la transformada Z se pueden escribir como polinomios en D:

$$C_0(D) = B_0(D) + DB_0(D) + D^3B_0(D) = (D^3 + D + 1)B_0(D)$$

$$C_1(D) = B_0(D) + DB_0(D) + D^2B_0(D) + D^3B_0(D) = (D^3 + D^2 + D + 1)B_0(D)$$

Este sistema de ecuaciones también se puede representar de forma matricial:

$$C(D) = B(D)G(D) = B_0(D)[D^3 + D + 1 \quad D^3 + D^2 + D + 1]$$

Donde $G(D)$ es la matriz del código convolucional.

Los parámetros que definen los códigos convolucionales son la tasa “R”, la memoria total del código “ M_t ” (es el número de unidades de retardo del codificador) y la longitud de restricción “K”, que se define como la máxima longitud de la respuesta del codificador convolucional a una señal impulso como entrada (es igual a 1 más el máximo grado de polinomios que conforman la matriz del código convolucional). La tasa “R” se puede considerar una división entre k símbolos de entrada y n símbolos a la salida.

Al igual que para los códigos de bloque, para los códigos convolucionales se puede definir una matriz de comprobación de paridad “ $H(D)$ ” y un vector de síndrome.

Un código convolucional puede ser sistemático si su matriz generadora tiene la siguiente forma:

$$G(D) = [I_k \quad P(D)]$$

Esto significa que las primeras salidas son iguales a las entradas [8].

Representación de los Codificadores Convolucionales

Existen varias formas de representar un codificador convolucional de manera gráfica: el diagrama del sistema, el diagrama de árbol y el diagrama de estados. Antes de analizar cada uno es importante explicar el estado del codificador, se define como el contenido de todas las células de retardo del codificador y es igual a 2^{M_t} .

En la figura 3.1 se muestra el diagrama del sistema del codificador convolucional con tasa $\frac{1}{2}$ que se explicó en la sección anterior. Representa el flujo de bits que entran, las operaciones que se realizan sobre este flujo, ya sean retardo y/o sumas módulo 2, y los flujos de bits que se obtienen a la salida del codificador.

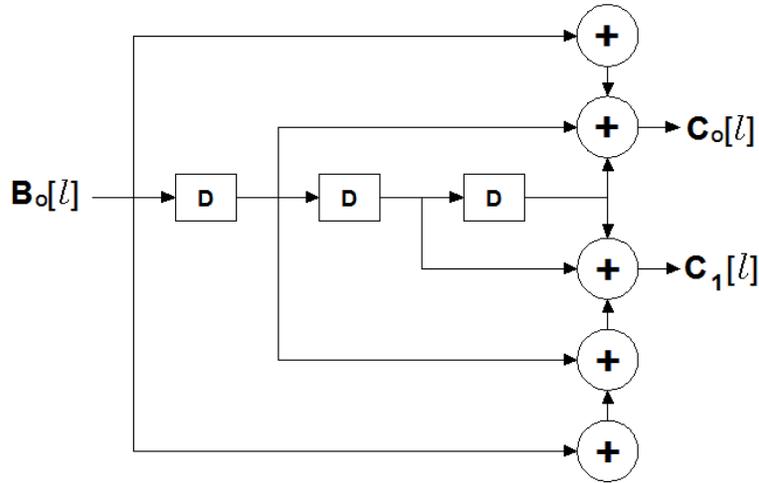


FIGURA 3. 1 DIAGRAMA DEL SISTEMA DEL CODIFICADOR CONVOLUCIONAL

El diagrama de estados para el mismo código se muestra en la figura 3.2. Los diagramas de estados son grafos dirigidos y etiquetados. Sobre las flechas se escribe el nuevo bit que entra en el codificador y la salida del codificador después de la diagonal, por otra parte en los círculos se escribe el estado de codificador.

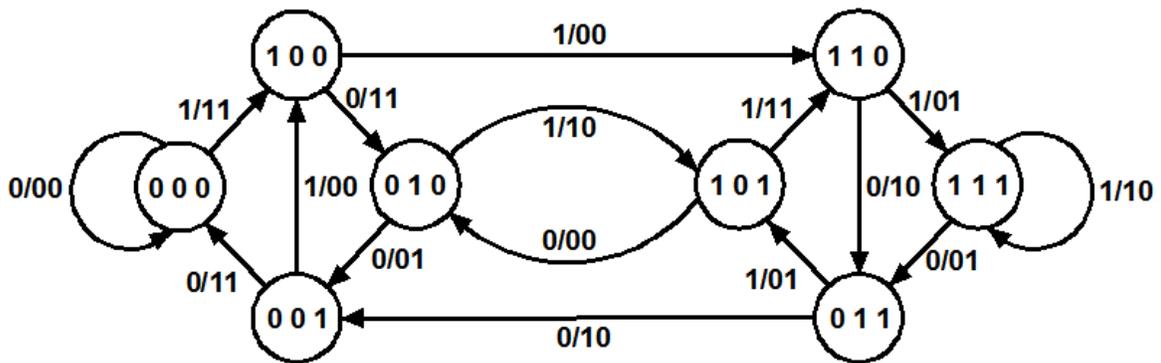


FIGURA 3. 2 DIAGRAMA DE ESTADOS

Para realizar el diagrama de árbol se contempla un estado inicial de solo ceros, para el caso del código que se ha utilizado para realizar los ejemplos esto significa 000, cada una de las dos ramas significa una entrada al codificador, la rama superior es una entrada al codificador de un bit 1 y la rama inferior es una entrada de un bit 0 sobre las ramas se

escriben el nuevo estado del codificador, una diagonal y la salida que genera el codificador. Se repite el proceso de dividir una rama en dos, a menos de que se regrese a un estado que se consiguió con anterioridad. En la figura 3.3 se muestra el diagrama de árbol para el mismo código que se ha utilizado para ejemplificar.

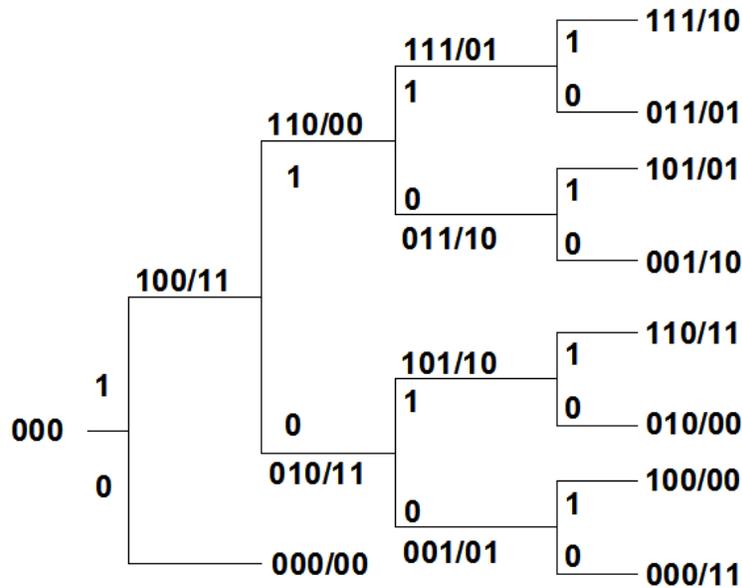


FIGURA 3. 3 DIAGRAMA DE ÁRBOL

Decodificación de los Códigos Convolucionales

Al ser sistemas con memoria los codificadores convolucionales, en el decodificador se detecta una secuencia completa en lugar de detectarse símbolo por símbolo. Se considera una secuencia de "L" símbolos generados a partir de kL símbolos a la entrada y nL símbolos a la salida transmitidos a través de un canal simétrico binario. El estado inicial y final del codificador se dan por conocidos y a partir de una secuencia "r" de n(L+K-2) bits recibidos se obtiene una secuencia "c" de L símbolos que signifique la mínima métrica de camino. Donde la métrica de camino es una suma de métricas de rama derivadas de la distancia de Hamming que permite el uso del algoritmo de Viterbi en el decodificador [8].

3.1.7 CODIFICADOR DE TRELIS

Codificador de Trellis, Modulación codificada en rejilla (TCM) o código de rejilla son los nombres con los que se le conoce a la técnica en la cual se seleccionan los símbolos de la constelación utilizada que se asignan como salida de cada rama de un codificador convolucional de forma que la distancia euclidiana sea máxima, son códigos específicamente para canales gaussianos digitales. Los codificadores de Trellis se componen de un codificador convolucional y de un modulador digital. [8]

3.1.8 INTERLEAVING

El interleaving o barajado de bits es un proceso donde se cambia el orden de una secuencia de bits para incrementar la eficiencia de un código de protección contra errores. Originalmente se propuso para evitar los errores en ráfagas pero también se utiliza para concatenar códigos y que aumente la eficiencia de la protección contra errores. Esto se debe a que los errores que se producen en ciertos canales de comunicaciones variantes en el tiempo no son independientes bit a bit y en algunos instantes la probabilidad de bit en error es superior a la probabilidad promedio. Por medio del interleaving los errores en ráfaga se pueden tratar con errores aislados como se puede ver en la figura 3.4.

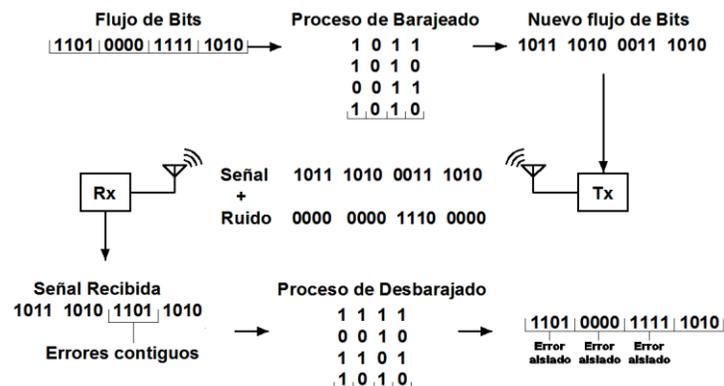


FIGURA 3. 4 PROCESO DE INTERLEAVING O BARAJADO

3.1.9 PUNCTURING

La técnica de puncturing o perforado consiste en descartar o eliminar bits en la salida del decodificador de acuerdo con un patrón cíclico. De forma que se reduce el número de bits a la salida del decodificador por cada bit de entrada. En el caso de los códigos convolucionales representa una ventaja, puesto que sin aumentar k ni modificar la complejidad del decodificador se obtienen tasas más próximas a 1. También se pueden utilizar diferentes patrones de eliminación para variar la tasa R y adaptarse a las condiciones del canal de transmisión, pero utilizando únicamente un codificador y un decodificador. [8]

3.1.10 CÓDIGOS CONCATENADOS

Utilizando los códigos concatenados se pueden obtener bajas tasas de bits error con poca redundancia de código. Con los códigos concatenados la información es codificada por dos códigos correctores de errores diferentes, como los explicados en las secciones anteriores. El código que tiene como entrada el flujo de bits sin codificar se le conoce como código externo o exterior, y el segundo código que se le aplica al flujo de bits se le conoce como código interno o interior. La principal ventaja de los códigos concatenados es la posibilidad de decodificarlos por separado con algoritmos de complejidad reducida.

En particular utilizando los códigos concatenados de forma serial compuestos por Reed-Solomon (código externo) y códigos convolucionales (código interno) y los intercaladores o interleaver entre ambos códigos (para disminuir el efecto de los errores en ráfaga) se pueden obtener eficiencias cercanas al límite de Shannon. Este esquema es el que se utiliza en los sistemas de televisión digital terrestre ATSC, ISDB-t y DVB-t, es decir los estándares más extendidos geográficamente.

3.2 MODULACIONES DIGITALES

Para transmitir señales a través de un medio físico generalmente se transforman a señales de voltaje o corriente, modificando la amplitud, la frecuencia o la fase de dichas señales es como se transforman las señales digitales o señales físicas. Estos procesos se conocen como modulaciones digitales y se describen a continuación.

3.2.1 MODULACIONES DE AMPLITUD

La modificación de la amplitud de las señales físicas para transmitir señales de comunicaciones fue uno de los primeros métodos utilizados sobre medios eléctricos como lo fue el telégrafo. La modulación más simple en amplitud se conoce como Modulación por Amplitud de Pulso o PAM, en esta los bits 1 se traducen al canal físico como un pulso de amplitud positiva y los bits 0 como un pulso de amplitud negativa. La constelación de una modulación digital se describe como el conjunto de posibles símbolos que corresponden a un bit o a un conjunto de bits. Por lo tanto la constelación para la PAM sería 1, -1. Derivadas de la modulación PAM existen un conjunto de modulaciones que agrupan varios bits para transformarlos en símbolos. Por ejemplo la modulación 4-PAM toma dos bits y los asocia con la constelación 2, 1, -1, -2. En la figura 3.5 se muestran ejemplos de estas modulaciones.

La modulación PAM es una modulación en banda base. Para convertirla en una modulación paso banda se puede multiplicar por una señal senoidal de frecuencia más elevada y filtrarla con un filtro paso banda, el resultado es un desplazamiento en el espectro y una modulación de banda lateral única (SSB) o de doble banda lateral (DSB) de acuerdo con la banda el filtro paso banda permita pasar. Sin embargo, la modulación

SSB solo se puede obtener utilizando un filtro ideal pues implica una caída muy abrupta, por lo que en la práctica en realidad se utiliza la modulación PAM-VSB o PAM en banda lateral vestigial que tiene una caída relajada y facilita su implementación pero resulta en un mayor ancho de banda. El estándar ATSC se utiliza la modulación 8VSB.

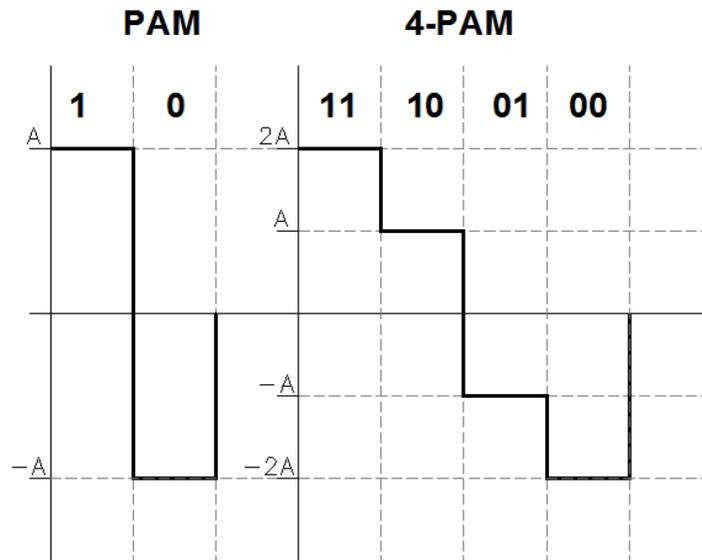


FIGURA 3. 5 MODULACIÓN PAM Y 4-PAM

Diagrama de Ojo

Los diagramas de ojo son una herramienta para visualizar el comportamiento de un sistema de comunicaciones. En estos diagramas se pueden visualizar fenómenos como el ruido, la interferencia entre símbolos o las fluctuaciones en el reloj de símbolo.

Modulaciones en Cuadratura

Una técnica para lograr una mejor eficiencia espectral consiste en modular simultáneamente dos señales reales con dos portadoras de la misma frecuencia pero desfasadas 90° entre sí. Este desfase significa que ambas portadoras están en cuadratura. Estas portadoras se conocen como $s_i(t)$ (componente en fase) y $s_q(t)$ (componente en cuadratura). Si estas portadoras modulan a dos señales PAM se crea la

modulación de amplitud en cuadratura o QAM. De acuerdo a la modulación PAM de estas señales se puede conseguir diferentes tipos de modulaciones QAM como son 16-QAM o 64-QAM. El número de bits por símbolo de esta modulación depende del número de símbolos de esta constelación. Si la modulación necesita P bits por símbolo utiliza una constelación de 2^P símbolos [8].

3.2.2 MODULACIONES DE FASE Y FRECUENCIA

La modulación en la fase de la portadora presenta la ventaja de una mayor inmunidad a las distorsiones no lineales (típicas de las etapas de amplificación en radiofrecuencia en saturación, limitadores de amplitud en los receptores y de permitir soluciones incoherentes).

La modulación por cambio de fase o PSK (Phase Shift Keying) se puede construir de la misma forma que la modulación QAM con dos portadoras desfasadas 90° , sin embargo, la constelación de símbolos de este tipo de modulaciones tiene un módulo constante y la información solo se transmite en la fase. La principal ventaja de este tipo de modulaciones es que cuando se combina con pulsos de transmisión rectangulares la señal resultante tiene envolvente constante y al estar la información en la fase también lo está en los cruces por cero, por eso es robusta ante distorsiones no lineales.

En la modulación PSK la distribución de símbolos es uniforme. Por lo que la diferencia de fases entre símbolos adyacentes para M-PSK es de $2\pi/M$ radianes.

La modulación por desplazamiento de frecuencia o FSK (Frequency Shift Keying) transmite la información variando la frecuencia, seleccionando la frecuencia de la portadora que se transmite durante un intervalo de símbolo de solo un conjunto discreto de frecuencias

Las principales ventajas de las modulaciones FSK son la sencillez y el bajo costo de implementación, además de su robustez ante las distorsiones que provocan los amplificadores en estado de saturación. Sin embargo, las desventajas que presentan son una baja eficiencia en ancho de banda y en potencia.

Las frecuencias que se seleccionan para la modulación pueden ser ortogonales, de forma que a la hora de detectarlas no se interfieran. A este tipo de modulación se le llama OFSK o Modulación por desplazamiento de frecuencias ortogonales.

3.2.3 MODULACIONES MULTIPORTADORA

Las modulaciones multiportadora utilizan un conjunto de pulsos que se generan multiplicando un filtro prototipo por un conjunto de N portadoras distintas. Un modulador de este tipo es equivalente a un multiplexor por división de frecuencias (FDM) como el que se muestra en la figura 3.6. Si la señal transmitida atraviesa un canal lineal no habrá interferencia entre símbolos transmitidos entre subcanales diferentes, aunque puede haber interferencia entre símbolos del mismo canal únicamente si no se cumple el criterio de Nyquist.

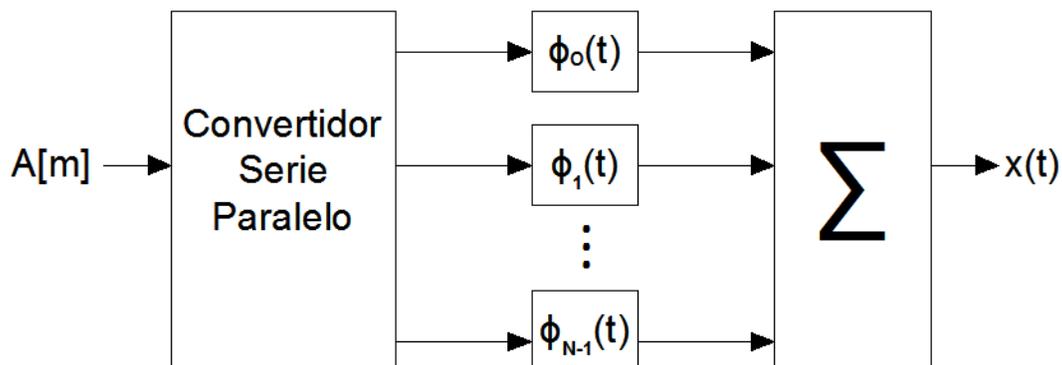


FIGURA 3. 6 MODULADOR FDM

Para poder construir este sistema FDM se necesita emplear filtros ideales, de lo contrario al utilizar bandas de guarda se reduce la eficiencia espectral. Por otra parte la complejidad del sistema es mayor puesto que se necesitan N filtros en el transmisor y N muestreadores operando síncronamente en el receptor. La Modulación OFDM representa una solución a estos problemas.

MODULACIÓN POR MULTIPLEXACIÓN DE FRECUENCIAS ORTOGONALES

La modulación por multiplexación de frecuencias ortogonales u OFDM resuelve el problema de los filtros ideales de FDM utilizando pulsos que manteniendo la ortogonalidad son realizables. El filtro prototipo que se utiliza es un pulso rectangular causal de duración T segundos. Las funciones base que utiliza OFDM siguiendo el esquema del modulador FDM en la figura 3.6 son de la forma:

$$\phi_k(t) = \frac{1}{\sqrt{T}} e^{j\frac{2\pi kt}{T}} \cdot \omega_T(t)$$

Donde $k=0, \dots, N-1$ y $\omega_T(t)$ es:

$$\omega_T(t) = \begin{cases} 1 & 0 \leq t < T \\ 0 & \text{para el resto} \end{cases}$$

El modulo al cuadrado de la transformada de Fourier de las funciones base es:

$$|\Phi_k(j\omega)|^2 = T \cdot \text{sinc}^2\left(\frac{\left(\omega - \frac{2\pi k}{T}\right)T}{2\pi}\right)$$

Donde $k=0, \dots, N-1$. La gráfica de estos símbolos para $N=16$ y $T=1$ se muestra en la figura 3.7 y se puede observar que se traslapan, sin embargo son ortogonales, además que son funciones infinitas por lo que los espectros también los son. Conociendo su espectro en el

tiempo la señal corresponde a un pulso triangular de ancho $2T$ segundos centrado en el origen.

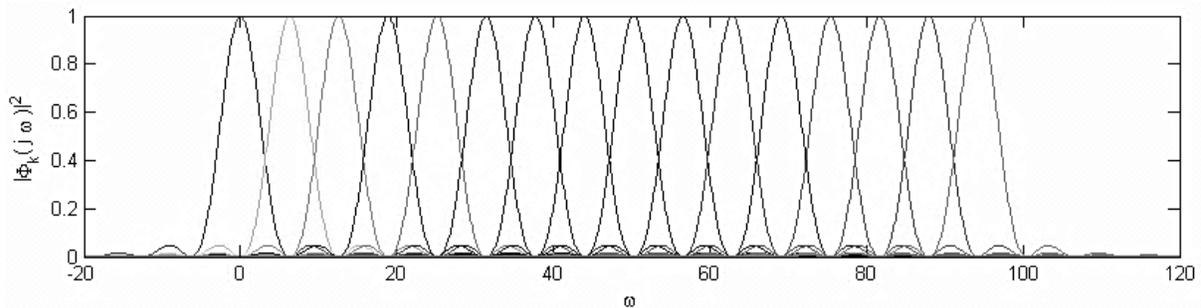


FIGURA 3. 7 GRÁFICA DE FUNCIONES BASE PARA $N=16$ Y $T=1$

Sin embargo la modulación OFDM necesita generar N portadoras complejas perfectamente enganchadas en fase de lo contrario existe interferencia entre portadoras (ICI), esto dificulta su implementación. Pero se puede generar una señal similar, sin los problemas de enganche de fase, generando una señal digital con periodo de muestreo T/N y aplicando la DFT (Discrete Fourier Transform) inversa a los símbolos A_k . De manera que la implementación de OFDM se realiza en tiempo discreto con IDFT como se muestra en la figura 3.8.

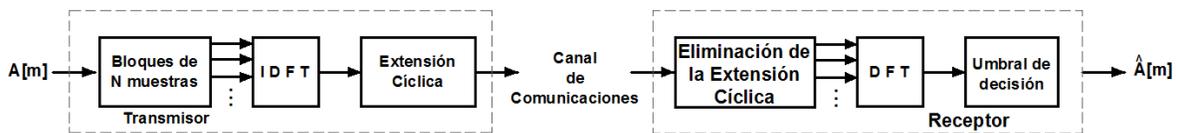


FIGURA 3. 8 DIAGRAMA A BLOQUES DE UNA TRANSMISIÓN OFDM

Una de las mayores ventajas de la modulación OFDM es su flexibilidad para conformar el espectro debido a que se puede actuar de forma independiente sobre cada portadora. De forma que en lugar de utilizar un filtro ideal se utilizan filtros con caídas menos abruptas, para facilitar la labor de los filtros de reconstrucción habitualmente se apagan las

portadoras situadas en frecuencias próximas a $\pm N\pi/T$ rad/seg y a pesar de que se reduce la velocidad de transmisión también se reduce el ancho de banda que ocupa la señal en la misma proporción y no se afecta la eficiencia espectral. Otra ventaja de OFDM es la posibilidad de aplicar algoritmos de carga de bits, para enviar más información en las portadoras con mejor relación señal a ruido.

La modulación OFDM también se puede explicar con un diagrama de bloques como el que se muestra en la figura 3.8. En este diagrama a la secuencia de símbolos a la entrada se le divide en bloques de N muestras, a cada bloque se aplica la transformación DFT inversa que resulta en N muestras, estas muestras se les añade un prefijo cíclico de M muestras antes de ser transmitidas secuencialmente. En el diagrama de bloques el canal discreto equivalente se denota $d[m]$ y una suma de ruido. En el receptor se elimina la extensión cíclica y se calcula la transformada DFT de N puntos a los bloques que se reciben y el resultado pasa al umbral de decisión.

La extensión cíclica se agrega para que el canal realice una convolución circular en lugar de una convolución lineal, ya que la convolución circular simplemente tiene un efecto multiplicativo en el dominio transformado de forma que no exista ICI. Además de que el prefijo cíclico impide que un bloque interfiera con el siguiente y esto significa que tampoco se presentara ISI en la transmisión. Ya que si la duración del prefijo es mayor que la longitud del canal el efecto de un bloque sobre otro se limita a corromper a su prefijo cíclico únicamente. Esta característica de ISI nula se puede lograr con cualquier guarda temporal pero solo la extensión cíclica logra eliminar la ICI. La desventaja de la extensión cíclica es que hace que las funciones base empleadas en el transmisor ya no sean ortogonales pero al quitar la extensión cíclica en el receptor las funciones vuelven a ser ortogonales. Otra desventaja es que al aumentar la duración temporal de las funciones base tipo $\text{sinc}(\)$, se vuelven más estrechas en frecuencia que antes y los máximos ya no

coinciden exactamente con los nulos de los demás y el espectro es rizado en vez de plano.

Al convertir el efecto convolucional del canal a un efecto multiplicativo, la implementación de igualadores lineales se vuelve muy sencilla, pero para ello se deben enviar símbolos piloto para estimar el canal o para construir un igualador adaptativo [8].

Los estándares ISDB-t, DVB-t y DTMB utilizan la modulación OFDM debido a su eficiencia espectral, únicamente ATSC no lo utiliza debido a que fue desarrollado antes de que vencieran las patentes de OFDM, a diferencia de los otros estándares.

4. RADIO DEFINIDO POR SOFTWARE

Hasta mediados de años 90 todos los equipos de comunicaciones se construían la flexibilidad de los dispositivos de comunicaciones era casi nula, pero con el desarrollo de los microprocesadores y la digitalización de las señales de comunicaciones se comenzó a realizar el procesamiento de la señal digital mediante software, sustituyendo a los moduladores, demoduladores, amplificadores, filtros, detectores y otros dispositivos contruidos con dispositivos electrónicos lo que implicó que un solo dispositivo pudiera llevar a cabo varias implementaciones de distintos protocolos.

El término de Software Radio fue acuñado por Joe Mitola en 1991 para referirse a las comunicaciones de radio reprogramables o reconfigurables. Donde la misma pieza de Hardware puede desarrollar diversas funciones en diferentes tiempos [9]. En la figura 4.1 se muestra el diagrama a bloques de un sistema Transmisor o Receptor definido por radio. Se puede ver que la modulación/demodulación y codificación/decodificación se realiza mediante software y la conversión analógica-digital o digital –analógica (según sea el caso) y el procesamiento en RF se implementan mediante hardware

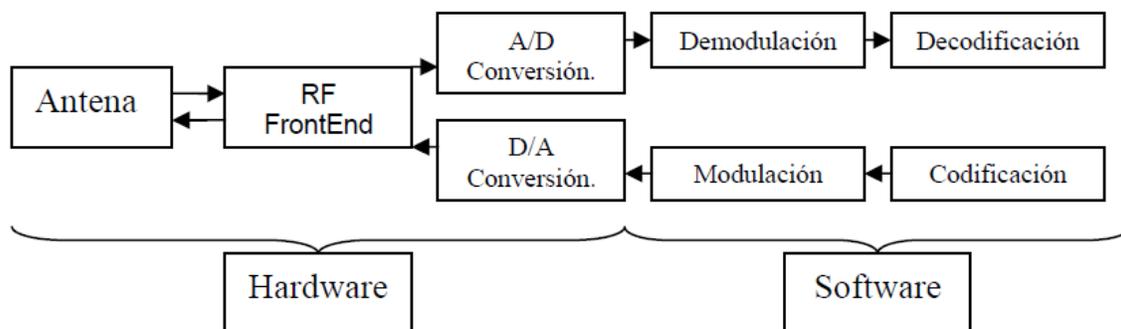


FIGURA 4. 1 DIAGRAMA A BLOQUES DE UN RADIO DEFINIDO POR SOFTWARE

En el ámbito comercial existen muchas implementaciones de radios definidos por software, por ejemplo en los teléfonos móviles la codificación se realiza por medio de software.

4.1 COMIENZO DE LOS RADIOS DEFINIDOS POR SOFTWARE

El primer proyecto donde se usó software para definir funciones de comunicaciones fue en el sistema conocido como Sistema de Comunicaciones de Navegación, Identificación y Aviónica Integrados (ICNIA, Integrated Communications Navigation, Identification and Avionics system). Éste sistema utilizó un DSP (Digital Signal Processor) para controlar una plataforma integrada para comunicaciones aéreas y se basó en un procesador GPP (General Purpose Processor) serie 6800 y fue desarrollado por la fuerza aérea de Estados Unidos a finales de la década de 1970 [10].

Sin embargo, la primera vez que se utilizó el término “Software Radio” por un equipo de la división de Garland Texas de la compañía E-Systems, misma empresa que tiempo después popularizo el Software Radio dentro de varias dependencias del gobierno de Estados Unidos en 1984 [11].

A finales de la década de los 80 el laboratorio de Investigación de la Fuerza Aérea de Estados Unidos inició el proyecto TAJPSP (Tactical Anti-Jam Programmable Signal Processor), desarrollando un procesador que fuera capaz de operar distintas formas de onda simultáneamente utilizando una aproximación militar.

Independientemente en 1991 Joe Mitola definió el término “Software Radio” cuando diseñó un plan para construir una estación base GSM. El mismo año Mitola describió los principios de la arquitectura del Software Radio sin los detalles de la implementación en el

artículo “Software Radio: Encuesta, Análisis Crítico y Direcciones Futuras”, que fue el primer artículo publicado en la revista de la IEEE que utilizó el término en 1992 [12].

4.2 GNU RADIO

GNU Radio es un software libre y de código abierto que proporciona herramientas para desarrollar radios definidos por software mediante un conjunto de bloques de procesamiento de señales. En conjunto con hardware de radiofrecuencia externo de bajo costo sirve para implementar radios definidos por software y sin este equipo se puede utilizar simplemente como un entorno de simulación.

El diseño de GNU Radio permite programar las conexiones entre bloques de procesamiento de señales mediante el lenguaje de programación Python, mientras que el procesamiento de señales se desarrolla mediante C++ utilizando un procesador de punto flotante. De manera que se puedan implementar sistemas de alto rendimiento en tiempo real en interfaces fáciles de usar.

GNU Radio es un proyecto GNU oficial desde 2001. Fue iniciado y ha sido sostenido con el financiamiento del filántropo John Gilmore a Eric Blossom por la creación del código y la administración del proyecto. En Septiembre de 2010 Eric Blossom fue sustituido como gerente por Tom Rondeau, que es graduado de Virginia Tech, experto de radio cognitivo y contribuyó en el proyecto de GNU Radio con anterioridad.

GNU Radio Companion es una interfaz gráfica para elabora conexiones entre los bloques de procesamiento de señales simplemente haciendo clic es la salida de un bloque y en la entrada del otro para conectarlas.

Entre los proyectos que se han desarrollado utilizando esta plataforma se encuentran:

WiFi Localization: software de localización sobre el protocolo 802.11

Simple_ra: aplicación de radioastronomía

UCLA ZigBee PHY: implementación del protocolo 802.15.4 con Mica2, MicaZ y Telos B.

Gr-bluetooth: decodificador Bluetooth de banda base para análisis de paquetes.

Gen 2 RFID: sistema de monitoreo Gen 2 RFID a través de un subsistema USB para reducir la latencia.

Logitech27MHzTransceiver: transceptor para teclados inalámbricos Logitech a 27 MHz.

OpenLTE: especificación de la capa física del enlace de bajada según las especificaciones LTE 3GPP.

4.2.1 INSTALACIÓN DE GNU RADIO

Existen varias formas de instalar GNU radio en diferentes sistemas operativos pero en esta tesis únicamente se explicará cómo instalarlo a partir de los precompilados de la compañía Ettus Research en Ubuntu 10.04, mediante los siguientes pasos:

1. Instalar las dependencias necesarias, abriendo una terminal y tecleando el siguiente comando:

```
$ sudo apt-get -y install libfontconfig1-dev libxrender-dev libpulse-dev \  
swig g++ automake autoconf libtool python-dev libfftw3-dev \  
libcppunit-dev libboost-all-dev libusb-dev fort77 sdcc sdcc-libraries \  
libsdl1.2-dev python-wxgtk2.8 git-core guile-1.8-dev \  
libqt4-dev python-numpy ccache python-opengl libgsl0-dev \  
python-cheetah python-lxml doxygen qt4-dev-tools \  

```

```
libqwt5-qt4-dev libqwtplot3d-qt4-dev pyqt4-dev-tools python-qwt5-qt4
```

2. Descargar los precompilados de GNU Radio y del driver universal del USRP (UHD) de <http://files.ettus.com/binaries/>

3. Instalar los paquetes desde una terminal mediante los comandos:

```
$ dpkg install uhd_003.004.000-release_Ubuntu-10.04-i686.deb
```

```
$ dpkg install gnuradio_3.6.0_Ubuntu-10.04-i686.deb
```

4. Establecer PYTHONPATH y LD_LIBRARY_PATH como variables de entorno y exportar con los siguientes comandos:

```
$ export PYTHONPATH=$PYTHONPATH:/usr/lib/python2.7/dist-packages
```

```
$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/bin
```

Además de instalar el software GNU Radio se pueden instalar extensiones escritas por personas que no pertenecen a GNU Radio. CGRAN (Comprehensive GNU Radio Archive Network) es un repositorio de código abierto para aplicaciones de GNU Radio que no son apoyadas oficialmente por GNU Radio. Los proyectos de CGRAN abarcan aplicaciones, codificadores y decodificadores de protocolos específicos, herramientas de plataformas y herramientas de medición. Particularmente la caja de herramientas de codificación de canal (Channel Coding Toolbox) de CGRAN se utiliza en el desarrollo de las prácticas diseñadas en esta tesis. En seguida se describen los pasos necesarios a seguir para instalar la extensión en Ubuntu:

1. Abrir una terminal y teclear el siguiente comando para obtener el código de la extensión:

```
$ svn co https://www.cgran.org/svn/projects/channel_coding/trunk/gr-chancoding
```

2. Cambiar de directorio con el siguiente comando:

```
$ cd gr-chancoding
```

3. Utilizar los siguientes comandos para instalar la extensión:

```
$ ./bootstrap
$ ./configure
$ make
$ make check
$ sudo make install
$ sudo ldconfig
```

Otra extensión que se utiliza para el desarrollo de estas prácticas se conoce como gr-baz.

Se deben seguir los siguientes pasos para instalarla:

1. Teclear el comando siguiente en una terminal para descargar el código de la extensión osmocon:

```
$ git clone git://git.osmocom.org/rtl-sdr.git
```

2. Utilizar los siguientes comandos para instalar la extensión osmocon:

```
$ cd rtl-sdr/
$ mkdir build
$ cd build
$ cmake ../
$ make
$ sudo make install
$ sudo ldconfig
$ cmake ../ -DINSTALL_UDEV_RULES=ON
$ cd rtl-sdr/
$ autoreconf -i
$ ./configure
$ make
$ sudo make install
```

```
$ sudo ldconfig
```

```
$ sudo make install-udev-rules
```

3. Teclear la siguiente instrucción para descargar el código de la extensión gr-baz:

```
$ git clone https://github.com/balint256/gr-baz
```

4. Utilizar las siguientes instrucciones para instalar gr-baz:

```
$ cd gr-baz
```

```
$ sh bootstrap
```

```
$ ./configure
```

```
$ make
```

```
$ sudo make install
```

```
$ sudo ldconfig
```

Para poder utilizar los bloques de las extensiones se deben seguir los siguientes pasos:

1. Desde una terminal crear el directorio donde se guardan las rutas a los bloques personales:

```
$ sudo mkdir .gnuradio
```

2. Entrar en el directorio:

```
$ cd .gnuradio
```

3. Crear el archivo de configuración con el siguiente comando:

```
$ touch config.conf
```

4. Editar el archivo con el comando:

```
$ gedit config.conf
```

que despliega una ventana para editar el archivo, copiar el texto siguiente y guardar el archivo:

```
[grc]  
local_blocks_path=/home/ubuntu/gr-chancoding/grc:/home/ubuntu/gr-  
baz/grc:/home/ubuntu/myblocks
```

Por último, algunas de las simulaciones necesitan librerías extra de python que se pueden instalar siguiendo los siguientes pasos:

1. Desde una terminal instalar las librerías con la siguiente instrucción:

```
$ sudo apt-get install python-matplotlib python-scipy
```
2. Antes de instalar preguntará si queremos instalar y se debe teclear “Y” y presionar enter. En ocasiones puede pedir la contraseña del administrador para realizar la instalación.

4.2.2 HERRAMIENTAS VIRTUALES PARA LAS SIMULACIONES EN GNU RADIO

En GNU Radio existen un grupo de herramientas para manejar el procesamiento de señales y para realizar simulaciones enseguida se describen los bloques que se utilizan en el software GNU Radio Companion para manejar algunas de ellas.

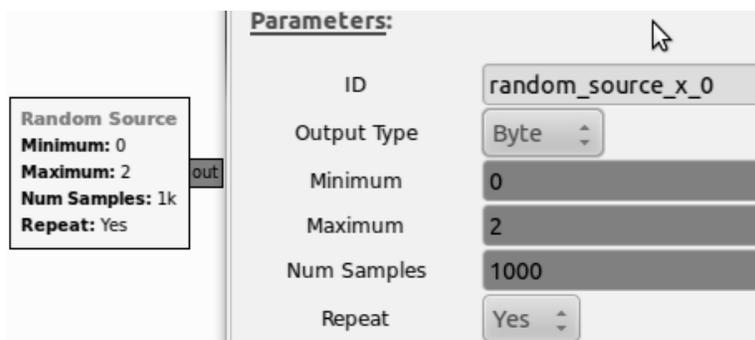


FIGURA 4. 2 BLOQUE FUENTE ALEATORIA Y SU CONFIGURACIÓN

El bloque fuente aleatoria que se muestra en figura 4.2 sirve para generar un flujo de datos de tipo byte, int o short. Se puede configurar este bloque para que la secuencia de datos que genera contenga datos dentro de un rango de valores y para que la secuencia no se repita hasta que llegue a un cierto número de muestras. En las prácticas que se desarrollaron en esta tesis este bloque se utiliza para generar secuencias aleatorias de bits para las simulaciones.

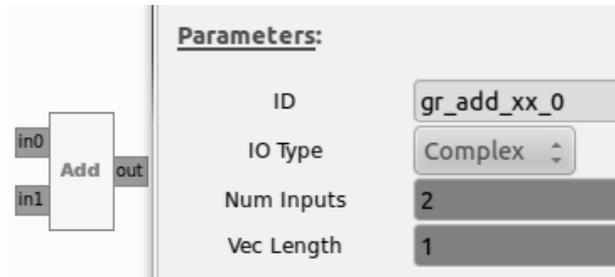


FIGURA 4. 3 BLOQUE SUMA Y SU CONFIGURACIÓN

En la figura 4.3 se muestra el bloque suma de señales que sirve para sumar dos o más señales que se generen durante la simulación, es importante que estas señales que se suman tengan el mismo tipo de dato y la misma longitud de vector.

Uno de los bloques más importantes en GNU Radio es el scope sink o sumidero osciloscopio, que funciona como un osciloscopio virtual y grafica en el tiempo las señales que recibe. Existen parámetros como la escala de voltaje y de tiempo que se pueden modificar cuando se ejecuta la simulación. Este bloque puede recibir dos tipos de datos complejos que grafica como dos señales distintas, una señal con los datos de la parte real y otra señal con los de la parte imaginaria, y flotantes que grafica como una sola señal. También se puede utilizar el bloque sumidero osciloscopio en el modo XY para observar la constelación de la señal, este modo únicamente se puede aplicar si se utilizan datos complejos.

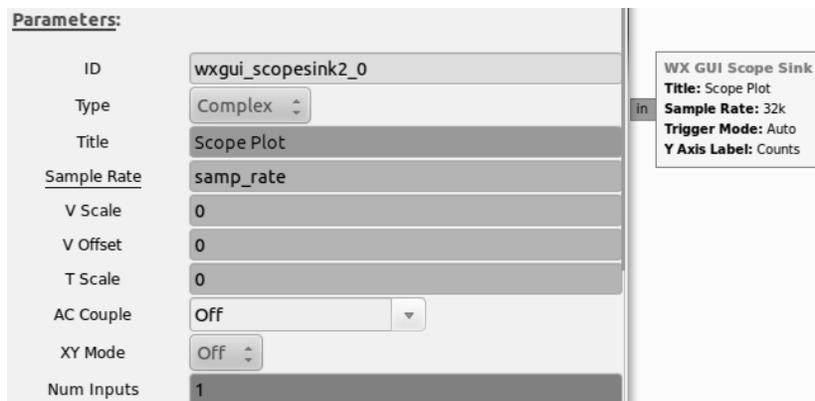


FIGURA 4. 4 BLOQUE SUMIDERO OSCILOSCOPIO Y SU CONFIGURACIÓN

Otro bloque que se utiliza en las simulaciones con fines didácticos es el bloque sumidero FFT. Este bloque aplica la transformada FFT a un bloque de la señal que recibe (El tamaño del bloque al que le aplica la transformación puede configurarse con el parámetro FFT Size). Durante la simulación este bloque funciona como un analizador de espectros para las señales que se generan. De forma que se puede visualizar el espectro de una señal durante la simulación y manejar esta herramienta para variar los decibeles por división que muestra, el nivel de referencia en decibeles, activar el modo promedio o el modo mantener picos.



FIGURA 4. 5 BLOQUE SUMIDERO FFT Y SU CONFIGURACIÓN

Otro bloque que sirve para analizar el espectro de una señal y sus cambios a través del tiempo es el bloque Waterfall que se muestra en la figura 4.6. Este bloque realiza una gráfica en tres dimensiones (Tiempo, frecuencia y potencia) el tiempo lo grafica en eje vertical, la frecuencia en el horizontal y la potencia mediante un código de colores que se muestra a la derecha al momento de la ejecución.

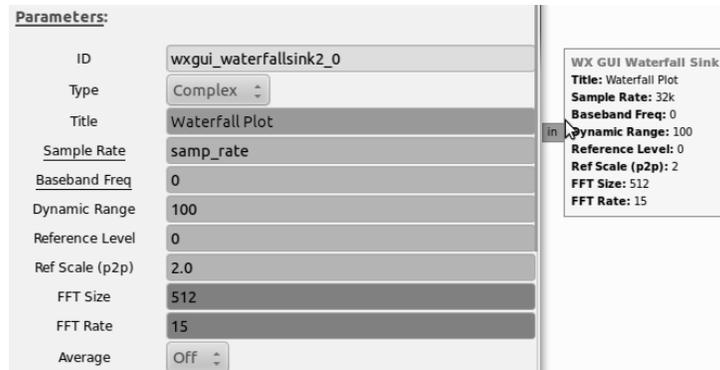


FIGURA 4. 6 BLOQUE SUMIDERO WATERFALL Y SU CONFIGURACIÓN

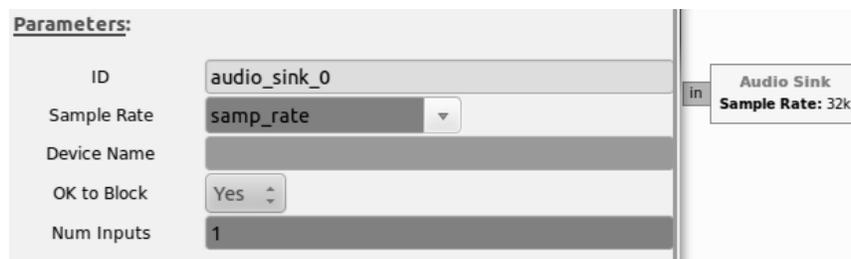


FIGURA 4. 7 BLOQUE SUMIDERO DE AUDIO Y SU CONFIGURACIÓN

El bloque sumidero de audio convierte la señal de la simulación en una señal de audio que reproduce mediante la tarjeta de audio y las bocinas de la computadora.

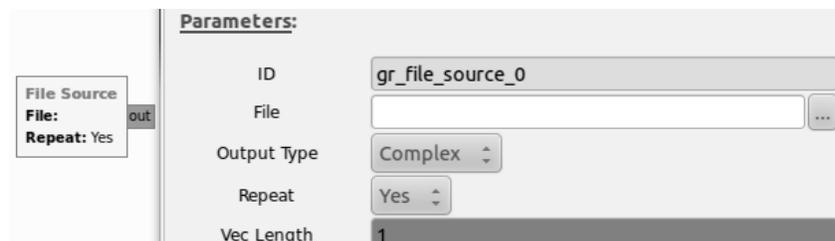


FIGURA 4. 8 BLOQUE FUENTE DE ARCHIVO Y SU CONFIGURACIÓN

Existen dos bloques que sirven para grabar una señal a un archivo y para tomar una señal grabada y utilizarla en una simulación, estos bloques son el bloque sumidero archivo y fuente de archivo respectivamente. Es importante configurar de manera adecuada el tipo de datos que se quieren capturar o que conforman el archivo que se utiliza, además de la

longitud del vector de la señal de datos y los nombres de archivos para que estos bloques funcionen de manera correcta.

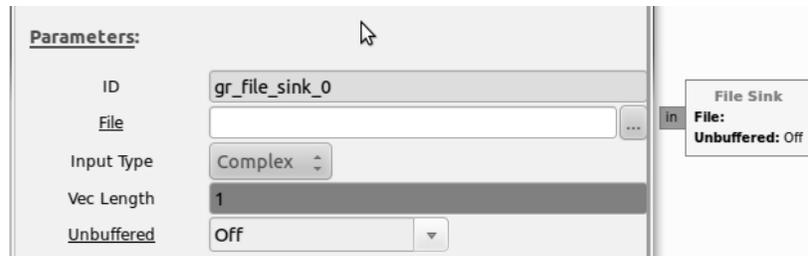


FIGURA 4. 9 BLOQUE SUMIDERO ARCHIVO Y SU CONFIGURACIÓN

4.2.3 MODULACIÓN EN AMPLITUD MEDIANTE GNU RADIO

La modulación para desplazar una señal de banda base a la frecuencia del canal donde se planea transmitir se realiza en el hardware externo de RF, del mismo modo que la modulación para desplazar de la frecuencia del canal de comunicaciones a banda base. Los dispositivos de RF más utilizados son los USRP (Universal Software Radio Peripheral). En el caso de los USRP las encargadas de la modulación son las daughterboard.

En GNU Radio se tienen los bloques sink y source del USRP donde se establece la frecuencia de transmisión o recepción de la señal como se ven en las figuras 4.10 y 4.11.

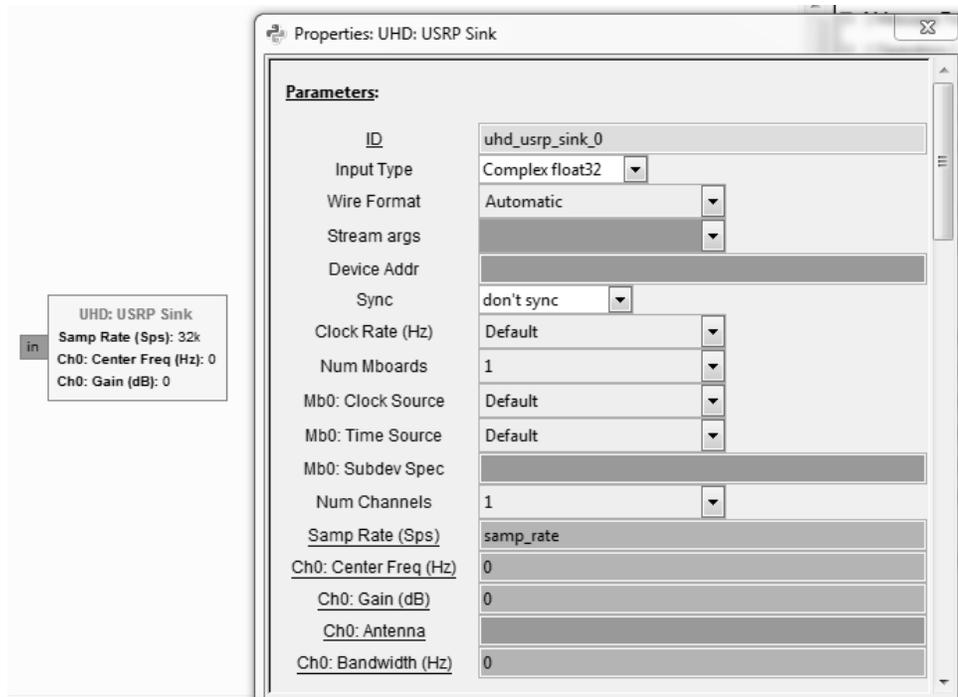


FIGURA 4. 10 SUMIDERO USRP Y SU CONFIGURACIÓN

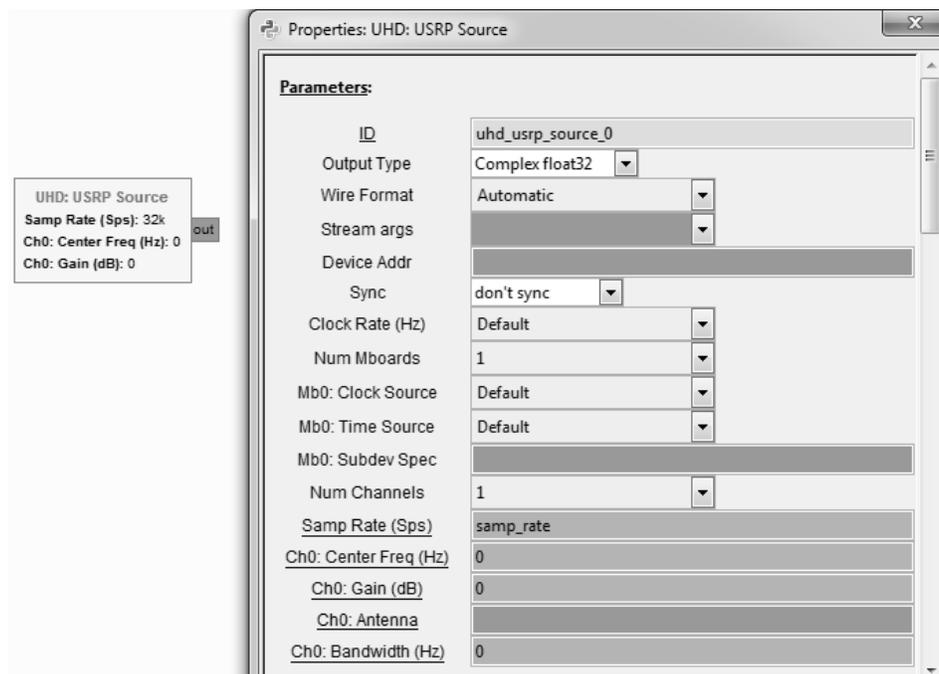


FIGURA 4. 11 FUENTE USRP Y CONFIGURACIÓN

4.2.4 SIMULACIÓN DEL CANAL DE COMUNICACIONES MEDIANTE GNU RADIO

En GNU Radio existen varios módulos y técnicas para simular el canal de comunicaciones, enseguida se describen algunas.

Una posibilidad es agregando ruido a la señal con un bloque de suma y una fuente de ruido. En la figura 4.12 se puede ver los bloques y sus conexiones en GNU Radio companion, además de una fuente de señal y un sumidero FFT que funciona como un analizador de frecuencias.

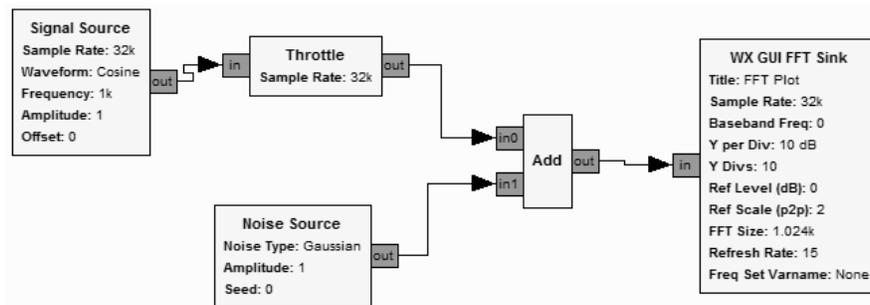


FIGURA 4. 12 SIMULACIÓN DEL CANAL DE COMUNICACIONES EN GNU RADIO COMPANION

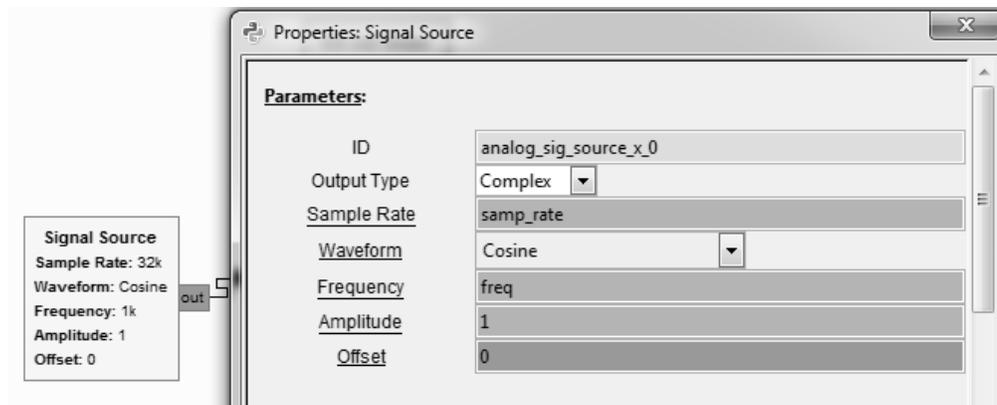


FIGURA 4. 13 FUENTE DE SEÑAL Y SU CONFIGURACIÓN EN GNU RADIO COMPANION

En el bloque de fuente de señal se puede configurar la tasa de muestreo, la forma de onda, la frecuencia de la señal, la amplitud y el nivel de Offset. La tasa muestreo se trata

del número de muestras por segundo puesto que se trata de una señal digital. La forma de onda se puede seleccionar entre una señal senoidal, cosenoidal, cuadrada, constante, triangular y dientes de sierra. La amplitud se establece en Volts, al igual que el Offset. Funciona como un generador de señales.

El bloque de Throttle sirve para que la computadora no utilice todos los recursos del procesador en la simulación y el bloque de suma (Add) sirve para sumar dos señales.

En el bloque de fuente de ruido se puede seleccionar el tipo de ruido entre uniforme, gaussian, laplaciano e impulso. La amplitud del ruido se especifica en Volts.

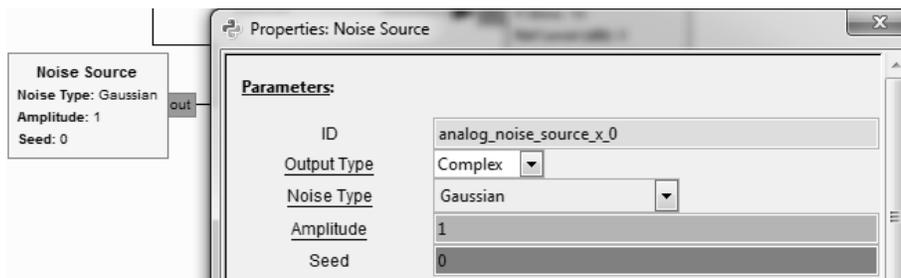


FIGURA 4. 14 BLOQUE FUENTE DE RUIDO Y SU CONFIGURACIÓN EN GNU RADIO COMPANION

Por último el bloque sumidero FFT funciona como un analizador de frecuencias virtual. La tasa de muestreo debe coincidir con la tasa de bits de la señal. El nivel de referencia y los decibeles por división se pueden ajustar cuando la simulación se encuentra en funcionamiento.

Otra forma de simular el canal de comunicaciones es mediante el bloque Binary Symmetric Channel. Este bloque sirve para simular un canal binario simétrico, en este bloque únicamente se puede configura el Bit Error Rate del canal y es de mucha ayuda cuando se busca probar el desempeño de los códigos correctores de errores.

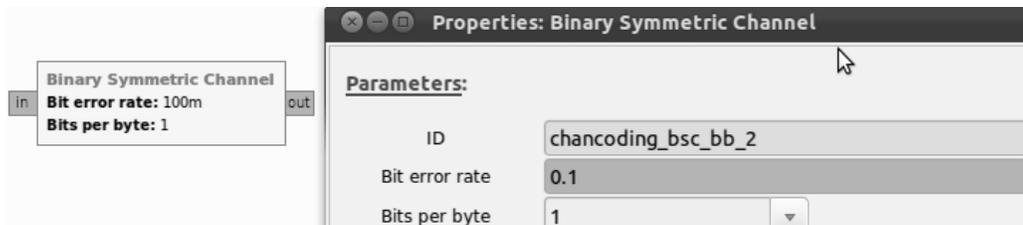


FIGURA 4. 15 BLOQUE BINARY SYMMETRIC CHANNEL Y SU CONFIGURACIÓN

El bloque de modelo de canal simula el canal de comunicaciones a partir de la permitividad dieléctrica del medio por el que viajaran las ondas electromagnéticas, el nivel de voltaje del ruido en la transmisión y la frecuencia doppler en la transmisión.

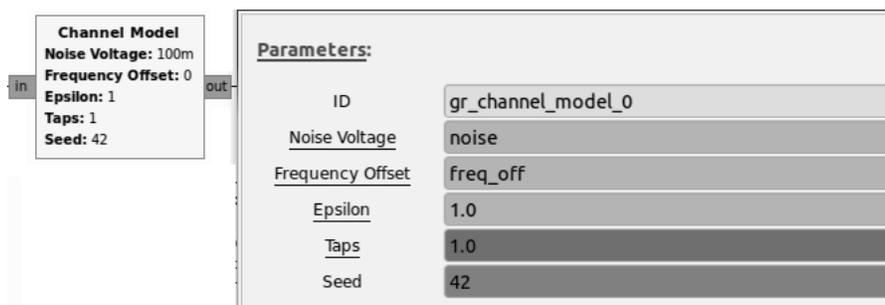


FIGURA 4. 16 BLOQUE DE MODELO DE CANAL Y SU CONFIGURACIÓN

4.2.5 MODULACIONES DIGITALES EN GNU RADIO

En GNU Radio existen varios módulos para aplicar las modulaciones digitales a un flujo de bits.

Para aplicar una modulación en fase a un flujo de bits se utiliza el módulo PSK Mod. En este bloque se configura el número de bits por símbolo y el número de puntos en la constelación que produce, además de tener la opción de realizar una codificación diferencial y/o el código Gray.

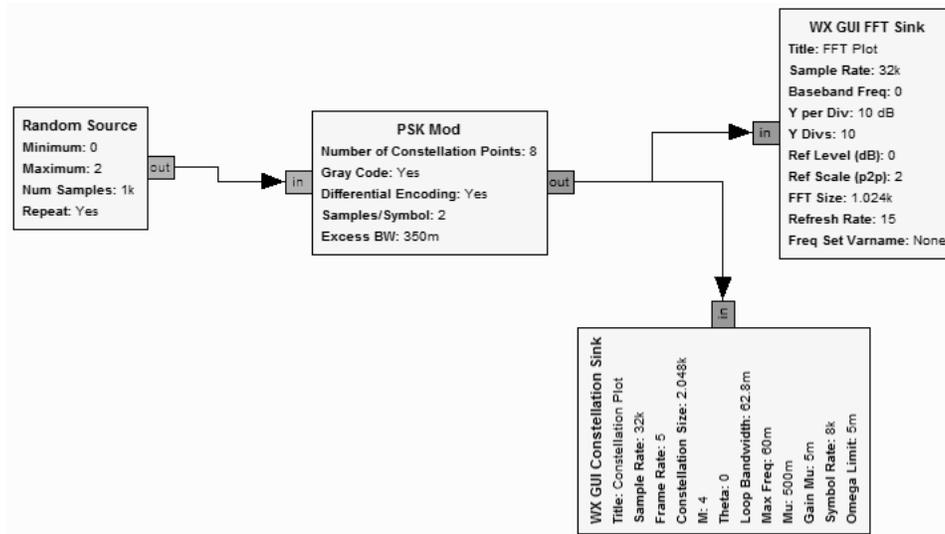


FIGURA 4. 17 MODULACIÓN PSK EN GNU RADIO COMPANION

En la figura 4.17 el bloque PSK Mod está conectado a una fuente aleatoria que genera un flujo de bits con una distribución de probabilidad uniforme, en él se puede establecer el número de muestras antes de que se repita el flujo de bits, además del rango de valores que se pueden tener y sí se repite o no cuando termine. Además está conectado a un bloque sumidero FFT, para analizar la señal en el dominio de la frecuencia, y a un sumidero de constelación, para analizar la constelación de la modulación.

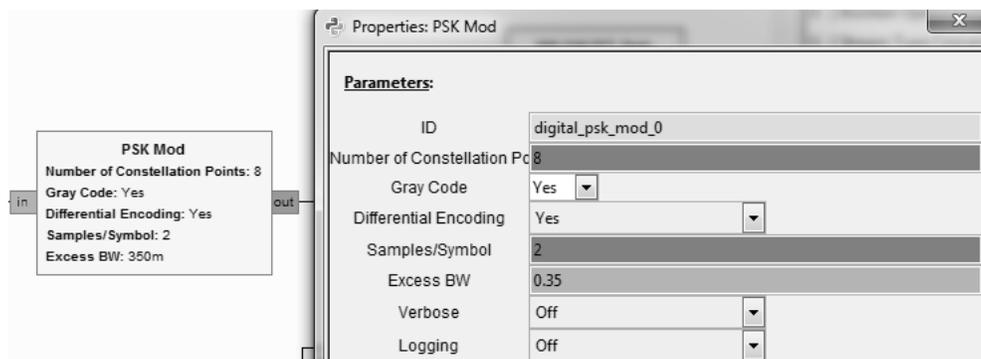


FIGURA 4. 18 BLOQUE MODULACIÓN PSK Y CONFIGURACIÓN

Las modulaciones QAM se pueden aplicar con el bloque QAM Mod donde se pueden configurar el número de bits por símbolo, si se utiliza codificación diferencial y/o codificación gray y el exceso de ancho de banda.

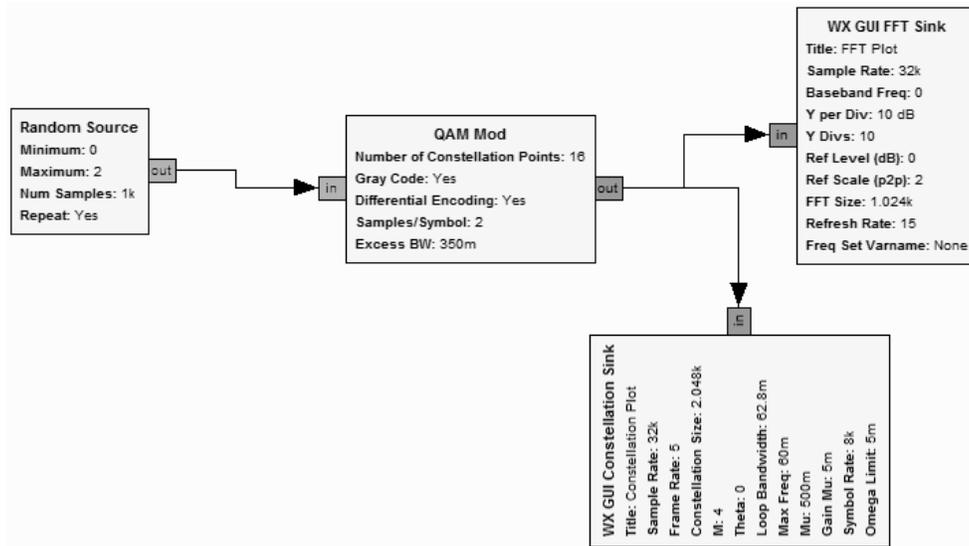


FIGURA 4. 19 MODULACIÓN QAM EN GNU RADIO COMPANION

Otros tipos de modulaciones digitales así como las modulaciones anteriores también se pueden realizar mediante el bloque Chunks to Symbols en este bloque se configura la constelación de la modulación y el número de bits por símbolo.

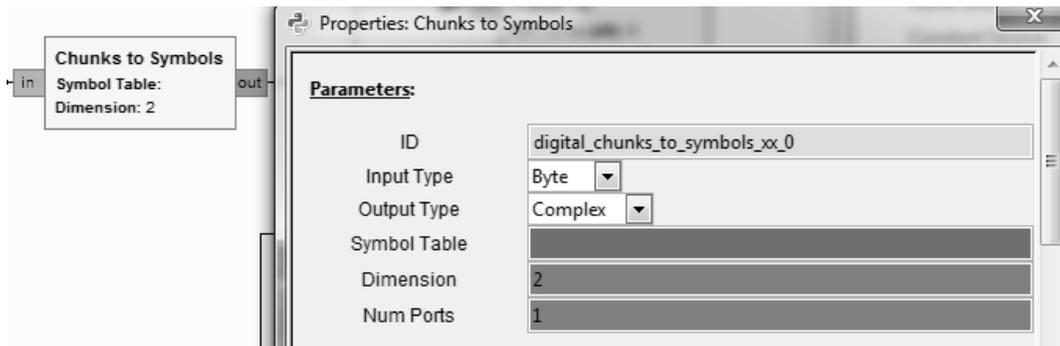


FIGURA 4. 20 BLOQUE CHUNKS TO SYMBOLS Y SU CONFIGURACIÓN

La salida del bloque Chunks to Symbols también pueden analizarse con un sumidero de constelación para verificar la constelación que se utiliza. Para decodificar la modulación que aplicar el bloque Chunks to Symbols se puede utilizar el bloque Decodificador de constelación que se muestra en la figura 4.21.



FIGURA 4. 21 BLOQUE DECODIFICADOR DE CONSTELACIÓN Y SU CONFIGURACIÓN

Otra modulación muy importante que se puede implementar o simular mediante GNU Radio Companion es la modulación OFDM, utilizando el bloque OFDM Mod que se muestra en la figura 4.22. En dicho bloque se puede configurar el tamaño de los bloques de transformada FFT, la longitud del prefijo cíclico que se ocupa en la modulación y la longitud de la carga de la modulación.

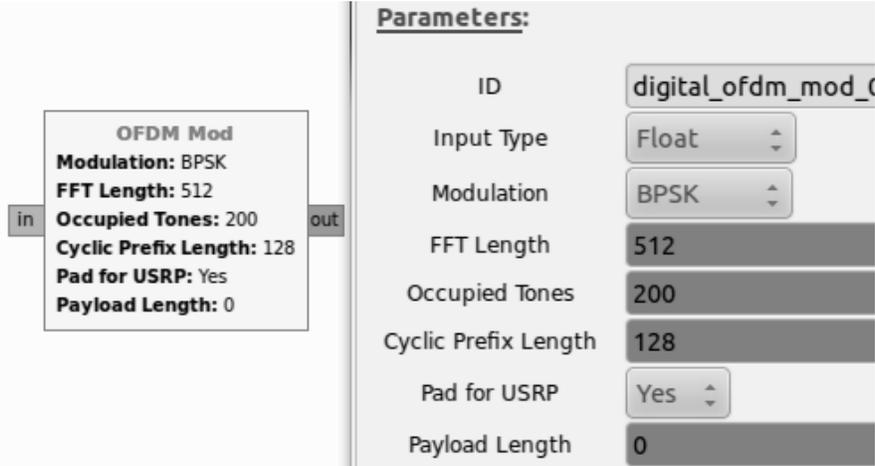


FIGURA 4. 22 BLOQUE OFDM MOD Y SU CONFIGURACIÓN

4.2.6 CODIFICACIÓN CONTRA ERRORES EN GNU RADIO

La codificación contra errores en GNU Radio se puede aplicar mediante los bloques codificador de trellis, codificador RM y codificador RMG.

En la configuración del bloque trellis encoder se establece la máquina de estados finitos que se utiliza para codificar el flujo de bits. La máquina de estados finitos se establece en un archivo aparte y el parámetro que se configura en el bloque es la ruta a dicho archivo.

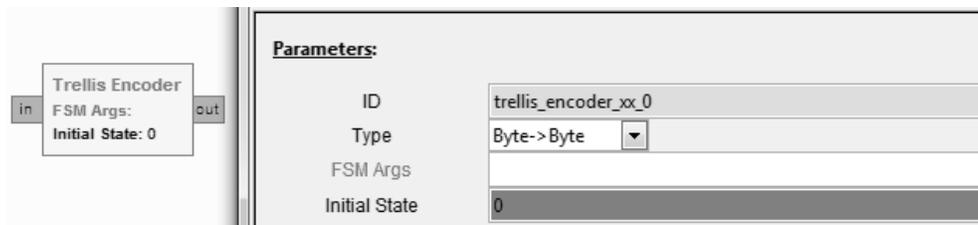


FIGURA 4. 23 BLOQUE DE CODIFICADOR TRELIS Y SU CONFIGURACIÓN

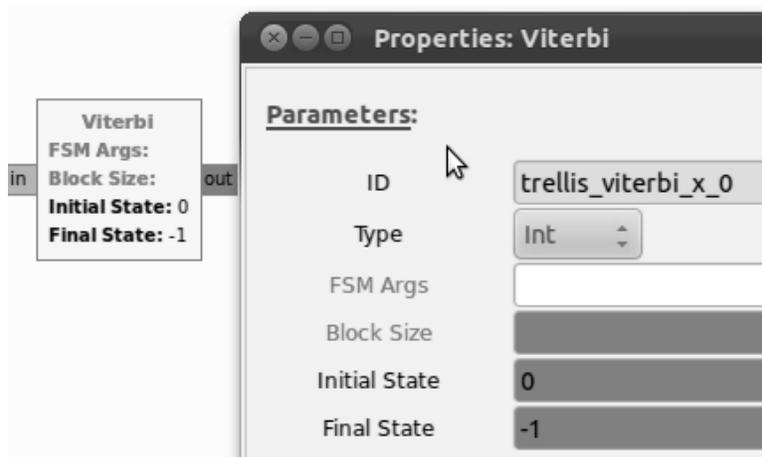


FIGURA 4. 24 BLOQUE VITERBI Y SU CONFIGURACIÓN

Los bloques que utilizan para decodificar una señal codificada mediante el bloque codificador de Trellis son los bloques métricas de Trellis y Viterbi. El bloque métricas de Trellis calcula las métricas que se necesitan para decodificar la señal mediante el algoritmo de Viterbi y el bloque Viterbi simplemente aplica el algoritmo de Viterbi de

acuerdo a las métricas calculadas y a la máquina de estados finitos que debe ser la misma que se utilizó en la codificación de la señal.

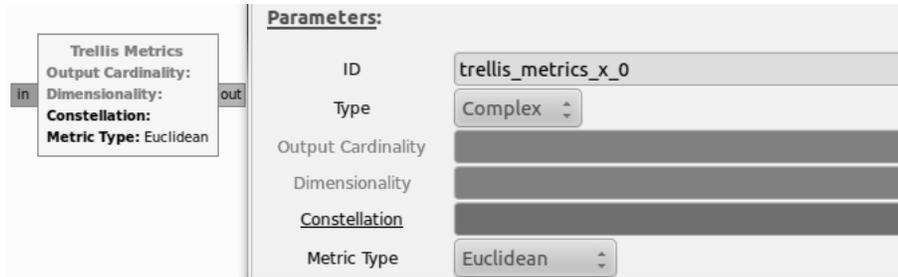


FIGURA 4. 25 BLOQUE MÉTRICAS DE TRELIS Y SU CONFIGURACIÓN

El bloque de codificación de Reed-Muller aplica la codificación contra errores a un flujo de bits, se puede configurar el código a utilizar mediante los parámetros r y m de los códigos Reed-Muller donde r es el grado del código Reed-Muller y 2^m es el tamaño de bloque. Para decodificar las señales a las que se aplica este proceso se utiliza el bloque de decodificación Reed-Muller que debe configurarse con los mismos parámetros que bloque utilizado para codificar la señal.

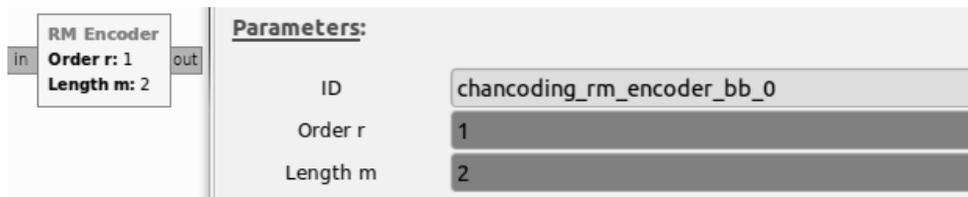


FIGURA 4. 26 BLOQUE DE CODIFICACIÓN REED-MULLER Y SU CONFIGURACIÓN

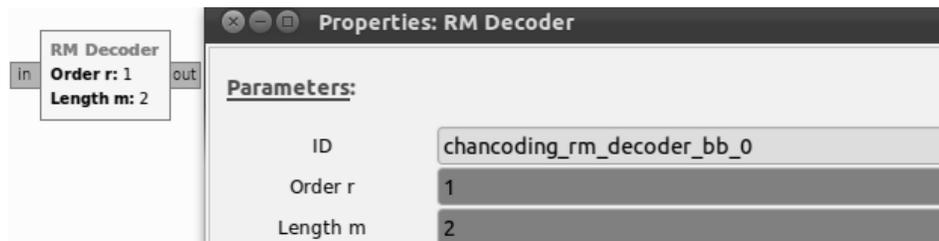


FIGURA 4. 27 BLOQUE DE DECODIFICACIÓN REED-MULLER Y SU CONFIGURACIÓN

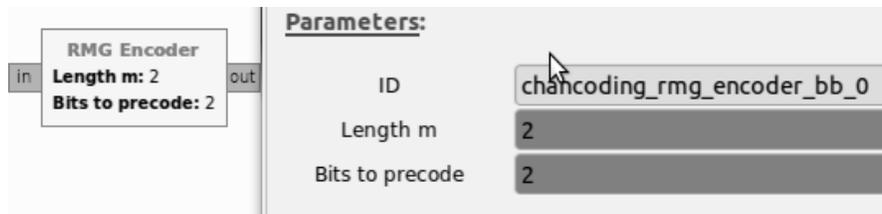


FIGURA 4. 28 BLOQUE DE CODIFICACIÓN RMG Y SU CONFIGURACIÓN

Otros bloques utilizados para agregar la redundancia y decodificarla son los bloques codificador RMG y decodificador RMG que utilizan un el código de Golay. El código de Golay forma parte de la familia de códigos Reed-Muller.

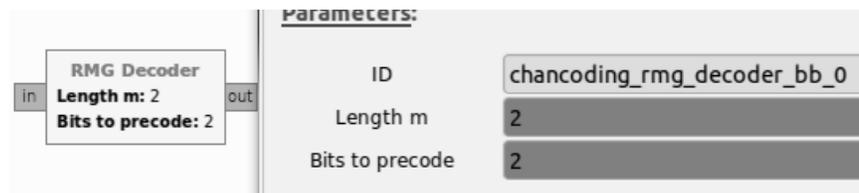


FIGURA 4. 29 BLOQUE DE DECODIFICACIÓN RMG Y SU CONFIGURACIÓN

Además de la codificación de errores existen otros bloques relacionados con la codificación de errores como son el interleaving y el puncturing.

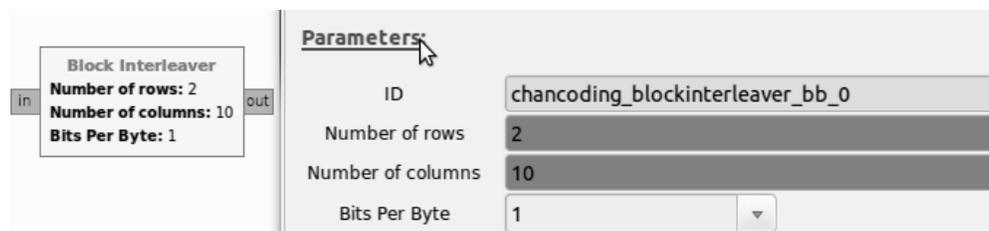


FIGURA 4. 30 BLOQUE INTERLEAVER Y SU CONFIGURACIÓN

El bloque interleaver sirve para intercalar el flujo de bits que sale de un codificador y entra a otro cuando se utilizan dos códigos concatenados. En el bloque se debe especificar el número de filas y de columnas que se llenarán para generar el nuevo flujo de bits en “desorden”.

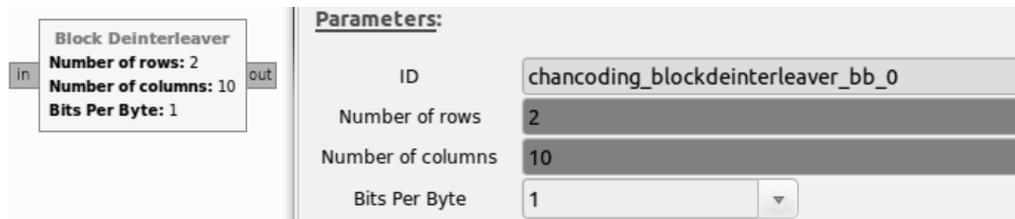


FIGURA 4. 31 BLOQUE INTERLEAVER Y SU CONFIGURACIÓN

El bloque deinterleaver realiza exactamente la función contraria que el bloque interleaver y es muy importante que tenga la misma configuración que la del bloque interleaver que se le aplicó a la señal.

El bloque puncture recorta uno de los bits de redundancia que el codificador contra errores agregó al flujo de bits. A primera vista esto resultaría en una desventaja porque el rendimiento del código disminuye, pero la tasa de bits también disminuye lo que implica que la señal necesita un ancho de banda menor para ser transmitida. Generalmente este proceso se utiliza cuando el código para protección contra errores introduce mucha redundancia y el canal de comunicaciones no es muy ruidoso.

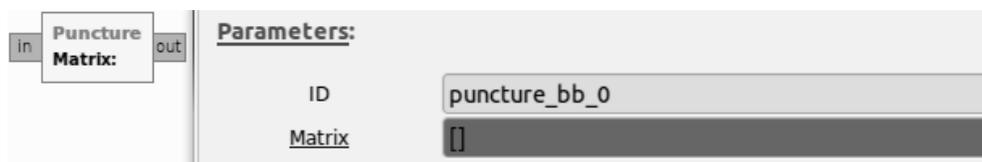


FIGURA 4. 32 BLOQUE PUNCTURE Y SU CONFIGURACIÓN

El bloque depuncture agrega un bit aleatoriamente de forma que la señal de nuevo tenga la tasa de bits de la señal original antes de que se le aplicará el proceso de puncture. De forma que no se tenga que aplicar una técnica especial para decodificar la protección contra errores aplicadas. Además agregar un bit aleatoriamente representa la posibilidad de que este bit no sea un error necesariamente.



FIGURA 4. 33 BLOQUE DEPUNCTURE Y SU CONFIGURACIÓN.

Existen otros bloques que sirven para calcular el BER en el receptor utilizando la señal original y la señal recibida. Estos bloques son el bloque muestra BER y tasa de errores.

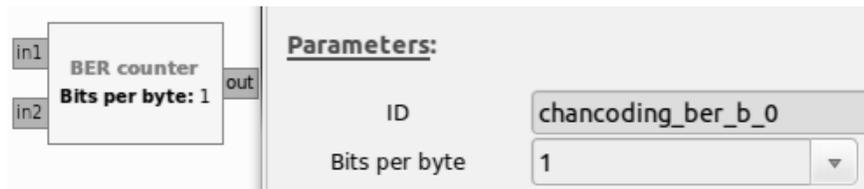


FIGURA 4. 34 BLOQUE CONTADOR BER Y SU CONFIGURACIÓN

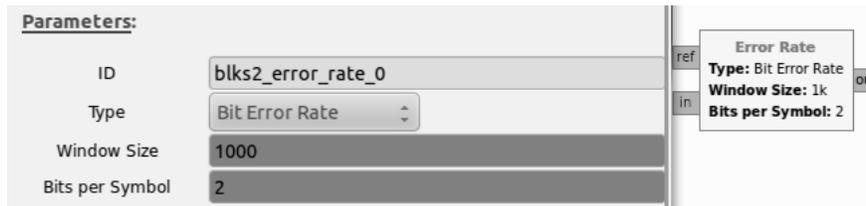


FIGURA 4. 35 BLOQUE TASA DE ERRORES Y SU CONFIGURACIÓN

5. PRÁCTICAS

Basadas en las tecnologías que se utilizan en la televisión digital y principalmente en el estándar ATSC (por ser el que México adopto) se diseñaron las siguientes prácticas para el laboratorio de televisión digital terrestre.

5.1 INTRODUCCIÓN A GNU RADIO

En esta práctica se introduce al estudiante a los tipos de datos, los distintos bloques de procesamiento, la interfaz de GNU Radio Companion y su funcionamiento, además de la programación mediante el lenguaje Python para conectar los bloques de procesamiento de señales de GNU Radio.

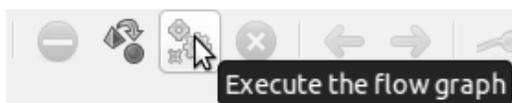


FIGURA 5. 1 BOTONES DE SIMULACIÓN EN GNU RADIO

En la primera parte de esta práctica se explican al estudiante los botones que sirven para depurar el diagrama, compilar el diagrama de GNU Radio Companion, ejecutar una simulación, detener la simulación en proceso. Estos botones se muestran en la figura 5.1.

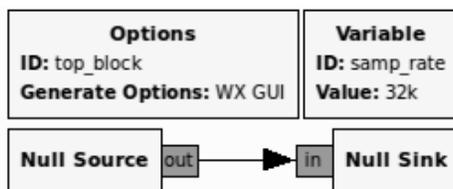


FIGURA 5. 2. DIAGRAMA CON BLOQUES NULL EN GNU RADIO COMPANION

La segunda parte de la práctica consiste en elaborar el diagrama que se muestra en la figura 5.2, compilar el diagrama y analizar el código que genera al compilar el diagrama, que se encuentra en el anexo 2.

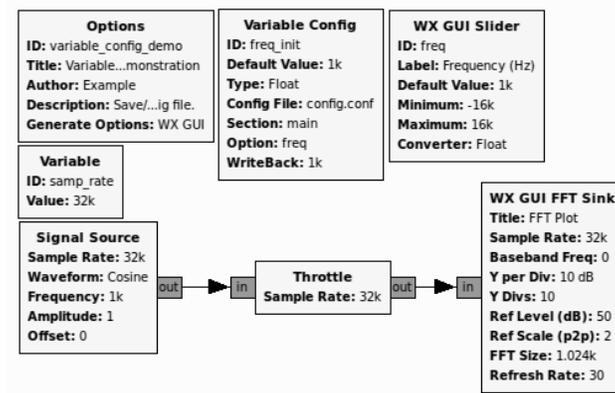


FIGURA 5. 3 DIAGRAMA DE ANÁLISIS EN FRECUENCIA DE LAS SEÑALES DEL GENERADOR DE SEÑALES EN GNU RADIO COMPANION

En la tercera parte de la práctica el estudiante debe generar un diagrama donde utilice el generador de señales y pueda observar el espectro de las señales, de forma que la frecuencia de la señal se pueda modificar durante la simulación. Este diagrama se puede observar en la figura 5.3.

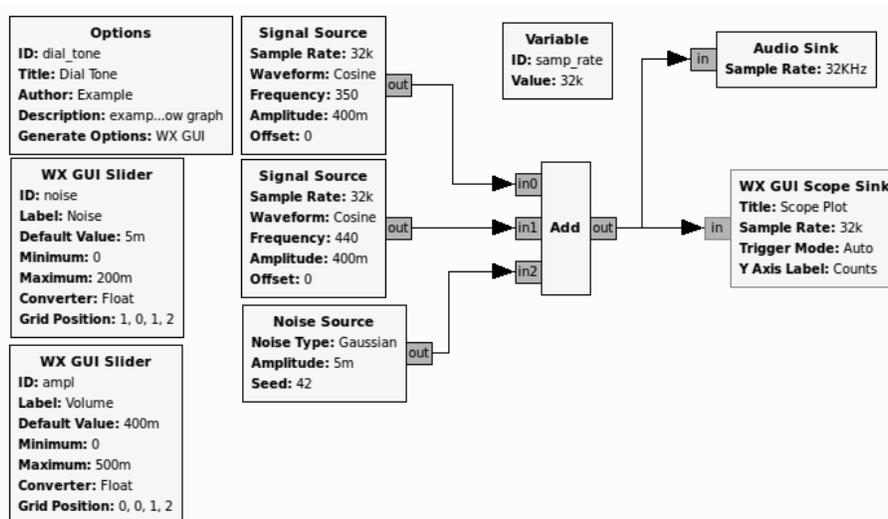


FIGURA 5. 4. DIAGRAMA DE SUMA DE SEÑALES Y SUMIDEROS DE AUDIO Y OSCILOSCOPIO EN GNU RADIO COMPANION

La última parte de esta práctica consiste en construir una simulación en GNU Radio Companion que suma dos señales y una señal de ruido para visualizarlas con el sumidero osciloscopio de GNU Radio Companion y escucharlas mediante el sumidero de audio, y ejecutar la simulación. Concretamente el diagrama que el estudiante debe construir se puede observar en la figura 5.4.

5.2 SIMULACIÓN DEL CANAL DE COMUNICACIONES CON GNU RADIO

En esta práctica el estudiante debe aprender a manejar los bloques que se utilizaran para simular el canal de comunicaciones durante las prácticas.

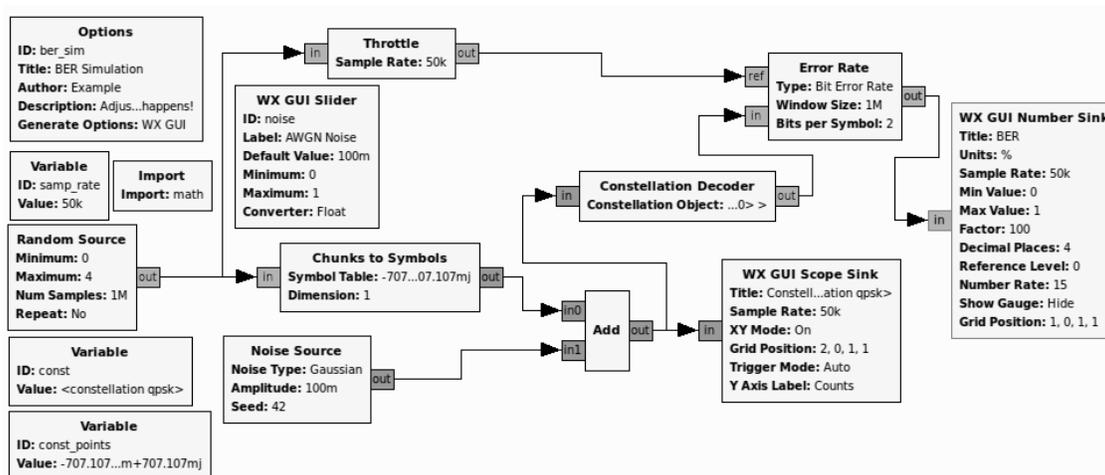


FIGURA 5. 5 SIMULACIÓN DEL CANAL DE COMUNICACIONES MEDIANTE EL BLOQUE FUENTE DE RUIDO EN GNU RADIO COMPANION

La primera forma de simular el canal de comunicaciones es sumando una señal de ruido a la señal que se busca transmitir de forma que el canal simplemente se considera en función del nivel de ruido que agrega a la señal y el nivel de la relación señal a ruido en el receptor. En las transmisiones digitales el resultado más importante a considerar para la transmisión es la tasa de bits en error o BER por lo que en las simulaciones del canal de transmisiones es importante obtener este parámetro. El diagrama que se ocupa en la práctica para entender la operación de este bloque se puede ver en la figura 5.5.

En la práctica el estudiante también debe simular el canal de comunicaciones mediante el bloque modelo de canal o Channel Model, el diagrama que se utiliza para simular el canal se puede observar en la figura 5.6. Las ventajas de este bloque es que incorpora la opción de aumentar el nivel de ruido, variar la permitividad del medio donde se transmiten la señal y agregar el efecto de Doppler, manipulando la frecuencia Doppler. Para poder visualizar los efectos de este bloque se utiliza un sumidero FFT para ver los efectos sobre el espectro de la señal y sumidero osciloscopio para observar los efectos sobre la constelación que se presenta en el receptor.

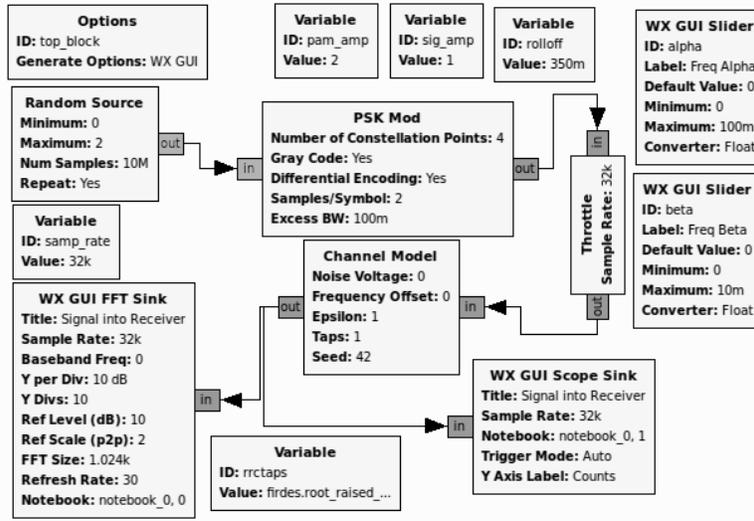


FIGURA 5. 6 SIMULACIÓN DEL CANAL DE COMUNICACIONES MEDIANTE EL BLOQUE MODELO DE CANAL EN GNU RADIO COMPANION

Por último en esta práctica se simula un canal simétrico binario utilizando el bloque Binary Symmetric Channel, este bloque simula un canal simétrico binario con una tasa de bits en error dado. Este tipo de simulación de canal sirve principalmente para ver el desempeño de los códigos correctores de errores. Al ejecutar la simulación con un diagrama en GNU Radio Companion como el que se muestra en la figura 5.7 se puede observar el BER que se obtiene cuando se utiliza el código corrector de errores y el que se obtiene cuando no se utiliza el código.

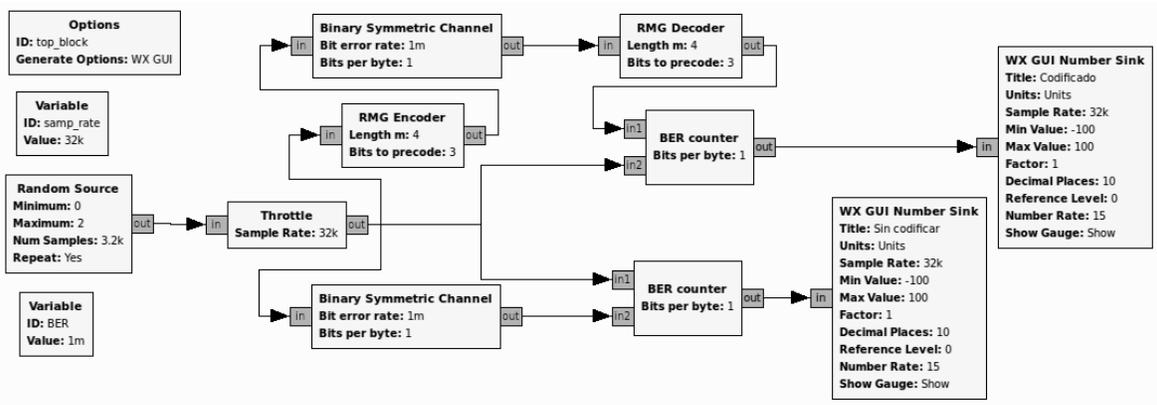


FIGURA 5. 7 SIMULACIÓN DE UN CANAL SIMETRICO BINARIO EN GNU RADIO COMPANION

5.3 MODULACIÓN EN FRECUENCIA CON GNU RADIO

El objetivo de esta práctica es que el estudiante conozca y aprenda a utilizar los bloques de modulación en frecuencia de GNU Radio, que simule una transmisión de datos empleando la modulación FSK y que se relacione con las características de esta transmisión.

En la práctica el estudiante debe crear un esquema en GNU Radio Companion semejante al de la figura 5.8, compilar y ejecutar la gráfica para ver el espectro de la señal que genera cuando se utiliza la modulación FSK mediante el bloque WX GUI FFT Sink y la representación de la señal en el tiempo mediante el bloque WX GUI Scope Sink.

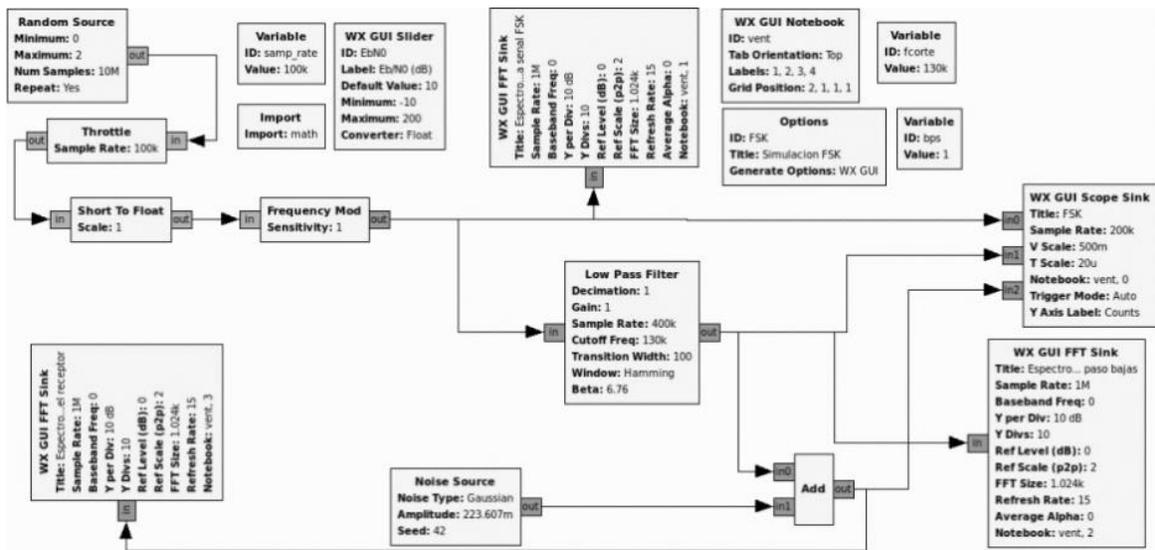


FIGURA 5. 8 DIAGRAMA DE LA SIMULACIÓN DE UNA TRANSMISIÓN CON LA MODULACIÓN FSK EN GNU RADIO COMPANION

En la figura 5.9 se muestra el espectro de la señal modulada en la simulación, que se obtiene a partir del bloque sumidero FFT, como parte de los resultados de la práctica el estudiante debe justificar teóricamente el espectro de la señal. Por otra parte en la figura

5.10 se puede ver una representación de la señal en el tiempo como resultado de la simulación, la señal con línea punteada es la señal que se transmite y la otra es la señal que se recibe. En los resultados del reporte de la práctica que el estudiante entregue debe comprobar que se cumple la teoría que describe la modulación.

Además de realizar el esquema en GNU Radio Companion y ejecutar la simulación desde el mismo programa, el estudiante debe analizar el código en lenguaje Python que el software GNU Radio Companion genera y comentar el código. El código comentado debe anexarse al reporte de la práctica.

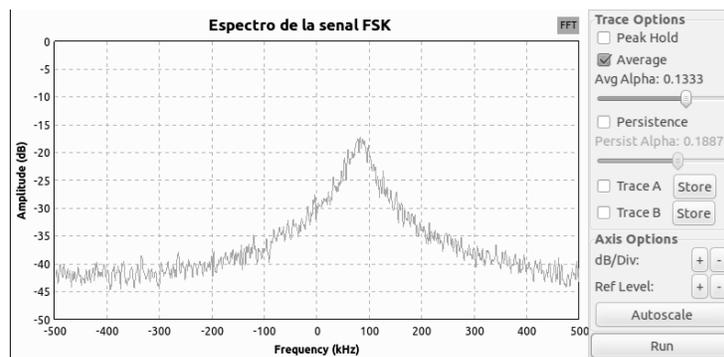


FIGURA 5. 9 ESPECTRO DE UNA SEÑAL CON MODULACIÓN FSK

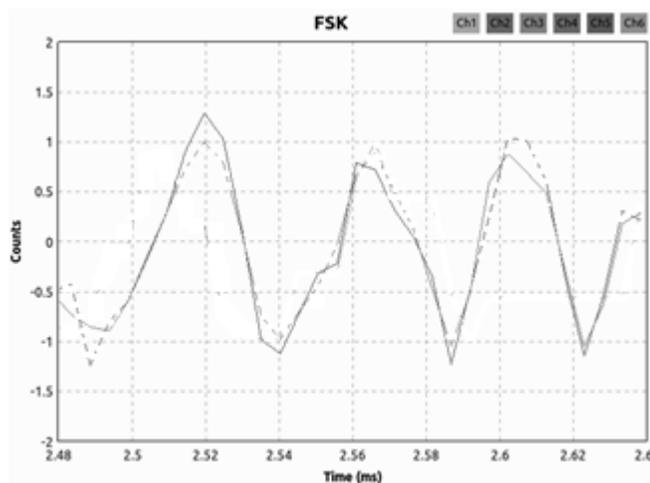


FIGURA 5. 10 SEÑAL CON MODULACIÓN FSK EN EL OSCILOSCOPIO

El estudiante también debe elaborar el código necesario para simular transmisiones de datos utilizando esta modulación sobre un canal de comunicaciones con diferentes relaciones señal a ruido y obtener el BER de cada transmisión para graficarlo. El resultado debe estar presente en el reporte de la práctica como parte de los resultados.

5.4 MODULACIONES DIGITALES EN AMPLITUD Y EN CUADRATURA

En esta práctica el estudiante aprenderá a utilizar el bloque chunks to symbols para modular señales digitales (específicamente las modulaciones PAM) y el bloque QAM mod para aplicar modulaciones en cuadratura, además de visualizar las modulaciones con un bloque sumidero osciloscopio, un bloque sumidero FFT y un bloque sumidero de osciloscopio. Por último simulará transmisiones aplicando la modulación QAM con diferentes relaciones señal a ruido y calculará el Bit Error Rate para cada transmisión. Con los resultados del BER que obtenga de la simulación deberá realizar una gráfica del BER contra la relación señal a Ruido.

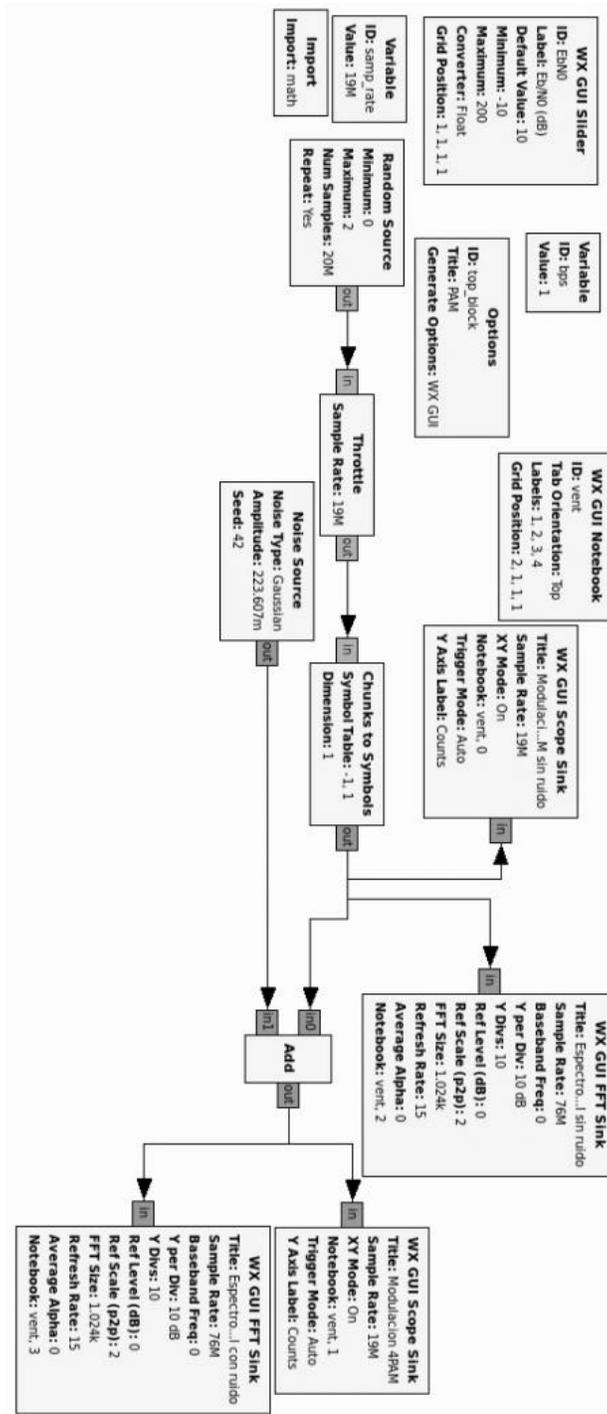


FIGURA 5. 11 SIMULACIÓN DE LA MODULACIÓN 4PAM EN GNU RADIO COMPANION

Para observar las características de la modulación PAM el estudiante debe desarrollar un diagrama como el de la figura 5.11 en GNU Radio Companion de modo que cuando

ejecute la simulación pueda ver las constelaciones de PAM y 4PAM como las que se muestran en las figuras 5.12 y 5.14 y sus espectros como en las figuras 5.13 y 5.15.

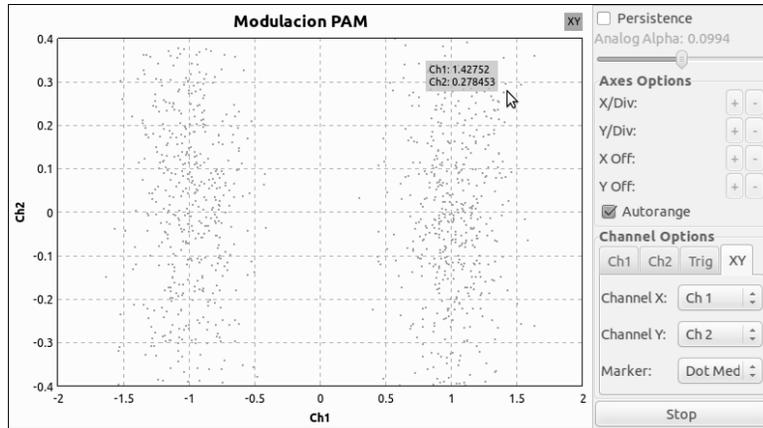


FIGURA 5. 12 CONSTELACIÓN DE UNA SEÑAL MODULADA CON PAM

El estudiante debe incluir en el reporte de la práctica las constelaciones teóricas de las modulaciones PAM y una explicación sobre sus diferencias o similitudes con las constelaciones que se pueden observar durante las simulaciones. De la misma manera debe repetir el ejercicio antes descrito con los espectros de las señales que se pueden observar en las simulaciones.

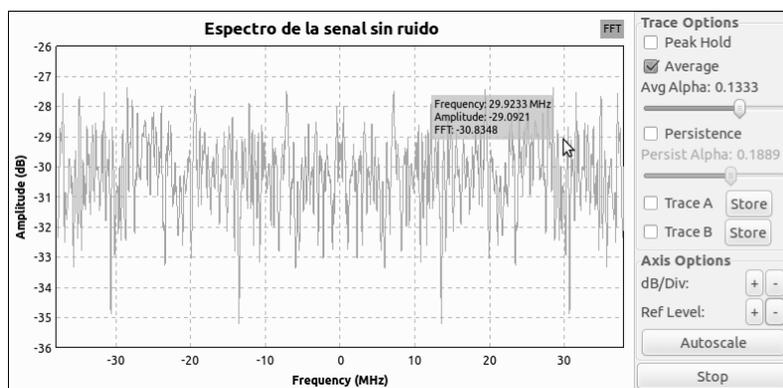


FIGURA 5. 13 ESPECTRO DE UNA SEÑAL MODULADA MEDIANTE PAM

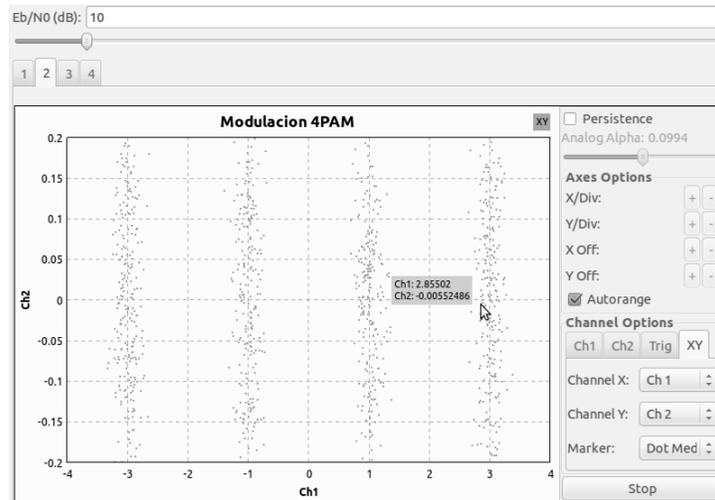


FIGURA 5. 14 CONSTELACIÓN DE UNA SEÑAL MODULADA CON 4PAM

Durante la simulación se pueden observar los efectos del ruido utilizando el control deslizante del nivel E_b/N_0 . De forma que es muy sencillo apreciar el efecto que provoca el ruido sobre las modulaciones que se aplican.

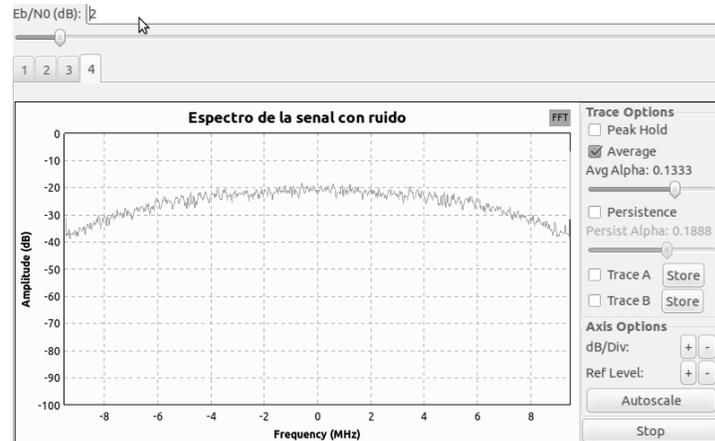


FIGURA 5. 15 ESPECTRO DE UNA SEÑAL MODULADA MEDIANTE 4PAM

Es importante que en esta práctica el alumno pueda apreciar la relación entre el ancho de banda que ocupa la señal y el número de símbolos de la constelación que se ocupa. El estudiante debe describir esta relación en las conclusiones del reporte de la práctica.

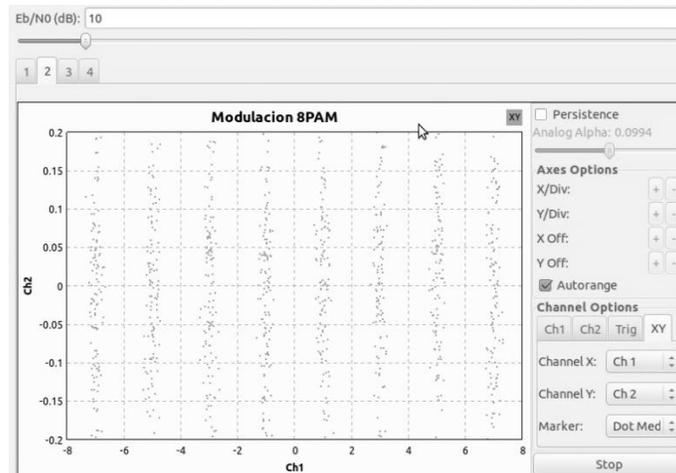


FIGURA 5. 16 CONSTELACIÓN DE UNA SEÑAL MODULADA CON 8PAM

A pesar de que la simulación de la modulación QAM se puede realizar con el bloque Chunks to Symbols, en esta práctica la simulación de la modulación QAM se realiza con el bloque QAM Mod de forma que el estudiante aprenda a utilizar otro bloque y a configurar sus parámetros.

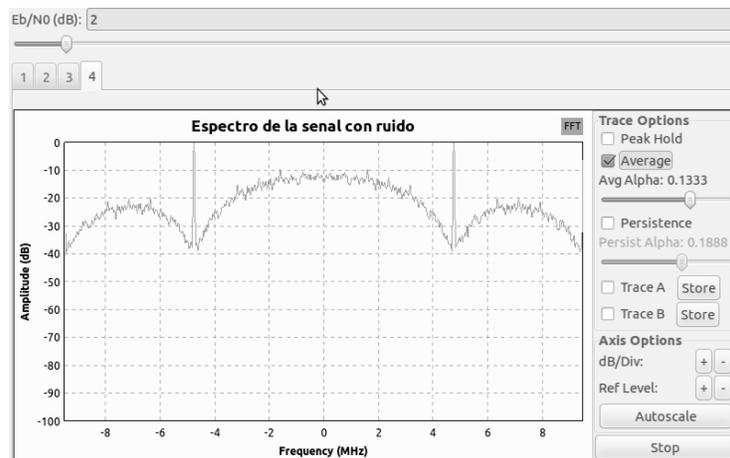


FIGURA 5. 17 ESPECTRO DE UNA SEÑAL MODULADA MEDIANTE 8PAM

En la figura 5.18 se muestra el diagrama que se debe construir en GNU Radio Companion para realizar la simulación. Al ejecutar la simulación el estudiante debe visualizar el

espectro de la señal como se muestra en la figura 5.17 y la constelación de la modulación que se recibe que corresponde con la figura 5.19.

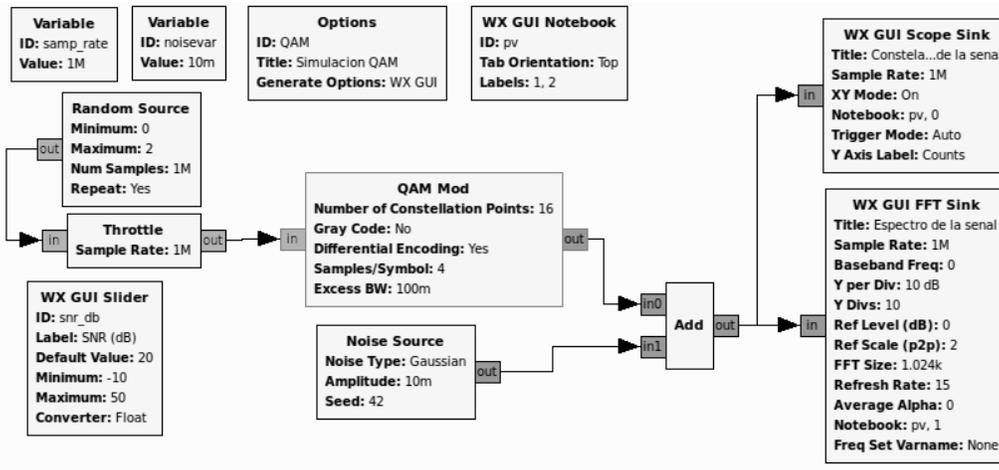


FIGURA 5. 18 SIMULACIÓN DE LA MODULACIÓN 16-QAM EN GNU RADIO COMPANION

Una ventaja de utilizar el bloque QAM Mod es que puede aplicarse codificación diferencial y/o el código Gray mediante el mismo bloque. Por otra parte el bloque solo necesita el número puntos de la constelación para construir la misma.

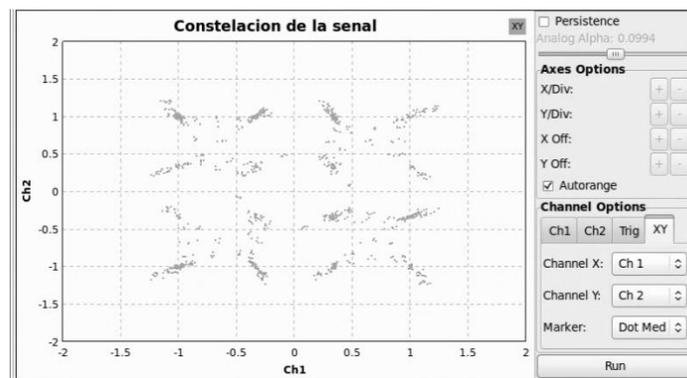


FIGURA 5. 19 CONSTELACIÓN DE UNA SEÑAL MODULADA CON 16-QAM

Al igual que otras simulaciones se puede observar el efecto que tiene el ruido sobre la señal que se recibe tanto en la constelación como en el espectro de la señal.

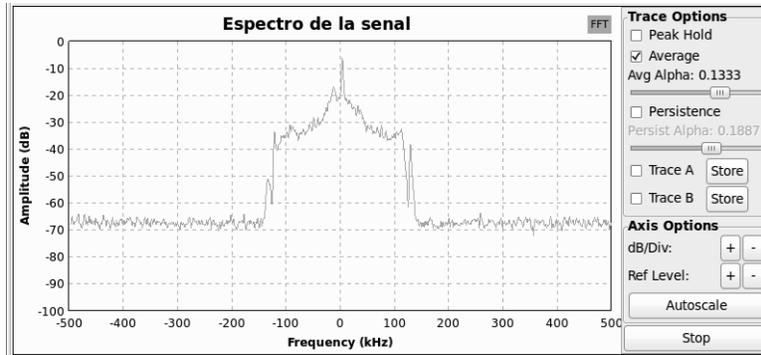


FIGURA 5. 20 ESPECTRO DE UNA SEÑAL MODULADA MEDIANTE 16-QAM

Además el estudiante debe analizar el código que GNU Radio Companion genera, comentarlo y agregarlo al reporte de la práctica. Por último debe escribir el código necesario para simular transmisiones utilizando la modulación QAM con diferentes relaciones señal a ruido y determinar el BER de cada simulación, de forma que pueda realizar una gráfica del BER y la SNR que deberá incorporar al reporte de la práctica.

5.5 MODULACIONES DIGITALES EN FASE

El propósito de esta práctica es que el estudiante aprenda a utilizar el bloque PSK Mod para aplicar modulación en fase a una secuencia bits de forma que pueda observar las constelaciones y los espectros de las modulaciones PSK. Además deberá obtener una gráfica para cada modulación del BER con respecto a la relación señal a ruido, de manera que el estudiante entienda la robustez de las modulaciones frente al ruido.

En las figuras 5.21 y 5.22 se muestran las constelaciones de las modulaciones BPSK y QPSK que el alumno deberá observar cuando ejecute las simulaciones correspondientes. De la misma forma que en otras prácticas el alumno puede ver el efecto que el nivel del

ruido tiene sobre la transmisión manipulando el nivel de ruido con el control deslizante mientras se ejecuta la simulación.

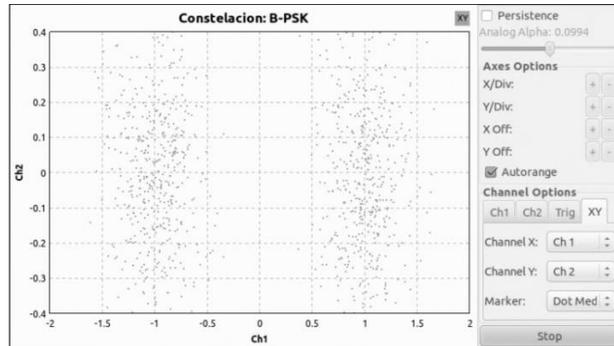


FIGURA 5. 21 CONSTELACIÓN DE MODULACIÓN 8PSK

De manera similar a prácticas anteriores el estudiante debe analizar el código que GNU Radio Companion genera para realizar las simulaciones y comentarlo, después escribir el código necesario para simular las transmisiones y poder graficar el BER y el E_b/N_0 .

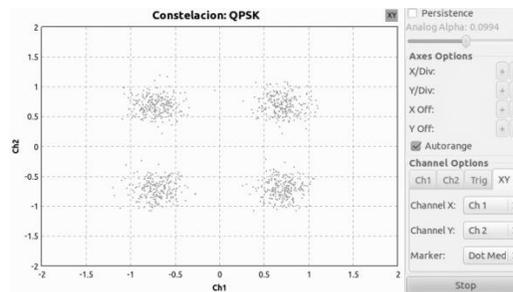


FIGURA 5. 22 CONSTELACIÓN DE MODULACIÓN QPSK

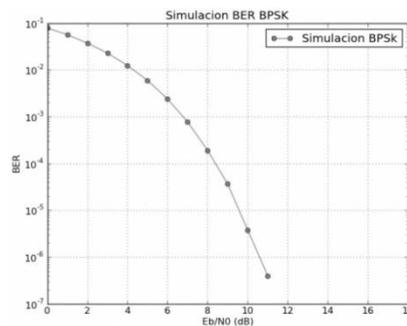


FIGURA 5. 23 RESULTADOS BER DE LA SIMULACIÓN BPSK

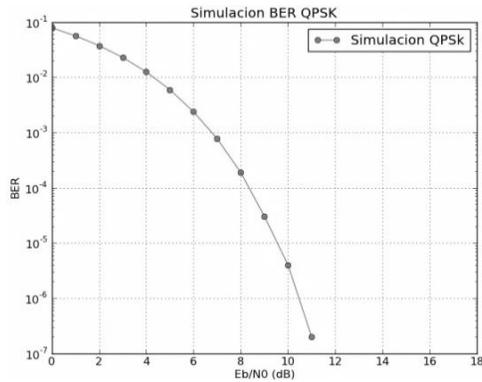


FIGURA 5. 24 RESULTADOS BER DE LA SIMULACIÓN QPSK

Las gráficas que relacionan el BER y la relación señal a ruido o la energía por bit a la potencia espectral del ruido sirven para poder ver el desempeño del código. Los resultados que el estudiante debe obtener en esta práctica se muestran en las figuras 5.23, 5.24 y 5.25.

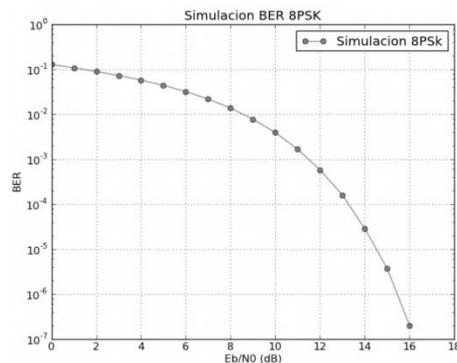


FIGURA 5. 25 RESULTADOS BER DE LA SIMULACIÓN 8PSK

5.6 MULTIPLEXACIÓN POR DIVISIÓN DE FRECUENCIAS ORTOGONALES

En esta práctica el estudiante realizará simulaciones con el bloque Mod OFDM para aplicar la modulación OFDM a un flujo de bits. La importancia de esta modulación es que

se utiliza en todos los sistemas de transmisión de televisión digital terrestre menos en el sistema ATSC. Por otra parte la implementación de esta modulación es necesariamente digital debido a que se debe aplicar la transformación IDFT en el transmisor y la transmisión DFT en su recepción.

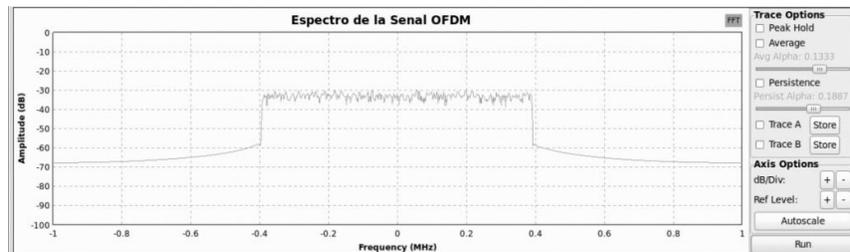


FIGURA 5. 26 ESPECTRO DE LA MODULACIÓN OFDM EN LA SIMULACIÓN

El alumno debe utilizar en GNU Radio Companion un diagrama como el que se muestra en la figura 5.29. Al ejecutar la simulación el estudiante debe observar el espectro de la modulación y la señal graficada en el tiempo.

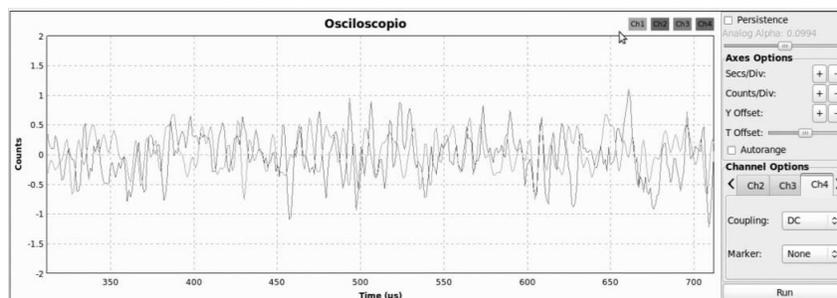


FIGURA 5. 27 MODULACIÓN OFDM EN LA SIMULACIÓN

Es importante utilizar el bloque sumidero FFT y el bloque sumidero Waterfall para ver el espectro de la señal, debido a que su espectro cambia de manera importante a lo largo del tiempo. El espectro instantáneo que se obtiene de la simulación con el sumidero FFT se muestra en la figura 5.26, que como se puede ver es similar al espectro teórico de la señal, y en la figura 5.28 se muestra el resultado que se obtiene a partir del bloque sumidero Waterfall. La representación Waterfall es una gráfica de 3 dimensiones, donde

un eje representa el tiempo, otro eje representa la frecuencia y potencia de la señal en cierta frecuencia se representa con colores. Y en la figura 5.27 se muestra la señal de OFDM que se transmite en el dominio del tiempo.

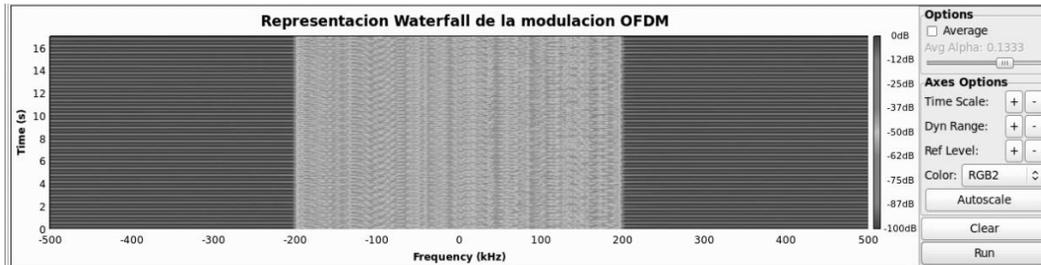


FIGURA 5. 28 REPRESENTACIÓN WATERFALL DE LA MODULACIÓN OFDM EN LA SIMULACIÓN

De forma similar a otras prácticas el alumno debe analizar el código que se genera mediante GNU Radio Companion y comentarlo. También realizar las simulaciones necesarias para generar una gráfica del BER y la relación señal a ruido.

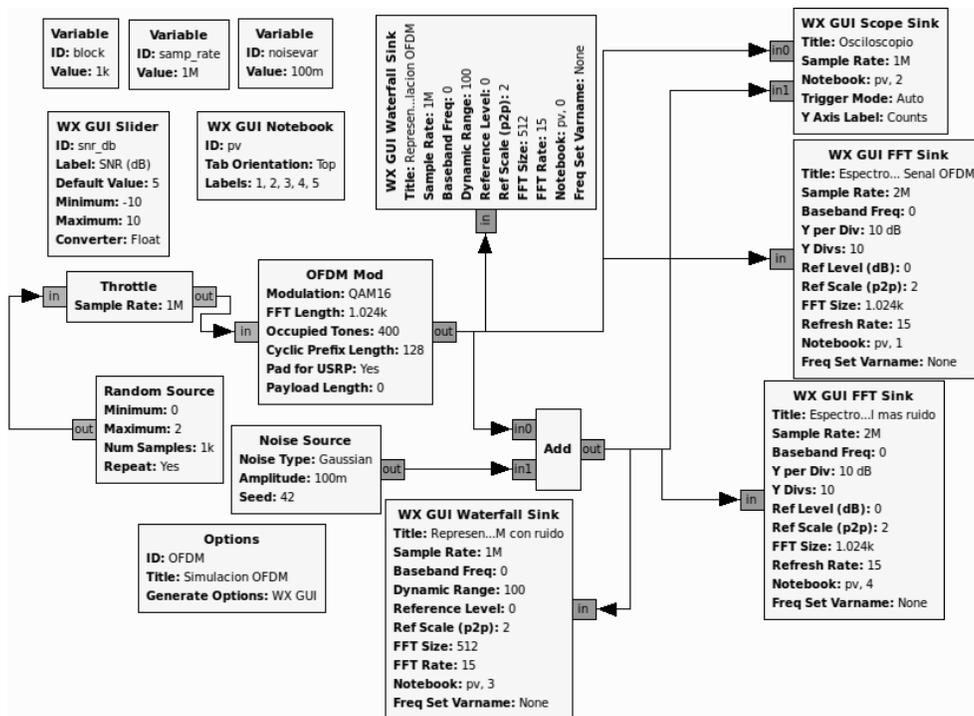


FIGURA 5. 29 SIMULACIÓN DE LA MODULACIÓN OFDM EN GNU RADIO COMPANION

5.7 CÓDIGOS LINEALES DE BLOQUES PARA PROTECCIÓN CONTRA ERRORES

El propósito de esta práctica es que el estudiante aprenda a simular transmisiones donde se utilizan códigos lineales de bloques para protección contra errores. Para ello simulará una transmisión donde al flujo de bits se le aplica el código y otra donde no se aplica, de manera que pueda apreciar el BER que se obtiene aplicando el código y el BER que se obtendría sin aplicarlo.

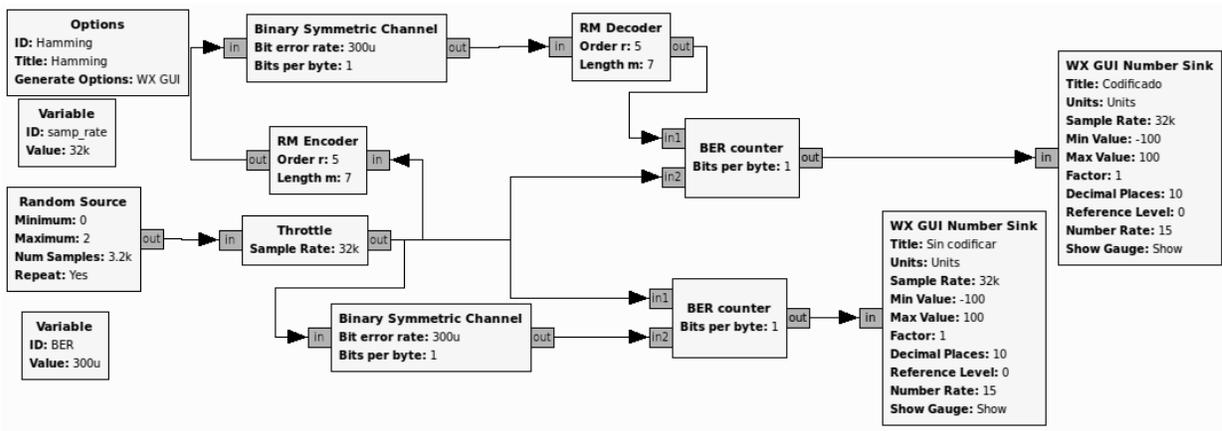


FIGURA 5. 30 SIMULACIÓN DE TRANSMISIÓN SOBRE UN CANAL SIMETRICO BINARIO DONDE SE APLICA UN CÓDIGO REED-MULLER EN GNU RADIO COMPANION

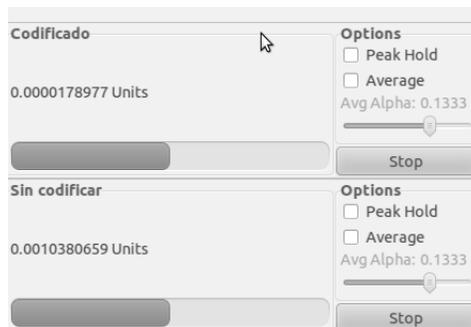


FIGURA 5. 31 BER RESULTADO DE LA SIMULACIÓN DE UNA TRANSMISIÓN SOBRE UN CANAL SIMETRICO BINARIO DONDE SE APLICA UN CÓDIGO REED-MULLER

El diagrama que se debe elaborar en esta práctica en GNU Radio Companion para simular la transmisión de una señal mediante un Trellis y su decodificación se muestra en la figura 5.32, se muestran los bloques necesarios para la decodificación como codificación de la señal.

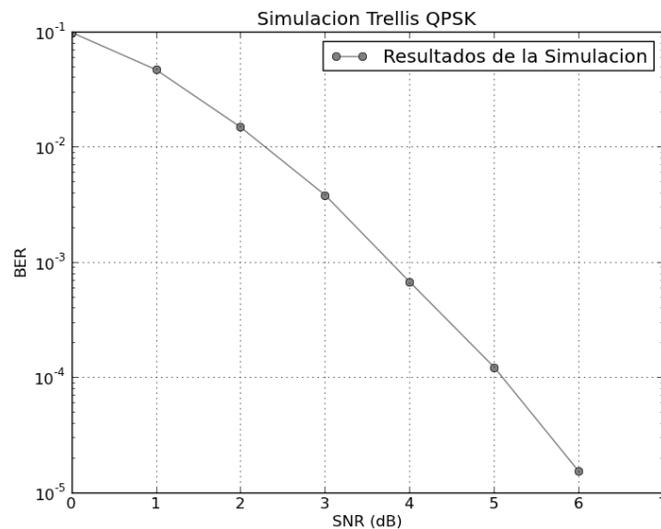


FIGURA 5. 33 RESULTADOS BER DE LA SIMULACIÓN TRELLIS QPSK

Debido a que los Trellis se componen de un codificador convolucional y un modulador digital siempre están asociados a un tipo de modulación. En esta práctica se utiliza una modulación QPSK en el Trellis y un código convolucional con tasa $\frac{1}{2}$. Los resultados de la simulación que se deben obtener en esta práctica se muestran en la figura 5.33.

5.9 INTERLEAVING Y CÓDIGOS CONCATENADOS

En esta práctica el estudiante simulará una transmisión aplicando dos códigos contra errores concatenados de manera que comparando con resultados que se obtienen con la

simulación pueda comprobar los beneficios de aplicar dos codificaciones contra errores concatenadas por un interleaver y sin el interleaver.

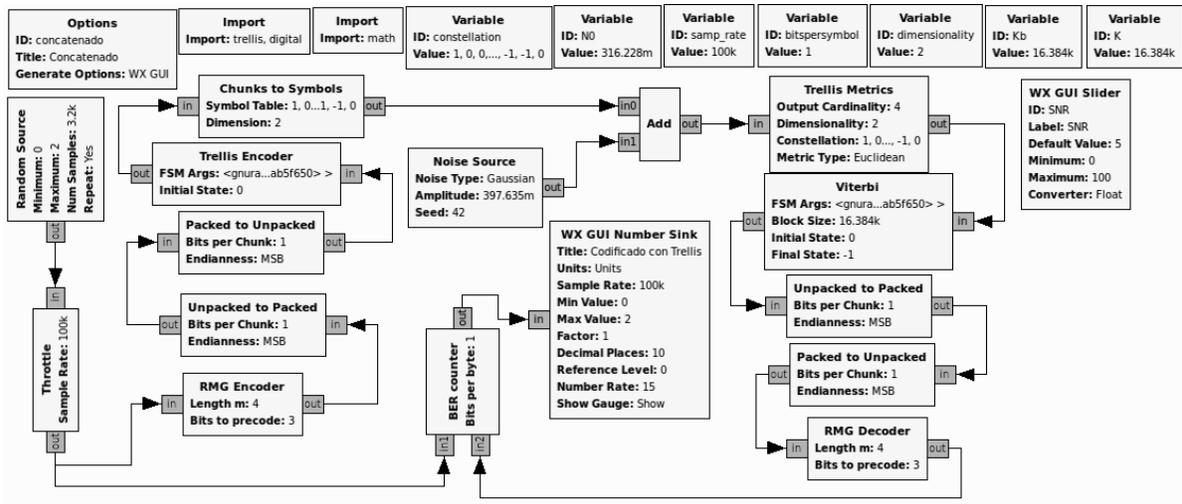


FIGURA 5. 34 SIMULACIÓN DE LA TRANSMISIÓN UNA SEÑAL CODIFICADA MEDIANTE DOS CÓDIGOS CONCATENADOS EN GNU RADIO COMPANION

El diagrama que se utiliza en esta práctica para simular la transmisión donde se codifica la señal mediante dos códigos de protección contra errores concatenados se muestra en la figura 5.34 y el resultado que se observa al ejecutar esta simulación se muestra en la figura 5.35.

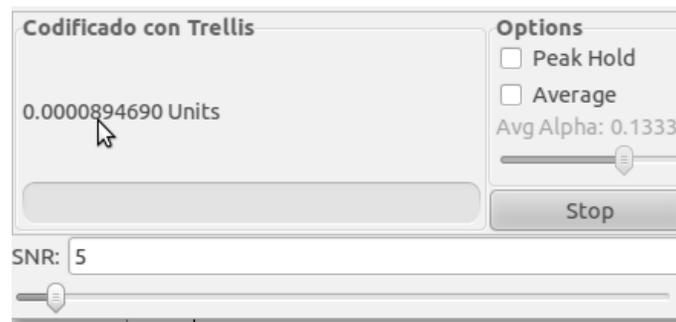


FIGURA 5. 35 RESULTADO DE LA SIMULACIÓN DE LA TRANSMISIÓN UNA SEÑAL CODIFICADA MEDIANTE DOS CÓDIGOS CONCATENADOS

Como se puede observar comparando los resultados que se obtienen en las dos simulaciones el desempeño de la codificación mejora cuando se emplea un interleaver para concatenar las dos codificaciones contra errores.

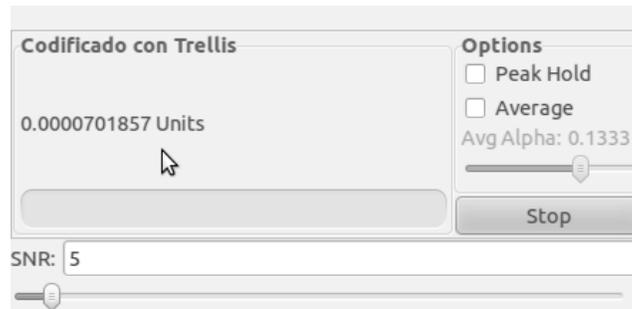


FIGURA 5. 37 RESULTADO DE LA SIMULACIÓN DE LA TRANSMISIÓN UNA SEÑAL CÓDIFICADA MEDIANTE DOS CÓDIGOS CONCATENADOS MEDIANTE UN INTERLEAVER

5.10 ESPECTRO DE LA SEÑAL CODIFICADA CON NTSC Y CON ATSC

El estudiante visualizará el espectro de una señal codificada con siguiendo el estándar ATSC y de una señal codificada con siguiendo el estándar NTSC.

El propósito de esta práctica es que identifique las diferencias entre una señal de NTSC y una señal de ATSC. En el espectro de la señal de NTSC el estudiante debe identificar las portadoras de audio y de video y en la de ATSC deberá identificar la portadora y comprobar la teoría.

El archivo de la señal de televisión codificada siguiendo el estándar NTSC fue grabada a través de un USRP el martes 19 de febrero de 2013 en la universidad Anáhuac.

Por otra parte el archivo de la señal de televisión codificada siguiendo el estándar ATSC fue grabada en Los Ángeles, Estados Unidos el 29 de octubre de 2010.

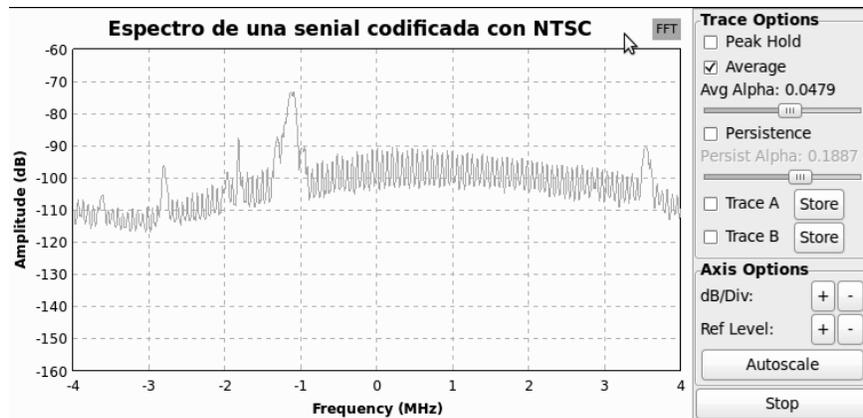


FIGURA 5. 38 ESPECTRO DE SEÑAL CODIFICADA CON NTSC

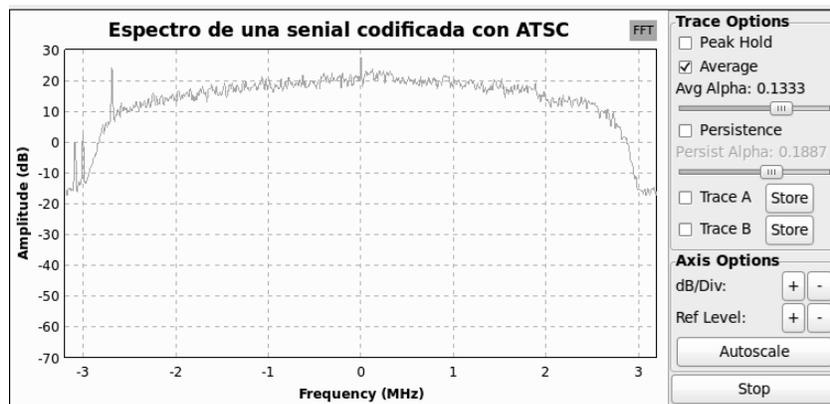


FIGURA 5. 39 ESPECTRO DE SEÑAL CODIFICADA CON ATSC

5.11 RESISTENCIA A ERRORES DEL VIDEO CODIFICADO CON H.264

En esta práctica el estudiante simulará la transmisión de un archivo de video codificado bajo el estándar H.264/AVC con un formato MPEG-TS. Para esto se debe crear un diagrama similar al que se muestra en la figura 5.40. De esta forma el estudiante puede simular la transmisión del archivo de video sobre un canal simétrico binario al que se le puede ajustar el BER en los parámetros del bloque Binary Symmetric Channel que después se comprobará con el bloque BER counter durante la simulación.

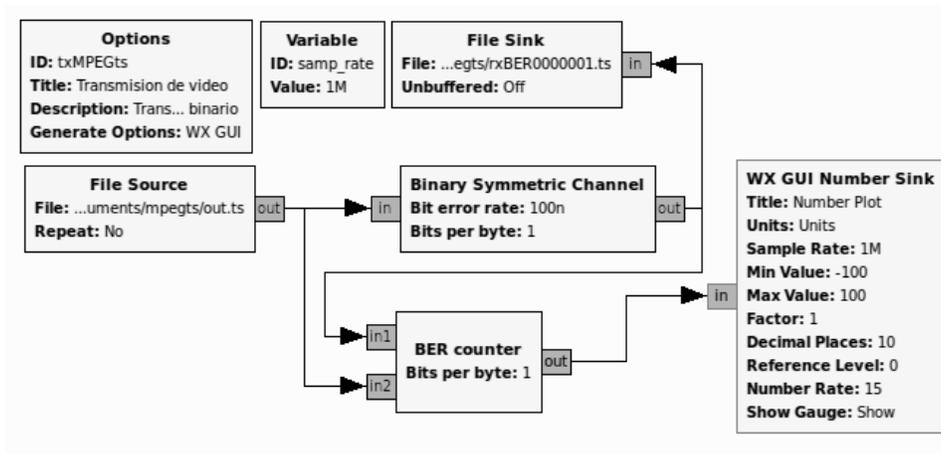


FIGURA 5. 40 DIAGRAMA DE LA SIMULACIÓN DE LA TRANSMISIÓN DE UN ARCHIVO DE VIDEO SOBRE UN CANAL SIMÉTRICO BINARIO

El alumno deberá simular diferentes VER durante la práctica hasta qué punto el archivo no sufrirá errores importantes y en qué punto el receptor simplemente no puede reconocer la transmisión de video. Ejemplos de resultados de la simulación con distintos BER se muestran en la figura 5.41.



FIGURA 5. 41 RESULTADOS DE LA SIMULACIÓN DE LA TRANSMISIÓN DE UN ARCHIVO DE VIDEO SOBRE UN CANAL SIMÉTRICO BINARIO CON DISTINTOS BER

CONCLUSIONES

La investigación que se realizó sobre los estándares de codificación de video, las modulaciones digitales y los esquemas de codificación contra errores que se utilizan en los sistemas de televisión digital terrestre y la plataforma GNU Radio y sus herramientas para crear radios definidos por software para diseñar las prácticas del Laboratorio de Televisión Digital Terrestre basado en Software Definido por Radio.

Se diseñaron y realizaron las simulaciones de los esquemas de modulación y de codificación contra errores que se utilizan en los sistemas de TDT de forma que el estudiante pueda comprenderlos explicándolos como un bloque de procesamiento de señales.

Se realizó un manual de prácticas [13] donde se describe el procedimiento para realizar cada práctica. De forma que el alumno entienda porque se utiliza una modulación o una codificación contra errores y conozca sus características, haciendo el aprendizaje más didáctico y no solamente un proceso de recepción.

El uso de GNU Radio para la enseñanza de los sistemas de Televisión Digital Terrestre resulta muy económico, debido a que no se tienen que comprar, ni construir moduladores, osciloscopios, analizadores de espectro, filtros, generadores de señales o de ruido, simplemente se necesita una computadora, la cual es indispensable para cualquier estudiante en la actualidad. Y aunque no es el caso de la tesis, se puede implementar los radios construidos a través de GNU Radio simplemente mediante un USRP (Universal Software Radio Peripheral).

BIBLIOGRAFÍA

- [1] M. Robin y M. Poulin, Digital television fundamentals : design and installation of video and audio systems, New York: McGraw Hill, 2000.

- [2] M. Cubero, La Televisión Digital: Fundamentos y Teorías, México: Alfaomega, 2009.

- [3] T. Perales Benito, Radio y televisión digitales : tecnología de los sistemas DAB, DVB, IBUC y ATSC, México: Limusa, 2006.

- [4] A. C. Bovik, The Essential guide to video processing, Amsterdam; Boston: Academic/Elsevier, 2009.

- [5] A. Bock, Video compression systems, Stevenage: Institution of Engineering and Technology, 2009.

- [6] B. Herve, Digital television : MPEG-1, MPEG-2, and principles of the DVB system, London; New York: Arnold ; J. Wiley, 1997.

- [7] F. Walter, Digital television: a practical guide for engineers, New York: Springer-Verlag, 2004.

- [8] A. Artés Rodríguez y F. Pérez González, Comunicaciones Digitales, Madrid: Pearson Educación, 2007.

- [9] J. H. Reed, *Software Radio: A Modern Approach to Radio Engineering*, 2002: Prentice Hall.
- [10] J. G. e. A. Delautre, «Review of SDR: past, present and future initiatives,» de *Wireless Research Forum*, 2004.
- [11] P. Johnson, «New Research Lab Leads to Unique Radio Receiver,» *E-Systems Team*, vol. Vol. 5, nº No. 4, pp. pp 6-7, 1985.
- [12] J. Mitola, «Software radios-survey, critical evaluation and future directions,» de *Telesystems Conference*, 1992.
- [13] Á. D. Juan Toral, *Manual del Laboratorio de Televisión Digital Terrestre basado en Radio Definido por Software*.
- [14] J. Boston, *DTV survival guide*, New York; Ciudad de México: McGraw-Hill, 2000.
- [15] J. H. Reed, *Software Radio: A Modern Approach to Radio Engineering*, Prentice Hall, 2002.

ANEXOS

ANEXO 1. GLOSARIO

AAC	Advance Audio Coding
AC	Alternate Component
AC-3	Audio Coding 3
ATSC	Advance Television System Committee
AVC	Advanced Video Coding
BER	Bit Error Rate
CABAC	Context-Adaptive Binary Arithmetic coding
COFDM	Coded Orthogonal Frequency Division Multiplex
DC	Direct Component
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DPCM	Differential Pulse Code Modulation
DSB	Double Side Band
DSP	Digital Signal Processor
DTMB	Digital Terrestrial Multimedia Broadcast
DVB	Digital Video Broadcast
E_b/N_0	Energy per bit to Noise power spectral density ratio
FFT	Fast Fourier Transform
FSK	Frequency Shift Keying
GOP	Group of Pictures
GPP	General Purpose Processor
GRC	Gnu Radio Companion
QAM	Quadrature Amplitude Modulation
ICI	Inter Carrier Interference
IDFT	Inverse Discrete Fourier Transform
IEC	International Electrotechnical Commission
ISDB-T	Integrated Services Digital Broadcasting – Terrestrial
ISI	Inter Symbolic Interference
ISO	International Organization for Standardization
ITU	International Telecommunication Union
JPEG	Joint Picture Expert Group
MDCT	Modified Discrete Cosine Transform
MPEG	Moving Picture Experts Group
NTSC	National Television Systems Committee
OFDM	Orthogonal Frequency Division Multiplex
PAM	Pulse Amplitude Modulation
PAL	Phase Alternating Line

PER	Packet Error Rate
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulator
SAP	Second Audio Program
SER	Symbol Error Rate
SNR	Signal to Noise Ratio
SSB	Single Side Band
UAE	Umbral de Audibilidad Enmascarada
USRP	Universal Software Radio Peripheral
VLC	Variable Length Coding
VSB	Vestigial Side Band

ANEXO 2. CÓDIGO FUENTE DEL DIAGRAMA DE FLUJO CON BLOQUES NULL EN GNU RADIO COMPANION

```
#!/usr/bin/env python
#####
# Gnuradio Python Flow Graph
# Title: Top Block
# Generated: Wed Mar 20 08:46:46 2013
#####

from gnuradio import eng_notation
from gnuradio import gr
from gnuradio.eng_option import eng_option
from gnuradio.gr import firdes
from grc_gnuradio import wxgui as grc_wxgui
from optparse import OptionParser
import wx

class top_block(grc_wxgui.top_block_gui):

    def __init__(self):
        grc_wxgui.top_block_gui.__init__(self, title="Top Block")
        _icon_path = "/usr/share/icons/hicolor/32x32/apps/gnuradio-grc.png"
        self.SetIcon(wx.Icon(_icon_path, wx.BITMAP_TYPE_ANY))

        #####
        # Variables
        #####
        self.samp_rate = samp_rate = 32000

        #####
        # Blocks
        #####
        self.gr_null_source_0 = gr.null_source(gr.sizeof_gr_complex*1)
        self.gr_null_sink_0 = gr.null_sink(gr.sizeof_gr_complex*1)

        #####
        # Connections
        #####
        self.connect((self.gr_null_source_0, 0), (self.gr_null_sink_0, 0))

    def get_samp_rate(self):
        return self.samp_rate

    def set_samp_rate(self, samp_rate):
        self.samp_rate = samp_rate
```

```
if __name__ == '__main__':  
    parser = OptionParser(option_class=eng_option, usage="%prog: [options]")  
    (options, args) = parser.parse_args()  
    tb = top_block()  
    tb.Run(True)
```