



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E S C I E N T I A H O M I N I S S A L U S "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**ELABORACIÓN DE MATERIAL DIDÁCTICO PARA LA
REALIZACIÓN DE PRÁCTICAS DE SISTEMAS DE
COMUNICACIONES USANDO HERRAMIENTAS
COMPUTACIONALES DE SIMULACIÓN**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERA EN ELECTRÓNICA Y TELECOMUNICACIONES**

VALERIA SORAYA MURILLO SALAZAR

DIRECTORA: ING. MARÍA SOLEDAD JIMÉNEZ JIMÉNEZ, MSc.

Quito, febrero 2021

AVAL

Certifico que el presente trabajo fue desarrollado por la Srta. Valeria Soraya Murillo Salazar, bajo mi supervisión.

Ing. María Soledad Jiménez, MSc.
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Valeria Soraya Murillo Salazar, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Valeria Soraya Murillo Salazar

DEDICATORIA

A mis padres

Soraya y Patricio, por su dedicación y constancia en el día a día a ser la mujer que soy; por su motivación, consejos, valores y sobre todo por su amor incondicional a continuar junto a mí en cada paso durante este largo trayecto, sin miedo a mi fracaso o éxito, gracias por su inmenso apoyo sin ustedes no lo hubiese logrado.

A mis hermanos

Me siento afortunada de tenerlos y saber que puedo contar con ellos cuando los necesito, es por eso que también dedico este trabajo a ***Johana, Ricardo, Alex y Esteban*** por haberme dado su voto de confianza, por creer en mí y ser parte de mi motivación a lograr mis metas, por mostrarme que tener hermanos es aprender a respetar, amar y perdonar. Dios no podía haberme dado unos hermanos mejor que ellos.

A mis abuelos

Marina, por sus consejos acertados y sobre todo por su desinteresado cariño y apoyo en el tiempo que pase junto a ella, le dedico mi trabajo porque me ha dado muchas razones para sonreír porque puso en mi niñez felicidad, cuidados y amor.

Finalmente, dedico mi trabajo a una persona muy especial que un día estuvo en los momentos más importantes de mi vida y que aun seguirá estando, por haber dejado en mí los más hermosos recuerdos de mi niñez y sobre todo por haber dejado sus valiosas herencias de responsabilidad, trabajo, esfuerzo, dedicación y humildad. Lamentablemente ya no está junto a mí pero donde sea que este siempre lo tengo presente y lo recuerdo, le dedico porque me hizo entender que el mejor abuelo del mundo si existía y era el mío, es por eso que esto va a la memoria de mi papi ***Edguitar Salazar*** /:

Valeria Murillo Salazar

AGRADECIMIENTO

Un infinito agradecimiento a toda mi familia, por la confianza y el apoyo que me han brindado durante mi carrera y mi vida, por formar parte de mis logros y triunfos, por los ánimos, por la fuerza en cada momento que hemos estado juntos. Gracias por todo su cariño e infinito amor.

De igual manera agradezco a Jimmy, quien ha estado en las buenas y las malas durante mi carrera universitaria, gracias por su apoyo, compañía, paciencia, cariño y por su amor incondicional.

A mis maestros y compañeros que han aportado con su ayuda, conocimientos, consejos y una grata estancia en las aulas de clase y de manera especial agradezco a quien fue parte de mi formación profesional Ing. María Soledad Jiménez MSc., quien indistintamente de haber sido mi directora de trabajo de titulación, llevo una gran admiración por el buen trabajo que desarrolla en la formación de profesionales, gracias por sus conocimientos, dedicación y esmero en esta carrera.

Finalmente agradezco a la Escuela Politécnica Nacional, por abrirme las puertas y darme la oportunidad de estudiar en su Facultad de Ingeniería en Eléctrica y Electrónica, en cuyas aulas he aprendido y me han enseñado lo que hoy viene a ser el punto de partida de un futuro exitoso.

Para ellos muchas Gracias y que Dios los bendiga.

Valeria Murillo Salazar

ÍNDICE DE CONTENIDO

AVAL	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
ÍNDICE DE CONTENIDO	V
ÍNDICE DE FIGURAS.....	IX
ÍNDICE DE TABLAS	XIV
RESUMEN.....	XV
ABSTRACT	XVI
1. INTRODUCCIÓN.....	1
1.1 Objetivos	2
1.2 Alcance	2
1.3 Marco Teórico	4
1.3.1 Laboratorios basados en Hardware versus laboratorios basados en Software 4	
1.3.2 Herramientas de Software	5
1.3.2.1 Matlab	5
1.3.2.2 Simulink.....	7
1.3.2.3 OrCAD.....	11
1.3.3 Procesamiento de señales y filtros digitales	16
1.3.3.1 Generalidades de análisis de señales	19
1.3.4 Sistemas de modulación y demodulación digital.....	23
1.3.5 Sistemas de transmisión multiportadora usando COFDM.....	30
1.3.5.1 Características del COFDM.....	30
1.3.5.2 Ventajas y desventajas de COFDM.....	31
1.3.5.3 Uso de múltiples portadoras	32
1.3.5.4 Diagrama de bloques del sistema COFDM.....	34
2. METODOLOGÍA.....	35

2.1	Guías de las herramientas de software	35
2.1.1	Guía 1: Una experiencia con procesamiento de señales usando Matlab	35
2.1.1.1	Objetivos	35
2.1.1.2	Descripción de guía	36
2.1.1.3	Desarrollo de guía	36
2.1.1.4	Conclusiones y recomendaciones	171
2.1.2	Guía 2: Una experiencia con digitalización de señales usando Simulink	44
2.1.2.1	Objetivos	44
2.1.2.2	Descripción de guía	45
2.1.2.3	Desarrollo de guía	45
2.1.2.4	Conclusiones y recomendaciones	52
2.1.3	Guía 3: Una experiencia aplicada a códigos de línea usando OrCAD Capture y PSpice	53
2.1.3.1	Objetivos	53
2.1.3.2	Descripción de guía	53
2.1.3.3	Desarrollo de guía	54
2.1.3.4	Conclusiones y recomendaciones	59
2.2	Procesamiento de señales y filtros digitales	60
2.2.1	Práctica 01: Introducción al procesamiento digital de señales	60
2.2.1.1	Objetivos	60
2.2.1.2	Marco teórico.....	60
2.2.1.3	Desarrollo práctico.....	62
2.2.2	Práctica 02: Análisis de señales (Series y Transformada de Fourier)	67
2.2.2.1	Objetivos	67
2.2.2.2	Objetivo específico	67
2.2.2.3	Marco teórico.....	67
2.2.2.4	Desarrollo práctico.....	70
2.2.3	Práctica 03: Diseño y simulación de un filtro FIR	79
2.2.3.1	Objetivos	79
2.2.3.2	Marco teórico.....	79

2.2.3.3	Desarrollo práctico.....	87
2.2.4	Práctica 04: Diseño y simulación de un filtro IIR	90
2.2.4.1	Objetivos	90
2.2.4.2	Marco teórico.....	90
2.2.4.3	Desarrollo práctico.....	94
2.3	Sistemas de modulación y demodulación digital	96
2.3.1	Práctica 05: Modulación y demodulación ASK.....	97
2.3.1.1	Objetivos	97
2.3.1.2	Marco teórico.....	97
2.3.1.3	Desarrollo práctico.....	99
2.3.2	Práctica 06: Modulación y demodulación FSK.....	106
2.3.2.1	Objetivos	106
2.3.2.2	Marco teórico.....	106
2.3.2.3	Desarrollo práctico.....	109
2.3.3	Práctica 07: Modulación PSK	120
2.3.3.1	Objetivos	120
2.3.3.2	Marco teórico.....	120
2.3.3.3	Desarrollo práctico.....	128
2.3.4	Práctica 08: Modulación QAM	134
2.3.4.1	Objetivos	134
2.3.4.2	Marco teórico.....	135
2.3.4.3	Desarrollo práctico.....	138
2.4	Sistema de transmisión multiportadora usando COFDM	141
2.4.1	Esquema de funcionamiento del transmisor COFDM	141
2.4.2	Esquema de funcionamiento de un receptor COFDM.....	143
2.4.3	Esquema de simulación en Simulink del sistema COFDM.....	144
3.	RESULTADOS Y DISCUSIÓN	146
3.1	Análisis de resultados sobre introducción al procesamiento digital de señales... 146	
3.2	Análisis de resultados sobre análisis de señales de Fourier.....	147
3.3	Análisis de resultados sobre diseño y simulación de un filtro FIR.....	148

3.4	Análisis de resultados sobre diseño y simulación de un filtro IIR	150
3.5	Análisis de resultados sobre modulación y demodulación ASK	152
3.6	Análisis de resultados sobre modulación y demodulación FSK	155
3.7	Análisis de resultados sobre modulación PSK.....	160
3.8	Análisis de resultados sobre modulación QAM.....	167
4.	CONCLUSIONES	170
5.	REFERENCIAS BIBLIOGRÁFICAS.....	179
6.	ANEXOS.....	183
	ANEXO I.....	183
	UNA EXPERIENCIA CON PROCESAMIENTO DE SEÑALES.....	183
	ANEXO II.....	188
	INTRODUCCIÓN AL PROCESAMIENTO DIGITAL DE SEÑALES	188
	ANEXO III.....	190
	DISEÑO Y SIMULACIÓN DE UN FILTRO FIR.....	190
	ANEXO IV.....	192
	DISEÑO Y SIMULACIÓN DE UN FILTRO IIR.....	192
	ANEXO V.....	194
	SCRIPT DEL ESQUEMA DE SISTEMA DE TRANSMISIÓN MULTIPORTADORA USANDO COFDM.....	194

ÍNDICE DE FIGURAS

MARCO TEÓRICO

Figura 1.1 Entorno de trabajo de Matlab	7
Figura 1.2 Entorno de trabajo de Simulink	8
Figura 1.3 Communications System Toolbox	10
Figura 1.4 Librería DSP System Toolbox	11
Figura 1.5 Componentes del paquete CADENCE v16.6	13
Figura 1.6 Inicio del software OrCAD Capture	14
Figura 1.7 Cuadro de diálogo para crear nuevo proyecto de PSpice	14
Figura 1.8 Cuadro de diálogo de opción para nuevo proyecto	15
Figura 1.9 Entorno de trabajo de OrCAD Capture.....	15
Figura 1.10 Estructura de un sistema PDS	17
Figura 1.11 Modulación ASK	25
Figura 1.12 Modulación OOK	26
Figura 1.13 Modulación FSK	26
Figura 1.14 Modulación PSK	27
Figura 1.15 Diagramas de constelación de un modulador QPSK y 8PSK	28
Figura 1.16 Diagrama de bloques de un modulador DPSK	28
Figura 1.17 Diagrama de sincronización de DPSK	29
Figura 1.18 Diagramas de constelación modulación M-QAM	29
Figura 1.19 Retraso largo en la trayectoria causa ISI	33
Figura 1.20 Retraso corto en la trayectoria causa ISI.....	33

METODOLOGÍA

Figura 2.1 Help de Matlab.....	37
Figura 2.2 Señales, 5[kHz] y 500[Hz].....	37
Figura 2.3 Suma de señales ($y_1 + y_2$)	38
Figura 2.4 Espectro de suma de señales.....	38
Figura 2.5 Características de un filtro Butterworth	39
Figura 2.6 Características paramétricas del filtro Butterworth paso bajo	40
Figura 2.7 Respuesta en (a) amplitud del filtro, frecuencia [rad/s]; (b) amplitud (dB) del filtro, frecuencia [rad/s]; (c) amplitud (dB) del filtro, frecuencia [Hz].....	42
Figura 2.8 Respuesta del filtrado de la señal, en el dominio de tiempo	44
Figura 2.9 Espectro de frecuencia de señal filtrada.....	44
Figura 2.10 Bloque de simulación de la señal seno	46
Figura 2.11 Esquema de suma de dos señales	47
Figura 2.12 Esquema para obtener la señal de voz de un dispositivo de audio	47
Figura 2.13 Bloque de simulación (a) correspondiente a From Audio Device, (b) correspondiente a To Multimedia File	47
Figura 2.14 Bloque de simulación de la señal diente de sierra.....	48
Figura 2.15 Esquema de digitalización, modulación PCM.....	48
Figura 2.16 (a)Señal de entrada, (b)muestreada y (c)cuantificada de SIGNAL_1 (Scope1)	50

Figura 2.17 Codificación de SIGNAL_1 (Scope2)	50
Figura 2.18 Current Directory de Matlab/ voicetest.wav	50
Figura 2.19 (a)Señal de entrada, (b)muestreada (c)cuantificada de SIGNAL_2 (Scope1)	51
Figura 2.20 Codificación de SIGNAL_2 (Scope2)	51
Figura 2.21 (a) Señal de entrada,(b)muestreada (c)cuantificada de SIGNAL_3 (Scope1)	52
Figura 2.22 Codificación de SIGNAL_3 (Scope2)	52
Figura 2.23 Cadence/Library/PSpice (Directorio)	55
Figura 2.24 Circuito codificador Manchester	55
Figura 2.25 Circuito codificador AMI-RZ	55
Figura 2.26 Creación del archivo Stimulus, Bits de entrada.	56
Figura 2.27 Elementos de la librería Source	56
Figura 2.28 Creación del perfil de simulación	57
Figura 2.29 (a) Datos de entrada y reloj del codificado Manchester (b) Datos de entrada y reloj del codificador AMI-RZ.....	57
Figura 2.30 (a) Codificación AMI-RZ (b)Codificación Manchester	58
Figura 2.31 Espectro de frecuencia,(a)Codificación Manchester,(b)Codificación AMI-RZ	58
Figura 2.32 Diagrama de bloques de un sistema LTI	62
Figura 2.33 Secuencia de entrada del sistema LTI	65
Figura 2.34 Respuesta impulsiva del sistema LTI	65
Figura 2.35 Respuesta a la salida del sistema LTI.....	66
Figura 2.36 Respuesta impulsiva, triangular	66
Figura 2.37 Respuesta a la salida del sistema LTI.....	66
Figura 2.38 Parámetros del elemento VPULSE	71
Figura 2.39 Generador de pulsos.....	72
Figura 2.40 Espectro del tren de pulsos, PW=50[us], PER=150[us].....	72
Figura 2.41 Tren de pulsos, PW=20[us], PER=150[us]	73
Figura 2.42 Espectro del tren de pulsos, PW=20[us], PER=150[us].....	73
Figura 2.43 Tren de pulsos, PW=20[us], PER=1000[s]	73
Figura 2.44 Espectro del tren de pulsos, PW=20[us], PER=1000[s].....	73
Figura 2.45 Filtro paso bajo RC [29]	74
Figura 2.46 Cuadro configuración de Output file options.....	75
Figura 2.47 Circuito sintetizador de onda cuadrada	76
Figura 2.48 En (a) se muestra los armónicos 1, 3, 5, 7,9,11,13 y 15 de acuerdo al circuito de la Figura 2.47, (b) muestra el resultado de la suma de los ocho armónicos de (a).....	77
Figura 2.49 Componentes de frecuencia de la salida del sintetizador	77
Figura 2.50 Salida de los Filtros paso bajo RC en el dominio del tiempo del (a) circuito generador de onda cuadrada (b) circuito sintetizador de onda cuadrada	78
Figura 2.51 Salida de los Filtros paso bajo RC en el dominio de la frecuencia del circuito (a) generador de onda cuadrada (b) sintetizador de onda cuadrada.....	78
Figura 2.52 Elementos básicos de una estructura [9].....	80
Figura 2.53 Estructura de forma directa, para M=5 [9]	81
Figura 2.54 Estructura de forma cascada, M=7 [9].....	81
Figura 2.55 Estructura de la forma fase lineal, (respuesta impulso simétrica)	81
Figura 2.56 Forma Frecuencia de muestreo, para M=4	82
Figura 2.57 Especificaciones de un filtro FIR, (a) absolutas(lineal)(b) relativas(logarítmica)	82
Figura 2.58 Ingreso de parámetros de diseño, script .m	88

Figura 2.59 Respuesta impulso que se desea diseñar.....	88
Figura 2.60 Respuesta impulso del filtro diseñado.....	89
Figura 2.61 Función ventana de Hamming, $M=33$	89
Figura 2.62 Respuesta en frecuencia del filtro diseñado.....	89
Figura 2.63 Respuesta en frecuencia del filtro diseñado.....	91
Figura 2.64 Método del impulso invariante.....	93
Figura 2.65 Método transformada bilineal.....	94
Figura 2.66 Ingreso de parámetros de diseño, script .m.....	94
Figura 2.67 Respuesta en frecuencia del filtro IIR diseñado, paso bajo.....	95
Figura 2.68 Respuesta impulso del filtro diseñado.....	96
Figura 2.69 Esquema de modulación 2ASK.....	98
Figura 2.70 Esquema demodulador ASK de la forma (a) no coherente (b) coherente.....	99
Figura 2.71 Esquema modulador ASK.....	99
Figura 2.72 (a) señal de datos $T_b=1\text{ ms}$, (b)señal portadora (c)espectro de la portadora.....	100
Figura 2.73 Señales (a) Señal de datos (b) señal modulada ASK.....	101
Figura 2.74 Espectro (a) Señal de datos (b) Señal modulada ASK.....	101
Figura 2.75 Señales (a) Señal de datos (b) señal modulada ASK.....	102
Figura 2.76 Espectro (a) Señal de datos (b) Señal modulada ASK.....	102
Figura 2.77 Señales (a) Señal de datos (b) señal modulada ASK.....	103
Figura 2.78 Espectro (a) Señal de datos (b) Señal modulada ASK.....	103
Figura 2.79 Señales (a) Señal de datos (b) señal modulada ASK.....	104
Figura 2.80 Espectro (a) Señal de datos (b) Señal modulada ASK.....	104
Figura 2.81 Diagrama de Modulación/Demodulación ASK.....	105
Figura 2.82 Pasos de demodulación no coherente (a) Modulación OOK (b) Señal rectificada (c) Detector de envolvente(d) Señal demodulada (datos).....	106
Figura 2.83 Esquema de modulación 2-FSK de fase continua.....	107
Figura 2.84 Demodulador FSK no coherente.....	109
Figura 2.85 Demodulador FSK coherente.....	109
Figura 2.86 Esquema de modulación FSK.....	110
Figura 2.87 (a) Señal de datos con $T_b=2\text{ ms}$ (b) Señal modulada 2FSK.....	110
Figura 2.88 Espectro de frecuencia de la señal modulada 2-FSK.....	111
Figura 2.89 Respuesta en frecuencia de un filtro pasa banda centrado en W_{cf} (frecuencia central del filtro).	112
Figura 2.90 Prototipo de filtro paso banda.....	114
Figura 2.91 Bloque de filtros paso banda.....	118
Figura 2.92 Señal recibida de los filtros pasabanda.....	118
Figura 2.93 Detector de envolvente.....	119
Figura 2.94 Señales resultantes del bloque detector de envolvente, para la línea del circuito del 0L (rojo) y del 1L (azul).....	119
Figura 2.95 Señal demodulada.....	119
Figura 2.96 Modulación B-PSK.....	120
Figura 2.97 Diagramas de constelación de B-PSK, Q-PSK y 8-PSK.....	121
Figura 2.98 Diagrama de bloques del modulador PSK.....	122
Figura 2.99 Modulador 2-PSK.....	122
Figura 2.100 Modulador M-PSK.....	123
Figura 2.101 Modulador Q-PSK.....	124
Figura 2.102 Modulación QPSK (b) Diagrama fasorial (c) Diagrama de constelación]...]	125

Figura 2.103 Señal a la salida del modulador QPSK	125
Figura 2.104 Modulador 8-PSK	126
Figura 2.105 Diagrama de constelación 8-PSK	128
Figura 2.106 Modulador B-PSK	129
Figura 2.107 Coeficiente de acoplamiento (a) 0.85 (b) 0.95.....	130
Figura 2.108 (a) Señal portadora (b) Datos (c) Modulación B-PSK.....	130
Figura 2.109 Espectro de frecuencia de (a) portadora (b) la modulación B-PSK.....	131
Figura 2.110 Modulador Q-PSK.....	131
Figura 2.111 (a) Datos del canal I (b) Datos del canal Q (c) Modulación Q-PSK.....	132
Figura 2.112 Espectro de frecuencia (a) portadora canal I y canal Q (b) modulación Q-PSK	132
Figura 2.113 Modulador 8-PSK.....	133
Figura 2.114 Señales a la (a) entrada de los conversores y (b) salida de los conversores de 2 a 4 niveles	133
Figura 2.115 Configuración del convertidor de nivel de 2 a 4.....	134
Figura 2.116 (a) Señal a la salida del convertidor de 2 a 4 niveles (rojo-canal I);.....	134
Figura 2.117 Modulación 8-QAM y su diagrama de constelación.....	135
Figura 2.118 Modulador 8-QAM.....	136
Figura 2.119 Modulador 8-QAM.....	138
Figura 2.120 Esquema de modulación 8-QAM.....	139
Figura 2.121 Señales a la (a) entrada y (b) salida del convertidor de 2 a 4 niveles.....	139
Figura 2.122 Señal portadora (rojo-canal I; azul –canal Q); y su respectivo espectro de frecuencia.....	140
Figura 2.123 (a) Salida del canal I(rojo) y canal Q(azul), Referenciar Figura 2.120 (b) Modulación 8-QAM.....	140
Figura 2.124 Espectro de frecuencia de la señal modulada.....	141
Figura 2.125 Transmisor COFDM	142
Figura 2.126 Señal COFDM de salida	142
Figura 2.127 Receptor COFDM	143
Figura 2.128 Sistema de modulación usando COFDM	144
Figura 2.129 Datos de audio de entrada modulados en 4- PSK.....	145
Figura 2.130 Decodificación de datos para obtención de audio ingresado.....	145

RESULTADOS Y DISCUSIÓN

Figura 3.1 Respuesta impulso que se desea diseñar.....	148
Figura 3.2 Función ventana de Bartlett, $M=67$	149
Figura 3.3 Respuesta impulso del filtro diseñado.....	149
Figura 3.4 Respuesta en frecuencia del filtro diseñado.....	149
Figura 3.5 Respuesta en frecuencia del filtro diseñado.....	151
Figura 3.6 Respuesta Impulso – método bilineal (para datos variación 1).....	151
Figura 3.7 Respuesta en frecuencia del filtro diseñado.....	151
Figura 3.8 Respuesta Impulso – método bilineal (para datos variación 2).....	152
Figura 3.9 Señal de datos (a) a la entrada del modulador OOK (b) a la salida del demodulador OOK.....	154
Figura 3.10 (a) Señal de datos, (b) Modulación 2-FSK	156
Figura 3.11 Espectro de frecuencia de una modulación FSK de una señal binaria	156

Figura 3.12 Modulación FSK de una señal binaria.....	156
Figura 3.13 Espectro de frecuencia de la modulación FSK.....	157
Figura 3.14 Forma de onda a la salida de los filtros pasa banda (a) para 0L, (b) para 1L, (c) espectro de (a), (d) espectro de (b)	158
Figura 3.15 Proceso de rectificación (a) para BANDA_0L, (b) para BANDA_1L	159
Figura 3.16 Señal demodulada	159
Figura 3.17 (a) Señal ENVOL_0L y ENVOL_1L, (b) Señal demodulada con parámetros erróneos de diseño	160
Figura 3.18 (a) Datos, (b) señal modulada BPSK, (c) espectro de la modulación BPSK.....	161
Figura 3.19 Modulación considerando un Coef=0.85	162
Figura 3.20 Modulación considerando un Coef=0.6.....	162
Figura 3.21 (a) Señal portadora $V_p=8$ [V] y señal de datos (rojo) $V=10$ [V], (b) Modulación B-PSK	163
Figura 3.22 (a) Señal portadora $V_p=2$ [V], (b) señal de datos $V=12$ [V], (c) Modulación BPSK.....	163
Figura 3.23 (a) portadora,(b)datos,(c) Modulación ASK, utilizando un esquema B-PSK.....	164
Figura 3.24 Espectro de frecuencia del modulador PSK	164
Figura 3.25 (a) Portadora, $f_p=20$ [KHz], (b) Datos, (c) Modulación B-PSK.....	165
Figura 3.26 (a) Espectro de la portadora $f_p=20$ [KHz], (b) Espectro de frecuencia de la señal modulada BPSK.....	165
Figura 3.27 (a) Datos y señales a la salida del canal I, (b) Datos y señal a la salida del canal Q, del modulador QPSK	166
Figura 3.28 Diagrama de constelaciones de la modulación 8-PSK.....	167
Figura 3.29(a)Señal a la salida del canal Q y canal I,(b) Modulación 8-QAM.....	168
Figura 3.30 Diagrama de constelaciones de la modulación 8-QAM	169

ÍNDICE DE TABLAS

METODOLOGÍA

Tabla 2.1 Parámetros iniciales de señal.....	37
Tabla 2.2 Comandos para diseño de filtros digitales	39
Tabla 2.3 Tipos de filtros Butterworth.....	43
Tabla 2.4 Tabla de valores de las componentes de Fourier de la respuesta v(PULSE) ...	75
Tabla 2.5 Tabla de tipos de filtros de fase lineal	83
Tabla 2.6 Respuestas de diseño de los tipos de filtros	84
Tabla 2.7 Tipos de funciones ventana	84
Tabla 2.8 Características de las funciones ventana	85
Tabla 2.9 Filtros analógicos	92
Tabla 2.10 Datos paramétricos para diseño de filtro paso banda Chebychev, con frecuencia central la frecuencia de espacio.	114
Tabla 2.11 Datos paramétricos para diseño de filtro paso banda Chebychev, con frecuencia central la frecuencia de marca.....	115
Tabla 2.12 Tabla de polinomios normalizados de Chebychev para filtro pasa bajo	115
Tabla 2.13 Tabla de verdad, Modulación de 4-PSK	125
Tabla 2.14 Tabla de verdad convertidor 2 a 4	126
Tabla 2.15 Tabla de verdad, Modulación de 8-PSK	127
Tabla 2.16 Tabla de verdad convertidor 2 a 4	137
Tabla 2.17 Tabla de verdad, Modulación 8-QAM	137

RESUMEN

El presente trabajo de titulación tiene como objetivo diseñar e implementar un compendio de experiencias de simulación estructuradas en forma de prácticas de laboratorio para cátedras relativas a sistemas de comunicaciones, teniendo como referencia los temas de estudio de pregrado en la carrera de Ingeniería Electrónica y Telecomunicaciones. Las herramientas de software a usar son Matlab, Simulink y OrCAD PSpice, con las cuales se abordarán temas de tratamiento y filtrado de señales, transformadas de Fourier, filtros FIR e IIR, técnicas de modulación y demodulación digitales (ASK, FSK, PSK y QAM) y sistemas de comunicaciones multiportadora COFDM (*Coded Orthogonal Frequency Division Multiplexing*). En cada uno de estos tópicos se iniciará con una introducción conceptual y práctica de los temas a tratar, luego se planteará ejercicios en función del marco teórico expuesto, para después continuar con un análisis de resultados y conclusiones.

Es importante destacar que hoy en día la formación de ingenieros en telecomunicaciones, no se basa solamente en un aprendizaje teórico por medio de cálculos numéricos, papel y lápiz; pues ahora viene su formación conectada con el manejo de herramientas de simulación, lenguajes de programación y técnicas de experimentación a través del uso de hardware, y gracias a las enormes revoluciones tecnológicas, en software. En Ecuador no existe un suministro amplio de tecnología, elementos electrónicos o kits de desarrollo, por lo cual el uso de software es una opción prometedora sin necesidad de inversiones económicas respecto al uso de hardware.

Actualmente, la simulación de bloques funcionales está ganando cada vez más espacio, lo cual, Universidades, Centros de Investigación y Empresas usan para simular sus prototipos, previo a trasladar el diseño a una plataforma de hardware.

Este hecho respalda el uso del software en el diseño de sistemas complejos de ingeniería, por consecuencia directa, también para el aprendizaje en la formación de ingeniería. El compendio de prácticas presentado en este trabajo con experiencias de laboratorio espera dejar plantado el inicio de un nuevo paradigma de enseñanza en los laboratorios de la Facultad de Ingeniería Eléctrica y Electrónica de la Escuela Politécnica Nacional.

ABSTRACT

The objective of this present work is to design and implement a compendium of structured simulation experiences in the form of laboratory practices for subjects related to communication systems, taking as reference the topics of undergraduate study in Electronic Engineering and Telecommunications career. The software tools to be used are Matlab, Simulink and OrCAD PSpice, in which to cover topics about treatment and filtering of signals, Fourier transform, FIR and IIR filters, modulation and demodulation techniques (ASK, FSK, PSK y QAM) and multi-carrier communication systems COFDM (Coded Orthogonal Frequency Division Multiplexing). In each of these topics, it will begin with a conceptual and practical introduction of the topics to be discussed, then exercises will be proposed according to the theoretical framework presented, and then continue with analysis of results and conclusions.

It's important to note that nowadays the training of telecommunications engineers, isn't based solely on theoretical learning to through numerical calculations, paper and pencil, now it's connected with the use of simulation tools, programming languages and experimentation techniques through the use of hardware, and thanks to the big technological revolutions in software. In Ecuador there isn't supply of technology, electronic elements or development kits, so the use of software is a promising option without the need for economic investments regarding the use of hardware.

At the present time, the simulation of functional blocks is getting more and more space, so Universities, Research Centers and Companies use to simulate their prototypes, before moving the design to a hardware platform.

The fact supports the use of software in the design of complex engineering systems, as a direct consequence, also for learning in engineering training. The compendium of practices presented in this work with laboratory experiences, hopes to leave planted the beginning of a new paradigm of teaching in the laboratories of the Faculty of Engineering in Electrical and Electronics of the National Polytechnic School.

1. INTRODUCCIÓN

Los sistemas de comunicaciones son hoy por hoy tan importantes que fácilmente pueden ser considerados como los pilares fundamentales de la sociedad actual. Por medio de ellos es posible realizar tareas como enviar un correo electrónico, monitorear sensores remotos, procesar grandes cantidades de información de forma conjunta entre varios centros de datos, entre otras. Todas estas tareas se basan en una misma idea, la transmisión de información, la misma que pueden viajar utilizando un medio de transmisión el cual puede ser un medio guiado (fibra óptica, par trenzado, cable coaxial) y no guiado (radio enlaces, microondas, etc.); de manera que permiten encaminar y controlar la información.

Para la transmisión de información son necesarios ciertos bloques funcionales, los cuales están basados en conceptos matemáticos y hardware electrónico. Estos bloques funcionales constituyen los sistemas de comunicaciones actuales en el vivir diario de la sociedad actual, industria, investigación, etc.

Hoy en día los sistemas de comunicaciones están formados por una interacción tan compleja de subsistemas y bloques funcionales, los cuales pueden resultar muy difíciles de entender de forma teórica o implementar directamente en hardware. Lenguajes de programación, electrónica, algorítmica, matemáticas y demás ciencias trabajan conjuntamente para dar vida a un sistema de comunicaciones. De este modo, existen un sin número de herramientas computacionales mediante las cuales es posible de manera virtual, ensamblar desde pequeñas secciones de un sistema de comunicaciones complejo, hasta un sistema de comunicaciones completo.

También es posible realizar simulación y ensayos de cómo este sistema responderá a ciertos escenarios críticos, aprovechando de esta manera la enorme potencialidad de los paquetes de herramientas computacionales que hacen cada vez más fiables los resultados en el entorno virtual. Por este motivo, empresas generadoras de tecnología y universidades hacen uso de simuladores y plataformas computacionales donde se facilita enormemente el entendimiento, simulación e implementación de sistemas de comunicaciones, primero en software luego en hardware.

La implementación en software tiende a ser más eficiente desde el punto de vista del rendimiento y complejidad de los objetivos a alcanzar, siendo este, el principal problema de los laboratorios en hardware. El alto costo de la instrumentación, acompañado con el deficiente mercado tecnológico, la poca disponibilidad de soluciones asequibles y el precario campo a nivel nacional en el Ecuador en el desarrollo de tecnología, limita el

progreso de laboratorios a realizar experimentos que tienen que ver con la realidad y retos a nivel laboral con los que se enfrentará el ingeniero graduado. Los retos laborales del ingeniero en comunicaciones (especialidades de telecomunicaciones, redes de la información y telemática) en el mercado laboral nacional están enfocados en la integración de tecnología de manera optimizada tanto económica como a nivel de recursos.

1.1 Objetivos

El objetivo general de este trabajo de titulación es:

Esquematizar y desarrollar mediante software de simulación prácticas de laboratorio de Sistemas de Comunicaciones usando herramientas Matlab, Simulink y OrCAD PSpice, enfocadas en el tratamiento y filtrado de señales, técnicas de modulación y demodulación digital y sistemas de comunicaciones multiportadora.

Los objetivos específicos de este trabajo de titulación son:

- Esquematizar y desarrollar mediante software de simulación prácticas de laboratorio orientadas al manejo de señales y sus principales operaciones de caracterización, series y transformadas de Fourier, filtros FIR e IIR.
- Esquematizar y desarrollar mediante software de simulación prácticas de laboratorio orientadas a abordar escenarios de experimentación relativos a técnicas de modulación digital (ASK, FSK, PSK y QAM).
- Esquematizar y desarrollar mediante software de simulación un esquema orientado a comprender el funcionamiento, estructura, principales características, ventajas y desventajas de los sistemas de transmisión multiportadora usando COFDM (*Coded Orthogonal Frequency Division Multiplexing*).
- Analizar los resultados de las prácticas de laboratorio realizadas, a través de las gráficas en función del tiempo y/o frecuencia obtenidas de la simulación de cada esquema.

1.2 Alcance

Este trabajo de titulación está orientado a afianzar los conocimientos teóricos relacionados con los sistemas de comunicaciones, que hoy en día han alcanzado altos niveles de complejidad, y la única manera de comprender adecuadamente cómo interactúan cada uno de sus elementos, es la experimentación continua en laboratorios especializados.

Comprender la importancia del procesamiento de señales, técnicas de modulación y demodulación, técnicas avanzadas de transmisión multiportadora y su aplicación a problemas prácticos, resulta esencial para la formación de un ingeniero de comunicaciones.

Este trabajo de titulación se basará en la esquematización y desarrollo de prácticas usando software de simulación especializado en el diseño de sistemas de comunicaciones, Matlab, Simulink y OrCAD PSpice. Como primera instancia se desarrollarán pequeñas guías de uso para familiarizar al estudiante con estas herramientas de simulación y para destacar la importancia y ventajas.

Los escenarios de simulación estarán basados en esquemas donde se aborda temáticas de series de Fourier, la transformada de Fourier; filtrado digital de señales, filtros FIR e IIR. Además, se analizarán y esquematizarán técnicas de modulación y/o demodulación digital, ASK, FSK, PSK y QAM. Finalmente se realizará un esquema de sistema de comunicaciones usando COFDM (*Coded Orthogonal Frequency Division Multiplexing*) en Simulink.

El objetivo de estas simulaciones no es abordar las prácticas a desarrollar bajo la idea de solucionar problemas de extremada complejidad, ya que en la etapa de aprendizaje esto más que ayudar a asimilar las ideas y conceptos fundamentales de los sistemas de comunicaciones, puede desembocar en complicar mucho su estudio, uso y la comprensión de estos temas. El objetivo es la pedagogía y el promover el entendimiento de las bases de los sistemas de comunicaciones mediante la experimentación interactiva.

Cada una de las prácticas estará estructurada con objetivos, marco teórico, esquema práctico en el cual se realizará el planteamiento y desarrollo del escenario a simular, análisis de resultados obtenidos, conclusiones y recomendaciones; además se plantearán posibles extensiones a los escenarios analizados.

Matlab, Simulink y PSpice funcionarán como plataformas de simulación basada en el uso de modelos y herramientas especializadas y optimizadas para obtener resultados profesionales en estudio y caracterización de sistemas de comunicaciones. En este punto hay que hacer hincapié en que se usarán las herramientas cuyo desarrollo ha tomado varios años para ser aprovechado a nivel industrial y pedagógico, no se dedicará tiempo a programar ni reimplementar herramientas, que de ningún modo funcionarán igual que las herramientas incluidas en Matlab y Simulink.

1.3 Marco Teórico

En la presente sección se plasmará un marco teórico introductorio de las temáticas a ser abordadas en este trabajo de titulación, el mismo que se enfocará en el marco teórico de cada una de las prácticas propuestas.

A lo largo del tiempo por la necesidad de optimización económica y usabilidad, se han ido planteando propuestas pedagógicas para el uso de laboratorios virtuales como actividad complementaria en asignaturas teóricas en distintas entidades de formación de profesionales, destacando características de simulación en diferentes áreas prácticas. El presente documento analizará las ventajas y desventajas de las herramientas de software propuestas para el uso en un laboratorio basado en software, que para muchos estudiantes son herramientas conocidas en el campo de la ingeniería electrónica ya que son parte de su formación, *Matlab*, *Simulink* y en algunos casos *OrCAD PSpice*. Estos recursos tienen gran potencial, que más allá de ser una herramienta matemática, amplía sus funcionalidades para realizar simulaciones en varios campos de la ingeniería como análisis de un sinnúmero de sistemas; pero aún no se ha utilizado por parte de la gran mayoría de docentes a pesar de su disponibilidad en su versión estudiantil por *The Mathworks* [1].

Es así que este trabajo de titulación, busca aportar a la comunidad universitaria una alternativa para realizar prácticas usando tres herramientas de software de simulación, que a diferencia de otro trabajo presentado, solamente se ha propuesto con una herramienta usando códigos de línea, *Systemview*.

1.3.1 Laboratorios basados en Hardware versus laboratorios basados en Software

En la actualidad existe una tendencia a la masificación del uso de las tecnologías de la información en el aprendizaje, remarcada desde la llegada de las computadoras de mejores capacidades de cálculo, donde las herramientas de software han ganado gran popularidad.

Hoy en día los simuladores poseen características muy avanzadas que acercan cada vez más las experiencias de simulación a la realidad de trabajar con hardware. Es un hecho que los laboratorios en hardware proporcionarán una experiencia más real y reforzada a las etapas de aprendizaje de los estudiantes de ingeniería, sin embargo, depender de factores externos limita las experiencias en los laboratorios basados en hardware.

Desde otro punto de vista, el disponer de financiamiento tampoco garantiza que el laboratorio estará siempre equipado y cien por ciento funcional, ya que de una u otra forma

se necesitarán realizar importaciones de instrumentación o a su vez de insumos para actividades de mantenimiento y repotenciación de las instalaciones, lo cual, bajo el escenario ecuatoriano puede tomar desde unas cuantas semanas a varios meses.

Además, la implementación de sistemas complejos en hardware, bajo ciertas circunstancias de experimentación en laboratorios para ingenieros en el aprendizaje, implica factores importantes como errores de uso, daños en equipos, adquisición de instrumentación y por ende tiempo, significando inversión continua en reposición y reemplazo de instrumental, actualización, entre otros gastos adicionales. Bajo este análisis, el uso de herramientas de software resulta una alternativa necesaria para evitar caer en este tipo de limitaciones y enfrentar equivocaciones sin echar a perder equipamiento y daño de instrumentación. Es por tal razón que los laboratorios basados en software, a pesar de no poseer las mismas fortalezas de los laboratorios basados en hardware, tienen una serie de ventajas, que para los estudiantes de ingeniería en Ecuador resultan grandiosas sin necesidad de una elevada inversión económica.

El presente trabajo de titulación se enfocará al uso de herramientas de software en una propuesta de laboratorio para Comunicaciones Digitales, cuyo objetivo se enfatiza en mejorar la calidad de aprendizaje optimizando tiempo y costo. Además, esta propuesta queda abierta al estudiante para que demuestre sus habilidades y aptitudes llegando a mejorar aún más cada experiencia contenida en este proyecto. Es así que este trabajo queda disponible para que se pueda eventualmente utilizar en un Laboratorio de Comunicaciones Digitales en la Escuela Politécnica Nacional.

1.3.2 Herramientas de Software

En esta sección se hará una descripción de las herramientas de software, *Matlab*, *Simulink* y *OrCAD PSpice*, que se utilizarán durante el desarrollo de este trabajo de titulación y permitirán experimentar escenarios prácticos; generando líneas de códigos, esquematización y simulación de diseños de circuitos para áreas de sistemas de comunicaciones. Donde mediante pruebas se garantizará el funcionamiento; abordando de esta manera los objetivos que presenta este trabajo en el aprendizaje y aplicación de estas herramientas de software en la ingeniería.

1.3.2.1 Matlab

The Mathworks presenta *Matlab* como un software computacional interactivo y amigable, que ha ganado gran popularidad en el campo de la ciencia e ingeniería, convirtiéndose en un estándar orientado al cálculo numérico, análisis y visualización de datos, desarrollo de

algoritmos, simulación de sistemas de comunicaciones, ingeniería de software, biología computacional, visión artificial, matemáticas aplicadas, entre otras, enfocando al desarrollo de proyectos de ingeniería proporcionando así un entorno adaptado a trabajos interactivos, simultaneidad en el manejo de datos y programas favoreciendo la optimización de tiempo. Además, está disponible para las plataformas Unix, Windows (32-bits; 64-bits), Mac OS X y GNU/Linux [2] [3].

Matlab es un software pagado y por lo tanto requiere una licencia de activación; dado que en las Universidades esta herramienta ha sido necesaria se mantienen convenios con *The Mathworks* para usar en laboratorios virtuales.

Existen dos versiones de *Matlab* disponibles comercialmente, estudiantil y profesional. La versión estudiantil tiene los *toolboxes* básicos y *Simulink*. La versión profesional incluye *toolboxes* estándar, sobre los cuales se puede instalar nuevos *toolboxes* para ampliar las funcionalidades de *Matlab*.

Entre las aplicaciones más relevantes, *Matlab* permite [3]:

- Integrar con lenguajes como C/C++, Java, .NET y Python, los mismos que se pueden ejecutar directamente desde Matlab.
- Generar *scripts*, en lenguaje propio el cual se puede ejecutar desde (archivos .m).
- Crear interfaces gráficas de usuario.
- Visualizar datos en gráficos bi y tridimensionales avanzados.
- Trabajar conjuntamente con herramientas complementarias como *toolboxes*.
- Simular, por medio de una interfaz basada en bloques funcionales, Simulink.
- Resolver problemas de procesamientos de señales, diseño de sistemas de control, sistemas dinámicos, análisis de técnicas de modulación y demodulación, etc.

Es importante destacar que, *Matlab* cuenta con un sistema de ayuda, donde detalla la sintaxis de comandos deseados, mediante ejemplos simples y explicativos. Para acceder al sistema de ayuda se puede ejecutar el comando "*help*" en la ventana de comandos el cual despliega una lista general de temas de ayuda o también mediante el ícono de acceso directo.

Entorno de trabajo de Matlab

Al acceder a la herramienta *Matlab* por su ícono de acceso directo, se presenta la interfaz gráfica compuesta por ciertas ventanas principales dando lugar al entorno de trabajo como se muestra en la captura de pantalla de la Figura 1.1.

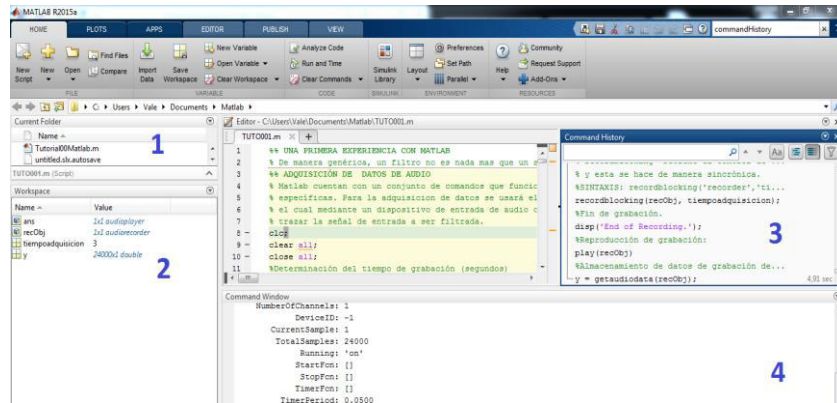


Figura 1.1 Entorno de trabajo de Matlab

- 1. *Current Folder* (Carpeta de directorio actual), se refiere al directorio del trabajo actual, donde se encuentran los ficheros que se están ejecutando.
- 2. *Workspace* (Ventana de trabajo), contiene las variables creadas de una función y su respectiva información, por lo tanto en esta ventana es posible suprimir o realizar posibles modificaciones.
- 3. *Command History* (Historial de comandos), lleva un registro de las declaraciones que se han ejecutado en la ventana de comandos en sesiones actuales o anteriores con hora y fecha.
- 4. *Command Window* (Ventana de comandos), se usa para ejecutar instrucciones u órdenes, permite mostrar resultados y errores que se presentan al compilar ficheros.

1.3.2.2 Simulink

Simulink es una herramienta de simulación que funciona sobre *Matlab*. Este entorno visual cuenta con varias librerías, que contienen componentes de simulación como, bloques para medición, visualización, procesamiento y monitoreo en sistemas de comunicaciones mediante el uso de una de las librerías como *Communications System Toolbox*.

Simulink presenta una serie de ventajas para estudiantes de Ingeniería en Electrónica y Telecomunicaciones, enfocado en el aprendizaje de procesamiento digital de señales, comunicación digital, adquisición de datos, sistemas embebidos, etc.

Entre las aplicaciones relevantes están:

- Permitir examinar resultados mediante un depurador de *Simulink*, donde es posible retroceder y avanzar paso a paso por los bloques de simulación y así analizar el comportamiento del mismo.

- Permitir visualizar resultados de simulación mediante pantallas y *scopes*.
- Proporcionar una interfaz para simulación de diseños usando bloques predefinidos localizados en sus librerías.
- Proporcionar compatibilidad integrada para la creación y ejecución de prototipos o modelos en plataformas de hardware de bajo coste como los arduinos.
- Crear y modelar sistemas de procesamiento de señales en tiempo continuo y discreto.

Entorno de trabajo de Simulink

Es posible acceder a *Simulink* en la interfaz de *Matlab* por medio de la ejecución del comando *simulink*, o pulsando el icono de acceso directo localizado en la barra de menú. El entorno de trabajo se muestra en la captura de pantalla de la Figura 1.2.

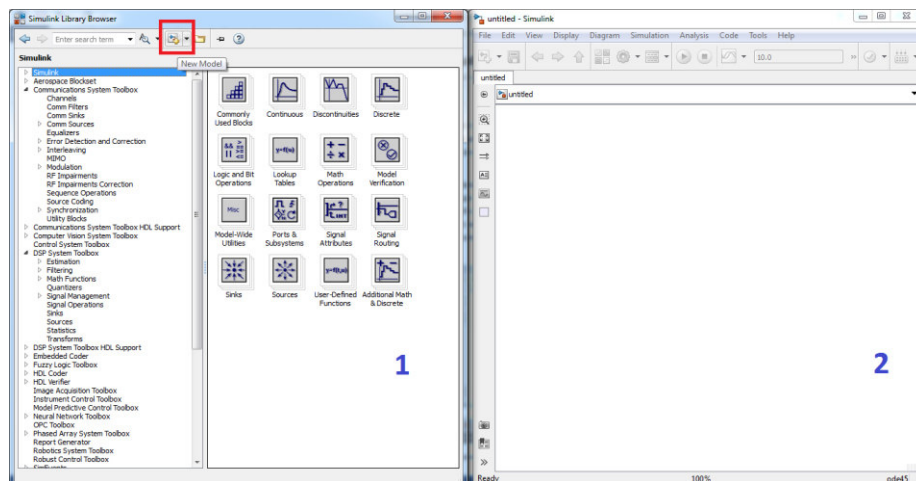


Figura 1.2 Entorno de trabajo de Simulink

- 1. *Simulink Library Browser* (Ventana de navegación de bloques de Simulink), tiene la funcionalidad de localizar en las distintas librerías de bloques, los elementos disponibles para simulación, diseño y modelado de sistemas.
- 2. Espacio de trabajo de Simulink, esta ventana se despliega al momento de dar clic en el ícono *New Model*, y es el espacio disponible para la esquematización de los modelos de simulación, insertando los bloques de la librería.

Communications System Toolbox

Se presenta como la librería que tiene herramientas que permiten realizar análisis, diseño, simulación y verificación de sistemas de comunicaciones en Simulink y Matlab. Cuenta con una serie de bloques funcionales que permiten realizar análisis y mediciones de parámetros

como diagramas de constelación, diagramas del ojo y BER¹. Algoritmos para diseño de capa física de sistemas de comunicaciones como OFDM², MIMO³, codificación de canal, modulación, ecualización y sincronización [4].

Los algoritmos proporcionados por *Communications System Toolbox* para componentes de transmisión y recepción cuentan con parámetros definidos, los cuales pueden ser modificados en base a los requerimientos de los respectivos esquemas, de esta manera se evita la creación de bloques que en su conjunto cumplan las mismas funciones. Los algoritmos de esta librería incluyen [5]:

- Generadores de señal y operadores
 - ✓ Generador de datos binarios
 - Barker, Gold, Hadamard, Kasami
 - ✓ Dispositivos de entrelazado y codificadores
- Modulación
 - ✓ Digital: FSK⁴, PSK⁵, DPSK⁶, MSK⁷, PAM⁸, QAM⁹, TCM¹⁰ y PCM¹¹.
 - ✓ Analógica: AM¹², FM¹³, PM¹⁴, SSB¹⁵, DSBSC¹⁶ y Broadcast FM.
- Técnicas multiportadora como:
 - ✓ Modulación y demodulación OFDM.
 - ✓ Ecualización de canal.
- Codificación de canal, detección y corrección de errores.
 - ✓ Convolutacional
 - Codificador convolutacional, Decodificador Viterbi, etc.
 - ✓ Comprobador de redundancia cíclica (CRC, CRC-N)

¹ BER: *Bit Error Rate*, (Tasa de bits errados).

² OFDM: *Orthogonal Frequency Division Multiplexing*, (Multiplexación por División de Frecuencias Ortogonales).

³ MIMO: *Multiple-input Multiple-output*, (Múltiple entrada múltiple salida).

⁴ FSK: *Frequency Shift Keying*, (Modulación por Desplazamiento de Frecuencia).

⁵ PSK: *Phase Shift Keying*, (Modulación por Desplazamiento de Fase).

⁶ DPSK: *Differential Phase Shift Keying*, (Modulación por Desplazamiento Diferencial de Fase).

⁷ MSK: *Minimum shift keying*, (Modulación por Desplazamiento Mínimo).

⁸ PAM: *Pulse Amplitude Modulation*, (Modulación por Amplitud de Pulsos).

⁹ QAM: *Quadrature Amplitude Modulation*, (Modulación por Amplitud en Cuadratura).

¹⁰ TCM: *Trellis Coded Modulation*, (Modulación codificado de Trellis).

¹¹ PCM: *Pulse Coded Modulation*, (Modulación por pulsos Codificados).

¹² AM: *Amplitude Modulation*, (Amplitud Modulada).

¹³ FM: *Frequency Modulation*, (Frecuencia Modulada).

¹⁴ PM: *Phase Modulation*, (Fase Modulada).

¹⁵ SSB: *Single Side Band*, (Banda lateral única).

¹⁶ DSBSC: *Double Sideband Suppressed Carrier*, (Doble Banda Lateral con portadora suprimida).

- ✓ Codigos bloque
 - Reed-Solomon, BHC¹⁷, LDPC¹⁸

La ventaja que presenta, para probar en hardware los diferentes modelos de implementación, es que tal cual como se lo hace en software con la integración de modelos de código generados en *Matlab*, se puede trasladar a plataformas de FPGAs¹⁹, o *kits* de desarrollo. La Figura 1.3 muestra las librerías de *Communication System Toolbox*.

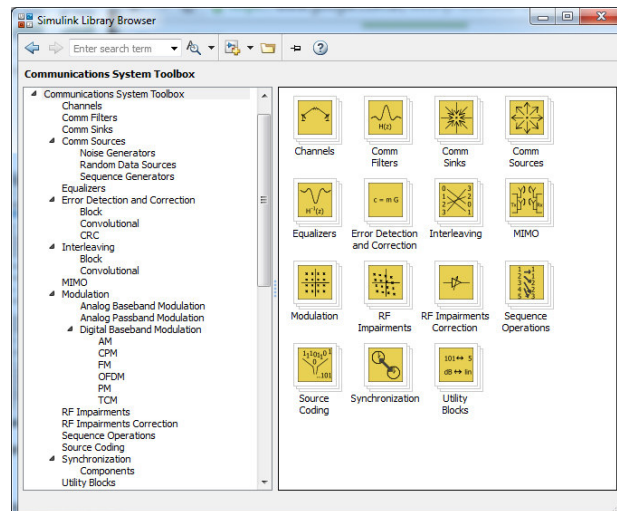


Figura 1.3 Communications System Toolbox

Digital Signal Processing System Toolbox

Las funciones que ofrece esta herramienta es compensar, normalizar, medir, transformar, filtrar, visualizar y generar señales; incluye algoritmos para realizar muestreo, *suavizado* y sincronización de señales. Cuenta con ciertas características que son herramientas, algoritmos y funciones para procesamiento de señales como: [6]

- Modelado predictivo, paramétrico y lineal.
- Diseño y análisis de filtros de tipo IIR²⁰ y FIR²¹.
- Estimación de espectros de potencia, medición de picos, ancho de banda y distorsión.
- En el espectro de frecuencia, medición de ancho de banda, armónicos, SNR²², etc.

¹⁷ BHC: *Bose-Chaudhuri Hocquenghem*

¹⁸ LDPC: *Low Density Parity Check*. (Comprobación de paridad de baja densidad)

¹⁹ FPGA: *Field Programmable Gate Array*, (Arreglo de compuertas programables).

²⁰ IIR: *Infinite Impulse Response*, (Respuesta infinita al impulso).

²¹ FIR: *Finite Impulse Response*, (Respuesta finita al impulso).

²² SNR: *Signal to Noise Ratio*, (Relación señal a ruido).

- En análisis y diseño de filtros digitales, permite implementar filtros IIR y FIR, como pasa bajos, pasa altos y supresión de banda; con los que se puede visualizar ciertos parámetros y características como magnitud, fase, retardos, respuestas impulso, escalón, evaluación de rendimiento, análisis de polos y ceros.
- En análisis y diseño de filtros analógicos, las herramientas soportan funciones de discretización como impulso invariante y métodos de transformación bilineal para la conversión de filtros analógicos a digitales.
- Análisis y comparación de señales en dominios de tiempo y frecuencia para identificación de patrones y tendencias para validación de datos.
- Procesar señales aplicando FFT²³ y STFT²⁴.
- Medición de ciertos rasgos distintivos claves de una señal como máximos, mínimos, valores pico-pico de amplitud, valores RMS²⁵ de señales, tiempo de subida y métricas de transición.

La Figura 1.4 muestra las distintas librerías que se encuentran en *DSP²⁶ System Toolbox* para esquematizaciones de estimaciones, filtrado, cuantificadores, fuentes, etc.

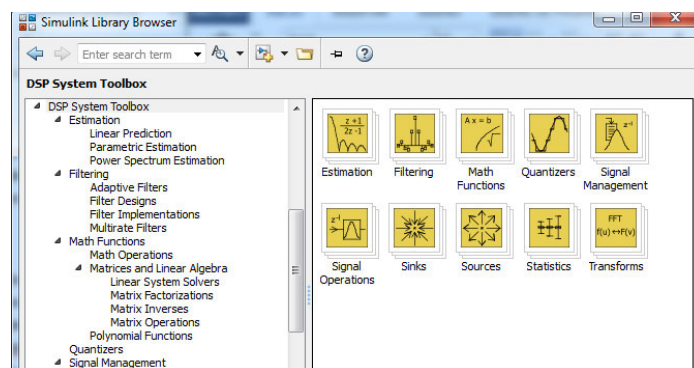


Figura 1.4 Librería DSP System Toolbox

1.3.2.3 OrCAD

El avance tecnológico y adquisición de computadoras personales con capacidades de memoria y posibilidades gráficas, da lugar al diseño electrónico asistido por computadora, cuyas características logran ahorro de tiempo y esfuerzos, lo cual implica productividad y calidad.

²³ FFT: *Fast Fourier Transform*, (Transformada rápida de Fourier).

²⁴ STFT: *Short Time Fourier Transform*, (Transformada de Fourier de tiempo reducido).

²⁵ RMS: *Root Mean Square*, (Valor cuadrático medio).

²⁶ DSP: *Digital Signal Precessing*, (Procesamiento digital de señales).

OrCAD es un software usado en aplicaciones técnicas y de ingeniería electrónica; este programa es propietario de *Cadence Design Systems* y contiene un conjunto de herramientas empleados para la esquematización y simulación de circuitos electrónicos. Incluye un editor de esquemas (*Capture*), un simulador de circuitos (*PSpice*) y un diseñador de PCB²⁷. [7]

- *OrCAD Capture* es una aplicación de captura esquemática, no tiene características de simulación construida pero si incluye un CIS²⁸ para gestión de datos de componentes y una funcionalidad de *scripting* TCL/TK²⁹ que permite escribir *scripts* para personalizar y automatizar.
- *OrCAD* como diseñador de PCB se ofrece siendo una herramienta para la impresión de placas de circuitos con un trazado de pistas; se integra con *OrCAD Capture* utilizando CIS para almacenamiento de señal.
- *PSpice* es parte del paquete de software que permite realizar la simulación del comportamiento de los circuitos para su análisis respectivo.

OrCAD EE PSpice es un espacio que posibilita la simulación y verificación de circuitos analógicos, la ejecución de circuitos definidos en *OrCAD Capture* y puede integrarse con Matlab y Simulink (SLPS³⁰), permitiendo simulación en áreas como electromecánica [7].

Existen varias versiones del paquete *OrCAD*, para el desarrollo de este trabajo se utilizará la versión 16.6 full, la misma que se encuentra disponible en la página de *OrCAD CADENCE* de manera gratuita para clientes, que al llenar un formulario se redirige a las descargas en (http://www.orcad.com/resources/orcad_downloads). Sus librerías incluyen numerosos componentes.

OrCAD incluye versiones demos de las herramientas *OrCAD Capture*, *OrCAD Capture CIS*, *PSpice A/D*, *PSpice Advanced Analysis*, *OrCAD PCB and SPECCTRA for OrCAD*.

OrCAD PSpice, como sus siglas (SP) lo indican, es un programa de Simulación Personal con énfasis en circuitos integrados, cuenta con la ventaja de un conjunto de herramientas que incluye un sinnúmero de elementos electrónicos como resistencias, condensadores,

²⁷ PCB: Printed Circuit Board, (Placa de circuito impreso).

²⁸ CIS: Component Information System, (Sistema de Información de Componentes).

²⁹ TCL/TK: Tool Command Language/Toolkit, (Lenguaje de herramientas de comando/Conjunto de herramientas).

³⁰ SLPS: Simulink to PSpice Interface, (Interfaz de *Simulink* a *PSpice*).

inductores, fuentes de poder, generadores de señales, amplificadores, dispositivos semiconductores e incluso hasta elementos de medida; por medio del cual es posible probar, cambiar y modificar diseños deseados con el fin de entender el comportamiento del circuito implementado. PSpice soporta análisis en DC³¹, AC³² y transitorios.

Además, esta herramienta de simulación es aplicable en numerosas industrias como medicina, electrónica de potencia, automatización, etc.; también es usada como una herramienta en el plano de la investigación, ya que es capaz de simular desde circuitos simples hasta sistemas de radiofrecuencia con funciones integradas, optimización de circuitos, funciones electromecánicas, entre otros. [7]

En Comunicación Digital, PSpice es una herramienta de aprendizaje que contribuye en gran medida a la práctica de teoría adquirida durante los programas de estudios realizados, donde el enfoque pedagógico proporcionará resultados en temas como:

- Análisis de Fourier
- Procesamiento de señales
- Técnicas de transmisión en banda base y pasabanda
- Técnicas de modulación y demodulación digital
- Instrumentos de prueba y rendimiento de sistema de transmisión
- Sistemas de Espectro Ensanchado por Secuencia Directa (DSSS³³)

Componentes del paquete CADENCE 16.6

El paquete CADENCE 16.6 contiene varias aplicaciones funcionales, Figura 1.5.

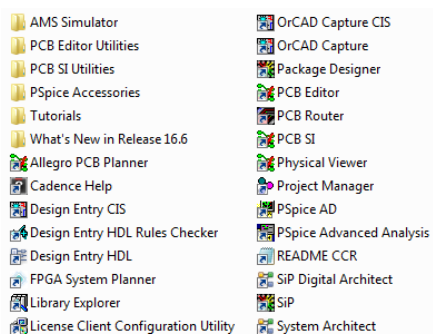


Figura 1.5 Componentes del paquete CADENCE v16.6

³¹ DC: “Direct Current” (Diseño Asistido por Computadora).

³² AC: “Alternative Current” (Diseño Asistido por Computadora).

³³ DSSS: “Direct-Sequence Spread Spectrum” (Sistemas de Espectro Ensanchado por Secuencia Directa).

Capture y PSpice son los componentes principales del paquete CADENCE v16.6, los cuales se encargan de la visualización y simulación del circuito implementado y los que se usarán para desarrollar este trabajo de titulación.

Entorno de trabajo de OrCAD CAPTURE

Para acceder al entorno de trabajo, se debe establecer una sesión de simulación como un nuevo archivo de proyecto, el mismo que se guardará en un directorio asignado por el usuario en la memoria de la computadora. El nuevo proyecto se creará desde la barra de menú “File > New >Project...” o directamente desde la ventana *Getting Started* accediendo por el icono *New*, Figura 1.6 [8].

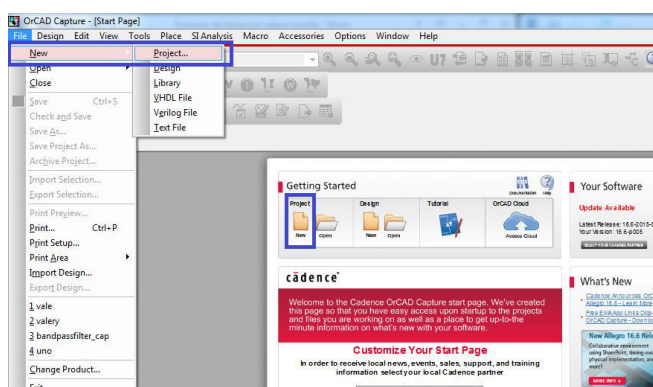


Figura 1.6 Inicio del software OrCAD Capture

Luego, se desplegará un cuadro de diálogo “New Project”, como indica en la Figura 1.7.

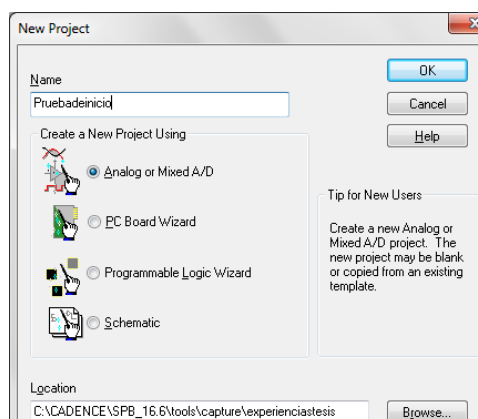


Figura 1.7 Cuadro de diálogo para crear nuevo proyecto de PSpice

En esta ventana se debe indicar el nombre del proyecto, tipo de proyecto (*Analog or Mixed A/D*) y dirección donde se guardarán todos los ficheros que describen los elementos utilizados en el circuito, las conexiones, las librerías, resultados, etc., generados; los ficheros utilizados en los diseños son leídos por *PSpice* para verificar errores y finalmente

realizar la simulación. Seguidamente, se desplegará un cuadro de diálogo, Figura 1.8, para elegir el tipo de proyecto a crear.

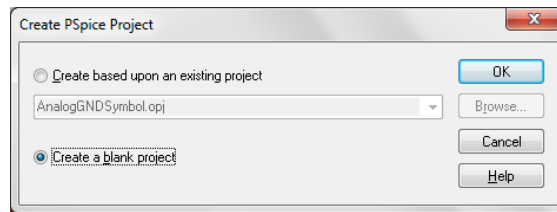



Figura 1.8 Cuadro de diálogo de opción para nuevo proyecto

Cuando se accede por primera vez es necesario añadir las librerías, seleccionando en la barra de menú la opción “Place > Part” o dando clic en el ícono ; se desplegará una ventana que permitirá agregar las librerías (para tener acceso a las componentes es recomendable agregar todas las librerías de PSpice para evitar errores de simulación). De esta manera se presenta el entorno de trabajo, Figura 1.9.

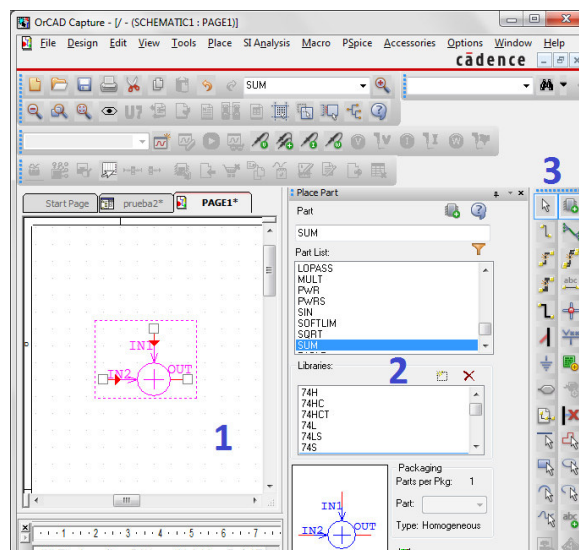


Figura 1.9 Entorno de trabajo de OrCAD Capture

- 1. Ventana de diseño de *OrCAD Capture*, donde se realiza la esquematización de los diseños a implementar.
- 2. Ventana *Place Part*, espacio para agregar las librerías con un clic en el ícono “Add Library”, permitiendo seleccionar las necesarias para abarcar los elementos que formarán parte del diseño a implementar.
- 3. Paleta de herramientas, contiene ciertos botones principales para la creación de circuitos, como: líneas de conexión entre puntos (cableado del esquema), etiquetado, conexión a tierra, introducción de figuras geométricas, así como también textos, entre otros.

1.3.3 Procesamiento de señales y filtros digitales [9] [10]

El procesamiento de señales a lo largo del tiempo se ha ido involucrando en algunos campos profesionales como en telecomunicaciones, control, medicina, entre otros. Los sistemas de comunicaciones al convertirse en sistemas multifunción, móviles, inalámbricos, etc., relevan la importancia de un procesamiento de señales sofisticado en los elementos del mismo.

El procesamiento de señales se considera una forma de interpretación de las señales que se originan de varios tipos de fuentes, en telecomunicaciones se tratan fuentes analógicas o digitales dando lugar así al DSP³⁴ o ASP³⁵. En el campo de la ingeniería, el procesamiento de señales es una técnica de tratamiento de señales en la que se involucran varios procesos como amplificación, filtrado, estimaciones, etc.

En DSP, el análisis de señales y filtros, son aspectos básicos e importantes dentro de esta temática. Es por esto que en el próximo capítulo se desarrollarán escenarios de simulación para prácticas orientadas a abordar temáticas de las series y transformadas de *Fourier*, y filtros digitales FIR e IIR. Para ello se utilizarán las herramientas computacionales mencionadas, permitiendo la aplicación de conocimientos de DSP y comunicación digital.

Descripción del procesamiento de señales

El medio y nuestro alrededor, se encuentra rodeado de un sinnúmero de señales, las cuales están en diversas formas y muchas de ellas son de origen natural o son provocadas por el hombre, como voz, música, video, motores de carros, el estampido de un trueno, el impacto de la lluvia en la tierra, el latido del corazón, etc.

En el campo de la ingeniería, las señales son portadoras de información y pueden ser transmitidas en forma digital al ser estas representadas por muestras de la señal tomadas en intervalos periódicos, las mismas luego expresadas en números y estos en códigos binarios.

En base a lo mencionado, se puede definir al procesamiento de señales como la operación diseñada para el tratamiento, almacenamiento, mejoramiento o extracción de información útil de un conjunto de señales deseadas y no deseadas.

³⁴ DSP: *Digital Signal Processing* (Procesamiento digital de señales)

³⁵ ASP: *Analog Signal Processing* (Procesamiento analógico de señales)

Las señales en su mayoría son analógicas, es decir varían de forma continua en tiempo y en amplitud, se procesan con circuitos los cuales contienen elementos activos y pasivos, cuyo enfoque se conoce como procesamiento analógico de señales. Ejemplo: receptores de radio y televisión analógicos.

Además, también es posible procesar señales mediante el uso de hardware digital los cuales contienen sumadores, multiplicadores y elementos lógicos, o se puede usar microprocesadores que son de uso especial ya que tienen circuitos de conversión de datos, permitiendo conseguir una forma de onda adecuada para hardware digital, o en otras palabras una señal digital.

DSP soporta tareas de análisis de señales, la cual se encarga de la medición de propiedades de la señal y cuya operación está en el dominio de la frecuencia, donde su aplicación generalmente está en el análisis de espectro de frecuencia. La segunda clasificación es la tarea de filtrado de señal, la cual se caracteriza por aplicaciones como: eliminación de ruido no deseado, separación de bandas de frecuencia, etc.

El procesamiento digital de señales implica tres etapas: primero la CAD³⁶, segundo el procesado de la señal digital y como tercero la conversión de la señal digital procesada en una señal continua o analógica, CDA³⁷. El diagrama de bloques que se presenta en la Figura 1.10, indica los elementos básicos de un sistema de procesado digital.

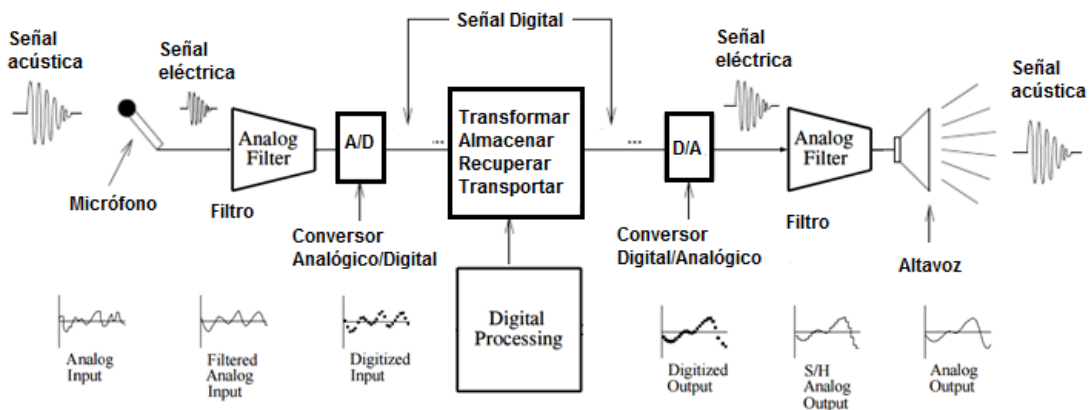


Figura 1.10 Estructura de un sistema PDS [12]

Los elementos como la señal analógica de entrada, un filtro analógico anti-aliasing (aliasing es el solapamiento del espectro o efecto que causa que señales continuas se tornen

³⁶ CAD: Conversión Analógica Digital

³⁷ CDA: Conversión Digital Analógica

indistinguibles cuando se muestrea digitalmente), un bloque CAD, un bloque PDS que puede ser representado por un procesador, ordenador, etc., un bloque CDA el mismo que realiza la operación inversa al CAD produciendo una señal escalonada desde una secuencia de números binarios, y por último un bloque que representa un filtro reconstructor para suavizar la forma de onda escalonada de la señal obtenida.

Una estructura de DSP, frente a una estructura de procesamiento analógico de señales es más robusta, porque cuenta con mayor complejidad y contenido de elementos, que hacen que tenga ventajas sobre el mismo.

Ventajas y desventajas de DSP frente a ASP

La principal limitación de ASP, se da en aplicaciones y diseños de procesamiento de señales complejas. Por lo que DSP proporciona algunas ventajas importantes como:

- DSP tiene menor costo por el uso de tecnología VLSI (*Very Large Scale Integration*) que comprende ahorro de elementos electrónicos como memorias, compuertas, microprocesadores, etc. Da la posibilidad de diseñar sistemas, con la capacidad de ejecutar procesamiento en tiempo real de muchas de las señales de interés, cuyas aplicaciones se las puede ejecutar en comunicaciones, control, procesamiento de imagen, multimedia, etc.
- Confiabilidad, por el uso de técnicas digitales, produce una tasa de error relativamente baja, generando señales de alta fidelidad con posibilidades de detección y corrección de errores.
- Las operaciones de DSP pueden ser fácilmente modificadas en tiempo real, por simples cambios de programación.
- El uso de hardware para sistemas digitales presenta mayor flexibilidad para su implementación que un hardware analógico, ante alguna modificación se puede configurar cambiando valores, no es necesario reemplazar o aumentar componentes.
- Los sistemas que usan DSP se pueden desarrollar usando el software que se usa en una computadora de propósito general. Por lo tanto, el DSP es muy conveniente usar ya que el software tiene la ventaja de ser portable.
- Operaciones de DSP se basan en sumas y multiplicaciones proporcionando estabilidad por su capacidad de procesamiento.
- Baja sensibilidad a condiciones ambientales, tolerancia y envejecimiento de componentes.

A pesar de las grandes ventajas que brinda el procesamiento digital de señales también es necesario tomar en cuenta las desventajas como [13]:

- Diseños complejos, por aspectos de hardware y software.
- Uso de grandes anchos de banda.
- Los sistemas digitales requieren alcanzar un alto grado de sincronización donde para los sistemas analógicos es mucho más fácil.
- Pérdidas de información por muestreo, ya que toda la información no se encuentra presente.
- Error debido a cuantización, dando lugar a una degradación de señal como consecuencia de pérdida de información al representar una señal analógica mediante muestreo digital con un cierto número de valores.

Aplicaciones de DSP en el campo tecnológico

Debido a las grandes ventajas que posee el procesamiento digital de señales, se destaca la aplicación del mismo en varios campos como [13]:

- Comunicación inalámbrica, codificación/decodificación, optimización de anchos de banda.
- Procesamiento de voz como la reducción de ruido, reconocimiento de voz.
- Audio y música como grabación, reproducción, manipulación.
- Procesamiento de imágenes como captura de fotografía, mejoramiento de calidad, reconocimiento de patrones, biometría, detección de estados emocionales.
- Multimedia en cine, video 3D, video juegos, TV digital.
- Control industrial en control de calidad, robótica, identificación de patrones.
- Aplicaciones militares como radares, detección de objetivos, navegación, seguimientos, comunicaciones seguras, entre otros.
- Biomedicina, como análisis de rayos X, análisis de electro-encefalogramas, etc.

1.3.3.1 Generalidades de análisis de señales

Señales y sistemas en tiempo discreto

- Señales en tiempo discreto [9]

Es importante tomar en cuenta la clasificación de señales, las mismas que son analógicas que se denotan como $x_a(t)$, donde la variable $[t]$ en las prácticas de comunicaciones representará valores de tiempo continuo en segundos; y las señales discretas las cuales

se denotan como $x(n)$, en donde la variable $[n]$ toma valores de instantes discretos en el tiempo.

Para análisis se podrán usar varios tipos de señales en tiempo discreto (secuencias), tales como secuencia impulso unitario, secuencia escalón unitario, secuencia exponencial de valor real, secuencia exponencial de valor complejo, secuencias aleatorias, secuencias sinusoidales y secuencias periódicas. Las secuencias sinusoidales se pueden representar de la forma como muestra la ecuación 1.1, y se puede usar la función seno o coseno.

$$x(n) = A * \cos(\omega_0 n + \theta_0), \forall n$$

Ecuación 1.1 Secuencia sinusoidal

Donde, A define la amplitud pico de la señal, ω la frecuencia angular y θ_0 indica la fase en radianes de la secuencia.

Una secuencia en tiempo discreto periódica se define cuando la misma se repite después de un intervalo de tiempo, matemáticamente se representa como: $x(n) = x(n + N)$, $\forall n$, donde N es el periodo e indica la duración de un ciclo completo. La inversa del período es la frecuencia de la secuencia. Una secuencia periódica es una señal de potencia y una secuencia no periódica es una señal de energía.

➤ Sistemas en Tiempo Discreto [9]

Un sistema en tiempo discreto toma una secuencia de entrada $x(n)$ y la transforma en una secuencia de salida $y(n)$ de características deseadas. La notación para efectuar y representar esta operación es $T[\cdot]$, ecuación 1.2

$$y(n) = T[x(n)] \quad \text{ó} \quad x(n) \rightarrow T[\cdot] \rightarrow y(n)$$

Ecuación 1.2 Sistema en tiempo discreto

Los sistemas en el dominio del tiempo discreto, se clasifican en sistemas lineales y no lineales o sistemas variantes e invariantes en el tiempo. Un sistema lineal e invariante se denota como LTI³⁸, tienen aplicaciones en radares, equipos biomédicos, equipos industriales y militares, entre otros. Matemáticamente, la salida de un sistema LTI está dada por la convolución de la secuencia de entrada $x(n)$ con la respuesta impulso $h(n)$, ecuación 1.3.

³⁸ LTI: *Linear and Time Invariant*, (Lineal e Invariante en el tiempo)

$$y(n) = x(n) * h(n)$$

Ecuación 1.3 Salida de un sistema LTI, caracterizado por $h(n)$

Un sistema discreto LTI, también puede ser descrito por una ecuación diferencial lineal de coeficientes constantes, ecuación 1.4 [9], con $a_0 = 1$.

$$y(n) = \sum_{m=0}^M b_m x(n-m) - \sum_{k=1}^N a_k y(n-k)$$

Ecuación 1.4 Ecuación diferencial lineal de coeficientes constantes de LTI

Donde a_k y b_m son los coeficientes del sistema y N determina el orden de la ecuación diferencial, si $a_N \neq 0$. Una solución a esta ecuación, puede obtenerse a partir de la suma de una solución homogénea y una solución particular, $y(n) = y_H(n) + y_P(n)$.

A un sistema LTI también se conoce con el nombre de filtro y se le designa el trabajo específico de selección o discriminación de frecuencias, es así que un sistema LTI en el dominio del tiempo discreto también es llamado filtro digital. Si, la respuesta impulso de un sistema LTI es de duración finita ($h(n) = 0$, para $n < n_1$ y $n > n_2$), se habla de un filtro de respuesta impulso de duración finita (FIR, *Finite Impulse Response*), también conocido como filtro no recursivo, matemáticamente se describe en la ecuación 1.5.

$$y(n) = \sum_{m=0}^M b_m x(n-m)$$

Ecuación 1.5 Descripción de un filtro FIR

La respuesta impulso para este filtro es ($h(n) = b_n$, para $n \leq M$ y $n \geq 0$), y para otros casos, cero. De igual manera, si la respuesta impulso de un sistema LTI es de duración infinita, se habla de un filtro de respuesta impulso de duración infinita (IIR, *Infinite Impulse Response*) también conocido como filtro recursivo, y se describe matemáticamente como se muestra en la ecuación 1.4.

En la materia de Análisis de Señales y Sistemas, se estudia la teoría de sistemas discretos en el dominio del tiempo y de la frecuencia. A manera de resumen se ha mencionado estos temas con el fin de dar a conocer las temáticas que se aplicarán para la realización de prácticas de procesamiento de señales. Esto a su vez implicará diseñar e implementar experiencias con aplicaciones de Filtros Digitales, empleando el software computacional, Matlab, Simulink u OrCAD PSpice. En el próximo capítulo se desarrollarán las prácticas correspondientes, en las cuales se introducirá su marco teórico en base al desarrollo de las experiencias.

Análisis de Fourier [9] [13] [14] [15]

El análisis de Fourier es una herramienta matemática que permite caracterizar señales y sistemas en el dominio de la frecuencia, en otras palabras permite expresar una función como la suma infinita de ondas sinusoidales (seno, coseno) para determinar las frecuencias contenidas en la misma; este análisis se aplica de acuerdo al tipo de señales que se tiene, señales periódicas y no periódicas. Es así que para analizar una señal periódica en el dominio de la frecuencia se usa las series de Fourier y para una señal no periódica se usa la transformada de Fourier.

➤ Transformada de Fourier

La transformada de Fourier en el tiempo discreto de una señal periódica $x(n)$, en función del valor exponencial complejo, $\{e^{j\omega n}\}$, se define como, ecuación 1.6. [9]

$$X(f) \triangleq \mathcal{F} [x(n)] = \int_{-\infty}^{\infty} x(n)e^{-j\omega n} dn$$

Ecuación 1.6 Transformada de Fourier

Mediante la transformada de Fourier se puede determinar el espectro de una señal o el conjunto de frecuencias que la constituyen; y, para obtener de manera exacta se requiere observar la forma de onda sobre un intervalo infinito, pero es posible tener una aproximación considerando un intervalo de tiempo finito [13]. De manera contraria, para obtener la señal original, se aplica la transformada inversa, por definición, ecuación 1.7 [9].

$$x(n) \triangleq \mathcal{F}^{-1} [X(f)] = \int_{-\infty}^{\infty} X(f)e^{j\omega n} df$$

Ecuación 1.7 Transformada inversa de Fourier

Dado que se trata de señales no periódicas, $x(n)$ la energía normalizada de la señal puede ser descrita mediante la ecuación 1.8 [9].

$$E_x = \int_{-\infty}^{\infty} |x(n)|^2 dn = \int_{-\infty}^{\infty} |X(f)|^2 df \quad [\text{Joules}]$$

Ecuación 1.8 Energía de una señal no periódica

Además, en base a la ecuación 1.9 [9], se puede determinar el aporte de energía de cada componente de frecuencia (ESD, *Energy Spectral Density*) que tiene la señal.

$$\varepsilon(f) = |X(f)|^2 \quad [\text{Joules/Hz}]$$

Ecuación 1.9 Densidad espectral de energía

➤ Series de Fourier

Con la aplicación de las series de Fourier, una señal periódica en el dominio del tiempo discreto o en el dominio del tiempo continuo, puede ser expresada como la combinación lineal de funciones exponenciales, ecuación 1.10 [9].

$$x(n) = \sum_{k=0}^{N-1} x(k) e^{j\frac{2\pi}{N}kn}, n = 0, \pm 1, \pm 2, \dots$$

Ecuación 1.10 Series de Fourier

El periodo fundamental está definido por N ; $X(k)$ son los coeficientes correspondientes a las series de Fourier y se determinan a partir de la ecuación 1.11 [9].

$$X(k) = \frac{1}{N} \int_{-N/2}^{N/2} x(n) e^{-j\frac{2\pi}{N}kn} dn, k = 0, \pm 1, \pm 2, \dots$$

Ecuación 1.11 Coeficientes de las series de Fourier

Una señal periódica contiene únicamente las componentes de frecuencia que son múltiplos de la frecuencia fundamental (o primer armónico $f_0 = (1/N)$), $f_0, 2f_0, 3f_0$, etc. Además, para señales periódicas es decir señales de potencia, la función densidad espectral de potencia $\rho(f)$ (PSD, *Power Spectral Density*) indica la cantidad de potencia contribuida por cada componente de frecuencia, de manera que la potencia normalizada de la señal está dada en la ecuación 1.12 [14].

$$P = \int \rho(f) df$$

Ecuación 1.12 Potencia normalizada

$\rho(f)$ está relacionado con los coeficientes de Fourier, P se obtiene a partir de la ecuación 1.13. [9]

$$P = \frac{1}{N} \int_{-N/2}^{N/2} |x(n)|^2 dn$$

Ecuación 1.13 Potencia normalizada promedio de una señal periódica

1.3.4 Sistemas de modulación y demodulación digital [14] [18] [19] [20]

Un sistema de comunicaciones está formado básicamente por tres elementos principales que son transmisor, receptor y canal de transmisión, cuya función permite la transmisión de información a manera de símbolos analógicos (voz humana, música, etc.) o digitales

(señales discretas), pero no siempre la información proveniente del transmisor tiene una forma adecuada para ser llevada en su formato original, es por eso que de acuerdo al tipo de sistema de comunicaciones se debe adaptar la información antes de transmitirla [19].

En comunicación digital se considera formas de transmisión en que se incluye la transmisión en banda base y transmisión con portadora que son técnicas de modulación para la representación de información en el sistema de comunicaciones. La transmisión en banda base hace referencia a la transmisión de pulsos digitales, y para el caso de la transmisión con portadora hace referencia a la transmisión de portadoras analógicas moduladas en forma digital en un sistema de comunicación.

La modulación digital es la técnica por la cual ciertas características (amplitud, frecuencia, fase o una combinación de estos parámetros) de una onda denominada portadora (sinusoide) se modifican en base a otra onda denominada moduladora, para que pueda ser transmitida. La portadora tiene la forma de la ecuación 1.14 [14].

$$A_c \cos(\omega_c t + \theta)$$

Ecuación 1.14 Sinusoide que actúa como portadora

Donde los parámetros representan, A_c la amplitud pico de la portadora, ω_c la frecuencia angular de la portadora y θ la fase de la portadora.

La modulación como tal tiene razones motivadoras para que se aplique en los entornos de sistemas de comunicaciones, entre ellas: facilitan la propagación de información por cable o aire, disminuyen dimensiones de antenas, permiten multiplexar o compartir un medio con otros usuarios simultáneamente, permiten mayores alcances de transmisión y radiación de energía efectiva.

Respecto al espectro de frecuencia, en la modulación el espectro de la señal original que se encuentra en banda base se traslada al intervalo de frecuencia de la portadora, que por lo general es una gama de alta frecuencia y por lo tanto al ser mayores frecuencias el ancho de banda aumenta y en cierta parte esta característica surte ventajas y desventajas frente a la modulación analógica dependiendo del tipo de modulación que se emplee.

Los esquemas de modulación digital más relevantes que se tratarán en las prácticas del presente trabajo de titulación son la modulación ASK, FSK, PSK y QAM: [13] [14] [19] [20]

ASK, (*Amplitude Shift Keying* / Modulación por desplazamiento de Amplitud), corresponde a la técnica de modulación más sencilla en la cual se hace variar la amplitud de la portadora en base a la información digital o modulante, manteniendo la frecuencia y la fase

constantes, en otras palabras ASK es básicamente una modulación AM con la diferencia que la modulante es digital, Figura 1.11, y se describe matemáticamente como la ecuación 1.15 asumiendo una portadora normalizada y de fase cero grados. [14]



Figura 1.11 Modulación ASK [13]

$$ASK(t) = [1 + k \cdot b_n(t)] \cos(w_c t)$$

Ecuación 1.15 Modulación ASK

Donde los parámetros representan, k como el índice de modulación ($0 \leq k \leq 1$), b_n como la modulante, w_c frecuencia angular de la portadora.

La modulación ASK puede ser del tipo binaria, en este tipo de modulación la amplitud pico de la portadora toma dos valores posibles determinados por la señal de datos binaria; pero también puede tomar más de dos valores diferentes ($M > 2$) dando lugar a la modulación ASK multiestado. Cada valor de M está conformado por un símbolo o estado de la señal modulada y cada estado por N bits. En base a lo mencionado, los estados de modulación ASK y el número de bits que conforma cada estado o símbolo de modulación se puede determinar matemáticamente por la ecuación 1.16.

$$M = 2^N [\text{Estados}], \quad N = \log_2 M [\text{bits/símbolo}]$$

Ecuación 1.16 Nivel de modulación ASK

La velocidad de los dígitos binarios está dada en bps³⁹ y la velocidad de los símbolos o velocidad de modulación está dada en baudios, para determinar se usa las ecuaciones 1.17 y 1.18.

$$v_{tx} = N * v_m$$

Ecuación 1.17 Velocidad de transmisión de los datos binarios

$$v_m = \frac{v_{tx}}{N}$$

Ecuación 1.18 Velocidad de modulación ASK

³⁹ bps: *bits per second*, (bits por segundo)

De todas las técnicas de modulación, ASK se considera como una técnica de modulación ineficaz por ser la de más bajo rendimiento debido a la distorsión y el ruido que afectan a la señal transmitida, pero también tiene sus aplicaciones en la transmisión de datos digitales en fibras ópticas con transmisores LED o laser, en donde se usa un pulso de luz para un elemento de señal mientras que para el otro se presenta como ausencia de luz.

Los esquemas de modulación M-ASK no son usados por las dificultades que se presentan en el receptor el proceso de discriminación entre valores de amplitud pico de la portadora modulada, muy cercanos [14].

Un caso particular de ASK es la técnica OOK⁴⁰, esta técnica es una de las más sencillas, cuyo funcionamiento se comporta como el de un interruptor, en el que se presenta como la presencia o la ausencia de una señal portadora, como se ilustra en la Figura 1.12.

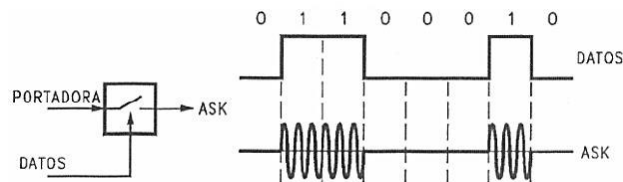


Figura 1.12 Modulación OOK [22]

FSK, Modulación por desplazamiento de Frecuencia, corresponde a la técnica de modulación de tipo angular, en la cual se varía la frecuencia de la portadora en base a la información digital, Figura 1.13, y se describe matemáticamente como se indica en la ecuación 1.19.

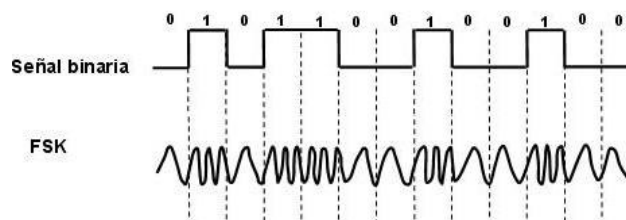


Figura 1.13 Modulación FSK [22]

$$FSK(t) = V_c \cdot \cos \left\{ w_c + \frac{[b_n(t) \cdot \Delta w]}{2} \right\} t$$

Ecuación 1.19 Modulación FSK

⁴⁰ OOK: *On-Off keying, Modulación por encendido y apagado*

Donde los parámetros representan, V_c como la amplitud pico de la portadora, b_n es una señal digital NRZ relacionada con los datos y que toma valores ($\pm 1, \pm 3, \pm 5 \dots$) dependiendo del número de estados, w_c como la frecuencia angular de la portadora sin modular, Δw como separación entre frecuencias adyacentes de la señal modulada.

Para este tipo de modulación se tiene dos formas que son: FSK Coherente, la cual mantiene continuidad de la fase cuando cambia la frecuencia de la portadora modulada; y FSK No Coherente, que presenta discontinuidad de fase en los cambios de estado.

PSK, Modulación por desplazamiento de Fase, corresponde a la técnica de modulación en la cual se hace variar la fase de la portadora en base a la información digital, mientras que los parámetros de amplitud y frecuencia de la portadora se mantienen constantes, Figura 1.14. Matemáticamente se define en la ecuación 1.20 [13].

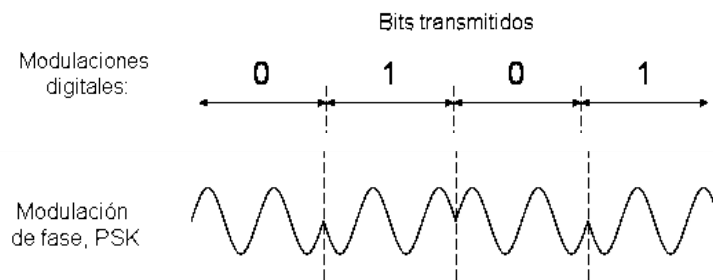


Figura 1.14 Modulación PSK [23]

$$PSK(t) = V_c \cdot \cos \left\{ w_c t + \frac{[b_n(t) \cdot \Delta\theta]}{2} \right\} \quad ; \quad \Delta\theta = 2\pi/M$$

Ecuación 1.20 Modulación PSK

Donde los parámetros representan, V_c como la amplitud pico de la portadora, b_n es una señal digital NRZ relacionada con los datos y que toma valores ($\pm 1, \pm 3, \pm 5 \dots$) dependiendo del número de estados, w_c como la frecuencia angular de la portadora, $\Delta\theta$ como separación entre fases adyacentes, este parámetro depende del número de estados (M).

Para valores de $M > 2$, al igual que la modulación M-ASK, se obtiene en base a la ecuación 1.16, lo cual también da lugar a las modulaciones de múltiples estados M-PSK (QPSK, 8PSK, 16PSK, etc.). De la misma manera, la relación entre velocidad de modulación y velocidad de transmisión se puede calcular usando las relaciones que se presentan en las ecuaciones 1.17 y 1.18.

La señal modulada M-PSK se puede generar mediante la suma de dos señales en cuadratura, moduladas en amplitud, y para la representación de la portadora modulada se hace referencia al diagrama de constelación, el cual permite representar en el plano complejo por un punto los estados del símbolo en relación de amplitud y fase. Este diagrama consta de dos ejes, el eje horizontal se denomina eje "I" (en fase) y el eje vertical "Q" (en cuadratura), este diagrama se ilustra en los ejemplos de la Figura 1.15. [20]

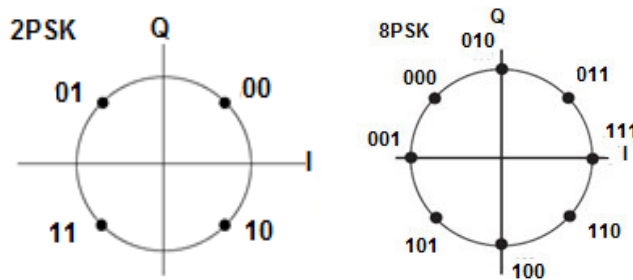


Figura 1.15 Diagramas de constelación de un modulador QPSK y 8PSK [20]

D-PSK⁴¹, Modulación Diferencial por Desplazamiento de Fase, se tiene cuando la información binaria está contenida en las variaciones de la fase de la portadora, con respecto a la fase del estado anterior. En este tipo de modulación cada símbolo produce un incremento de fase en la portadora, con respecto del estado anterior.

Como ventaja la demodulación DPSK no requiere recuperar una portadora coherente en fase, no necesita por tanto un circuito de recuperación, pues la condición lógica se determina a partir de la diferencia de fases consecutivas. DPSK en sí modula solo los cambios de fase y no la fase absoluta.

La Figura 1.16 muestra un diagrama de bloques simplificado de un modulador D-BPSK⁴², cuyo esquema se puede implementar anteponiendo a un modulador PSK un codificador diferencial a partir de la compuerta XNOR.

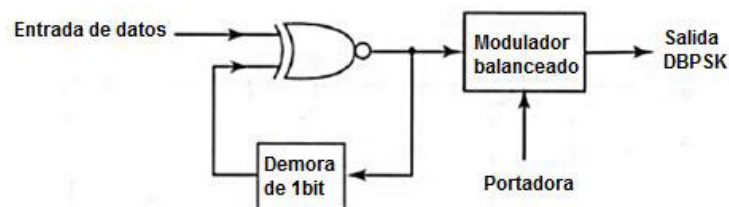


Figura 1.16 Diagrama de bloques de un modulador DBPSK [19]

⁴¹ D-PSK: *Differential Phase Shift Keying*, (Modulación Diferencial PSK)

⁴² D-BPSK: *Differential Binary Phase Shift Keying*, (Modulación Diferencial PSK Binario)

Es así que en base a lo descrito, la Figura 1.17 muestra la respuesta a la salida de la compuerta XNOR a partir de una secuencia de datos binaria de entrada, y las fases a la salida del modulador, en donde se puede decir que la fase de la señal modulada cambia 180° cuando el bit a transmitirse es un 0L y mantiene la fase anterior cuando es un 1L. Además, como para el primer bit de datos no hay un bit anterior se debe considerar un bit de referencia.

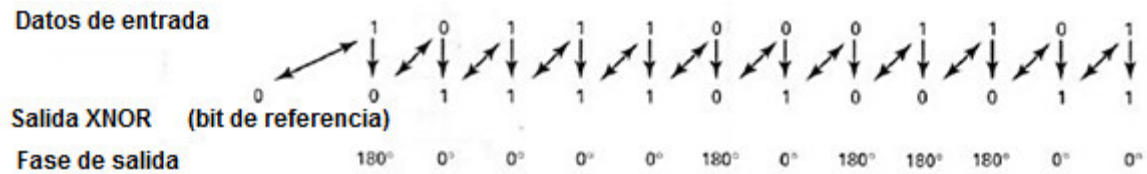


Figura 1.17 Diagrama de sincronización de DPSK [19]

QAM⁴³, Modulación de Amplitud en Cuadratura, corresponde a la técnica de modulación en la cual se hace variar de forma simultánea la fase y amplitud de la portadora en base a la información digital, esto se consigue cuando se modula dos portadoras de igual frecuencia pero desfasada entre sí 90°, aprovechando de esta manera el ancho de banda disponible.

La modulación QAM también cuenta con esquemas de modulación multiestados (16-QAM, 64-QAM, etc.) donde para cada una de ellas varía la característica de amplitud y fase como se observa en los diagramas de constelación, Figura 1.18.

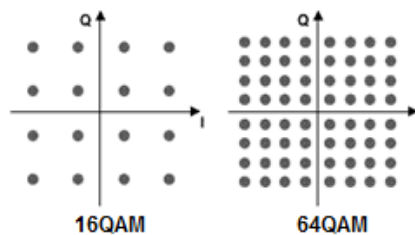


Figura 1.18 Diagramas de constelación modulación M-QAM [24]

Matemáticamente una señal modulada QAM puede expresarse de la forma como se indica en la ecuación 1.21. [13]

$$QAM(t) = V_i * \cos (w_c t + \theta_i)$$

Ecuación 1.21 Modulación QAM

⁴³ QAM: *Quadrature Amplitude Modulation*, (Modulación de amplitud en cuadratura)

Donde los parámetros, V_i indica cambios de amplitud pico de la portadora modulada, w_c la frecuencia angular y θ_i los cambios de fase de la portadora modulada.

En esencia en el próximo capítulo se abordarán prácticas referentes a estas técnicas de modulación y/o demodulación digital, ASK, FSK, PSK y QAM. Los cuales se dividirán en temas de prácticas, donde se simulará y analizará los resultados correspondientes a los mismos, y donde se extenderá el marco teórico en base a la aplicación práctica.

1.3.5 Sistemas de transmisión multiportadora usando COFDM

Hoy en día los sistemas de transmisión inalámbricos exigen mayor velocidad de transmisión de datos y transmisión confiable, por lo que el uso de multiplexación por división de frecuencia ortogonal (OFDM, *Orthogonal Frequency Division Multiplexing*) es una técnica adecuada para la comunicación inalámbrica de datos de alta velocidad en canales multitrayectoria, además es eficaz resolviendo problemas del canal de desvanecimiento selectivo en frecuencia, por lo que es adoptado por muchos estándares como DAB⁴⁴, ADSL, WLAN, IEEE 802.11 a/g/n etc.

Los códigos de control de errores se han convertido en una parte importante en los sistemas digitales inalámbricos modernos los cuales permiten una transmisión de datos confiable de alta velocidad que puede lograrse sobre canales ruidosos OFDM, que al combinarse con los métodos FEC toma el nombre de multiplexación por división de frecuencia ortogonal codificada (COFDM, *Coded Orthogonal Frequency Division Multiplexing*) y permite al sistema OFDM mejorar el rendimiento.

El COFDM es una técnica de modulación de banda ancha con múltiples portadoras ortogonales, cada una modulada en amplitud y fase que además se combina junto con codificación de canal para detección y corrección de errores. Se utiliza para transmisión de información (audio, video y datos, etc.) a altas velocidades.

1.3.5.1 Características del COFDM

COFDM resulta de una mejora del OFDM para canales muy selectivos o variantes ya que puede soportar la presencia de interferencias de banda estrecha co-canal, el ruido de impulsos, multitrayecto severo, y la reducción rápida de la amplitud de la señal.

⁴⁴ DAB: *Digital Audio Broadcasting*

Las características principales de COFDM son:

- Ortogonalidad
- Sincronización
- Ecuación
- Adición de intervalo de Guarda
- Codificación contra errores
- El entrelazamiento de las portadoras de datos en frecuencia o en tiempo y frecuencia.
- La información de estado del canal (Channel State Information) combinado con la decodificación con decisión flexible (Soft-Decision Decoding) para incrementar el desempeño del decodificador de Viterbi.

1.3.5.2 Ventajas y desventajas de COFDM

Ventajas

- **Protección contra desvanecimiento selectivo de las portadoras:** COFDM promedia el desvanecimiento local mediante la inserción conjunta del código convolucional y del entrelazador de frecuencia a la modulación OFDM.
- **Alta eficiencia espectral:** Mediante la técnica de modulación OFDM se logra aumentar considerablemente la tasa binaria útil a transmitir por lo que se aprovecha eficientemente la banda de frecuencia.
- **Simplificación de la ecualización:** Se facilita la ecualización en el receptor, debido a que se distribuye una serie de portadoras llamadas portadoras pilotos a lo largo de todo el ancho de banda que se va a usar en la transmisión, con esto se consigue eliminar o reducir la influencia del canal sobre los datos transmitidos.
- **Protección contra interferencia intersímbolo (ISI):** Mediante la utilización del intervalo de guarda se consigue evitar que unos símbolos OFDM se vean afectados por otros.
- **Ampliaciones en Redes de Frecuencia Única (SFN: Single-Frequency Networks):** Mediante la utilización de varios transmisores para generar un multitrayecto produce un efecto llamado ganancia de red, con lo que se consigue una reducción de la potencia del transmisor, mejorando la cobertura y reduciendo costos de la infraestructura.

Desventajas

- **Sensibilidad a la sincronización en tiempo y frecuencia:** Para el receptor le es difícil encontrar el comienzo del símbolo OFDM y encontrar la posición de las portadoras dentro del símbolo OFDM para establecer la sincronización en tiempo y frecuencia.
- **Mayor complejidad del sistema:** Los requerimientos de control y corrección de error de fase, la alta linealidad del amplificador de potencia en el transmisor, el entrelazamiento de frecuencia y otros requerimientos provocan un incremento de la complejidad del sistema
- **Transmisores y receptores más costosos:** Debido a la complejidad del sistema en COFDM el costo del transmisor y receptor aumentan, dando lugar a receptores de baja, mediana o alta calidad.
- **Pérdida de eficiencia espectral:** Causada por la duración del intervalo de guarda y tasa de codificación utilizada, que deben ser ajustados para varias condiciones de funcionamiento.
- **Más sensible al ruido de fase y al desplazamiento en frecuencia en las portadoras:** El ruido de fase es causado por todos los osciladores locales que hay desde la salida de la IFFT del transmisor hasta la entrada de la FFT en el receptor. El desplazamiento de frecuencia causa ICI⁴⁵ y una reducción en la potencia en las portadoras.

1.3.5.3 Uso de múltiples portadoras

El uso de múltiples portadoras se emplea para reducir la interferencia intersímbolo a partir de niveles significativos de multitrayecto. Suponiendo que se modula una portadora con información digital, durante cada símbolo, se transmite la portadora con una fase y amplitud particular que se elige del diagrama de constelación en uso. Cada símbolo transmite una cantidad de bits de información, igual al logaritmo del número de estados diferentes en la constelación.

Suponiendo que la señal es recibida a través de dos trayectorias, con un retardo relativo entre ellas. Se toma como ejemplo el símbolo transmitido n , donde el receptor intentará demodular los datos que fueron enviados en este símbolo, examinando toda la información

⁴⁵ ICI: *Inter Carrier Interference*

recibida relativa al símbolo n , tanto la información directamente recibida como la información retardada.

Cuando el retardo relativo es más de un periodo del símbolo, ver Figura 1.19, la señal recibida a través del segundo trayecto actúa puramente como interferencia, ya que sólo lleva información perteneciente a un símbolo o símbolos anteriores. Esta interferencia entre símbolos (ISI) implica que sólo se pueden tolerar niveles muy pequeños de la señal retardada.

Cuando el retardo relativo es menor que un periodo del símbolo, Figura 1.20, parte de la señal recibida por el segundo trayecto actúa puramente como interferencia, puesto que sólo lleva información perteneciente a su símbolo anterior.

El resto de la señal lleva la información del símbolo deseado, pero puede añadir constructiva o destructivamente a la información de la trayectoria principal.

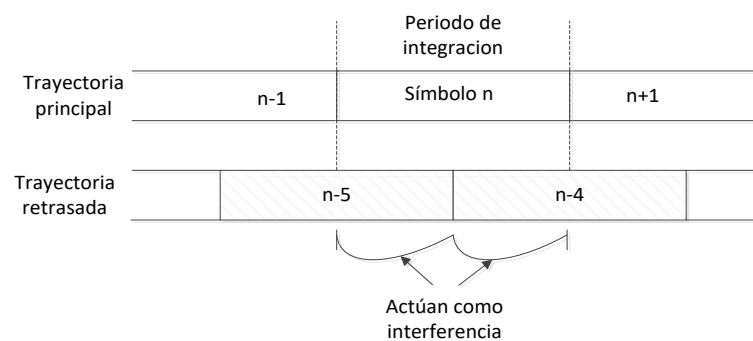


Figura 1.19 Retraso largo en la trayectoria causa ISI

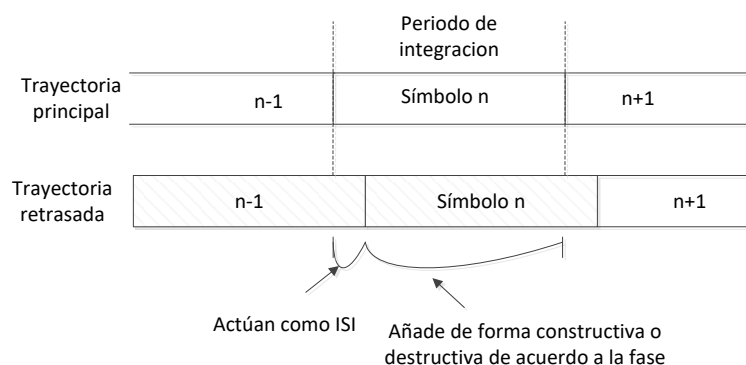


Figura 1.20 Retraso corto en la trayectoria causa ISI

Esto nos dice que, si queremos resolver cualquier nivel apreciable de señales retardadas, la tasa de símbolos debe reducirse lo suficiente para que la propagación total de retardo (entre el primer y último trayecto recibido) sea sólo una fracción modesta del periodo de

símbolo. La información que puede ser transportada por una sola portadora es así limitada en presencia de multitrayecto.

Si una portadora no puede llevar la tasa de información que se requiere, esto conduce a la idea de dividir los datos de alta velocidad en muchos flujos paralelos de baja velocidad, cada uno transportado por su propia portadora de las que hay un gran número. Esta es una forma de FDM el primer paso hacia COFDM [32].

1.3.5.4 Diagrama de bloques del sistema COFDM

El diagrama de bloques para un sistema COFDM se muestra en la Figura 1.21. En el COFDM los datos del lado del transmisor son cargados en un aleatorizador, el cual generará un número aleatorio de secuencia para ser codificados a través del FEC (*Forward Error Correction*) encoder, el cual además se encargará de dar redundancia a los bits transmitidos y de esta manera posibilita la corrección de errores en el receptor.

Luego se utiliza un entrelazador de frecuencias para mejorar el rendimiento del sistema, una vez terminado este proceso los bits ingresan de dos en dos para modularlo con QPSK para luego transmitir la información aplicando la IFFT (Transformada rápida de Fourier. En el lado del receptor la señal es demodulada, de-entrelazada y decodificada a través del FEC decoder para recuperar la información transmitida.

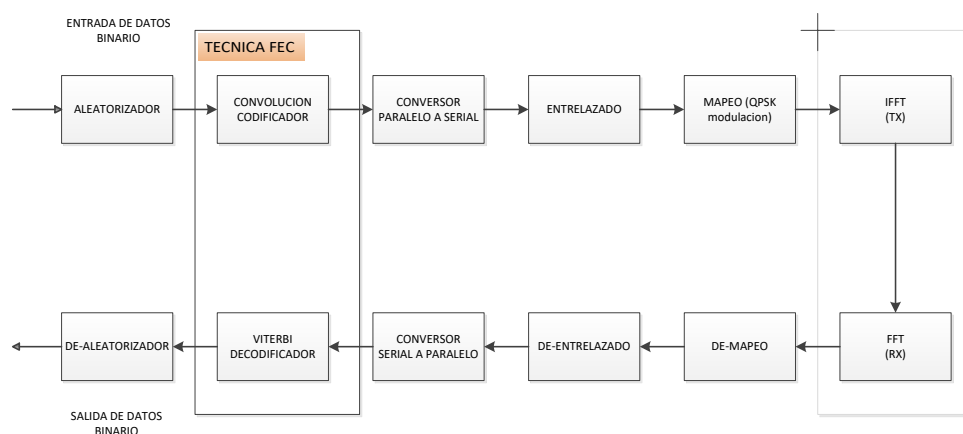


Figura 1.21 Diagrama de bloques del COFDM

En el próximo capítulo se abordará la descripción del esquema de un transmisor y un receptor usando COFDM en base al diagrama de bloques descrito en la Figura 2.21, el cual será simulado usando la herramienta Simulink para validar su funcionamiento aplicado para una señal de audio.

2. METODOLOGÍA

En el presente capítulo se desarrollarán las prácticas planteadas en el plan de este trabajo de titulación, se emplearán las herramientas de software descritas en el marco teórico de este trabajo, Matlab, Simulink y OrCAD PSpice. En cada práctica propuesta se expondrán objetivos, marco teórico referente al desarrollo de la práctica y la simulación correspondiente. Los resultados de las simulaciones se analizarán en el próximo capítulo de resultados y discusión.

En la primera sección se presentarán tres guías introductorias de los softwares de simulación; se considerará objetivos, descripción de la guía y un desarrollo de un breve ejemplo cuyo fin será familiarizar al estudiante con las herramientas de cada software.

En la segunda sección se presentarán las prácticas relacionadas a introducción al procesamiento de señales, análisis de señales (Fourier) y Filtros Digitales (FIR e IIR).

En la tercera sección se presentarán las prácticas respectivas a las técnicas de modulación y/o demodulación digital de ASK, FSK, PSK y QAM.

Y finalmente se considerará una esquematización de un sistema de comunicaciones usando COFDM.

2.1 Guías de las herramientas de software

De acuerdo a las herramientas de software que se emplearán en el desarrollo de este trabajo, en esta primera sección se pretende elaborar tres guías introductorias a través de unos sencillos ejemplos prácticos con el fin de apreciar y aplicar las herramientas, Matlab, Simulink y OrCAD PSpice.

2.1.1 Guía 1: Una experiencia con procesamiento de señales usando Matlab

2.1.1.1 Objetivos

- Familiarización con el uso de la herramienta Matlab, conociendo y entendiendo sus ventajas en laboratorios basados en software.
- Entender la importancia y ventajas que presenta el sistema de ayuda de Matlab, para conocer el funcionamiento y sintaxis de comandos.
- Conocer las componentes de *Signal Processing Toolbox* de Matlab utilizados en sistemas de comunicaciones.

- Desarrollar una experiencia con procesamiento de señales utilizando el software Matlab, a modo de guía de uso.
- Realizar un análisis de resultados, en base a la aplicación de Matlab en prácticas de un laboratorio virtual.

2.1.1.2 Descripción de guía

En esta experiencia se pretende realizar una pequeña guía inductiva, por medio de un sencillo ejemplo práctico con procesamiento de señales, donde se aplicará el uso de la herramienta Matlab, y en el que se conocerán las ventajas que presenta en el desarrollo de prácticas de laboratorio.

- Se presentará un escenario que se ejecutará en un fichero script (*extensión .m*). Se generará una señal de dos tonos a partir de la suma de dos señales de frecuencias 500 [Hz] y 5 [KHz] respectivamente.
- Se aplicará una de las funciones disponibles en *Signal Processing Toolbox Functions* de la sección *Digital Filter Design* de *Matlab* para realizar el diseño de un filtro (tipo paso bajo), para atenuar la frecuencia de 5 [KHz] de la señal de dos tonos generada.
- Se presentará los esquemas gráficos en función de tiempo y frecuencia en sus pasos respectivos, para análisis de resultados de la guía.

2.1.1.3 Desarrollo de guía

En Matlab es posible generar y presentar una señal de varias maneras, como un vector fila, vector columna, usando comandos y funciones predefinidas, representando la misma en base a parámetros requeridos.

Para esta guía se generarán señales continuas de la forma indicada en la ecuación 2.1 [9] evaluada en un intervalo de tiempo, segundos:

$$y = A * \sin (2 * \pi * F_o * t)$$

Ecuación 2.1 Sinusoide

En esta instancia, el sistema de ayuda de *Matlab* servirá como respaldo para conocer la sintaxis y funciones que cumplen ciertos comandos que se requieran saber, editando en el *Command Window* la palabra “*helpwin*” más el comando deseado. Tras un “*enter*” despliega una ventana que contiene la información del mismo y un breve ejemplo, como muestra la Figura 2.1.

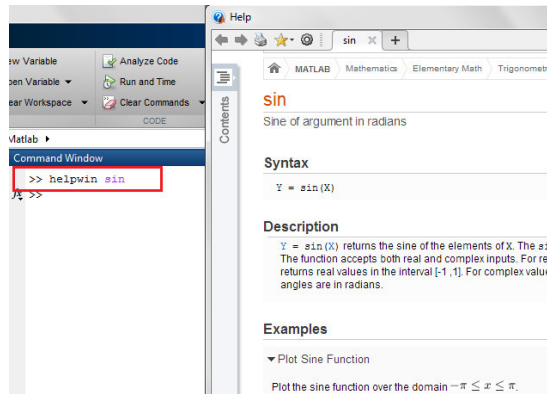


Figura 2.1 Help de Matlab

Para generar las señales mencionadas, se requieren definir ciertos parámetros iniciales, como se presentan en la Tabla 2.1.

Tabla 2.1 Parámetros iniciales de señal

Parámetro	Descripción
Fs	Frecuencia de muestreo en [Hz]
Ts	Período de muestreo en [s]
t	Definición del vector tiempo, para evaluación de la señal
N	Número de muestras y/o longitud de señal

De acuerdo a la guía, se generarán 2 señales (y_1, y_2), como se muestra en la Figura 2.2, los ejes de la gráfica están en función del parámetro tiempo.

- (y_1) de frecuencia 5 [KHz], Amplitud pico 2 [V]
- (y_2) de frecuencia 500 [Hz], Amplitud pico 2 [V]

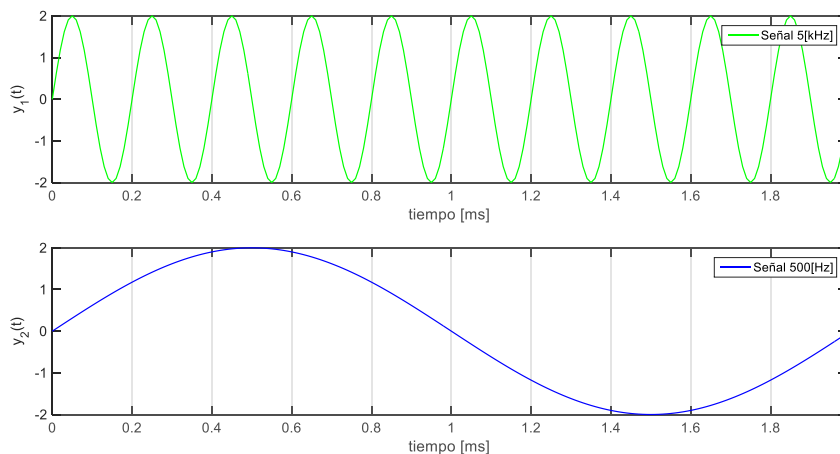


Figura 2.2 Señales, 5 [KHz] y 500 [Hz]

Dado el caso que se presente algún error al ejecutar el script (.m), Matlab indica en el *Command Window* en letras color rojo la descripción del mismo.

Dentro de las funciones fundamentales de Matlab se encontrarán varios comandos que permitirán realizar operaciones elementales tales como suma, resta, multiplicación, división, etc., así también en las librerías como *Signal Processing Toolbox* se encontrarán funciones para medir, generar, transformar, filtrar y visualizar señales, que además incluyen algunos algoritmos para diseñar y analizar filtros, entre otras funciones más. En base a lo descrito se procederá a realizar la suma '+', de las señales (y_1, y_2), cuyo resultado será una señal de dos tonos, como se muestra en la Figura 2.3, se visualizará un solo período.

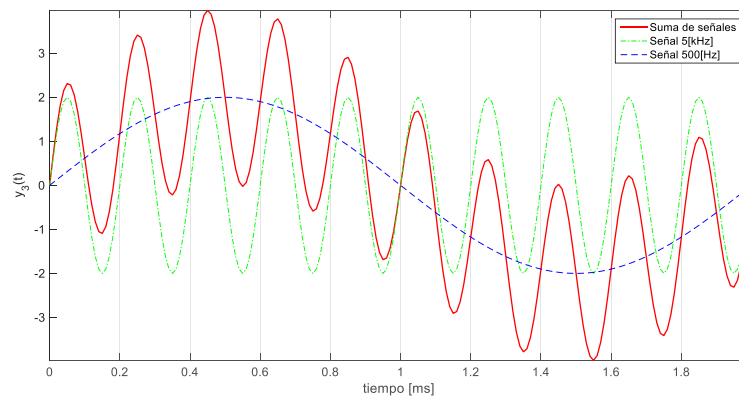


Figura 2.3 Suma de señales ($y_1 + y_2$)

Para determinar e identificar las componentes de frecuencia que contiene la señal, se usará el comando *fft*⁴⁶, que brinda la facilidad de convertir una señal del dominio del tiempo al dominio de la frecuencia, como se ilustra en la Figura 2.4.

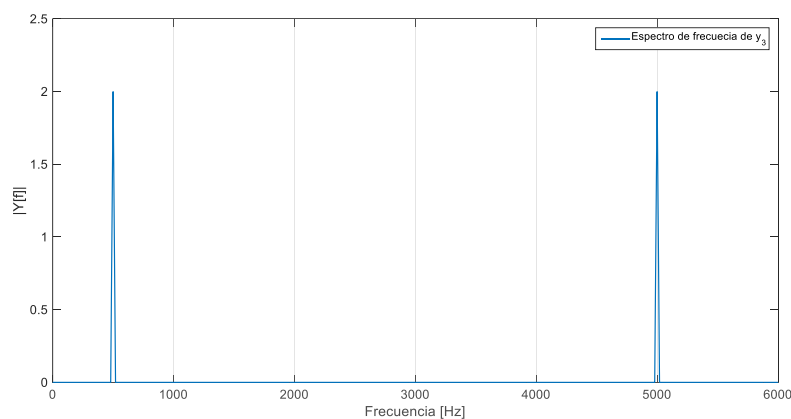


Figura 2.4 Espectro de suma de señales

De acuerdo a lo establecido en el procedimiento, se diseñará un filtro tipo pasa bajo; para lo cual en la subsección *Digital and Analog Filters* del contenido de *Signal Processing*

⁴⁶ *fft*: "Fast Fourier Transform" (Transformada rápida de Fourier).

Toolbox se encontrará una serie de comandos para diseño de filtros, como algunos que se indican en la Tabla 2.2.

Tabla 2.2 Comandos para diseño de filtros digitales

COMANDO	DESCRIPCIÓN	COMANDO	DESCRIPCIÓN
butter	Butterworth filter design	fir2	Frequency sampling-based FIR filter design
buttord	Butterworth filter order and cutoff frequency	firpm	Parks-McClellan optimal FIR filter design
cheby1	Chebyshev Type I filter design	firpmord	Parks-McClellan optimal FIR filter order estimation
cheb1ord	Chebyshev Type I filter order	gaussdesign	Gaussian FIR pulse-shaping filter design
cheby2	Chebyshev Type II filter design	intfilt	Interpolation FIR filter design
cheb2ord	Chebyshev Type II filter order	kaiserord	Kaiser window FIR filter design estimation parameters
ellip	Elliptic filter design	rcosdesign	Raised cosine FIR pulse-shaping filter design
fir1	Window-based FIR filter design	designfilt	Design digital filters

Para continuar con esta guía, en base a la Tabla 2.2, se usará el comando “*buttord*” (selecciona el mínimo orden de filtro digital o analógico *Butterworth* requerido para cumplir con un conjunto de especificaciones de diseño), y el comando “*butter*” que permitirá diseñar el filtro *Butterworth* en base al orden que se determinará a partir de “*buttord*”.

Un filtro *Butterworth* produce una respuesta plana en la banda de paso y a medida que aumenta el orden del filtro aumenta la pendiente en la banda de transición aproximándose así a un filtro de característica ideal, Figura 2.5.

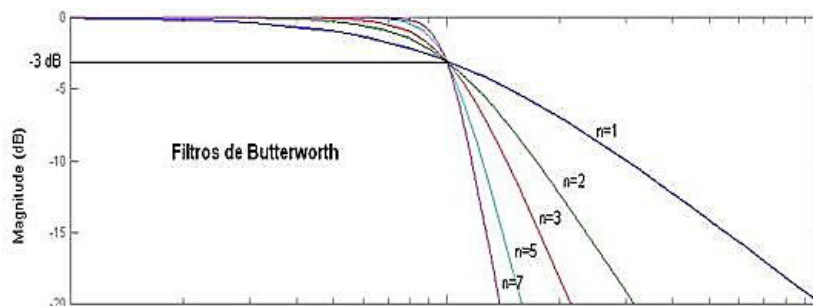


Figura 2.5 Características de un filtro Butterworth [25]

Dependiendo de las aplicaciones es posible diseñar filtros tipo pasa bajo, pasa alto y pasa banda. Las características paramétricas se muestran en la Figura 2.6.

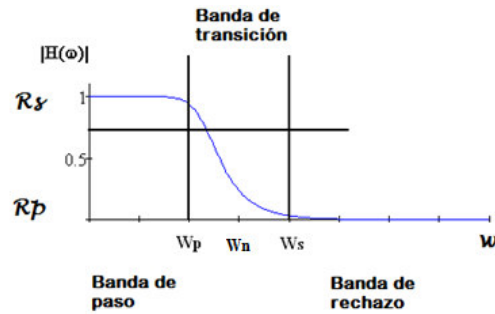


Figura 2.6 Características paramétricas del filtro Butterworth pasa bajo

La sintaxis de los comandos se verifica en el sistema de ayuda de Matlab, lo cual se presenta de la siguiente manera:

Commando: *"buttord"*

Sintaxis: $[n, W_n] = \text{buttord}(W_p, W_s, R_p, R_s)$

Donde:

- $n \rightarrow$ orden del filtro, describe el grado de aceptación o rechazo de frecuencia por arriba o por debajo de la correspondiente frecuencia de corte.
- $W_n \rightarrow$ frecuencia de corte en [rad/s], es el punto en el cual la señal ha sido atenuada 3 [dB] por el filtro. Además es el punto en el cual el filtro empieza a recortar frecuencias dependiendo del tipo de filtro (pasa bajo, pasa alto, pasa banda).
- $W_p \rightarrow$ frecuencia de paso en [rad/s], es el punto o borde de banda que indica hasta donde las frecuencias pueden pasar.
- $W_s \rightarrow$ frecuencia de rechazo en [rad/s], es el punto o límite de banda que indica el comienzo de la banda de rechazo, donde se producirá una alta atenuación.
- $R_p \rightarrow$ Tolerancia de paso [dB], indica la atenuación máxima permitida en la banda de paso.
- $R_s \rightarrow$ Tolerancia de rechazo [dB], indica la atenuación mínima en la banda de rechazo.

Cuya resultante será el orden y la frecuencia de corte del filtro, que será útil para aplicar en el diseño del filtro a través del comando:

Comando: *"butter"*

Sintaxis: $[A, B] = \text{butter}(n, W_n)$

Donde:

- $n \rightarrow$ orden del filtro
- $W_n \rightarrow$ Frecuencia de corte en [rad/s]
- $A, B \rightarrow$ coeficientes para determinar la función de transferencia

De acuerdo al procedimiento de esta guía, se aplicará el filtro paso bajo con el fin de dejar pasar la señal de 500 [Hz] y atenuar la señal de 5 [KHz], para lo cual se considerará una frecuencia de paso de 1[KHz] y una frecuencia de rechazo de 3 [KHz], con este par de datos a partir del comando “buttord” se determinará la frecuencia de corte, la misma que indica el paso del rango de frecuencias que estén por debajo de este valor y así mismo la atenuación de las frecuencias que estén por encima de este valor. La figura 2.7, muestra la respuesta en amplitud del diseño del filtro, en función de la frecuencia normalizada y frecuencia en Hertz, donde:

- $F_s \rightarrow 100E3$ [Hz] \rightarrow Frecuencia de muestreo
- $f_p = 1000$ [Hz] $\rightarrow W_p \rightarrow 0.02$ [rad/s] \rightarrow Frecuencia de paso
- $f_s = 3000$ [Hz] $\rightarrow W_s \rightarrow 0.06$ [rad/s] \rightarrow Frecuencia de rechazo
- $R_p \rightarrow 1$ [dB]
- $R_s \rightarrow 30$ [dB]

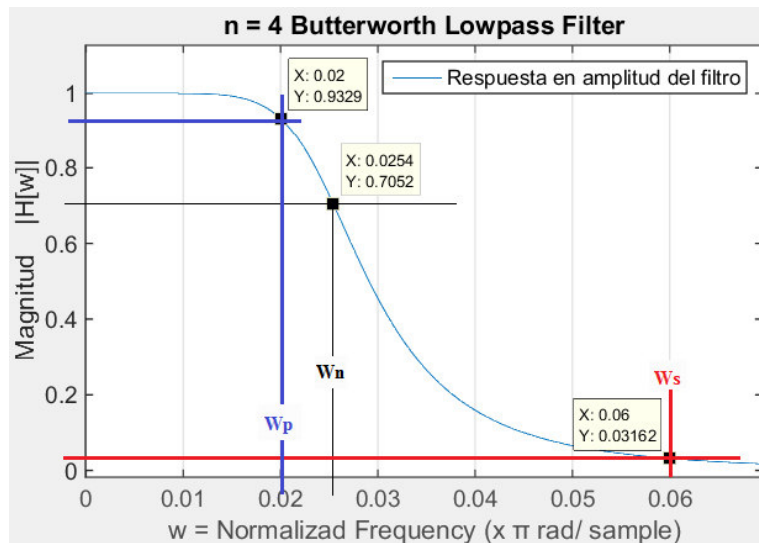
Y cuya línea de código de esta sección para determinar el orden y la frecuencia de corte es la siguiente (Ver Script completo en el Anexo I)

```
Fs=100e3; %frecuencia de muestreo, indica el número de muestras por segundo que se
toma de una señal analógica. En este caso será un parámetro para determinar las
frecuencias normalizadas a través del teorema de Nyquist.
% frecuencia angular W=2*π*f
%frecuencia normalizada W=f/Nyq
Nyq=Fs/2; %Teorema de Nyquist
fp=1000; %Frecuencia de paso en Hz
fs=3000; %Frecuencia de rechazo en Hz
Wp=fp/Nyq; % Frecuencia de paso normalizada;
Ws=fs/Nyq; % Frecuencia de rechazo normalizada
%resultado, orden y frecuencia de corte del filtro.
[n,Wn] = buttord(Wp,Ws,1,30)
%cálculo de la frecuencia normalizada en [Hz]
fn= Wn*Nyq
```

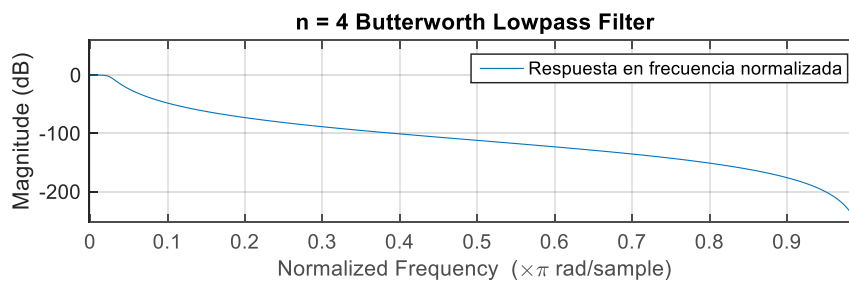
Resultado del cual se obtiene: $n=4$, el orden del filtro y $W_n = 0.0254$ [rad/s] , frecuencia de corte normalizada la misma que equivale a 1270 [Hz].

La respuesta en amplitud con magnitud [dB] está dada por $20\log(|H(w)|)$, Figura 2.7(b) y Figura 2.7(c). Con la Figura 2.7, se puede determinar que se cumple con las condiciones iniciales del filtro, es así como se observa en la Figura 2.7(a) que para la frecuencia de paso la atenuación está por debajo de 1 [dB], ($20*\log(0.9329) = - 0.60$ [dB]), la misma que constituye la ganancia mínima en la banda de paso; de igual manera para el inicio de la banda de rechazo que es la frecuencia de rechazo la atenuación es de 30 [dB],

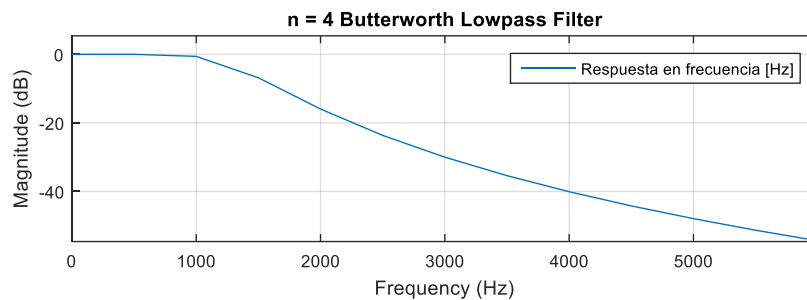
($20 \cdot \log(0.03162) = -30$ [dB]) la misma que es la ganancia máxima en la banda atenuada, y para la frecuencia de corte se tiene la ganancia de -3 [dB], ($20 \cdot \log(0.7052) = -3$ [dB]).



(a)



(b)

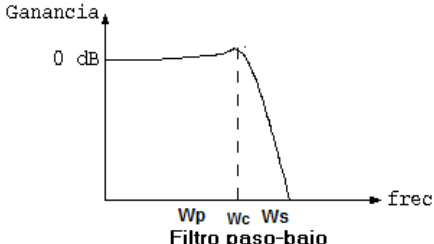
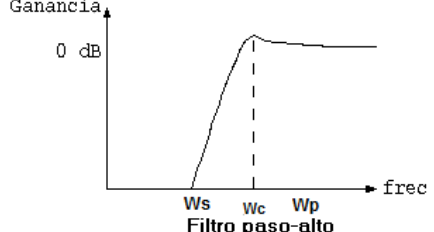
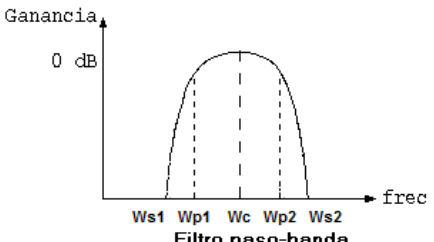
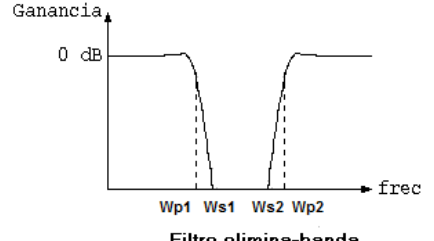


(c)

Figura 2.7 Respuesta en (a) amplitud del filtro, frecuencia [rad/s] (b) amplitud [dB] del filtro, frecuencia [rad/s] (c) amplitud [dB] del filtro, frecuencia [Hz]

La Tabla 2.3, muestra los tipos de filtros Butterworth que se pueden obtener a partir de las variaciones de las condiciones entre los parámetros de frecuencia de paso y frecuencia de rechazo (W_p , W_s), como se observa en la imagen de cada caso.

Tabla 2.3 Tipos de filtros Butterworth

Tipo de Filtro	Condiciones de Stopband y Passband
 <p>Filtro paso-bajo</p>	$W_p < W_s$, Ambos escalares.
 <p>Filtro paso-alto</p>	$W_p > W_s$, Ambos escalares.
 <p>Filtro paso-banda</p>	El intervalo especificado por W_s contiene las especificaciones por W_p $(W_s(1) < W_p(1) < W_p(2) < W_s(2))$
 <p>Filtro elimina-banda</p>	El intervalo especificado por W_p contiene las especificaciones por W_s $(W_p(1) < W_s(1) < W_s(2) < W_p(2))$

Luego, se realizará el filtrado de la señal de dos tonos, usando el comando “*filter*”, del cual se obtendrá la señal de un solo tono (500 [Hz]), en otras palabras se atenuará la de 5 [KHz]. Este proceso se muestra en la Figura 2.8, y en el script se describe de la siguiente manera, (Ver Script completo en el Anexo I):

```

%Aplicación de filtro en y3.
sigfil=filter(num,den,y3);
%Gráfica de la Señal Filtrada en función del tiempo.
plot(1000*t(1:250),sigfil(1:250),'LineWidth',1.2)
%Señal Filtrada en frecuencia.
N=5000;
F=fft(sigfil);
Fab=abs(F/N);
Fab1=Fab(1:N/2+1);
f = Fs*(0:(N/2))/N;
plot(f,2*Fab1,'LineWidth',1.3)
    
```

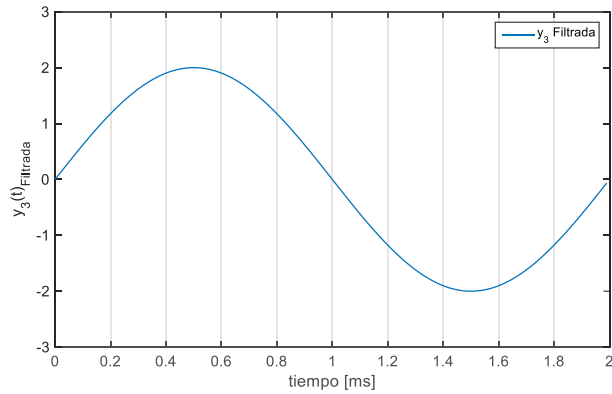


Figura 2.8 Respuesta del filtrado de la señal, en el dominio de tiempo

Finalmente, se visualizará la señal filtrada (Y_3), en su espectro de frecuencia, Figura 2.9.

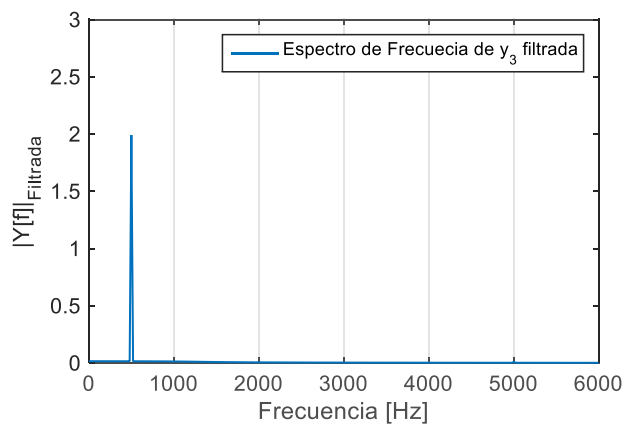


Figura 2.9 Espectro de frecuencia de señal filtrada.

El código de esta aplicación se detalla en el Anexo I.

2.1.2 Guía 2: Una experiencia con digitalización de señales usando Simulink

2.1.2.1 Objetivos

- Familiarizar al estudiante con el uso de la herramienta Simulink, conociendo sus componentes para la simulación, medición, visualización y análisis de esquemas en el área de sistemas de comunicaciones.
- Conocer las componentes de *Communication Systems Toolbox* y *DSP System Toolbox* en la creación de esquemas para sistemas de comunicaciones.
- Realizar un esquema PCM a modo de ejemplo de inducción al simulador *Simulink*, el mismo que servirá de guía para conocer las ventajas en construcción y modelado para la elaboración de prácticas en laboratorios.
- Determinar la importancia que implica el uso de Simulink en prácticas y desarrollo de experiencias para un laboratorio de sistemas de comunicaciones.

2.1.2.2 Descripción de guía

En esta experiencia se propone realizar una pequeña guía inductiva teórica, por medio de un sencillo ejemplo práctico de un sistema PCM, donde se aplicará el uso de la herramienta Simulink, y en el que se conocerán las ventajas que presenta este simulador en el desarrollo de prácticas de laboratorio. Con esta guía, se pretende familiarizar al estudiante con el uso que implica este entorno interactivo usando bloques funcionales en el modelamiento, diseño, simulación e implementación de sistemas basados en comunicación digital y procesamiento de señales.

El escenario a simular, permite la inducción al uso de componentes del *Communication System Toolbox* realizando un esquema que consiste en un proceso PCM.

El escenario propone probar el esquema con tres señales generadas de distinta manera, señal de audio, señal que será resultado de la suma de dos ondas de distintas frecuencias y finalmente una señal proveniente de un bloque generador de señales (diente de sierra). Luego se procederá a digitalizar dichas señales por medio de PCM, la cual se caracteriza por ser una modulación que permite la conversión de señales analógicas a digitales.

Para realizar esta guía se considerarán tres etapas principales que son:

- Muestreo, consiste en tomar muestras de la señal analógica, involucra un proceso de discretización de la señal en el tiempo.
- Cuantificación, consiste en un proceso de discretización en amplitud, en otras palabras, establece niveles a los cuales se deben ajustar las muestras.
- Codificación, consiste en dar a cada nivel de cuantificación un código binario distinto.

Presentación de los esquemas gráficos en el dominio del tiempo y frecuencia en sus pasos respectivos, para análisis de resultados del funcionamiento del proceso del sistema PCM.

2.1.2.3 Desarrollo de guía

Para el desarrollo de esta guía se usa la interfaz gráfica que presenta *Simulink*, para poder realizar el esquema de simulación del sistema PCM. Esta herramienta, cuenta con la ventaja de que el uso de su interfaz es amigable e interactiva, cuya construcción de escenarios de simulación se basan en un clic y arrastre de los bloques desde las librerías, para después ser interconectados mediante líneas. Para la implementación de esta simulación se usarán los bloques necesarios que se presentan en:

- *Simulink*
 - *Sine Wave, Manual Switch, Zero Order Hold, Scope, Sum*
- *Communications System Toolbox*
 - *A Law Compressor, Unipolar to Bipolar Converter*
- *DSP System Toolbox*
 - *From Audio Device, To Multimedia File, Uniform Encoder, Buffer, Unbuffer*

Para empezar, se considerará tres tipos de señales de entrada: la primera es la resultante de la suma de dos señales de distintas frecuencias, la segunda es una señal de voz y la tercera en una señal que proviene de un generador de señales (diente de sierra).

➤ **Suma de señales**

Simulink, *DSP System Toolbox* y *Communication System Toolbox* presenta los bloques necesarios para dar lugar a las señales de entrada, como es el bloque *Sine Wave*, el cual permitirá generar una señal *seno/coseno* dependiendo de la configuración que presenta el bloque de simulación, como se muestra en la Figura 2.10.

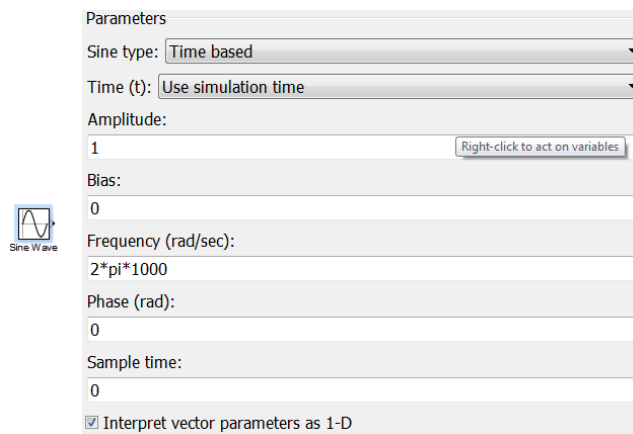


Figura 2.10 Bloque de simulación de la señal seno

Los parámetros de configuración de las señales que darán lugar a la señal resultante después de aplicar el bloque sumador, tendrá los siguientes parámetros:

- *Sine Wave 1*, amplitud pico de 1 [V], frecuencia 3 [KHz]
- *Sine Wave 2*, amplitud pico de 1 [V], frecuencia 1 [KHz]

El bloque *Sum* o también llamado combinador, permite realizar operaciones de adición o sustracción de sus datos de entrada, como se muestra en la Figura 2.11, permitiendo obtener la primera señal que se digitalizará mediante PCM.

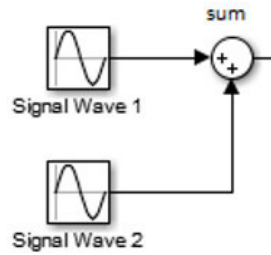


Figura 2.11 Esquema de suma de dos señales

➤ **Lectura de datos de audio**

La segunda señal de entrada a ser digitalizada será una pequeña muestra de una señal de audio. Mediante el bloque *From Audio Device* de la librería de *DSP System Toolbox*, se leerán los datos provenientes del dispositivo de audio en tiempo real, este elemento se complementará con el bloque *To Multimedia File*, como se muestra en la Figura 2.12, cuya función será, guardar los datos de audio en el *current directory* de *Matlab* como archivo multimedia con extensión *.WAV*, *.AVI*, *.MJ2000*, *.MPEG4*, *.WMV*, *.OGG* ó *.WMA.*, dependiendo de la configuración en el parámetro *File Type* (tipo de archivo).

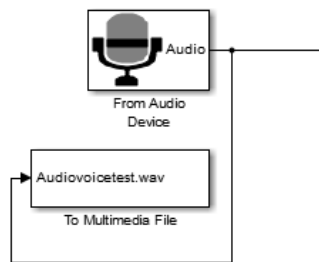


Figura 2.12 Esquema para obtener la señal de voz de un dispositivo de audio

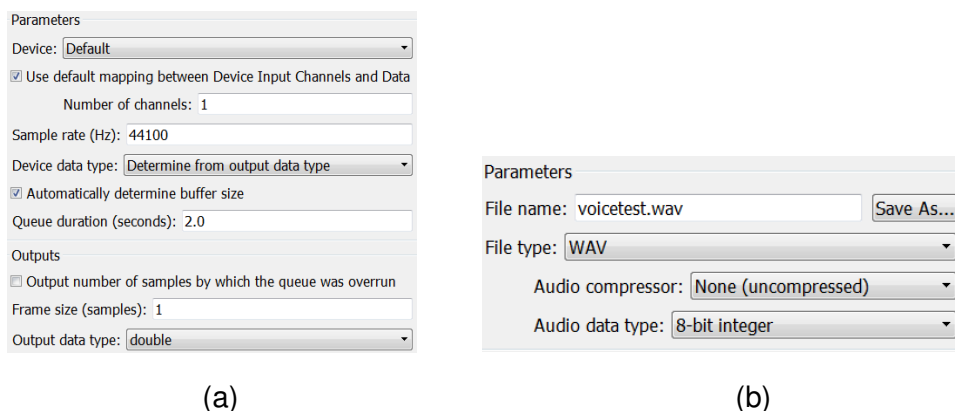


Figura 2.13 Bloque de simulación (a) correspondiente a *From Audio Device*, (b) correspondiente a *To Multimedia File*

Los parámetros de configuración de los bloques de simulación *From Audio Device* y *To Multimedia File* se presentan cada uno en la Figura 2.13.a y Figura 2.13.b respectivamente. El parámetro “frecuencia de muestreo” por ser audio se considerará de 44100 [Hz], y el

tiempo de duración será de 2 segundos. El tipo de archivo que se generará tendrá extensión *.wav*.

➤ **Oscilador de onda diente de sierra**

La tercera señal de entrada a ser digitalizada será una señal diente de sierra de amplitud 1 y frecuencia 3 [KHz]. *Communication System Toolbox* presenta el bloque *Signal Generator*, este bloque puede generar diferentes tipos de señales entre las cuales se tiene: diente de sierra (*sawtooth*), sinusoidal (*sine*), cuadrada (*square*) o aleatoria (*random*). De acuerdo a los requerimientos se configurará el bloque de simulación que se observa en la Figura 2.14.

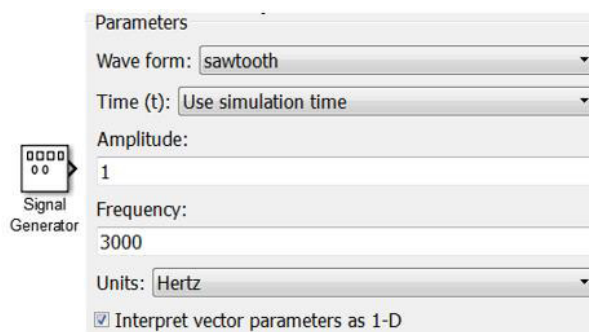


Figura 2.14 Bloque de simulación de la señal diente de sierra

➤ **Proceso PCM**

De acuerdo a la descripción de esta guía se presentarán los bloques funcionales que darán lugar a la digitalización de las señales de entrada descritas anteriormente, para la digitalización se lleva a cabo tres etapas que son: muestreo, cuantificación y codificación.

El esquema de simulación para PCM se ilustra en la Figura 2.15.

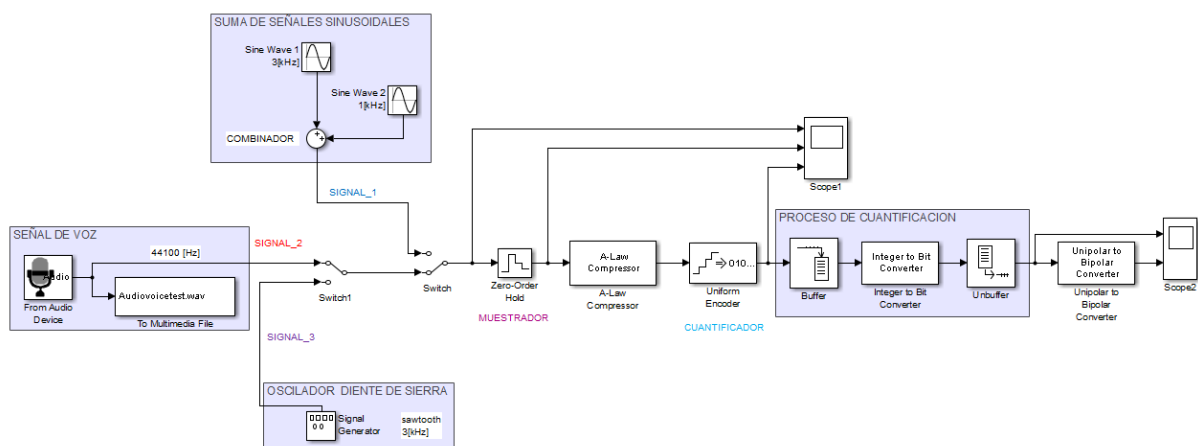



Figura 2.15 Esquema de digitalización, modulación PCM.

El bloque *Zero–Order Hold*, permitirá realizar el muestreo de la señal. En la configuración de los parámetros de este bloque se tomará en cuenta el teorema de Nyquist, el cual dice que una señal debe ser muestreada a una velocidad de por lo menos 2 veces la frecuencia más alta contenida en dicha señal.

Para la cuantificación, se usará el método no uniforme ya que permite tener niveles de cuantización más pequeños para niveles de señal de menor amplitud, se puede obtener cuando se hace pasar la señal por un compresor y luego por un cuantizador uniforme. En Simulink para el proceso de cuantificación no uniforme se usará el bloque *A-Low Compressor* el cual permitirá comprimir la señal en forma logarítmica es decir cambia la distribución de las magnitudes en la señal de entrada que luego pasará por el bloque *Uniform Encoder* el cual permitirá cuantificar la señal de manera que a los valores obtenidos después del bloque *A-Low Compressor* de la señal de entrada asignará en este caso 256 (2^8) valores posibles de cuantificación.

La codificación de los valores cuantificados se realizará a través del bloque *Integer to Bit Converter*, que en conjunto con los bloques *Buffer* (almacenamiento de datos de señal) y *Unbuffer* (des-almacenamiento de datos de señal), se encargarán de convertir los niveles de cuantización en formato binario, que luego serán modificados por el bloque *Unipolar to Bipolar Converter* el cual permitirá que los valores de entrada que se encuentran entre 0 y 1 (unipolar), cambien sus valores entre -1 y 1 (polar).

Finalmente, para el análisis de resultados se ha colocado un bloque denominado *Scope*, el cual permite visualizar gráficas de señales obtenidas. Mediante un doble clic en el bloque, se despliega una ventana que presenta dos ejes y además es posible modificar parámetros de presentación, número de ejes (entradas), rangos, colores, líneas, entre otros.

Para observar los resultados, se ejecutará la simulación en el icono  *Run* que presenta Simulink. El análisis de resultados se dará de acuerdo a la señal de entrada seleccionada mediante los bloques *switch* implementados.

Si la selección de la señal de entrada es “*SIGNAL_1*”, los resultados se presentan en la Figura 2.16 en la cual se visualiza la gráfica de la señal de entrada, muestreo y cuantificación, respectivamente. Y la Figura 2.17, presenta la codificación de la misma.

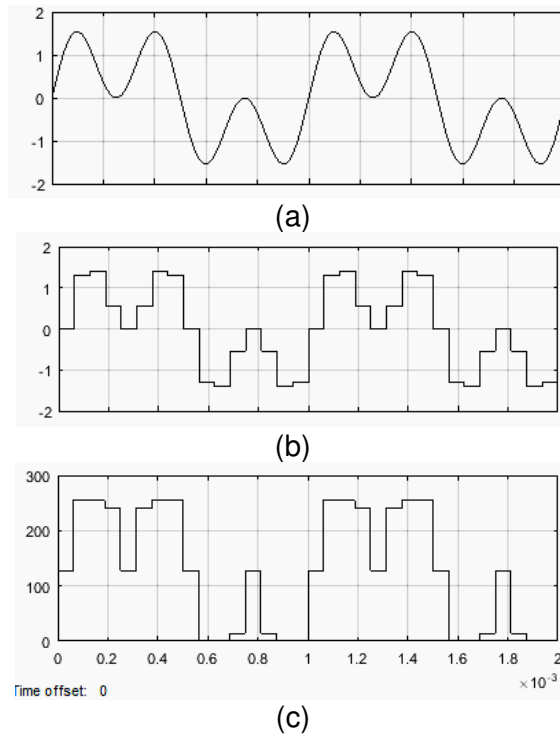


Figura 2.16 (a) Señal de entrada, (b)muestreada y (c)cuantificada de SIGNAL_1 (Scope1)

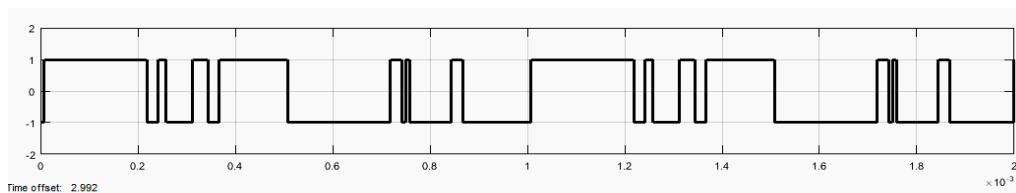


Figura 2.17 Codificación de SIGNAL_1 (Scope2)

Si la selección de entrada es “*SIGNAL_2*”, los datos respectivos a esta señal son captados por un dispositivo de entrada de audio (micrófono), cuyos datos pueden ser ingresados por el usuario y los mismos son almacenados como un archivo multimedia con extensión .wav en el directorio de *Matlab*. Para esta guía se ingresará como nota de audio la palabra “MUNDO” misma que se guardará en el directorio como voicetest.wav, Figura 2.18.

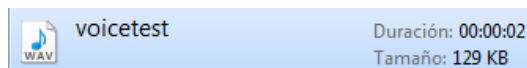


Figura 2.18 Current Directory de Matlab/ voicetest.wav

En la Figura 2.19, se ha considerado parte de la señal ingresada con el fin de tener una mejor visualización del proceso de muestreo y cuantificación, así también en la Figura 2.20 se muestra parte del resultado de la digitalización de la señal de audio ingresada.

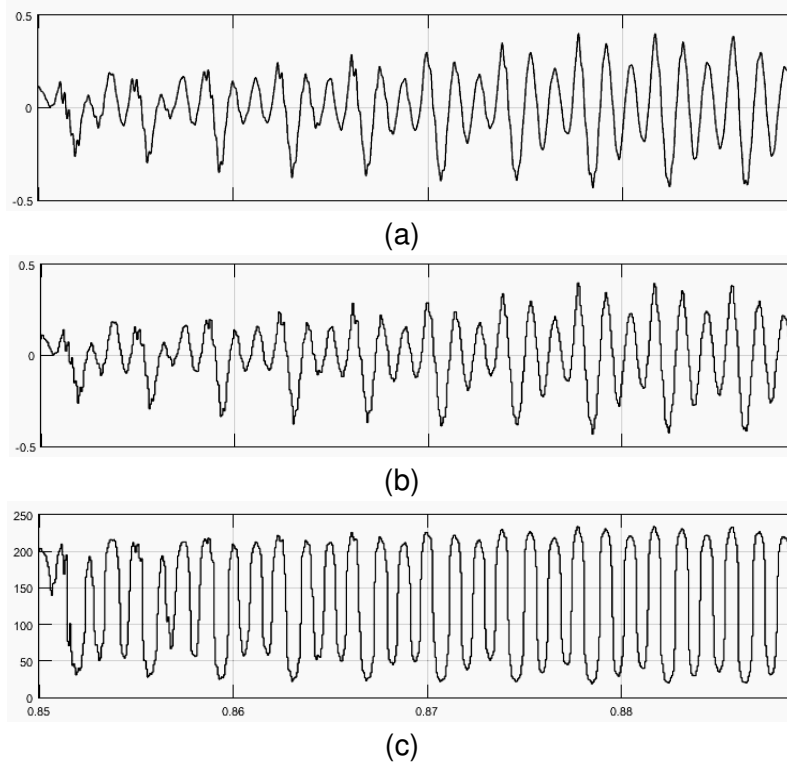
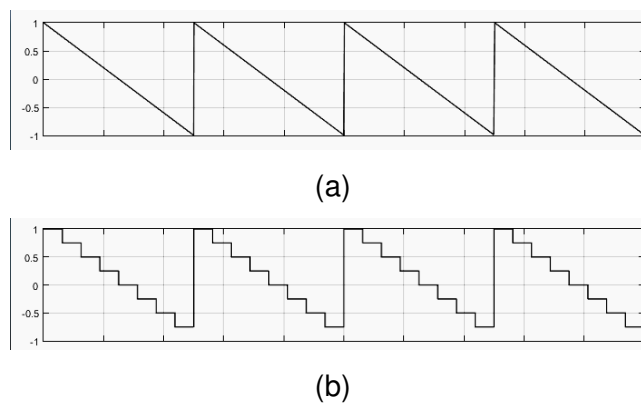


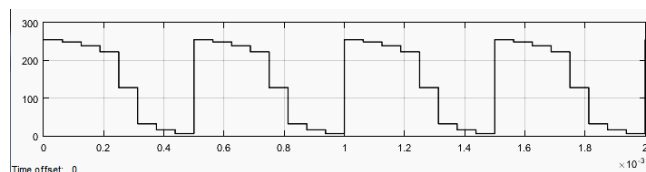
Figura 2.19 (a) Señal de entrada, (b) muestreada y (c) cuantificada de SIGNAL_2 (Scope1)



Figura 2.20 Codificación de SIGNAL_2 (Scope2)

Si la selección de la señal de entrada es "SIGNAL_3", se tiene una señal diente de sierra de 3 [KHz]. La señal de entrada, muestreada y cuantificada se observan en la Figura 2.21.





(c)

Figura 2.21 (a) Señal de entrada, (b) muestreada y (c) cuantificada de SIGNAL_3 (Scope1)

La Figura 2.22 muestra parte de la codificación de la misma.

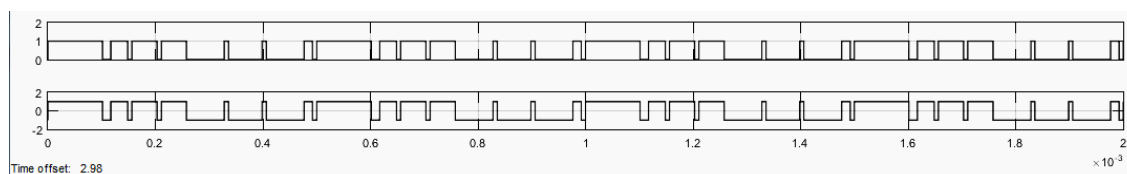


Figura 2.22 Codificación de SIGNAL_3 (Scope2)

2.1.2.4 Conclusiones y recomendaciones

- Simulink es una herramienta cuya interfaz gráfica contiene distintas librerías con componentes de simulación, en la cual es posible crear esquemas para simulación y modelación de sistemas de comunicaciones, procesamiento de señales, etc.
- Simulink aplicado a laboratorios, resulta ventajoso ya que por medio de esta herramienta se incentiva al estudiante al desarrollo de destrezas en la modelación e investigación de esquemas funcionales aplicados a distintas áreas de ingeniería.
- El uso de Simulink en laboratorios, prueba también el ahorro que compete a la compra de dispositivos físicos, para obtener resultados similares. Esta herramienta además cuenta con un tiempo de procesamiento relativamente bajo y de gran eficiencia.
- Los resultados de esta guía incentivan a estudiantes y maestros a interactuar con esta herramienta, aplicado a laboratorios de comunicaciones con el fin de experimentar el procesamiento de cualquier tipo de señal, cuenta con la ventaja de que siendo un software se puede guardar, modificar y compilar cuando sea necesario.
- Se recomienda modificar parámetros de visualización para mejorar la apreciación de resultados, así como también la utilización del sistema de ayuda, para mejorar el entendimiento del uso y conocimiento de los parámetros de configuración de los bloques de simulación.

2.1.3 Guía 3: Una experiencia aplicada a códigos de línea usando OrCAD Capture y PSpice

2.1.3.1 Objetivos

- Familiarizar al estudiante, con el uso de las librerías que comprende OrCAD Capture en aplicaciones sobre sistemas de comunicaciones, conociendo de esta manera las ventajas de su uso en laboratorios basados en software.
- Relacionar al estudiante con el uso de los componentes de *Cadence*, como *Stimulus Editor* y herramientas de análisis de resultados como *PSpice A/D*.
- Establecer una guía de uso de OrCAD Capture simulado sobre PSpice, por medio de una experiencia en la implementación de los códigos de línea AMI-RZ⁴⁷ y Manchester, enfatizando el uso de sus herramientas.
- Comprender la relación entre OrCAD Capture y PSpice, en el desarrollo esquemático e implementación de circuitos.

2.1.3.2 Descripción de guía

Los códigos de línea son usados por la técnica de transmisión en banda base, para transmitir información en forma digital sin necesidad de modular, generalmente esta técnica de transmisión se usa en cortas distancias sobre un medio guiado en la cual se debe garantizar ciertos parámetros como sincronismo, acoplamiento AC al medio de transmisión, transparencia en la transmisión y codificación de bits, entre otros, para adaptar el espectro de la señal digital que se requiere transmitir al canal de transmisión [14].

La señal de datos se representa mediante pulsos, cadenas de unos y ceros lógicos, la misma que para la transmisión requiere una señal de reloj para definir el inicio y fin de cada bit de datos.

Los códigos de línea se clasifican de acuerdo al tipo de polaridad (polar, unipolar, bipolar) y el tiempo de duración de bit (NRZ, RZ). Las señales polares se refieren a que la señal codificada toma valores solo positivos y negativos, las unipolares toma valores solo positivos o negativos y cero, y bipolares implica una señal polar más el nivel cero. En el tiempo de duración RZ se refiere a que cuando se codifica la señal, esta retorna al nivel cero por una fracción del tiempo de duración de bit, mientras que NRZ no.

⁴⁷ AMI-RZ: *Alternate Mark Inversion Return to Zero* (Inversión de marca alternada con retorno a cero)

En esta experiencia se propone realizar una pequeña guía inductiva teórica de la esquematización y simulación del uso de las herramientas que comprende OrCAD Capture simulado sobre PSpice. Como escenario se realizará la esquematización y simulación de los códigos de línea, AMI-RZ y Manchester, aplicado a una secuencia de bits generada a partir del componente *Stimulus Editor* o el elemento fuente *STIM*, para luego ser analizados de acuerdo a los resultados con *PSpice A/D*.

Para la codificación Manchester el 1L se codifica con una transición negativa (de +A a -A) mientras que el 0L se codifica con una transición positiva (de -A a +A) a mitad del periodo de bit, lo cual implica mayor número de transiciones traducido así a mejor sincronismo; y DC nulo para cualquier secuencia que se codifique.


La codificación AMI-RZ codifica el 1L con un pulso positivo o negativo de forma alternada y el 0L se codifica con nivel cero, en este caso como se trata de RZ, los pulsos de los unos no ocupan el intervalo de tiempo de bit completo, en este ejemplo se emplea un código RZ al 50% con el fin de verificar el funcionamiento.

La codificación AMI-RZ cuando se emplea en cadenas largas de unos tiene mejor sincronismo mientras que para cadenas largas de ceros pierde sincronización.

2.1.3.3 Desarrollo de guía

Para la implementación de los respectivos esquemas de simulación se utilizarán las herramientas que se encuentran en la librería *PSpice*, con el fin de evitar errores al momento de ejecutar la simulación.

➤ Implementación del codificador Manchester y AMI-RZ

En el área de trabajo que presenta OrCAD *Capture* CIS, se implementará un circuito simple que tendrá la funcionalidad de un codificador Manchester y será aplicado a una cadena de bits de entrada arbitraria, en otra hoja de proyecto se implementará un circuito más elaborado cuya funcionalidad será un codificador AMI-RZ. Una vez creado el nuevo proyecto de simulación para cada esquema en la herramienta, se procederá agregando la librería *PSpice*, dando clic en el ícono  (*Add Library*) de la ventana *Place Part*, para obtener acceso a los componentes que permitirán armar el circuito, como indica la Figura 2.23.

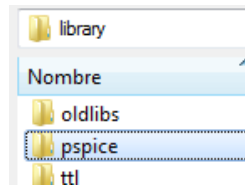


Figura 2.23 Cadence/Library/PSpice (Directorio)

Esta herramienta cuenta con un buscador denominado *Place Part*, el cual facilita la localización de los elementos, tan solo tipiendo el nombre que corresponde al elemento, o también se puede navegar por las librerías, buscando en cada una de ellas a través del *Part List*, los elementos que se requieren desplazar al área de trabajo se da un doble clic.

La construcción de los esquemas de codificación respectivos se realizarán aplicando las compuertas lógicas XOR, AND, NAND, OR, entre otros elementos más. De acuerdo al funcionamiento de cada código de línea Manchester y AMI-RZ se presentan en la Figura 2.24 y Figura 2.25 correspondientemente.

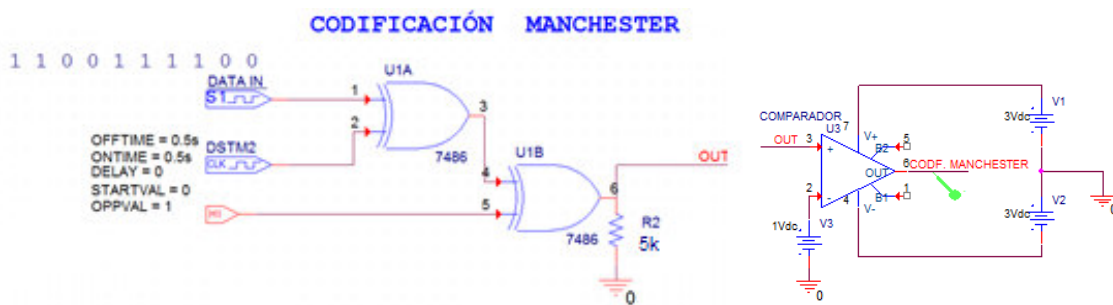


Figura 2.24 Circuito codificador Manchester

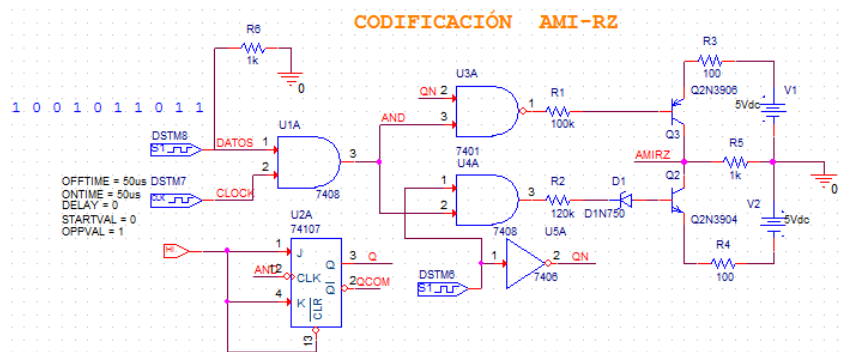


Figura 2.25 Circuito codificador AMI-RZ

➤ Stimulus Editor

Los datos de entrada que se consideran para verificar resultados en esta guía se indican en cada esquema de codificación, 1100111100 y 1001011011 respectivamente. Para generar estos datos se puede crear un archivo con extensión *.stl* (*name.stl*) usando

Stimulus Editor, siendo este un componente que se localiza en el directorio *Cadence/PspiceAccesories*.

Para crear el archivo primero se ejecutará *Stimulus Editor*, luego se hace clic en *File>New*, después en *Stimulus>New* y tras la selección se desplegará una ventana que permitirá escoger la señal a generar, como se indica en la Figura 2.26.

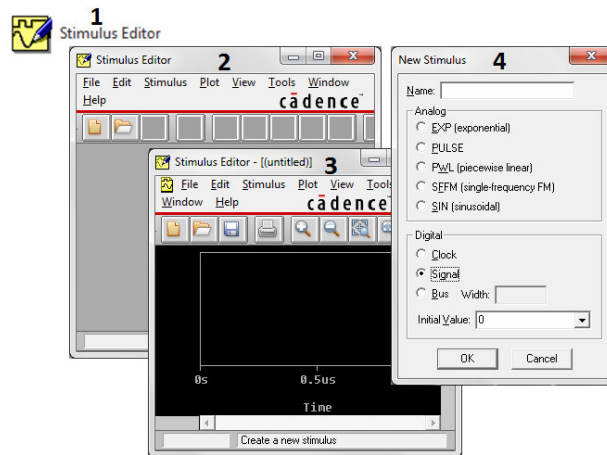


Figura 2.26 Creación del archivo Stimulus, Bits de entrada.

Por otro lado, la librería *Source* cuenta con varios elementos que permiten generar distintos tipos de señales, entre los más destacados se muestran en la Figura 2.27, y para estos ejemplos se usará los elementos DSTM1 y CLK para generar la señal de datos y la señal de reloj respectivamente.

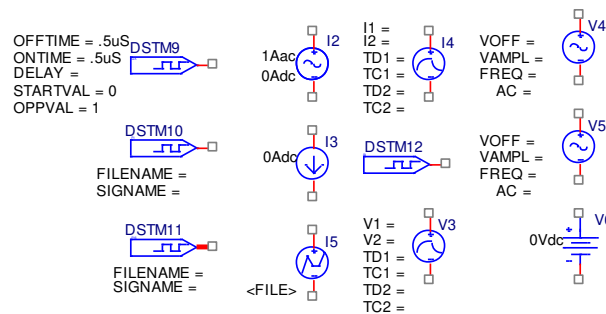



Figura 2.27 Elementos de la librería *Source*

Para la visualización de los datos considerados y la señal de reloj respectiva de cada esquema o cualquier punto donde se requiera analizar resultados, se debe agregar marcadores de voltaje .

Para el caso en que se necesite cargar un archivo con datos de *Stimulus Editor* a un elemento fuente se realizará lo siguiente:

- En la barra de menú se seleccionará *PSpice>New Simulation Profile*.
- Se desplegará una ventana en la que se agregará el nombre que se le dará al perfil y luego clic en crear.
- Tras el paso anterior se desplegará una ventana de configuración, Figura 2.28, donde se agregarán los datos de entrada; es así que en la pestaña *Configuration Files*, en la opción *Browse* se buscará el directorio del archivo de la señal de entrada que se haya creado con extensión *.stl*; una vez localizado, se debe hacer clic en abrir y después clic en *Add to Design*. Seguidamente clic en aplicar y aceptar.

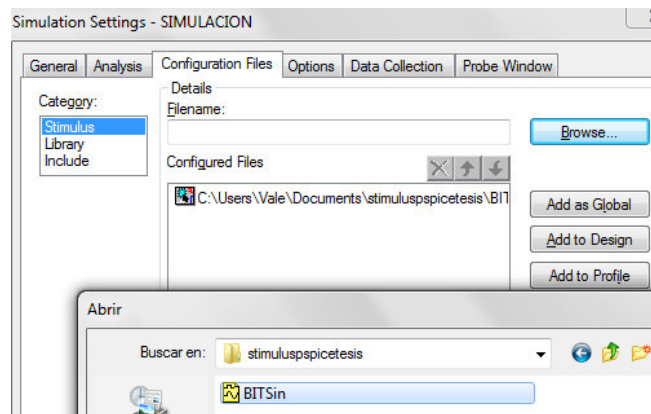
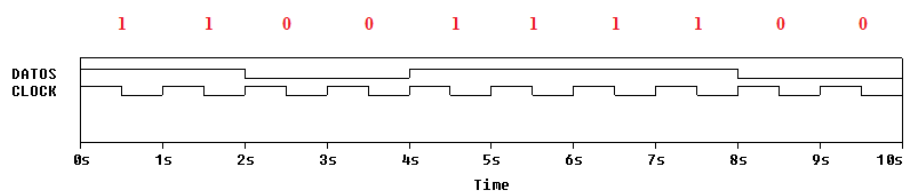
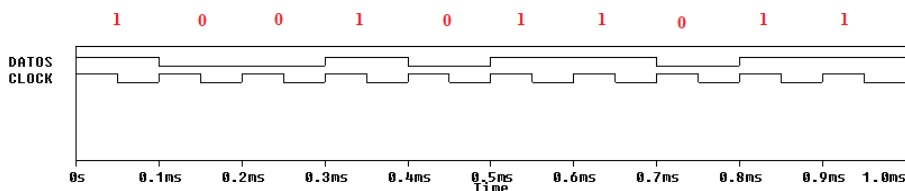


Figura 2.28 Creación del perfil de simulación

En base a lo mencionado para la visualización de los datos de entrada en cada elemento DSTM de cada esquema (Figuras 2.24 y 2.25) y la señal de reloj, aplicando las respectivas marcas de voltaje en cada punto, se ilustra las capturas en la Figura 2.29.




(a)

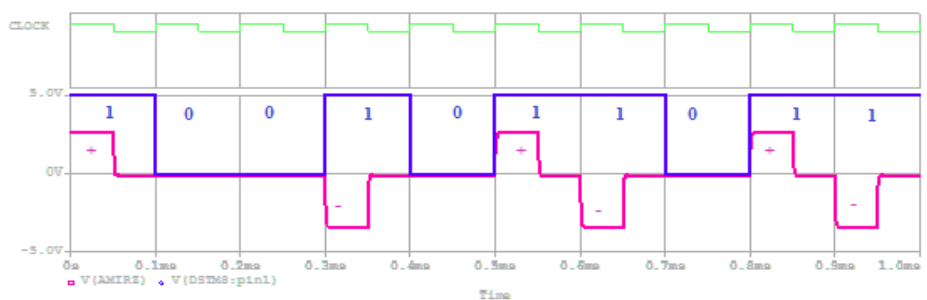


(b)

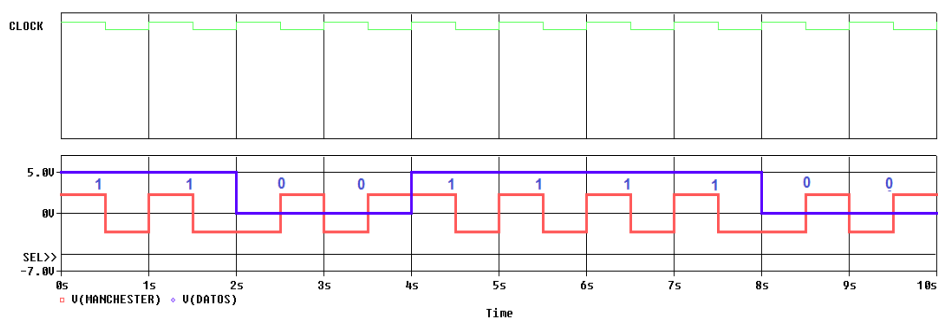
Figura 2.29 (a) Datos de entrada y reloj del codificador Manchester, (b) Datos de entrada y reloj del codificador AMI-RZ

- **Simulación y análisis de resultados**

La visualización de la simulación se obtiene al dar clic en el ícono *Run PSpice* . Donde los resultados se presentarán en *PSpice A/D*, como se muestra en la Figura 2.30.



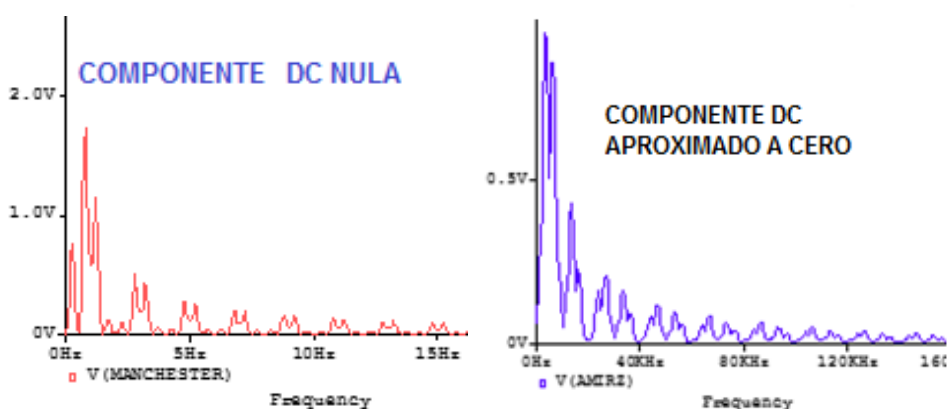
a)



(b)

Figura 2.30 (a) Codificación AMI-RZ, (b) Codificación Manchester

OrCAD PSpice, también permite visualizar el espectro de frecuencia de una señal usando la opción que se presenta como FOURIER, para este caso esta herramienta permitirá verificar si los codificadores esquematizados presentan o no, componente DC, Figura 2.31.



(a)

(b)

Figura 2.31 Espectro de frecuencia (a) Codificación Manchester, (b) Codificación AMI-RZ

En base a la teoría, para el espectro de frecuencia de cada código de línea se debería obtener una forma de señal sampling, sin embargo se tiene una aproximación debido a que

la longitud de datos de la señal es limitada y es por esta razón que se obtiene una resolución muy pobre para el espectro de frecuencia.

2.1.3.4 Conclusiones y recomendaciones

- La aplicación de esta herramienta propone un esquema de enseñanza-aprendizaje de entornos de simulación PSpice, aplicada al laboratorio de Comunicación Digital; ya que en la actualidad los simuladores son herramientas útiles para comprender y entender el comportamiento de circuitos electrónicos, de una forma asequible a alumnos y maestros de la rama de Telecomunicaciones.
- La ventaja de PSpice en conjunto con *OrCAD Capture* en la asignatura de comunicación digital, permite ilustrar esquemas aprendidos teóricamente para mejorar el entendimiento del comportamiento de ciertos sistemas, brindado así la ventaja de verificación y aprendizaje de funcionamiento.
- OrCAD PSpice permite comprobar el funcionamiento de esquemas, con el fin de tener una idea previa antes de su implementación física dado el caso que se requiera; evitando excesivos gastos o errores de implementación.
- OrCAD PSpice permite generar diferentes tipos de señales utilizando la herramienta *PSpice Stimulus Editor*, la cual permite la visualización de las señales a la vez que se está diseñando.
- Las librerías que contiene esta herramienta de software abarcan un sinnúmero de elementos para la esquematización y simulación de circuitos en distintas áreas de ingeniería, como análisis de señales y sistemas, comunicación digital, alta frecuencia, circuitos eléctricos, convertidores estáticos, entre otros.
- Se recomienda agregar todas las librerías de PSpice para evitar errores en la simulación de los mismos.
- Se recomienda para la esquematización de circuitos robustos, probar de acuerdo a lo que se vaya implementando para evitar errores en la simulación y con el fin de verificar el correcto funcionamiento a la salida de los elementos que lo conforman.
- Es recomendable crear un perfil de simulación cada vez que se realice una simulación con algún cambio de datos, para evitar errores de simulación y a la vez para guardar como registro las pruebas que se vayan realizando con el circuito implementado.

2.2 Procesamiento de señales y filtros digitales

En esta sección se desarrollarán escenarios de simulación para prácticas orientadas a procesamiento de señales, como series y transformada de *Fourier*, filtros digitales FIR e IIR. Para ello se utilizarán las herramientas PSpice y Matlab, las cuales permitirán la aplicación de conocimientos de PDS y comunicación digital.

2.2.1 Práctica 01: Introducción al procesamiento digital de señales

2.2.1.1 Objetivos

- Aplicar la herramienta Matlab para desarrollar un archivo script .m, que permita determinar la respuesta de un sistema LTI en tiempo discreto introduciendo el concepto de convolución con una secuencia de entrada determinada.
- Analizar la salida de LTI, para una secuencia de entrada de un pulso rectangular, una onda triangular y una secuencia exponencial de valor real.
- Probar que la herramienta Matlab usando los comandos necesarios permite determinar el análisis matemático para encontrar las respectivas soluciones en aplicaciones con PDS.
- Determinar las ventajas de PDS en sistemas de comunicaciones.
- Establecer el respectivo análisis matemático con una de las respuestas impulso que se considerarán para el análisis de LTI y comprobar con los resultados que se presentarán con Matlab.

2.2.1.2 Marco teórico

En la ingeniería las señales son la base fundamental de las comunicaciones ya que son portadoras de información. La forma más simple de procesamiento de señales es la serie de operaciones que se requiere para mejorar o adaptar la información a un sistema predeterminado. Para comprender más se introducirá el concepto de sistema, el cual se describe como un conjunto de elementos interrelacionados que cumplen una determinada función de acuerdo a los requisitos de una aplicación.

Una señal se define como una magnitud que varía de acuerdo a una variable como puede ser el tiempo, y matemáticamente se representa por funciones de una o más variables independientes.

Las variables de las que depende la señal pueden ser continuas o discretas. El término digital se usa para señales de valores y variables independientes discretas, mientras que el término analógico se usa para valores y variables independientes continuas.

Las señales de gran importancia en los sistemas de comunicación hoy en día son digitales (señales discretas en tiempo y amplitud), las mismas que pueden ser procesadas para un sinnúmero de aplicaciones y pueden ser expuestas a ciertas modificaciones a través de operaciones matemáticas entre secuencias. La forma más general de las señales puede ser continuas en tiempo y amplitud, discretas en tiempo y amplitud, discretas en tiempo y continuas en amplitud, continuas en tiempo y discretas en amplitud.

El procesamiento digital de señales hace referencia al proceso de modificación de una señal digital dentro de un sistema, con el fin de cambiar características o parámetros de las señales para una aplicación en particular. Para proceder al procesamiento de señales se lleva al cabo una serie de operaciones matemáticas, las cuales son especificadas en un programa de software.

➤ **Operaciones en variable dependiente [9]**

Las operaciones que se pueden realizar son multiplicaciones, sumas y escalamiento de secuencias discretas, ecuaciones 2.2, 2.3 y 2.4 respectivamente.

$$y(n) = x_1(n) + x_2(n)$$

Ecuación 2.2 Operación suma de dos secuencias

$$y(n) = x_1(n) \cdot x_2(n)$$

Ecuación 2.3 Operación multiplicación de dos secuencias

$$y(n) = A \cdot x(n)$$

Ecuación 2.4 Operación escalamiento

➤ **Operaciones en variable independiente**

Las operaciones posibles son:

- Desplazamiento en el tiempo, implica un retardo o adelanto de la señal en un valor de k unidades.
- Escalado temporal, es una operación que implica reducción del número de muestras de una señal discreta en el tiempo.
- Sobremuestro, es una operación inversa al escalado temporal.

➤ **Convolución**

La convolución es una operación que se usa para describir la respuesta de un sistema LTI, en la que interviene una entrada y la respuesta al impulso, en PDS tiene algunos usos y puede ser evaluada de diferentes formas, matemáticamente se representa como se indica en la ecuación 2.5.

$$y(n) = LTI[x(n)] = \sum_{k=-\infty}^{\infty} x(k) h(n-k) = x(n) * h(n)$$

Ecuación 2.5 Convolución lineal [9]

Donde:

- $y(n)$ = Salida del sistema LTI de una secuencia $x(n)$
- $h(n)$ = respuesta impulso
- $x(n)$ = secuencia de entrada al sistema.

El diagrama de bloques se representa en la Figura 2.32.

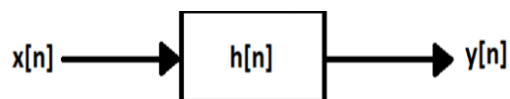


Figura 2.32 Diagrama de bloques de un sistema LTI

➤ Tipos de secuencias de entrada

En PDS se usan algunas secuencias elementales tales como:

- Secuencia impulso unitario: $\delta(n) = 1, si n = 0$ $\delta(n) = 0, si n \neq 0$
En Matlab se utiliza la función *zeros*.
- Secuencia escalón unitario: $u(n) = 1, si n \geq 0$ $u(n) = 0, si n < 0$
En Matlab se utiliza la función *zeros* y *ones*.
- Secuencia exponencial de valor real: $x(n) = a^n, \forall n; a \in R$
En Matlab se escribe como indica la ecuación en este caso de usa el signo “^”
- Secuencia exponencial de valor complejo: $x(n) = e^{(r+jw)n}$
En Matlab se utiliza la función *exp*.
- Secuencia sinusoidal: $x(n) = A * \cos (wn + \theta), \forall n$
En Matlab se utiliza la función *sin* o *cos*.

2.2.1.3 Desarrollo práctico

Se propone realizar la representación matemática de la respuesta a la salida de un sistema LTI caracterizado por una respuesta impulso, para después comprobar con los resultados que se presentan al ser implementados en Matlab. Para realizar esta aplicación se

considerará una secuencia de entrada de un pulso rectangular, una respuesta impulso tipo onda triangular y otra de tipo exponencial de valor real.

- Como primer escenario se deberá presentar un análisis matemático, de la respuesta del sistema LIT considerando una de las respuestas impulso.
- Como segundo escenario se debe desarrollar el script de Matlab para comprobar los resultados del primer punto, el que se debe presentar las respectivas gráficas obtenidas a la entrada y salida del sistema, así como también su correspondiente respuesta impulsiva, mismas que servirán para el análisis de resultados en el siguiente capítulo.
- Se propone realizar un escenario adicional aplicando la convolución, en el que se considerará una respuesta impulsiva diferente, sobre la misma secuencia de entrada, con el fin de evaluar su respectivo análisis.

Para proceder con el desarrollo de la presente práctica se considerará una secuencia de entrada de escalón unitario y una secuencia exponencial de valor real, en la que los datos serán propuestos a modo de ejemplo.

Desarrollo de análisis matemático

- a) Sea un pulso rectangular de la forma de una secuencia escalón unitario, que se describe en la ecuación 2.6 y una respuesta impulsiva de la forma de una secuencia exponencial como se detalla en la ecuación 2.7.

$$x(n) = u(n) - u(n - 15)$$

Ecuación 2.6 Secuencia de entrada

$$h(n) = (0.8)^n u(n)$$

Ecuación 2.7 Respuesta impulsiva

Matemáticamente para determinar la salida del sistema LTI, y para el desarrollo de los cálculos se aplicará la ecuación 2.5, cuya ecuación depende de los valores n y k en donde, para este caso se considerará para fines de evaluación y análisis de resultados un valor de n=14, como se indica en la ecuación 2.8.

$$y(n) = \sum_{k=0}^{14} (1) (0.8)^{(n-k)} u(n - k) = (0.8)^n \sum_{k=0}^{14} (0.8)^{-k} u(n - k)$$

Ecuación 2.8 Respuesta del sistema LTI

Para evaluar la salida, se considerarán tres casos dependiendo de los valores distintos que puede tomar (n, k) en el término $u(n - k)$.

Para el caso 1: El valor n puede tomar valores menores a cero y el valor de k , valores entre 0 y 14; en base a estas condiciones, la ecuación 2.8, la salida del sistema $y(n) = 0$

Para el caso 2: Para este caso se considera que los valores distintos de cero de la señal de entrada y la respuesta impulso no se sobrelapan, es así que para:

$$0 \leq n < 14: u(n - k) = 1, \quad 0 \leq k \leq n$$

De la ecuación 2.5, se tiene el siguiente desarrollo:

$$y(n) = (0.8)^n \sum_{k=0}^n (0.8)^{-k} = (0.8)^n \sum_{k=0}^n [(0.8)^{-1}]^k$$

$$y(n) = (0.8)^n \frac{1 - (0.8)^{-(n+1)}}{1 - (0.8)^{-1}}$$

$$y(n) = 10[1 - (0.8)^{n+1}], \quad 0 \leq n \leq 14$$

Para el caso 3: En este caso se considera que la respuesta impulsiva sobrelapa parcialmente a la secuencia de entrada, es así que para:

$$n \geq 14, \quad u(n - k) = 1, \quad 0 \leq k \leq 14$$

De la ecuación 2.5, se tiene el siguiente desarrollo:

$$y(n) = (0.8)^n \sum_{k=0}^{14} (0.8)^{-k}$$

$$y(n) = (0.8)^n \frac{1 - (0.8)^{-15}}{1 - (0.8)^{-1}}$$

$$y(n) = 10(0.8)^{n-14}[1 - (0.8)^{15}], \quad n \geq 14$$

En este caso como resultado se tiene que la respuesta impulsiva sobrelapa completamente a la secuencia de entrada. El resultado final del sistema LTI, es la suma de las respuestas obtenidas en el caso 1,2 y 3.

Respuesta obtenida utilizando la herramienta Matlab

En base a los datos obtenidos, y al desarrollo realizado con la herramienta Matlab se presenta la visualización de las gráficas obtenidas, con ayuda de ciertos comandos necesarios que se presentan en el *script* del Anexo II, con el fin de determinar la salida del sistema LTI considerando la respuesta impulsiva $h(n)$. La Figura 2.33, ilustra la gráfica de la secuencia de entrada del sistema LTI.

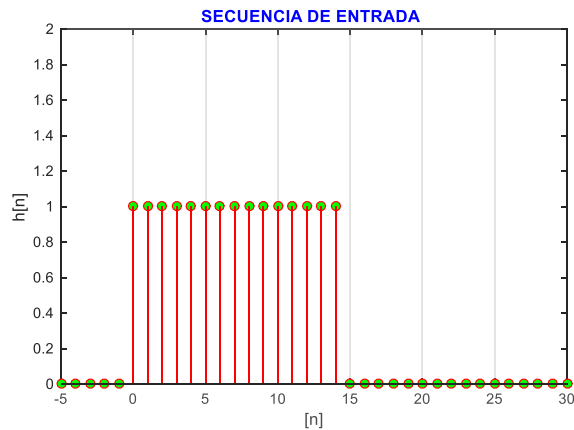


Figura 2.33 Secuencia de entrada del sistema LTI

La Figura 2.34, ilustra la gráfica correspondiente a la respuesta impulsiva del sistema LTI.

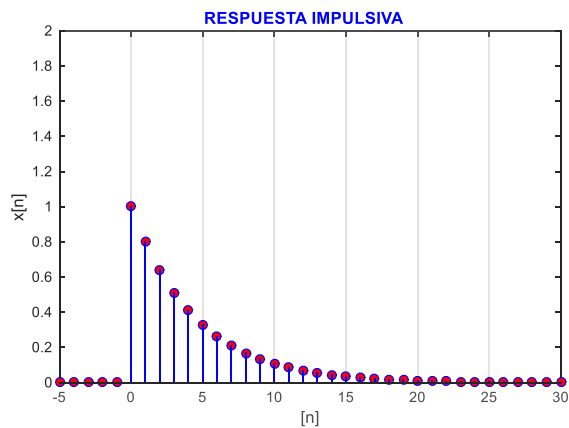


Figura 2.34 Respuesta impulsiva del sistema LTI

La Figura 2.35, ilustra la gráfica correspondiente a la salida del sistema de LTI.

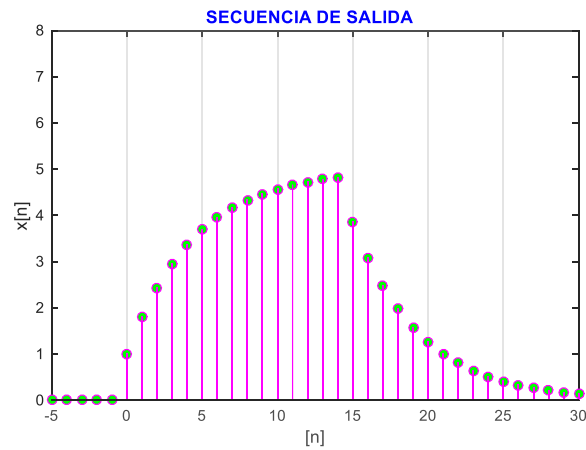


Figura 2.35 Respuesta a la salida del sistema LTI

b) Como siguiente escenario se tiene la variación de la respuesta impulsiva (triangular), Figura 2.36, cuya respuesta a la salida del sistema a través de la convolución con la misma secuencia de entrada del escenario anterior, se tiene los resultados que se ilustran en la Figura 2.37, el respectivo análisis se realizará en el próximo capítulo.

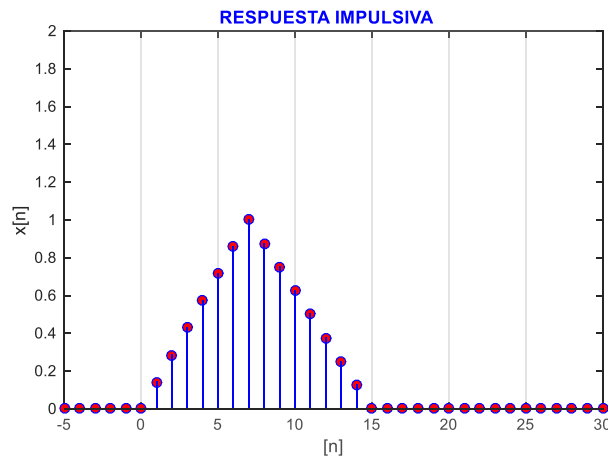


Figura 2.36 Respuesta impulsiva, triangular

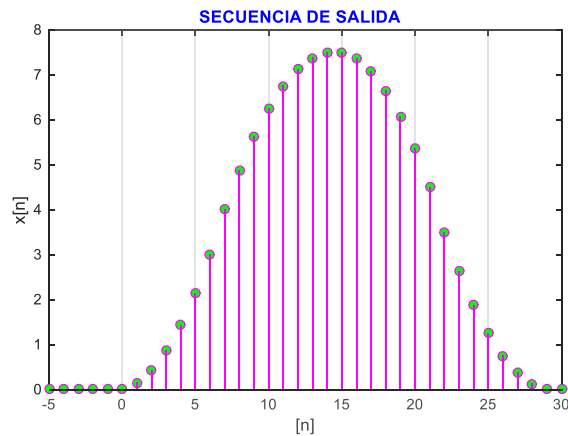


Figura 2.37 Respuesta a la salida del sistema LTI

2.2.2 Práctica 02: Análisis de señales (Series y Transformada de Fourier)

2.2.2.1 Objetivos

- Comprender el análisis de señales en el dominio del tiempo y de la frecuencia.
- Entender el Análisis de *Fourier* para señales periódicas y no periódicas, y determinar la relación entre las Series y Transformada de *Fourier*.
- Conocer las limitaciones de la Transformada de *Fourier*.
- Aplicar herramientas de simulación (MATLAB, SIMULINK u OrcadPspice), para desarrollar escenarios de simulación donde se ejecuta esta temática.

2.2.2.2 Objetivo específico

Mediante el uso de la herramienta OrCAD PSpice esquematizar un generador de pulsos rectangulares, el cual permita observar el efecto sobre la forma del espectro de frecuencias cuando se cambia el ancho de pulso y su período.

2.2.2.3 Marco teórico [9] [11] [14] [17] [26]

Las señales en general se pueden expresar como valores en el tiempo, o como la suma de señales sinusoidales de distinta amplitud y frecuencia. La representación en el dominio del tiempo de la intensidad de una señal continua, varía suavemente en el tiempo, sin saltos o discontinuidades. La intensidad de la señal discreta se mantiene constante durante un intervalo de tiempo, tras el cual cambia a otro valor constante.

De forma general en comunicaciones una señal en el dominio de la frecuencia está directamente relacionada con el Análisis de Fourier, y es aplicado para determinar las componentes de frecuencia que contiene dicha señal.

Análisis de Series de Fourier

Para aplicar el análisis de Fourier, es importante tener en cuenta que para describir señales en el dominio de la frecuencia, hay que identificar qué tipo de señal se va a analizar, las cuales pueden ser periódicas o aperiódicas.

- Series de Fourier

Las Series de Fourier se emplean para describir señales periódicas en tiempo discreto ecuación 2.9, o en tiempo continuo ecuación 2.10, como la descomposición de la suma infinita de funciones sinusoidales de frecuencia $f, 2f, 3f...$ denominados armónicos. Por medio de esta herramienta es posible el analizar su contenido frecuencial, en donde una

función periódica se puede representar matemáticamente en series de Fourier como se indica en la ecuación 2.11.

$$f(t) = f(t + kT) \quad ; \text{ donde } T \rightarrow \text{Periodo}, \neq 0; \quad -\infty < t < +\infty$$

Ecuación 2.9 Función periódica en tiempo continuo

$$f(n) = f(n + kN) \quad \text{donde } N \rightarrow \text{Periodo fundamental} \quad ; \quad \forall n$$

Ecuación 2.10 Función periódica en tiempo discreto

$$f(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} [A_n \cos(kt) + B_n \sin(kt)]; \quad \text{donde } A_n, B_n \in R$$

Ecuación 2.11 Serie de Fourier de una señal periódica, forma trigonométrica

Donde:

- $k = 2 * \pi * n * f_0$, Donde f_0 es la frecuencia fundamental.
- $f_0 = \frac{1}{T}$, T es el periodo de la señal.
- $t \rightarrow$ variable de tiempo

El cálculo de los coeficientes del desarrollo de las series de Fourier de una señal periódica, se obtienen a partir de las definiciones matemáticas (ecuaciones 2.12, 2.13 y 2.14).

$$A_0 = \frac{1}{T} \int_{-T/2}^{T/2} f(t) dt$$

Ecuación 2.12 Coeficiente A_0 de la serie de Fourier

$$A_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos(2 * \pi * f_0 * t) dt$$

Ecuación 2.13 Coeficiente A_n de la serie de Fourier

$$B_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin(2 * \pi * f_0 * t) dt$$

Ecuación 2.14 Coeficiente B_n de la serie de Fourier

➤ Transformada de Fourier

La Transformada de Fourier es una operación que se aplica a señales no periódicas y permite determinar el espectro de dicha señal. El espectro consiste en el conjunto continuo de componentes de frecuencia que conforman la señal original.

La expresión matemática que describe la Transformada de *Fourier*, de una señal no periódica e integrable $g(t)$, se define en la ecuación 2.15.

$$G(\omega) = F[g(t)] = \int_{-\infty}^{\infty} g(t)e^{-j\omega t} dt$$

Ecuación 2.15 Transformada de Fourier

El módulo de la transformada de Fourier de la señal no periódica se denomina espectro de magnitud (amplitud), y la fase se denomina espectro de fase. [14]

Para recuperar u obtener la señal en el dominio del tiempo, se aplica la Transformada Inversa de Fourier, ecuación 2.16.

$$g(t) = F^{-1}[G(\omega)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(\omega) e^{j\omega t} dt$$

Ecuación 2.16 Transformada Inversa de Fourier

Una señal periódica es una señal de potencia, y la señal puede evaluarse a partir de los coeficientes de su correspondiente serie de Fourier. Mientras que las señales de duración finita es decir no periódicas son señales de Energía y se pueden evaluar a partir de la transformada de Fourier.

Aplicaciones

La variedad de campos de aplicación del análisis de Fourier se desarrolla en algunas ramas como la matemática, física, mecánica, comunicaciones, entre otras. Las series de Fourier se aplican donde surgen señales con carácter periódico como son los procesos oscilantes tales como la acústica, óptica, electrónica, etc.

En el campo de las telecomunicaciones, las Series de Fourier tienen una amplia e importante aplicación en la rama de las comunicaciones digitales, ya que con esta operación matemática se puede reconstruir señales a partir de componentes sinusoidales.

En el campo de la medicina, como la ecografía, donde mediante el registro de las vibraciones de las membranas del corazón se obtiene un registro de una señal periódica la cual es útil para el análisis de una de las válvulas del corazón, y mediante la aplicación de un software se puede realizar el cálculo de los primeros coeficientes de *Fourier*, que serán suficientes para determinar un diagnóstico y de esta manera se optimizará costos u otras pruebas médicas costosas y molestosas al paciente.

Entre otras aplicaciones de las series de Fourier son la solución de ecuaciones diferenciales, flujo de calor, cálculo de la temperatura de la tierra, problema isoperimétrico, ecuación de ondas, fórmula de Poisson, entre otras.

En el campo de la ingeniería, la Transformada de Fourier tiene aplicaciones como el cálculo de la cantidad de energía concentrada en cierto ancho de banda, solución rápida y fácil de ecuaciones diferenciales, tratamiento digital de imágenes, diseño de filtros, análisis espectral de la voz, etc.

2.2.2.4 Desarrollo práctico

En un sistema de comunicaciones, el canal de comunicación es el medio para la transmisión de información, constituyendo el camino entre el transmisor y el receptor. Existen dos canales básicos que se clasifican en medios guiados (fibra óptica, cable coaxial, par trenzado) y medios no guiados (microondas, comunicaciones satelitales). Gracias a la transmisión por canales no guiados se puede dar servicio de telefonía celular, llevar señales de radio, entre otros.

Una señal al atravesar por un determinado canal de transmisión sufre ciertas alteraciones que degradan la señal como: la distorsión lineal la cual atenúa ciertas frecuencias con respecto a otras en menor o mayor grado, para contrarrestarla se utiliza la ecualización, la distorsión no lineal implica la generación de nuevas frecuencias y para combatirla se utilizan compansores y expansores. Otra alteración de señal también son la interferencia, el ruido que constituye señales indeseables, además también se puede mencionar que en el medio no guiado se genera interferencia por multitrayectoria debido a que las señales al ser transmitidas no atraviesan edificios (fenómenos de reflexión y difracción), lo cual origina múltiples trayectos para que la señal llegue a su destino.

En telecomunicaciones se aprovecha la multitrayectoria que permite que la señal llegue al receptor por varios caminos diferentes, pero con la desventaja de que cada una de ellas llega con diferentes tiempos de retardo implicando de esta manera que la señal recibida se manifieste mediante ensanchamientos temporales y consecuentemente solapamiento de

pulsos hasta el punto en que el receptor no pueda distinguir correctamente los cambios de estado, este fenómeno se denomina ISI⁴⁸ y puede ser medido utilizando el diagrama del ojo.

En fin, los canales de transmisión tienen varias anomalías que afectan a la señal, donde el mayor problema a vencer es el ruido. En un canal sin ruido la limitación en la velocidad de la señal es el ancho de banda de la misma, debido al ISI, cuya principal causa es la distorsión por retardo. Una de las técnicas para reducir ISI y conseguir una velocidad de transmisión máxima es formar los pulsos antes de la transmisión.

Se propone mediante la herramienta OrCAD PSpice examinar la forma y espectro de una señal correspondiente a un tren de pulsos para diferentes anchos y periodos, y obtenerla a través de un circuito generador de pulsos. Con el fin de ver cómo mediante la reconstrucción de la señal y uso de filtros, se pueda minimizar el problema de la interferencia Intersímbolo (ISI). Para examinar las limitaciones del canal se considerará el espectro de las señales usando análisis de Fourier (Series y Transformada de Fourier).

Para generar el tren de pulsos rectangulares, la herramienta OrCAD PSpice cuenta con el elemento VPULSE, que se encuentra en la librería *source* y dispone de ciertos parámetros que al ser modificados adecuadamente originan la señal deseada como se ilustra y detalla en la Figura 2.38.

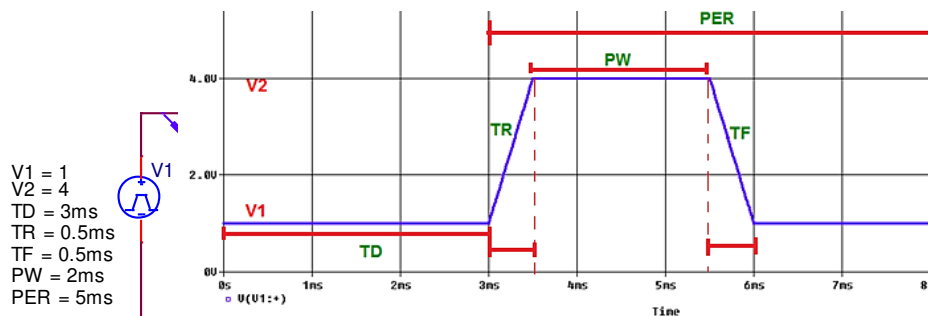


Figura 2.38 Parámetros del elemento VPULSE

Donde:

- *V1* → *voltaje en bajo*
- *V2* → *voltaje en alto*
- *TD* → *tiempo de retardo de la señal*
- *TR* → *tiempo de subida*
- *TF* → *tiempo de bajada*

⁴⁸ ISI: *Intersymbol Interference (Interferencia Intersímbolo)*

- *PW* → tiempo de la señal que permanece en *V2*
- *PER* → período

En este caso, como se desea obtener un tren de pulsos rectangulares, los parámetros clave que se deberán modificar son *TR* y *TF*, de tal manera que al colocar valores significativamente pequeños con respecto al valor del ancho de pulso, las transiciones verticales se van tornando perpendiculares obteniendo de esta manera la señal.

En base a lo indicado, se construirá el circuito generador del tren de pulsos rectangulares que se ilustra en la Figura 2.39, el cual será el esquema fundamental para realizar las próximas pruebas y modificaciones de parámetros requeridos como ancho de pulso y período, mismos que permitirán realizar el respectivo análisis de resultados en el siguiente capítulo.

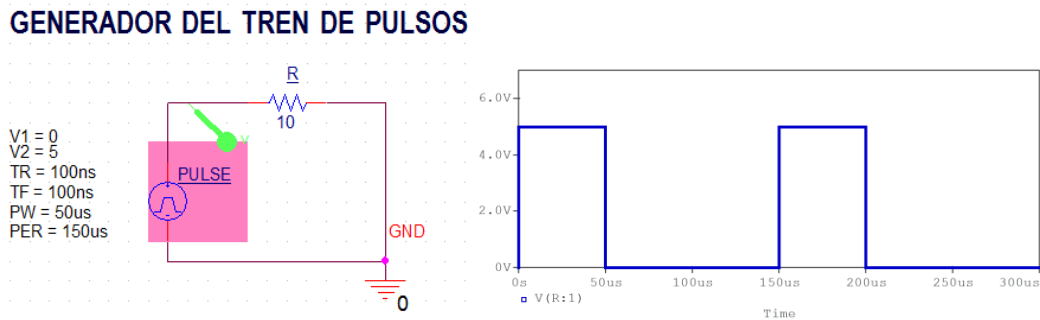


Figura 2.39 Generador de pulsos

De acuerdo a los parámetros descritos en el generador del tren de pulsos de la Figura anterior, se tiene el espectro de dicha señal en la Figura 2.40.

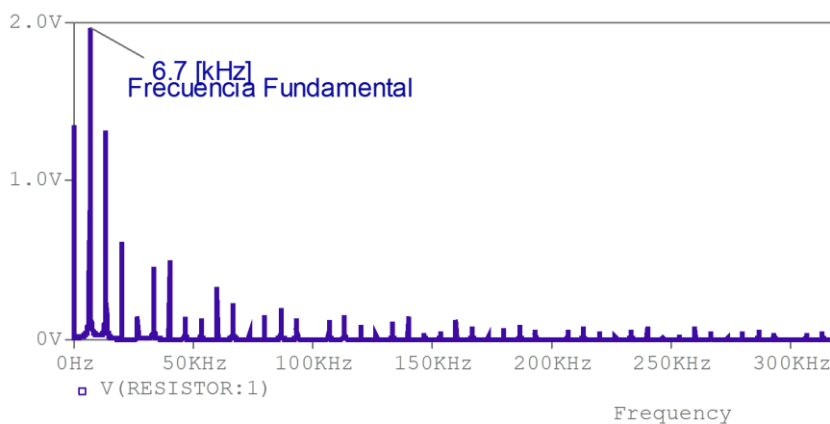


Figura 2.40 Espectro del tren de pulsos, *PW*=50 [us], *PER*=150 [us]

Las Figuras 2.41 y 2.42, muestran los resultados gráficos de la forma del tren de pulsos rectangulares en el tiempo y su espectro, considerando un ancho de pulso de 20 [us] y un período de 150 [us].

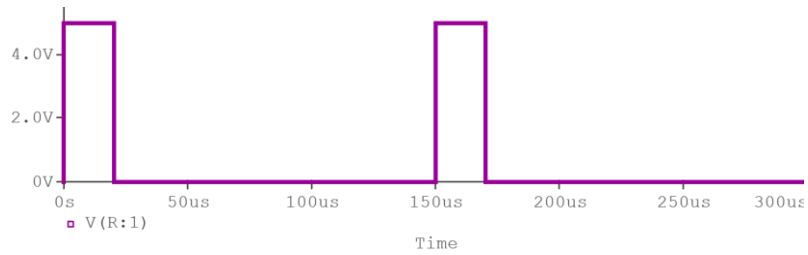


Figura 2.41 Tren de pulsos, PW=20 [us], PER=150 [us]

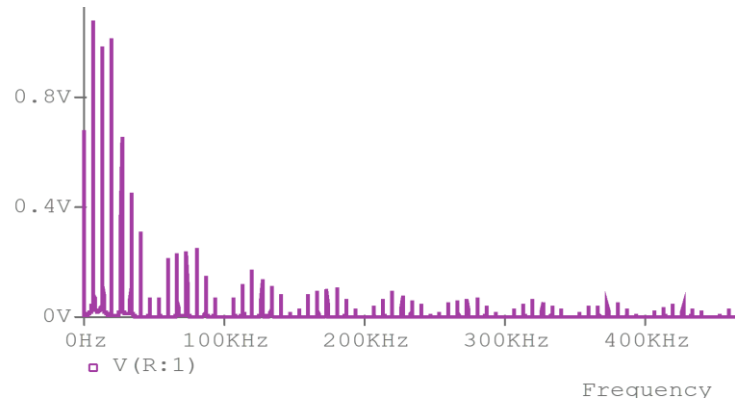


Figura 2.42 Espectro del tren de pulsos, PW=20 [us], PER=150 [us]

Las Figuras 2.43 y 2.44, muestran los resultados gráficos de la forma del tren de pulsos rectangulares en el tiempo y su espectro, considerando un ancho de pulso de 20 [us] y un período de 1000 [s].

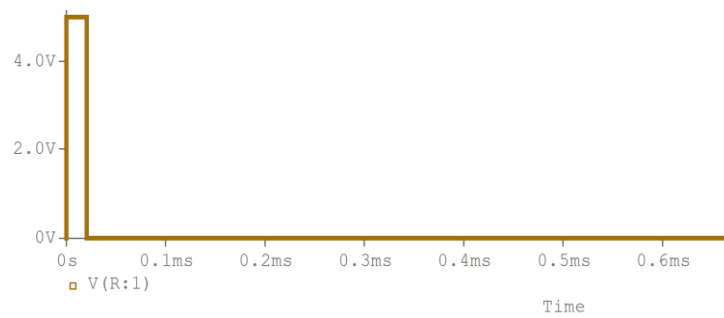


Figura 2.43 Tren de pulsos, PW=20 [us], PER=1000 [s]

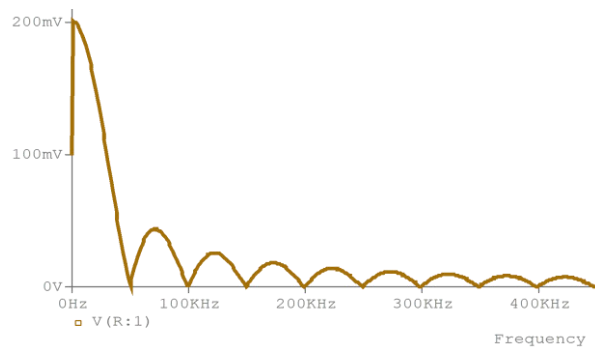


Figura 2.44 Espectro del tren de pulsos, PW=20 [us], PER=1000 [s]

La transmisión de pulsos digitales infinitamente estrechos sin distorsión significa que la velocidad de transmisión aumenta a tal punto que requiere un canal con ancho de banda infinito y condiciones de respuesta de fase lineal que no son realizables físicamente. A través de la remodelación del circuito generador de pulsos se podrá minimizar el problema de ISI, ya que por medio de un filtro pasa bajo se puede limitar el requerimiento de ancho de banda.

Para el desarrollo del circuito denominado sintetizador de onda cuadrada, se determinará los armónicos de la onda cuadrada por análisis de *Fourier* usando la FTT de la herramienta PSpice, para luego por medio de la herramienta VSIN ir representando cada uno de los armónicos en un circuito en serie para dar forma a lo que será finalmente la onda cuadrada.

Una de las características de los filtros RC es limitar el ancho de banda y para esta práctica se diseñará un filtro tipo RC de la forma como se ilustra en la Figura 2.45.

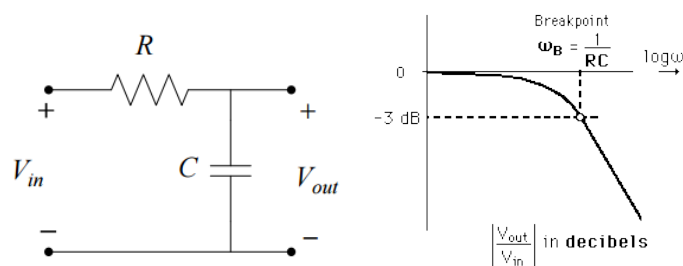


Figura 2.45 Filtro pasa bajo RC [29]

Cuya función de transferencia, que determina el comportamiento del filtro pasa bajo, se describe matemáticamente en la ecuación 2.17.

$$H(s) = \frac{V_{out}}{V_{in}} = \frac{1}{1 + j\omega RC}$$

Ecuación 2.17 Función de transferencia del filtro RC pasa bajo

Donde, la frecuencia de corte f_c se da cuando la reactancia capacitiva X_c es igual a la resistencia (R), por lo tanto se tiene la ecuación 2.18: [29]

$$X_c = \frac{1}{j\omega C} \rightarrow R = \frac{1}{2 * \pi * f_c * C} \quad ; \quad \text{donde: } \omega = 2\pi f_c$$

Ecuación 2.18 Cálculo de frecuencia de corte en el filtro RC

En base a los parámetros de la Figura 2.39, se considerará para el siguiente escenario un tren de pulsos con PW=75 [us] y un PER=150 [us], para lo cual se calculará la frecuencia fundamental mediante la ecuación 2.11, la misma que resulta: $f_o = \frac{1}{T} = \frac{1}{150 \text{ us}} = 6.66 \text{ [KHz]}$.

La frecuencia fundamental será un parámetro importante para realizar el cálculo de los armónicos. En el simulador *Pspice A/D*, se configurará en el *Edit Simulation Profile* en la opción *Output file options* del circuito generador del tren de pulsos, el valor de la frecuencia fundamental y el número de armónicos que se desee evaluar para la reconstrucción de la señal en este caso se ha considerado evaluar para 15 armónicos, como se muestra en la Figura 2.46.

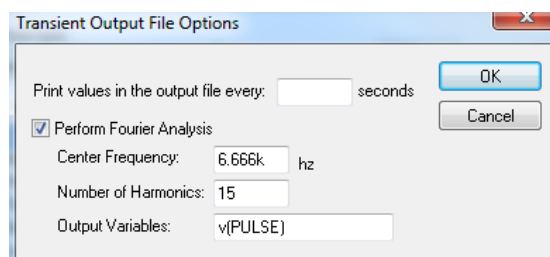


Figura 2.46 Cuadro configuración de Output file options

Una vez configurado, se extraerá la tabla de valores de las componentes de *Fourier* del pulso rectangular. Para lo cual se debe hacer clic en “*Run PSpice*”, luego se desplegará la ventana *PSpice A/D*, seguidamente clic en *View>Output File*, donde se encontrarán los datos que se muestran en la Tabla 2.4.

Tabla 2.4 Tabla de valores de las componentes de Fourier de la respuesta v(PULSE)

DC COMPONENT = 2.533163E+00					
HARMONI C	FREQUENCY	FOURIER COMPONEN T	NORMALIZED COMPONENT	PHASE (DEG)	NORMALIZE D PHASE (DEG)
NO	(HZ)				
1	6.666E+03	3.182E+00	1.000E+00	-1.224E+02	0.000E+00
2	1.333E+04	6.629E-02	2.083E-02	-1.548E+02	9.000E+01
3	2.000E+04	1.058E+00	3.326E-01	-7.265E+00	3.600E+02
4	2.666E+04	6.617E-02	2.079E-02	-3.969E+01	4.500E+02
5	3.333E+04	6.320E-01	1.986E-01	1.079E+02	7.200E+02
6	4.000E+04	6.598E-02	2.073E-02	7.547E+01	8.100E+02
7	4.666E+04	4.483E-01	1.409E-01	-1.370E+02	7.200E+02
8	5.333E+04	6.571E-02	2.065E-02	-1.694E+02	8.100E+02
9	5.999E+04	3.454E-01	1.085E-01	-2.180E+01	1.080E+03
10	6.666E+04	6.537E-02	2.054E-02	-5.422E+01	1.170E+03
11	7.333E+04	2.793E-01	8.778E-02	9.336E+01	1.440E+03
12	7.999E+04	6.495E-02	2.041E-02	6.094E+01	1.530E+03
13	8.666E+04	2.330E-01	7.323E-02	-1.515E+02	1.440E+03
14	9.332E+04	6.446E-02	2.026E-02	1.761E+02	1.890E+03
15	9.999E+04	1.986E-01	6.242E-02	-3.633E+01	1.800E+03
TOTAL HARMONIC DISTORTION = 4.490043E+01 PERCENT					

A través de las componentes de Fourier es posible reconstruir el tren de pulsos, donde se considera la frecuencia fundamental y los armónicos impares, en el caso de una señal cuadrada las componentes pares se anulan [28], para demostrar se puede aplicar las ecuaciones 2.12, 2.13 y 2.14, de donde se obtiene que para las componentes pares $A_n=0$, y para las componentes impares B_n es un valor determinado como se puede verificar en la Tabla 2.4 extraída con ayuda del software de simulación PSpice.

La Figura 2.47, se muestra el esquema del circuito, el cual consta de generadores en serie comportándose como un sumador de ondas *seno*, cada uno con los parámetros de los armónicos 1, 3, 5, 7, 9, 11, 13 y 15 correspondientemente.

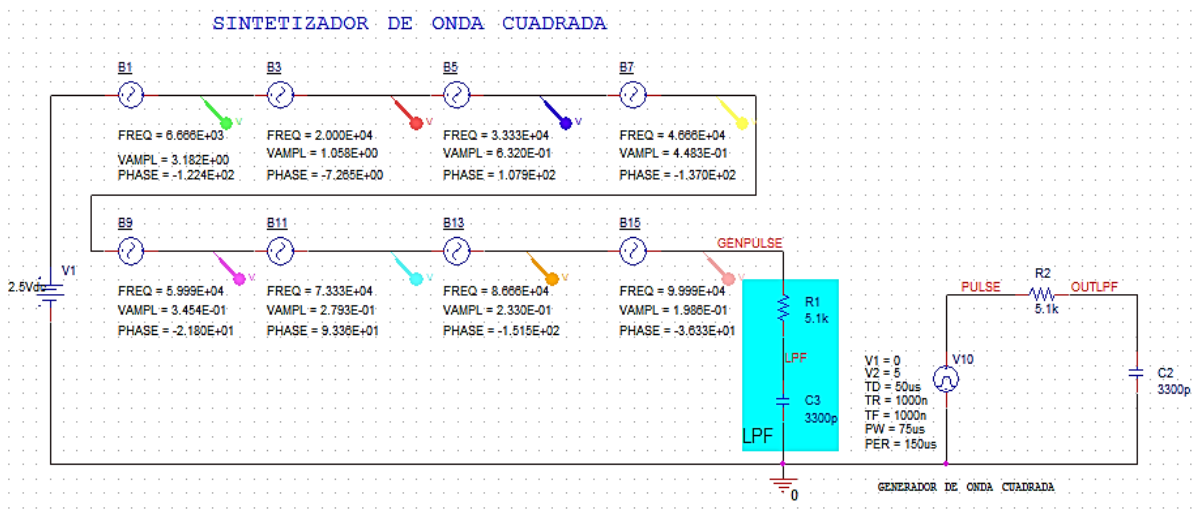
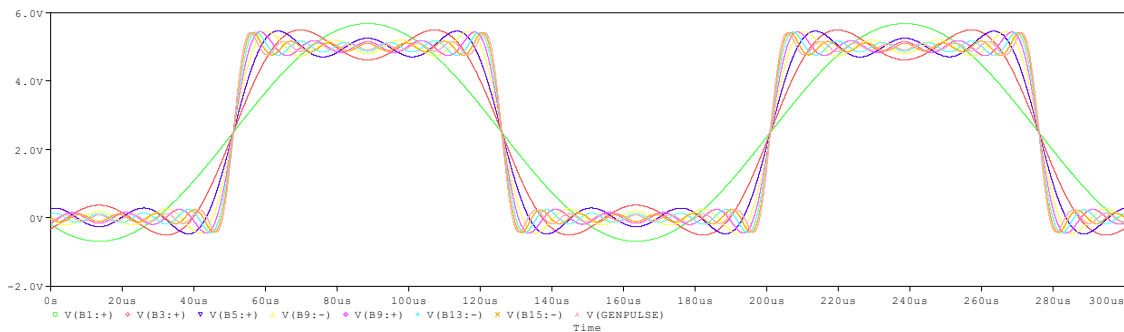
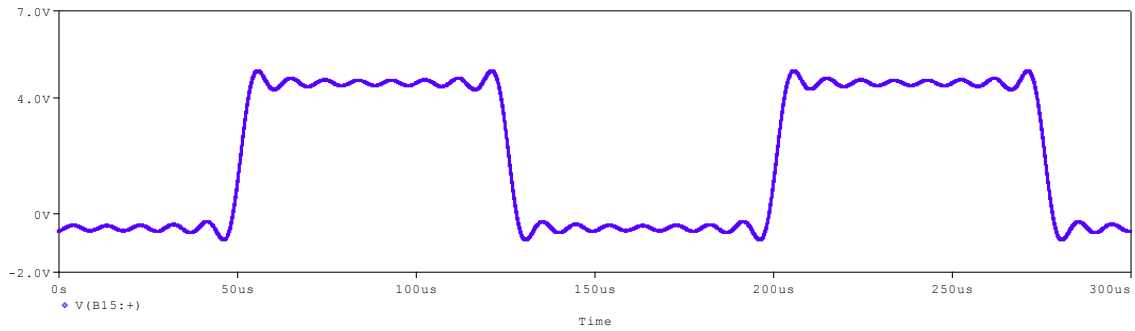


Figura 2.47 Circuito sintetizador de onda cuadrada

Los resultados de este sintetizador de pulso se ilustran en la Figura 2.48, en donde en la parte (a) se muestra la suma de la serie de armónicos y cómo se van sintetizando en la forma de una onda cuadrada; y, la parte (b) muestra claramente el resultado de la suma de los ocho armónicos impares considerados.



(a)



(b)

Figura 2.48 En (a) se muestra los armónicos 1, 3, 5, 7,9,11,13 y 15 de acuerdo al circuito de la Figura 2.47, (b) muestra el resultado de la suma de los ocho armónicos de (a).

En la Figura 2.49 se ilustra las componentes de frecuencia de los armónicos que conforman el sintetizador de onda cuadrada.

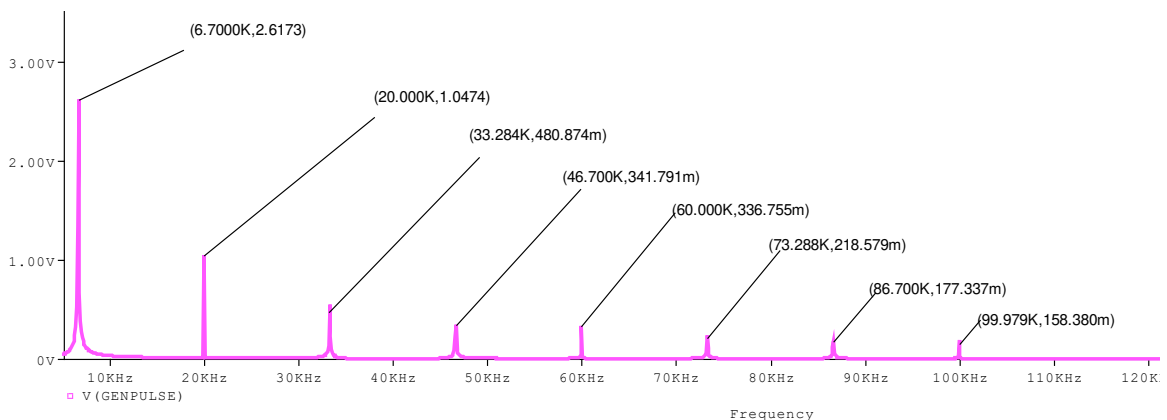


Figura 2.49 Componentes de frecuencia de la salida del sintetizador

Seguidamente, el circuito sintetizador se conecta en serie a un circuito tipo filtro RC como el esquema de la Figura 2.45, el cual se diseña en base al criterio paramétrico de la Figura 2.6, donde la frecuencia de corte se calcula con la ecuación 2.18. Para el cálculo se asume una resistencia estándar en el mercado de 5.1 [Kohm] y un capacitor de 3300 [pF], ecuación 2.19.

$$f_c = \frac{1}{2 * \pi * 5.1K * 3300p} = 9.4 [KHz]$$

Ecuación 2.19 Cálculo de la frecuencia de corte

De acuerdo al cálculo realizado; a la salida del filtro del circuito sintetizador y del generador del tren de pulsos que se ilustra en la Figura 2.47, se tiene las gráficas que muestran en la Figura 2.50, en el dominio del tiempo.

Donde la señal en color rojo indica la gráfica correspondiente a la señal del filtro y la señal en color azul a la señal tren de pulsos, tanto para el circuito sintetizador como para el generador del tren de pulsos.

Y en el dominio de la frecuencia se aprecia en la figura 2.51, donde la línea color rojo muestra la limitación de ancho de banda establecida por el filtro pasa bajo, tanto para el sintetizador como en el generador del tren de pulsos, lo cual se explica de mejor manera en el respectivo análisis de resultados.

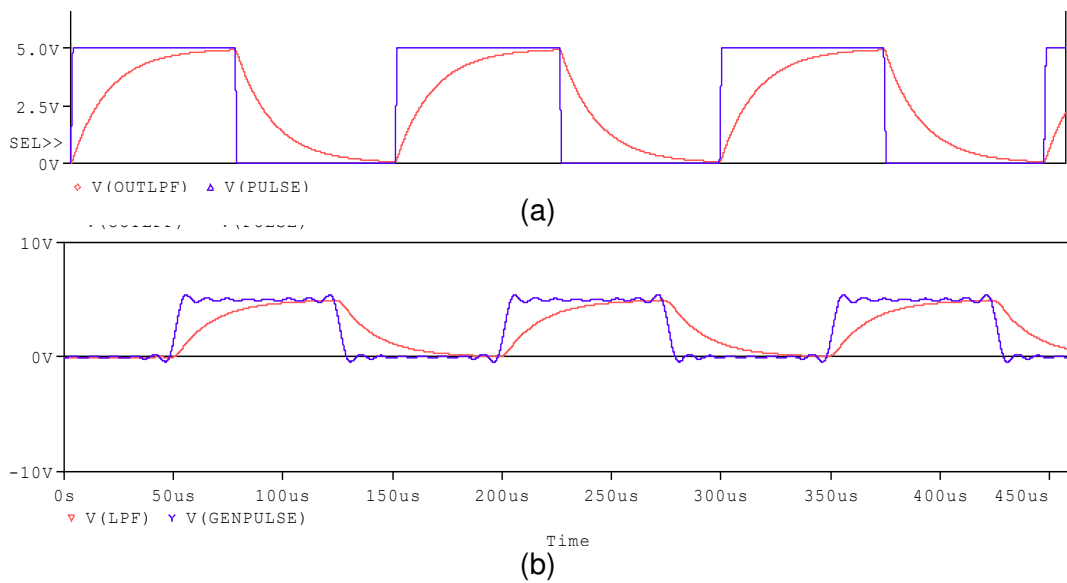


Figura 2.50 Salida de los Filtros pasa bajo RC en el dominio del tiempo del (a) circuito generador de onda cuadrada (b) circuito sintetizador de onda cuadrada

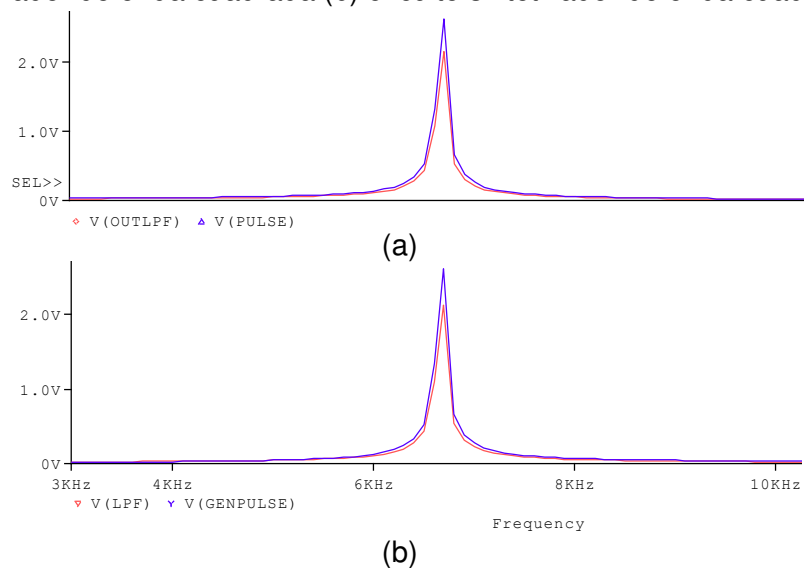


Figura 2.51 Salida de los Filtros pasa bajo RC en el dominio de la frecuencia del circuito (a) generador de onda cuadrada (b) sintetizador de onda cuadrada

El análisis de resultados se presenta en el siguiente capítulo.

2.2.3 Práctica 03: Diseño y simulación de un filtro FIR

2.2.3.1 Objetivos

- Conocer los diferentes tipos de filtros digitales de fase lineal que pueden ser obtenidos.
- Conocer las técnicas, métodos y tipos de diseño de un filtro FIR.
- Entender qué requerimientos y parámetros son necesarios definir, para el diseño de un filtro digital FIR.
- Utilizar la herramienta Matlab para analizar resultados realizando un archivo script para diseñar y simular un filtro digital tipo FIR, aplicando un método conveniente.
- Conocer las ventajas y campos de aplicación de este tipo de filtros en comunicaciones digitales.

2.2.3.2 Marco teórico [9] [30]

Los filtros digitales se caracterizan por operar sobre señales digitales. Y no son más que sistemas que se usan para modificar el espectro de una señal. Los filtros digitales tienen aplicaciones en el campo del procesamiento digital de señales, procesamiento digital de audio, procesamiento de voz, procesamiento de imágenes, entre otros. Se utilizan generalmente para acondicionar señales de ciertas características específicas, separar o extraer información de acuerdo a los requerimientos del usuario.

En general los filtros digitales se caracterizan por ser sistemas predecibles, flexibles, simulables y precisos. El diseño de filtros digitales involucra aproximación, cuya forma deberá satisfacer las especificaciones de una aplicación.

En términos de su respuesta al impulso los filtros digitales se clasifican en FIR (*Finite Impulse Response*) e IIR (*Infinite Impulse Response*). Un filtro FIR, es aquel que tiene una respuesta impulso de duración finita, además son filtros de tipo no recursivo es decir la respuesta a una señal impulso de entrada tiene un número finito de términos no nulos (polos), razón por la cual se caracterizan por ser filtros estables. Las características principales de los filtros FIR son:

- Los filtros FIR cuentan con la ventaja de que pueden ser diseñados para tener una exacta fase lineal, esto quiere decir que presentan propiedades en la simetría de los coeficientes, lo cual es importante en la transmisión de datos y ventajoso en aplicaciones biomédicas, audio, imágenes, video, etc.

- Los filtros FIR requieren mayor tiempo de procesamiento con respecto a los filtros IIR ya que requieren más coeficientes y por lo tanto más memoria, lo cual puede resultar una desventaja en aplicaciones de tiempo real.
- Sus requerimientos en cuanto a recursos son mayores respecto a los filtros IIR y dada la mayor cantidad de operaciones estos filtros son más sensibles a errores.
- No tienen contraparte analógica, esto quiere decir que no se puede diseñar filtros FIR a partir de prototipos analógicos.
- El atractivo de estabilidad es porque todos los polos se encuentran en el origen en el plano Z, lo cual es importante para el diseño de filtros pasa bajo, pasa alto, pasa banda o elimina banda.

Estructura del filtro FIR [9]

La representación matemática de la ecuación diferencial del filtro FIR tiene la siguiente forma, ecuación 2.20:

$$y(n) = b_0x(n) + b_1x(n-1) + \dots + b_{M-1}x(n-M+1)$$

Ecuación 2.20 Ecuación diferencial del filtro FIR

Donde M es el número de coeficientes y (M-1) es el orden del filtro, con la respuesta impulso $h(n)$ dada por la ecuación 2.21:

$$h(n) = \begin{cases} b_n, & 0 \leq n \leq M-1 \\ 0, & \text{otro caso} \end{cases}$$

Ecuación 2.21 Respuesta impulso de un filtro FIR

Los filtros FIR pueden ser diseñados para que tengan respuestas de fase lineal, para lo cual se consideran las siguientes estructuras, en donde cada una está compuesta de tres elementos básicos que son: sumadores, multiplicadores y elementos de retardo como se detalla en la Figura 2.52.

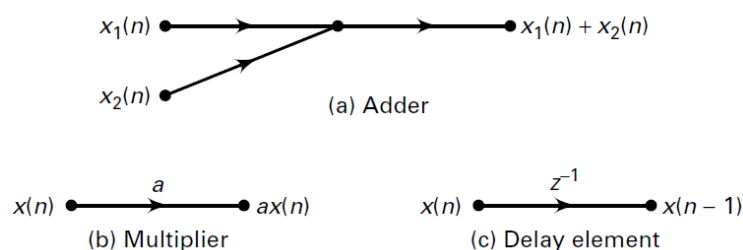


Figura 2.52 Elementos básicos de una estructura [9]

1. Forma Directa, la ecuación 2.20 es implementada directamente como el ejemplo de la estructura de la Figura 2.53 para M=5.

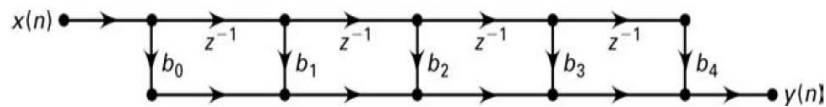


Figura 2.53 Estructura de forma directa, para M=5 [9]

2. Forma de cascada, consiste en descomponer en factores de segundo orden la función del sistema de FIR ($H(z) = \sum_{n=0}^{M-1} b_n z^{-n}$), ecuación 2.22:

$$H(z) = b_0 \prod_{k=1}^K (1 + B_{k,1}z^{-1} + B_{k,2}z^{-2})$$

Ecuación 2.22 Respuesta impulso de un filtro FIR [9]

Donde k es la media del valor de los coeficientes y $B_{k,1}$ y $B_{k,2}$, son los coeficientes de las secciones de segundo orden, como se ilustra en el esquema del ejemplo de la Figura 2.54, para M=7.

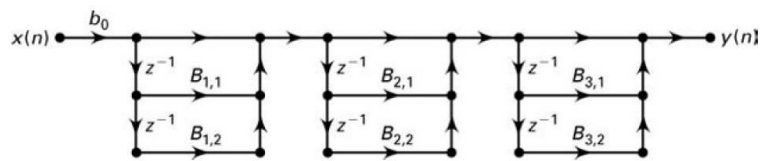


Figura 2.54 Estructura de forma cascada, M=7 [9]

3. Forma de fase lineal, se da cuando el filtro FIR de fase lineal se caracteriza por una respuesta impulso simétrica o asimétrica. Las condiciones de simetría hacen que se reduzcan las multiplicaciones de la estructura alrededor de la mitad. En la Figura 2.55 se ilustra un ejemplo de este tipo de estructura para un valor de M=6.

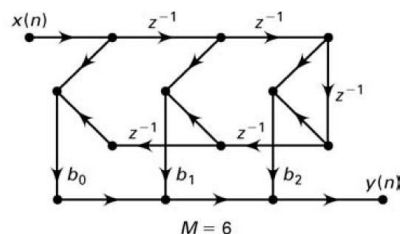


Figura 2.55 Estructura de la forma fase lineal, (respuesta impulso simétrica) [9]

4. Forma Frecuencia de muestreo, $H(z)$ del filtro FIR puede ser reconstruida a partir de muestras, como se ilustra en el ejemplo esquemático de la Figura 2.56 para M=4.

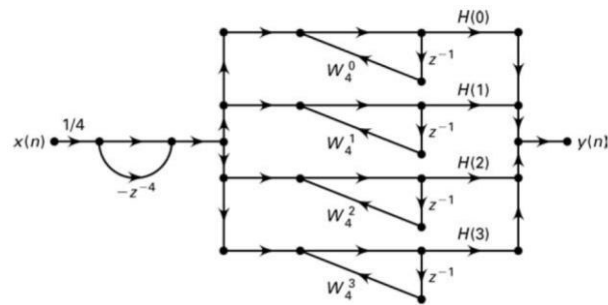


Figura 2.56 Forma Frecuencia de muestreo, para $M=4$

Diseño de un filtro FIR

Para diseñar un filtro FIR hay que considerar ciertas especificaciones, las cuales se definen de acuerdo a la aplicación que se vaya a ejecutar, las cuales se muestran en la Figura 2.57.

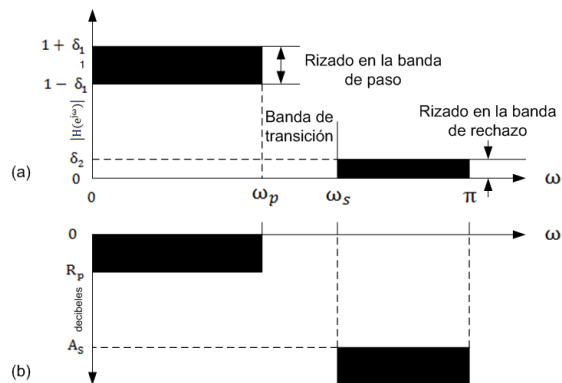


Figura 2.57 Especificaciones de un filtro FIR, (a) absolutas (lineal) (b) relativas (logarítmica) [9]

Las especificaciones absolutas o lineales se describen de la siguiente manera:

- El intervalo $[0, \omega_p]$ es la banda de paso, y δ_1 es el factor de rizado respectivo.
- El intervalo $[\omega_s, \pi]$ es la banda de rechazo, y δ_2 es el factor de rizado respectivo.
- El intervalo $[\omega_p, \omega_s]$ es la banda de transición, no hay restricciones en la respuesta en magnitud en esta banda.

Las especificaciones relativas o logarítmicas se describen de la siguiente manera:

- R_p , es la tolerancia en decibelios, en la banda de paso.
- A_s , es la atenuación en decibelios, en la banda de rechazo.

Una de las características de los filtros FIR es que pueden diseñarse para que tengan una respuesta de fase lineal exacta, para lo cual existen cuatro tipos de filtros FIR de fase lineal que se resumen en la Tabla 2.5, si es par o impar y si $h(n)$ tiene simetría positiva (simétrico) o negativa (anti-simétrico).

Tabla 2.5 Tabla de tipos de filtros de fase lineal

Tipo	Simetría	Términos	Descripción gráfica de $h(n)$
Tipo I	Respuesta Impulso simétrica	Impar M	
Tipo II	Respuesta Impulso simétrica	Par M	
Tipo III	Respuesta Impulso anti-simétrica	Impar M	
Tipo IV	Respuesta Impulso anti-simétrica	Par M	

Para el diseño de los filtros FIR, existen tres métodos principales, los cuales son:

- Diseño por el método de ventanas.
- Diseño por muestreo en frecuencia.
- Diseño por método aplicando la técnica de rizado óptimo.

Para el desarrollo de esta práctica se hará hincapié en el método de ventanas, ya que es la técnica más usada, pues permite obtener filtros de características más exactas.

Método de ventanas

La idea de diseñar, es elegir un adecuado filtro ideal de frecuencia selectiva que tenga una respuesta impulso de duración finita con carácter no causal ($h_d(n)$), para luego por algún método de diseño obtener un filtro FIR causal que sea de fase lineal.

La respuesta impulso para diseñar un filtro está dada por, ecuación 2.23:

$$h(n) = h_d(n) \cdot w(n)$$

Ecuación 2.23 Respuesta impulso para diseño de un filtro

Tabla 2.6 Respuestas de diseño de los tipos de filtros [30]

Filtro pasa alto	$hd(n) = \begin{cases} \frac{\sin \pi (n - \frac{M-1}{2}) - \sin \omega_c (n - \frac{M-1}{2})}{\pi (n - \frac{M-1}{2})} \\ 1 - \frac{\omega_c}{\pi} \end{cases}, \text{Si } n = \frac{M-1}{2}$
Filtro pasa bajo	$hd(n) = \begin{cases} \frac{\sin \pi (n - \frac{M-1}{2}) - \sin \omega_c (n - \frac{M-1}{2})}{\pi (n - \frac{M-1}{2})} & n \neq \frac{M-1}{2} \\ \frac{\omega_c}{\pi} & n = \frac{M-1}{2} \end{cases}$
Filtro pasa banda	$hd(n) = \begin{cases} \frac{\sin(\omega_c + A\omega_c)(n - \frac{M-1}{2}) - \sin(\omega_c - A\omega_c)(n - \frac{M-1}{2})}{\pi (n - \frac{M-1}{2})} \\ \frac{\omega_c + A\omega_c}{\pi} - \frac{\omega_c - A\omega_c}{\pi} \end{cases}$
Filtro rechazo de banda	$hd(n) = \begin{cases} \frac{\sin \pi (n - \frac{M-1}{2}) + \sin(\omega_c - A\omega_c)(n - \frac{M-1}{2}) - \sin(\omega_c + A\omega_c)(n - \frac{M-1}{2})}{\pi (n - \frac{M-1}{2})} \\ 1 + \frac{\omega_c - A\omega_c}{\pi} - \frac{\omega_c + A\omega_c}{\pi} \end{cases}$

Donde; $w(n)$ es la función de ventana.

La respuesta en frecuencia, de un filtro diseñado está dado por la convolución periódica de la respuesta impulso en el tiempo y la respuesta de frecuencia de la función ventana $W(\omega)$ del filtro deseado, ecuación 2.24.

$$H(w) = H_d(w) * W(w)$$

Ecuación 2.24 Respuesta en frecuencia para un filtro diseñado

De acuerdo a la respuesta impulso $h_d(n)$ deseada se tienen varios tipos de filtros, entre los principales son:

- Pasa bajo, Pasa alto, Pasa banda, Elimina banda.

Los cálculos respectivos al tipo de filtro que se desee emplear, se definen matemáticamente como se indica en la Tabla 2.6.

Una vez determinado el tipo de filtro que se desee emplear, se debe aplicar el método de diseño de ventanas más adecuado, cuyas funciones más conocidos y ciertas características comparativas se detallan respectivamente en las Tablas 2.7 y 2.8.

Tabla 2.7 Tipos de funciones ventana [30]

Ventana	Secuencia temporal $0 \leq n \leq M-1$	$W(n) / W(w)$
Rectangular	$\begin{cases} 1 & 0 \leq n \leq M-1 \\ 0 & n \geq M \end{cases}$	
Bartlett/ Triangular	$\begin{cases} \frac{2n}{M-1} & 0 \leq n \leq M-1 \\ 2 - \frac{2n}{M-1} & \frac{M-1}{2} \leq n \leq M-1 \end{cases}$	
Blackman	$0.42 - 0.5 \cos \frac{2\pi n}{M-1} + 0.08 \cos \frac{4\pi n}{M-1}$	
Hamming	$0.54 - 0.46 \cos \frac{2\pi n}{M-1}$	
Hanning	$\frac{1}{2} \left(1 - \cos \frac{2\pi n}{M-1} \right)$	

Tabla 2.8 Características de las funciones ventana [9] [30]

Ventana	Anchura del lóbulo principal de la ventana	Anchura de la banda de transición del filtro diseñado Δw	Pico del lóbulo secundario de la ventana [dB]	Atenuación del filtro diseñado con esta ventana R_s [dB]
Rectangular	$4\pi/M$	$1.8\pi/M$	-13	-21
Bartlett	$8\pi/M$	$6.1\pi/M$	-25	-25
Hamming	$8\pi/M$	$6.6\pi/M$	-41	-53
Hanning	$8\pi/M$	$6.2\pi/M$	-31	-44
Blackman	$12\pi/M$	$11\pi/M$	-57	-74

En base a la Tabla 2.8, se puede observar que la anchura del lóbulo principal es inversa al valor del orden del filtro, la anchura de la banda de transición también depende del orden y la atenuación del lóbulo secundario depende del tipo de ventana. De acuerdo a estas características este tipo de ventanas no son versátiles para el diseño de filtros ya que no es posible controlar la anchura de la banda de transición y la atenuación independiente.

Sin embargo, existen otros tipos de ventanas como es la ventana de Kaiser, en donde si es posible controlar el parámetro de anchura de la banda de transición y la atenuación independiente. La definición matemática de la función ventana de Kaiser se describe en la ecuación 2.25, y está relacionada con las funciones de Bessel, y modificando ciertos valores de (β) se puede llegar a las formas similares de las ventanas mencionadas anteriormente.

$$w(n) = \frac{I_0 \left[\beta \sqrt{1 - \left(1 - \frac{2n}{M-1}\right)^2} \right]}{I_0[\beta]}, \quad 0 \leq n \leq M-1$$

$$I_0[\beta] = 1 + \sum_{k=1}^{\infty} \frac{(\beta/2)^k}{k!}$$

Ecuación 2.25 Función ventana de Kaiser ajustable

Donde:

- I_0 , es la función de Bessel de orden cero
- β , es el parámetro que depende de M
- n, determina el tamaño de la ventana

Método del muestreo en frecuencia

La técnica basada en el diseño por muestreo en frecuencia consiste en fijar una serie de puntos de la respuesta en frecuencia del sistema, y usando la transformada de *Fourier*

discreta inversa obtener los coeficientes del filtro. Con este método se tiene un filtro que tiene una respuesta en frecuencia que pasa por los puntos muestreados, sin embargo, no es posible llevar un control de resto de valores como los factores de rizado.

Este tipo de diseño tiene dos vías, la primera se denomina método de diseño ingenuo, donde las muestras de frecuencia son fijas tanto para valores de pasa banda y banda suprimida; cuyo efecto lleva a una cantidad inaceptable de ondulaciones cerca de los bordes de la banda, específicamente para filtros de órdenes superiores. En el método de orden óptimo las muestras en el ancho de banda de transición, se pueden variar con el fin de minimizar la atenuación.

Método aplicando la técnica de rizado óptimo

Los métodos mencionados anteriormente tienen la desventaja de que el rizado no se distribuye uniformemente en las bandas, por lo cual este método permite la redistribución de este error (error de aproximación entre la respuesta de frecuencia ideal y la real), permitiendo el diseño de filtros FIR de orden más pequeño en base a especificaciones dadas, no se pueden especificar los factores de rizado simultáneamente y no se puede especificar la banda de frecuencia precisa para el diseño.

2.2.3.3 Desarrollo práctico

En esta práctica se propone realizar un script generado por *Matlab* en el que se permita diseñar un filtro FIR por el método de ventanas, para analizar las respuestas del mismo, de acuerdo al marco teórico de la práctica.

- El script deberá permitir ingresar los parámetros absolutos y logarítmicos que describirán el funcionamiento del filtro FIR, y además posibilitar probar el script con otros datos ingresados verificando resultados y funcionamiento del mismo.
- Se propone establecer un filtro pasabanda usando el método de ventaneo Hamming.
- El script debe contar con los comandos necesarios para su correspondiente análisis gráfico de la respuesta impulso ideal, la respuesta impulso del filtro diseñado, la función de la ventana y la respuesta en frecuencia de la función ventana.

Este filtro satisface los requerimientos paramétricos que se describen en la Figura 2.58.

```

% ws1=input('ingrese el valor de ws1: '); % lower stopband edge
% wp1=input('ingrese el valor de wp1: '); % lower passband edge
% wp2=input('ingrese el valor de wp2: '); % upper passband edge
% ws2=input('ingrese el valor de ws2: '); % upper stopband edge
% Rp=input('ingrese el valor de Rp: '); % passband ripple
% As=input('ingrese el valor de As: '); % stopband attenuation

```

Figura 2.58 Ingreso de parámetros de diseño, script .m

Para ejecutar la práctica en base a los parámetros detallados en la Figura 2.58 se consideran los siguientes datos:

- $ws1=0.2\pi$, $ws2=0.8\pi$
- $wp1=0.4\pi$, $wp2=0.8\pi$
- $Rp=0.5$, $As=40$

El script de Matlab (Ver Anexo III) se programa de tal forma que se ingresen los datos que requiera probar el estudiante, es así que:

Primero, de acuerdo a los datos ingresados, el programa realizará el cálculo del ancho del lóbulo mínimo de transición que se requiere, con el fin de calcular el orden del filtro, tal como muestra la Tabla 2.8, en este caso con los datos ingresados resultó $M=33$.

Segundo, dado que se requiere diseñar un filtro pasabanda se procede a realizar los cálculos respectivos con los datos de la Tabla 2.6 del filtro pasabanda, para obtener la respuesta impulso que se desea diseñar, Figura 2.59.

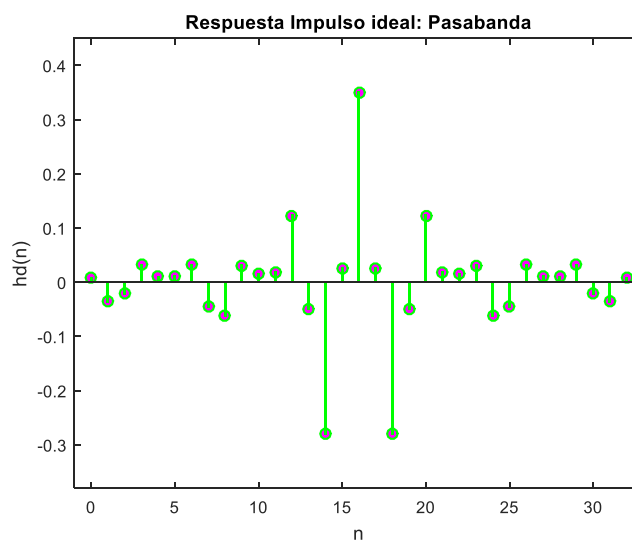


Figura 2.59 Respuesta impulso que se desea diseñar

Finalmente se aplica la ecuación 2.23 para hallar la respuesta impulso del filtro diseñado, usando la función ventana escogida, en este caso Hamming; y aplicando la ecuación 2.24 para hallar la respuesta en frecuencia, como se ilustra en las gráficas de las Figuras 2.60, 2.61 y 2.62. Los resultados se analizarán en el siguiente capítulo.

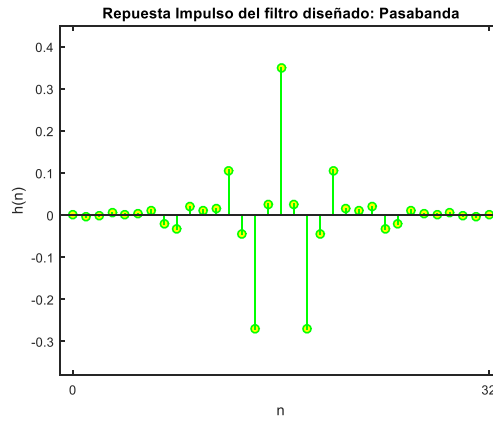


Figura 2.60 Respuesta impulso del filtro diseñado

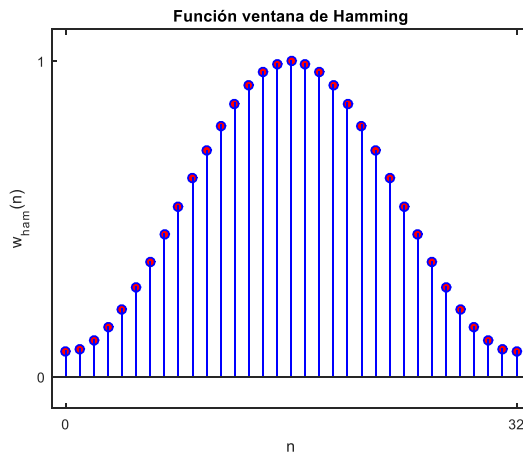


Figura 2.61 Función ventana de Hamming, $M=33$

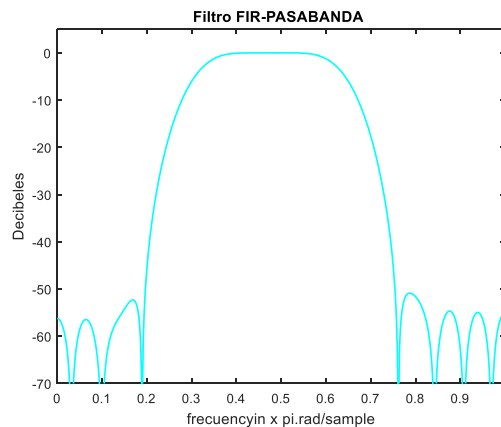


Figura 2.62 Respuesta en frecuencia del filtro diseñado

2.2.4 Práctica 04: Diseño y simulación de un filtro IIR

2.2.4.1 Objetivos

- Conocer las técnicas y métodos de diseño de un filtro IIR.
- Entender qué requerimientos y parámetros son necesarios definir, para el diseño de un filtro digital IIR.
- Utilizar la herramienta Matlab para analizar resultados sobre la elaboración de un archivo script para diseñar y simular un filtro digital tipo IIR, aplicando un método conveniente.
- Conocer las ventajas y campos de aplicación de este tipo de filtros en comunicaciones digitales.

2.2.4.2 Marco teórico

El filtro IIR (*Infinite Impulse Response*), es aquel que tiene una respuesta impulso de duración infinita, además son filtros de tipo recursivo, es decir la respuesta a una señal impulso de entrada tiene un número infinito de términos de ceros y polos, razón por la cual no garantiza estabilidad. Las características principales de los filtros IIR son:

- Requieren menor tiempo de procesamiento con respecto a los filtros FIR ya que necesitan menos coeficientes implicando el uso de menor memoria.
- La respuesta al impulso infinita de los filtros IIR cuenta con ceros y polos, lo cual implica una estabilidad no garantizada.
- Sus requerimientos en cuanto a recursos son mayores respecto a los filtros IIR y dada la mayor cantidad de operaciones estos filtros son más sensibles a errores.
- Pueden ser diseñados a partir de filtros analógicos, como: *Butterword*, *Chebyshev* o Elíptico.
- No pueden diseñarse para tener fase lineal.

Estructura del filtro IIR [9]

La función sistema de un filtro IIR está dada por la ecuación de transferencia, ecuación 2.26.

$$H(z) = \frac{B(z)}{A(z)} = \frac{\sum_{n=0}^M b_n z^{-n}}{\sum_{n=0}^N a_n z^{-n}}$$

Ecuación 2.26 Descripción de un filtro IIR

Donde, los valores de a_n, b_n son los coeficientes del filtro, $a_0 = 1$ y el orden es N.

La representación de la ecuación diferencial del filtro IIR tiene la forma, ecuación 2.27:

$$y(n) = \sum_{m=0}^M b_m x(n-m) - \sum_{m=1}^N a_m y(n-m)$$

Ecuación 2.27 Descripción de un filtro IIR

Los filtros IIR pueden ser diseñados considerando tres tipos de estructuras como: la forma directa, la forma cascada y la forma paralela. La forma directa y la forma cascada son implementadas con el mismo criterio de los filtros FIR, y la forma paralela es similar a la forma cascada pero después de la factorización se debe usar la expansión de fracciones parciales para representar la ecuación de transferencia del sistema como una suma directa de secciones pequeñas de segundo orden y cada una de ellas nuevamente es implementada de la forma directa resultando a la final una red paralela.

Diseño de un filtro IIR

Dado que se parte de filtros analógicos, se discute ciertas generalidades de la escala relativa lineal. Donde se considera la respuesta en frecuencia de un filtro analógico como $H_a(j\Omega)$ tipo pasa bajo sobre las especificaciones de la respuesta en magnitud que se indica en la Figura 2.63.

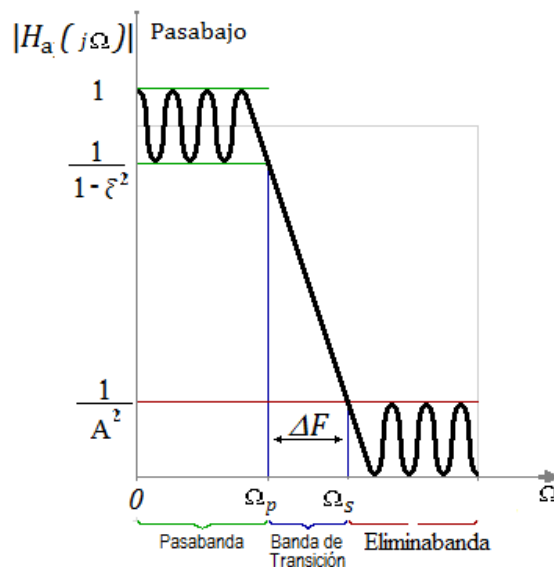


Figura 2.63 Respuesta en frecuencia del filtro diseñado

Donde, ϵ es el parámetro de rizado en el intervalo pasabanda, Ω_p corresponde a la frecuencia de corte pasabanda, A es el parámetro de rizado en el intervalo de la banda de

rechazo (elimina banda) y Ω_s es la frecuencia de rechazo. En la escala de [dB], ε y A están relacionados con R_p y A_s .

Adicionalmente, si se requiere analizar los polos y ceros de la respuesta en frecuencia de un filtro analógico se considera el plano s , matemáticamente la transformada de Laplace, y para el caso de los filtros digitales se considera el plano z .

Las técnicas que se emplean para implementar un filtro IIR, se basan en diseños de filtros analógicos los cuales se detallan en la Tabla 2.9 ó también se realizan a partir de prototipos de filtros pasa bajo analógicos, donde se debe determinar la función de transferencia del filtro analógico que satisfaga los requerimientos de diseño para luego transformar a la función de transferencia del filtro digital.

Tabla 2.9 Filtros analógicos [9]

Filtro	Respuesta magnitud cuadrada
Butterworth	$ H_a(\Omega) ^2 = \frac{1}{1 + \left(\frac{\Omega}{\Omega_c}\right)^{2N}}$
Chevyshev-I	$ H_a(\Omega) ^2 = \frac{1}{1 + \varepsilon^2 T_N^2\left(\frac{\Omega}{\Omega_c}\right)}$
Chevyshev-II	$ H_a(\Omega) ^2 = \frac{1}{1 + \varepsilon^2 T_N^2\left(\frac{\Omega}{\Omega_c}\right)^{-1}}$
Elíptico	$ H_a(\Omega) ^2 = \frac{1}{1 + \varepsilon^2 U_N^2\left(\frac{\Omega}{\Omega_c}\right)}$

Para la transformación del filtro analógico al filtro digital deseado, se puede aplicar el método basado en el mapeo entre el plano s (Transformada Laplace) y el plano z (Transformada Z), con el fin de mantener las propiedades fundamentales de la respuesta en frecuencia.

Las transformaciones más comunes de filtros analógicos a digitales son:

- Método del impulso invariante
- Método de la transformación bilineal

Método del impulso invariante

En este método se debe desarrollar una función de transferencia IIR, cuya respuesta al impulso sea igual a la versión muestreada uniformemente de la respuesta al impulso de la

función de transferencia del prototipo de filtro analógico. La ecuación 2.28 muestra la función mapeo entre los dominios s y z, ver Figura 2.64.

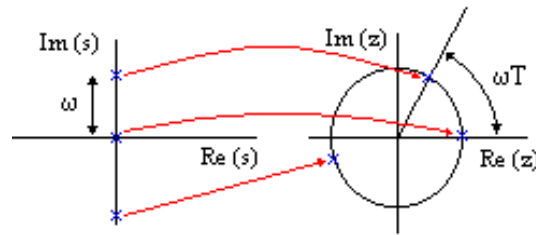


Figura 2.64 Método del impulso invariante

$$z = e^{sT}$$

Ecuación 2.28 Frecuencias complejas correspondientes, método impulso invariante

Donde T es el período de muestreo, para este caso se considera T=1.

El proceso de diseño usando este método invariante consiste en que, dado un filtro pasa bajo de especificaciones w_p, w_s, R_p y A_s , primero se debe calcular las frecuencias de la banda usando, $\Omega_p = w_p/T$ y $\Omega_s = w_s/T$. Luego se debe diseñar un filtro analógico usando uno de los prototipos de los filtros para satisfacer las especificaciones de Ω_p, Ω_s, R_p y R_s . Por último se deberá mapear los polos en el plano z.

Los parámetros w_p, w_s, R_p y A_s , son las especificaciones de: frecuencia de la banda de paso, frecuencia de la banda de parada, factor de rizado de la banda de paso y el factor de rizado de la banda de parada correspondientemente a un filtro digital, y las especificaciones Ω_p, Ω_s, R_p y R_s son las correspondientes para un filtro analógico.

Método de la transformada bilineal

En este método se trata de obtener una función de transferencia IIR en la que la respuesta en frecuencia sea una aproximación a la respuesta en frecuencia de la función de transferencia del prototipo analógico, la función de mapeo entre los dominios s y z, está dada por la ecuación 2.29.

$$s = \frac{2}{T} \left(\frac{1 - z^{-1}}{1 + z^{-1}} \right) \rightarrow z = \frac{1 + sT/2}{1 - sT/2}$$

Ecuación 2.29 Descripción de un filtro IIR

Las frecuencias reales están relacionadas con: $w = 2 \tan^{-1}(\Omega T/2)$ o $\Omega = (2/T) \tan(w/2)$, el mapeado del plano complejo en la transformada bilineal se ilustra en la Figura 2.65, en donde la transformación se hace uno a uno.

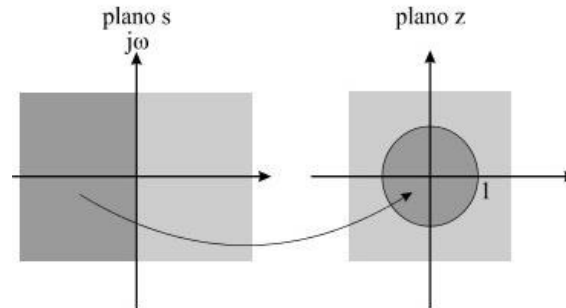


Figura 2.65 Método transformada bilineal

El proceso de diseño usando este método consiste en que, dado un filtro pasa bajo digital de especificaciones w_p, w_s, R_p y A_s , primero se debe calcular las frecuencias de la banda usando, $\Omega_p = (2/T) \tan(w_p/2)$ y $\Omega_s = (2/T) \tan(w_s/2)$. Luego se debe diseñar un filtro analógico usando uno de los prototipos de los filtros para satisfacer las especificaciones de Ω_p, Ω_s, R_p y R_s . Por último se deberá obtener $H(z)$ como una función racional en z^{-1} .

2.2.4.3 Desarrollo práctico

En esta práctica se propone realizar un script generado por *Matlab* en el que se permita diseñar un filtro IIR digital, utilizando uno de los prototipos de filtros analógicos.

- El script deberá permitir ingresar los parámetros absolutos y logarítmicos que describirán el funcionamiento del filtro IIR, y además posibilitar probar el script con otros datos ingresados verificando resultados y funcionamiento del mismo.
- Se propone diseñar un filtro pasa bajo usando un filtro prototipo elíptico y usando el método bilineal.
- El script debe contar con los comandos necesarios para su correspondiente análisis gráfico de la respectiva respuesta impulso del filtro diseñado, y la respuesta en frecuencia.

Este filtro satisface los requerimientos paramétricos que se describen en la Figura 2.66.

```

% ws=input('ingrese el valor de ws1: '); % lower stopband edge
% wp=input('ingrese el valor de wp1: '); % lower passband edge
% Rp=input('ingrese el valor de Rp: '); % passband ripple
% As=input('ingrese el valor de As: '); % stopband attenuation

```

Figura 2.66 Ingreso de parámetros de diseño, script .m

Para ejecutar la práctica en base a los parámetros detallados en la Figura 2.66 se consideran los siguientes datos:

- $w_s=0.4\pi$, $w_p=0.2\pi$
- $R_p=1$, $A_s=70$

El script de Matlab (Ver Anexo IV) se programa de tal forma que se ingresen los datos que requiera probar el estudiante, es así que:

Primero, de acuerdo a los datos ingresados, el programa realizará el cálculo de las frecuencias de la banda de paso y de la banda de rechazo, usando $\Omega_p = (2/T)\tan(w_p/2)$ y $\Omega_s = (2/T)\tan(w_s/2)$, que una vez obtenidos estos parámetros se procederá a determinar el diseño del filtro deseado usando el respectivo comando de Matlab para un prototipo de filtro analógico elíptico, cuyo orden resultante con los datos ingresados resulta ser 5.

Aplicando el método de la transformada bilineal, se obtiene la función de transferencia del prototipo analógico, la visualización de la conversión del filtro analógico digital se presenta en la Figura 2.67.

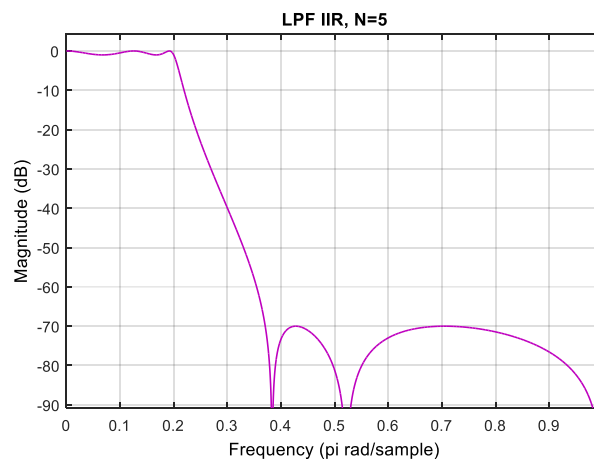


Figura 2.67 Respuesta en frecuencia del filtro IIR diseñado, pasa bajo

Se puede determinar que la respuesta impulso del filtro, tiene un número infinito de términos no nulos, Figura 2.68.

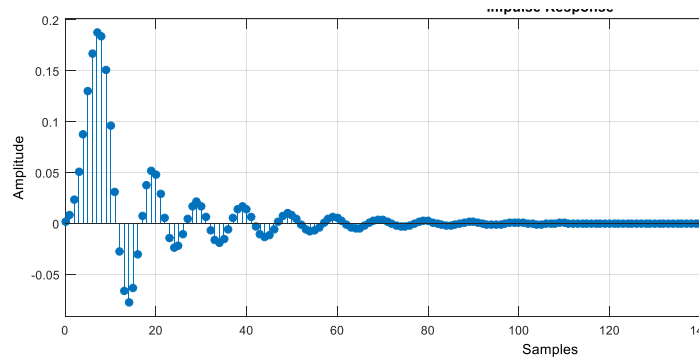


Figura 2.68 Respuesta impulso del filtro diseñado

Los resultados se analizarán en el siguiente capítulo.

2.3 Sistemas de modulación y demodulación digital

Un sistema de comunicaciones está formado básicamente por tres elementos principales transmisor, receptor y medio de transmisión. En telecomunicaciones la información se transmite por un sistema de comunicaciones a manera de símbolos que pueden ser analógicos o digitales, pero no siempre la información es adecuada para transmitir de manera original, por lo cual se debe convertir a una forma más adecuada antes de transmitir.

Las comunicaciones digitales abarcan varias técnicas de comunicación como es la transmisión digital y radio digital. La transmisión digital hace referencia a la transmisión de pulsos digitales, y el radio digital hace referencia a la transmisión de portadoras analógicas moduladas en forma digital en un sistema de comunicación.

La modulación es la técnica por la cual ciertas características de una onda denominada portadora se modifican en base a otra onda denominada moduladora o modulante, para que pueda ser transmitida. En otras palabras, la modulación es la técnica empleada para modificar una señal a fin de que esta pueda ser transmitida por un canal de comunicaciones de la mejor manera, y en su recepción esta pueda ser recuperada.

La modulación se basa en el control de algunos parámetros como la amplitud, frecuencia y fase de una señal portadora. Es así que la modulación digital es un proceso por el cual se transforman los símbolos digitales en forma de ondas adecuadas para la transmisión sobre un canal de comunicaciones.

En esencia en esta sección se abordarán prácticas referentes a técnicas de modulación y demodulación digital, como ASK, FSK, PSK y QAM. Las cuales se dividirán en temas de prácticas, donde se simulará para después en el capítulo de análisis evaluar los resultados.

2.3.1 Práctica 05: Modulación y demodulación ASK

2.3.1.1 Objetivos

- Comprender la forma de representar los datos digitales de una secuencia de entrada usando la técnica de modulación ASK.
- Analizar resultados de la simulación ASK mediante las herramientas OrCAD PSpice y *Simulink* en sus escenarios respectivos.
- Conocer las ventajas y desventajas que brinda esta técnica de modulación digital en aplicaciones de ingeniería.
- Comparar el espectro de frecuencia de la secuencia de datos y la señal modulada.
- Analizar la modulación OOK matemáticamente.

2.3.1.2 Marco teórico [13] [14] [19] [22]

La modulación implica técnicas que se usan para transmitir información sobre una onda sinusoidal portadora, permitiendo aprovechar el canal de comunicación y de esta manera posibilitando la transmisión de mayor información en forma simultánea y consiguiendo mayores alcances en la transmisión.

En esta práctica se tratará la técnica de modulación digital por desplazamiento de amplitud ASK cuyas siglas en inglés significa *Amplitude Shift Keying*, la cual representa los datos digitales como variaciones de amplitud de la onda portadora manteniendo la frecuencia y la fase constante.

Las características principales de este tipo de modulación son:

- Requiere circuitos poco complejos para su implementación.
- Puede ser binaria en la que la amplitud pico de la portadora modulada toma dos posibles valores, los cuales se asocian al dato modulado.
- La modulación 2ASK cuenta con un caso particular denominado modulación OOK, cuyo funcionamiento se comporta como un interruptor On/Off, significando presencia o ausencia de la señal portadora.
- Puede ser de múltiples estados (M-ASK), donde la amplitud pico de la portadora modulada toma múltiples valores, cada uno está asignado a un símbolo de $\log_2 M$ bits, esta técnica de modulación multiestado no es muy usada ya que se puede presentar en el receptor dificultad para la discriminación de valores de amplitud pico de la portadora modulada que se encuentren muy cercanos.

- El uso de portadoras de amplitud modulada para transmitir información digital es un tipo de radio digital por lo que implica ser de baja calidad y por ende de bajo costo.
- ASK es una técnica ineficaz de modulación por ser de bajo rendimiento ya que el ruido y la distorsión afectan a la señal transmitida.

Ecuación general ASK

La ecuación 2.30, describe la modulación ASK considerando una portadora de amplitud pico unitaria y fase cero grados: [13]

$$ASK(t) = [1 + k \cdot b_n(t)] \cos(w_c t)$$

Ecuación 2.30 Expresión matemática de la modulación ASK

Donde:

- $b_n \rightarrow$ secuencia de datos digitales
- $k \rightarrow$ índice de modulación, $(0 \leq k \leq 1)$
- $w_c = 2 * \pi * f_c \rightarrow$ frecuencia de la portadora en radianes por segundo

Esquema del modulador y demodulador 2ASK

De acuerdo al modelo matemático, el esquema de modulación se ilustra en la Figura 2.69.

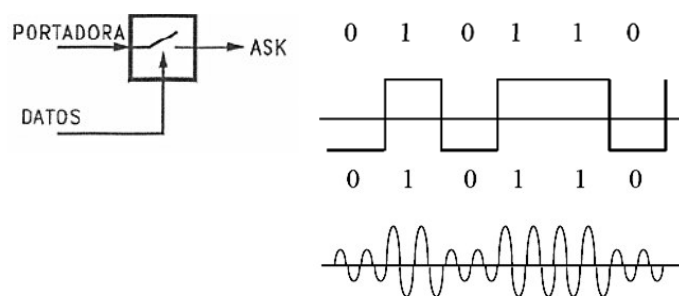


Figura 2.69 Esquema de modulación 2ASK [22]

En el caso del esquema de demodulación se puede realizar aplicando dos formas las cuales se describen como coherente y no coherente. El demodulador coherente es más eficiente, ya que su complejidad cuenta con la ventaja frente a efectos del ruido, su funcionamiento se basa en la mezcla de la señal de un oscilador local sincronizado con la portadora Figura 2.70 (b). Mientras que en el caso no coherente su funcionamiento es el de un detector de envolvente cuyo esquema se ilustra en la Figura 2.70 (a). Los dos casos cuentan con un filtro pasa bajo que cumple la función de eliminar las componentes

residuales de la portadora; y, además como se indica en el esquema se agrega un bloque umbral, el cual se encargará de la conformación de la señal.

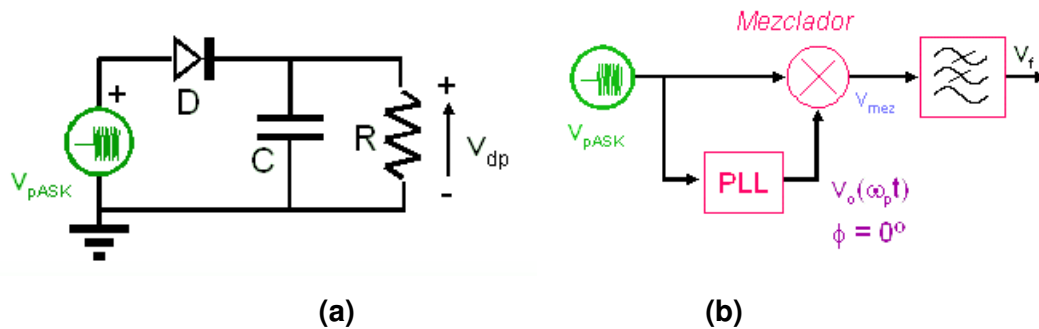


Figura 2.70 Esquema demodulador ASK de la forma (a) no coherente (b) coherente [30]

2.3.1.3 Desarrollo práctico

Esta práctica propone esquematizar un modulador ASK usando la herramienta OrCAD PSpice para analizar los resultados a partir de la variación del índice de modulación de la señal de datos hasta obtener una modulación al 100%, además se mostrarán las gráficas tanto en el dominio del tiempo como de la frecuencia para análisis de resultados.

Desarrollar el análisis matemático, de los resultados del escenario en el cual el índice de modulación alcanza su máximo valor.

Como segundo escenario se propone esquematizar la demodulación ASK a partir de una señal modulada, en el que se usarán bloques funcionales de Simulink. Donde será posible analizar los pasos de demodulación de acuerdo a la teoría establecida en el punto anterior.

Esquema práctico de modulación

Se referencia el esquema de la Figura 2.69, para la construcción del esquema de modulación 2ASK, el cual contiene los elementos necesarios para ver la respectiva simulación y cuyo funcionamiento se basará en la ecuación 2.29, ver Figura 2.71.

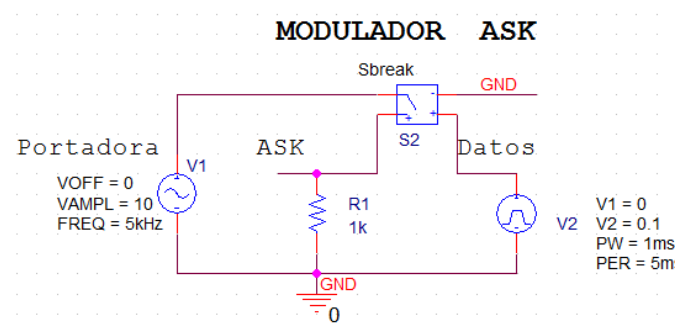


Figura 2.71 Esquema modulador ASK

Los elementos que comprenden este esquema se describen a continuación:

- **Sbreak:** este elemento cumple la función de interruptor, el mismo que se activa con la señal portadora ingresada a través de *VSIN*, donde la frecuencia de señal portadora se considera de 5 [Khz] y una amplitud de 10 [V].
- Los datos de entrada se ingresarán utilizando el elemento *VPULSE*, el cual cuenta con el parámetro V2 mismo que estará relacionado con la amplitud de la modulante y que a su vez al ser modificado en el intervalo de 0 a 1 y modularse con la señal portadora, a la salida del esquema modulador realiza una variación en la amplitud dependiendo de los datos de la modulante.

En base al esquema de modulación de la Figura 2.71, se presenta los datos de entrada en Figura 2.72 (a), la señal portadora en la Figura 2.72 (b) y el espectro de la portadora en la Figura 2.72 (c).

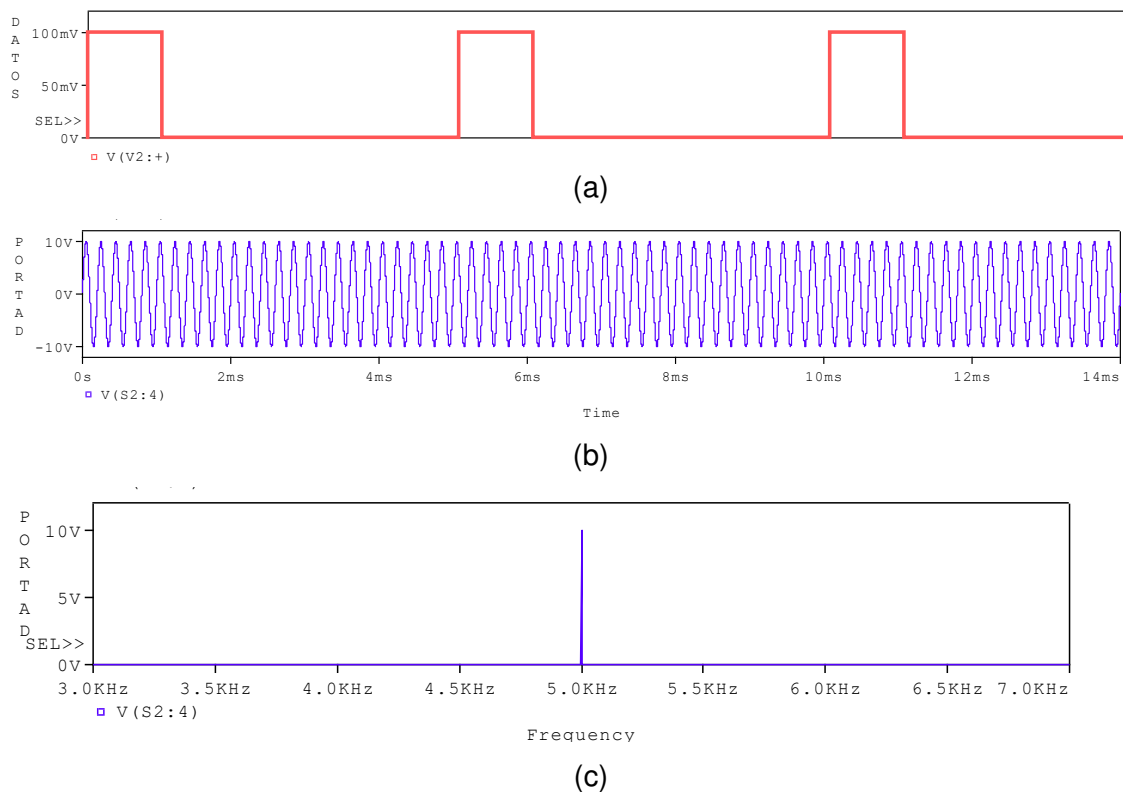


Figura 2.72 (a) Señal de datos $T_b=1$ ms, (b) Señal portadora, (c) Espectro de la portadora

La señal de datos tiene una duración de tiempo de bit (T_b) de 1 [ms] y para realizar esta simulación se utiliza señales periódicas, pero también es posible ingresar cualquier secuencia de bits utilizando el elemento STM1 cuyo detalle y modo de aplicación se describió en la Guía número 3 de este trabajo de titulación.

En esta práctica se simularán tres escenarios de modulación, en donde se variará el parámetro V2 para tres valores distintos a fin de establecer el respectivo análisis de resultados. Es así que para fines prácticos y probar el esquema ASK, con la misma señal portadora de la Figura 2.72 (b) se considerarán distintos valores para el parámetro V2, (0.1, 0.3 y 1).

Con **V2=0.1**, los resultados de esta variación se presenta en la Figura 2.73 en función del tiempo y en la Figura 2.74 en función de la frecuencia.

T_b=1ms

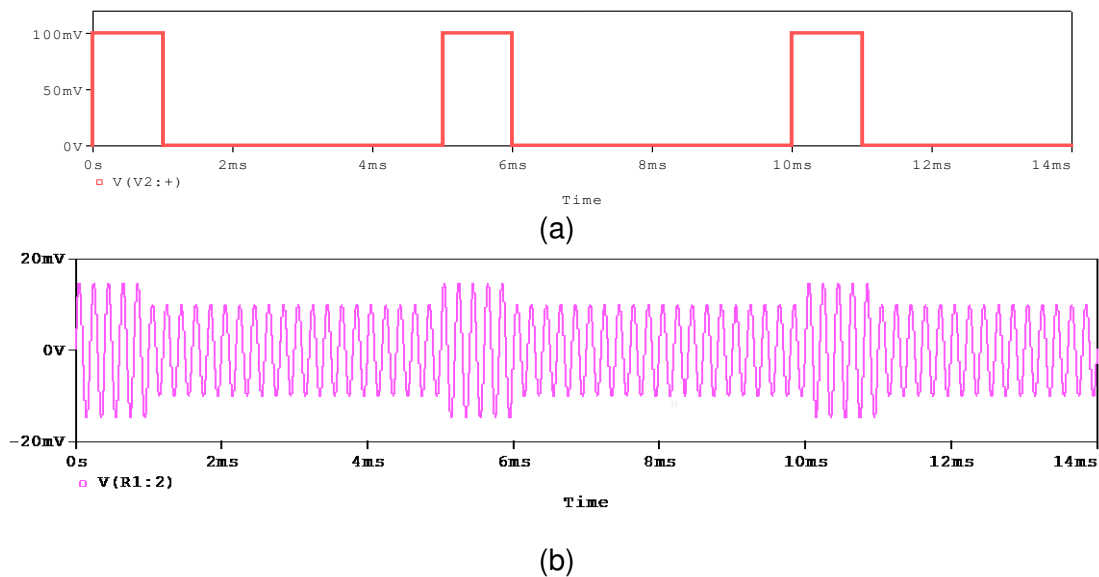


Figura 2.73 Señales (a) Señal de datos, (b) Señal modulada ASK

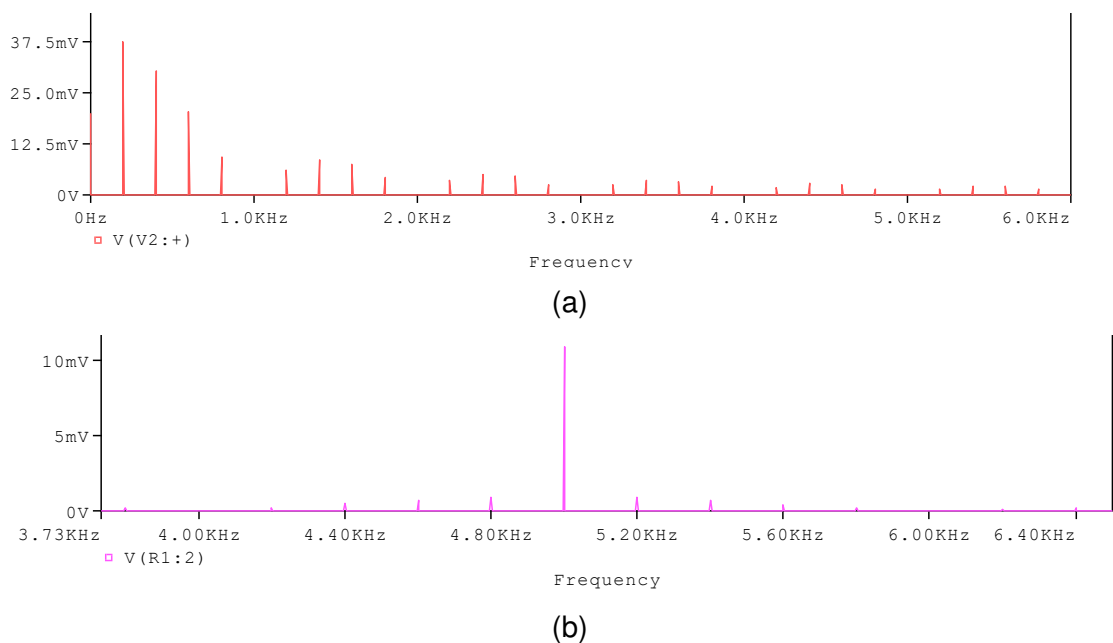


Figura 2.74 Espectro (a) Señal de datos, (b) Señal modulada ASK

Con $V_2=0.3$, los resultados de esta variación se presenta en la Figura 2.75 en función del tiempo y en la Figura 2.76 en función de la frecuencia.

$T_b=1\text{ms}$

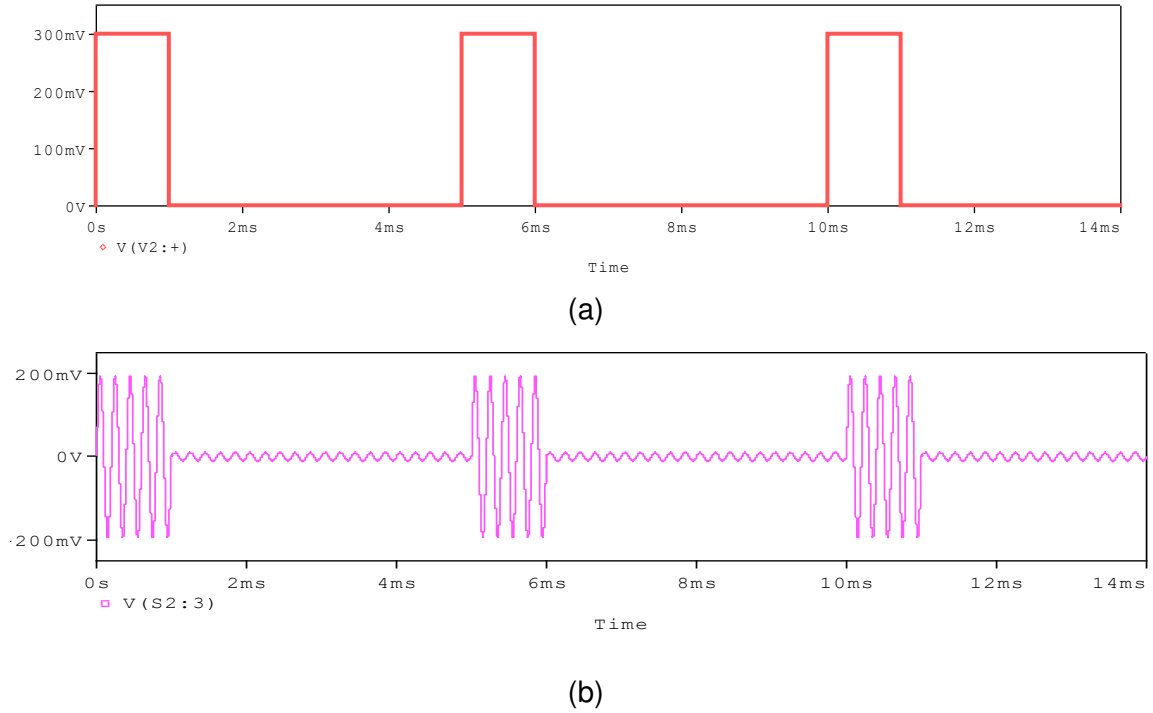


Figura 2.75 Señales (a) Señal de datos, (b) Señal modulada ASK

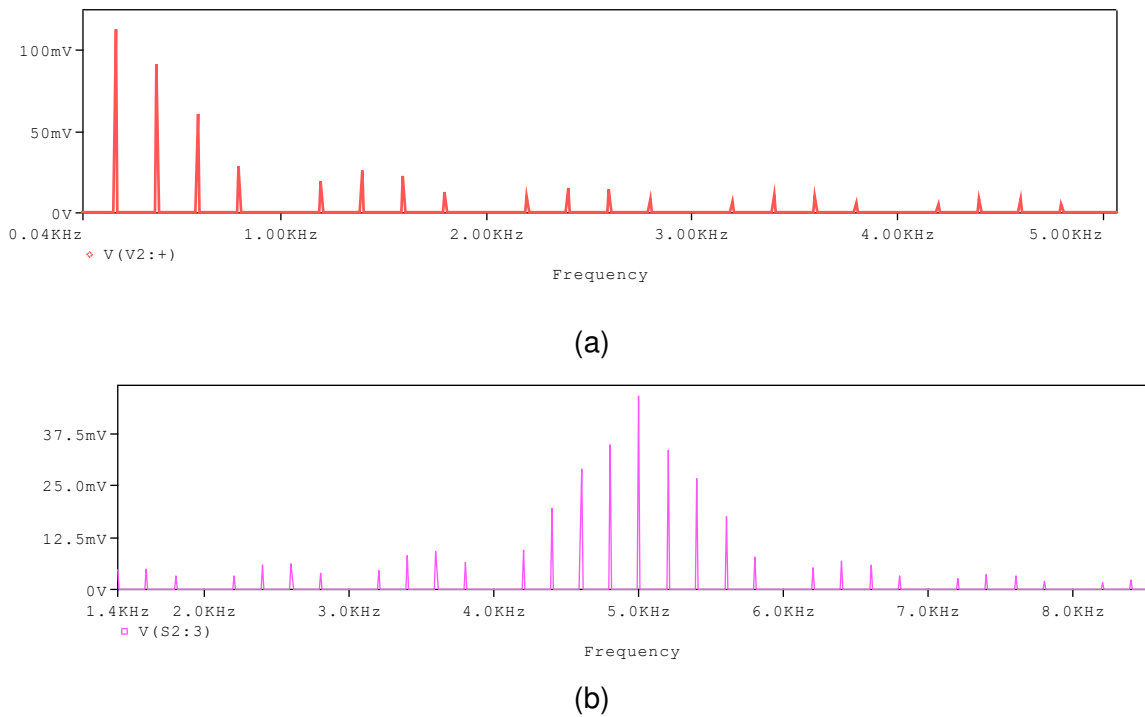
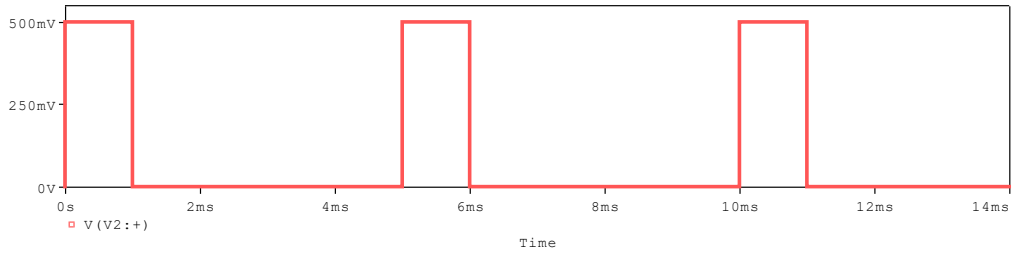


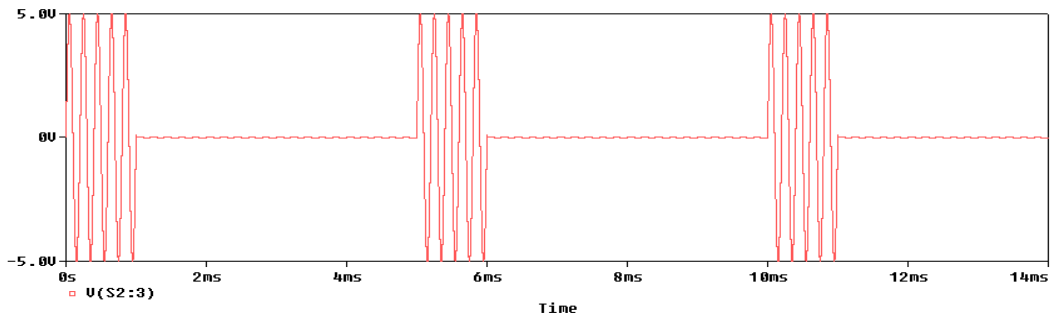
Figura 2.76 Espectro (a) Señal de datos, (b) Señal modulada ASK

Con $V_2=0.5$, los resultados de esta variación se presenta en la Figura 2.77 en función del tiempo y en la Figura 2.78 en función de la frecuencia.

$T_b=1\text{ms}$

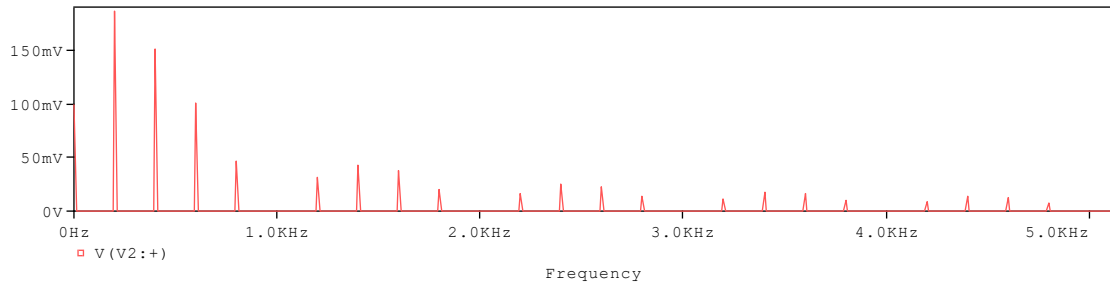


(a)

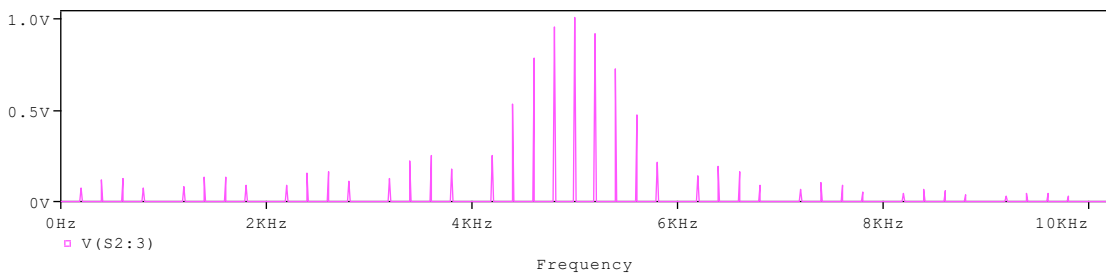


(b)

Figura 2.77 Señales (a) Señal de datos, (b) Señal modulada ASK



(a)



(b)

Figura 2.78 Espectro (a) Señal de datos, (b) Señal modulada ASK

Con $V_2=1$, los resultados de esta variación se presenta en la Figura 2.79 en función del tiempo y en la Figura 2.80 en función de la frecuencia.

$T_b=1\text{ms}$

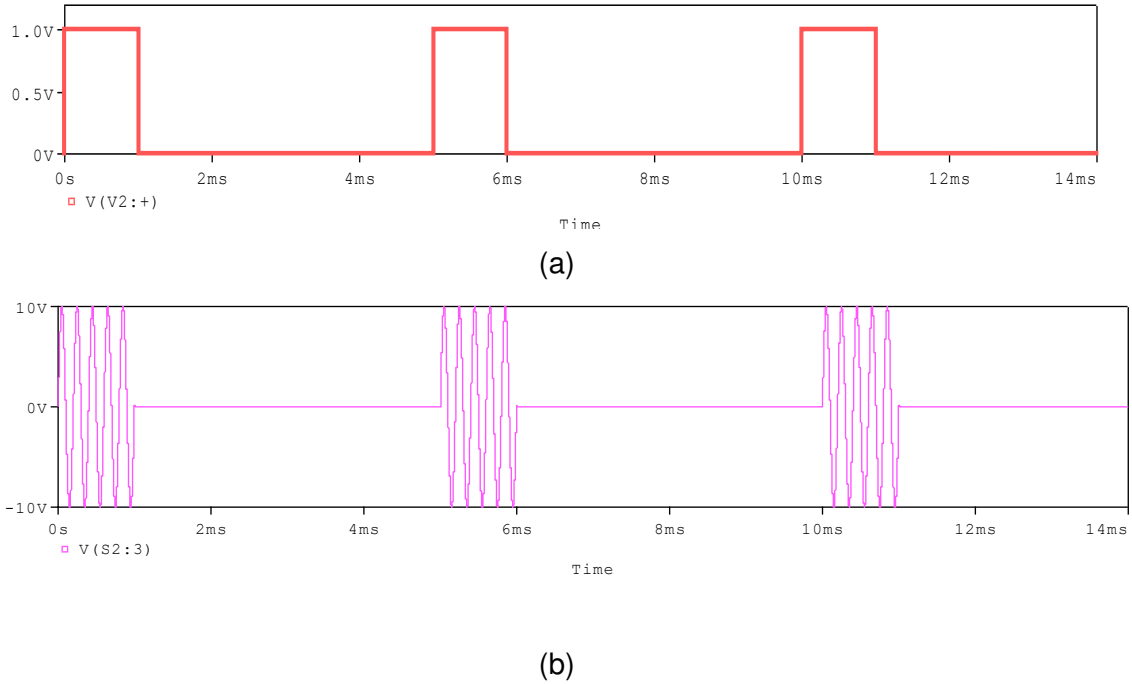


Figura 2.79 Señales (a) Señal de datos, (b) Señal modulada ASK

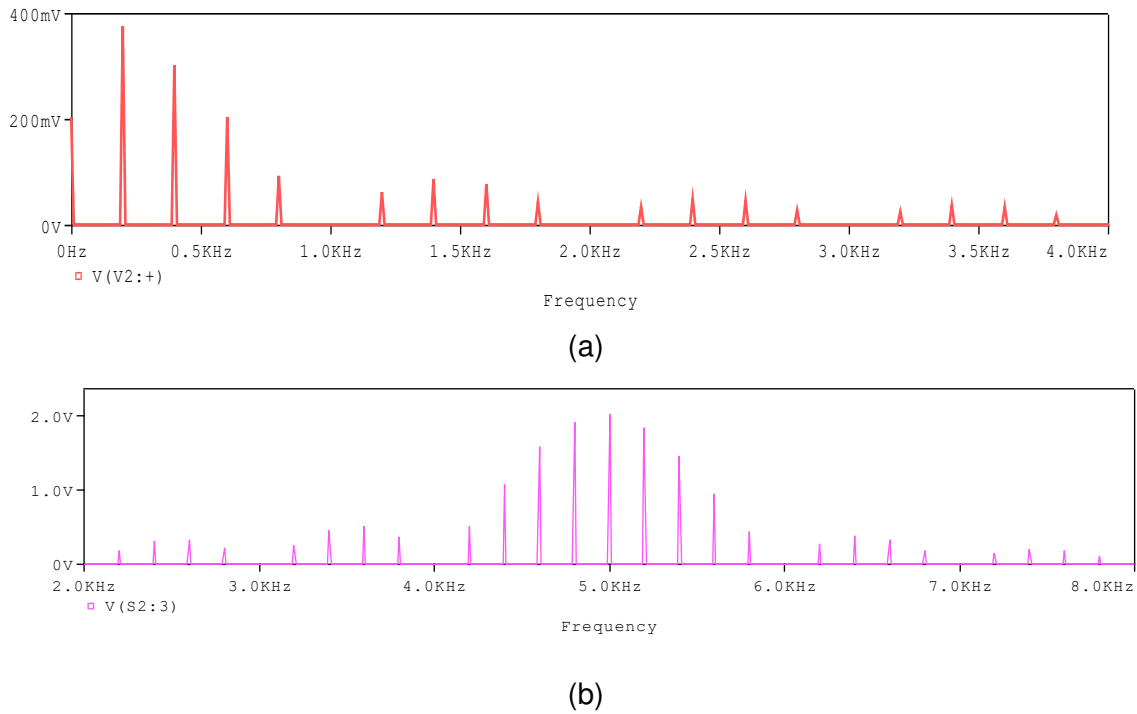


Figura 2.80 Espectro (a) Señal de datos, (b) Señal modulada ASK

Esquema práctico de demodulación

El esquema de demodulación se presenta en la Figura 2.81. Este se esquematiza con los bloques necesarios para simular un demodulador de una señal modulada OOK. El modulador OOK tiene como entrada de datos una señal binaria cualquiera (en la práctica se usa el bloque de Simulink Generador de datos Aleatorios). Además, cuenta con un sistema que permite observar el espectro de la señal modulada con los respectivos bloques funcionales.

La señal ASK que ingresa al demodulador realiza el siguiente proceso para obtener los datos binarios, primero la señal modulada ingresa a un bloque interruptor el cual se desempeña como un rectificador de media onda. Después se colocará un filtro pasa bajo en base a la función de transferencia, en donde este permitirá a través de una frecuencia de corte obtener un aproximado de la función de entrada (envolvente de la señal). Para mejorar esta reconstrucción de la señal se agregará un bloque interruptor que se comportará como un comparador para determinar los datos de entrada.

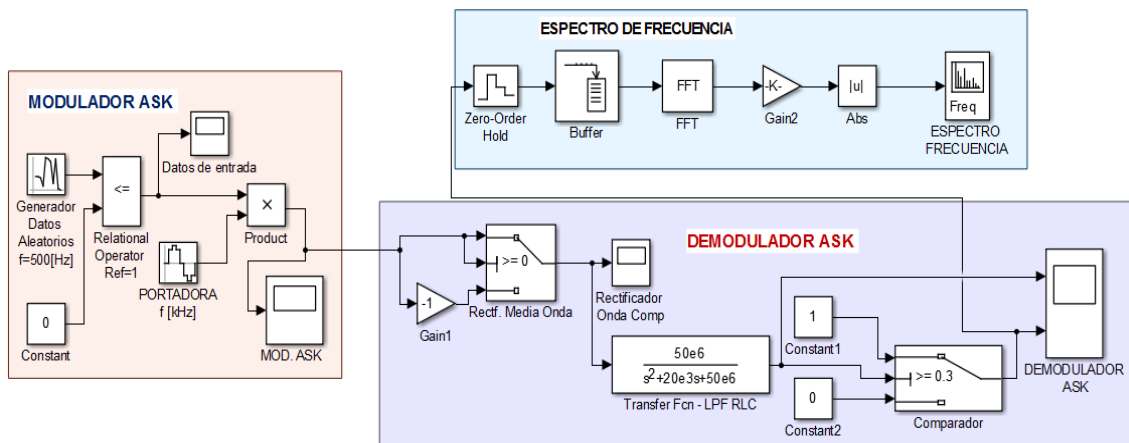


Figura 2.81 Diagrama de Modulación/Demodulación ASK

El bloque modulador contiene una señal portadora de 2 [KHz] y una amplitud pico de 5 [V]. La entrada de datos del modulador se origina a partir de un generador de datos aleatorios, en base a estos parámetros se tiene una modulación OOK, la cual será la base para empezar el proceso de demodulación utilizando el bloque demodulador.

De acuerdo al marco teórico del demodulador ASK no coherente se obtienen los resultados que se presentan en la Figura 2.82.

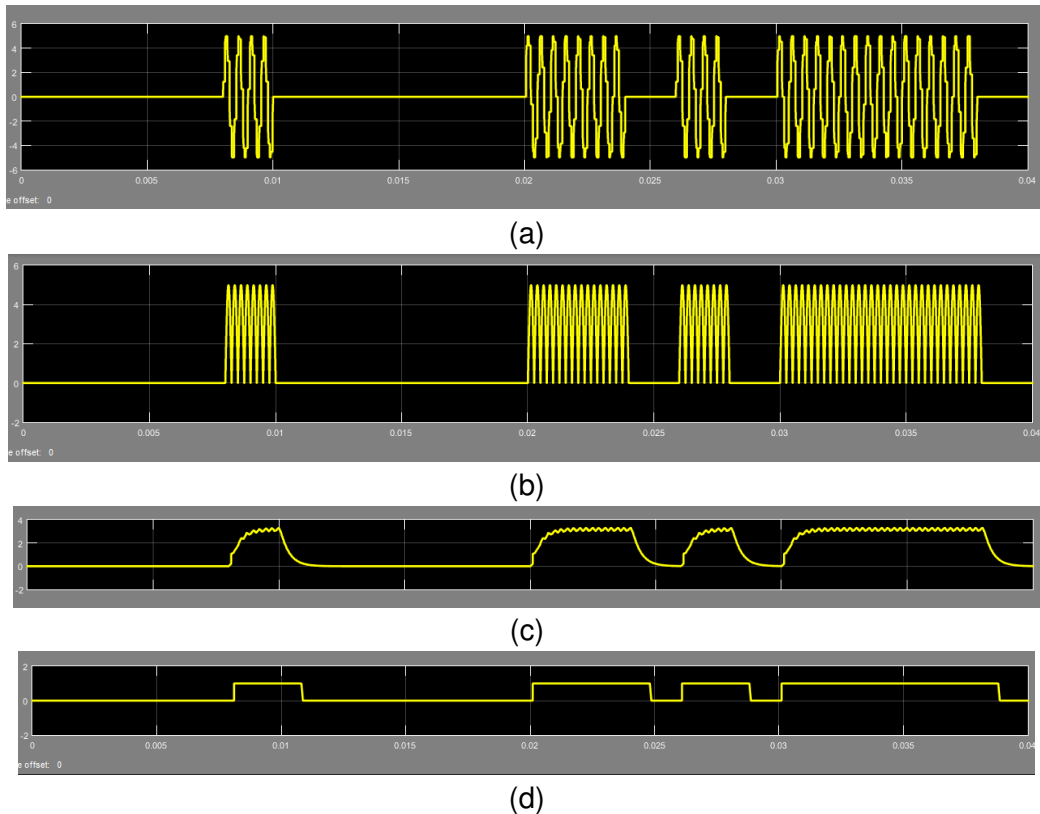


Figura 2.82 Pasos de demodulación no coherente (a) Modulación OOK, (b) Señal rectificada, (c) Detector de envolvente, (d) Señal demodulada (datos)

Los resultados se analizarán en el siguiente capítulo.

2.3.2 Práctica 06: Modulación y demodulación FSK

2.3.2.1 Objetivo

- Comprender el funcionamiento de la técnica de modulación / demodulación FSK.
- Diseñar y esquematizar un modulador tipo 2-FSK y un demodulador FSK coherente, con el fin de establecer escenarios de análisis.
- Analizar los resultados de la simulación aplicando el uso de la herramienta OrCAD PSpice.
- Conocer las ventajas y desventajas que brinda esta técnica de modulación digital.

2.3.2.2 Marco teórico [13] [14] [19] [22]

En esta práctica se analizará la modulación 2-FSK ya que esta modulación es una de las más utilizadas pues la modulación de múltiples estados FSK no es muy usada debido a su gran ancho de banda. Además, se considerará un demodulador coherente para el respectivo análisis en el siguiente capítulo de este trabajo de titulación.

En general la modulación por desplazamiento de frecuencia FSK (*Frequency Shift Keying*) es una técnica de modulación de tipo angular en donde la frecuencia de la portadora varía de acuerdo a la información digital, ver Figura 2.83. La representación matemática se muestra en la ecuación 2.31.

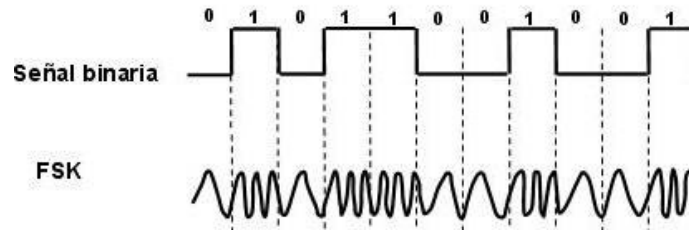


Figura 2.83 Esquema de modulación 2-FSK de fase continua [19]

$$FSK(t) = V_c \cdot \cos \left\{ w_c + \frac{[b_n(t) \cdot \Delta w]}{2} \right\} t$$

Ecuación 2.31 Expresión matemática de la modulación FSK

Donde:

- $b_n(t) \rightarrow$ señal digital NRZ en banda base (datos dos estados ± 1 , para 2FSK)
- $\Delta w \rightarrow$ desviación de frecuencia angular pico a pico
- $w_c = 2 * \pi * f_c \rightarrow$ frecuencia de la portadora sin modular [rad/s]

Para 2FSK la señal moduladora es una onda binaria en el que al dato 1L corresponde la función $b_n(t) = 1$ y al 0L le corresponde una función $b_n(t) = -1$.

Los valores instantáneos de la señal portadora se expresan de la siguiente manera en las ecuaciones 2.32 y 2.33: [13]

$$FSK(t) = V_c \cos \left(w_c + \frac{\Delta w}{2} \right) t, \quad b_n = +1$$

Ecuación 2.32 Expresión matemática de la modulación FSK para 1L

$$FSK(t) = V_c \cos \left[\left(w_c - \frac{\Delta w}{2} \right) t \right], \quad b_n = -1$$

Ecuación 2.33 Expresión matemática de la modulación FSK para 0L

Donde:

- $\Delta w/2 \rightarrow$ desviación de frecuencia pico

Las características principales de la modulación 2-FSK son las siguientes:

- La frecuencia de la portadora conmuta entre dos valores.

- Se emplea en enlaces asincrónicos.
- Resulta un sistema ideal para trabajar a bajas velocidades.
- Requiere circuitos de media complejidad.
- Tiene dos formas de modulación, la modulación de fase discontinua o no coherente, y la modulación coherente o de fase continua.
- Inmune al ruido aditivo del canal, gracias a la modulación por cambios de frecuencia.
- Tiene aplicaciones en radiodifusión pública.

Modulador FSK

La modulación FSK tiene como objetivo asignar una frecuencia diferente a cada estado a transmitir (se considera dos estados). Para lo cual, se tiene dos tipos:

- Modulación coherente, cuyo funcionamiento se parece a un oscilador controlado por voltaje. Un modulador VCO-FSK funciona en modo de barrido, el cual consiste en que la salida del VCO se desplaza entre las frecuencias de 1L (frecuencia de marca) y 0L (frecuencia de espacio), es decir se mantiene la fase de la señal cuando se produce un cambio de frecuencia.
- Modulación no coherente, las fases no mantienen continuidad, es decir se establece una conmutación o cambio brusco al pasar de la frecuencia marca a la frecuencia espacio o viceversa.

Consideraciones de FSK respecto del ancho de banda

De acuerdo a Carlson, el ancho de banda mínimo requerido para una modulación FSK binaria está dado por la ecuación 2.34.

$$AB_{FSK} = \Delta f_{pp}(1 + 1/\beta); \quad \beta = \Delta f_{pp} Tb$$

Ecuación 2.34 Ancho de banda por Carlson

- $\beta \rightarrow$ *índice de modulación*
- $\Delta f_{pp} \rightarrow$ *desviación de frecuencia pico a pico = $\Delta w/2\pi$*
- $\Delta f_{pp} = 2 \Delta f_p$, $\Delta f_p \rightarrow$ *desviación de frecuencia pico*
- $Tb \rightarrow$ *duración de tiempo de bit*

Demodulador FSK

Para esquematizar un demodulador FSK se cuenta con la técnica de demodulación no coherente, la cual resulta la más sencilla y la demodulación coherente; sus respectivos diagramas se muestran en las Figuras 2.84 y Figura 2.85 respectivamente.

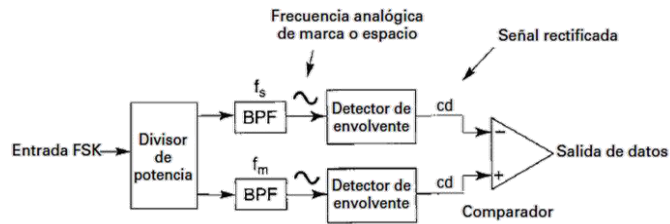


Figura 2.84 Demodulador FSK no coherente [19]

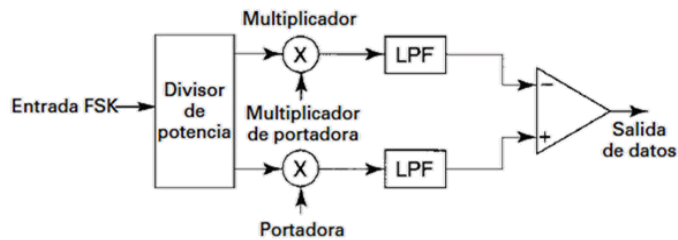


Figura 2.85 Demodulador FSK coherente [19]

En el demodulador no coherente, la señal modulada FSK ingresa simultáneamente a los dos filtros pasa banda por medio de un divisor de potencia. El filtro se encargará de pasar solo las frecuencias de marca o de espacio hacia el detector de envolvente. El bloque detector de envolvente indicará la potencia total respectiva en cada banda pasante, de tal manera que el comparador responderá a la señal de mayor potencia.

En el demodulador coherente, la señal modulada FSK que ingresa por medio de un divisor de potencia se multiplica por una señal portadora recuperada, de fase y frecuencia igual a la referente del transmisor. Dado que la frecuencia de marca y de espacio en general no son continuas (las frecuencias del receptor y el transmisor deben sincronizarse), no es posible reproducir una referencia coherente con las frecuencias de marca y de espacio, por lo tanto este tipo de demodulación no es usado, por lo que se usa como alternativa un demodulador con un circuito PLL, (Phase Locked Loop). [19]

2.3.2.3 Desarrollo práctico

En esta práctica se propone diseñar, esquematizar y simular un modulador 2FSK y un demodulador no coherente 2FSK usando la herramienta OrCAD PSpice para determinar el funcionamiento y análisis de resultados respectivamente.

De acuerdo al marco teórico de esta práctica el modulador a implementar será de tipo coherente, y de acuerdo a su funcionamiento se usará un bloque oscilador controlado por voltaje (VCO) y un elemento generador de pulsos rectangulares que será la señal de datos a la entrada del modulador.

El esquema demodulador a implementar estará basado en el funcionamiento del demodulador FSK no coherente.

Esquematización e implementación de un modulador FSK

La esquematización del modulador FSK, Figura 2.86, comprende un generador de datos de entrada usando el elemento VPULSE, para ingresar la señal binaria a ser modulada. Además, se usará el elemento EVALUATE el cual será configurado como un VCO, en donde se ingresará una expresión de la forma de una señal sinusoidal (ecuación 2.1) con una frecuencia de 4 [KHz] siendo esta la portadora, se usa una frecuencia baja para poder visualizar la modulación para varios períodos, en la práctica se podrá realizar cualquier variación de frecuencia y por medio de un acercamiento poder visualizar los cambios de frecuencia.

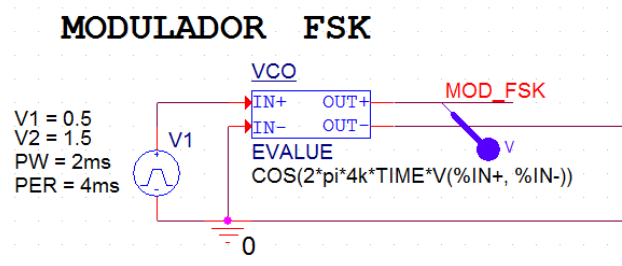


Figura 2.86 Esquema de modulación FSK

Los valores de frecuencia de FSK para 1L y 0L están determinados por una variación de frecuencia respecto a la frecuencia de la portadora ver Figura 2.87 (b). La desviación de frecuencia será controlada a partir de los niveles de voltaje, Figura 2.87(a).

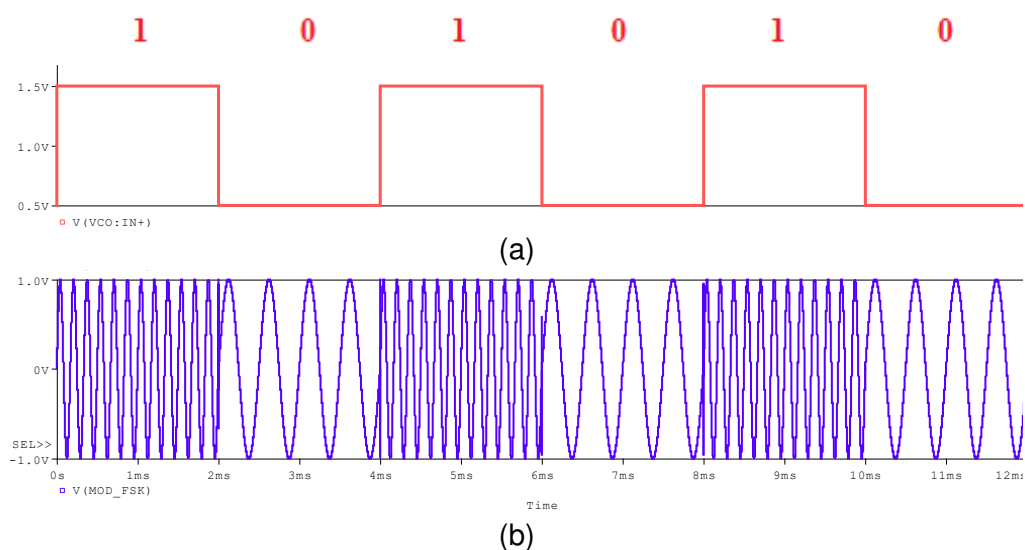


Figura 2.87 (a) Señal de datos con $T_b=2$ [ms], (b) Señal modulada 2FSK

Para determinar el espectro de frecuencia y el ancho de banda correspondiente a esta modulación 2-FSK, por fines prácticos y verificación de resultados, se considera una desviación de frecuencia pico (Δf_p) de 2 [KHz]. Es así que a partir de las ecuaciones 2.32 y 2.33, se realizará el cálculo de las frecuencias de marca y de espacio, ver resultados en las ecuaciones 2.35 y 2.36:

$$f_{marca} = f_m = f_c + \Delta f_p = 4 [KHz] + 2 [KHz] = 6 [KHz]$$

Ecuación 2.35 Frecuencia de marca

$$f_{espacio} = f_s = f_c - \Delta f_p = 4 [KHz] - 2 [KHz] = 2 [KHz]$$

Ecuación 2.36 Frecuencia de espacio

Los resultados usando la herramienta de Orcad PSpice para determinar la simulación del espectro se muestran en la Figura 2.88.

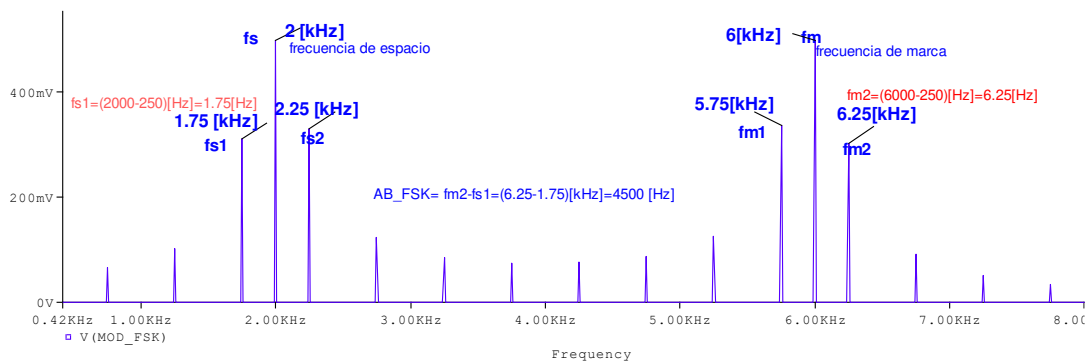


Figura 2.88 Espectro de frecuencia de la señal modulada 2-FSK

De acuerdo a la Figura 2.88, se verifica el ancho de banda a través de la ecuación 2.34, cuyos cálculos son:

- $\beta = \Delta f_{pp} * Tb = 4 [KHz] * 2 [ms] = 8$
- $AB_{FSK} = 4 [KHz] (1 + 1/8) = 4.5 [KHz]$

La señal binaria de entrada tiene un tiempo de duración de bit de 2 [ms] y un período de $T=4$ [ms], siendo una señal tren de pulsos rectangulares las componentes de frecuencia respectivas están dadas por los múltiplos de f_o (frecuencia fundamental, parámetro descrito en la ecuación 2.11) $f_o = 1/T = 1/4 ms = 250 [Hz]$.

Las componentes (fm1, fm2) respecto a la frecuencia de marca y las componentes (fs1, fs2) respecto a la frecuencia de espacio están a una distancia de $\pm f_o$, ver Figura 2.88.

Esquematización e implementación de un demodulador FSK

La propuesta de esquematización de un demodulador FSK se realizará de acuerdo al diagrama de demodulación no coherente que se presenta en la Figura 2.84, se escoge esta forma de demodulación porque en la construcción no interviene una frecuencia que se sincronice en fase, frecuencia o en ambas con la señal modulada en FSK que llega al receptor desde el transmisor; por lo tanto, el demodulador se construirá de la siguiente manera:

Primera parte: la señal recibida viajará por dos trayectos simultáneamente, el un trayecto tendrá un filtro pasa banda centrado en la frecuencia de espacio y el otro trayecto un filtro pasa banda centrado en la frecuencia de marca. El diseño que se implementa se realiza en base a los datos que se obtienen en el espectro de frecuencia de la señal modulada, ver Figura 2.88, cuyos parámetros son para un filtro pasa banda Chebychev que además resulta útil ya que permite separar bandas de frecuencia. La Figura 2.89 muestra la respuesta en frecuencia de un filtro pasa banda.

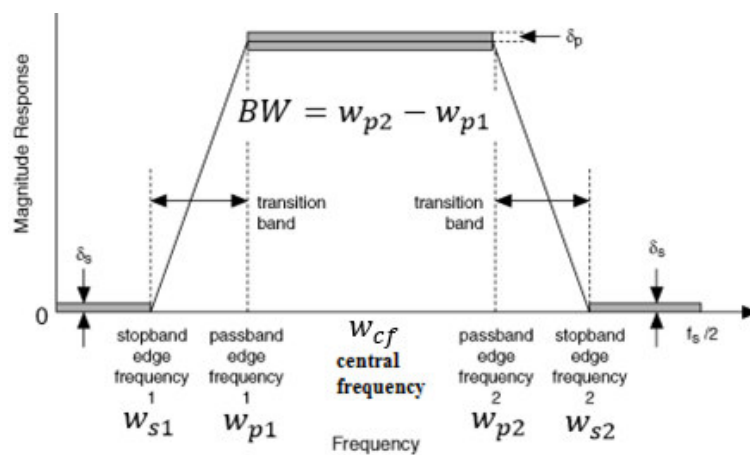


Figura 2.89 Respuesta en frecuencia de un filtro pasa banda centrado en Wcf (frecuencia central del filtro).

Para el diseño se realizarán los siguientes pasos: [36]

- Cálculo del orden del filtro, ecuación 2.37:

$$N = \frac{\cosh^{-1} \left[\frac{10^{0.1A_{min}} - 1}{\epsilon^2} \right]^{\frac{1}{2}}}{\cosh^{-1}(\alpha)}$$

Ecuación 2.37 Ecuación de orden de un filtro [33]

A_{min} = atenuación en la banda de rechazo

$$\alpha = \frac{f_{s2} - f_{s1}}{f_{p2} - f_{p1}}, \text{ ganancia de la banda de paso [33]}$$

$$\varepsilon = \text{parámetro de rizado}$$

- Cálculo del parámetro de rizado, ecuación 2.38:

$$\varepsilon = \sqrt{10^{0.1A_{max}} - 1}$$

Ecuación 2.38 Parámetro de rizado [34]

A_{max} = atenuación en la banda de paso

A_{max} , es la atenuación máxima tolerada en la banda de paso, se especifica en el intervalo desde el valor ideal 0 [dB] hasta un valor de 3 [dB], típicamente para diseño oscila de 0.05 a 3 [dB], [40] [41]. A_{min} , en cambio es el valor mínimo de atenuación en la banda de rechazo con respecto a la atenuación en la banda de paso; en este caso el valor ideal tiende hacia el valor más alto. [35][40]

- Obtención de la función de transferencia $H(s)$ del filtro; se usan las tablas de los polinomios de Chebychev [37] para un valor de atenuación máxima y el factor de rizado en la banda de paso, que corresponden con el polinomio del denominador de la función de transferencia. El uso de tablas permite simplificar el proceso de diseño evitando el uso de fórmulas para cálculo de polos, simplemente se debe calcular el orden del filtro y posteriormente usar las tablas adecuadas [34], es necesario recalcar que el uso de tablas da como resultado un filtro normalizado para el prototipo de filtro pasa bajo, por lo cual se debe realizar la transformación o desnormalización para obtener el filtro pasa banda, aplicando la siguiente expresión, ecuación 2.39 [33][38]:

$$s = \frac{s^2 + w_{cf}^2}{sBw}$$

Ecuación 2.39 Expresión para desnormalizar un prototipo pasa bajo a un prototipo pasa banda. [38]

$$w_{cf} = \sqrt{wp_2 * wp_1} = \text{frecuencia central del filtro en rad/s}$$

$$Bw = fp_2 - fp_1 = \text{ancho de banda a } -3 \text{ [dB]}$$

La Figura 2.90 muestra el prototipo de filtro pasa banda a implementar en base a los pasos de diseño mencionados.

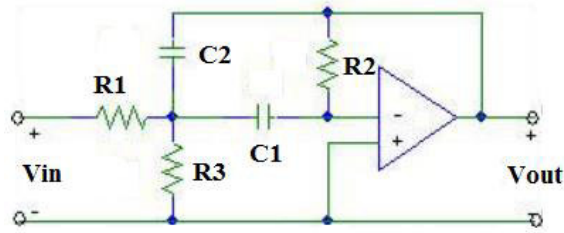


Figura 2.90 Prototipo de filtro pasa banda [37]

Cuya función de transferencia se define en la ecuación 2.40 [39]:

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{-\frac{s}{R_1 C_2}}{s^2 + s \frac{C_1 + C_2}{C_1 C_2 R_2} + \frac{1}{C_1 C_2 R_1 R_2}}$$

Ecuación 2.40 Función de transferencia del prototipo de filtro pasa banda

- Finalmente se determinan los valores de R y C, los cuales se obtienen igualando la función de transferencia del filtro obtenido por la tabla de polinomios de Chebychev y la función de transferencia del prototipo del filtro paso banda (ecuación 2.40).

Para determinar el diseño de los filtros pasa banda del receptor no coherente, se parte de los datos del espectro de la modulación FSK que se muestran en la Figura 2.88; que en base a los parámetros de la Figura 2.89, se resumen en las tablas 2.10 y 2.11.

Tabla 2.10 Datos paramétricos para diseño de filtro pasa banda Chebychev, con frecuencia central la frecuencia de espacio.

Parámetro	$w_{cf} = f_s$ $f_s = 2[\text{kHz}]$	Valor
Frecuencia de borde superior (stopband). (f_{s2})	$f_s + \Delta f_p$	4 [kHz]
Frecuencia de borde inferior (passband). (f_{p1})	$f_s - f_0$	1.75 [kHz]
Frecuencia de borde superior (passband). (f_{p2})	$f_s + f_0$	2.25 [kHz]
Frecuencia de borde inferior (stopband). (f_{s1})	$f_s - \Delta f_p$	0 [kHz]
<i>f_0 es la frecuencia fundamental</i>		

Tabla 2.11 Datos paramétricos para diseño de filtro pasa banda Chebychev, con frecuencia central la frecuencia de marca

Parámetro	$w_{cf} = f_m$ $f_m = 6[\text{kHz}]$	Valor
Frecuencia de borde superior (stopband). (f_{s2})	$f_m + \Delta f_p$	8 [kHz]
Frecuencia de borde inferior (passband). (f_{p1})	$f_m - f_o$	5.75 [kHz]
Frecuencia de borde superior (passband). (f_{p2})	$f_m + f_o$	6.25 [kHz]
Frecuencia de borde inferior (stopband). (f_{s1})	$f_m - \Delta f_p$	4 [kHz]
<i>f_o es la frecuencia fundamental</i>		

Continuando con los pasos del diseño, se calcula el orden del filtro con la ecuación 2.36, para lo cual se considerará una atenuación máxima (A_{max}) de 0.5 [dB] y una atenuación mínima (A_{min}) de 5 [dB] [40] [41] y con los valores de las tablas 2.10 y 2.11, se realizará los cálculos respectivos para determinar los filtros pasa banda correspondiente a cada línea del circuito como se indicó anteriormente en la Figura 2.84:

$$N = \frac{\cosh^{-1} \left[\frac{10^{0.1*5} - 1}{10^{0.1*0.5} - 1} \right]^{\frac{1}{2}}}{\cosh^{-1} \left(\frac{4[\text{kHz}] - 0[\text{kHz}]}{3.25[\text{kHz}] - 2.75[\text{kHz}]} \right)} = 0.77 \approx 1$$

Cálculo del parámetro de rizado a partir de la ecuación 2.37:

$$\epsilon = \sqrt{10^{0.1*0.5} - 1} = 0.3493$$

En base al parámetro A_{max} y el parámetro de rizado (ϵ), se determinará la función de transferencia $H(s)$ del filtro, usando la tabla de polinomios de Chebychev, tabla 2.12.

Tabla 2.12 Tabla de polinomios normalizados de Chebychev para filtro pasa bajo [37]

N	($A_{max}=0.5\text{dB}$, $\epsilon=0.349$)
1	($s + 2.863$)
2	($s^2 + 1.426s + 1.516$)
3	($s + 0.626$)($s^2 + 0.626s + 1.142$)
4	($s^2 + 0.351s + 1.064$)($s^2 + 0.847s + 0.356$)
5	($s + 0.362$)($s^2 + 0.224s + 1.036$)($s^2 + 0.586s + 0.477$)

Por lo tanto, como el orden del filtro calculado es $N=1$ (N , determina la pendiente de atenuación, ver Figura 2.5), la función de transferencia normalizada es la siguiente $H(s)$: (1).

$$H(s) = \frac{1}{s + 2.863}$$

Que al aplicar la ecuación 2.39, se desnormaliza para obtener el prototipo de filtro pasa banda requerido, de la siguiente manera: (2)

$$H(s) = \frac{1}{\left[\frac{s^2 + w_{cf}^2}{sBw} \right] + 2.863}$$

De la ecuación 2.39; la frecuencia central del filtro y el ancho de banda a -3 [dB] en [rad/s] está dado de la siguiente manera:

$$f_{cf} = \sqrt{f_{p1} * f_{p2}} = 1984.31 [Hz] \approx 2 [KHz];$$

$$w_{cf} = 2\pi f_{cf} = 12467.81 [rad/s] \rightarrow 12468 [rad/s]$$

Ancho de banda a -3 [dB]:

$$Bw = 2\pi (2,25 [KHz] - 1,75 [KHz]) = 3141.59 [rad/s] \rightarrow 3142 [rad/s]$$

Reemplazando en (2), se obtiene la siguiente función de transferencia: (3)

$$H(s) = \frac{1}{\frac{s^2 + 12468^2}{3142s} + 2.863}$$

$$H(s) = \frac{3142s}{s^2 + 8995.55s + 12468^2}$$

Finalmente se calcula los valores de R y C , en base a la función de transferencia de la ecuación 2.39:

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{-\frac{s}{R_1 C_2}}{s^2 + s \frac{C_1 + C_2}{C_1 C_2 R_2} + \frac{1}{C_1 C_2 R_1 R_2}}$$

Considerando $C = C_1 = C_2$, se tiene: (4)

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{-\frac{s}{R_1 C}}{s^2 + s \frac{2}{C R_2} + \frac{1}{C^2 R_1 R_2}}$$

Igualando los términos de las ecuaciones (3) y (4), se determinan:

Con $R_1 = 1000 [\Omega]$

$$12468^2 = \frac{1}{C^2 * 1000 * R_2} \rightarrow R_2 = \frac{1}{C^2 * 1000 * 12468^2}$$

$$8995.55 = \frac{2}{R_2 * C} \rightarrow C = \frac{2}{R_2 * 8995.55}$$

Se reemplaza la igualdad de C en R2:

$$R_2 = \frac{1}{\left(\frac{2}{R_2 * 8995.55}\right)^2 * 1000 * 12468^2} = 7.68 [K\Omega] \approx 10 [KHz]$$

Con $R_2=10 [K\Omega]$, se determina el valor de C:

$$C = \frac{2}{10 [K\Omega] * 8995.5}$$

$$C = C_1 = C_2 = 22,23 [nF] \approx 22 [nF]$$

Para el diseño del filtro con los datos de la tabla 2.11, se repite el mismo análisis anterior. De la ecuación 2.39; se realiza el cálculo de la frecuencia central del filtro y el ancho de banda en [rad/s] a -3 [dB] de la siguiente manera:

$$f_{cf} = \sqrt{f_{p1} * f_{p2}} = 5994.79 [Hz] \approx 6 [KHz]$$

$$\omega_{cf} = 2\pi f_{cf} = 37666.37 [rad/s] \rightarrow 37666 [rad/s]$$

$$Bw = 2\pi (6,25 [KHz] - 5,75 [KHz]) = 3141.59 [rad/s] \rightarrow 3142 [rad/s]$$

Se reemplaza en (2), los valores obtenidos de frecuencia central en [rad/s] y ancho de banda, para obtener la función de transferencia: (5)

$$H(s) = \frac{3142s}{s^2 + 8995.55s + 37666^2}$$

Que al igualar los términos de (4) y (5), se determina los valores de R4, C3 y C4.

El diseño de filtros pasa banda se observa en el diagrama circuital de la Figura 2.91, y el resultado de la señal modulada que ingresa simultáneamente a los filtros pasa banda se ilustra en la Figura 2.92.

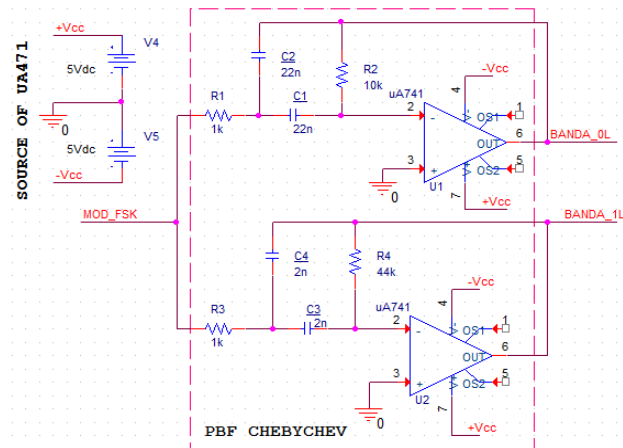
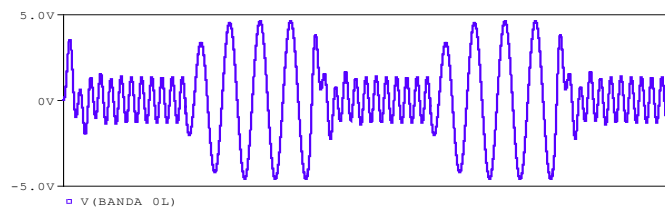
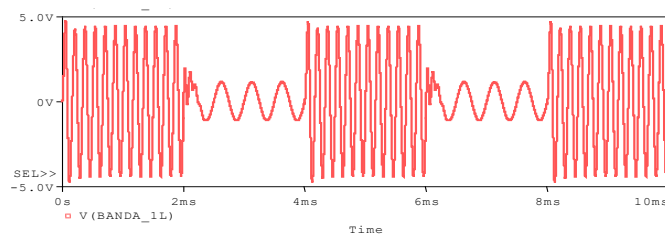


Figura 2.91 Bloque de filtros paso banda



(a)



(b)

Figura 2.92 Señal recibida de los filtros pasa banda

(a) para frecuencia de espacio, (b) para frecuencia de marca

Las señales de la Figura 2.92 (a y b), transmiten la frecuencia asignada para 0 lógico (frecuencia de espacio) o para 1 lógico (frecuencia de marca) a su correspondiente bloque detector de envolvente.

El bloque detector de envolvente forma la segunda parte del receptor FSK propuesto, Figura 2.93, este esquema está compuesto de un multiplicador y un filtro pasa bajo en cada línea del circuito. Su funcionamiento consiste en que las señales que provienen de los filtros

pasa banda se multiplican por sí mismas dando lugar a una señal rectificada, la misma que luego se expone a una etapa de filtro pasa bajo (utilizando el elemento *LOPASS* de PSpice), para obtener como resultado dos niveles, los cuales finalmente ingresan a una etapa de decisión.

La etapa de decisión se representa por un comparador diferencial cuyo elemento esencial en el circuito de la Figura 2.93 es el amplificador operacional, el cual permitirá determinar el dato (0 ó 1) transmitido al calcular o comparar la amplitud de envolvente.

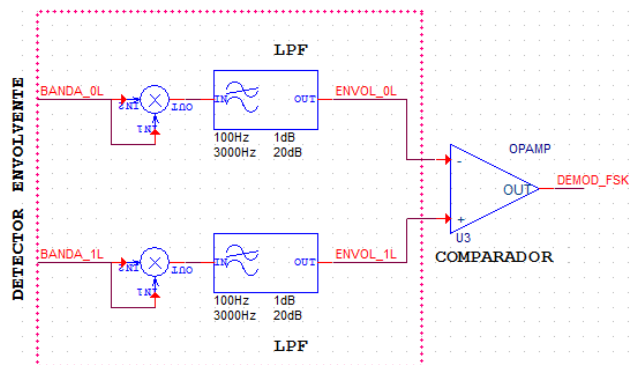


Figura 2.93 Detector de envolvente

Las señales obtenidas del proceso detector de envolvente se muestran en la Figura 2.94.

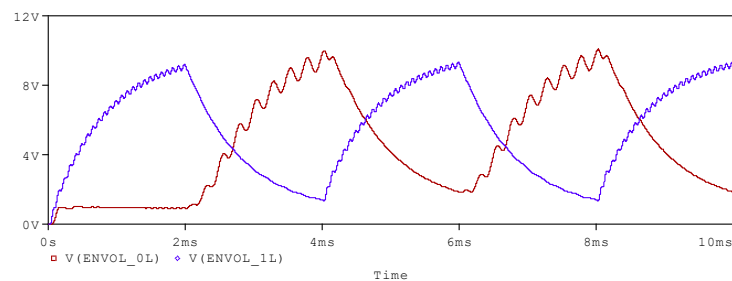


Figura 2.94 Señales resultantes del bloque detector de envolvente, para la línea del circuito del 0L (rojo) y del 1L (azul)

Estas señales obtenidas, se someten a un bloque comparador para establecer la demodulación o reconstrucción de los datos de entrada al modulador (Figura 2.87), como se ilustra en la Figura 2.95.

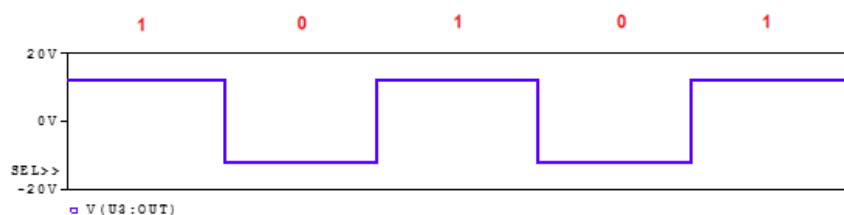


Figura 2.95 Señal demodulada

Los diagramas de las Figuras 2.86, 2.91 y 2.93 en conjunto forman un circuito modulador/demodulador FSK. Los resultados del circuito desarrollado se analizarán en el siguiente capítulo.

2.3.3 Práctica 07: Modulación PSK

2.3.3.1 Objetivos

- Comprender y entender el funcionamiento de la técnica de modulación PSK.
- Analizar resultados de simulación de la modulación B-PSK, Q-PSK y 8-PSK mediante la herramienta OrCAD PSpice en los escenarios respectivos.
- Conocer las ventajas y desventajas que brinda esta técnica de modulación digital.
- Analizar el espectro de modulación de B-PSK con respecto a la modulación Q-PSK.
- Establecer escenarios comparativos de modulación M-PSK con B-PSK y analizar los resultados, verificando sus respectivas ventajas y desventajas.
- Analizar el espectro de frecuencia PSK respecto al espectro de frecuencia ASK.

2.3.3.2 Marco teórico [13] [14] [19] [22]

En general la modulación PSK (*Phase-Shift Keying*) es un tipo de modulación digital donde la fase de la portadora varía de acuerdo a la señal de entrada(datos) mientras que la amplitud y la frecuencia se mantienen constantes, ver Figura 2.96.

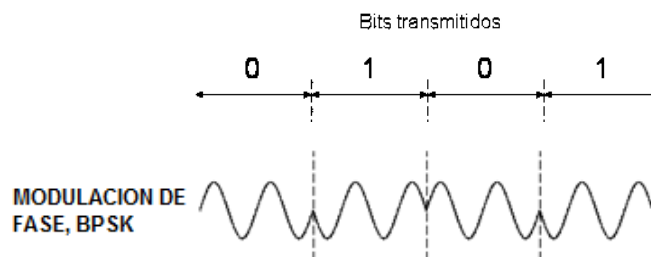


Figura 2.96 Modulación B-PSK [23]

Matemáticamente se describe como la ecuación 2.41.

$$PSK(t) = V_c \cos \left\{ w_c t + \frac{[b_M(t) \cdot \Delta\theta]}{2} \right\} ; \quad \Delta\theta = 2\pi/M$$

Ecuación 2.41 Expresión matemática de la modulación PSK [13]

Donde:

- V_c → Amplitud pico de la portadora [Voltios]

- $\Delta\theta \rightarrow$ Separación entre las fases adyacentes de la señal modulada
- $M = 2^N \rightarrow$ Número de estados, N indica el número de bits por estado
- $w_c = 2 * \pi * f_c \rightarrow$ frecuencia de la portadora en [rad/s]
- $b_M =$ señal digital NRZ en banda base de M niveles, toma valores de $\pm 1, \pm 3, \pm 5, \dots$

La relación entre velocidad de transmisión (V_t) y velocidad de modulación (V_m) está dada en la ecuación 2.42.

$$V_m = V_t / N$$

Ecuación 2.42 Relación de velocidad de modulación para M-PSK [14]

Donde:

- $N = \log_2 M \rightarrow$ número de bits por estado
- $V_m \rightarrow$ velocidad de modulación [baudios]
- $V_t \rightarrow$ velocidad de transmisión (inversa del tiempo de bit) [bps]

La modulación de múltiples estados (M-PSK), siendo los más comunes B-PSK ó 2-PSK, Q-PSK ó 4-PSK, 8-PSK y 16-PSK, puede ser representado por la suma de dos señales en cuadratura usando el diagrama de constelaciones (eje horizontal "I" (en fase) y eje vertical "Q" (en cuadratura)). En donde se representa con un punto los estados de la portadora modulada en relación de amplitud y fase, como se ilustra en la Figura 2.97, en los diagramas de constelación de las modulaciones B-PSK, Q-PSK Y 8-PSK.

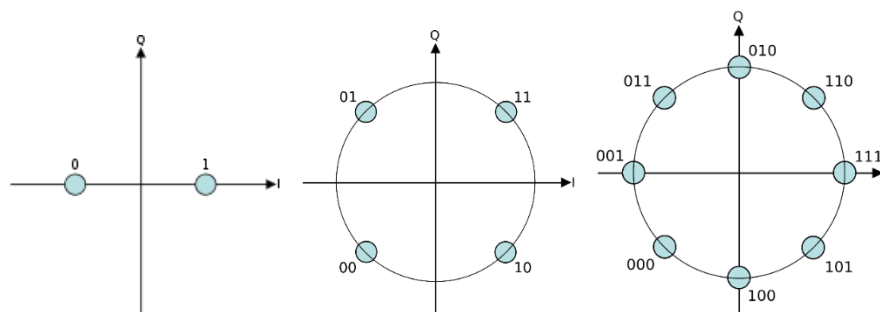


Figura 2.97 Diagramas de constelación de B-PSK, Q-PSK y 8-PSK [13]

En esta práctica se analizará el esquema de modulación 2-PSK y se establecerá escenarios comparativos con la modulación Q-PSK y 8-PSK usando sus respectivos esquemas de modulación que se simularán y se detallarán más adelante.

Esquema de un modulador 2-PSK ó B-PSK

El diagrama de modulación 2-PSK utiliza un modulador balanceado el cual funciona como un conmutador de fase entre 0 y π radianes con referencia al oscilador de la portadora, de

acuerdo a los estados lógicos de la señal de datos, como se muestra en el diagrama de bloques de la Figura 2.98.

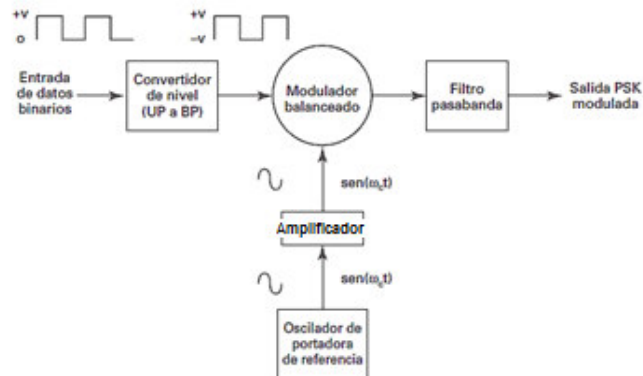


Figura 2.98 Diagrama de bloques del modulador PSK [19]

La Figura 2.99, muestra el esquema de un circuito para modular 2-PSK, cuya función consiste en dos entradas, de las cuales en una ingresará la señal portadora y en otra ingresarán la señal de datos (digitales).

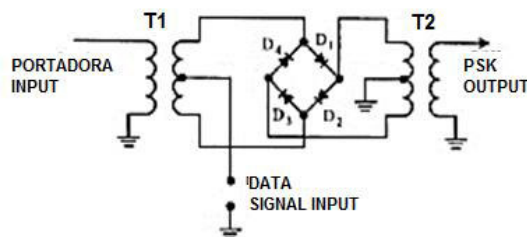


Figura 2.99 Modulador 2-PSK [19]

El componente principal de este esquema son los diodos⁴⁹, los cuales permiten la circulación de la corriente en un solo sentido. Su diseño cuenta con un arreglo de diodos los cuales son controlados (encienden o apagan) en base a los niveles de voltaje de entrada de la señal de datos. Para una operación adecuada del modulador, la amplitud de la modulante debe ser un valor mucho mayor que el máximo de la portadora.

El funcionamiento se caracteriza en la polarización de los diodos D1, D2, D3 y D4, cuyo encendido y apagado depende de la condición lógica de la señal de datos ingresada, si es 1L implica un voltaje positivo y si es 0L implica un voltaje negativo. De acuerdo a estas

⁴⁹ Los diodos cuando se someten a una diferencia de tensión se dice que están polarizados; polarización directa implica paso de corriente y polarización inversa no conduce corriente.

condiciones los diodos se polarizan y determinan la fase o el desfase (180°) de los voltajes generados a través de los transformadores T1 y T2.

Esquemas de modulador M-PSK

En general un modulador M-PSK (PSK multinivel), asigna una fase por cada condición de entrada con N bits agrupados, el diagrama de bloques se presenta en la Figura 2.100(b).

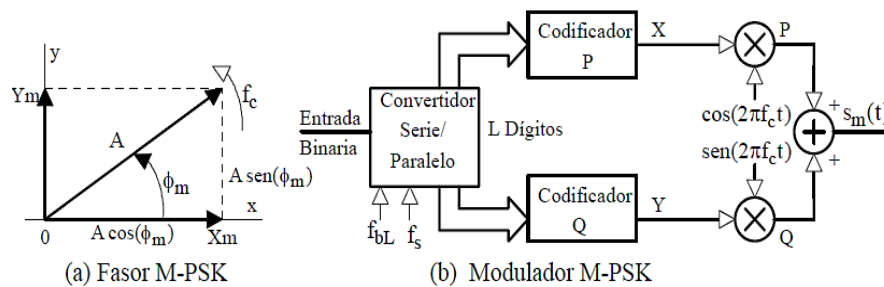


Figura 2.100 Modulador M-PSK [19]

Los M estados a la salida del modulador con la respectiva desviación de fase, se pueden calcular utilizando la ecuación 2.41, y la representación de los estados en el diagrama de constelaciones en relación de la amplitud y fase, ver Figura 2.100(a) se determinan a partir de la ecuación 2.43.

$$A = A_x \cos(\theta_m) + A_y \sen(\theta_m)$$

Ecuación 2.43 Componentes en cuadratura, M-PSK [14]

Donde:

- $|A| = \sqrt{Ax_m^2 + Ay_m^2} \rightarrow \text{Amplitud}$
- $\theta_m = \arctg \frac{Y_m}{X_m} \rightarrow \text{Fase}$

La representación de la modulación en el diagrama de constelación permitirá observar que a medida que aumenta el número de posibles fases a la salida del modulador, los ángulos de desfase se tornarán más pequeños, lo cual resulta una desventaja ya que en el demodulador induce errores al tratar de detectar cuál estado se está transmitiendo.

Además, también se considera que, a mayor número de posibles fases, mayor es la cantidad de información a transmitir, ya que dependiendo del valor del estado M, se agrupan N bits por cada estado (símbolo); ejemplo para Q-PSK se agrupan 2 bits por símbolo, para 8PSK se agrupan 3 bits por símbolo, para 16-PSK se agrupan 4 bits por símbolo, etc. Lo cual también implica que se obtiene mayores velocidades de transmisión.

Respecto al cálculo del ancho de banda mínimo necesario para una modulación M-PSK, se considera a partir de la modulación B-PSK, donde se puede demostrar a través de la ecuación 2.42 que la velocidad de transmisión resulta equivalente a la velocidad de modulación, de tal manera que un modulador B-PSK requiere un ancho de banda correspondiente al valor de la velocidad de transmisión.

Para el caso de la modulación multiestados M-PSK, el ancho de banda se determinará utilizando la expresión de la ecuación 2.44, donde AB_{BPSK} es el ancho de banda requerido para la modulación B-PSK.

$$AB_{MPSK} = \frac{AB_{BPSK}}{N}$$

Ecuación 2.44 Ancho de banda requerido para M-PSK [14]

Además, se puede determinar que, relacionado con la ecuación 2.42, se establece que el ancho de banda mínimo requerido para modulación M-PSK se reduce en la misma proporción que se reduce la velocidad de modulación de la señal.

➤ Esquema de modulación 4-PSK o Q-PSK

Q-PSK, es un tipo de modulación de cuatro fases distintas que se encuentran equidistantes una de la otra en $\pi/2$ [rad], la cual requiere cuatro combinaciones distintas de dos bits para representar las condiciones de entrada; el diagrama de bloques que lleva al cabo el proceso de modulación 4-PSK se describe en la Figura 2.101.

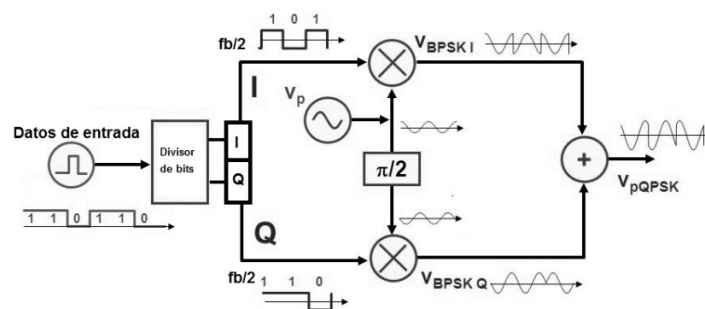


Figura 2.101 Modulador Q-PSK

El esquema Q-PSK describe que requiere de dos canales (I y Q) para realizar la modulación de una señal de datos de entrada, donde el canal "I" se usa para la modulación de una portadora en fase y el canal "Q" para la modulación de una portadora desfasada (90°). El sumador se encargará de la combinación de las dos señales en cuadratura, en este caso resultarán cuatro fasores posibles que resumidamente se expresan en la Tabla 2.13, y las

expresiones correspondientes a los estados de acuerdo a la ecuación 2.41 se definen de la siguiente manera:

$$\text{PSK}(t) = \cos\{\omega_c t + 3\pi/4\} \rightarrow 10$$

$$\text{PSK}(t) = \cos\{\omega_c t + \pi/4\} \rightarrow 11$$

$$\text{PSK}(t) = \cos\{\omega_c t - \pi/4\} \rightarrow 01$$

$$\text{PSK}(t) = \cos\{\omega_c t - 3\pi/4\} \rightarrow 00$$

Tabla 2.13 Tabla de verdad, Modulación de 4-PSK

Entrada binaria		Salida 4-PSK $\Delta\theta = \pi/2$			
Q	I	b_M	Fase $\frac{[b_M(t) \cdot \Delta\theta]}{2}$	Coefficientes en cuadratura $\text{sen}(\omega t)$	Coefficientes en fase $\text{cos}(\omega t)$
1	0	+3	135°	0.7071	-0.7071
1	1	+1	45°	0.7071	0.7071
0	1	-1	315°	-0.7071	0.7071
0	0	-3	225°	-0.7071	-0.7071

En base a la tabla 2.13, se determina el diagrama fasorial y de constelaciones que se muestra en la Figura 2.102 y la forma de la modulación en el tiempo en la Figura 2.103.

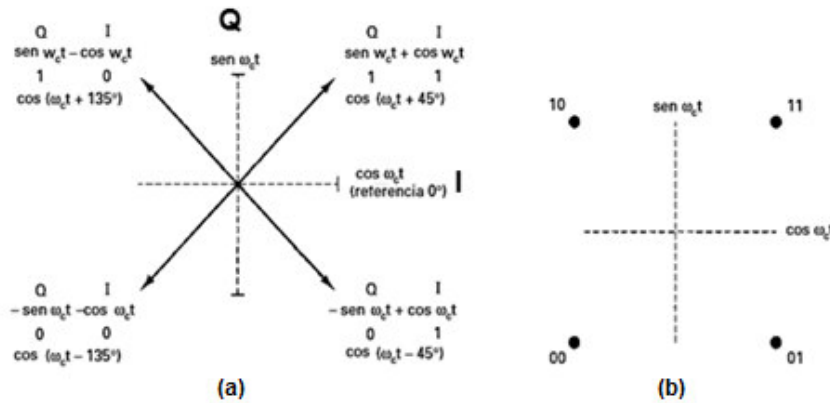


Figura 2.102 Modulación QPSK (b) Diagrama fasorial, (c) Diagrama de constelación [19]

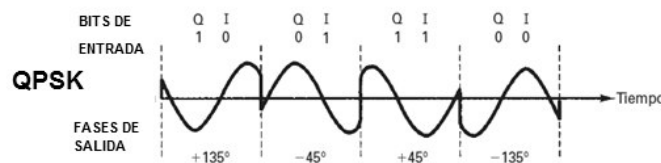


Figura 2.103 Señal a la salida del modulador QPSK [19]

➤ Esquema de un modulador 8-PSK

8-PSK, es un tipo de modulación digital que permite obtener a la salida ocho posibles fases distintas, las cuales se encuentran equidistantes una de la otra en $\pi/4$ [rad], para la modulación a la entrada requiere ocho combinaciones distintas de tres bits para representar las condiciones de entrada; el diagrama de bloques que lleva al cabo el proceso de modulación 8-PSK se presenta en la Figura 2.104.

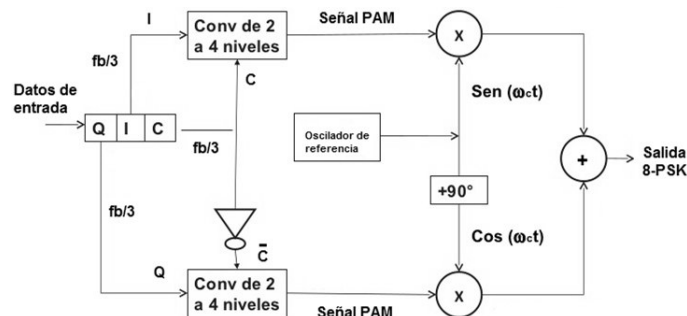


Figura 2.104 Modulador 8-PSK [19]

Cuya modulación requiere de tres canales (I, Q, C), en donde el canal “I” se usa para la modulación de una portadora en fase, el canal “Q” para la modulación de una portadora desfasada (90°) y el canal “C” como un canal de control. Los bits del canal “I” y “Q” entran al convertidor de 2 a 4 niveles del canal “I” y de la misma manera los bits del canal “Q” y “ \bar{C} ” (inversa del canal “C”) ingresan al convertidor de 2 a 4 niveles del canal “Q”; los convertidores de 2 a 4 niveles hacen posible que con dos bits de entrada se obtengan 4 voltajes de salida.

Los bits de I y Q indican la polaridad de la señal y el bit de C determina la magnitud, el sumador se encargará de la combinación de las dos señales en cuadratura. Los niveles que se obtienen a la salida del conversor de dos a cuatro son valores relativos, Tabla 2.14, esto quiere decir que se puede asignar a cada nivel cualquier valor siempre y cuando la relación sea equivalente a la fase que corresponde. 8-PSK presenta una desviación de fase de $\Delta\theta = \pi/4$ [rad] entre las fases de la señal modulada y las magnitudes de cada uno de los fasores son iguales.

Tabla 2.14 Tabla de verdad convertidor 2 a 4

I/Q	C	Salida	C'	Salida
0	0	-0.383	1	-0.924
0	1	-0.924	0	-0.383
1	0	+0.383	1	+0.924
1	1	+0.924	0	+0.383

Los valores relativos detallados en la tabla 2.14, resultan del siguiente análisis, 8-PSK de acuerdo a la ecuación 2.41 tiene un desplazamiento de fase al menos de +/- 22.5°, como se detalla en la tabla 2.15, donde los valores de los niveles pueden ser 0.383 y 0.924 ya que cumplen con lo descrito al determinar la relación de arco tangente de estos valores. Si se requiere aumentar o disminuir la amplitud del fasor, se debe cambiar el valor de la relación de los niveles de tal manera que se cumpla con el valor de desplazamiento de fase.

Las expresiones a la salida del modulador 8-PSK de acuerdo a la ecuación 2.41 se definen de la siguiente manera:

$$\text{PSK}(t) = \cos\{w_c t + 7\pi/8\} \rightarrow \mathbf{000} \quad \text{PSK}(t) = \cos\{w_c t - 7\pi/8\} \rightarrow \mathbf{111}$$

$$\text{PSK}(t) = \cos\{w_c t + 5\pi/8\} \rightarrow \mathbf{001} \quad \text{PSK}(t) = \cos\{w_c t - 5\pi/8\} \rightarrow \mathbf{110}$$

$$\text{PSK}(t) = \cos\{w_c t + 3\pi/8\} \rightarrow \mathbf{010} \quad \text{PSK}(t) = \cos\{w_c t - 3\pi/8\} \rightarrow \mathbf{101}$$

$$\text{PSK}(t) = \cos\{w_c t + 1\pi/8\} \rightarrow \mathbf{011} \quad \text{PSK}(t) = \cos\{w_c t - 1\pi/8\} \rightarrow \mathbf{100}$$

La tabla 2.15 muestra los parámetros para obtener el diagrama de constelaciones, Figura 2.105.

Tabla 2.15 Tabla de verdad, Modulación de 8-PSK

Entrada binaria			Salida 8-PSK $\Delta\theta = \pi/4 [rad]$			
Q	I	C	b_M	$\frac{Fase [b_M(t) \cdot \Delta\theta]}{2}$	Coeficientes en cuadratura sen (wt)	Coeficientes en fase cos(wt)
0	0	0	+7	157.5°	0.383	-0.924
0	0	1	+5	112.5°	0.924	-0.383
0	1	0	+3	67.5°	0.924	0.383
0	1	1	+1	22.5°	0.383	0.924
1	0	0	-1	-22.5	-0.383	0.924
1	0	1	-3	-67.5	-0.924	0.383
1	1	0	-5	-112.5	-0.924	-0.383
1	1	1	-7	-157.5	-0.383	-0.924

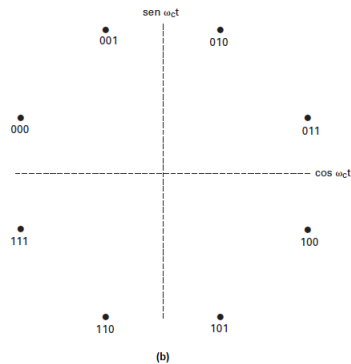


Figura 2.105 Diagrama de constelación 8-PSK [19]

2.3.3.3 Desarrollo práctico

En este desarrollo práctico se propone esquematizar y simular un modulador BPSK, QPSK y 8PSK utilizando la herramienta OrCAD PSpice. Los esquemas permitirán analizar el funcionamiento de acuerdo a la descripción que se presenta en el marco teórico de esta práctica y además establecer escenarios comparativos para su respectivo análisis.

- La esquematización debe componerse de los elementos necesarios para producir la modulación por desplazamiento de fase de los datos digitales ingresados de cada uno de los diagramas (B-PSK, Q-PSK, 8-PAK) y deberá permitir visualizar los resultados para su correspondiente verificación a través de cálculos matemáticos.
- Los esquemas permitirán variar parámetros como frecuencia de portadora, amplitud de datos de entrada, tiempos de bit, etc., para verificar su correcto funcionamiento y análisis de resultados.

Esquematización de un modulador B-PSK

Se realiza el esquema de modulación en base a la Figura 2.99, descrita en el marco teórico, este esquema cuenta con los elementos necesarios para la modulación B-PSK como se muestra en la Figura 2.106.

La funcionalidad consiste en un conmutador de inversión de fase. Es así que parte de sus componentes principales son dos señales de entrada, una se genera a partir de un circuito oscilador sinusoidal que da lugar a la señal portadora, y la otra a partir de un generador de señal pulso que determina los datos o condición lógica de la entrada de datos. Dependiendo del dato binario a la entrada, se determina la polaridad de los diodos D1, D2, D3 y D4.

- Si la señal de datos que ingresa es 1L, implica voltaje positivo; los diodos D1 y D2 conducen (polarización directa), mientras que los diodos D3 y D4 no conducen, se

apagan (polarización inversa), causando de esta manera que el voltaje de la portadora que se genera a través del transformador T2 esté en fase con el voltaje de la portadora producido por medio de T1.

- Si la señal de datos que ingresa es 0L, implica voltaje negativo; los diodos D1 y D2 se abren (polarización inversa), mientras que los diodos D3 y D4 conducen (polarización directa), causando de esta manera que el voltaje de la portadora que se genera por medio de T2 se encuentre desfasado 180° con respecto al voltaje de portadora producido por T1.

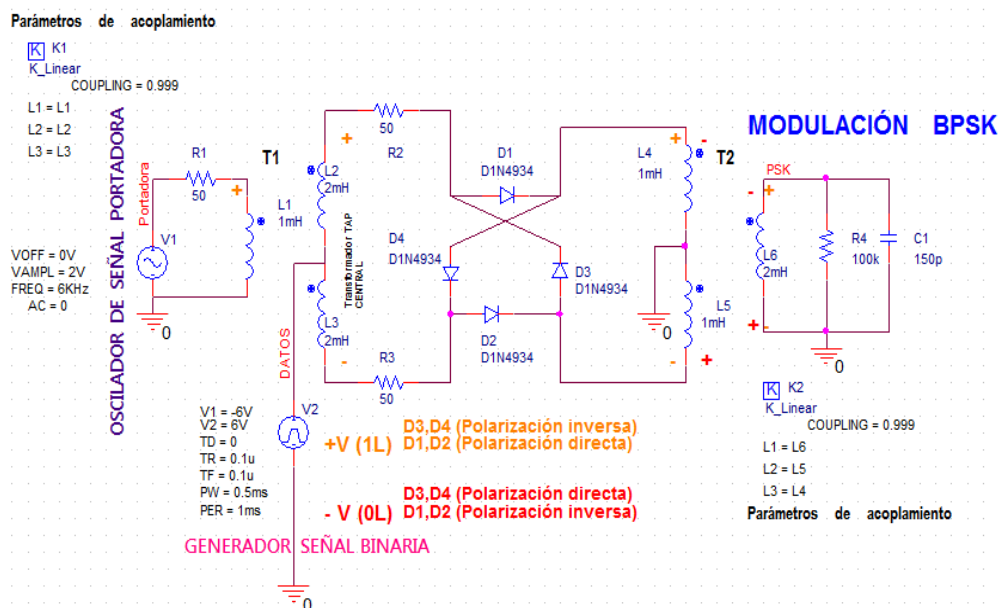


Figura 2.106 Modulador B-PSK

El transformador de tap central cumplirá la función de desfaseador (0 a π), el cual variará de acuerdo al nivel de voltaje ingresado por el bloque generador de pulsos, el cual determina la condición lógica de entrada.

Es importante establecer en la práctica un coeficiente de acoplamiento adecuado para el correcto funcionamiento del transformador, cuyo intervalo oscilará entre los valores de 0 a 1 , donde cero indica que no existe acoplamiento y 1 establece el acoplamiento de un transformador ideal, en la simulación de la práctica si se coloca un valor fuera del rango mencionado se generará un error.

Es así que entre más cercano al valor 1 , se evitará que en la simulación se genere una oscilación en la conmutación de condición lógica, como se indica en la Figura 2.107 para distintos valores de coeficientes de acoplamiento.

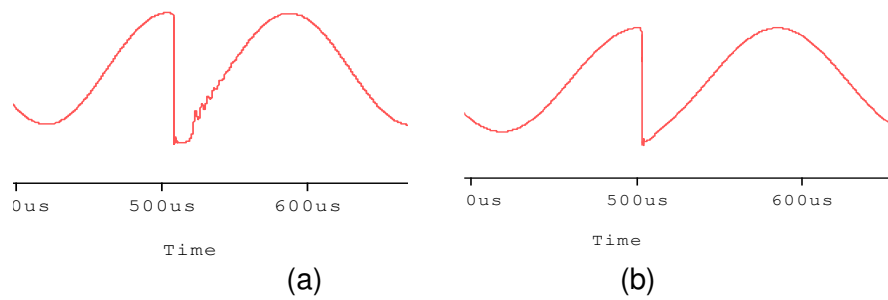


Figura 2.107 Coeficiente de acoplamiento (a) 0.85, (b) 0.95

La modulación B-PSK de una secuencia de bits de entrada modulada con una portadora de 6 [KHz], se visualiza en la Figura 2.108, donde se verifica la variación de fase que se presenta al cambiar la condición lógica para una señal de datos con $T_b = 0.5$ [ms]

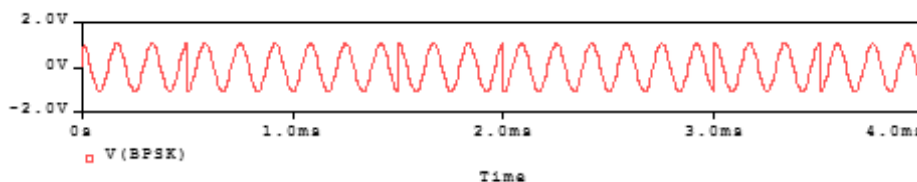
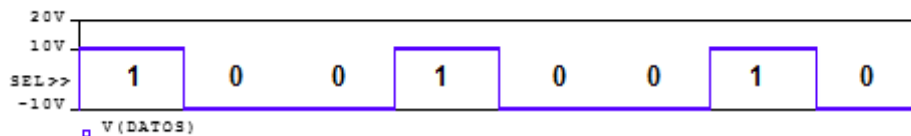
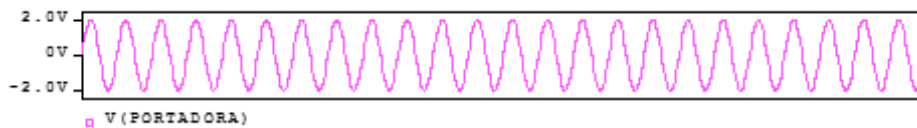
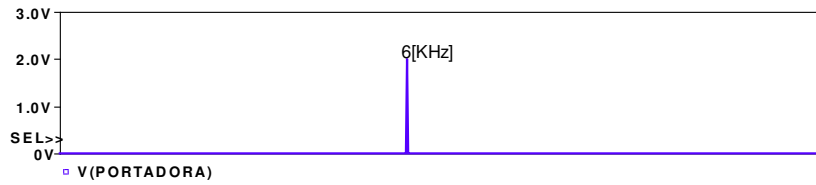
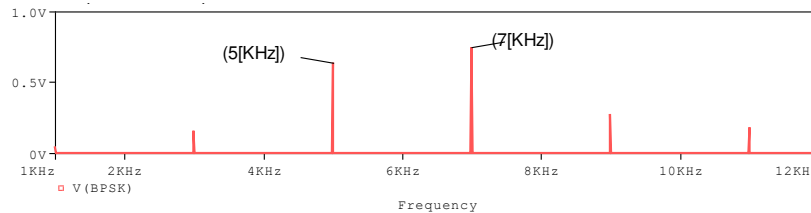


Figura 2.108 (a) Señal portadora, (b) Datos, (c) Modulación B-PSK

El espectro de frecuencia se presenta en la Figura 2.109, donde gráficamente se puede determinar el ancho de banda mínimo necesario para la modulación, ver Figura 2.109 (b).



(a)



(b)

Figura 2.109 Espectro de frecuencia de (a) portadora, (b) la modulación B-PSK

Que comparado matemáticamente usando la ecuación 2.44, se verifica lo siguiente:

$$AB_{BPSK} = \frac{1}{0.5 [ms]} = 2 [KHz]$$

Esquematización de un modulador Q-PSK

Se realiza el esquema de modulación en base a la Figura 2.101, cuyo funcionamiento consiste en implementar dos bloques de modulación, el uno simulará el canal I (fase) y otro el canal Q (cuadratura). Cada bloque contiene una señal de entrada usando un generador de pulsos rectangulares con $T_b = 0.25 [ms]$, y un generador de onda sinusoidal con frecuencia de 6 [KHz], de tal manera que se modulan en B-PSK, y cuya salida del canal I y Q ingresa paralelamente al sumador para dar lugar de esta manera que se formen las 4 posibilidades de 2 bits como se indica en la tabla 2.13, y las señales se sumen en cuadratura; el esquema de simulación se ilustra en la Figura 2.110.

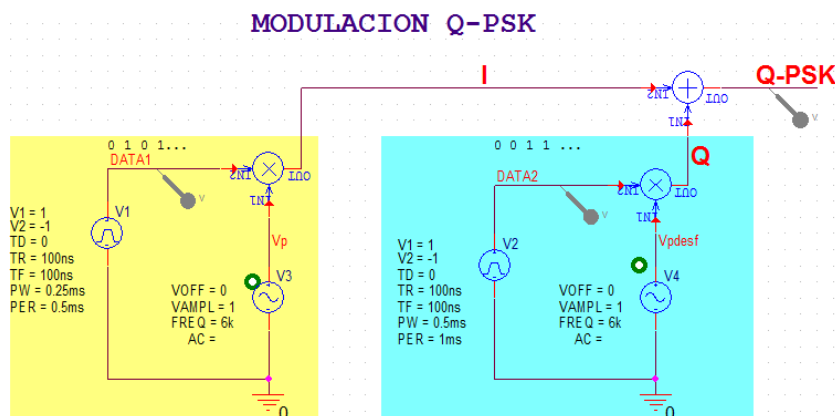


Figura 2.110 Modulador Q-PSK

La Figura 2.111 muestra el resultado de la modulación obtenido en función del tiempo, y la Figura 2.112 el respectivo espectro de frecuencia.

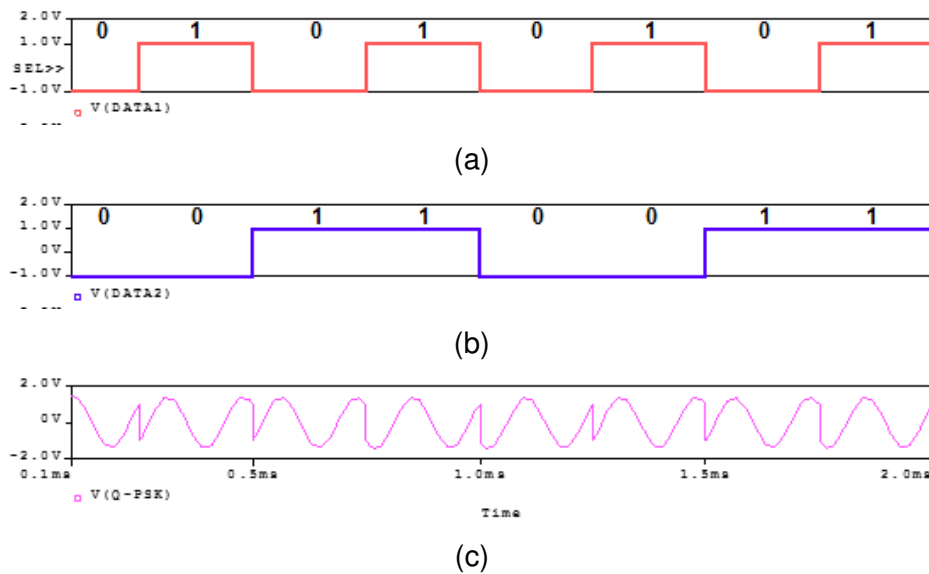


Figura 2.111 (a) Datos del canal I (b) Datos del canal Q (c) Modulación Q-PSK

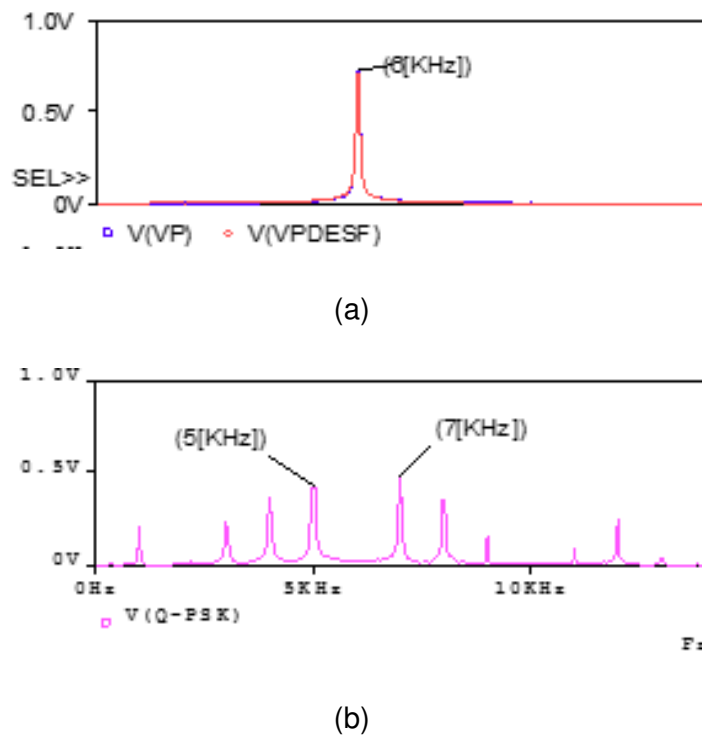


Figura 2.112 Espectro de frecuencia (a) portadora canal I y canal Q, (b) modulación Q-PSK

Si, se compara el resultado de la diferencia de frecuencias obtenidas de la simulación de la Figura 2.112 (b) con el respectivo cálculo matemático de acuerdo a la ecuación 2.44, se verifica lo siguiente:

$$AB_{QPSK} = \frac{1/0.25 [ms]}{\log_2 4} = 2 [KHz]$$

Esquematación de un modulador 8-PSK

El esquema de modulación se construirá de acuerdo al diagrama de bloques que se presentó en la Figura 2.104. En donde el modulador simulado Figura 2.113, consta de un elemento DSTM el cual permitirá el ingreso de una serie de bits de datos a través de un archivo en .txt, luego con un bit splitter se convertirá los datos ingresados en forma serial en una salida paralela para así obtener los canales necesarios Q, I, y C. Después las salidas de I y C ingresarán al convertidor de 2 a 4 niveles así como también el canal Q y C', donde C' representa la señal proveniente del inverso del canal C, como se detalla en la Figura 2.114 (Ver referencia de etiquetas en las conexiones del circuito en la Figura 2.113).

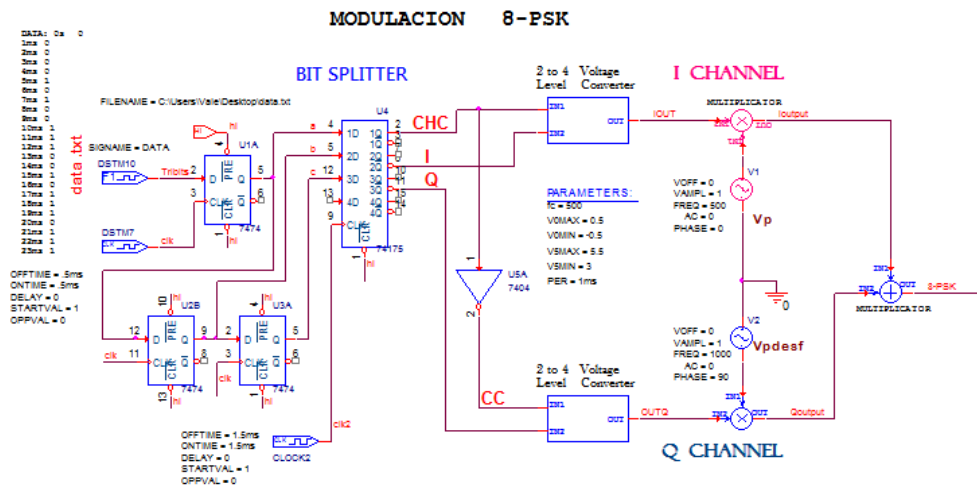


Figura 2.113 Modulador 8-PSK

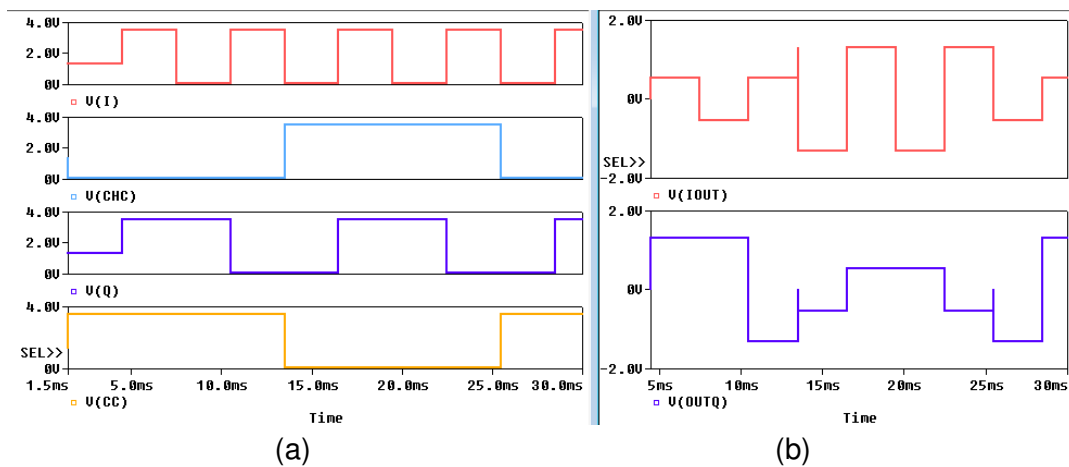


Figura 2.114 Señales a la (a) entrada de los convertidores y (b) salida de los convertidores de 2 a 4 niveles

Los convertidores de nivel de dos a cuatro funcionan de tal manera que con dos bits de entrada se obtenga a la salida 4 posibles niveles de voltaje, Figura 2.114 (b). El elemento en la práctica debe ser configurado dependiendo de los niveles relativos que se vayan a colocar, como se muestra en la captura de pantalla de la Figura 2.115.

ABM2.Normal	
A	
SCHEMATIC1 : PAGE1 : ABM4	
Designator	
EXP1	$\text{if}((\text{V}(\text{CHC}) > \text{v0min}) \& (\text{V}(\text{CHC}) < \text{v0max}) \& (\text{V}(\text{I}) > \text{v0min}) \& (\text{V}(\text{I}) < \text{v0max})), -0.541, 0)$
EXP2	$\text{+if}(((\text{V}(\text{CHC}) > \text{v5min}) \& (\text{V}(\text{CHC}) < \text{v5max})) \& ((\text{V}(\text{I}) > \text{v0min}) \& (\text{V}(\text{I}) < \text{v0max}))), -1.307, 0)$
EXP3	$\text{+if}(((\text{V}(\text{CHC}) > \text{v0min}) \& (\text{V}(\text{CHC}) < \text{v0max})) \& ((\text{V}(\text{I}) > \text{v5min}) \& (\text{V}(\text{I}) < \text{v5max}))), 0.541, 0)$
EXP4	$\text{+if}(((\text{V}(\text{CHC}) > \text{v5min}) \& (\text{V}(\text{CHC}) < \text{v5max})) \& ((\text{V}(\text{I}) > \text{v5min}) \& (\text{V}(\text{I}) < \text{v5max}))), 1.307, 0)$

Figura 2.115 Configuración del convertidor de nivel de 2 a 4

En la figura 2.116 se muestran los resultados obtenidos de la modulación 8-PSK donde se verifica los cambios de fase al pasar de un nivel a otro.

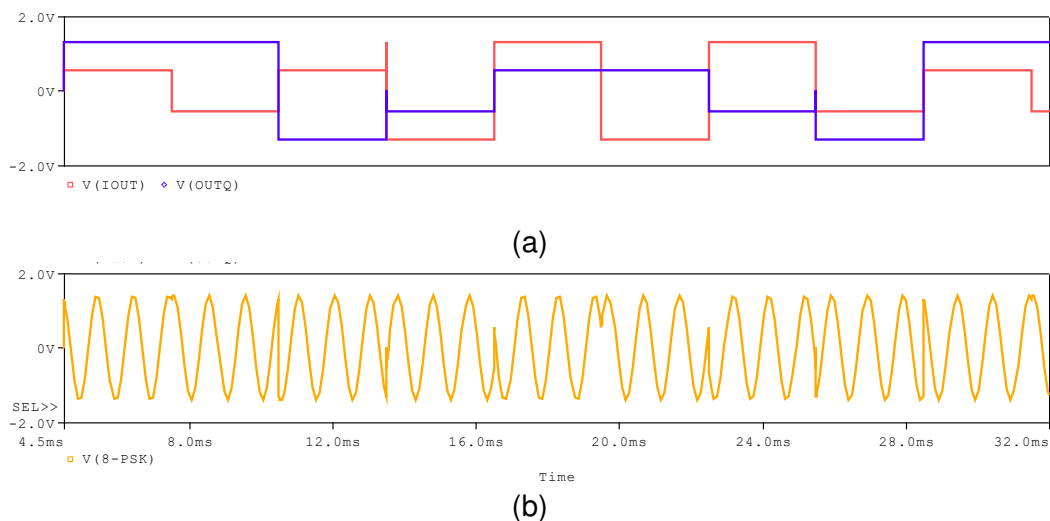


Figura 2.116 (a) Señal a la salida del conversor de 2 a 4 niveles (rojo-canal I); (azul-canal Q) (b) Modulación 8-PSK

Los resultados de la modulación BPSK, QPSK, y 8PSK, se analizarán en el siguiente capítulo.

2.3.4 Práctica 08: Modulación QAM

2.3.4.1 Objetivos

- Comprender el funcionamiento de la técnica de modulación QAM.
- Analizar resultados de simulación de un esquema planteado de modulación QAM utilizando la herramienta OrCAD PSpice.

- Conocer las ventajas y desventajas que brinda esta técnica de modulación digital en aplicaciones de ingeniería.
- Establecer la relación que tiene con la técnica de modulación PSK.

2.3.4.2 Marco teórico

QAM (*Quadrature Amplitude Modulation*), es un tipo de modulación digital que combina la modulación PSK y la modulación ASK, es decir la variación de la amplitud y la fase de la señal portadora, ver Figura 2.117.

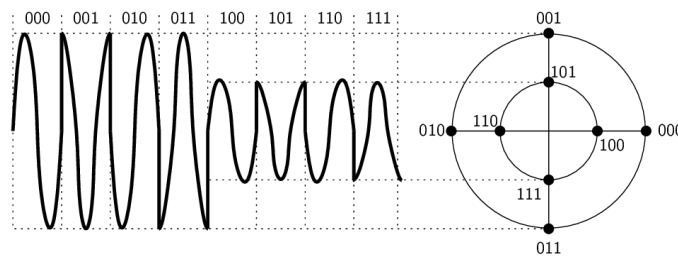


Figura 2.117 Modulación 8-QAM y su diagrama de constelación

Matemáticamente se describe la modulación QAM como la ecuación 2.45.

$$QAM(t) = V_i * \cos (w_c t + \theta_i)$$

Ecuación 2.45 Expresión matemática de la modulación QAM [13]

Donde:

- $V_i \rightarrow$ Cambios de amplitud de la portadora modulada [Voltios]
- $\theta_i \rightarrow$ Cambios de fase de la portadora modulada
- $w_c = 2 * \pi * f_c \rightarrow$ frecuencia de la portadora en [rad/s]

El funcionamiento consiste en sumar y modular por desplazamiento de amplitud (ASK) dos señales portadoras de igual frecuencia de forma independiente ($I(t)$, $Q(t)$), con un desfase entre sí de 90° (canales ortogonales).

Este tipo de modulación también se describe para múltiples estados M-QAM, y sus nombres se designan de acuerdo al número de niveles que representan, donde M puede tomar valores desde 8 hasta 1024, como 8-QAM, 16-QAM, 32-QAM, etc.

Al igual que la modulación M-PSK, se considera y se aplica la misma relación de la velocidad de transmisión con la velocidad de modulación de acuerdo a la ecuación 2.42; y, de la misma manera, para el cálculo del ancho de banda mínimo necesario de M-QAM, ecuación 2.44.

La representación en el diagrama de constelaciones de M-QAM tiene mejor comportamiento que M-PSK, ya que los puntos cuentan con mayor separación, como se muestra en la Figura 2.117.

El funcionamiento destaca ciertas ventajas tales como:

- Posibilidad de transmitir dos señales usando una sola frecuencia de manera que favorece el aprovechamiento del ancho de banda disponible.
- Cuenta con mayor inmunidad al ruido ya que la representación de los estados en el diagrama de constelaciones cuenta con mayor separación con respecto a PSK.
- Uso eficiente de energía y por ende reducción de costos.
- Mayor calidad y seguridad de transmisión.
- Aplicaciones en comunicaciones de radio, televisión digital, tecnología inalámbrica y celular.

Esta modulación tiende a ser multiestados, por lo que también presenta sus desventajas:

- Aumento de sensibilidad al ruido, entre mayor sea el número de bits a transmitir.
- Los receptores requieren demoduladores síncronos.

Modulador 8-QAM

La Figura 2.118 muestra el diagrama de bloques de un modulador 8-QAM, donde se puede observar que cuenta con cierta similitud con el diagrama de bloques del modulador 8-PSK simulado en la práctica anterior, la diferencia que presenta es que no tiene el inversor entre el canal C y Q.

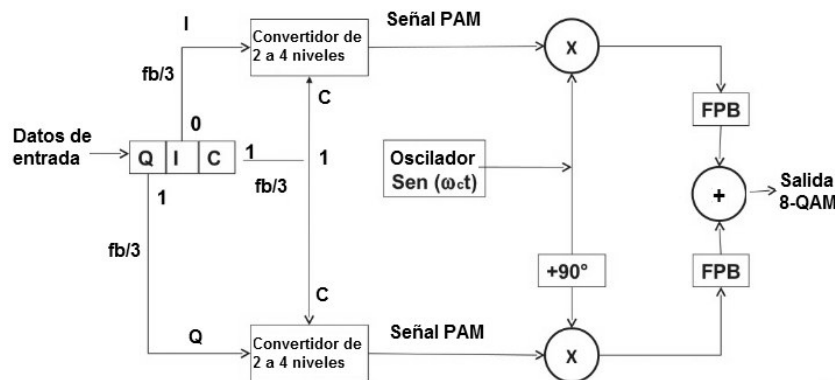


Figura 2.118 Modulador 8-QAM

El funcionamiento al igual que el modulador 8-PSK, consiste en que los datos de entrada digitales que ingresan se dividen en grupos de 3 bits usando el canal Q, I y C, con una velocidad equivalente a la tercera parte de la velocidad de datos de entrada.

Los bits I y Q indican la polaridad de la señal y el bit C determina la magnitud. Como este modulador no presenta inversión de canal C, los resultados obtenidos de los convertidores serán iguales, los cuales se combinarán con la portadora en fase y cuadratura correspondientemente, que finalmente se sumarán y darán lugar a la modulación de la señal.

Tomando como referencia la tabla de verdad del convertidor del modular 8-PSK con las amplitudes calculadas en la Tabla 2.14, se establece la Tabla 2.16 considerado solo para un convertidor de 2 a 4.

Tabla 2.16 Tabla de verdad convertidor 2 a 4

I/Q	C	Salida
0	0	-0.383
0	1	-0.924
1	0	+0.383
1	1	+0.924

Por lo tanto a la salida del modulador 8-QAM de la Figura 2.118; para las posibles 8 combinaciones se tiene lo siguiente, Tabla 2.17.

Tabla 2.17 Tabla de verdad, Modulación 8-QAM

Entrada binaria			Salida 8-QAM			
Q	I	C	Coeficientes en cuadratura $\sin(wt)$	Coeficientes en fase $\cos(wt)$	AMPLITUD	FASE (arco tangente)
0	0	0	-0.383	-0.383	0.383	-135
0	0	1	-0.924	-0.924	0.924	-135
0	1	0	-0.383	0.383	0.383	-45
0	1	1	-0.924	0.924	0.924	-45
1	0	0	+0.383	-0.383	0.383	+135
1	0	1	+0.924	-0.924	0.924	+135
1	1	0	0.383	0.383	0.383	+45
1	1	1	0.924	0.924	0.924	+45

En base a la Tabla 2.16, se determina el diagrama de constelaciones que se muestra en la Figura 2.119.

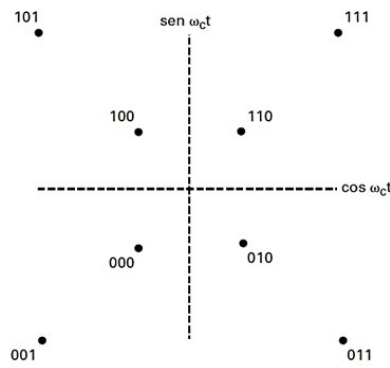


Figura 2.119 Modulador 8-QAM

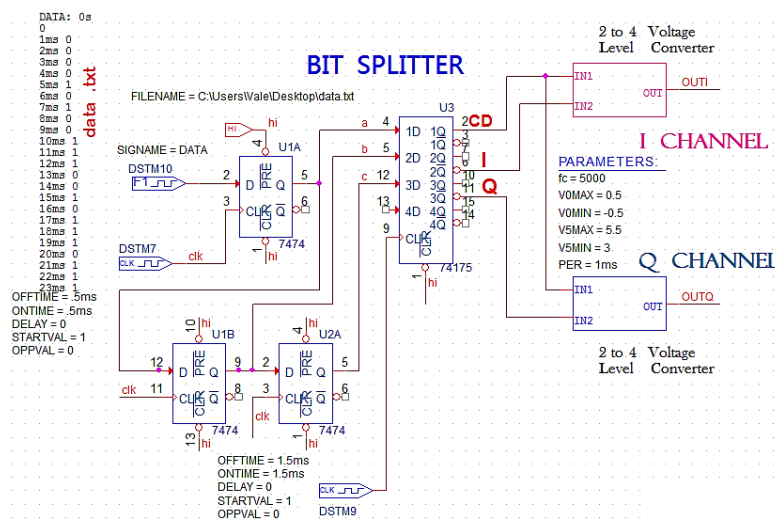
2.3.4.3 Desarrollo práctico

En esta práctica se propone esquematizar y simular un modulador 8-QAM utilizando la herramienta OrCAD PSpice. El esquema permitirá analizar el funcionamiento de acuerdo a la descripción que se presenta en el marco teórico de esta práctica.

La esquematización debe componerse de los elementos necesarios para producir la modulación 8-QAM de los datos digitales ingresados, y permitirá el análisis de la señal modulada y verificar el ancho de banda a partir del espectro de frecuencias obtenido, además permitirá establecer un escenario comparativo con respecto a la modulación PSK.

Esquematización de un modulador 8-QAM

La Figura 2.120 ilustra el esquema de modulación simulado de acuerdo al diagrama de bloques descrito en el marco teórico, este esquema cuenta con los elementos necesarios para la modulación 8-QAM, el cual opera con ciertos parámetros que se especifican en el mismo circuito como es la frecuencia de la portadora 3 [KHz]



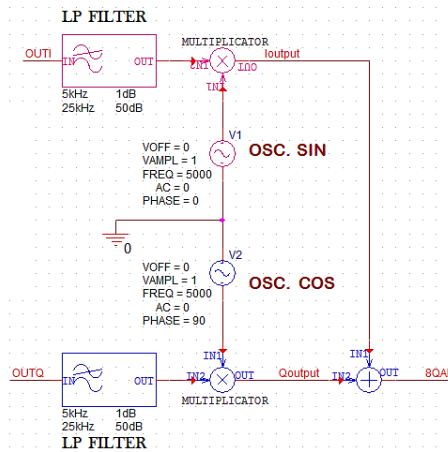


Figura 2.120 Esquema de modulación 8-QAM

El modulador consta de un primer bloque que conforma un divisor de bits o *Bit Splitter*, cuya función es convertir la serie de datos de entrada a datos en paralelo, para lo cual se dividen en tres canales cuya tasa de bit es equivalente a la tercera parte de la tasa de bits de entrada al modulador.

Seguidamente se utiliza los bloques de OrCAD PSpice ABM2 (*2 to 4 Voltaje Level Converter*), los cuales se configurarán de tal modo que cumplan la función de convertidores de 2 a 4 niveles colocados en cada canal I y Q. Los convertidores de 2 a 4 niveles de voltaje se encargan de convertir la salida de datos del canal I (salida de datos I y C) y el canal Q (salida de datos Q y C) a sus correspondientes voltajes analógicos. La Tabla 2.16 muestra, que para las cuatro posibles combinaciones de bits implica 4 niveles de voltaje.

Los datos de los canales Q, I y C, que ingresan y salen por los convertidores de dos a cuatro niveles, se muestran en la Figura 2.121.

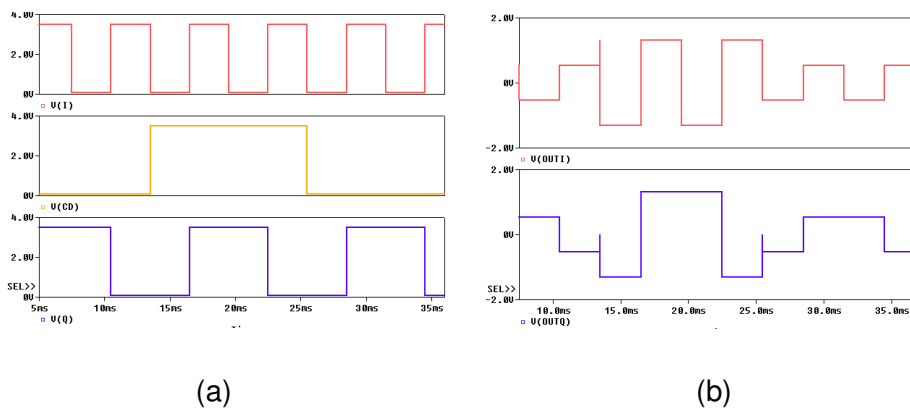


Figura 2.121 Señales a la (a) entrada, (b) salida del conversor de 2 a 4 niveles

La configuración de los elementos convertidores de nivel, se realizará en base a los datos de la Tabla 2.17, de la misma forma que se realizó en la modulación 8-PSK, ver Figura 2.115.

La señal portadora de 3 [KHz] que interviene en la modulación y su respectivo desfase se presenta en la Figura 2.122.

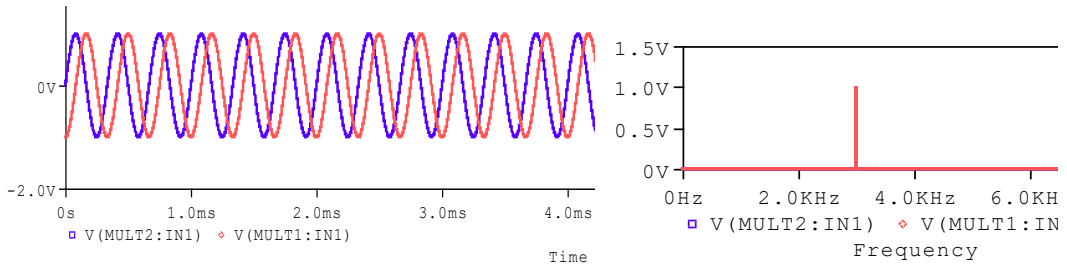
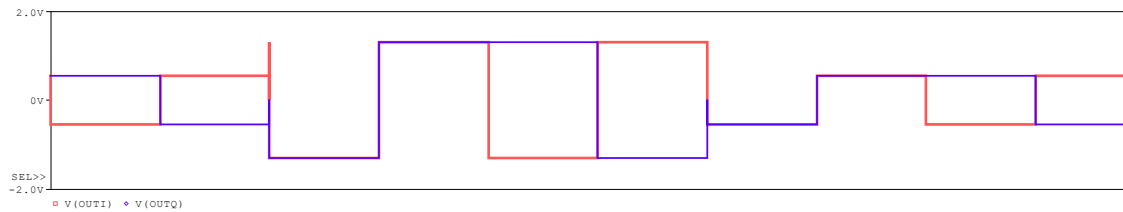
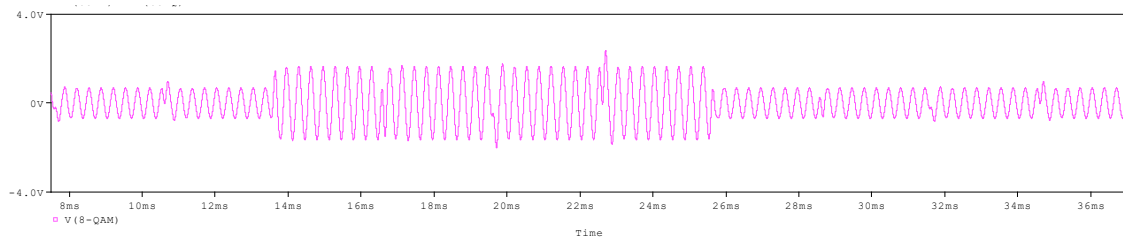


Figura 2.122 Señal portadora (rojo - canal I; azul – canal Q); y su respectivo espectro de frecuencia

En la Figura 2.123 se muestran los resultados obtenidos de la modulación 8-QAM donde se verifica los cambios de fase y de amplitud al pasar de una condición lógica a otra.



(a)



(b)

Figura 2.123 (a) Salida del canal I(rojo) y canal Q(azul), (b) Modulación 8-QAM

La Figura 2.124 muestra el espectro de frecuencia de la señal modulada.

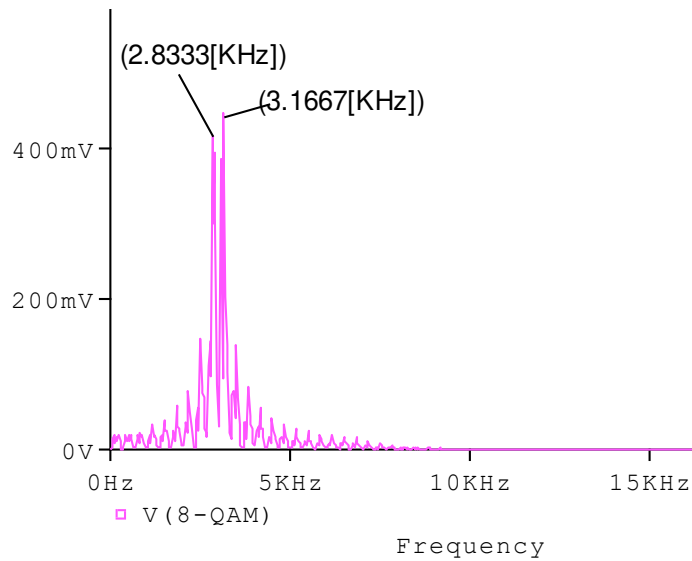


Figura 2.124 Espectro de frecuencia de la señal modulada

El ancho de banda práctico equivale a:

$$AB_{8-QAM} = 3.1667 [KHz] - 2.8333 [KHz] = 333,4 [Hz]$$

Los resultados de la modulación 8 QAM se analizarán en el siguiente capítulo.

2.4 Sistema de transmisión multiportadora usando COFDM

2.4.1 Esquema de funcionamiento del transmisor COFDM [41] [42]

En la Figura 2.125, muestra el modelo simplificado de un transmisor COFDM cuyo objetivo es la transmisión de bits de datos los cuales ingresan al codificador convolucional FEC el cual se encargará de la corrección de errores en el demodulador por lo cual se encargará de dar redundancia a los datos transmitidos. En el entrelazador de frecuencia se agrupan varios bits para causar un desorden aleatorio para conseguir distribuir los bits de datos entre portadoras dentro de un símbolo OFDM para mejorar el funcionamiento del decodificador de Viterbi en el demodulador COFDM. Luego los datos ingresan al modulador OFDM el cual toma pares de bits para modularlo con QPSK sucesivamente cambiando de bits a símbolos de datos, después se realiza la IFFT que agrupará 8 símbolos que corresponde a 8 puntos que va procesar la IFFT a la vez para luego generar un símbolo útil formado por 8 muestras con duración $T\mu$ constituido por 8 portadoras ortogonales entre sí (separadas en frecuencias en múltiplos de $1/T\mu$) y finalmente se inserta un intervalo de guarda cuya función será eliminar el ISI, es decir IFFT genera un

símbolo que es llamado símbolo OFDM. El cual está formado por un símbolo útil y un intervalo de guarda.

La señal a transmitir se debe ajustar debido que está en banda base, la salida Q-I es todavía digital para ello es necesario convertirla a una señal analógica usando dos conversores digital-analógica (DAC), para luego ser modulada en RF usando un modulador en cuadratura. La señal de salida COFDM, Figura 2.126, se aprecia que hay 8 portadoras que son ortogonales entre sí, que están centradas a una frecuencia central en un ancho de banda (BW), donde una trama OFDM está constituida por 4 símbolos OFDM generados por la IFFT para este ejemplo.

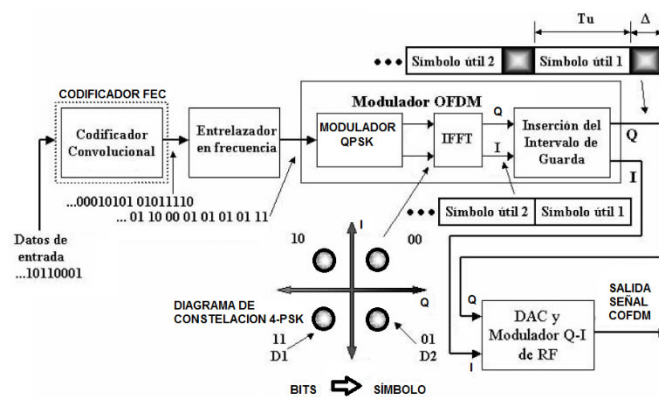


Figura 2.125 Transmisor COFDM [41]

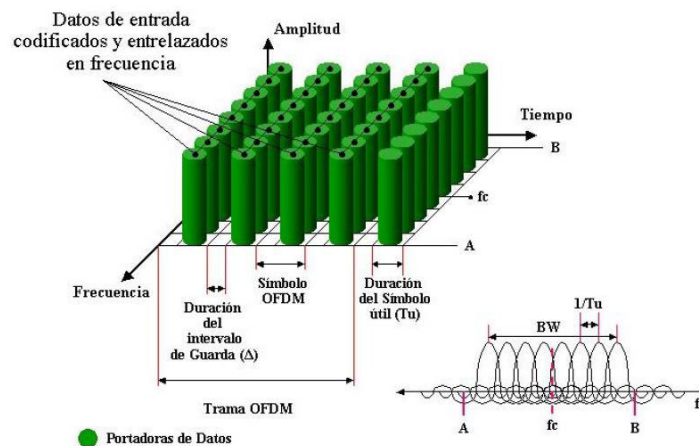


Figura 2.126 Señal COFDM de salida [41]

El ancho de banda (BW) es el resultado de dividir el número de portadoras (N_p) y la duración útil (T_μ), ecuación 2.46.

$$BW = N_p / T_\mu \text{ [Hz]}$$

Ecuación 2.46 Ancho de banda de COFDM [42]

La tasa binaria a transmitir (T_b) es, ecuación 2.47:

$$T_b = \frac{N_P}{T_{\mu} + \Delta} \times N_{bm} \times T_c \quad [bps]$$

Ecuación 2.47 Velocidad de transmisión [42]

Donde:

- N_P = número de portadoras de datos
- N_{bm} = número de bits por símbolo del esquema de modulación
- T_{μ} = duración del símbolo útil
- T_c = tasa del codificador FEC
- Δ = duración del intervalo de guarda.

2.4.2 Esquema de funcionamiento de un receptor COFDM

En la Figura 2.127 se muestra el modelo simplificado de un receptor COFDM, se asume que la señal transmitida no sufre de ningún tipo de degradación o interferencia y vuelve en fase al receptor para así obviar la parte de sincronización en el receptor y ecualización.

La señal recibida ingresa al demodulador de RF y las salidas Q-I analógicas son convertidas a Q-I digitales usando dos conversores analógico – digitales para ello se hace un muestreo de la señal COFDM con un periodo de muestreo T_{μ}/N_P , luego ingresan las señales Q-I en banda base al demodulador OFDM, el cual tiene como función eliminar el intervalo de guarda con lo que se elimina las primeras muestras que contienen el intervalo de guarda, después se realiza FFT donde se agrupan 8 muestras que va a procesar para entregar 8 símbolos que van a ser demodulados cada uno finalmente por el demodulador QPSK pasando de símbolo a bits. Después varios bits se agrupan en el desentrelazador en frecuencia para reordenarlos y pasarlos al decodificador de Viterbi que tiene como función habilitar la corrección de errores en el receptor, recuperando así los bits de datos transmitidos.

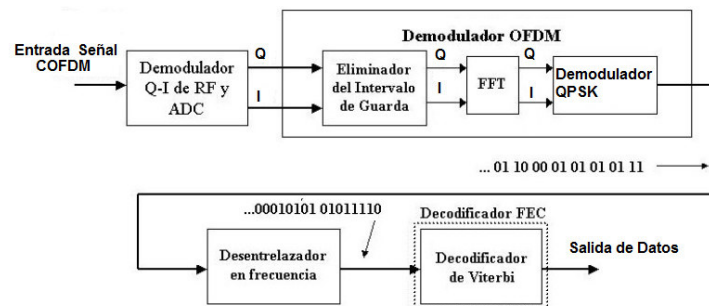
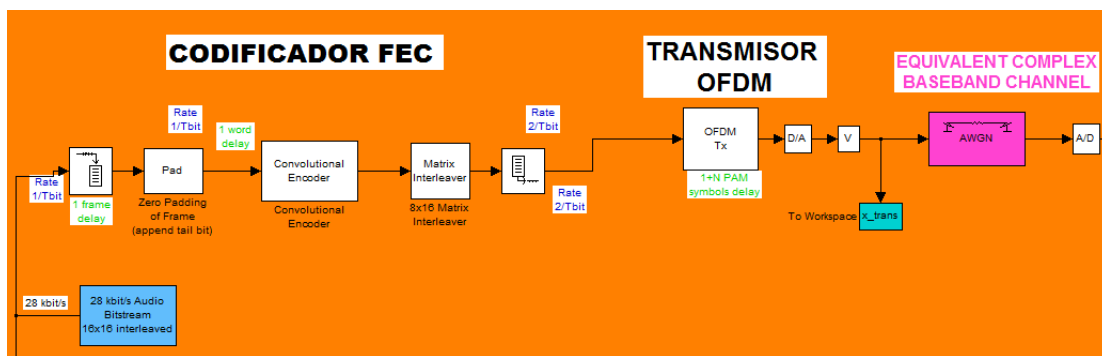


Figura 2.127 Receptor COFDM

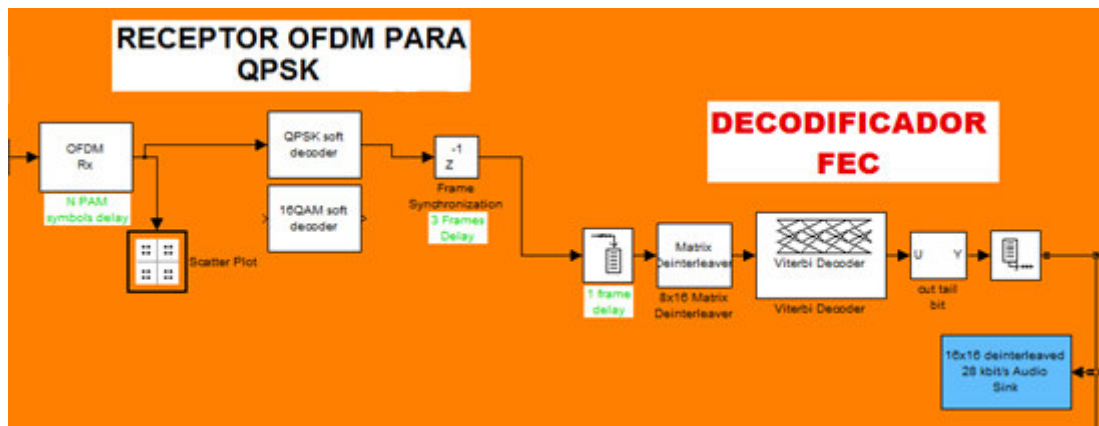
2.4.3 Esquema de simulación en Simulink del sistema COFDM

En base al diagrama de bloques del transmisor y el receptor COFDM descrito en la parte teórica de esta sección se propone realizar un sistema de transmisión multiportadora donde la información a transmitir se considera un intervalo de audio.

De acuerdo al funcionamiento descrito en la teoría de la Figura 2.125 y Figura 2.127 se realiza la esquematización de la Figura 2.128 utilizando la herramienta Simulink la cual cuenta con los bloques necesarios para llevar a cabo el proceso de transmisión y recepción usando COFDM.



(a)



(b)

Figura 2.128 Sistema de modulación usando COFDM

Ejecutando la simulación se tiene la transmisión y la recepción de un intervalo de audio de 0.03 segundos probado, cuyo resultado se verifica en las capturas de pantalla de las Figuras 2.129 y 2.130.

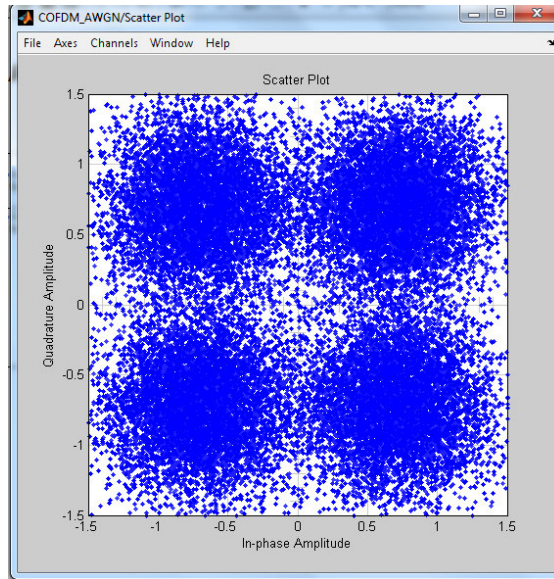


Figura 2.129 Datos de audio de entrada modulados en 4- PSK

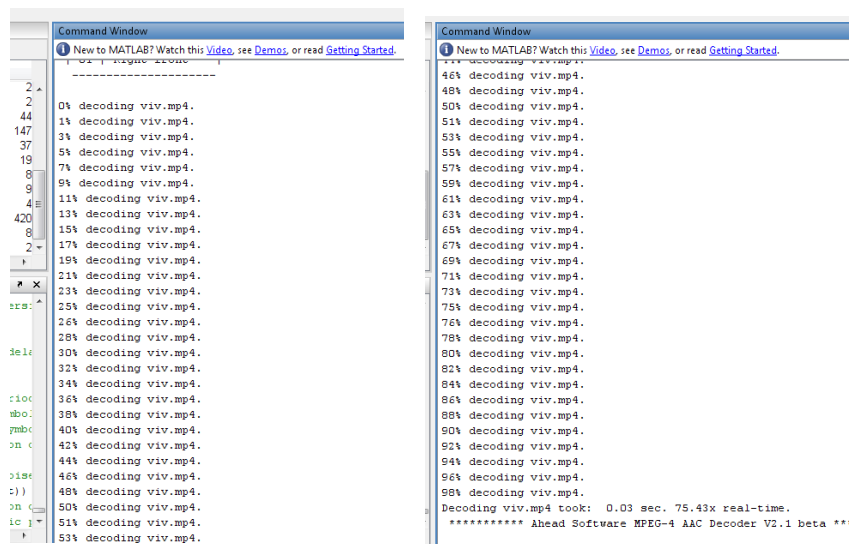


Figura 2.130 Decodificación de datos para obtención de audio ingresado

El script que describe el esquema se puede verificar en el Anexo V.

3. RESULTADOS Y DISCUSIÓN

En la presente sección se discutirán los resultados obtenidos de los escenarios desarrollados en cada una de las prácticas del capítulo anterior.

3.1 Análisis de resultados sobre introducción al procesamiento digital de señales

El análisis y desarrollo matemático para determinar la salida de un sistema LTI, puede llegar a ser un método muy largo el cual demanda tiempo su respectivo desarrollo para su análisis, por lo cual en esta práctica se ha introducido el uso de la herramienta Matlab de tal manera que se ha creado un script el cual permitirá el análisis respectivo a la salida de un sistema, en el cual interviene una señal de entrada con una respuesta al impulso.

A través del desarrollo matemático se determina la respuesta respectiva a la salida del sistema LTI, por medio del análisis de los tres casos en la ecuación respectiva de la respuesta del sistema LTI en donde los valores de n y k pueden variar en el término $u(n-k)$, obteniendo de esta manera la respuesta del sistema como la suma de las respuestas de los tres casos como $y(n)$.

Como segundo escenario, usando la herramienta Matlab e introduciendo el concepto de convolución lineal se verifica los resultados obtenidos en el análisis matemático, considerando la respuesta impulsiva exponencial y la secuencia de entrada del pulso rectangular, en donde se visualiza la respuesta obtenida matemáticamente, esta forma gráfica en el análisis es más efectiva ya que se facilita la interpretación de resultados.

Cambiando de escenario en donde utilizando el mismo *script* de Matlab se varía la respuesta impulsiva al de un pulso triangular y con la misma secuencia de entrada de un pulso rectangular, se tiene a la salida del sistema una forma de onda similar al lóbulo principal de una señal *sampling*, Figura 2.37. En este punto se hace referencia que si se aplica la transformada de Fourier a un pulso triangular, a la salida se tiene la forma de una *sampling*, pero en este caso se habla de espectro de señal.

En la convolución discreta el parámetro importante a considerar es el número de muestras mínimo requerido para obtener las respuestas, por lo que se ha tomado en cuenta que por lo menos debe ser igual a la longitud de la señal de entrada más la longitud de respuesta impulso menos uno, esto se realiza con el fin de evitar cálculos innecesarios donde no hay datos. La aplicación de Matlab en esta práctica resulta eficaz ya que se puede visualizar

los resultados de forma gráfica, facilitando de esta manera el análisis que implica cuando se cambia de tipo de señales.

Como análisis adicionales se podrá ejecutar el script modificando los parámetros de la respuesta impulsiva o también la secuencia de entrada, lo que se deja abierto al criterio de quien haga uso de esta práctica.

3.2 Análisis de resultados sobre análisis de señales (Series y Transformada de Fourier)

De acuerdo al primer escenario (generador de tren de pulsos), se ha considerado el parámetro ancho de pulso como PW y un período como P; PW del tren de pulsos tiene un determinado tiempo de duración al cual se le considera como T_b , es así que a medida que este tiempo va disminuyendo la velocidad de transmisión va aumentando ($V_{tx}=1/T_b$), se puede decir que la duración del tiempo de pulso es inverso a la velocidad de transmisión, lo cual también se puede determinar paralelamente en que a medida que se va estrechando el pulso, el lóbulo principal espectral va aumentando y consigo también va disminuyendo la componente DC, esto quiere decir que al aumentar la velocidad de transmisión los requerimientos de ancho de banda también aumentan.

Una de las desventajas de la transmisión de señales digitales es que se requiere un ancho de banda infinito, y en la práctica esto no es posible conseguir, es por eso que para evitar degradaciones de señal como es el ISI y baja calidad del sistema, se usan filtros tipo pasa bajo para limitar el uso de ancho de banda.

Este análisis muestra que por medio del sintetizador de onda cuadrada y usando análisis de señales se puede minimizar los inconvenientes de ISI y ancho de banda infinito; limitando el ancho de banda con un filtro tipo pasa bajo, como se ilustra en la Figura 2.51.

La respuesta del circuito implementado en la Figura 2.48, muestra la reconstrucción de la onda cuadrada, definiendo los pulsos de manera que exista interferencia controlada, y de esta forma el receptor pueda distinguir cada pulso, la ventaja que se consigue es que se ocupará menor ancho de banda, pero consumirá más potencia.

Además, para la reconstrucción de la señal se consideró solamente ocho armónicos, pero para precisar de mejor manera la forma de onda cuadrada se puede considerar más generadores adicionales al circuito, permitiendo enfatizar que esta señal cuadrada puede estar formada por infinitas componentes de frecuencia implicando también un ancho de banda infinito.

3.3 Análisis de resultados sobre diseño y simulación de un filtro FIR

El script diseñado para determinar los resultados que presenta la práctica 03, puede ser modificado para ser ejecutado con otro tipo de método o para variar el tipo de filtro, tan solo incorporando a la línea de código del script los comandos necesarios para poder ejecutar lo que se requiera implementar, cabe recalcar que no es necesario borrar las líneas innecesarias ya que con ayuda de comentarios anteponiendo a la línea de código el signo de porcentaje (%) se puede mantener en el programa sin generar ningún tipo de error, de esta manera también se puede crear un código robusto multifuncional y aplicable cuando sea necesario en cualquier práctica.

A modo de ejercicio se modificó el programa para diseñar un filtro pasa bajo por el método de Bartlett, para este ejercicio se consideran los siguientes parámetros:

- $ws1=0.4\pi$, $ws2=0.7\pi$, $wp1=0.5\pi$, $wp2=0.6\pi$, $Rp=0.5$, $As=40$

En este caso no es necesario cambiar la línea de código para el cálculo del ancho del lóbulo mínimo de transición, ya que si se observa en la Tabla 2.8 se aplica el mismo criterio que en el método ventana de Hamming, obteniendo de esta manera el orden del filtro $M=67$. La respuesta impulso que se desea diseñar es para el filtro pasa banda por lo cual se mantiene la misma línea de código, que resulta lo que se ilustra en la Figura 3.1.

Este resultado obtenido se multiplica con la función ventana de Bartlett, Figura 3.2, la cual tiene una forma triangular para obtener la respuesta impulso del filtro diseñado, Figura 3.3, y luego a partir de este la respuesta en frecuencia, Figura 3.4. Con ayuda de los comandos de Matlab al aplicar un método de diseño resulta fácil el cambio ya, que solo se trata de comandos más no el ingreso de una ecuación característica a la función.

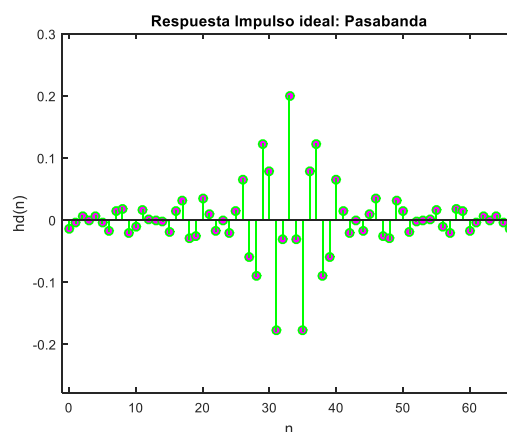


Figura 3.1 Respuesta impulso que se desea diseñar

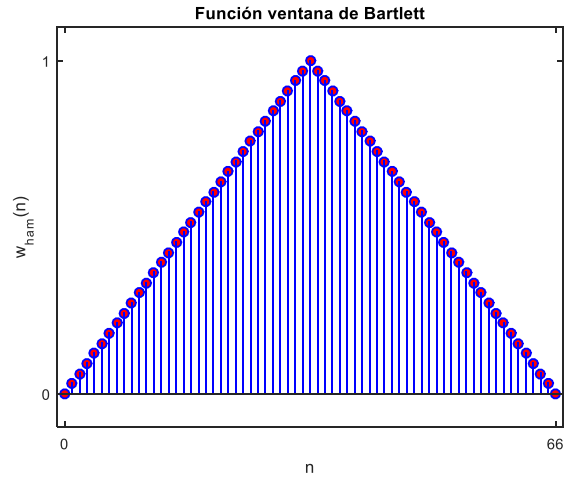


Figura 3.2 Función ventana de Bartlett, $M=67$

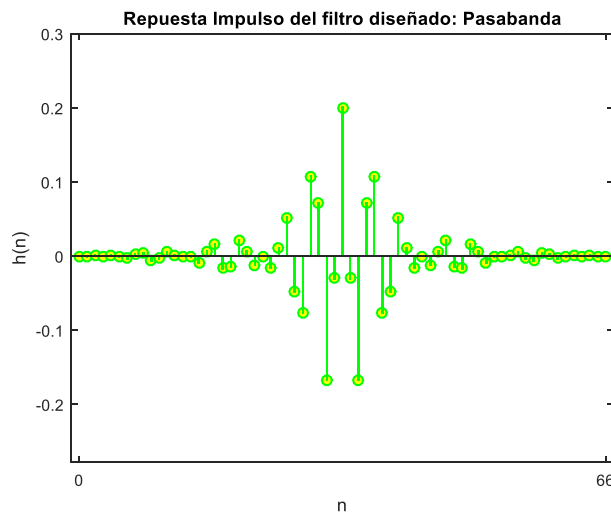


Figura 3.3 Respuesta impulso del filtro diseñado

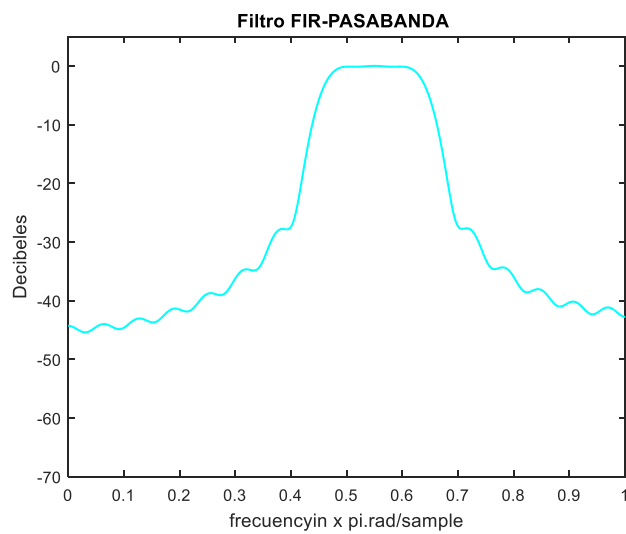


Figura 3.4 Respuesta en frecuencia del filtro diseñado

Este script resulta útil para que pueda ser usado en múltiples aplicaciones de acuerdo al criterio y parámetros requeridos, realizando los cambios necesarios.

Este tipo de filtro permite el paso frecuencial de aquellas bandas que se encuentran en el rango $[wp1 \text{ } wp2]$, en otras palabras definen el ancho de banda del filtro. Las frecuencias $ws1$ y $ws2$ definen la banda de frecuencia suprimida.

3.4 Análisis de resultados sobre diseño y simulación de un filtro IIR

El script diseñado presenta los resultados de esta práctica, la misma que puede variar de acuerdo a los parámetros que se requiera probar. Dado que se trata de un script, cuenta con la ventaja de que se puede reutilizar sus líneas de comandos en otros programas o a su vez fortalecer el programa de tal manera que al aumentar líneas de código se consiga un programa multifuncional para ciertos escenarios o para aplicar a la práctica necesaria.

De acuerdo a las especificaciones iniciales del filtro digital, el programa se encarga de realizar los cálculos necesarios para determinar las especificaciones correspondientes para el diseño del filtro analógico usando uno de los prototipos posibles, en este caso se usa el elíptico. Para este proceso se considera un valor de $T=2$.

Los comandos de *Matlab* utilizados para determinar la respuesta son *bilinear* y *ellip*, El comando *bilinear* permite obtener la conversión de filtro analógico a digital, mientras que el comando *ellip* permite obtener el diseño del filtro a partir de sus parámetros requeridos.

El usuario puede ingresar los parámetros que necesite verificar los resultados, en esta práctica para análisis se consideró los siguientes datos:

- (VARIACION 1) $ws=0.4\pi$, $wp=0.2\pi$, $Rp=1$, $As=70$

Pero con el fin de verificar el funcionamiento del script se variará los parámetros iniciales para observar los resultados, para lo cual se considera:

- (VARIACION 2) $ws=0.8\pi$, $wp=0.3\pi$, $Rp=1$, $As=50$

Los resultados que se obtienen de los primeros parámetros planteados se observan en la Figura 3.5 y su respectiva respuesta al impulso en la Figura 3.6.

Los resultados que se obtienen de la variación de los parámetros se observan en la Figura 3.7 y su respectiva respuesta al impulso en la Figura 3.8.

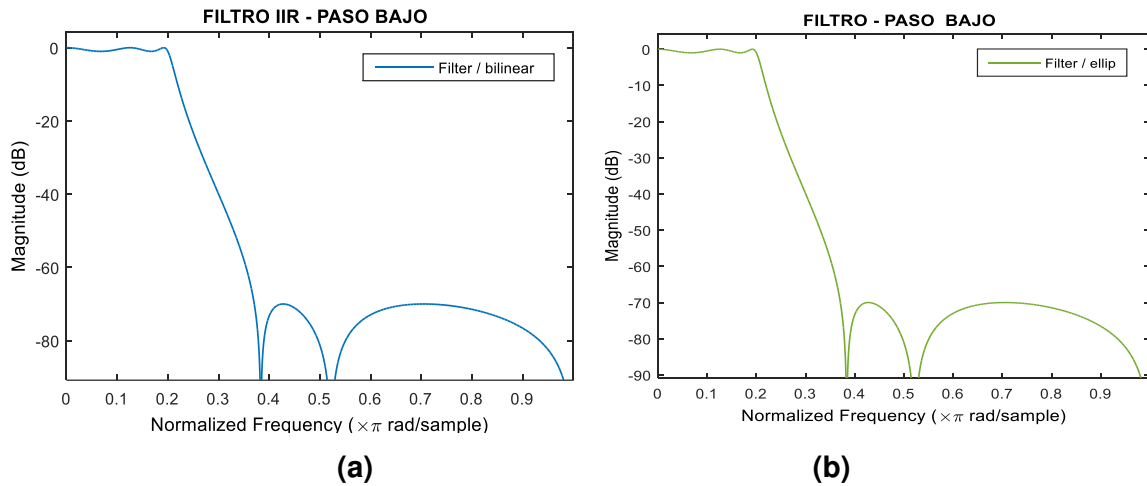


Figura 3.5 Respuesta en frecuencia del filtro diseñado
 (a) usando comando bilineal (b) usando ellip – (para datos variación 1)

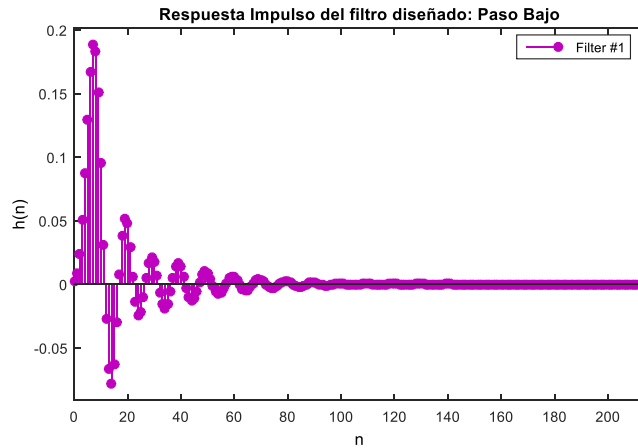


Figura 3.6 Respuesta Impulso – método bilineal (para datos variación 1)

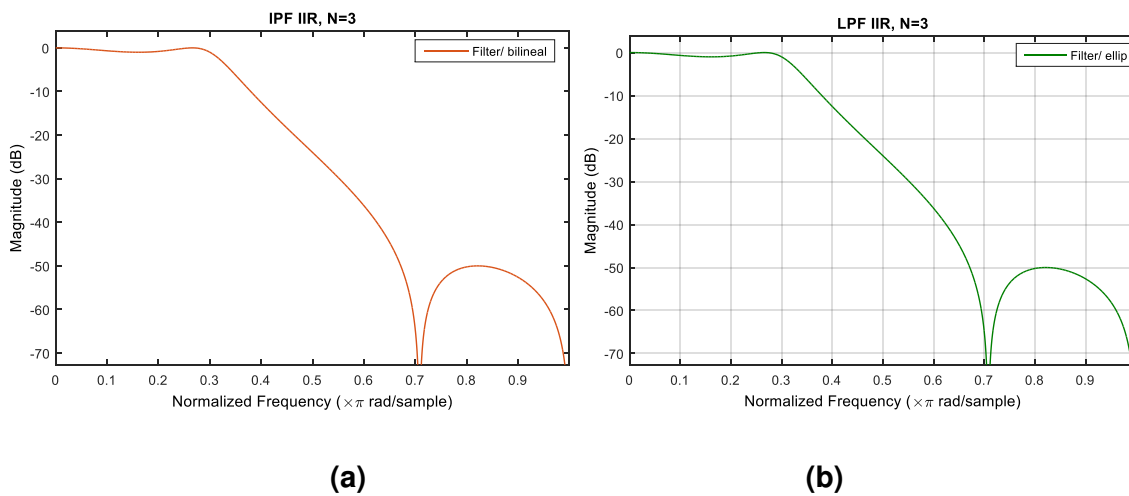


Figura 3.7 Respuesta en frecuencia del filtro diseñado
 (a) usando comando bilineal (b) usando ellip (para datos variación 2)

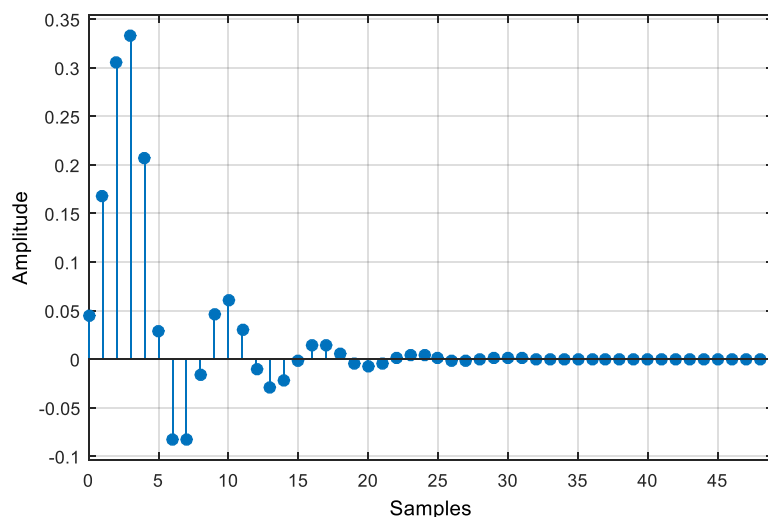


Figura 3.8 Respuesta Impulso – método bilineal (para datos variación 2)

Como se observa en las gráficas obtenidas y usando los comandos mencionados, resulta el mismo diseño usando los dos comandos, tanto para los primeros parámetros como para los segundos, con la diferencia que una de sus respuestas impulso no corresponde al prototipo de filtro analógico.

3.5 Análisis de resultados sobre modulación y demodulación ASK

Análisis del modulador en OrCAD PSpice

En base a la ecuación que describe la modulación ASK se determina que a medida que se va variando el índice de modulación (k), también va variando la profundidad o grado de modulación, así:

- Si se considera el índice de modulación 0.1 se puede evidenciar que cuando se modula un 1L este toma un determinado valor de amplitud y cuando se modula un 0L este toma un valor de amplitud menor al de 1L, este tipo de modulación es de dos niveles como se muestra en la Figura 2.73(c).
- Si se considera el índice de modulación 0.3 se puede evidenciar que cuando se modula un 0L el nivel de modulación se atenúa siendo este un nivel de amplitud mucho menor al de 1L, como se muestra en la Figura 2.75(c).
- Si se considera el índice de modulación 1, la profundidad de modulación es al 100% ya que la portadora en 0L se atenúa totalmente quedándose en nivel cero, justificando por medio del análisis matemático donde se puede determinar que la portadora se mantiene activa (on) cuando ingresa un 1L y la portadora se apaga (off) cuando ingresa un 0L, por lo tanto, en base a este análisis se determina que

se trata de una modulación OOK. Adicional también se puede decir que en este tipo de modulación las envolventes nunca se cruzan es decir no se sobremodula.

El análisis matemático correspondiente a la modulación OOK, se detalla a partir de la ecuación 2.30, con $k=1$, se tiene:

$$ASK(t) = [1 + k \cdot b_n(t)] \cos(w_c t)$$

- Cuando se trata de 1L, considerando una amplitud de la modulante de valor 1:
 $ASK(t) = [1 + 1] \cos(w_c t)$, indica presencia de portadora.
- Cuando se trata de 0L, considerando una amplitud de la modulante de valor -1:
 $ASK(t) = [1 + 1(-1)] \cos(w_c t)$
 $ASK(t) = [1 - 1] \cos(w_c t)$
 $ASK(t) = 0$, indica ausencia de portadora

Entonces:

- De la señal de datos de Figura 2.72(a), se puede verificar analíticamente lo siguiente:

$$Tb = 1 [ms] \rightarrow \text{Tiempo de duración de bit}$$

$$T = 5 [ms] \rightarrow \text{Periodo de la señal de datos}$$

$$v_{tx} = \frac{1}{Tb} = \frac{1}{1 [ms]} = 1000 [bps] \rightarrow \text{velocidad de transmisión}$$

$$f_0 = \frac{1}{T} = \frac{1}{5 [ms]} = 200 [Hz] \rightarrow \text{frecuencia de la señal de datos}$$

- En la señal portadora, Figura 2.72(b)

$$f_c = 5 [KHz]$$

En la modulación B-ASK se puede verificar que la velocidad de transmisión es igual a la velocidad de la señal modulada, por lo tanto se tiene de acuerdo a la Figura 2.73 (b):

$$Ts = Tb = 1 [ms] \rightarrow \text{Tiempo de duración de símbolo}$$

$$v_m = \frac{1}{Ts} = \frac{1}{1 [ms]} = 1000 [baudios] \rightarrow \text{velocidad de modulación}$$

En el espectro de modulación se puede determinar que la frecuencia central es el de la señal portadora y las componentes siguientes son múltiplos de la frecuencia de datos. Donde se determina que el ancho de banda de la señal modulada es, Figura 2.74(b)

$$f_{min} = 5 [KHz] - 200 [Hz] = 4.8 [KHz]$$

$$f_{max} = 5 [KHz] + 200 [Hz] = 5,2 [KHz]$$

$$AB_{ASK} = (5.2 - 4.8) [KHz] = 400 [Hz]$$

Por lo tanto se determina que el ancho de banda requerido para la modulación 2ASK es el doble del ancho de banda de la señal en banda base.

Análisis del demodulador en Simulink

El proceso de demodulación no coherente que se realiza para determinar la señal de datos a la entrada del modulador ASK se observa en la Figura 2.82.

La señal de entrada al demodulador es del tipo OOK Figura 2.82(a), que después pasa por un rectificador de media onda, Figura 2.82(b), y luego por otro bloque que detecta la envolvente, Figura 2.82(c), después es filtrada y con ayuda de un conformador se tiene la señal demodulada Figura 2.82(d).

Comparando la señal modulada y la demodulada, se puede determinar que los datos son recuperados exitosamente, como se muestra en la Figura 3.9.

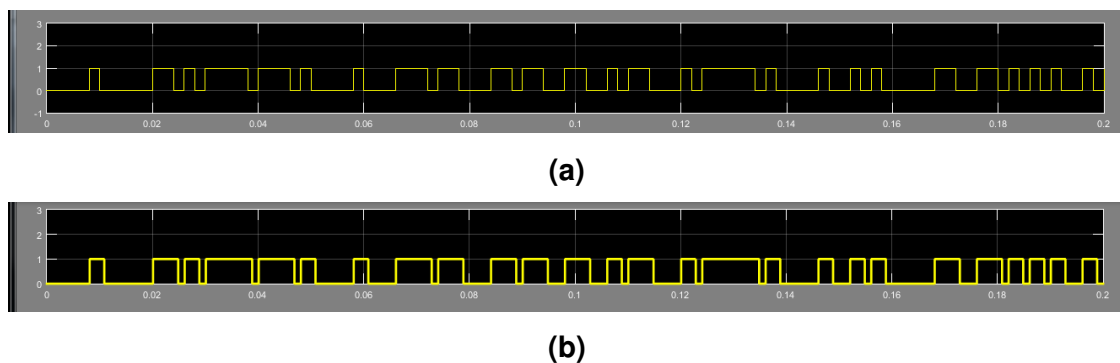


Figura 3.9 Señal de datos (a) a la entrada del modulador OOK (b) a la salida del demodulador OOK

Este tipo de modulación no tiene muchas aplicaciones ya que presenta problemas de distorsión. Sin embargo, para aplicaciones en conjunto con la modulación en frecuencia, resulta más eficaz.

También es usada para la transmisión de datos entre computadores utilizando bandas no comerciales (Banda ISM), las cuales están abiertas sin necesidad de licencia respetando las regulaciones que implican su uso.

3.6 Análisis de resultados sobre modulación y demodulación FSK

El esquema de modulación/demodulación presentado en la práctica 06, permite observar el proceso que se lleva al cabo para modular una señal de entrada binaria en FSK y de la misma manera demodular para obtener los datos ingresados.

Análisis de resultados del modulador FSK

Como resultado de la modulación FSK obtenida en la Figura 2.87(b), se presenta una señal de salida donde se verifica la variación de la frecuencia de acuerdo a los datos de entrada, además se nota que cuando se presenta la conmutación de frecuencia no se tiene saltos de fase, es decir es una modulación FSK de fase continua o FSK coherente.

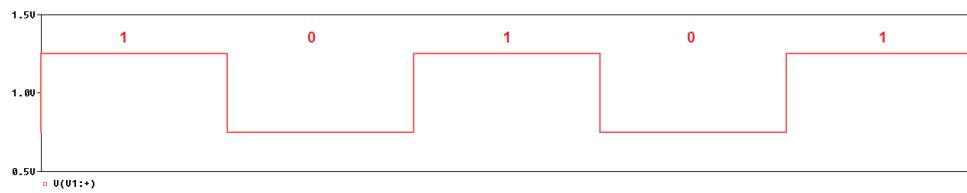
Es importante mencionar que para analizar la modulación de la señal de datos de entrada se debe establecer niveles de voltaje de la señal binaria ya que el modulador construido funciona de forma que al variar el voltaje de entrada al VCO a la salida se obtendrá una variación de frecuencia de oscilación, en otras palabras, la portadora varía en frecuencia de acuerdo a los niveles de voltaje de la señal de entrada.

Para determinar los niveles de voltaje de la señal de entrada en el esquema propuesto de esta práctica, se establece relaciones entre la frecuencia de espacio (0L) y la frecuencia de marca (1L) previamente calculadas (ecuación 2.35, ecuación 2.36) con la frecuencia de la portadora respectivamente, obteniendo de esta manera el valor de voltaje para el estado 0L y el nivel de voltaje para el estado 1L. Para el caso en que se tengan los datos de los niveles de la señal binaria se ingresará al esquema de modulación para verificar los resultados.

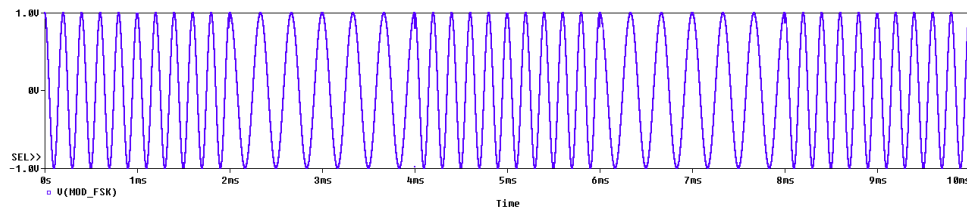
A modo de práctica, si se varía el valor de la desviación de frecuencia pico o la frecuencia de la portadora, los valores de la frecuencia de espacio y de marca cambian y por lo tanto también cambian los niveles de voltaje de la señal binaria de entrada.

Para verificar resultados se considera una desviación de frecuencia pico de 1 [KHz], una portadora de 4 [KHz]; que aplicando la ecuación 2.36 y la ecuación 2.35 se obtiene una frecuencia de espacio de 3 [KHz] y una frecuencia de marca de 5 [KHz] respectivamente y estableciendo las relaciones para determinar los niveles de voltaje para 0L se tiene un nivel equivalente a 0.75 [V] y para el 1L un nivel de 1.25 [V].

La modulación se observa en la Figura 3.10 y su respectivo espectro en la Figura 3.11.



(a)



(b)

Figura 3.10 (a) Señal de datos, (b) Modulación 2-FSK

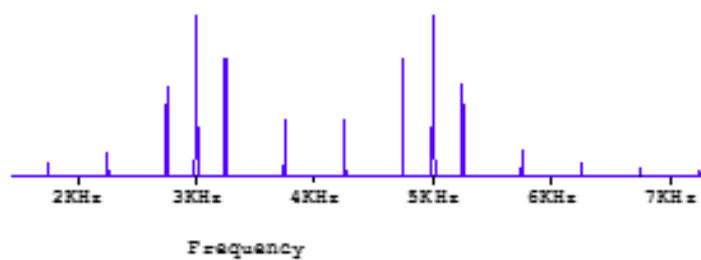


Figura 3.11 Espectro de frecuencia de una modulación FSK de una señal binaria

Para el análisis de resultados se usan frecuencias bajas ya que si se trabaja con frecuencias altas no es posible evidenciar resultados de la modulación en las figuras que se presentan, obviamente que en la práctica si es posible porque se puede hacer uso de la herramienta zoom para verificar resultados, es así que de acuerdo a lo anteriormente dicho; se considera frecuencias altas y los resultados obtenidos de la modulación en el mismo intervalo anterior se muestran en la Figura 3.12, considerando una portadora de 500 [KHz] y una variación de frecuencia pico de 10 [KHz].

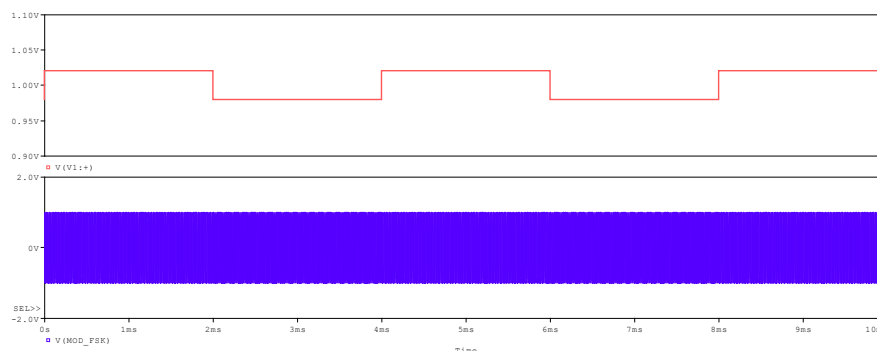


Figura 3.12 Modulación FSK de una señal binaria

Como se observa en la gráfica anterior, no es visible la modulación obtenida, el resultado es un barrido de color, a menos que se realice un *zoom* en la práctica, justo en la conmutación de frecuencia para verificar el cambio de estado o también se puede verificar el espectro de frecuencia como se ilustra en la Figura 3.13.

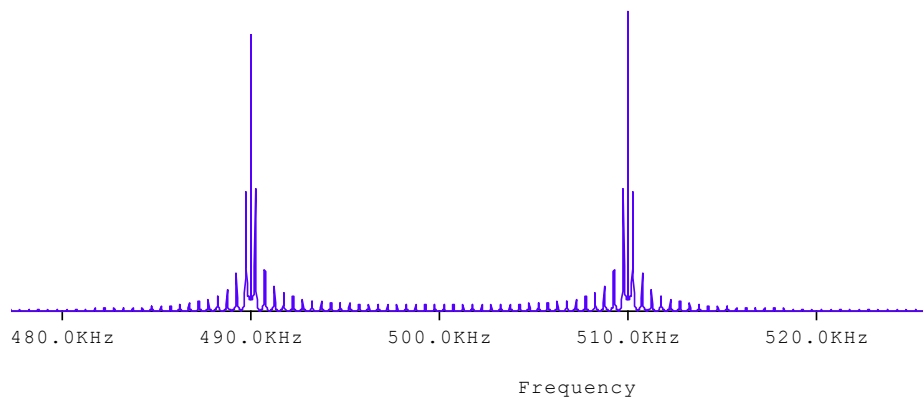


Figura 3.13 Espectro de frecuencia de la modulación FSK

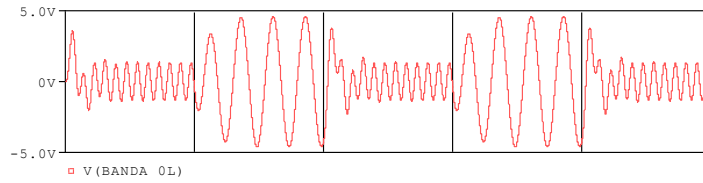
Verificando los resultados de los espectros de frecuencia de las Figuras 2.88, 3.11 y 3.13, se determinan las frecuencias (f_s y f_m) que se obtienen de la modulación de la señal binaria.

Para este caso de modulación 2-FSK se determina que se requiere dos frecuencias distintas para representar los niveles de entrada separadas por una desviación de frecuencia, por lo tanto, para el caso de modulación multinivel las frecuencias necesarias para modular cada nivel aumentarán implicando la necesidad de mayor ancho de banda.

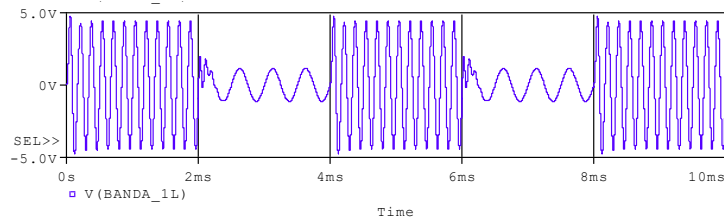
Análisis de resultados del demodulador

En el esquema demodulador presentado en el desarrollo práctico, cuya referencia de implementación se observa en la Figura 2.84, se verifica que las señales que se obtienen a la salida de cada bloque que lo compone comparadas con el proceso de demodulación ASK (Figura 2.82), se nota que el proceso de demodulación y las señales obtenidas a la salida son similares.

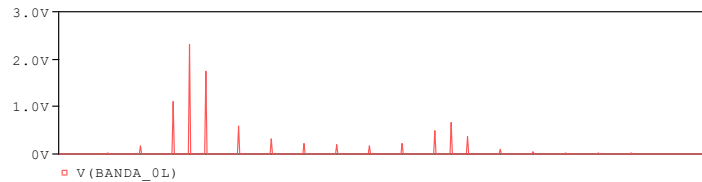
Es así que, como resultado del diseño del primer bloque de la Figura 2.91, con los datos paramétricos de la tabla 2.10 y tabla 2.11 (señal modulada en FSK) se obtiene a la salida de cada filtro la frecuencia que permite el paso del estado 0L y del estado 1L, ver Figura 3.14 a y b; las que son respectivamente la frecuencia de espacio y la frecuencia de marca, como se ilustra en las Figuras 3.14 c y d.



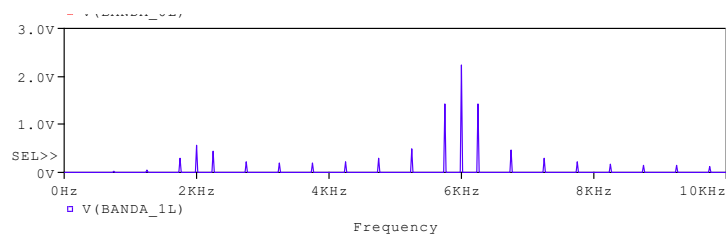
(a)



(b)



(c)

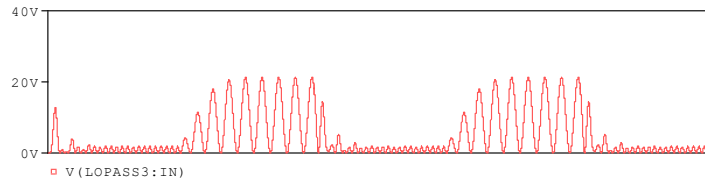


(d)

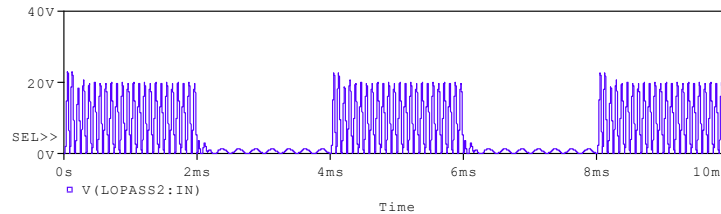
Figura 3.14 Forma de onda a la salida de los filtros pasa banda (a) para 0L, (b) para 1L, (c) espectro de (a), (d) espectro de (b)

Los resultados de la Figura 3.14 varían de acuerdo a la señal modulada de entrada, a lo cual también se debe acoplar el diseño de los filtros.

Luego, las señales resultantes pasan al bloque detector de envolvente donde se realiza un proceso de rectificación de señal, ver Figura 3.15, para luego después del filtro pasa bajo obtener la envolvente de la señal rectificada, ver Figura 2.94, finalmente el bloque comparador permitirá establecer la diferencia de los niveles de las envolventes.



(a)



(b)

Figura 3.15 Proceso de rectificación (a) para BANDA_0L, (b) para BANDA_1L

En el bloque comparador, Figura 2.93, para regenerar la señal de entrada al modulador se establece una diferencia de envolventes, es así que si la salida de ENVOL_1L es mayor a ENVOL_0L como resultado se tiene el estado 1L y en caso contrario se tiene un 0L, como se observa en la Figura 3.16.

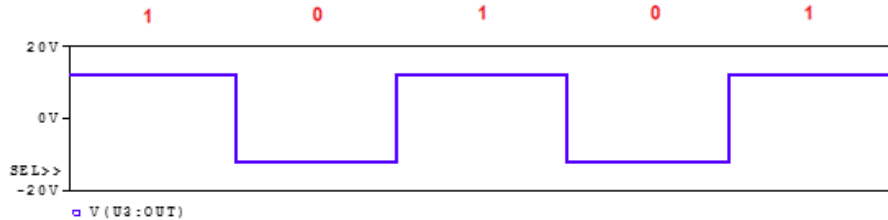


Figura 3.16 Señal demodulada

Comparando la señal binaria de entrada de la Figura 2.87 con la señal obtenida de la demodulación, se determina que los datos son recuperados exitosamente.

Dado el caso que el resultado del demodulador sea la gráfica de la Figura 3.17, una señal de un solo nivel donde no se distingue la recuperación de datos a la entrada del modulador, se deberá analizar la forma de las señales a la salida de los bloques que conforman el demodulador para detectar la línea del circuito en donde ocurre el error, estos resultados se obtienen de la alteración de los parámetros en el diseño del filtro para la ENVOL_1L. De acuerdo a lo mencionado se observa en la Figura 3.17 la forma de las señales del bloque detector de envolvente cuyos resultados reflejan que el nivel de la envolvente del estado 0L (Figura 3.17a, color rojo) es mayor que el nivel del estado 1L (Figura 3.17a, color

azul), por lo tanto, a la salida del demodulador se obtiene una señal que interpreta solamente el estado 0L.

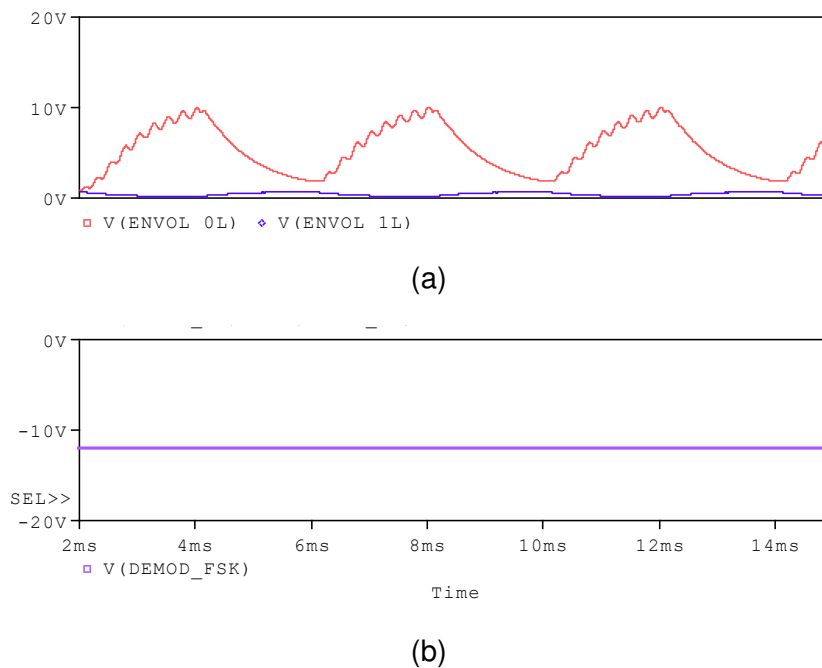


Figura 3.17 (a) Señal ENVOL_0L y ENVOL_1L, (b) Señal demodulada con parámetros erróneos de diseño

Para solucionar este caso, se deberá revisar el diseño del bloque que conforma el filtro pasa banda, en este caso el que implica la detección del estado 1L para obtener el correcto nivel comparativo de envolventes y así la correcta recuperación de datos.

Esta práctica permite el análisis de la modulación y demodulación FSK de una señal de entrada determinada, usando un esquema circuital diseñado en base a ciertos datos paramétricos iniciales. Además, el esquema modelo presentado puede ser reutilizado para aplicar un rediseño matemático respectivo de acuerdo a los parámetros requeridos.

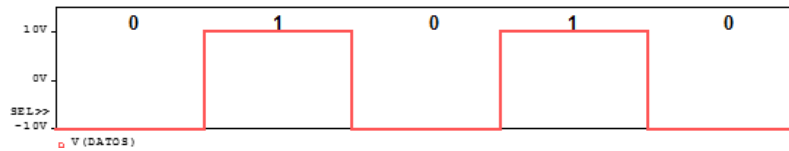
Con respecto a la modulación ASK, FSK cuenta con una mayor robustez ante la presencia de ruido y la interferencia siendo este un circuito de mayor complejidad como se nota en el diseño simulado.

3.7 Análisis de resultados sobre modulación PSK

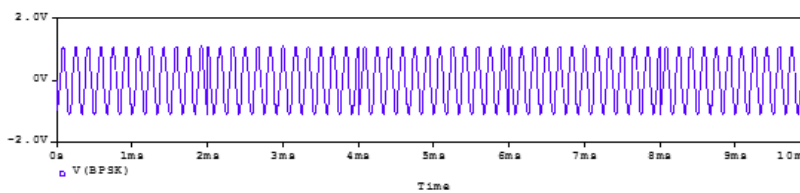
Los esquemas de modulación BPSK, QPSK y 8PSK presentados en la práctica 07, permiten observar el proceso que se lleva a cabo para modular en fase una señal binaria de entrada

Análisis de resultados esquema B-PSK

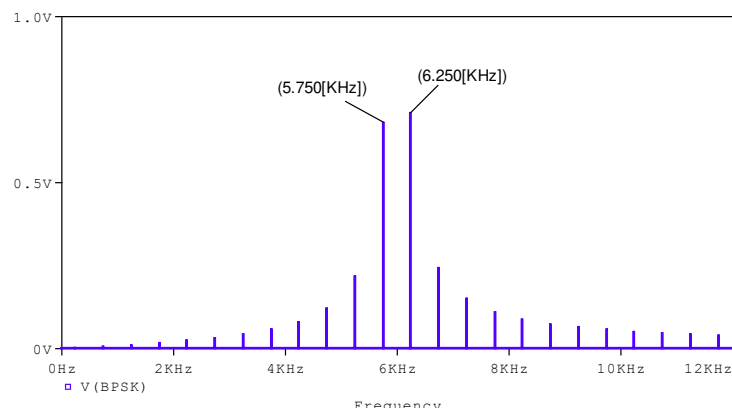
El esquema presentado en la Figura 2.106, cumple con el proceso de modulación en fase descrito en la teoría de la práctica. La Figura 3.18 presenta la modulación con el respectivo espectro de frecuencia que resulta de la secuencia de datos.



(a)



(b)



(c)

Figura 3.18 (a) Datos, (b) señal modulada BPSK, (c) espectro de la modulación BPSK

Los resultados obtenidos permiten determinar el correcto funcionamiento del modulador para realizar la prueba con cualquier secuencia de entrada. Además, en los resultados se confirma que para obtener las 2 posibles fases (M) a la salida del modulador, es necesario ingresar un estado lógico (N) por cada M , lo cual cumple con lo establecido en el detalle de variables de la ecuación 2.41 donde se tiene que $M = 2^N$.

Con respecto a la relación de velocidades de transmisión y modulación, se observa que para el primer caso de la Figura 2.108 donde se tiene una secuencia de entrada con un tiempo de bit de 0.5 [ms]; y, para el segundo caso de la Figura 3.18 donde T_b es de 2[ms], les corresponde una velocidad de transmisión de 2 [Kbps] y 500 [bps] respectivamente, cuyos

valores con el resultado de la velocidad de modulación resultan equivalentes, que aplicando la ecuación 2.42 son correctos los resultados.

La representación gráfica del espectro de frecuencia permite determinar el ancho de banda mínimo requerido para la modulación de una secuencia de entrada que para estos casos simulados se comprueba también que su valor es equivalente al valor de la velocidad de transmisión de datos en BPSK. Por lo tanto, los resultados demuestran que para una modulación BPSK el resultado de la velocidad de transmisión es equivalente al valor de la velocidad de modulación y ancho de banda necesario.

El modulador B-PSK implementado, de acuerdo a su esquema requiere parámetros de acoplamiento, es importante escoger un coeficiente de acoplamiento de tal manera que sea lo más próximo al valor de 1, con el fin de evitar las oscilaciones que se forman al conmutar de un estado al otro, como se muestra en la Figura 3.19.

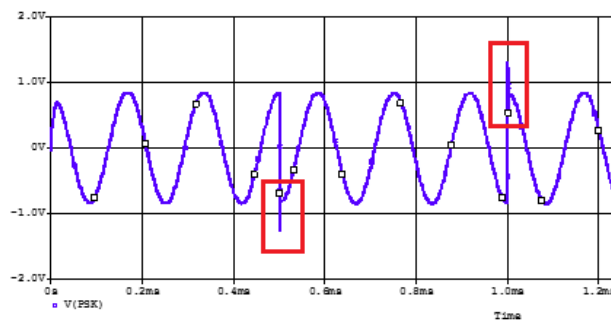


Figura 3.19 Modulación considerando un Coef=0.85

A pesar de las oscilaciones que se presentan cuando se considera un coeficiente de valor menor a 1, el funcionamiento que cumple el modulador no varía, pero si impide la visualización del desfase que se presenta, como se muestra en la Figura 3.20.

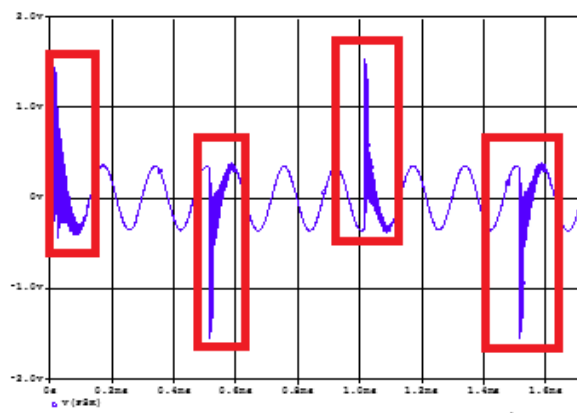


Figura 3.20 Modulación considerando un Coef=0.6

Para evitar este inconveniente en la modulación se ha considerado un coeficiente a 0.999.

Además, a manera de práctica se ha variado los niveles de voltaje de las entradas de las señales tanto portadora como datos digitales. Es así que, al plantear un valor de nivel de voltaje de portadora próximo al nivel de voltaje de la señal de datos, la señal tiende a deformarse de sinusoidal a un aspecto de señal cuadrada, Figura 3.21(b).

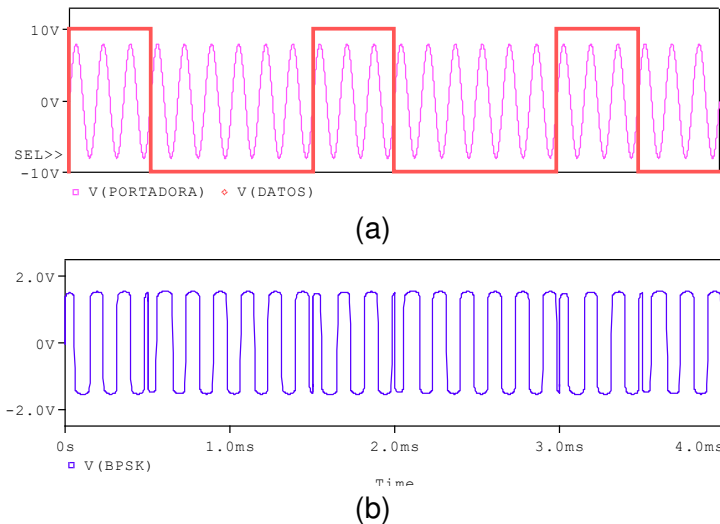


Figura 3.21 (a) Señal portadora $V_p=8$ [V] y señal de datos (rojo) $V=10$ [V],
(b) Modulación B-PSK

Con este antecedente es necesario que el nivel de voltaje de la señal digital sea un valor mucho mayor al nivel de la señal portadora. Ya que de esta manera se consigue que el control del funcionamiento de los diodos (encendido y apagado), recaiga en la señal de datos, Figura 3.22.

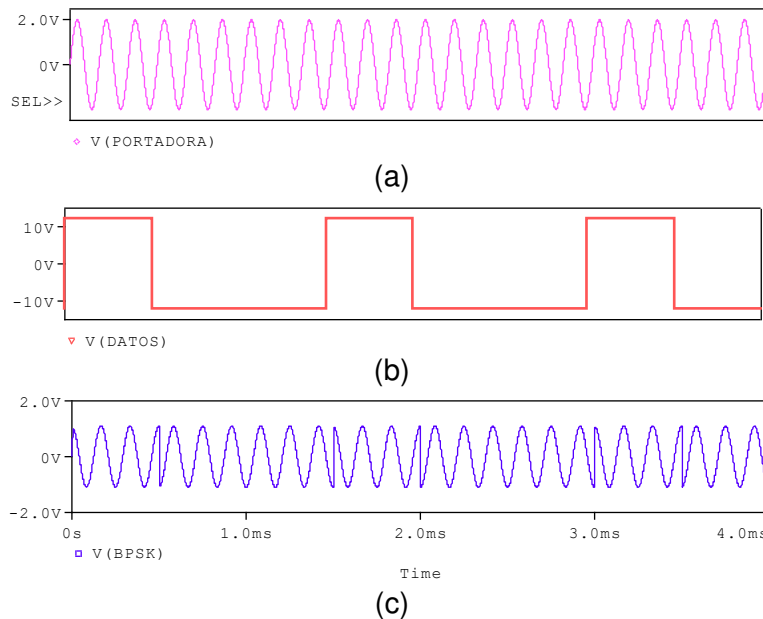


Figura 3.22 (a) Señal portadora $V_p=2$ [V], (b) señal de datos $V=12$ [V],
(c) Modulación BPSK

A modo de práctica para verificar resultados, se considera una señal de entrada unipolar donde el estado 0L está representado por un nivel de voltaje cero y el 1L con un cierto nivel de amplitud, se obtiene a la salida del modulador una modulación ASK del tipo OOK donde se verifica que cada vez que se detecta el bit cero se ausenta la portadora en la modulación, Figura 3.23.

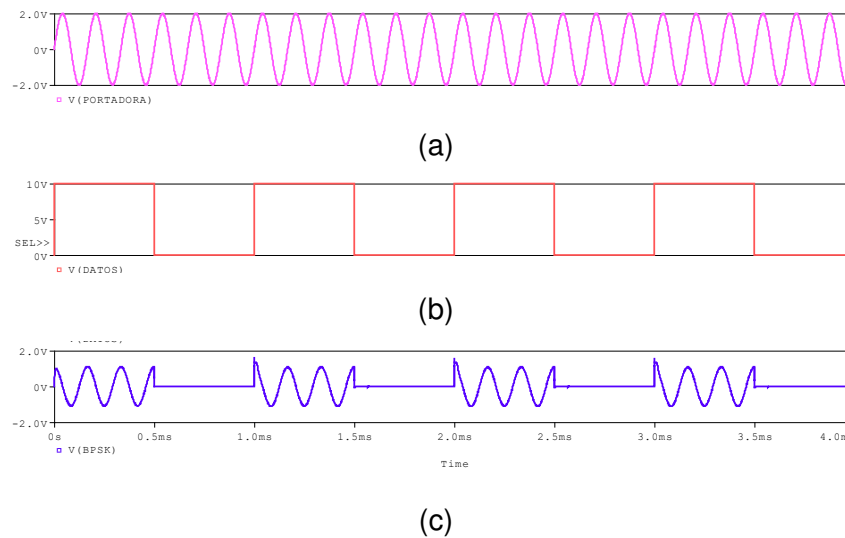


Figura 3.23 (a) portadora, (b) datos, (c) Modulación ASK, utilizando un esquema B-PSK

El ancho de banda ASK en este caso no se ve afectado y resulta igual al de la modulación PSK, Figura 3.24.

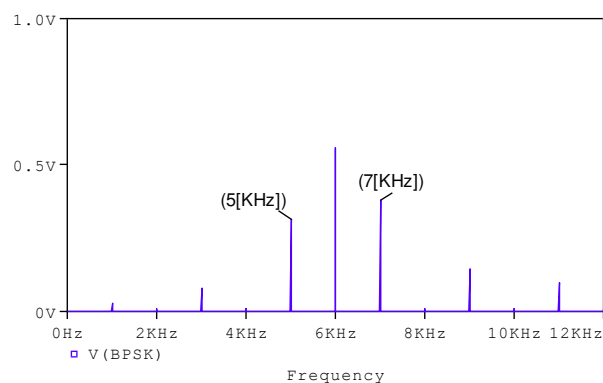


Figura 3.24 Espectro de frecuencia del modulador PSK

Ahora pues, si se analiza un esquema donde se varíe el parámetro de frecuencia de la señal portadora, de 6 [KHz] a 20 [KHz], se determina que no influye en absoluto el consumo del ancho de banda requerido, la modulación se cumple adecuadamente, lo único que sucede es que aumenta el número de oscilaciones por ancho de pulso, Figura 3.25. El ancho de banda variará de acuerdo al nivel de señal M-PSK.

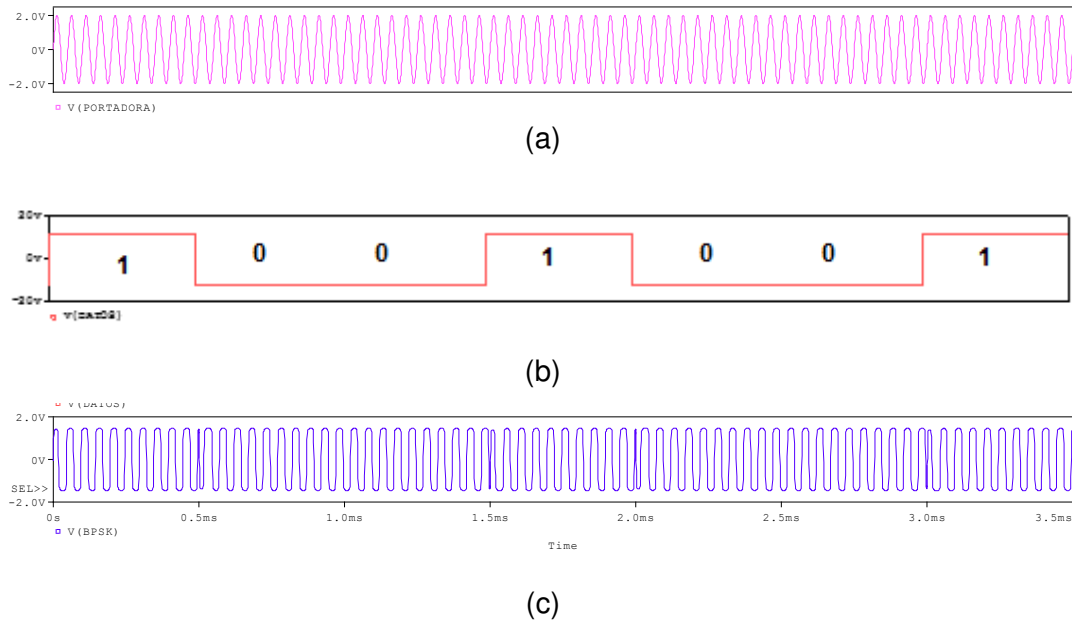


Figura 3.25 (a) Portadora, $f_p=20$ [KHz], (b) Datos, (c) Modulación B-PSK

El espectro de frecuencia se ilustra en la Figura 3.26.

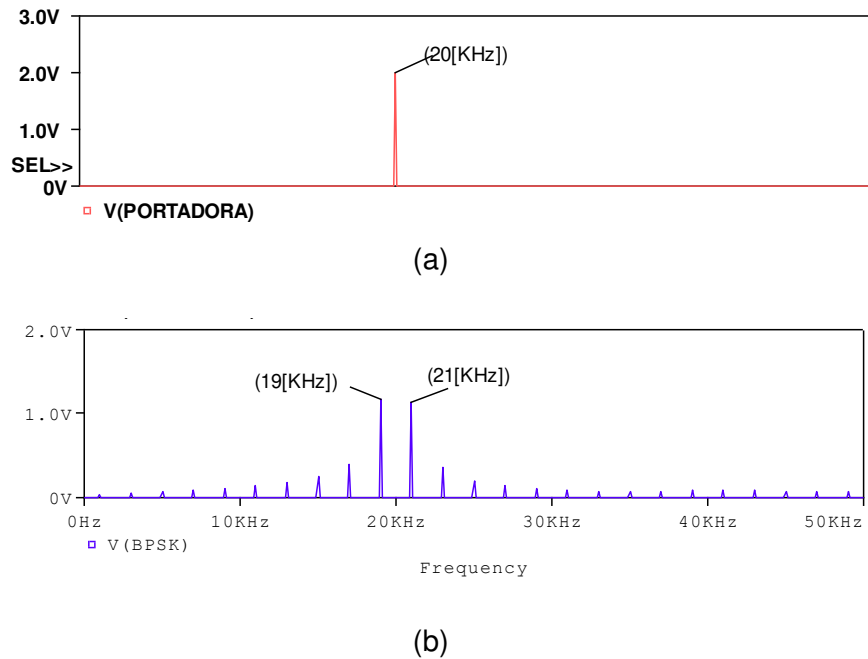


Figura 3.26 (a) Espectro de la portadora $f_p=20$ [KHz], (b) Espectro de frecuencia de la señal modulada BPSK

Análisis de resultados de los esquemas Q-PSK y 8-PSK

En los resultados de la simulación de la Figura 2.111 se verifica el funcionamiento del esquema Q-PSK implementado, donde la respuesta tiene 4 posibles variaciones de fases,

teniendo a la entrada 4 combinaciones de dos bits cada una, ingresadas a través del canal I y Q.

Si se verifica las señales a la salida del canal I y del canal Q, como se observa en la Figura 3.27, comparado con la respectiva señal de datos ingresada en cada canal, se determina que la modulación Q-PSK está formada por dos moduladores 2-PSK.

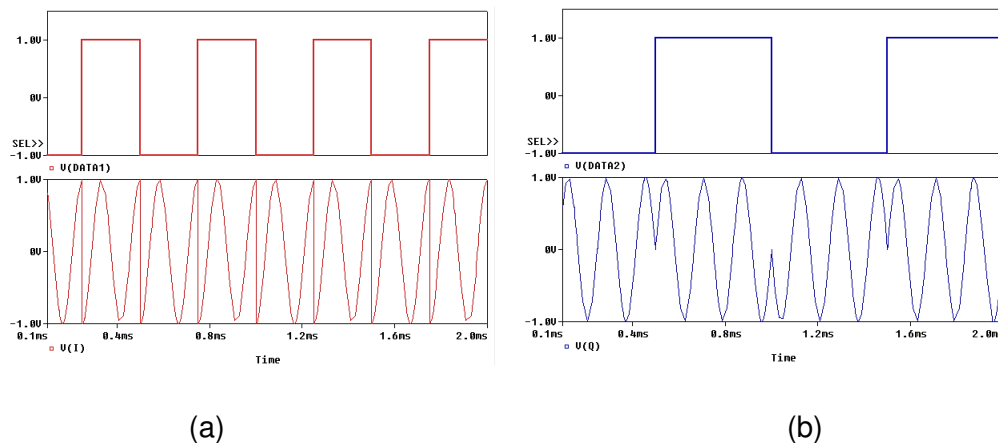


Figura 3.27 (a) Datos y señales a la salida del canal I, (b) Datos y señal a la salida del canal Q, del modulador QPSK

Esto quiere decir que el circuito implementado en la Figura 2.106, se puede simplificar con un simple circuito compuesto de un generador de pulsos, un generador de onda sinusoidal y un elemento multiplicador para comprobar el funcionamiento de un modulador B-PSK.

Comparando los resultados de la modulación QPSK versus la modulación BPSK, se verifica que es posible transmitir mayor número de bits en un estado, consiguiendo de esta manera mayor velocidad de transmisión y de acuerdo a la ecuación 2.42, la velocidad de modulación disminuye en la mitad de la velocidad de transmisión.

Con respecto a los espectros de frecuencia obtenidos de la modulación BPSK en la Figura 2.109 y QPSK en la Figura 2.112, se determina que el ancho de banda necesario para QPSK disminuye en la mitad del valor necesario para BPSK.

De los resultados obtenidos de la simulación de la modulación 8-PSK del esquema de la Figura 2.113, se verifica que a medida que aumenta el número de estados de modulación, las posibles fases a la salida del modulador también deben aumentar, se verifica además que la amplitud se mantiene constante, así como también la frecuencia de la señal modulada. Con este tipo de modulación de múltiples estados se puede conseguir la

transmisión de mayor información, lo cual implica además que la velocidad de modulación se reduzca, así como también en ancho de banda requerido.

En el esquema de modulación 8-PSK, el elemento que realiza la operación de conversión de la señal de voltaje de dos a cuatro niveles puede ser modificado con los niveles relativos que se desee, siempre y cuando cumplan con la relación arco tangente del ángulo mínimo de desfase, como se indica en la Figura 2.115.

Además, también se puede comprobar a través del diagrama de constelaciones que se presenta en la Figura 3.28 de la modulación 8 PSK, que, a mayor número de estados de modulación, la separación entre fases adyacentes va disminuyendo, lo cual implica que entre más cercanos están los puntos del diagrama de constelación la modulación se hace más vulnerable frente al ruido.

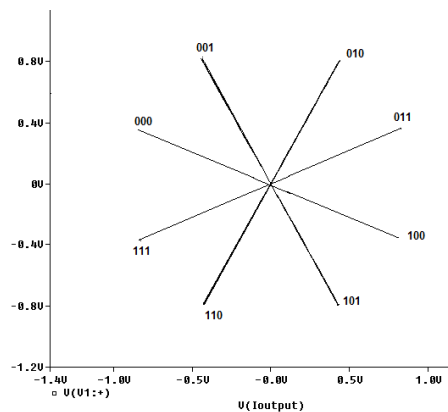


Figura 3.28 Diagrama de constelaciones de la modulación 8-PSK

En general, la modulación PSK tiene ciertas aplicaciones como el caso de BPSK que siendo la forma más básica puede ser utilizada en estándares de identificación por radio frecuencia en tarjetas de identificación electrónica, usadas principalmente en tarjetas de crédito.

3.8 Análisis de resultados sobre modulación QAM

El esquema presentado en la practica 08 funciona adecuadamente con los lineamientos descritos en el marco teórico de esta práctica, con la opción de TRACE de OrCAD PSpice, es posible visualizar el proceso de modulación, verificando las señales que se desee en los puntos de salida de los bloques establecidos del esquema simulado.

En los resultados de la modulación se verifica los cambios de fase y también la variación de amplitud que presenta la señal al ser modulada con 8-QAM de la siguiente manera: para

las primeras 4 condiciones de entrada la señal varía su fase y mantiene constante su amplitud y para las siguientes 4 condiciones la señal varía su amplitud, pero la variación de fases es similar al de las 4 primeras condiciones, como se observa en la Figura 3.29.

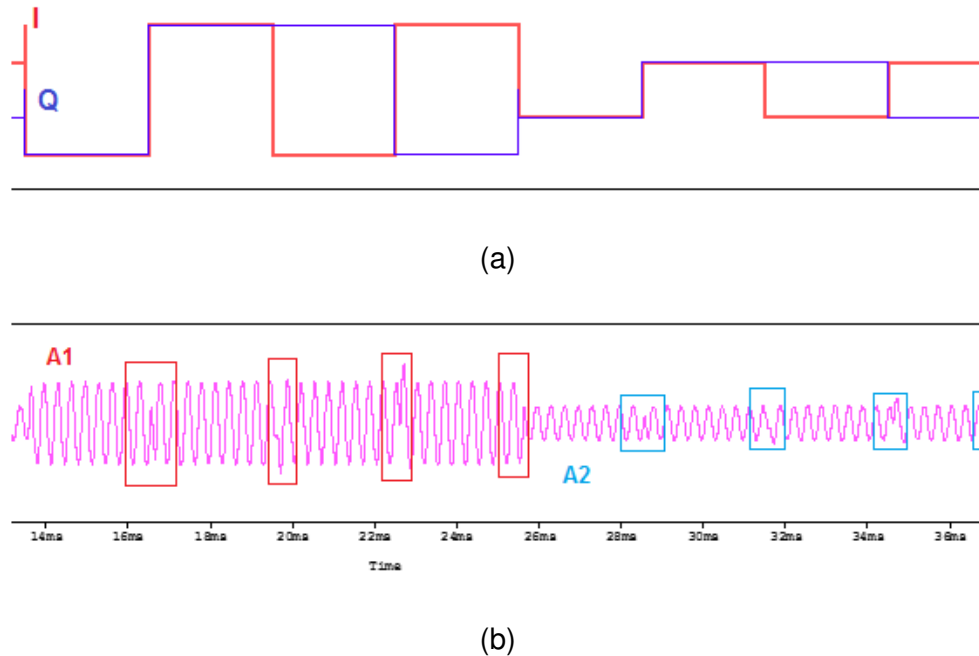


Figura 3.29 (a) Señal a la salida del canal Q (azul) y canal I (rojo), (b) Modulación 8-QAM. Esto quiere decir que si se hace una breve comparación con los valores referenciales de la Tabla 2.17, la modulación se cumple adecuadamente a lo detallado.

Observando además el resultado de la modulación 8-QAM, se puede decir que da lugar a dos modulaciones 4-PSK, ya que esta última se caracteriza por modular solo en fase.

Si se analiza las respuestas de un modulador 4-PSK y un 4-QAM son prácticamente las mismas, ya que solo se modula en fase y no varía la amplitud en ninguno de los dos, además el diagrama de constelaciones también resultaría igual.

Estableciendo un escenario de comparación entre el modulador 8-PSK obtenida en la práctica anterior, Figura 2.116, con la respuesta obtenida en esta práctica de la modulación 8-QAM, se verifica que en 8-PSK la fase tiene 8 posibles valores y una sola amplitud, mientras que 8-QAM tiene 4 posibles valores de fase con dos valores de amplitud. Lo cual conlleva a decir que el diagrama de constelación para los dos esquemas de modulación varía a partir de $M > 4$. Es así que para 8-QAM el diagrama de constelaciones se observa en la Figura 3.30.

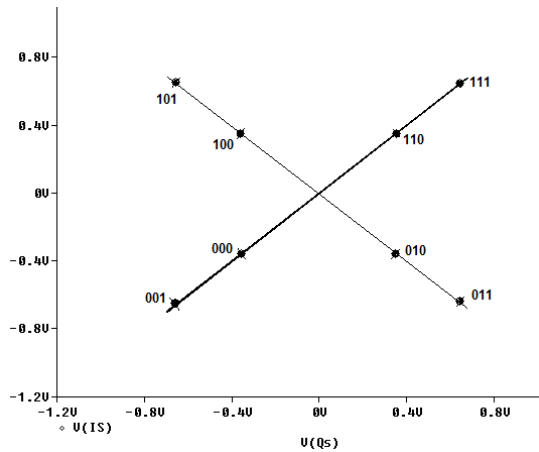


Figura 3.30 Diagrama de constelaciones de la modulación 8-QAM

Frente a PSK, la modulación QAM es mejor, debido a que, comparado con el diagrama de constelaciones de la Figura 3.28, para la misma relación señal a ruido QAM presenta mayor distancia entre los puntos adyacentes, resultando ventajoso ya que es menos vulnerable a errores, en cambio PSK a medida que aumenta el número de estados de modulación la separación entre fases es más pequeña y por ende implica que entre más cercanos los puntos existe mayor vulnerabilidad al ruido.

Con respecto al resultado del espectro de frecuencia obtenido en la Figura 2.124, se verifica el ancho de banda requerido y matemáticamente se determina aplicando la ecuación 2.44:

$$AB_{8QAM} = \frac{v_{tx}}{N} \rightarrow v_{tx} = \frac{1}{1 \text{ ms}} = 1000 \text{ [bps]}$$

$$AB_{8QAM} = \frac{1000}{3} = 333.33 \text{ [Hz]}$$

La respuesta indica que el ancho de banda calculado es coherente con la respuesta obtenida de la simulación.

Por las ventajas que presenta frente a PSK, las modulaciones QAM como 16QAM y 64QAM son aplicadas en sistemas de transmisión de señal de televisión digital por cable o terrestre utilizando *Digital Video Broadcasting*.

QAM se aplica para diseñar módems ADSL que trabajan en el rango de frecuencias entre 24 [KHz] – 1104 [KHz], en el cual se puede obtener velocidades de hasta 9 [Mbps].

4. CONCLUSIONES

- El desarrollo del presente trabajo de titulación incorpora el uso de herramientas computacionales como Matlab, Simulink y OrCAD PSpice para realizar prácticas que involucran temas de comunicación digital, buscando de esta manera el ahorro de elementos físicos y por ende costos de implementación.
- La realización de este trabajo de titulación busca incentivar a estudiantes de la carrera de ingeniería en Electrónica y Telecomunicaciones a investigar y probar sus propios diseños aprovechando las bondades de las herramientas computacionales en distintos escenarios de simulación solo utilizando una computadora portátil.
- Las prácticas desarrolladas en este proyecto de titulación han sido basadas directamente en la materia de Comunicación Digital y Procesamiento Digital de Señales, con el fin de aplicar los conocimientos adquiridos y a través de la experimentación comprender los distintos escenarios en donde además los estudiantes podrán entender de mejor manera los resultados obtenidos.
- El desarrollo de estas prácticas virtuales cuenta con la ventaja de poder variar parámetros, los cuales implican distintos escenarios de análisis, en el que los estudiantes pueden ampliar los conocimientos y despejar inquietudes.
- La aplicación de Simulink, Matlab y OrCAD PSpice en este trabajo resultaron ventajosas por sus amplias librerías y herramientas utilizadas, que hicieron posible la simulación, el desarrollo y análisis del comportamiento de cada práctica. Además, se demostró que estas herramientas pueden ser muy útiles en un laboratorio convencional de comunicaciones, ya que es posible obtener respuestas aproximadas, lo cual ayuda al estudio de los conceptos teóricos.
- Matlab es una herramienta en la cual es posible representar y visualizar sistemas en términos de funciones, mediante los comandos que ofrecen las librerías respectivas, además proporciona al usuario la capacidad de desarrollar aplicaciones e incentiva a colaborar con la investigación de las ventajas que implica el uso de sus herramientas como complemento de conocimientos teóricos.
- El uso de Simulink en laboratorios, permite crear esquemas para simulación y modelación de sistemas de comunicaciones, procesamiento de señales, etc., resultando de esta manera un ahorro en implementación, evitando la compra de dispositivos físicos, para obtener resultados similares. Esta herramienta además cuenta con un tiempo de procesamiento relativamente bajo y de gran eficiencia.
- OrCAD PSpice en el desarrollo de este trabajo permitió crear esquemas de modulación digital en los que fue posible verificar el funcionamiento en distintos

puntos del circuito. Además, PSpice permite esquematizar de tal modo que si se requiere realizar una implementación física evita de alguna manera excesivos gastos o errores en ejecución del circuito.

- En OrCAD PSpice se recomienda modificar parámetros de visualización para mejorar la apreciación de resultados, así como también el uso del sistema de ayuda, para mejorar el entendimiento del uso y conocimiento de los parámetros de configuración de los bloques de simulación.
- En Matlab y Simulink se recomienda el uso del sistema de ayuda para verificar la sintaxis de comandos que no se esté claro la aplicación, así como también la configuración de parámetros de los bloques de simulación.
- En las distintas prácticas ejecutadas se recomienda dado el caso sea necesario la reutilización de circuitos implementados para establecer distintos escenarios de análisis de resultados y así complementar conocimientos adquiridos.

GUIA MATLAB

Conclusiones y recomendaciones

- Matlab es una herramienta en la cual es posible representar sistemas en términos de funciones, mediante los comandos que ofrecen las librerías respectivas, de igual manera posibilita la visualización usando los comandos adecuados.
- La guía permite que el lector se familiarice con este software, en el desarrollo de programas en *scripts* en lenguaje .m, cuyos códigos pueden ser modificados o reutilizados en otras prácticas donde se requiera la aplicación.
- Matlab siendo un software, proporciona al usuario la capacidad de desarrollar aplicaciones e incentiva a colaborar con la investigación de las bondades que implica el uso de sus herramientas, como complemento en aplicaciones de conocimientos teóricos adquiridos.
- Durante el desarrollo de esta guía se pudo dar a conocer algunos de los comandos y funciones que se pueden utilizar para análisis y visualización de datos.
- Matlab es práctico y fácil de usar, ya que sus problemas se pueden expresar de la misma manera que se escriben matemáticamente y, en el ámbito de la ingeniería se puede simular y modelar sistemas relacionados con una gama de temas que se abarcan en materias como comunicación digital, análisis de señales y sistemas, etc.
- Se recomienda el uso del sistema de ayuda para verificar sintaxis y utilización de comandos que no se tenga claro su aplicación.

Conclusiones y recomendaciones prácticas

Práctica 01

- La convolución de señales discretas, permite manejar de manera arbitraria la señal a la entrada que puede tener infinitos valores, ya que resulta una suma ponderada de impulsos unitarios desplazados.
- La respuesta del sistema cuando se usa convolución discreta, resulta una suma ponderada de la respuesta a los impulsos, lo cual cumple con el principio de superposición de los sistemas lineales.
- La convolución continua está relacionada con el análisis de Fourier. Si se desea conocer la convolución de dos funciones es más factible obtener la transformada de Fourier de cada una de las funciones y luego multiplicar.
- El análisis de Fourier, es una herramienta muy útil, para realizar un sinnúmero de operaciones habituales en sistemas de comunicaciones.
- Se recomienda que para realizar las pruebas con la herramienta Matlab, se considere un número adecuado de muestras.
- Con el fin de asegurar que las gráficas obtenidas con Matlab son correctas, se recomienda realizar su respectivo análisis matemático.

Práctica 02

- Los sistemas de transmisión digital tienen la desventaja de tener ancho de banda infinito lo cual implica degradaciones de la señal.
- En la práctica el estrechamiento del ancho de pulso significa mayor velocidad de transmisión y consecuentemente también mayor consumo de ancho de banda.
- Si los tiempos de duración de pulso no tienen una diferencia considerable con el periodo, la forma del espectro se muestra como un conjunto de componentes de frecuencia en forma discreta y de manera contraria si la diferencia es muy considerable entre dichos tiempos de pulsos se presenta como una señal sampling continua.
- Para la reconstrucción de la señal basta con usar la frecuencia fundamental y cierto número de armónicos, para sintetizar la señal.
- Se puede decir que el ícono *Fourier* que presenta *OrCAD Capture*, es una herramienta eficaz que ayudará a ver el espectro de frecuencias que presentan las señales, ayudando de esta manera a su correspondiente análisis.

- El uso del filtro pasa bajo RC, permite regular el ancho de banda y se diseñará de acuerdo a los requerimientos de la señal.
- Por medio de la remodelación de señales usando análisis de *Fourier* se puede minimizar la interferencia intersímbolo, limitando el ancho de banda de manera que el receptor pueda rescatar su secuencia de mejor manera.
- Para la simulación de los espectros de las señales, se recomienda configurar los parámetros de simulación PSpice A/D como el parámetro *Run Time, Maximun step size* (mucho menor que *Run time*), *Output File Options/Print values in the output file*, para obtener una buena resolución de FFT.

Práctica 03

- Conociendo los datos paramétricos, las respectivas ecuaciones características de las funciones de los métodos de diseño y las respuestas impulsivas de los tipos de filtros (pasa bajo, pasa alto, pasa banda o elimina banda) se puede elaborar el script respectivo para verificar el funcionamiento.
- Teniendo una plantilla programada, es más eficiente usar la misma para realizar cualquier cambio o modificación y pueda ser ejecutado en otras aplicaciones.
- Un mismo tipo de filtro FIR puede ser diseñado fácilmente en Matlab aplicando cualquier método de ventanas como Bartlett, Hamming, Hanning, entre otros, ya que los comandos que tiene esta herramienta tienen una sintaxis simple, en la que no es necesario usar la ecuación característica de la función.
- Los filtros FIR presentan simetría en los coeficientes lo cual es ventajoso para la transmisión de datos, resultando ventajoso en aplicaciones médicas.
- Matlab resulta ideal para ver el comportamiento de los filtros FIR pasabanda por ventaneo Hamming sin interesar los valores iniciales, ya que el script está diseñado para ingresar los valores que se requieran.
- Es recomendable agregar comentarios en los scripts realizados para tener en cuenta los cambios realizados o los pasos a considerar al momento de ejecutar.

Práctica 04

- Teniendo una plantilla programada, es más eficiente reutilizar el mismo código para realizar cualquier cambio o modificación y poder aplicar en otras prácticas o aplicaciones que se requieran.

- Un filtro IIR puede ser diseñado fácilmente en Matlab aplicando los respectivos comandos que permitan diseño de filtros por el método de aproximación de filtros analógicos como Butterworth, Chebyshev o elíptica.
- Los filtros IIR básicamente consisten en obtener un filtro digital considerando un prototipo de filtro analógico, las situaciones que se benefician de estos tipos de filtros con los casos en los que se requiere reemplazar o actualizar un filtro analógico por otro digital equivalente.
- Cuando no es necesario mantener la respuesta al impulso o la fase de la respuesta en frecuencia del filtro analógico, la transformación bilineal resulta mejor que la invariante de impulso debido a que no tiene problemas de solapamiento.
- De los métodos que utiliza IIR para diseño a partir de filtros analógicos está el método del impulso invariante y el método de la transformada bilineal.
- Los filtros IIR en comparación con los filtros FIR, se ejecutan más rápido ya que requieren menos coeficientes para el proceso de filtrado.

Práctica 05

- Conforme el índice de modulación se aproxima a su máximo valor, la señal que se tiene en el nivel cero se va atenuando y en base al análisis matemático se determina que la portadora se mantiene activa cuando ingresa el valor 1L y se apaga mientras ingresa un 0L, estableciendo así que se trata de una modulación *ON/OFF* (OOK).
- Se comprobó la simulación de la modulación OOK, la cual se indica como una modulación al 100%.
- Se comprobó que el espectro de frecuencia de los datos de entrada, al ser modulados se trasladan a altas frecuencias.
- La modulación ASK resulta una variante de AM, y para el caso digital resulta un método no conveniente ya que estos tipos de sistemas son susceptibles al ruido, debido a que la información se transmite por la amplitud de la portadora, lo cual disminuye la confiabilidad de reconstrucción de la señal especialmente en secuencias largas de datos.
- El ancho de banda requerido para esta transmisión en pasa banda resulta el doble al requerido en banda base.
- La velocidad de transmisión en la modulación ASK de dos niveles resulta igual a la velocidad de modulación y de igual valor al ancho de banda requerido para la modulación.

- Se recomienda ajustar los parámetros de configuración para mostrar las gráficas y obtener de mejor manera los resultados de la modulación y demodulación, para un correcto análisis.

Práctica 06

- La implementación realizada con ayuda de la herramienta PSpice, permite observar los procesos de modulación y demodulación de la señal de datos ingresada de forma detallada. Lo que permite determinar que el diseño es correcto ya que se cumplió con el objetivo para obtener la reconstrucción de datos de entrada al sistema FSK.
- El sistema de modulación FSK requiere un diseño más complejo con respecto al de un modulador ASK, por lo que implica un consumo de mayor ancho de banda así como también implica robustez frente al ruido e interferencia.
- El diseño de un modulador / demodulador FSK multinivel implica un esquema más complejo por lo que resulta una desventaja ya que implicaría mayor costo de implementación.
- Utilizando el parámetro de desviación de frecuencia y frecuencia de portadora se podrá obtener los valores de la frecuencia de marca y la frecuencia de espacio, cuyos resultados serán importantes para obtener datos paramétricos para el diseño de los filtros pasa banda que conformarán el esquema de demodulación FSK.
- Es necesario establecer los niveles de voltaje de la señal de entrada ya que de esta manera se puede controlar la desviación de frecuencia a través del elemento VCO.
- La comparación del espectro de frecuencia de una modulación ASK con una FSK, analizando para una sola frecuencia, en los resultados obtenidos se puede ver que se asemejan, pero como para esta práctica FSK varía entre dos frecuencias se puede decir que el espectro de FSK es el resultado de la combinación de dos espectros de modulación ASK.
- En el esquema práctico de modulación/ demodulación, se realizó una variación de valores diferentes a los obtenidos en el diseño como es el de la resistencia R1 o R4 del bloque de filtros pasa banda, lo cual al verificar resultados se determinó que el proceso de demodulación de la señal se ve afectado y no puede ser exitosa la recuperación de datos, ya que las señales a la salida del bloque detector de envolvente no establecen un nivel comparativo de voltajes.
- Este proceso de modulación contemplada es de tipo FSK Coherente ya que en los resultados obtenidos la fase de la señal se mantiene al pasar de un estado al otro.

Prácticamente también se podría aplicar una modulación no coherente, pero para la implementación del esquema se requiere emplear osciladores independientes para la generación de la frecuencia de marca y de espacio. Un modulador FSK con VCO resulta más fácil ya que las frecuencias de conmutación dependen solo de los niveles de voltaje de la señal de entrada.

- Se recomienda establecer los parámetros adecuados de simulación para observar de mejor manera las formas de onda resultantes a la salida de cada bloque implementado.
- En el diseño de filtros pasa banda se recomienda variar los parámetros calculados a valores aproximados a los calculados para mejorar los resultados del proceso de modulación dado el caso que se vea necesario y así obtener mejores resultados.
- Se recomienda trabajar con bajas frecuencias para verificar el proceso de modulación y dado el caso que se usen altas frecuencias utilizar las herramientas necesarias para evidenciar resultados, por lo que al ejecutar, como resultado se obtiene un barrido o mancha de color.

Práctica 07

- En el esquema de modulación BPSK implementado, el nivel de voltaje de la señal de datos es el que se encargará del correcto funcionamiento de los diodos. Es decir, de la polarización de los mismos.
- En la implementación del modulador BPSK, es posible obtener un modulador ASK, modificando los parámetros de niveles de voltaje de la señal de datos, en este caso para el 0L se le da el nivel cero voltios. Es decir, hay que trabajar con una señal de datos unipolar positiva.
- Se recomienda que para el acoplamiento de los bloques que comprenden el modulador BPSK es necesario que se establezca un coeficiente de acoplamiento lo más próximo a 1, con el fin de eliminar las oscilaciones que se pueden producir en el cambio de condición lógica.
- La modulación BPSK es óptima para protección frente a errores, siendo una modulación eficiente para la transmisión de datos binarios.
- El ancho de banda de una señal modulada en BPSK resulta igual al de una modulación ASK, y equivalente al valor de la velocidad de transmisión.
- Los resultados de la modulación BPSK, QPSK y 8PSK realizados en esta práctica, permiten determinar que a medida que aumentan el número de estados de modulación es posible la transmisión de mayor información consiguiendo de esta

manera mayores velocidades de transmisión sin necesidad de aumentar al ancho de banda.

- Considerando una misma velocidad de transmisión en M-PSK, el ancho de banda requerido disminuye en la misma proporción que la velocidad de modulación.
- En el esquema realizado de QPSK se determina que el modulador está compuesto por dos moduladores de BPSK.
- El modulador QPSK, permite además verificar que es posible la construcción simplificada de un modulador BPSK.
- En el esquema modulador 8PSK, se recomienda realizar la configuración adecuada de los valores en cada elemento conversor de dos a cuatro niveles para obtener el correcto funcionamiento del modulador.
- El ancho de banda en MPSK, depende del número de bits codificados (N) por condición posible a la salida (M). En el caso de BPSK donde es una secuencia de unos y ceros alternados cuenta con un ancho de banda máximo.
- El diagrama de constelación permite representar los estados de modulación en términos de amplitud y fase.

Práctica 08

- La técnica de modulación QAM presenta mayor aprovechamiento de ancho de banda en comparación a otras técnicas de modulación ya que ofrece la posibilidad de transmitir dos señales sobre la misma portadora.
- La modulación QAM multinivel presenta variación de amplitud y fase, implicando decir que la modulación QAM es una combinación de la modulación PSK y ASK.
- En 8-QAM la velocidad con la que ingresan los datos a los canales I, Q y C es un tercio de la velocidad de transmisión de entrada, al igual que 8-PSK.
- Con el espectro de frecuencia se puede determinar que el ancho de banda mínimo necesario para la modulación tanto 8-PSK como 8-QAM es la tercera parte de la tasa de transmisión de entrada.
- La representación en el diagrama de constelaciones de QAM versus PSK, para señales multinivel, QAM tiene un mejor comportamiento porque las distancias entre los puntos adyacentes son mayores frente a PSK.
- Las modulaciones QAM y 4PSK, tienen las mismas respuestas a excepción de la representación en el diagrama de constelaciones.
- Se recomienda que a partir del esquema presentado para la modulación 8-QAM se establezca un modulador 8-PSK y determine la diferencia que marca con 8-QAM.

Para esto solo basta aumentar un inversor en el canal CD y realizar una configuración en los parámetros del convertidor de 2 a 4 del canal Q.

Esquema de un sistema de transmisión multiportadora usando COFDM

- COFDM es el resultado de una mejora de OFDM el cual utiliza múltiples portadoras ortogonales para la modulación, cada una modulada en amplitud y fase. Esta técnica al combinarse con codificación de canal para detección y corrección de errores se denomina COFDM.
- La simulación realizada en Simulink del sistema de transmisión multiportadora con COFDM funciona adecuadamente con todos los bloques necesarios de acuerdo al esquema y diagramas de bloque descritos en la teoría; la información a ser transmitida la cual es una señal de audio de 0.03s se decodifica como se puede verificar en el workspace de matlab de la simulación del diagrama de bloques implementado.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] Smith Steven W., "The Scientist and Engineer's Guide to Digital Signal Processing" , 1st ed., Copyright ©1997-1998 ,Inc.ISBN-13: 978-0966017632
- [2] Robert J. Schilling, Sandra L. Harris, "Fundamentals of Digital Signal Processing using Matlab", Clarkson University, 2010
- [3] Arellano, Miguel, "Matlab & Simulink para Ingeniería Nivel 1", Universidad de Ciencias y Humanidades, 2013
- [4] MathWorks, "Communications System Toolbox/ Key features".
- [5] MathWorks Documentantation "Getting Started with Communications System Toolbox".
- [6] MathWorks Documentantation "DSP System Toolbox".
- [7] OrCAD PSpice Designer, Advanced circuit simulation and analysis for analog and mixed-signal circuits, 2016, Available: <http://www.orcad.com/resources/orcad-downloads>
- [8] Aguillar J.D., Lozano P., Morales R., López M., Manual de Simulación de circuitos electrónicos de potencia con OrCAD PSpice, Escuela Politécnica Superior de la Universidad de Jaen, España, 2011
- [9] Vinay K. Ingle, John G. Proakis, "Digital Signal Processing using Matlab" Copyright © 2010 Cengage Learning, Inc. ISBN-13: 978-1-111-42737-5
- [10] Prandoni P., Vetterli M., "Signal Processing for Communications" Copyright © 2008 CRC Press, ISBN: 978-1-4200-7046-0
- [11] Smith Steven W., "The Scientist and Engineer's Guide to Digital Signal Processing", 1st ed., Copyright ©1997-1998 ,Inc.ISBN-13: 978-0966017632
- [12] Paul Tobin, "PSpice for Filters and Transmission Lines" First Edition, Morgan y Claypool, Copyright © 2007, ISBN: 9781598291599

- [13] Ing. Pablo Hidalgo, Diapositivas Comunicaciones Digitales EPN
- [14] Ing. Ma. Soledad Jiménez, Comunicación Digital EPN, Enero 2012
- [15] Usategui Angulo J., Zapirain García B., Martinez Angulo I., Sáez J., "Microcontroladores Avanzados dsPIC, Controladores Digitales de Señales. Arquitectura, programación y aplicaciones", España, Copyright ©2006 ,Inc.ISBN: 84-9732-385-8
- [16] Silage Denis, "Digital Communication system using System Vue", First Edition, USA, Copyright © 2005, ISBN: 1-58450-850-7. Cap 3
- [17] Leon W. Couch, "Digital and Analog Communication Systems", Eighth Edition, Prentice Hall, Copyright © 2013, ISBN-13: 978-0-13-2915380, Cap 5
- [18] Terman J., Verghese C., Balakrishnan H., "Bits, Signals, and Packets, An Introduction to Digital Communications & Networks", Lecture Notes, 2012, chapter 14.
- [19] Tomasi W., "Sistemas de Comunicaciones Electrónicas", Cuarta Edición, México, Prentice Hall, Copyright © 2003, ISBN-13: 970-26-0316-1, Cap. 12
- [20] Eugene Estino, Eritek, Inc., "VisSim Tutorial Series Simulation of Communication Systems", Copyright © 1998, Cap. Modulation, pag 6
- [21] Carlson Bruce A., Crilly Paul B., "Communications Systems An Introduction to Signals and Noise in Electrical Communication", First Edition, Copyright © 2002, ISBN: 978-0-07-338040-7, Cap 14
- [22] Universidad Don Bosco Facultad de Ingeniería Electrónica, Guías prácticas de Sistemas de Comunicaciones II.
Available: <http://www.udb.edu.sv/udb/archivo/guia/electronica-ingenieria/sistemas-de-comunicacion-ii/2013/ii/guia-5.pdf>
- [23] Manrique S., Comunicaciones mediante voz digital, Artículo n°231, CQ Radio Amateur, marzo 2003. Available: <http://www.qsl.net/e/ea3du//2003/03/DV1.html>

- [24] SRTV, Sistemas de radio digital, Copyright 2017. Available: <https://srtv-2011-2012.wikispaces.com/4.7+Sistemas+de+radio+digital>
- [25] Departamento de Física y Arquitectura de computadores, Análisis de circuitos y sistemas lineales Diseño de filtros por el método de aproximación, Universidad Miguel Hernández. Available: http://repositorio.innovacionumh.es/Proyectos/P_19/Tema_5/UMH_06.htm
- [26] John G. Proakis, Masaud Salehi, "Communication systems engineering", second edition Prentice Hall© 2002 New Jersey, Inc. ISBN:0-13-061793-8
- [27] Paul Tobin, "PSpice for Digital Communications Engineering", 1st ed., Copyright©2007, ISBN: 9781598291636.
- [28] Lali Barrière, Series de Fourier, Octubre 2011, PDF, pag 4. Available: <https://mat-web.upc.edu/people/lali.barriere/as/serie-fourier.pdf>
- [29] Angulo J.M., García B., Angulo I., Sáez J.V., Microcontroladores Avanzados dsPIC Controladores Digitales de Señales Arquitectura, programación y aplicaciones, España, Copyright©2006, ISBN: 84-9732-385-8, pag 26
- [30] J.H.STOTT, The how and why of COFDM, TUTORIAL-COFDM, EBU Technical Review-winter 1998.
- [31] International Journal of Science and Modern Engineering (IJISME), ISSN:2319-6386, Volume-1, Issue-5, abril 2013.
- [32] TESIS DE GRADO, ALFONSO JARA, MULTIPLEXACION POR DIVISION DE FRECUENCIA ORTOGONAL CODIFICADA, UNIVERSIDAD AUSTRAL DE CHILE, 2005.
- [33] Tratamiento Digital de Señal, Diseño de Filtros Digitales Parte I, Cap 8, pag 25. Available: <http://www4.tecnun.es/asignaturas/tratamiento%20digital/tema8.pdf>
- [34] Departamento de teoría de la señal y comunicaciones, "Análisis y síntesis de circuitos", Universidad de Alcalá, Pag. 15. [Online]. Available: <http://agamenon.tsc.uah.es/Asignaturas/ittse/asc/apuntes/Tema2.pdf>

- [35] Pablo Turmero, Filtros Electronicos, Especificaciones reales para el diseño de filtros, monografias.com. Available: <http://www.monografias.com/trabajos106/filtros-electronicos/filtros-electronicos.shtml>
- [36] F.V. Fernandez, "TEMA 4. APROXIMACION DE FILTROS", Dpto. de Electrónica y Electromagnetismo, ESI, US, Pag. 76, 77, 78. [Online]. Available: http://www2.imse-cnm.csic.es/elec_esi/assignat/ASC/1997-98/pdf/tema4.pdf
- [37] Ignacio Moreno Velasco, " Filtrado", Área de tecnología Electrónica, Universidad de Burgos, pág. 24. [Online]. Available: <http://www.unet.edu.ve/~ielectro/6-Filtrado.pdf>
- [38] J.I. Huircan, "Filtros Activos: Conceptos Básicos", Universidad de la Frontera, Enero 25, 2012. Pág. 2,3. [Online]. Available: <http://146.83.206.1/~jhuircan/PDF ELECTRONICA/FAct01g.pdf>
- [39] Lucio D. Torres, "Filtros Activos, Filtros de Retroalimentación Múltiple", Cap. 3, Pág. 44, 45, 46, Escuela de Ingeniería y Ciencias, Universidad de las Américas de Puebla, México, 2012. [Online]. Available: http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/torres_d_ld/capitulo3.pdf
- [40] Francisco J. Franco Peláez," Filtros Analógicos", Ingeniería Superior Electronica, Facultad de Físicas, Universidad Complutense de Madrid. Pág. 12 -16. [Online]. Available:[https://cv3.sim.ucm.es/access/content/group/portal-uatducma-43/webs/material original/apuntes/PDF/09_filtros analogicos.pdf](https://cv3.sim.ucm.es/access/content/group/portal-uatducma-43/webs/material%20original/apuntes/PDF/09_filtros_analogicos.pdf)
- [41] Departamento de Electrónica y Automática, "Introducción al diseño de Filtros Activos", Facultad de Ingeniería, Universidad Nacional de San Juan, Argentina, Pág.2 - 5. [Online]. Available: <http://dea.unsj.edu.ar/sredes/filtrosactivos.pdf>
- [42] Schertz, A. and Weck, C. "Hierarchical Modulation: The Transmisión of two independent DVB-T Multiplex on a Single Frequency". EBU. Technical Review. April 2003.

6. ANEXOS

ANEXO I

UNA EXPERIENCIA CON PROCESAMIENTO DE SEÑALES

CODIGO .m

```
%%GUIA1: UNA EXPERIENCIA CON PROCESAMIENTO DE SEÑALES
%ESCUELA POLITÉCNICA NACIONAL
%TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
%ELECTRONICA Y TELECOMUNICACIONES
%BY: VALERIA MURILLO SALAZAR
%MAYO2017

% Desarrollo de código (script con extensión .m), en el cual se aplica el
% uso de herramientas que cuenta Matlab para realizar una guía práctica
% como introducción al uso de esta poderosa herramienta. Este
procedimiento
% de guía se resume en la aplicación de un filtro, con el fin de atenuar
% una componente de frecuencia de una señal de dos tonos.
% La primera parte generará la señal de dos tonos.
% La segunda parte diseñará un filtro paso bajo.
% La tercera parte aplicará el filtro a la señal de entrada
% Finalmente se analizará resultados de acuerdo a las gráficas generadas.

%% GENERACIÓN DE SEÑALES
% Creación de dos tipos de señales sinusoidales de diferentes
frecuencias.
% Las señales se generarán de acuerdo a los parámetros iniciales como
% frecuencia, número de muestras, intervalos de evaluación de señal etc.
clc;
clear all;
close all;
%Parámetros iniciales.
Fs = 100e3;      %Frecuencia de muestreo.
Ts = 1/(Fs);    %Tiempo de muestreo.
N = 5000;       %Número de muestras y/ó Longitud de señal.
%Vector tiempo
%Intervalo de duración y evaluación de señal
t = (0:N-1)*Ts;
% Señal sinusoidal a considerar:
% SINTAXIS:  $y = \sin(2\pi \cdot F_o \cdot t)$ ; donde  $F_o$  es la frecuencia de señal.

% SIGNAL1
% y1=Señal sinusoidal de frecuencia 5KHz
y1=2*sin(2*pi*5000*t);
figure(1)
subplot(2,1,1);
%Visualización de onda en función del parámetro tiempo.
plot(1000*t(1:200),y1(1:200),'g','LineWidth',1.2)
%Formato gráfica.
xlabel('tiempo [ms]');
ylabel('y_1(t)');
leg = legend('Señal 5[kHz]');
```

```

set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1 , ...
    'FontName'     , 'Arial' , ...
    'FontSize'     , 14);
axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')

% SIGNAL2
% y2=Señal sinusoidal de frecuencia 500Hz
y2=2*sin(2*pi*500*t);
subplot(2,1,2);
plot(1000*t(1:200),y2(1:200),'b','LineWidth',1.2)
%Formato gráfica.
xlabel('tiempo [ms]'          );
ylabel('y_2(t)'              );
leg = legend('Señal 500[Hz]');
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1 , ...
    'FontName'     , 'Arial' , ...
    'FontSize'     , 14);
axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
print -djpeg -r300 TutoMATLAB2FIG1

%% SUMA DE SEÑALES y1+y2
% Las señales sinusoidales previamente generadas, serán el principio del
% desarrollo de esta guía, ya que se sumarán para generar la señal y3.
%Adición de señales previamente creadas.
y3=y1+y2; % Suma de señales
figure(2)
%Visualización de y3, y2,y1
plot(1000*t(1:200),y3(1:200),'r','LineWidth',1.5)
hold on
plot(1000*t(1:200),y1(1:200),'g-','LineWidth',1.3)
plot(1000*t(1:200),y2(1:200),'b--','LineWidth',1.3)
%Formato gráfica.
xlabel('tiempo [ms]'          );
ylabel('y_3(t)'              );
leg = legend('Suma de señales','Señal 5[kHz]','Señal 500[Hz]');
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1 , ...
    'FontName'     , 'Arial' , ...
    'FontSize'     , 14);
axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
print -djpeg -r300 TutoMATLAB2FIG2
hold off

```

```

%% CONTENIDO FRECUENCIAL DE LA SEÑAL A PRUEBA.
% Es complicado identificar las distintas componentes de frecuencia al
ver
% la señal de prueba que se ha creado. Por tal motivo se aplicará la
% Transformada de Fourier, para encontrar las componentes de frecuencia de
% esta señal, considerando una frecuencia de muestreo ya establecida
% anteriormente en los parámetros iniciales.
%Parámetros iniciales
%Número de puntos para evaluar FFT, basado en la longitud de señal.

Y=fft(y3);
Yab=abs(Y/N);
Yab1=Yab(1:N/2+1);
%Intervalo de frecuencia.
%SINTAXIS: linspace(X1, X2, N)
%Genera N puntos entre X1 y X2.
f = Fs*(0:(N/2))/N;
% Gráfica de espectro de frecuencia de y3
figure(3)
% Visualización de FFT de la señal Y
plot(f,2*Yab1,'LineWidth',1.3)
%Formato gráfica.
xlabel('Frecuencia [Hz]')
ylabel('|Y[f]|')
leg = legend('Espectro de frecuencia de y_3');
xlim([0 6e3])
set(gca, ...
    'Box'      , 'on'      , ...
    'XGrid'    , 'on'      , ...
    'XColor'   , [.3 .3 .3], ...
    'LineWidth', 1 , ...
    'FontName' , 'Arial' , ...
    'FontSize' , 14);
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
print -djpeg -r300 TutoMATLAB2FIG3
%% DISEÑO DE FILTRO PASO BAJO
% Se aplicará un filtro paso bajo, para atenuar la componente de
frecuencia
% de 5[kHz] de la señal y3.
% Se diseñará un filtro con característica de respuesta frecuencial de un
% filtro A/D. Para lo cual se aplicará un filtro Butterworth paso bajo.
% Cuya sintaxis se describe de la siguiente manera:
% [N, Wn] = buttord(Wp, Ws, Rp, Rs)
%Rp Tolerancia de paso
%Rs Tolerancia de rechazo

fs=100e3; %frecuencia de muestreo
Nyq=fs/2; %Nyquist
Wp=1000/Nyq; % de 0 a Wp Banda de paso
Ws=3000/Nyq; % Mayor a Ws es Banda de rechazo
%resultado, orden y frecuencia de corte del filtro.
[n,Wn] = buttord(Wp,Ws,1,30)
%calculo de coeficientes numerador y denominador del filtro.
[num,den]=butter(n,Wn);

figure(4)
[H,W]=freqz(num,den,N);

```

```

plot(W./pi,abs(H))
%Formato gráfica.
xlabel('w = Normalized Frequency (x ? rad/ sample)')
ylabel('Magnitud |H[w]|')
title(sprintf('n = %d Butterworth Lowpass Filter',n))
leg = legend('Respuesta en amplitud del filtro')
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1            , ...
    'FontName'     , 'Arial'      , ...
    'FontSize'     , 14);
axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%Grafica de filtro en funcion de la frecuencia normalizada.
print -djpeg -r300 TutoMATLAB2FIG4

tf(num,den);
%Respuesta en frecuencia normalizada Wn= 2pi*(fn/fs)
figure(5)
[num,den] = butter(n,Wn);
freqz(num,den)
%Formato gráfica.
title(sprintf('n = %d Butterworth Lowpass Filter',n))
leg = legend('Respuesta en frecuencia normalizada');
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1            , ...
    'FontName'     , 'Arial'      , ...
    'FontSize'     , 14);
axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%Grafica de filtro en funcion de la frecuencia normalizada.
print -djpeg -r300 TutoMATLAB2FIG5

figure(6)
[z,p,k] = butter(n,Wn);

%zp2sos Zero-pole-gain to second-order sections model conversion.
sos = zp2sos(z,p,k);
%Respuesta en frecuencia del filtro diseñado .
freqz(sos,100,fs)
%Formato gráfica.
title(sprintf('n = %d Butterworth Lowpass Filter',n))
leg = legend('Respuesta en frecuencia [Hz]');
axis([0 6e3 0 -400]);
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1            , ...
    'FontName'     , 'Arial'      , ...
    'FontSize'     , 14);
axis('tight')

```

```

set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%# save as jpeg
print -djpeg -r300 TutoMATLAB2FIG6

%% PROCESO DE FILTRADO DE SEÑAL
% Aplicación de filtro a y3 con el fin de extraer la señal sinusoidal
% de frecuencia 500Hz.
figure(7);
%Aplicación de filtro en y3.
sigfil=filter(num,den,y3);
%Señal Filtrada en funcion del tiempo.
plot(1000*t(1:250),sigfil(1:250),'LineWidth',1.2)
%Formato gráfica.
xlabel('tiempo [ms]');
ylabel('y_3(t)_Filtrada');
leg = legend('y_3 Filtrada');
axis([0 2.2 -3 3]);
set(gca, ...
    'Box'      , 'on'      , ...
    'XGrid'    , 'on'      , ...
    'XColor'   , [.3 .3 .3], ...
    'LineWidth', 1 , ...
    'FontName','Arial', ...
    'FontSize', 14);
%axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%# save as jpeg
print -djpeg -r300 TutoMATLAB2FIG7
%% ESPECTRO DE SEÑAL FILTRADA
% Se visualizará el espectro de la señal y3, usando el comando "fft".

figure(8)
%Señal Filtrada en frecuencia.
N=5000;
F=fft(sigfil);
Fab=abs(F/N);
Fab1=Fab(1:N/2+1);
f = Fs*(0:(N/2))/N;

plot(f,2*Fab1,'LineWidth',1.3)
%Formato gráfica.
xlabel('Frecuencia [Hz]');
ylabel('|Y[f]|_Filtrada');
leg = legend('Espectro de Frecuencia de y_3 filtrada');
axis([0 6e3 0 3]);
set(gca, ...
    'Box'      , 'on'      , ...
    'XGrid'    , 'on'      , ...
    'XColor'   , [.3 .3 .3], ...
    'LineWidth', 1 , ...
    'FontName','Arial', ...
    'FontSize', 14);
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%# save as jpeg
print -djpeg -r300 TutoMATLAB2FIG8

```

ANEXO II

INTRODUCCION AL PROCESAMIENTO DIGITAL DE SEÑALES

CODIGO .m

```
%PRACTICA01: Introducción al procesamiento digital de señales
%ESCUELA POLITECNICA NACIONAL
%TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
%ELECTRONICA Y TELECOMUNICACIONES
%BY: VALERIA MURILLO SALAZAR
%MARZO2018

%% Funcionamiento de un sistema LTI
% Esta práctica destaca el funcionamiento de un sistema LTI en el tiempo
% discreto, considerando una secuencia de entrada de un pulso rectangular
% y una respuesta impulsiva exponencial de valor real.

clc;
clear all;

x=[zeros(1,10),ones(1,15),zeros(1,40)]; % secuencia de entrada x(n)
k=length(x);
nx=(-10:1:k-11); %Intervalo de evaluación de señal

figure(1)
stem(nx,x,'MarkerFaceColor','green','color','r');
%Etiquetado informativo de gráfica.
Title = title('SECUENCIA DE ENTRADA ','color','b');
xlabel('[n] ');
ylabel('x[n]');
axis([-5,30,0,2]);
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1            , ...
    'FontName'     , 'Arial'      , ...
    'FontSize'     , 10);
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%# save as jpeg
print -djpeg -r300 PDS_01

nxx=(0:k-1); %intervalo de evaluacion de la respuesta impulsiva
h=[zeros(1,5),triangularPulse(0, 7, 15, nxx)] % Onda triangular

%h=[zeros(1,5),0.8.^nxx]; %forma de respuesta impulsiva
k1=length(h);
nh=(-5:1:k1-6); xlabel('[n]'); ylabel('h[n]');

figure(2)
stem(nh,h,'MarkerFaceColor','r','color','b') % visualizacion de la
respuesta impulsiva en el tiempo discreto
```

```

%Título, etiquetado informativo de gráfica.
Title = title ('RESPUESTA IMPULSIVA ', 'color', 'b');
xlabel(' [n] ');
ylabel('x[n]');
axis([-5,30,0,2]);
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1            , ...
    'FontName'     , 'Arial'      , ...
    'FontSize'     , 10);
%axis('tight')
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
set(gcf, 'PaperPositionMode','auto')
%# save as jpeg
print -djpeg -r300 PDS_02

% intervalo de evaluacion de la señal a la salida del sistema LTI
n=(-15:1:length(x)+length(h)-17);
%aplicaicon de la convolucion lineal para determinar la salida del
sistema LTI
y=conv(x,h);
z=y(21:84);

figure(3)
stem(n,y, 'MarkerFaceColor','g', 'color','m');
xlabel(' [n]'); ylabel('y[n]');
%Título, etiquetado informativo de gráfica.
Title = title ('SECUENCIA DE SALIDA ', 'color', 'b');
xlabel(' [n] ');
ylabel('x[n]');
axis([-5,30,0,8]);
set(gca, ...
    'Box'          , 'on'          , ...
    'XGrid'        , 'on'          , ...
    'XColor'       , [.3 .3 .3], ...
    'LineWidth'    , 1            , ...
    'FontName'     , 'Arial'      , ...
    'FontSize'     , 10);
%axis('tight')
%# set size of figure's "drawing" area on screen
set(gcf, 'Units','centimeters', 'Position',[0 0 15 10])
%# set size on printed paper
%#set(gcf, 'PaperUnits','centimeters', 'PaperPosition',[0 0 5 10])
%# WYSIWYG mode: you need to adjust your screen's DPI (*)
set(gcf, 'PaperPositionMode','auto')
%# save as jpeg
print -djpeg -r300 PDS_03

```

ANEXO III

DISEÑO Y SIMULACION DE UN FILTRO FIR

CODIGO .m

```
%PRACTICA03: DISEÑO Y SIMULACIÓN DE UN FILTRO FIR
%ESCUELA POLITÉCNICA NACIONAL
%TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
%ELECTRONICA Y TELECOMUNICACIONES
%BY: VALERIA MURILLO SALAZAR
%MARZO2018

%FILTRO FIR POR EL METODO DE VENTANEO HAMMING;
clear;
clear all;
disp('El filtro FIR satisface los siguientes requerimientos');
% ws1=input('ingrese el valor de ws1: '); % lower stopband edge
% wp1=input('ingrese el valor de wp1: '); % lower passband edge
% wp2=input('ingrese el valor de wp2: '); % upper passband edge
% ws2=input('ingrese el valor de ws2: '); % upper stopband edge
% Rp=input('ingrese el valor de Rp: '); % passband ripple
% As=input('ingrese el valor de As: '); % stopband attenuation

ws1=0.2*pi; % lower stopband edge
wp1=0.4*pi; % lower passband edge
wp2=0.5*pi; % upper passband edge
ws2=0.8*pi; % upper stopband edge
Rp=0.5; % passband ripple
As=40; % stopband attenuation
delta1=0.1; % tolerancia en la banda pasante
delta2=0.1; % tolerancia en la banda de paso

%Comprobación de valores de tolerancia de acuerdo a los valores
ingresados
% Select the min(delta1,delta2) since delta1=delta2 in window design
%[delta1,delta2] = db2delta(Rp,As);

if (delta1 < delta2) % Comparación de valores
    delta2 = delta1;
    disp('Delta1 is smaller than delta2')

    %function [Rp,As] = delta2db(delta1,delta2)
    %Conversion from Absolute delta specs to Relative dB
    [Rp,As] = delta2db(delta1,delta2)
end

%Calculo del mínimo ancho de banda de transición del filtro a ser
diseñado
%para el calculo del orden del Filtro
ABt = min((wp1-ws1), (ws2-wp2));
%De acuerdo a la Tabla 2.8 se tiene que el ancho de pulso en Hamming es
%6.6pi/M
%Aproximacion al inmediato si la respuesta es un numero decimal, el valor
%del orden debe ser un número entero
M = ceil(6.6*pi/ABt);
M=2*floor(M/2)+1, % odd M
```



```

% n = 0:M-1;
wham = (hamming(M))';
% w_hann = (hann(M))';
% w_black = (blackman(M))';
% w_bart = (bartlett(M))'
% Cálculo de las frecuencias de corte de los respectivos filtros ideales
wc1 = (ws1+wp1)/2;
wc2 = (ws2+wp2)/2;
% Caracterización del filtro tipo paso banda /Ver Tabla 2.6
alpha = (M-1)/2;
n = [0:1:(M-1)];
m = n-alpha+eps;
% Respuesta impulso deseada (paso banda)
hd = sin(wc2*m) ./ (pi*m) - sin(wc1*m) ./ (pi*m)
% Respuesta impulso del filtro diseñado
% hd = ideal_lp(wc2,M) - ideal_lp(wc1,M);
h = hd.*wham;
% [db,mag,pha,grd,w] = freqz_m(h,1);
[H,w] = freqz(h,1,1000,'whole');
H = (H(1:1:501))'; w = (w(1:1:501))';
mag = abs(H);
db = 20*log10((mag+eps)/max(mag));
pha = angle(H);
grd = grpdelay(h,1,w);
delta_w = pi/500;
Asd = floor(-max(db([1:floor(ws1/delta_w)+1])))
Rpd = -min(db(ceil(wp1/delta_w)+1:floor(wp2/delta_w)+1)),
%
%% Filter Response Plots
% Hf_1 = figure('Unit','normalized','position',[0.1,0.1,0.8,0.8],
'color',[0,0,0]);
figure(1)
stem(n,hd,'MarkerFaceColor','m','color','g','LineWidth',1.2);
title('Respuesta Impulso ideal: Pasabanda');
axis([-1,M,min(hd)-0.1,max(hd)+0.1]); xlabel('n'); ylabel('hd(n)')
% set(gca,'XTickMode','manual','XTick',[0:M-1],'fontsize',10)

figure(2)
stem(n,wham,'MarkerFaceColor','r','color','b','LineWidth',1.2);
title('Función ventana de Hamming');
axis([-1,M,-0.1,1.1]); xlabel('n'); ylabel('w_h_a_m(n)')
set(gca,'XTickMode','manual','XTick',[0:M-1],'fontsize',10)
set(gca,'XTickMode','manual','YTick',[0:1],'fontsize',10)

figure(3)
stem(n,h,'MarkerFaceColor','y','color','g','LineWidth',1.2);
title('Respuesta Impulso del filtro diseñado: Pasabanda');
axis([-1,M,min(hd)-0.1,max(hd)+0.1]); xlabel('n'); ylabel('h(n)')
set(gca,'XTickMode','manual','XTick',[0:M-1],'fontsize',10)

figure(4)
plot(w/pi,db,'MarkerFaceColor','y','color','c','LineWidth',1.2);
title('Filtro FIR-PASABANDA ');
axis([0,1,-As-30,5]); xlabel('frecuencia in x
pi.rad/sample'); ylabel('Decibeles')

```

ANEXO IV

DISEÑO Y SIMULACION DE UN FILTRO IIR

CODIGO .m

```
%%PRACTICA04: DISEÑO Y SIMULACIÓN DE UN FILTRO IIR
%%ESCUELA POLITÉCNICA NACIONAL
%%TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
%%ELECTRONICA Y TELECOMUNICACIONES
%%BY: VALERIA MURILLO SALAZAR
%%MARZO 2018
clear;
clear all;
% ws=input('ingrese el valor de wsl: '); % Frecuencia para banda rechazo
% wp=input('ingrese el valor de wpl: '); % Frecuencia para banda de paso
% Rp=input('ingrese el valor de Rp: '); % Rizado banda de paso
% As=input('ingrese el valor de As: '); %Rizado banda de rechazo
%% Especificaciones paramétricas.
wp = 0.3*pi; % Passband edge in rad/sam
Rp = 1; % Passband Riple in dB
ws = 0.8*pi; % Stopband edge rad/sam
As = 50; % Stopband attenuation in dB
%
%% Transformada Bilineal, T = 2
T = 2;
Op = (2/T)*tan(wp/2); % Analog Prototype Passband edge
Os = (2/T)*tan(ws/2); % Analog Prototype Stopband edge
%[cs,ds] = afd_elip(OmegaP,OmegaS,Rp,As); % Analog Prototype Desing
if Op <= 0
    error('Passband edge must be larger than 0')
end
if Os <= Op
    error('Stopband edge must be larger than Passband edge')
end
if (Rp <= 0) | (As < 0)
    error('PB ripple and/or SB attenuation ust be larger than 0')
end
ep = sqrt(10^(Rp/10)-1);
A = 10^(As/20);
k = Op/Os;
k1 = ep/sqrt(A*A-1);
capk = ellipke([k.^2 1-k.^2]);
capk1 = ellipke([(k1 .^2) 1-(k1 .^2)]);
N = ceil(capk(1)*capk1(2)/(capk(2)*capk1(1)));
fprintf('\n*** Elliptic Filter Order = %2.0f \n',N)
%[b,a]=u_elipap(N,Rp,As,wp);
%Prototipo/ Filtro Eliptico;
[z,p,k] = ellipap(N,Rp,As);
    a = real(poly(p));
    aNn = a(N+1);
    p = p*Op;
    a = real(poly(p));
    aNu = a(N+1);
    b = real(poly(z));
    M = length(b);
    bNn = b(M);
    z = z*Op;
```

```

    b = real(poly(z));
    bNu = b(M);
    k = k*(aNu*bNn)/(aNn*bNu);
    b0 = k;
    b = k*b;

%[b,a] = bilinear (cs,ds,l/T);    % Bilinear Transformation
%
Fs =0.5;                        % Sampling Frequency
[num,den]=zp2tf(z,p,k);        % Convert to transfer function form
fvtool(num,den)
[numd,dend]=bilinear(num,den,Fs); % Analog to Digital conversion
fvtool(numd,dend)              % Visualize the filter

%% (b) Use of the 'Ellip' function
[N,wn] = ellipord(wp/pi,ws/pi,Rp,As);
[b,a] = ellip(N,Rp,As,wn);
h = fvtool(b,a)

```



```

    constellation_Q = [ -1  -1  -1  -1 -1/3 -1/3 -1/3 -1/3  1  1  1  1
1/3 1/3 1/3 1/3];
else,
    disp('ERROR: UNKNOWN PAM TYPE');
    clear constellation_I
    clear constellation_Q
end;
Pow_symbol = mean(constellation_I.^2+constellation_Q.^2);
    % mean power per symbol (e.g. 5/9 for 4ASK bipolar)

% delay due to serial/parallel conversion in PAM transmitter
delay_SP = 0;
if(m>1),
    delay_SP = 1; % 1 symbol duration delay if SP-converter is active
end;

% TIME PERIODS
Tbit; % bit period
Tpam = N*m*Tbit*Reff; % PAM symbol period
Tofdm = (L/(L+Lcp))*Tpam; % OFDM symbol period
Tcp = (Lcp/(L+Lcp))*Tpam; % duration of cyclic prefix

% channel parameters
No = 2.5*10^-5 % double-sided noise power density (in Watt/Hz)
SNR = 10*log10( S/(2*No/(Reff*m*Tbit)) ); % HF-S/N = NF-S/N for I/Q-
demodulation
SNR = SNR*L/(L+Lcp); % only fraction of signal power is in the useful
% OFDM symbol (the rest in the cyclic prefix)

% general paramters
Nbit = 11200 % number of transmitted bit
PAMtype = 'QPSK' % signalconstellation '16QM' for 16QAM or 'QPSK' for
QPSK
S = 1 % signal power (in Watt)
KANALtype = 'AWGN' % possible: '2pth' ---> frequency selective 2-path
channel
% 'AWGN' ---> AWGN channel

% source prameters
Tbit = 1/28000 % duration of the source (information) bit (in sec)

% FEC parameters (framing and channel coding)
Reff = 1/2; % effective code rate
Ninfo = 64; % number of information bit per frame
%
% (frame of 64 infobit plus 6 tailbit results in 127 transmitted code
bit)
K_conv = 7 % constraint length (memory length + 1)
G_conv = [133,171] % generator polynomials in octal representation
punct_conv = ones(140,1);
punct_conv([11,22,33,44,55,66,77,88,99,110,121,133]) = zeros(12,1); %
puncturing pattern

```

```

% OFDM PARAMETERS
N = 32          % number of subcarriers
L = 64          % IDFT and DFT length (L>N, L has to be a power of 2)
Lcp = 6         % length of cyclic prefix in sample periods

% PAM SIGNAL CONSTELLATION
                % M=2^m constellation points for the M binary
                % vectors (addresses) of m elements in increasing
                % binary order (e.g. '00', '01', '10', '11' for m=2)
if(PAMtype=='QPSK'), % BPSK fuer I (2.Bit) und Q (1.Bit)
    m = 2; % number of bit per modulation symbol
    constellation_I = [ 1 -1  1 -1]/sqrt(2);
    constellation_Q = [ 1  1 -1 -1]/sqrt(2);
elseif(PAMtype=='16QM'), % 4ASK für I (1. DiBit) und Q (2.DiBit)
    m = 4; % number of bit per modulation symbol
    %
    %           0   1   2   3   4   5   6   7   8   9   10  11
12 13 14 15
    constellation_I = [ -1 -1/3  1  1/3 -1  -1/3  1  1/3 -1 -1/3  1  1/3
-1 -1/3  1  1/3];
    constellation_Q = [ -1  -1  -1  -1 -1/3 -1/3 -1/3 -1/3  1  1  1  1
1/3 1/3 1/3 1/3];
else,
    disp('ERROR: UNKNOWN PAM TYPE');
    clear constellation_I
    clear constellation_Q
end;
Pow_symbol = mean(constellation_I.^2+constellation_Q.^2);
                % mean power per symbol (e.g. 5/9 for 4ASK bipolar)

% delay due to serial/parallel conversion in PAM transmitter
delay_SP = 0;
if(m>1),
    delay_SP = 1; % 1 symbol duration delay if SP-converter is active
end;

% TIME PERIODS
Tbit; % bit period
Tpam = N*m*Tbit*Reff; % PAM symbol period
Tofdm = (L/(L+Lcp))*Tpam; % OFDM symbol period
Tcp = (Lcp/(L+Lcp))*Tpam; % duration of cyclic prefix

% channel parameters
if(KANALtype=='2pth'),
    hchannel = [sqrt(3/4) zeros(1,4/(m*Reff)) (1+i)*sqrt(1/8)]; %
multipath impulse response
elseif(KANALtype=='AWGN'),
    hchannel = 1; % AWGN channel impulse response
else,
    disp('ERROR: UNKNOWN CHANNEL TYPE');
    clear hchannel
end;

```

```

figure(1); set(1,'Name','channel frequency response'); subplot(211);
Hchannel = fft(hchannel,1024);
plot(((L+Lcp)/Tpam)*[-511:512]/1024,abs(Hchannel([1024-
510:1024,1:513])), 'r-');
axis([-
(L+Lcp)/(2*Tpam),(L+Lcp)/(2*Tpam),0,ceil(10*max(abs(Hchannel)))/10]);
ylabel('|H(f)|'); xlabel('f in Hz');
subplot(212);
plot(((L+Lcp)/Tpam)*[-511:512]/1024,-angle(Hchannel([1024-
510:1024,1:513]))/pi, 'r-');
axis([- (L+Lcp)/(2*Tpam),(L+Lcp)/(2*Tpam),-1,1]);
ylabel('-angle(H(f))/pi'); xlabel('f in
Hz');
%
No = 2.5*10^-5 % double-sided noise power density (in Watt/Hz)
SNR = 10*log10( S/(2*No/(Reff*m*Tbit)) ); % HF-S/N = NF-S/N for I/Q-
demodulation
SNR = SNR*L/(L+Lcp); % only fraction of signal power is in the useful
% OFDM symbol (the rest in the cyclic prefix)

% NUMBER OF OFDM PILOT FRAMES
Pofdm = 32;

% PILOT PHASE SIMULATION FOR CHANNEL ESTIMATION
sim('COFDM_pilotphase');
channel_meas_COFDM;

```

ORDEN DE EMPASTADO