

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Facultad de Ingeniería en Electricidad y Computación

**“DISEÑO, SIMULACIÓN E IMPLEMENTACIÓN DE UN ESQUEMA DE
DETECCIÓN DE TRAMA Y DE CORRECCIÓN DEL DESPLAZAMIENTO DE
FRECUENCIA EN UN SISTEMA DE COMUNICACIÓN INALÁMBICO DE
UNA SOLA PORTADORA”**

TESINA DE SEMINARIO

Previo a la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Presentada por:

Wellington Eduardo Espinoza Velasteguí

Maricela Eliset Freire Barba

Guayaquil – Ecuador

2013

AGRADECIMIENTO

Agradezco a mis padres por la bendición de darme la vida, la paciencia y apoyo que han sido en cada paso de mi vida y más en esta última prueba. A Mis hermanas por su comprensión y palabras de motivación para seguir adelante. Agradezco al PhD. Boris Romos por su apoyo constante, al Ing. Gómez por sus consejos, sus conocimientos y su amistad. A mi compañera y amiga Maricela por acompañarme en el desarrollo de este último proyecto, por su dedicación y entusiasmo. A cada uno de los muchachos que conformamos el seminario de graduación.

A Dios por permitirme seguir viviendo hasta el día de hoy.

Wellington Eduardo Espinoza Velasteguí

Agradezco a mi familia por su apoyo incondicional, a mis amigos del Seminario de Graduación por compartir sus conocimientos, a Wellington mi compañero de proyecto por su apoyo y paciencia durante este tiempo, al PhD. Boris Ramos por permitirme formar parte de este grupo de investigación, al Ing. Jorge Gómez por brindarnos su ayuda durante la realización de este proyecto.

Finalmente gracias a Dios se culmina una etapa y otros grandes retos se abren ante mí.

Maricela Eliset Freire Barba

DEDICATORIA

Dedicado a Dios, a mis padres por su paciencia, mis hermanas, a mis amigos que me aconsejaron cada momento y, en especial, a mis amigas Mirian Lucero y Giselle Velásquez por su bella amistad, constante apoyo, motivándome cada día.

Wellington Eduardo Espinoza Velasteguí

Dedico este trabajo a Dios por ofrecer ese apoyo espiritual que nos da fortalezas y guiarnos en el camino correcto. A mi familia que siempre ha sido mi apoyo en las buenas y las malas de esta exigente carrera. A amigos que me enseñaron lo valioso de tener una amistad verdadera e incondicional en especial a mi amiga Nury Cornejo por su apoyo en todo momento.

Maricela Eliset Freire Barba

TRIBUNAL DE SUSTENTACIÓN

A handwritten signature in blue ink, appearing to read 'B. Ramos', written over a horizontal line.

PhD. BORIS RAMOS

PROFESOR DEL SEMINARIO
DE GRADUACIÓN

A handwritten signature in blue ink, reading 'Jorge Gómez Ponce', written over a horizontal line.

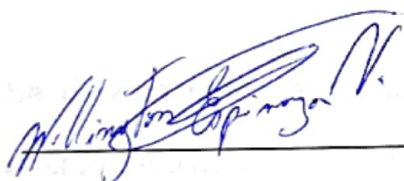
Ing. JORGE GÓMEZ

PROFESOR DELEGADO POR
LA UNIDAD ACADÉMICA

DECLARACIÓN EXPRESA

"La responsabilidad por los hechos, ideas y doctrinas expuestas en esta tesina de seminario, nos corresponden exclusivamente; y el patrimonio intelectual de la misma, a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL".

(Reglamento de Graduación de la ESPOL)



Wellington Eduardo Espinoza

Velasteguí



Maricela Eliset Freire Barba

RESUMEN

Basado en el desarrollo de la práctica 5 del manual de laboratorio [1] de los equipos USRP, sobre sincronización de trama, buscamos la viabilidad de adaptar un método, aplicado para OFDM, al sistema propuesto por el manual. Lo que nos permitió conocer más acerca de los equipos, desarrollar diferentes esquemas de modulación, canales de propagación.

Se analiza los cambios necesarios para adaptar el algoritmo Schmidl & Cox, diseñado para sistemas OFDM, a sistemas de portadora única. Por lo que se hace una revisión completa y sencilla de todos los bloques involucrados en el sistema de una sola portadora. Así como la descripción de los métodos de sincronización de trama implementados tanto en la práctica como el que se busca adaptar.

Se utilizaron métodos de observación para visualizar la efectividad de los sincronizadores y realizar un efectivo análisis de los mismos, comparación y funcionalidad. Se verá cómo afecta la modulación a los dos sistemas de sincronización, así como el cambio de ambiente de propagación.

ÍNDICE GENERAL

AGRADECIMIENTO	I
DEDICATORIA.....	III
TRIBUNAL DE SUSTENTACIÓN.....	IV
DECLARACIÓN EXPRESA	V
RESUMEN	VI
ÍNDICE GENERAL.....	VII
ABREVIATURAS	XI
ÍNDICE DE FIGURAS.....	XII
ÍNDICE DE TABLAS	XXIV
INTRODUCCIÓN.....	XXV
CAPÍTULO 1.....	27
1.1 DESCRIPCIÓN DEL PROBLEMA.....	27
1.2 OBJETIVOS GENERALES.....	29
1.3 OBJETIVOS ESPECÍFICOS	30
1.4 JUSTIFICACIÓN DEL PROYECTO.....	31
1.5 DESCRIPCIÓN DEL PROYECTO.....	32
CAPÍTULO 2.....	34
2.1 CONCEPTOS DE COMUNICACIÓN.....	34
2.1.1 SISTEMA DE COMUNICACIÓN DE SEÑALES EN BANDA BASE.	35
2.1.2 SISTEMA DE PORTADORA ÚNICA.....	35

2.1.2.1 MODULACIÓN Y DEMODULACIÓN	37
2.1.2.2 INSERCIÓN DE SÍMBOLOS DE ENTRENAMIENTO	47
2.1.2.3 SOBRE MUESTREO (OVERSAMPLING).....	50
2.1.2.4 FILTRO FORMADOR DE PULSOS Y FILTRO DE ACOPLAMIENTO	52
2.1.2.5 CONVERTIDORES	53
2.1.2.6 ECUALIZADOR	54
2.1.2.7 DETECTOR.....	54
2.1.3 MEDIOS DE PROPAGACIÓN.....	55
2.1.3.1 RUIDO BLANCO GAUSSIANO ADITIVO	56
2.1.3.2 PROPAGACIÓN MULTICAMINO.....	56
2.1.3.3 INTERFERENCIA INTERSIMBÓLICA.....	57
2.1.3.4 PARAMETROS DE CANAL INDOOR.....	66
2.1.3.5 CANAL DE DESVANECIMIENTO PLANO.....	68
2.1.3.6 CANAL CON SELETIVIDAD DE FRECUENCIA	68
2.1.3.7 CANAL IMPLEMENTADO: MULTIPASO CON DISTRIBUCION RAYLEIGH	68
2.1.1 SINCRONIZACIÓN DE TRAMA Y CORRECCIÓN DE FRECUENCIA.....	71
2.1.1.1 SINCRONIZACIÓN.....	71
2.1.1.2 SINCRONIZACIÓN DE TRAMA	72
2.1.1.3 SINCRONIZACIÓN DE TRAMA POR CORRELATOR	77

2.1.1.4 SINCRONIZACIÓN DE TRAMA SCHMILD & COX	78
2.1.1.5 CORRECCIÓN DE FRECUENCIA	83
2.1.2 DIAGRAMA DE OJO	86
CAPÍTULO 3.....	91
3.1 ALGORITMOS.....	92
3.1.1 MODULADOR	92
3.1.2 ADD CONTROL (Secuencia de Entrenamiento)	125
3.1.3 CANAL	135
3.1.4 SINCRONIZADOR DE TRAMA SCHMILD & COX.....	154
3.1.5 CORRECCIÓN DE FRECUENCIA Moose_ASC.....	169
CAPÍTULO 4.....	174
4.1 <i>FUNCIONAMIENTO DEL SINCRONIZADOR SCHMILD AND COX</i>	175
4.2 <i>COMPORTAMIENTO DEL SISTEMA FRENTE DIFERENTES CANALES</i>	180
4.2.1 CANAL AWGN.....	181
4.2.2 CANAL ISI.....	188
4.2.3 CANAL RAYLEIGH.....	194
4.3 <i>COMPORTAMIENTO DE LOS ALGORITMOS DE SINCRONIZACIÓN FRENTE DIFERENTES CANALES Y SECUENCIAS DE ENTRENAMIENTO</i>	203
4.3.1 CANAL AWGN.....	203

4.3.2 CANAL RAYLEIGH.....	209
4.4 DESEMPEÑO DEL SISTEMA AL INSERTAR UN DELAY AL CANAL	216
4.5 DESEMPEÑO DEL SISTEMA AL INSERTAR UN DELAY Y FRECUENCIA DE DESPLAZAMIENTO.....	222
6 CONCLUSIONES Y RECOMENDACIONES.....	235
CONCLUSIONES.....	235
RECOMENDACIONES	238
REFERENCIAS	240
ANEXOS A.....	242
ANEXOS B.....	243

ABREVIATURAS

NI	National Instruments
USRP	Universal Software Radio Peripheral
SDR	Software Defined Radio
QPSK	Quadrature Phase Shift Keying
AWGN	Additive White Gaussian Noise
ISI	Inter-Symbol Interference
BER	Bit Error Rate
SNR	Signal-to Noise Ratio
OFDM	Orthogonal Frequency Division Multiplexing
QAM	Quadrature Amplitude Modulation
RF	Radio Frequency
WSS	Wide Sence Stacionary
SAC	Schmidl and Cox
f_c	Frequency Carrier
VI	Virtual Instruments

ÍNDICE DE FIGURAS

Fig. 2.1. Sistema de Comunicación de Señales en Banda Base [1]	36
Fig. 2.2. Constelación QPSK	40
Fig. 2.3. Constelación 16 - QAM	42
Fig. 2.4. Constelación 64 - QAM	47
Fig. 2.5. Resultado de auto-correlación de las Secuencias a usar [18]	50
Fig. 2.7. Esquema de modelo de tres rayos para Canal ISI, basado en el Lab. De Simulación de Telecomunicaciones, Vista Superior.....	60
Fig. 2.8 Constelación canal ISI	65
Fig. 2.9 Respuesta Amplitud Vs. Tiempo canal ISI.....	65
Fig. 2.10 Respuesta en frecuencia del canal ISI	66
Fig. 2.11 Constelación canal Rayleigh	70
Fig. 2.12 Respuesta en el tiempo del canal Rayleigh.....	70
Fig. 2.13 Respuesta en la frecuencia del canal Rayleigh	71
Fig. 2.14. a) Trama OFDM compuesta por dos secuencias de entrenamiento (S.T), el prefijo cíclico (C.P) y los datos b) Trama de un sistema de una sola portadora...	81
Fig. 3.1. Diagrama de Bloques del Transmisor	92
Fig. 3.2. Diagrama de Bloques del Receptor.....	92
Fig. 3.3. Diagrama de Bloques del Modulador	93

Fig. 3.4. Control String Modulation Type	93
Fig. 3.5. Control Tipo Array	94
Fig. 3.6. Indicador Tipo Array Complex	94
Fig. 3.7. Ciclo While Loop	95
Fig. 3.8. Spilt 1D Array	95
Fig. 3.9. Not Equal To 0?	96
Fig. 3.10. Array Size	96
Fig. 3.11. Greater Or Equal?	96
Fig. 3.12. Index Array.....	97
Fig. 3.13. Array Constant	97
Fig. 3.14. Reverse 1D Array.....	97
Fig. 3.15. Expresiones Square Root y Divide	98
Fig. 3.16. Boolean Array To Number	98
Fig. 3.19. Condition While Loop	102
Fig. 3.20. Add Shift Registers	102
Fig. 3.21. Dividir la secuencia de entrada QPSK.....	103
Fig. 3.22. Invertir los bits a ser modulados.....	104
Fig. 3.23. Conversión de arreglo entero a booleano	105
Fig. 3.24. Conversión de booleano a entero	105
Fig. 3.25. Bits mapeados a sus respectivos símbolos QPSK	106

Fig. 3.26. Condición de Stop del Lazo While.....	107
Fig. 3.27. Diagrama de bloques del Sub VI - QPSK.....	108
Fig. 3.28. Diagrama de flujo del Sub VI 16-QAM	109
Fig. 3.29. Dividir la secuencia de entrada 16 - QAM	110
Fig. 3.30. Bits mapeados a sus respectivos símbolos 16 - QAM.....	113
Fig. 3.31. Normalizar los símbolos 16 - QAM.....	113
Fig. 3.32. Diagrama de bloques del Sub VI 16 - QAM.....	114
Fig. 3.33. Diagrama de flujo Sub VI 64 - QAM	115
Fig. 3.34. Dividir la secuencia de entrada 64 - QAM	116
Fig. 3.35. Dividir los 6 bits de 64 - QAM.....	117
Fig. 3.36. Invertir los bits a ser modulados 64 - QAM.....	118
Fig. 3.37. Crear datos enteros dado un arreglo booleano	119
Fig. 3.38. Arreglo de símbolos 64 – QAM región 0.....	120
Fig. 3.39. Arreglo de símbolos 64 – QAM región 1.....	121
Fig. 3.40. Arreglo de símbolos 64 – QAM región 2.....	122
Fig. 3.41. Arreglo de símbolos 64 – QAM región 3.....	123
Fig. 3.42. Normalizar los símbolos 64 – QAM	124
Fig. 3.43. Diagrama de bloques del Sub VI 64 – QAM.....	125
Fig. 3.44. Control Cluster Training Sequences.....	126
Fig. 3.45. Control Array Input	126

Fig. 3.46. Control Cluster Modulation Parameters In.....	127
Fig. 3.47. Indicador tramas con secuencia de entrenamiento.....	127
Fig. 3.48. Indicador Símbolos de secuencia de entrenamiento	128
Fig. 3.49. Indicador Cluster Modulation Parameters Out.....	128
Fig. 3.50. Insert Into Array.....	129
Fig. 3.51. Interleave 1D Array	129
Fig. 3.52. Array Constant (Willard Sequence)	129
Fig. 3.53. Diagramas de flujo VI Add Control	130
Fig. 3.54. Add Case After	131
Fig. 3.55. Secuencia Barker 16 – QAM.....	132
Fig. 3.56. Secuencia Barker 64 - QAM.....	133
Fig. 3.57. Duplicate Case Barker Sequence	134
Fig. 3.58. Arreglo de Bits Willard Sequence.....	135
Fig. 3.59. Control Noise Power	136
Fig. 3.60. Input Complex Waveform.....	136
Fig. 3.61. Output Complex Waveform	137
Fig. 3.62. Unbundle y Bundle By Name	138
Fig. 3.63. Negate	138
Fig. 3.64. RMS, Desviation & Variance y Mean.....	138
Fig. 3.65. Funciones Complex	139

Fig. 3.66. Funciones Numeric	139
Fig. 3.67. 10^x	139
Fig. 3.68. Gaussian White Noise.....	140
Fig. 3.69. Inverse Tangent	140
Fig. 3.70. Reshape Array	140
Fig. 3.71. Diagrama de flujo VI RayChannel	141
Fig. 3.72. Diagrama de flujo SUB VI Distr Ray.....	142
Fig. 3.73. Diagrama de flujo SUB VI Power Signal.....	142
Fig. 3.74. Diagrama de flujo SUB VI Delay	143
Fig. 3.75. Obtener dt y número de muestras.....	144
Fig. 3.76. Suma de varianza real e imaginaria de Y	144
Fig. 3.77. Suma de varianzas dividida para Noise Power	145
Fig. 3.78. Varianza que ingresa a DISTR RAY	146
Fig. 3.79. VI DISTR RAY	147
Fig. 3.80. Insertar DISTR RAY	147
Fig. 3.81. Sumar un valor RMS a la envolvente	148
Fig. 3.82. Comparar la media de la envolvente con menor a 0	149
Fig. 3.83. Extraer retardo	150
Fig. 3.84. VI DELAY	151
Fig. 3.85. Añadiendo Potencia del trayecto a la señal.....	152

Fig. 3.86. Shift Register inicializado	152
Fig. 3.87. VI POWER SIGNAL	153
Fig. 3.88. Formar la señal de salida de una canal Rayleigh	153
Fig. 3.89. Diagrama de bloques del Sub VI RayChannel	154
Fig. 3.90. Indicadores Numéricos	155
Fig. 3.91. Stacked Sequence Structure	155
Fig. 3.92. Decrement y Subtract	156
Fig. 3.93. Absolute Value	156
Fig. 3.94. Add Array Elements.....	156
Fig. 3.95. Select.....	157
Fig. 3.96. Max & Min Array.....	157
Fig. 3.97. Array Subset	157
Fig. 3.98. Search Array	158
Fig. 3.99. Complex Conjugate.....	158
Fig. 3.100. Get Date/ Time in Seconds	158
Fig. 3.101. Diagrama de flujo VI Schmild & Cox	159
Fig. 3.102. Obtener tiempo actual de la PC	160
Fig. 3.103. Descomponer cluster Y y secuencia de entrenamiento	161
Fig. 3.104. Valor de iteración N.....	161
Fig. 3.105. Generando $y[n+d]$ y $y[n+d+L]$	162

Fig. 3.106. Generando $R[d]$	163
Fig. 3.107. Generando $P[d]$	163
Fig. 3.108. Obtener Métrica $M[d]$	164
Fig. 3.109. Obtener el Índice del inicio de Trama	165
Fig. 3.110. Frame Offset	165
Fig. 3.111. Angulo $R[d]/P[d]$	166
Fig. 3.112. Insertar moose_ASC	167
Fig. 3.113. Tiempo de ejecución (ms)	168
Fig. 3.114. Diagrama de bloques del Sub VI SAC	168
Fig. 3.115. Reciprocal	169
Fig. 3.116. Diagrama de flujo VI Moose_ASC	170
Fig. 3.117. Obteniendo f	171
Fig. 3.118. Obtener parámetro $2\pi f T_{NT}$	171
Fig. 3.119. Obteniendo Frequency Offset y $E^{\wedge}$	172
Fig. 3.120. Trama corregida frecuencia de desplazamiento	173
Fig. 3.121. Diagrama de bloques del Sub VI Moose_ASC	173
Fig. 4.1. Constelación Modulada en QPSK	175
Fig. 4.2. Constelación antes y después de la Sincronización de Trama SC	176
Fig. 4.3. Constelación antes y después de la Sincronización de Trama SAC.....	177
Fig. 4.4. Inicio de Trama de SC y SAC respectivamente.....	177

Fig. 4.5. BER de los algoritmos SC y SAC respectivamente	178
Fig. 4.6 Métrica de correlación a) Grafica Completa de la Métrica b) Pico de la Métrica. Inicio de la trama.....	179
Fig. 4.7. SNR vs. BER de un canal AWGN [22].....	182
Fig. 4.8. SNR vs. BER de un canal AWGN Simulación	183
Fig. 4.9. SNR vs. BER de un canal AWGN Transmisión Real	184
Fig. 4.10. SNR vs. BER de un canal AWGN Transmisión Real y Simulación	185
Fig. 4.11. SNR vs. BER Canal AWGN Modulación QPSK Transmisión Real y Simulación	186
Fig. 4.12. SNR vs. BER Canal AWGN Modulación 16 - QAM Transmisión Real y Simulación	187
Fig. 4.13. SNR vs. BER Canal AWGN Modulación 64 - QAM Transmisión Real y Simulación	188
Fig. 4.14. SNR vs. BER de un canal ISI Simulación.....	189
Fig. 4.15. SNR vs. BER de un canal ISI Transmisión Real.....	190
Fig. 4.16. SNR vs. BER de un canal AWGN Transmisión Real y Simulación	191
Fig. 4.17. SNR vs. BER Canal ISI Modulación QPSK Transmisión Real y Simulación	192
Fig. 4.18. SNR vs. BER Canal ISI Modulación 16 - QAM Transmisión Real y Simulación	193

Fig. 4.19. SNR vs. BER Canal ISI Modulación 64 - QAM Transmisión Real y Simulación	194
Fig. 4.20. SNR vs. BER de un canal RAYLEIGH [22].....	195
Fig. 4.21. SNR vs. BER de un canal RAYLEIGH Simulación	196
Fig. 4.22. SNR vs. BER de un canal RAYLEIGH Transmisión Real	197
Fig. 4.23. SNR vs. BER de un canal RAYLEIGH Transmisión Real y Simulación .	198
Fig. 4.24. SNR vs. BER Canal RAYLEIGH Modulación QPSK Transmisión Real y Simulación	199
Fig. 4.25. SNR vs. BER Canal RAYLEIGH Modulación 16 - QAM Transmisión Real y Simulación	200
Fig. 4.26. SNR vs. BER Canal RAYLEIGH Modulación 64 - QAM Transmisión Real y Simulación	201
Fig. 4.27. SNR vs. BER QPSK canal AWGN Simulación- Transmisión Real SC y SAC Secuencia Barker	204
Fig. 4.28. SNR vs. BER 16 –QAM canal AWGN Simulación- Transmisión Real SC y SAC Secuencia Barker	205
Fig. 4.29. SNR vs. BER 64 - QAM canal AWGN Simulación- Transmisión Real SC y SAC Secuencia Barker	206
Fig. 4.30. SNR vs. BER QPSK canal AWGN Simulación- Transmisión Real SC y SAC Secuencia Willard.....	207

Fig. 4.31. SNR vs. BER 16 –QAM canal AWGN Simulación- Transmisión Real SC y SAC Secuencia Willard.....	208
Fig. 4.32. SNR vs. BER 64 - QAM canal AWGN Simulación- Transmisión Real SC y SAC Secuencia Willard.....	209
Fig. 4.33. SNR vs. BER QPSK canal RAYLEIGH Simulación- Transmisión Real SC y SAC Secuencia Barker	210
Fig. 4.34. SNR vs. BER 16 –QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC Secuencia Barker	211
Fig. 4.35. SNR vs. BER 64 - QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC Secuencia Barker	212
Fig. 4.36. SNR vs. BER QPSK canal RAYLEIGH Simulación- Transmisión Real SC y SAC Secuencia Willard.....	213
Fig. 4.37. SNR vs. BER 16 –QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC Secuencia Willard	214
Fig. 4.38. SNR vs. BER 64 - QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC Secuencia Willard	215
Fig. 4.39. Parámetros de Configuración del Simulador Sliding Correlator – Barker Sequence	218
Fig. 4.40. Parámetros de Configuración del Simulador Sliding Correlator – Willard Sequence	218
Fig. 4.41. Parámetros de Configuración del Simulador Schmild & Cox – Barker Sequence	219

Fig. 4.42. Parámetros de Configuración del Simulador Schmild & Cox – Willard	
Sequence	219
Fig. 4.43. Diagrama de Ojo antes y después del canal ISI.....	220
Fig. 4.44. Diagrama de Ojo Barker Sequence SC y SAC respectivamente	220
Fig. 4.45. Diagrama de Ojo Willard Sequence SC y SAC respectivamente.....	221
Fig. 4.46. Parámetros de Configuración del Simulador Sliding Correlator – Barker	
Sequence Sin Correct Frequency Offset.....	223
Fig. 4.47. Parámetros de Configuración del Simulador Sliding Correlator – Willard	
Sequence Sin Correct Frequency Offset.....	224
Fig. 4.48. Parámetros de Configuración del Simulador Sliding Correlator – Barker	
Sequence Correct Frequency Offset.....	225
Fig. 4.49. Parámetros de Configuración del Simulador Sliding Correlator– Willard	
Sequence Correct Frequency Offset.....	225
Fig. 4.50. Parámetros de Configuración del Simulador Schmild & Cox – Barker	
Sequence Sin Correct Frequency Offset.....	226
Fig. 4.51.. Parámetros de Configuración del Simulador Schmild & Cox– Willard	
Sequence Sin Correct Frequency Offset.....	227
Fig. 4.52. Parámetros de Configuración del Simulador Schmild & Cox – Barker	
Sequence Correct Frequency Offset.....	228
Fig. 4.53. Parámetros de Configuración del Simulador Schmild & Cox– Willard	
Sequence Correct Frequency Offset.....	228

Fig. 4.54. Antes y Después de Correct Frequency Offset.....	229
Fig. 4.55. Antes y Después de Correct Frequency Offset Constelación	230
Fig. 4.56. Frequency Offset Vs BER en transmisión real Sliding Correlator vs Schimdl and Cox con secuencia Barker.....	232
Fig. 4.57. Frequency Offset Vs BER en transmisión real Sliding Correlator vs Schimdl and Cox con secuencia Willard	233

ÍNDICE DE TABLAS

Tabla 2.1 Modulación QPSK	39
Tabla 2.2 Modulación 16 - QAM.....	41
Tabla 2.3 Modulación 64 - QAM.....	46
Tabla 2.4 Secuencia de código Barker [1]	48
Tabla 2.5 Secuencia de código Willard [18].....	49
Tabla 2.6 Parámetros de Canal Indoor.....	67
Tabla 2.7 Parámetros del Canal Rayleigh	67
Tabla 2.8 Código Barker y Willard Longitud 11 [1] y [18]	74

INTRODUCCIÓN

En busca de mejorar la enseñanza de las telecomunicaciones, se buscó la forma de implementar un nuevo laboratorio para el aprendizaje de las comunicaciones inalámbricas. Mediante la adquisición de nuevos equipos y el desarrollo de prácticas provistas en el manual surge la necesidad de dar más opciones para mejorar o generar variantes al sistema.

Basándonos en la práctica 5 del manual de prácticas de los USRP, se buscó nuevos algoritmos de sincronización de trama con el objetivo de estudiarlos y analizar su funcionamiento dentro un sistema de portadora única.

Encontrando así el algoritmo de Schmidl and Cox, una metodología que se aplica a sistemas que trabajan con OFDM (con múltiples portadoras), viéndonos en el reto de buscar la viabilidad de adaptar dicho algoritmo al sistema presentado en la práctica 5 del manual. Así como la generación de

perturbaciones para ver el correcto funcionamiento del algoritmo y compararlo con el de la práctica presente en cuestión.

CAPÍTULO 1

PLANTEAMIENTO DEL PROBLEMA

1.1 DESCRIPCIÓN DEL PROBLEMA

Los mensajes que se emiten a través de un medio, sufren ciertos retardos o desplazamientos en frecuencia producidos por los efectos de reflexión, dispersión y difracción durante su propagación dentro un ambiente, y más si este presenta condiciones de multi-trayectoria, y por el procesamiento de la señal en el receptor.

Esto provoca que la estructura inicial de la trama enviada se pierda produciendo una errónea recuperación del mensaje enviado.

En la búsqueda de poder realizar una recuperación de la información transmitida, se da la utilización de algoritmos de sincronización. Su objetivo es estimar las características necesarias para poder detectar y recuperar la señal transmitida originalmente. Particularizando el tema, nos enfocaremos en la sincronización de trama, es decir en buscar el inicio de la trama y poder estimar la estructura de la misma.

Basándonos en la práctica 5 del manual de laboratorio que viene con los USRP, buscamos un nuevo método para realizar el objetivo planteado en la práctica, encontrándonos con el algoritmo propuesto por Schmidl and Cox, en donde ellos muestran un algoritmo que utiliza las propiedades de auto-correlación para encontrar el inicio de la trama, siendo más robusto ante los cambios y alteraciones provocados por el canal.

Este algoritmo es desarrollado para sistemas OFDM (Orthogonal Frequency Division Multiplexing), siendo un método robusto y presenta, conjuntamente un proceso para corregir los desplazamientos de frecuencia. Viéndonos en la tarea de encontrar la forma de implementar y viabilidad que presenta el uso de éste algoritmo al sistema en que trabajamos.

En ambientes denominados multi-trayectoria, además de presentar retardos a la señal, estos conllevan a desplazamientos en la fase del mensaje a su llegada en el receptor y, en conjunto, provoca lo que se conoce como ISI (Intersymbol Interference). Estos desplazamientos se ven reflejados en la frecuencia provocando que los demás bloques siguientes, en la recepción, no presenten un funcionamiento correcto. Para esto en la búsqueda de un método de sincronización se incluirá un método de corrección para frecuencia desplazada.

Viendo esto el algoritmo Schmidl and Cox presenta una mejora en la metodología de corregir la frecuencia, convirtiendo al algoritmo una excelente opción para implementar en el sistema en que trabajamos.

1.2 OBJETIVOS GENERALES

El trabajo presentado tiene como planteamientos el desarrollar algoritmos en el software Labview, con la finalidad de desarrollar prácticas para el laboratorio de Comunicaciones Inalámbricas y el uso de los nuevos equipos USRP (Universal Software Radio Peripheral) para el estudio y enseñanza de las telecomunicaciones. Teniendo presente lo antes dicho planteamos como objetivos generales:

- Estudio del sistema de comunicación de portadora única propuesto por el manual.
- Explicación teórica de los algoritmos necesarios para el funcionamiento correcto del sistema.
- Simulación y explicación de los Vi's implementados junto con el funcionamiento de los USRP.

1.3 OBJETIVOS ESPECÍFICOS

Como objetivos específicos nos propusimos a:

- Verificar la viabilidad y funcionalidad de una adaptación del algoritmo de Schmidl and Cox que nosotros desarrollaremos.
- Comparar los dos métodos de Sincronización de trama verificar la funcionalidad de uno respecto al otro.
- Comprobar la robustez de los dos algoritmos con sus respectivas variantes, incluyendo la corrección de frecuencia.

- Ver cómo influyen el cambio de secuencia de entrenamiento ante los algoritmos de sincronización y corrección aquí presentados.
- Explicar el manejo de los USRP y LABVIEW para el desarrollo de aplicaciones en el área de telecomunicaciones
- Este Documento sirva de guía para futuros desarrollo de algoritmos y trabajos con los USPR.

1.4 JUSTIFICACIÓN DEL PROYECTO

Con la realización de este proyecto, se desea lograr que los estudiantes tengan una herramienta que permita evaluar cuáles serían los resultados obtenidos en un sistema de comunicación al variar parámetros tanto en el lado transmisor, en el canal de comunicación, así como en el lado del receptor (cambios en los tipos de modulación y demodulación, y tipo de ambiente de propagación generado por el canal).

Por otra parte, se busca adaptar un algoritmo de sincronización de trama de múltiples portadoras a portadora única, con el fin de incentivar a los estudiantes a investigar nuevos métodos más eficientes que permitan

una correcta detección de inicio de trama sin necesidad de conocer una secuencia de entrenamiento en el receptor.

Finalmente se busca alcanzar alta precisión en la simulación, permitiendo de esta manera un importante acercamiento a lo que ocurre en una transmisión real de un sistema de comunicaciones, brindando un mejor conocimiento de los elementos que conforman el sistema simulado.

1.5 DESCRIPCIÓN DEL PROYECTO

Después de desarrollar la práctica número 5 del manual de laboratorio que viene junto a los USPR, nos emprendimos en la búsqueda de otro algoritmo para poder implementar en el sistema y mostrándolo como una variante aplicable respecto al provisto por el manual.

Esto nos llevó al algoritmo de Schmidl & Cox, que es aplicado para sistemas de múltiples portadores (OFDM), en el cual nos toca comparar y adaptar ciertos parámetros para que se ajuste a nuestro sistema de única portadora.

Pero al mismo tiempo nos vimos en la tarea de ver cómo generar perturbaciones a la señal, ya que los equipos no están lo suficientemente

separados como para crear maquetas que simulen un ambiente de propagación, generamos algoritmos que emulen un ambiente multi-trayectoria con una distribución conocida. Diseñando así el canal Rayleigh.

Aprovechando los canales provistos en el sistema, se desarrolla un diseño para la utilización de canal ISI, utilizando al laboratorio de Comunicaciones Inalámbricas como lugar de pruebas. De esta manera se tendrá tres canales emulados, AWGN, ISI, Y RAYLEIGH, donde los dos primeros son los pre-establecidos en el sistema en que trabajamos y el último una implementación que nosotros desarrollamos.

Adicionalmente se crea otra variante: el cambio de la secuencia de entrenamiento, ya que los algoritmos trabajan con las propiedades de auto correlación de la secuencia de entrenamiento, observando cómo influye una secuencia diferente, con similares propiedades, en el funcionamiento de los algoritmos.

CAPÍTULO 2

SINCRONIZADOR DE TRAMA Y CORRECCIÓN DE FRECUENCIA DE PORTADORA

2.1 CONCEPTOS DE COMUNICACIÓN

En el presente capítulo describiremos los conceptos que envuelve a los sistemas y algoritmos utilizados en el desarrollo de nuestro tema.

Como todos sabemos un sistema de comunicación, en forma general, se compone de un transmisor que emite mi señal mensaje, un medio de

transmisión (canal o ambiente que recorre nuestra señal transmitida) y un receptor que recupera la señal mensaje.

2.1.1 SISTEMA DE COMUNICACIÓN DE SEÑALES EN BANDA BASE

Se denomina sistema de comunicación banda base aquellos sistemas que transmiten señales que trabajan en frecuencias que van desde 0 Hz hasta unos cuantos MHz, como la voz, videos y ciertas señales digitales [2].

Nos enfocaremos en las señales binarias provenientes de computadores o sistemas de procesamiento digital como convertidores análogos-digitales.

2.1.2 SISTEMA DE PORTADORA ÚNICA

Se denomina sistemas de portadora única a aquellas técnicas de comunicación analógica o digital en el que el mensaje se transmite mediante una sola frecuencia.

Figura 2.1 muestra el diagrama de bloques de un modem banda base digital, siendo este nuestra base para la implementación del proyecto.

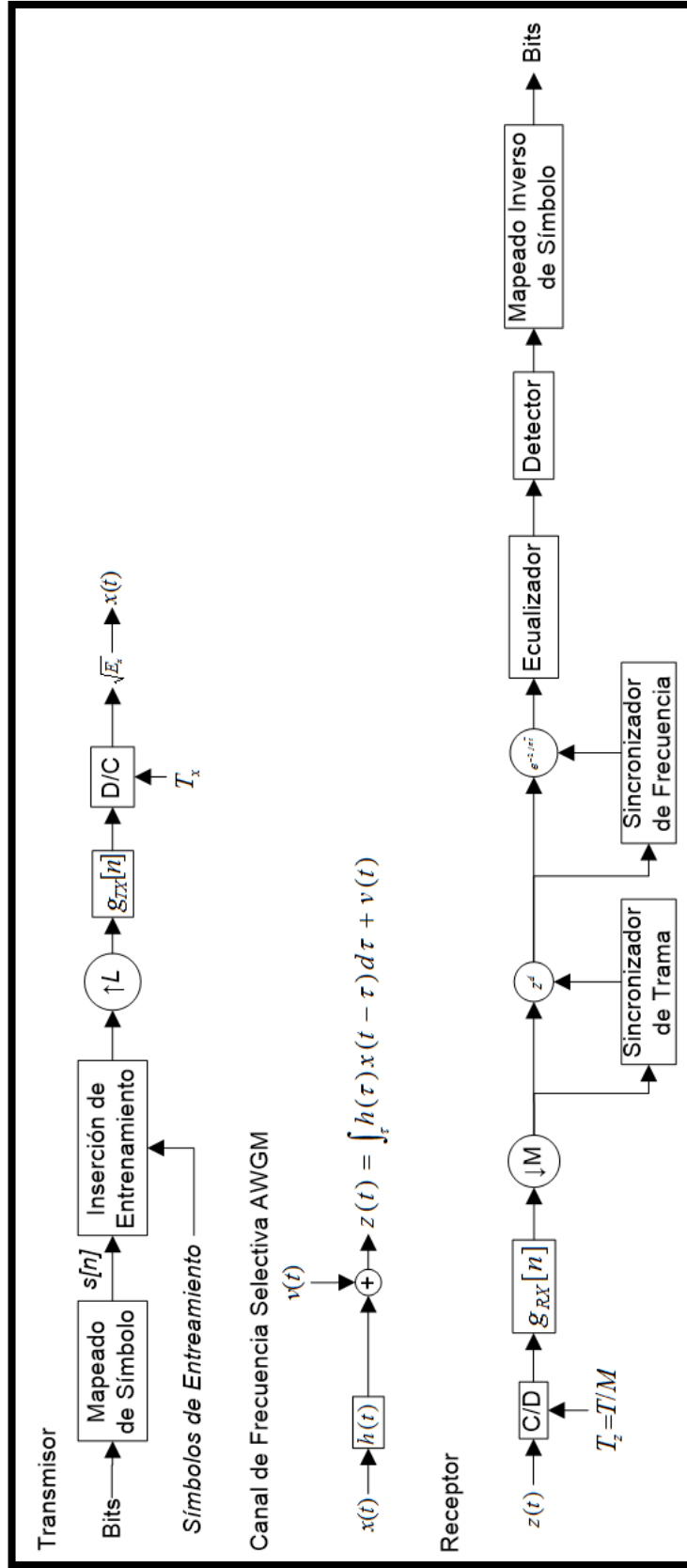


Fig. 3.1. Sistema de Comunicación de Señales en Banda Base [1]

2.1.2.1 MODULACIÓN Y DEMODULACIÓN

Modulación: es el proceso mediante el cual una señal portadora modifica alguna de sus características (frecuencia, amplitud, fase) según la señal mensaje que se va a transmitir.

Demodulación: proceso que recupera la señal mensaje basándose en las características modificadas de la señal portadora recibida.

El sistema transmisor trabaja con modulación por desplazamiento de fase o PSK (Phase Shift Key) y modulación de amplitud en cuadratura QAM (Quadrature Amplitude Modulation), que son modulaciones digitales en donde la información digital es mapeada dentro de símbolos complejos según el bit (o grupos de bits) de entrada. La modulación se realiza en el Bloque de Mapeo de Símbolo o Modulador.

El conjunto de símbolos formados durante el mapeo se denomina constelación. Entonces una secuencia de b_n

bits se mapea en símbolos s_n el cual corresponde a una constelación C , donde

$$C = \{s_0, s_1, s_2, \dots, s_{M-2}, s_{M-1}\} \quad (2.1)$$

Donde M es la cantidad de símbolos de la constelación, y se lo calcula $M = 2^b$, donde b es el número de bits que fueron mapeados.

El receptor recibe la señal y realiza el proceso inverso, la desmodula, realizando la recuperación de los bits transmitidos

QPSK: La modulación por desplazamiento cuaternario de fase es un tipo de modulación M-aria en donde el factor M es igual a 4 lo que indica 4 fases de salida para una sola frecuencia portadora. Debido a que hay 4 fases de salida diferentes, tiene que haber cuatro condiciones de entrada diferentes.

Se necesitan 2 bits en la entrada del modulador para producir 4 posibles condiciones 00, 01, 10 y 11 a la salida, los datos de entrada se combinan en grupos de 2 bits llamados di-bits, cada código di-bits genera cuatro fases de entrada como se muestra en la Tabla 2.1.

Di-bits		Magnitud	Fase	Símbolo
0	0	$\frac{\sqrt{2}}{2}$	$+45^\circ$	$1 + 1j$
0	1	$\frac{\sqrt{2}}{2}$	-45°	$1 - 1j$
1	0	$\frac{\sqrt{2}}{2}$	$+135^\circ$	$-1 + 1j$
1	1	$\frac{\sqrt{2}}{2}$	-135°	$-1 - 1j$

Tabla 3.1 Modulación QPSK

La figura 2.2 muestra la constelación de QPSK.

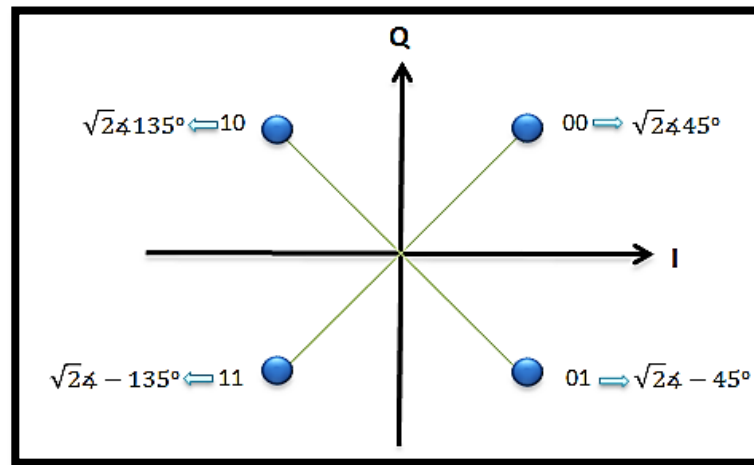


Fig. 3.2. Constelación QPSK

16-QAM: Es un sistema M-ario, en donde $M=16$. Actúa sobre los datos de entrada en grupos de cuatro. Como con 8-QAM, tanto la fase y la amplitud de la portadora transmisora son variados.

Como se puede observar en la figura 2.3, la señal modulada tiene variaciones en amplitud y fase. En cada cuadrante existen tres valores de fase y de amplitud diferentes. Por lo que cada punto de la constelación representa 4 *bits*, y en total existen 16 puntos, de ahí el nombre de 16-QAM. La representación de cada combinación de los 4 *bits* se presenta en la tabla 2.2.

4 Bits				Magnitud	Fase	Símbolo
0	0	0	0	$\sqrt{2}$	$+45^\circ$	$1 + 1j$
0	0	0	1	$\sqrt{10}$	$+72^\circ$	$1 + 3j$
0	0	1	0	$\sqrt{2}$	-45°	$1 - 1j$
0	0	1	1	$\sqrt{10}$	-72°	$1 - 3j$
0	1	0	0	$\sqrt{10}$	$+18^\circ$	$3 + 1j$
0	1	0	1	$\sqrt{18}$	45°	$3 + 3j$
0	1	1	0	$\sqrt{10}$	-18°	$3 - 1j$
0	1	1	1	$\sqrt{18}$	-45°	$3 - 3j$
1	0	0	0	$\sqrt{2}$	$+135^\circ$	$-1 + 1j$
1	0	0	1	$\sqrt{10}$	$+108^\circ$	$-1 + 3j$
1	0	1	0	$\sqrt{2}$	-135°	$-1 - 1j$
1	0	1	1	$\sqrt{10}$	-108°	$-1 - 3j$
1	1	0	0	$\sqrt{10}$	$+162^\circ$	$-3 + 1j$
1	1	0	1	$\sqrt{18}$	$+135^\circ$	$-3 + 3j$
1	1	1	0	$\sqrt{10}$	-162°	$-3 - 1j$
1	1	1	1	$\sqrt{18}$	-135°	$-3 - 3j$

Tabla 3.2 Modulación 16 - QAM

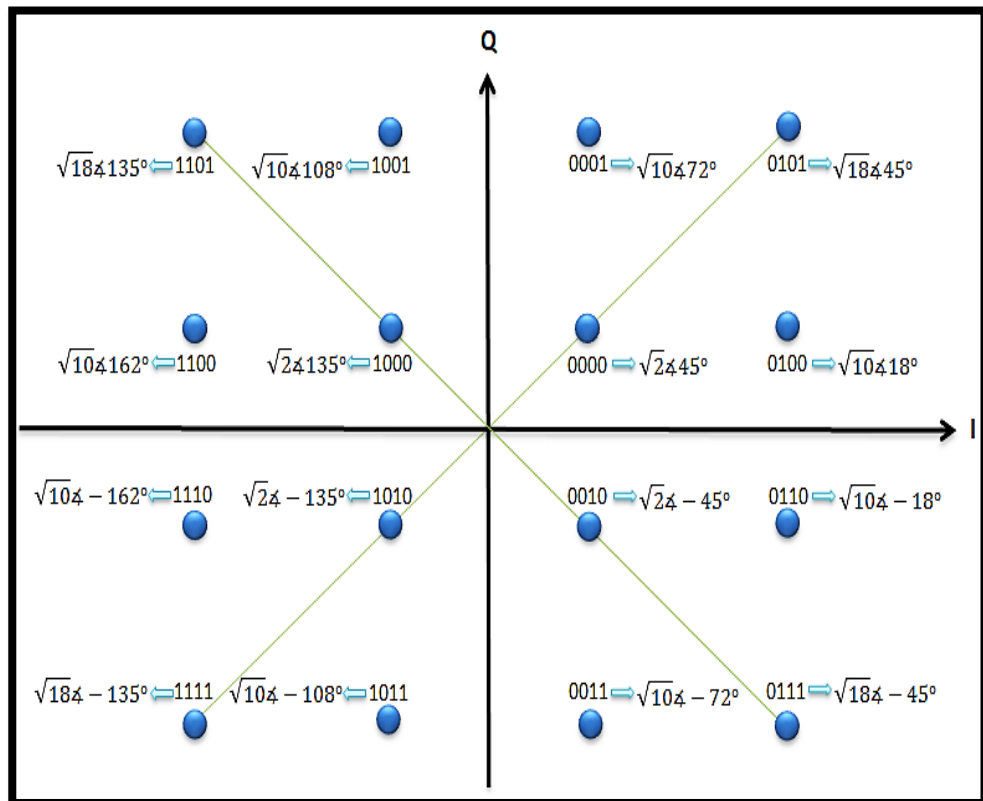


Fig. 3.3. Constelación 16 - QAM

64-QAM: Esta técnica de modulación digital es principalmente usada para enviar datos sobre el canal downstream en redes de cable coaxial. Es una técnica muy eficiente, soporta velocidades de transmisión arriba de 28 Mbps sobre un canal simple de 6 MHz. Aunque es susceptible a señales de interferencia, lo que hace que no sea usado en el canal upstream, ya que es muy sensible al ruido. El diagrama de constelación de la modulación 64-QAM se presenta en la figura 2.4.

En cada cuadrante existen 16 puntos. Cada punto de la constelación representa 6 bits, y en total existen 64 puntos, de ahí el nombre de 64-QAM. La representación de cada combinación de los 6 bits se presenta en la tabla 2.3.

6 Bits						Magnitud	Símbolo
0	0	0	0	0	0	$\overline{98}$	$-7 + 7j$
0	0	0	0	0	1	$\overline{74}$	$-7 + 5j$
0	0	0	0	1	0	$\overline{50}$	$-7 + 1j$
0	0	0	0	1	1	$\overline{58}$	$-7 + 3j$
0	0	0	1	0	0	$\overline{98}$	$-7 - 7j$
0	0	0	1	0	1	$\overline{74}$	$-7 - 5j$
0	0	0	1	1	0	$\overline{50}$	$-7 - 1j$
0	0	0	1	1	1	$\overline{58}$	$-7 - 3j$
0	0	1	0	0	0	$\overline{74}$	$-5 + 7j$
0	0	1	0	0	1	$\overline{50}$	$-5 + 5j$
0	0	1	0	1	0	$\overline{26}$	$-5 + 1j$
0	0	1	0	1	1	$\overline{34}$	$-5 + 3j$

0	0	1	1	0	0	$\overline{74}$	$-5 - 7j$
0	0	1	1	0	1	$\overline{50}$	$-5 - 5j$
0	0	1	1	1	0	$\overline{26}$	$-5 - 1j$
0	0	1	1	1	1	$\overline{34}$	$-5 - 3j$
0	1	0	0	0	0	$\overline{50}$	$-1 + 7j$
0	1	0	0	0	1	$\overline{26}$	$-1 + 5j$
0	1	0	0	1	0	$\overline{2}$	$-1 + 1j$
0	1	0	0	1	1	$\overline{10}$	$-1 + 3j$
0	1	0	1	0	0	$\overline{50}$	$-1 - 7j$
0	1	0	1	0	1	$\overline{26}$	$-1 - 5j$
0	1	0	1	1	0	$\overline{2}$	$-1 - 1j$
0	1	0	1	1	1	$\overline{10}$	$-1 - 3j$
0	1	1	0	0	0	$\overline{58}$	$-3 + 7j$
0	1	1	0	0	1	$\overline{34}$	$-3 + 5j$
0	1	1	0	1	0	$\overline{10}$	$-3 + 1j$
0	1	1	0	1	1	$\overline{18}$	$-3 + 3j$
0	1	1	1	0	0	$\overline{58}$	$-3 - 7j$
0	1	1	1	0	1	$\overline{34}$	$-3 - 5j$

0	1	1	1	1	0	$\overline{10}$	$-3 - 1j$
0	1	1	1	1	1	$\overline{18}$	$-3 - 3j$
1	0	0	0	0	0	$\overline{98}$	$7 + 7j$
1	0	0	0	0	1	$\overline{74}$	$7 + 5j$
1	0	0	0	1	0	$\overline{50}$	$7 + 1j$
1	0	0	0	1	1	$\overline{58}$	$7 + 3j$
1	0	0	1	0	0	$\overline{98}$	$7 - 7j$
1	0	0	1	0	1	$\overline{74}$	$7 - 5j$
1	0	0	1	1	0	$\overline{50}$	$7 - 1j$
1	0	0	1	1	1	$\overline{58}$	$7 - 3j$
1	0	1	0	0	0	$\overline{74}$	$5 + 7j$
1	0	1	0	0	1	$\overline{50}$	$5 + 5j$
1	0	1	0	1	0	$\overline{26}$	$5 + 1j$
1	0	1	0	1	1	$\overline{34}$	$5 + 3j$
1	0	1	1	0	0	$\overline{74}$	$5 - 7j$
1	0	1	1	0	1	$\overline{50}$	$5 - 5j$
1	0	1	1	1	0	$\overline{26}$	$5 - 1j$
1	0	1	1	1	1	$\overline{34}$	$5 - 3j$

1	1	0	0	0	0	$\overline{50}$	$1 + 7j$
1	1	0	0	0	1	$\overline{26}$	$1 + 5j$
1	1	0	0	1	0	$\overline{2}$	$1 + 1j$
1	1	0	0	1	1	$\overline{10}$	$1 + 3j$
1	1	0	1	0	0	$\overline{50}$	$1 - 7j$
1	1	0	1	0	1	$\overline{26}$	$1 - 5j$
1	1	0	1	1	0	$\overline{2}$	$1 - 1j$
1	1	0	1	1	1	$\overline{10}$	$1 - 3j$
1	1	1	0	0	0	$\overline{58}$	$3 + 7j$
1	1	1	0	0	1	$\overline{34}$	$3 + 5j$
1	1	1	0	1	0	$\overline{10}$	$3 + 1j$
1	1	1	0	1	1	$\overline{18}$	$3 + 3j$
1	1	1	1	0	0	$\overline{58}$	$3 - 7j$
1	1	1	1	0	1	$\overline{34}$	$3 - 5j$
1	1	1	1	1	0	$\overline{10}$	$3 - 1j$
1	1	1	1	1	1	$\overline{18}$	$3 - 3j$

Tabla 3.3 Modulación 64 - QAM

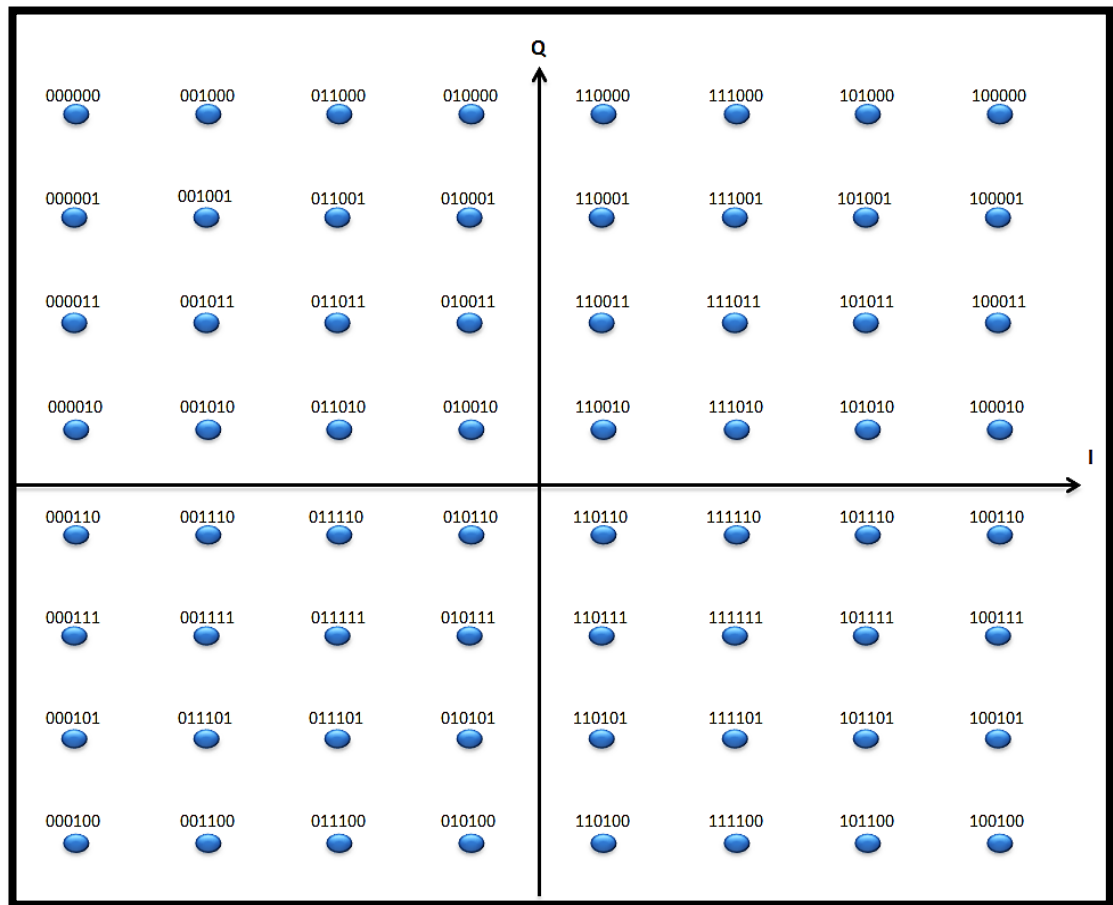


Fig. 3.4. Constelación 64 - QAM

2.1.2.2 INSERCIÓN DE SÍMBOLOS DE ENTRENAMIENTO

El transmisor genera unos símbolos que representan una secuencia. Esta secuencia se inserta al inicio de los símbolos de datos y se conocen como secuencia de entrenamiento.

SECUENCIA DE ENTRENAMIENTO

Las secuencias de entrenamiento son tramas de bits (palabras) que poseen grandes propiedades de auto correlación y correlación cruzada. Estas son utilizadas para la sincronización y la estimación de canal.

La secuencia es conocida tanto en el transmisor como en el receptor. Se inserta al inicio de la trama de los datos modulados antes de ser enviados al canal.

Longitud del código	Secuencia Barker
2	[-+,--]
3	[--+]
4	[--+-,-+++]
5	[---+-]
7	[---++-+]
11	[---++++-++-+]
13	[-----++-++-+]

Tabla 3.4 Secuencia de código Barker [1]

Código Barker: es la secuencia de entrenamiento con buenas propiedades de auto correlación. Como se

observa en la tabla 2.4, donde -1 y +1 son representados por – y + respectivamente.

Códigos Willard: bajo ciertas condiciones, ofrecen un mejor rendimiento que los códigos Barker. Una lista de los códigos de Willard se tabula en la tabla 2.5.

Longitud	Willard Sequence
3	110
4	1100
5	11010
7	1110100
11	11101101000
13	1111100101000

Tabla 3.5 Secuencia de código Willard [18]

Las versiones invertidas o bits invertidos de los códigos que se muestran en la tabla 2.5 se pueden usar ya que todavía mantienen las propiedades de auto-correlación deseada.

La auto-correlación de los códigos de Barker y Willard con longitud $N=13$ se muestra en la figura 2.5, respectivamente.

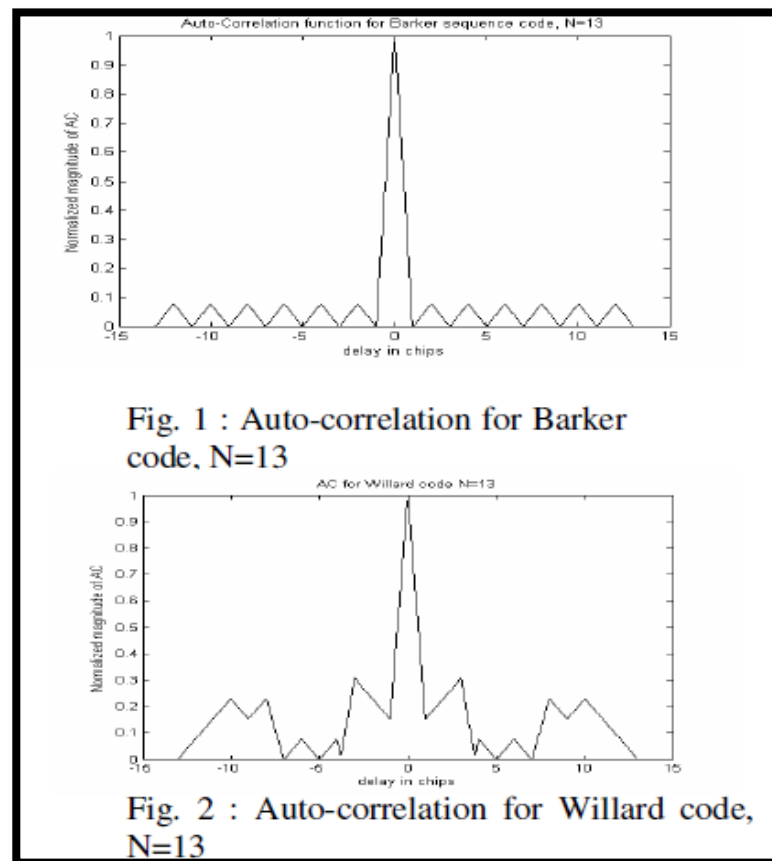


Fig. 3.5. Resultado de auto-correlación de las Secuencias a usar [18]

2.1.2.3 SOBRE MUESTREO (OVERSAMPLING)

Son técnicas para comprimir el espectro de la señal que se envía al canal, evitando el alaising y pérdidas de

información, a través de la inserción de símbolos ceros en la trama transmitida.

La inserción se realiza en el dominio del tiempo, esto quiere decir que el tiempo de duración de la señal se incrementa lo que representa una disminución de su espectro en el dominio de la frecuencia.

Esto nos asegura que la señal tendrá mayor probabilidad de pasar a través del canal sin sufrir ningún desvanecimiento o atenuación surgida por el ancho de banda del mismo.

UPSAMPLING: Este proceso se realiza en el transmisor, siendo la inserción de ceros $0 + 0j$ entre el símbolo incrementado la tasa de muestreo de la señal, esto nos asegura en buen funcionamiento del Filtro Formador de Pulsos.

DOWNSAMPLING: Una vez recibida la señal en el receptor y después del Filtro de Acoplamiento se procede

al retiro de los ceros para recuperar mi señal original. Esto se conoce como downsampling. Es decir reducir la tasa de muestreo de la misma.

La señal resultante después del upsampling sería representada como $s[n]$, una señal discreta en el tiempo.

2.1.2.4 FILTRO FORMADOR DE PULSOS Y FILTRO DE ACOPLAMIENTO

Los símbolos formados deben ser convertidos en una señal analógica en el transmisor y en el receptor la conversión inversa. Este proceso lo realiza el Filtro formador de Pulsos y el Filtro de acoplamiento respectivamente.

FILTRO FORMADOR DE PULSOS: Dentro de transmisor es un filtro que nos permite construir un tren de pulsos, tomando como base a los símbolos a transmitir, determinado para evitar la ISI, volviendo más confiable el sistema.

FILTRO DE ACOPLAMIENTO: En el receptor después del convertidor Continuo-Discreto, el filtro de acoplamiento elimina la forma de pulso de la señal devolviendo los coeficientes que son los símbolos de datos.

2.1.2.5 CONVERTIDORES

En el Transmisor la señal saliente del Formador de pulsos, es del tipo discreta, esta ingresa a un convertidor Discreto-a-Continuo obteniendo así una señal RF (radio frecuencia) lista para ser enviada. En la última etapa se incrementa la potencia de la señal, en un valor deseado, para ser transmitido a través del medio inalámbrico (canal de comunicación).

En el receptor, el convertidor realiza el proceso inverso, es decir que convierte la señal de entrada, que es una señal continua, en una señal discreta. Recalco que la señal de ingreso es la señal que proviene del medio inalámbrico.

2.1.2.6 ECUALIZADOR

El Bloque ecualizador lo que busca es la eliminación del ISI (Interferencia inter-simbólica) y las afectaciones producidas por el canal a la señal de entrada.

2.1.2.7 DETECTOR

El detector o bloque de decisión es el que realiza la estimación más aproximada de los símbolos de datos reales transmitidos y los envía al demodulador o al Mapeado inverso de símbolos. Donde se realiza la recuperación de los bits enviados.

Teniendo presente las definiciones explicadas anteriormente, tenemos que la secuencia de símbolos enviados, de nuestro sistema, a través de medio inalámbrico está compuesto de:

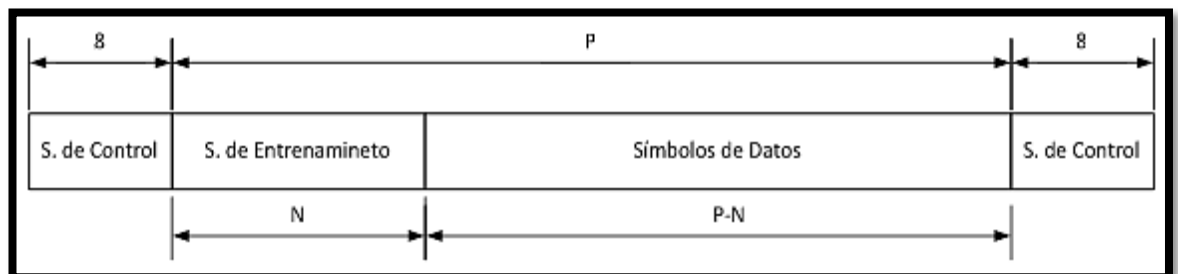


Fig. 3.6. Trama a la salida del Transmisor

Esta es la estructura de la señal transmitida. No modifica su forma al cambiar la modulación de los bits de datos.

La trama tiene una longitud P , la secuencia de entrenamiento N_t y una secuencia de datos de $P - N_t$. Adicionalmente, durante el proceso de transmisión se añaden Símbolos de control correspondientes al Zero Pad Length. Son ocho bits “0” modulados BPSK que se insertan al inicio y al final de la trama.

2.1.3 MEDIOS DE PROPAGACIÓN

Los medios o canales de propagación es el ambiente por el cual nuestro mensaje deberá atravesar para llegar a nuestro receptor.

La utilidad de poder simular estos medios de propagación es la facilidad con que nos permite generar perturbaciones e imperfecciones a las señales transmitidas, como si fueran afectados por un medio real facilitando el estudio y análisis de funcionamiento de nuestro sistema ante los mismos.

2.1.3.1 RUIDO BLANCO GAUSSIANO ADITIVO

Es un canal el cual añade ruido, pérdidas y desvanecimientos de la señal transmitida. Es el modelo de canal más sencillo que se aplica a un sistema. Es generado, en su mayoría, por el receptor.

Se representa como la suma de pequeña señales que perturbaban a la señal mensaje. Éstas se originan debido al calor de los circuitos en el receptor y perturbaciones electromagnéticas

2.1.3.2 PROPAGACIÓN MULTICAMINO

La propagación multi-camino hace referencia a la combinación de señales provenientes de reflexiones, dispersiones y difracciones producidas por edificios, árboles u personas (ambientes exteriores y outdoors) y por paredes u objetos dentro de una oficina o casa (ambientes interiores o indoors), en la antena del receptor.

DISTRIBUCIÓN RAYLEIGH: Distribución estadística que muestra la naturaleza aleatoria de un ambiente multi-camino sin línea de vista (NLOS, Non-line-of-sight), en

donde cada camino de arribo afecta la amplitud de la señal que transcurre sobre él según la siguiente fórmula [12 y 10]:

$$p(r) = \frac{r}{\sigma^2} e^{-r^2 / 2\sigma^2} \quad (3.2)$$

Dónde:

- r : Representa la magnitud de una variable aleatoria compleja gaussiana.
- σ : Es la variación estándar de la variable aleatoria compleja gaussiana, siendo σ^2 la potencia de la misma.

2.1.3.3 INTERFERENCIA INTERSIMBÓLICA

Debido a los efectos multi-camino del canal de propagación, la señal recibida, en comparación con la señal transmitida, presenta desplazamientos en el tiempo de los símbolos, generando que estos se filtren o traslape

unos con otros. A esto se conocen como Interferencia Inter Simbólica (ISI, Interference Intersymbol).

Este problema limita la transmisión de la tasa de bit's del canal, impidiendo aprovechar el ancho de banda del mismo.

El sistema en donde desarrollamos nuestro proyecto viene implementado un canal ISI, requiere el ingreso de valores complejos, $(a + bj)$, que representan la respuesta del canal, es decir una representación fasorial, $A_r e^{j\phi_i}$, de un ambiente multi-camino, donde A_r es la amplitud de la señal a una distancia d y $\phi_r = -2\pi d / \lambda$ es la fase de llegada de la señal [20].

Como es un ambiente multi-camino, esta respuesta recibida se compone de la suma de todas las señales posibles a través de los diferentes caminos que puede la señal recorrer. Entonces siendo L el número de caminos que la onda recorre, d_i la distancia del i -ésimo camino

con su respectivo factor de reflexión a_i , se define a la amplitud y fase de la señal como:

$$A_r e^{j\phi_r} = A_0 \prod_{i=1}^L \frac{a_i}{d_i} e^{j\phi_i} \quad (3.3)$$

En donde $\phi_i = -2\pi d_i / \lambda$.

Para nuestras pruebas se realizara un diagrama de tres rayos utilizando las dimensiones del laboratorio de simulación de telecomunicaciones.

Para el desarrollo de este modelo se asume que cada equipo se encuentra uno frente al otro, en cada extremo del laboratorio, a la misma distancia de la pared. Tengamos en cuenta que las antenas son omnidireccionales por lo que se asumirá que los equipos están a la misma altura respecto al suelo y están colocados de tal forma que las antenas son apuntadas hacia el techo.

Se realizaran tres rayos tal y como se describe en la figura 2.7

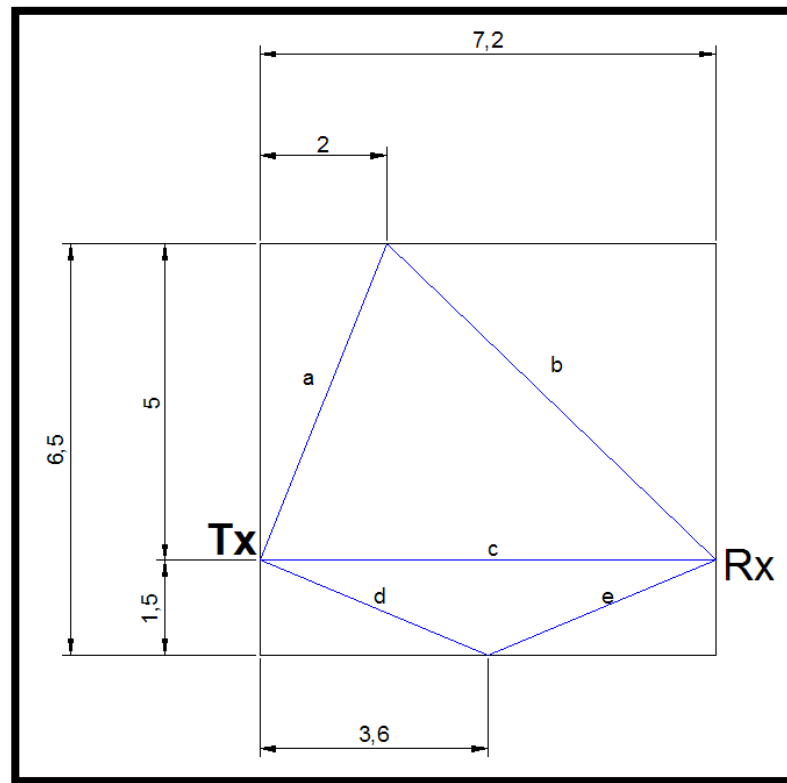


Fig. 3.7. Esquema de modelo de tres rayos para Canal ISI, basado en el Lab. De Simulación de Telecomunicaciones, Vista Superior.

La Figura 2.7 muestra la vista superior del laboratorio de simulación, esto se entiende que uno observara el laboratorio desde el techo hacia el piso. Las líneas de color azul corresponden a las trayectorias que las señales

cruzarán hasta llegar al receptor (Rx) desde el transmisor (Tx). Todas las medidas están en metros.

Dibujando un rayo de línea directa y dos reflejados en cada pared del laboratorio. Reescribiendo la ecuación anterior para un $L = 3$ tenemos:

$$A_r e^{j\phi_r} = A_0 \frac{a_1}{d_1} e^{j\phi_1} + \frac{a_2}{d_2} e^{j\phi_2} + \frac{a_3}{d_3} e^{j\phi_3} \quad (3.4)$$

Quedando los tres fasores que se colocaran el canal ISI:

$$\text{Fasor 1} = A_0 \frac{a_1}{d_1} e^{j\phi_1} \quad (3.5)$$

$$\text{Fasor 2} = A_0 \frac{a_2}{d_2} e^{j\phi_2} \quad (3.6)$$

$$\text{Fasor 3} = A_0 \frac{a_3}{d_3} e^{j\phi_3} \quad (3.7)$$

Donde el Fasor 1 corresponde a la componente de línea de vista. Calculando las distancias que recorren cada onda tenemos:

$$d_1 = c = 7.2 \text{ m} \quad (3.8)$$

$$d_2 = a + b = \sqrt{2^2 + 5^2} + \sqrt{5.2^2 + 5^2} = 12.5990 \text{ m} \quad (3.9)$$

$$d_3 = d + e = 2e = 2\sqrt{1.5^2 + 3.6^2} = 7.8 \text{ m} \quad (3.10)$$

Los coeficientes de atenuación se basaran en un caso similar que se estudia en el libro de Pahlavan [20] siendo estos:

$$a_1 = 1 \quad (3.11)$$

$$a_2 = a_3 = -0.7 \quad (3.12)$$

Para calcular los ángulos de desfase de los caminos primero obtenemos la longitud de onda, esta se calcula en base de la frecuencia portadora, que para nuestro sistema es $f_c = 915$ MHz, recordando que la señal en el aire viaja a la velocidad de la luz, $c = 3 \times 10^8$ m/s obtenemos que $\lambda = 0.3278$ m. Calculando los desfases tenemos:

$$\phi_1 = -2\pi d_1 / \lambda = -2\pi 7.2 / 0.3278 = -138.007^\circ \quad (3.13)$$

$$\phi_2 = -2\pi d_2 / \lambda = -2\pi 12.5990 / 0.3278 = -241.4943^\circ \quad (3.14)$$

$$\phi_3 = -2\pi d_3 / \lambda = -2\pi 7.8 / 0.3278 = -149.51^\circ \quad (3.15)$$

La potencia a un metro se la midió con el NARDA y se obtuvo que esta es de $P_0 = 14.72$ dB $\frac{mV}{m}$, entonces calculando la amplitud de la señal a un metro se obtiene:

$$A_0 = 10^{\frac{14.72}{20}} = 5.445 \quad (3.16)$$

Con estos valores obtenemos los fasores deseados:

$$\begin{aligned}\text{Fasor 1} &= 5.445 \frac{1}{7.2} e^{j -138.0077} = 0.75625 e^{j 221.9923} \\ &= -0.562 - j0.5059\end{aligned}\quad (3.17)$$

$$\begin{aligned}\text{Fasor 2} &= 5.445 \frac{-0.7}{12.5990} e^{j -241.4943} = 0.32246 e^{j 241.4943} \\ &= -0.138 - j0.2833\end{aligned}\quad (3.18)$$

$$\begin{aligned}\text{Fasor 3} &= 5.445 \frac{-0.7}{7.8} e^{j -149.51} = 0.048865 e^{j 149.51} \\ &= -0.42 + j0.2479\end{aligned}\quad (3.19)$$

Y así obtenemos los parámetros de respuesta de canal para un ambiente ISI con tres desplazamientos para una propagación interior.

Respuesta del canal: Emitiendo un pulso con un muestreo igual al que se trabaja en nuestro sistema obtenemos:

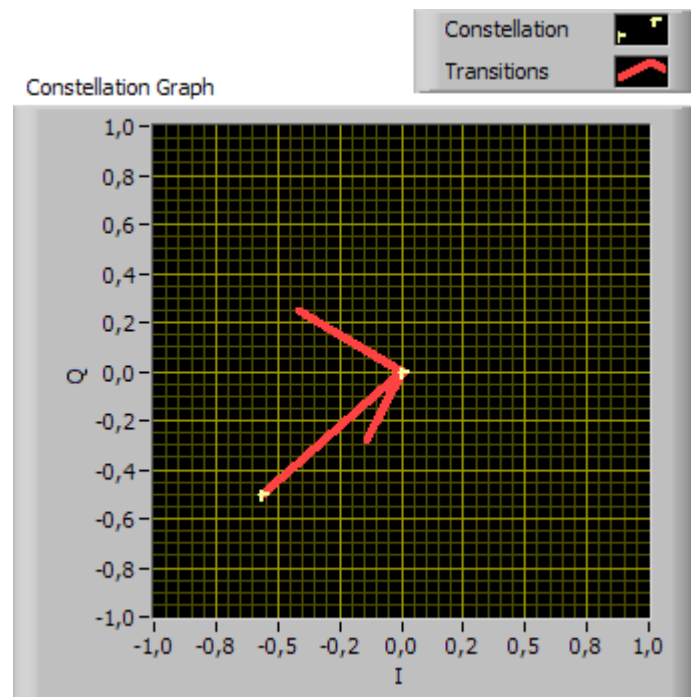


Fig. 2.8 Constelación canal ISI

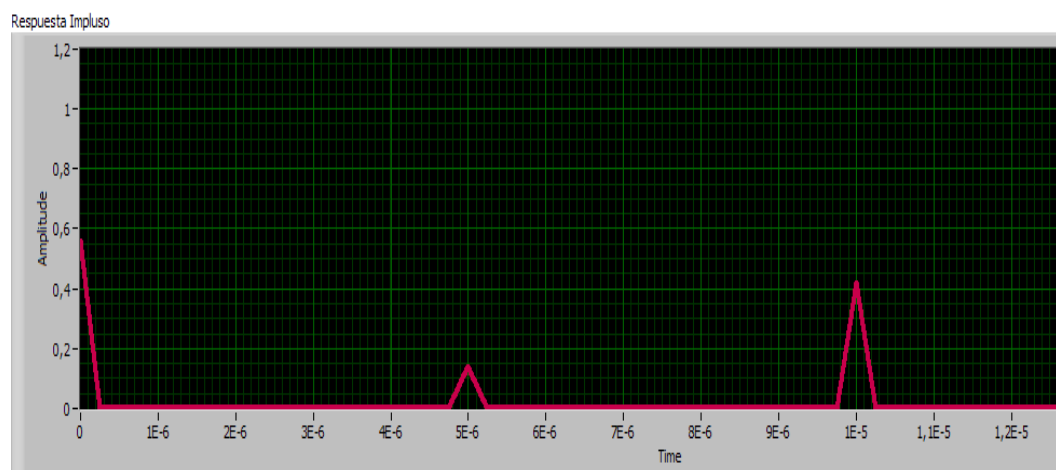


Fig. 2.9 Respuesta Amplitud Vs. Tiempo canal ISI

Como se aprecian en la constelación y la señal recuperada, el presenta diversos desplazamientos y diferentes magnitudes, las cuales corresponde al ángulo y a la componente real de cada fasor respectivamente.

Cada pulso de llegada corresponde a cada fasor, los cuales esta separados la misma distancia.

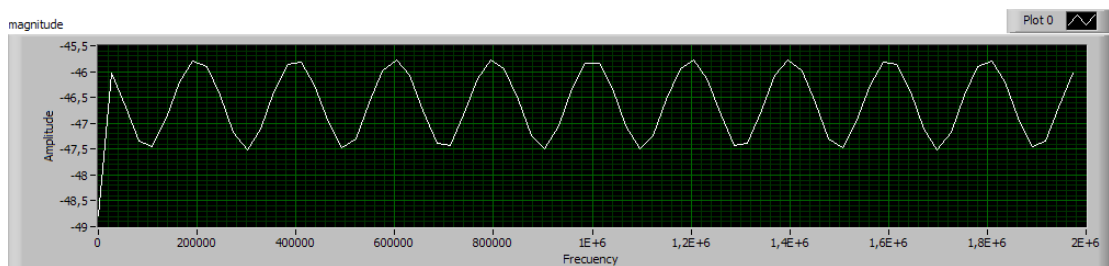


Fig. 2.10 Respuesta en frecuencia del canal ISI

2.1.3.4 PARAMETROS DE CANAL INDOOR

Se usara como referencia el siguiente canal indoor establecido por la ITU en el “Recomendation ITU-R M.1225” [21]:

<u>Tap's</u>	Canal tipo B	
	Retardo (ns)	Potencia (dB)
1	0	0
2	100	-33.6
3	200	-7.2
4	300	-10.8
5	500	-18.0
6	700	-25.2

Tabla 3.6 Parámetros de Canal Indoor

Modificando los parámetros para la implementación del canal Rayleigh acorde a nuestro sistema se plantea los siguientes valores:

<u>Tap's</u>	Basado en Canal tipo B	
	Retardo (us)	Potencia (dB)
1	1	0
2	2	-3.6
3	3	-7.2
4	5	-10.8
5	7	-25.2

Tabla 2.7 Parámetros del Canal Rayleigh

2.1.3.5 CANAL DE DESVANECIMIENTO PLANO

Se dice de aquellos canales en donde el tiempo de símbolo es mayor al tiempo de desvanecimiento por retardo. Interpretándose como el ancho de banda de la señal es menor que el ancho de banda del canal.

2.1.3.6 CANAL CON SELETIVIDAD DE FRECUENCIA

Se denomina a los canales con selectividad en frecuencia cuando el desvanecimiento por retardo está en el mismo orden o mayor que la duración de un símbolo, lo que representa a que el ancho de banda del canal es menor al ancho de banda de la señal transmitida.

2.1.3.7 CANAL IMPLEMENTADO: MULTIPASO CON DISTRIBUCION RAYLEIGH

Basándonos en la distribución Rayleigh se generan la respuesta del canal, el cual se implementara basando en la siguiente formula:

$$y(t) = h(t) * s(t) \quad (3.20)$$

Siendo $h(t)$ la respuesta del canal donde:

$$h(t) = \sum_{i=1}^L a_{p_i} \delta(t - \tau_i) \quad (3.21)$$

Donde L es la cantidad de pasos que tienen el canal, a_{p_i} la variable con el desvanecimiento Rayleigh en el i -ésimo paso y $\delta(t - \tau_i)$ el retardo en el i -ésimo paso.

Siendo a_{p_i} :

$$a_p = \frac{a_i * f_{\text{rayleigh}}}{\sum_{i=1}^N a_i^2}^{1/2} \quad (3.22)$$

$$a_i = 10^{-G_i(\text{dB})/20} \quad (3.23)$$

Donde a_i es la amplitud del i-ésimo paso y f_{rayleigh} es la variable con distribución Rayleigh. Utilizando la tabla 2.7 de parámetros descrita se obtiene la respuesta del canal implementado:

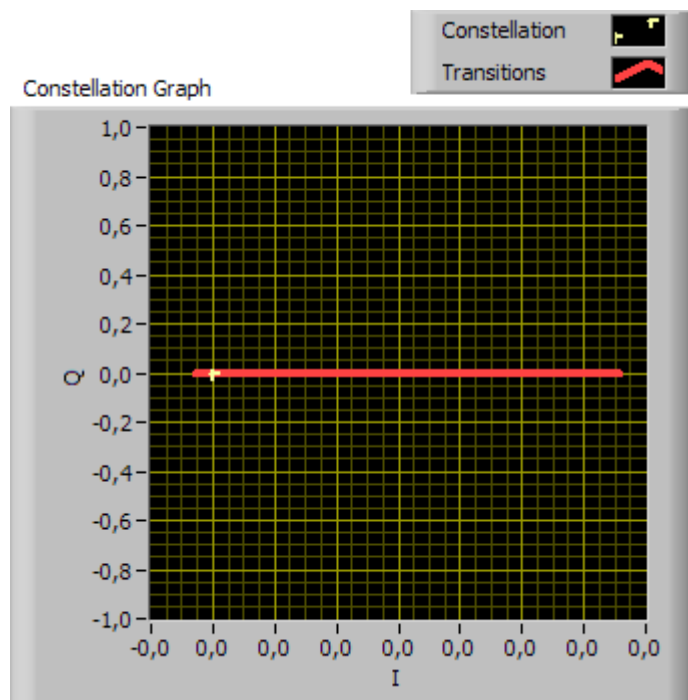


Fig. 2.11 Constelación canal Rayleigh

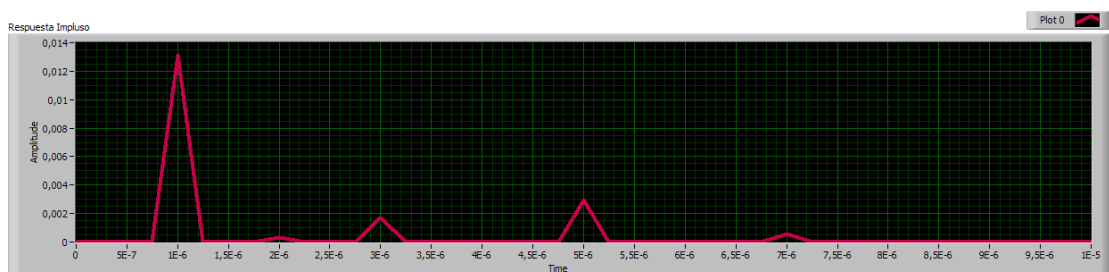


Fig. 2.12 Respuesta en el tiempo del canal Rayleigh

La Figura 2.12 muestra los diversos pasos ubicados según como se indica en la tabla 2.7, respecto a las ganancias y a la fase, la última hace referencia a la constelación, no son constantes ya que ambas provienen de una distribución Rayleigh la cual es aleatoria en el tiempo.

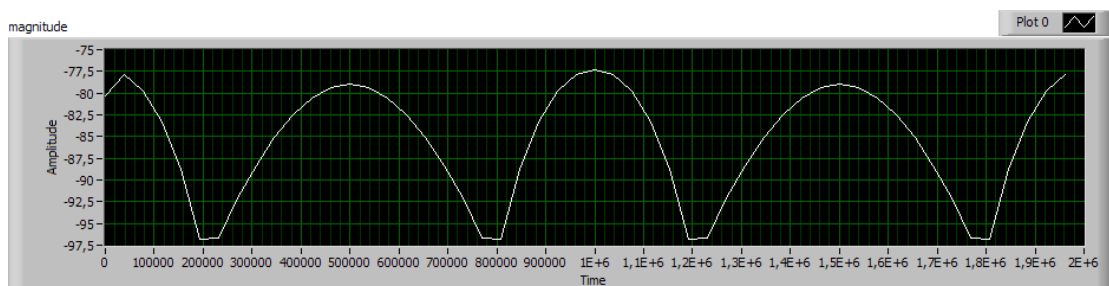


Fig. 2.13 Respuesta en la frecuencia del canal Rayleigh

2.1.1 SINCRONIZACIÓN DE TRAMA Y CORRECCIÓN DE FRECUENCIA

2.1.1.1 SINCRONIZACIÓN

En los sistemas de comunicación, una señal transmitida que pasa a través de un canal de comunicación siempre se ve influenciada por retardos, ruidos e interferencias que provocan alteraciones que se ven reflejadas en la frecuencia y fase de la señal.

EL funcionamiento del receptor es recuperar la señal transmitida, realizando un número de labores auxiliares que le permiten indagar dentro de la señal con el fin de obtener la información necesaria para realizar una interpretación correcta de la misma.

Una de esas labores se conoce como sincronización, el objetivo es generar en el receptor una señal con la misma frecuencia y fase que la señal transmitida por el emisor. Por lo cual se estima, mediante algoritmos matemáticos, la frecuencia y fase que están ocultos en la señal recibida.

2.1.1.2 SINCRONIZACIÓN DE TRAMA

Como sabemos, toda señal transmitida posee una estructura pre establecida, ya sea por diseño de proveedor o basándose en algún estándar de transmisión, que debe ser conocida por el receptor.

El objetivo de la sincronización de trama es resolver los múltiples retardos en los periodos de símbolos,

determinado la estructura de la trama, ubicado al inicio de la misma.

Para realizar la búsqueda del inicio de la trama se utiliza una palabra codificada o como la llamaremos una secuencia de entrenamiento. Como se vio en la revisión del sistema digital la inserción de dicha secuencia se la realiza el en transmisor delante de la trama de datos.

Una vez transmitida la señal, el receptor recibe la señal y después del proceso de RF, el filtro de acoplamiento y el downsampling, se ejecutan el proceso de sincronización de trama.

La señal antes del sincronizador es:

$$y = h s[n - d] + v[n] \quad (3.24)$$

Donde h es un coeficiente complejo del canal y d es el desplazamiento de trama que deseamos corregir.

SECUENCIA DE ENTRENAMIENTO: Recordado que la secuencia de entrenamiento son aquellos símbolos que representa a un grupo de bits (palabras) codificados, con excelentes propiedades de auto correlación. Se trabajara con dos secuencias, la secuencia Barker y la secuencia Willard, ambas de longitud 11.

Longitud del código	Secuencia Barker
11	[---+++--+-+]
Longitud del código	Secuencia Willard
11	[++++--+-+---]

Tabla 3.8 Código Barker y Willard Longitud 11 [1] y [18]

Se utilizara 4 secuencias de Barker y 4 secuencias de Willard, ambas de longitud 11. Las cuales se modularan con igual método que la modulación de los bits de datos.

Cabe considerar que el símbolo modulado de la secuencia de entrenamiento debe conservar las mismas propiedades de la secuencia.

AUTO-CORRELACIÓN: La auto correlación se define como el valor esperado del producto $X_1 = X[t_1]$ y $X_2 = X[t_2]$, que son muestras tomadas en tiempos diferentes de la señal $X[t]$ [7].

La auto-correlación me muestra el proceso de interdependencia y relación (correlación) de dos funciones de $X[t]$ desplazadas $\tau = t_2 - t_1$.

Esto es fundamental para la sincronización de trama ya que por medio de este cálculo nos permite determinar el inicio de la trama, aprovechando las propiedades de auto correlación de la secuencia de entrenamiento.

Recordemos que las propiedades de la auto-correlación son:

1. La auto-correlación es máxima en $\tau = 0$ es decir

$$R_{xx}(\tau) \leq R_{xx}(0).$$

2. La auto-correlación de un proceso WSS (wide sense stationary, estrictamente estacionario) es una función par, es decir $R_{xx}(\tau) \leq R_{xx}(-\tau)$.

3. La auto-correlación de un proceso WSS es máximo en el Origen: $R_{xx}(\tau) \leq R_{xx}(0)$.

4. Si $X(t)$ es ergódica y no periódica, entonces:

$$\lim_{\tau \rightarrow \infty} R_{xx}(\tau) = \mu_X^2.$$

5. Si $X[t]$ es periódica con período T , $R_{xx}(\tau)$ será periódica con el mismo período T .

Para nuestro estudio las propiedades 1 y 3 son las importantes, y que nos permitirán encontrar el punto deseado.

2.1.1.3 SINCRONIZACIÓN DE TRAMA POR CORRELATOR

Este método se basa en correlacionar la señal, o secuencia, transmitida que ingresa al receptor con la secuencia de entrenamiento que es previamente conocido por el receptor [1].

Suponiendo que $t[n]$ es la secuencia de entrenamiento modulada con longitud N_t

La operación se define de la siguiente manera:

$$R[n] = \sum_{k=0}^{N_t-1} t^*[k] y[n+k] \quad (3.25)$$

Y calculando luego:

$$d = \max_n R[n] \quad (3.26)$$

En donde $y[n]$ es la señal recibida, $R[n]$ la correlación de las dos señales y d es desplazamiento de trama.

Como se planteó anteriormente el objetivo es encontrar la estructura de la trama enviada.

Son estas dos ecuaciones las que se implementan en el laboratorio presentado en el manual.

2.1.1.4 SINCRONIZACIÓN DE TRAMA SCHMILD & COX

Este Timothy M. Schmild and Donald C. Cox presentaron un método robusto de sincronización ([9] y [1]) aplicado a sistema OFDM, en donde aprovechaban una secuencia de entrenamiento muy larga para realizar la búsqueda del inicio de la trama.

El transmisor OFDM inserta dos secuencias de entrenamiento, de igual longitud, que son insertadas al inicio de la trama en el transmisor. Otorgándonos una gran similitud con la trama del sistema de portadora única facilitado la adaptación del método.

El método Schimidt and Cox (SAC) es descrito por:

$$R[d] = \sum_{n=0}^{N-1} y^* [n + d] y[n + d + L] \quad (3.27)$$

La ecuación representa el producto de dos muestras diferentes de la señal de entrada. Donde d es el índice en el tiempo donde empieza la secuencia de entrenamiento. L es la longitud de la primera mitad de la secuencia de entrenamiento y N es la longitud de la subportadoras en OFDM.

Luego determinamos la Energía de la segunda muestra:

$$P[d] = \sum_{n=0}^{N-1} y[n + d + L]^2 \quad (3.28)$$

Y calculamos la métrica de tiempo:

$$M[d] = \frac{R[d]^2}{P[d]} \quad (3.29)$$

La cual nos permite estimar el desplazamiento de trama producido por el canal, por medio de:

$$d = \underset{d}{\operatorname{argmax}} M[d] \quad (3.30)$$

Donde d es el inicio de nuestra trama.

ADAPTACIÓN: Nuestro sistema trabaja con múltiples portadoras por lo que el valor de L deberá ser referenciado a otro parámetro de nuestra trama original.

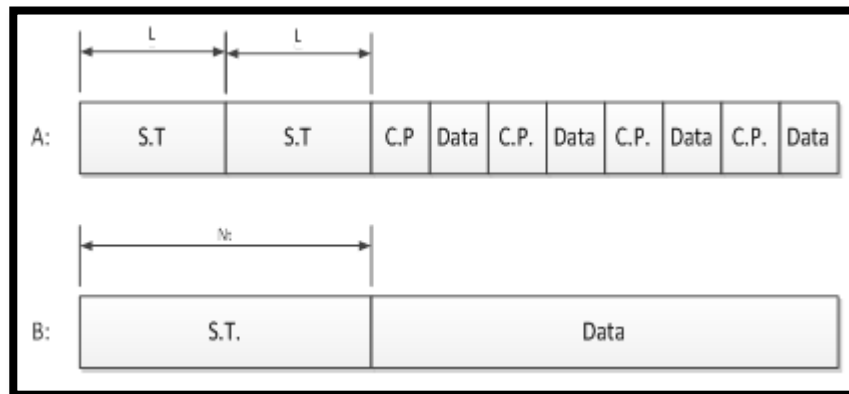


Fig. 3.14. a) Trama OFDM compuesta por dos secuencias de entrenamiento (S.T), el prefijo cíclico (C.P) y los datos b) Trama de un sistema de una sola portadora

Como se muestra en la figura 2.14, la estructura de la trama de cada sistema tiene una similitud, por lo que podemos plantear:

1. $N_t = 2L$, se iguala las dimensiones de la secuencias de entrenamiento.
2. N_t es la cantidad de la ventana de operación (número de muestras) el cual tendrá una dimensión de $N_t/2$, la mitad de la dimensión de la secuencia insertada.

Quedando la expresión:

$$R[d] = \sum_{i=0}^{\frac{N_t}{2}} y^* [i + d] y[i + d + \frac{N_t}{2}] \quad (3.31)$$

$$P[d] = \sum_{i=0}^{\frac{N_t}{2}} y[i + d + \frac{N_t}{2}]^2 \quad (3.32)$$

Donde se toma en consideración:

$$d[0, Z - 1] \quad (3.33)$$

$$Z = [\frac{P}{N_t} - 1] \frac{N_t}{2} \quad (3.34)$$

Siendo Z el valor que limita a la correlación, ya que se toman muestras de igual dimensión y se calcula su correlación, luego se obtiene la métrica con la ecuación 2.29 y el inicio de trama d con la ecuación 2.30.

2.1.1.5 CORRECCIÓN DE FRECUENCIA

Generalmente en sistemas de comunicaciones inalámbricas se envía información modulada con una frecuencia de portadora f_c . Debido a las variaciones de temperatura y al hecho que la frecuencia de portadora es generada por un oscilador local diferente en el transmisor y receptor, en la práctica f_c en el transmisor no es igual a f'_c en el receptor como se ilustra en la figura 2.15.

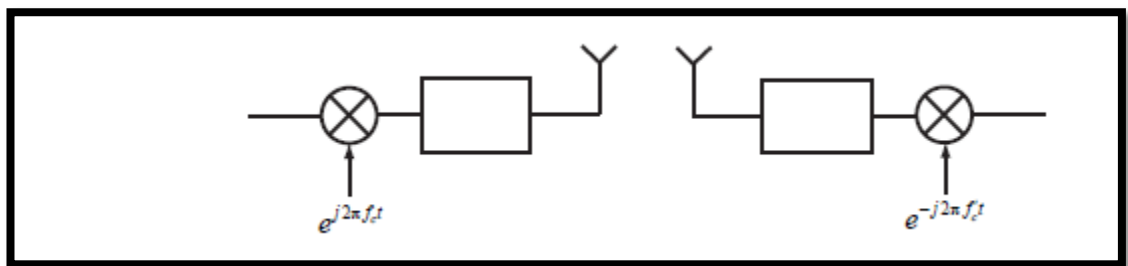


Fig. 3.15. Diagrama de bloques con f_c diferente en el trasmisor y receptor [8]

La diferencia entre el transmisor y receptor es $f_e = f'_c - f_c$ es la frecuencia de portadora de desplazamiento (o frecuencia offset) y es generalmente medida en Hertz. Algunas intuiciones sobre el efecto de desplazamiento de frecuencia pueden ser obtenidas considerando el caso especial de un canal de desvanecimiento plano. En un canal de frecuencia plana

$$y[n] = e^{j2\pi\epsilon n} h s[n] + v[n] \quad (3.31)$$

Note que el símbolo transmitido está siendo rotado por $\exp(j2\pi\epsilon n)$. Mientras n aumenta, el desplazamiento aumenta, y la constelación de símbolo rota aún más. El impacto de esto es un aumento en el número de errores de símbolos como los símbolos rotan fuera de sus respectivas regiones como se ilustra en la figura 2.16.

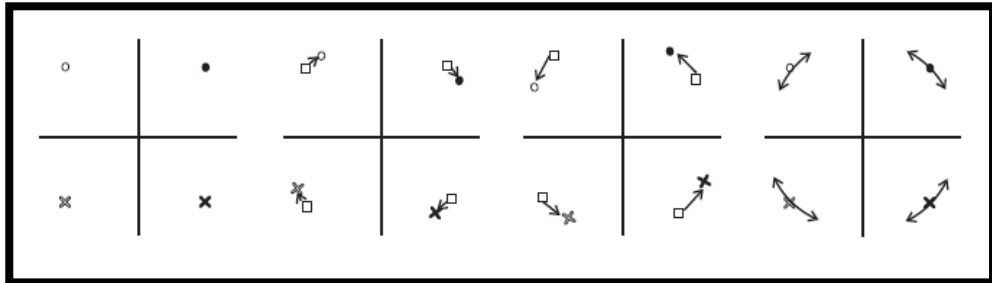


Fig. 3.16. Efectos de la frecuencia de desplazamiento

El proceso de corrección de ϵ se conoce como sincronización de frecuencia de desplazamiento. El típico método para sincronización de frecuencia de desplazamiento consiste en estimar primero el

desplazamiento ϵ luego de corregir esto por la formación de la nueva secuencia $\exp -j2\pi\epsilon n$ y[n] con la fase eliminada. Hay varios métodos para la corrección, muchos emplean un estimador de desplazamiento de frecuencia seguido por una corrección de fase. Estimadores utilizan las propiedades más específicas de la secuencia de entrenamiento.

Para el sincronismo en frecuencia el método de Schmid y Cox propone el siguiente algoritmo:

1. Se parte del primer paso del algoritmo para la sincronización trama, donde se calculó la correlación entre las dos mitades del símbolo $R[d]$ de forma que se han cancelado los efectos del canal.
2. Se calcula el desplazamiento en frecuencia de la siguiente forma:

$$\hat{\epsilon} = \frac{1}{N} \sum_{d=0}^{N-1} R[d] \quad (3.32)$$

$$\Delta f = \frac{\phi}{\pi T} \quad (3.33)$$

ϕ : es la fase que forma la correlación $P[d]$.

Δf : es la desviación en frecuencia que ha sufrido la señal y que habrá que corregir.

ε : es Δf pero normalizada.

De esta forma cuando la métrica temporal supere el umbral de detección se habrá estimado que la trama acaba de empezar y, cuando supere el de estimación, se tomará en ese instante el valor correspondiente de ε y éste será el offset de frecuencia a corregir en el resto de la trama.

2.1.2 DIAGRAMA DE OJO

El diagrama de ojo es una herramienta de visualización que nos permite analizar y estudiar las señales en un sistema de comunicación digital. Esta nos proporciona información acerca del funcionamiento de nuestro sistema e imperfecciones producidos por el canal a nuestra señal transmitida.

Imaginase un osciloscopio que puede tomar muestras de la señal recibida una tras de otra sin perder la imagen de la muestra anterior, todas con el mismo intervalo de tiempo. Es decir una superposición de imágenes de todas las posibles relaciones de la señal muestreada. Se denomina patrón de ojo debido a su similitud con el ojo humano.

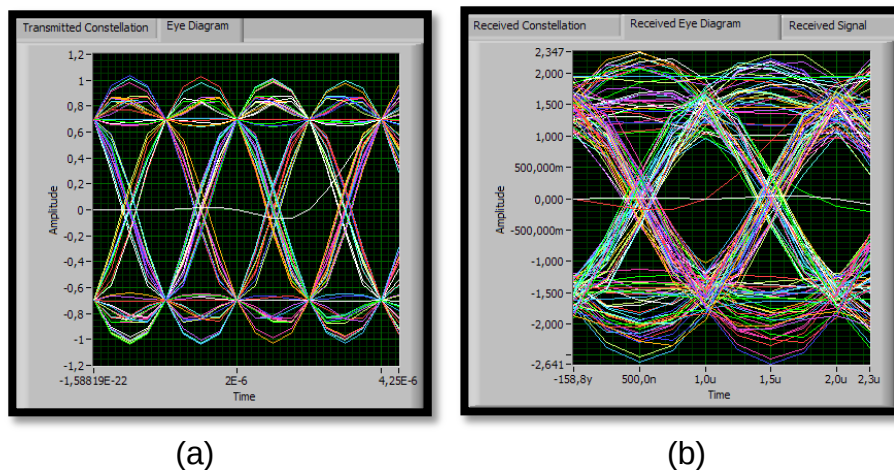


Fig. 3.17. Diagramas de ojo (a). Transmisión y recepción de una señal QPSK. (b) Con efectos de canal.

En la Figura 2.17 se puede apreciar la transmisión y recepción de una señal QPSK. La primera, correspondiente al receptor, es el caso ideal, donde el canal no interfiere a la señal. La segunda, correspondiente al receptor, se puede visualizar claramente los efectos de canal, tanto en amplitud como en el tiempo de muestra.

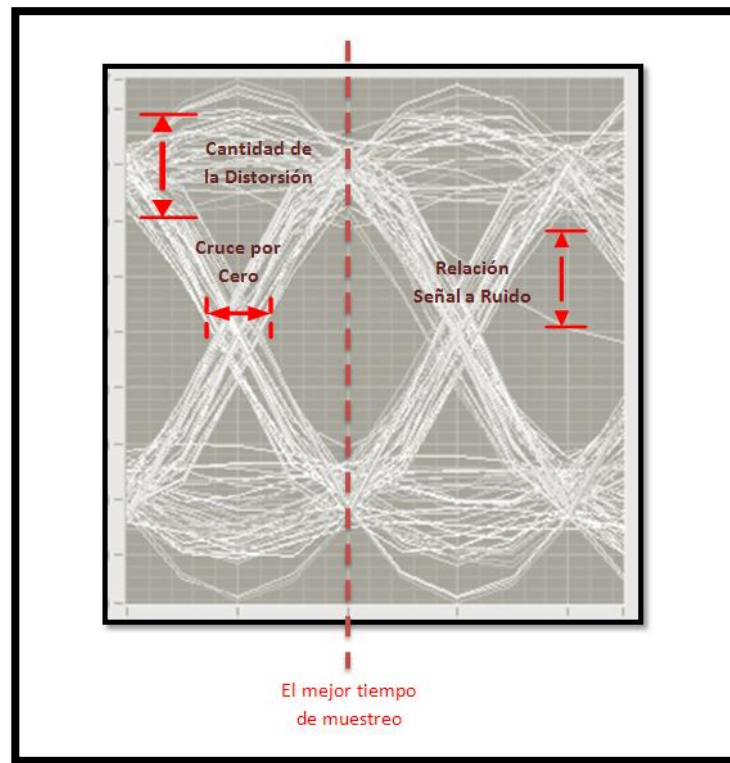
ANÁLISIS:

Fig. 3.18. Información que se obtiene del diagrama de ojo.

Los impedimentos pueden ocurrir en los filtros de transmisor, en los amplificadores, en los circuitos de RF, en la propagación, en los circuitos IF (Intermediate Frequency), y en el procesamiento de señal. El diagrama nos permite visualizar una posible solución al problema en el lugar donde se origine.

En una transmisión digital lo ideal es que el diagrama de ojo genere una gráfica similar a pulsos cuadrados (como se puede apreciar en el gráfico), por esa razón se tiene presente los siguientes valores los siguientes factores:

Mejor tiempo de muestreo: es el punto en el diagrama en donde la apertura del ojo es mayor, que es la mitad del intervalo de muestreo sin presencia de ISI.

Relación señal a ruido: me indica la apertura del ojo respecto al cruce por cero.

Cruce por cero: es el punto donde se genera la transición. Aquí se determina la sensibilidad del sistema según la velocidad en que se produzca dicha transición (pendiente).

Cantidad de distorsión: es la cantidad de ruido e ISI que se introduce en el sistema.

Una gráfica más suavizada, es decir como onda sinusoidal, visualizándose como un ojo cerrado, es decir que los trazos

superiores se cruzan con los inferiores, por lo consiguiente mayor error en la decodificación.

CAPÍTULO 3

IMPLEMENTACIÓN DEL SISTEMA DE COMUNICACIÓN USANDO LABVIEW.

En el presente capítulo, se describen los diferentes algoritmos realizados en Labview, los datos se transmiten a través del hardware USRP para mediante una interfaz gráfica en el software Labview poder realizar el análisis de la señal recibida y de los parámetros que se verán afectados debido al medio de propagación.

En la Figura 3.1 y 3.2 se puede observar la arquitectura propuesta para el sistema de comunicación, en este proyecto se omitieron los bloques de filtros y ecualizadores ya que corresponden a otro trabajo desarrollado de manera conjunta al presente proyecto.

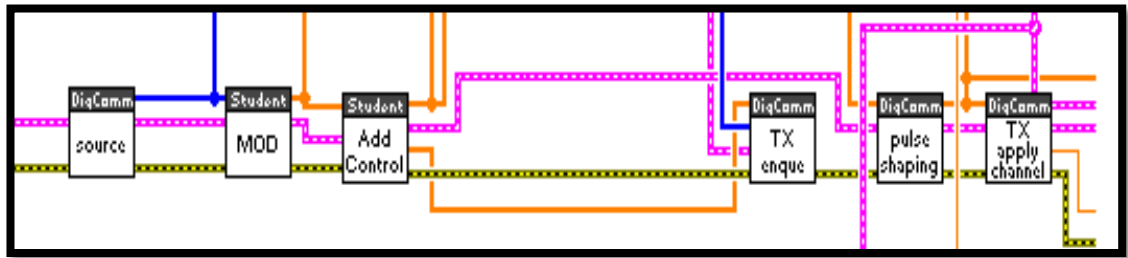


Fig. 4.1. Diagrama de Bloques del Transmisor

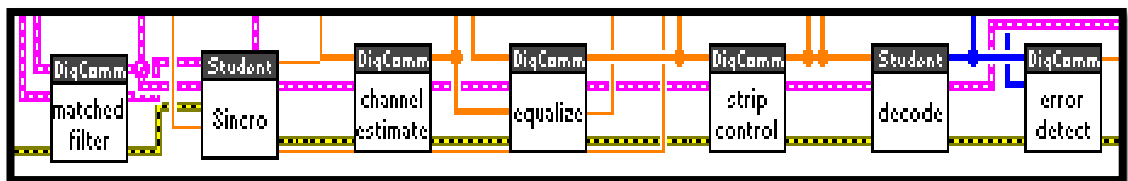


Fig. 4.2. Diagrama de Bloques del Receptor

3.1 ALGORITMOS

En esta sección se describirán los algoritmos con sus respectivos diagramas de bloques realizados en Labview para el sistema de comunicación de única portadora.

3.1.1 MODULADOR

En el bloque que se muestra en la Figura 3.3 se observa que del cluster ***modulation parameters in*** se recibe el tipo de modulación a escoger en nuestro caso podemos seleccionar 5 tipos de modulación: BPSK, QPSK, 8 QAM, 16 QAM y 64 QAM; dependiendo del tipo de modulación la secuencia de bits de

entrada **input bit stream** se modula en sus respectivos símbolos los cuales se muestran en el arreglo **output symbols**.

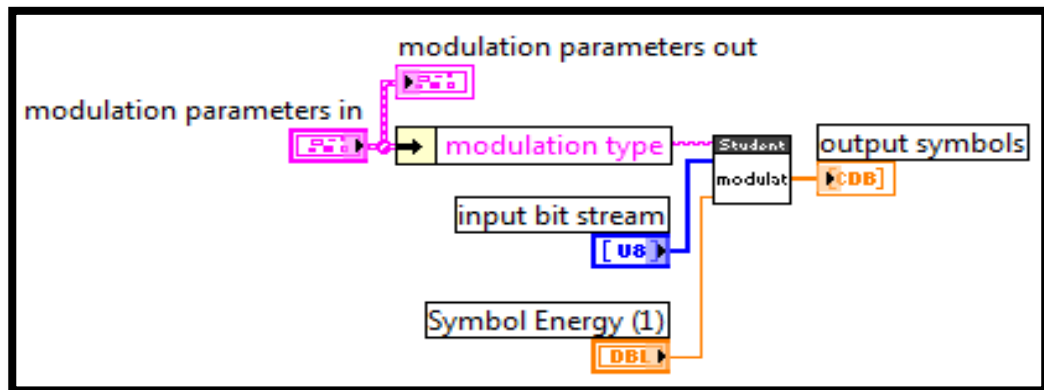


Fig. 4.3. Diagrama de Bloques del Modulador

Panel Frontal del Instrumento Virtual: Para visualizar los símbolos codificados se ha utilizado un arreglo de datos numéricos tipo entero flotante, como se muestra en la Figura 3.4.

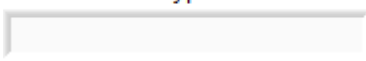
Pestaña de Controles >> String & Path >> String Control	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
Modulation type 	Modulation type 
Control que permite ingresar por teclado el tipo de modulación que se desee seleccionar.	

Fig. 4.4. Control String Modulation Type

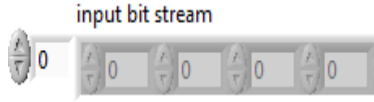
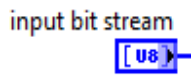
Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Bits que se envían al para que sean codificados en su respectivo símbolo, según el tipo de modulación.	

Fig. 4.5. Control Tipo Array

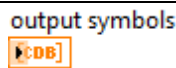
Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Indicador que permite mostrar los respectivos símbolos complejos codificados.	

Fig. 4.6. Indicador Tipo Array Complex

Diagrama de bloques del Instrumento Virtual: A continuación describiremos de manera resumida las funciones utilizadas en las diferentes modulaciones.

Programming>> Structures >> While Loop
DIAGRAMA DE BLOQUES

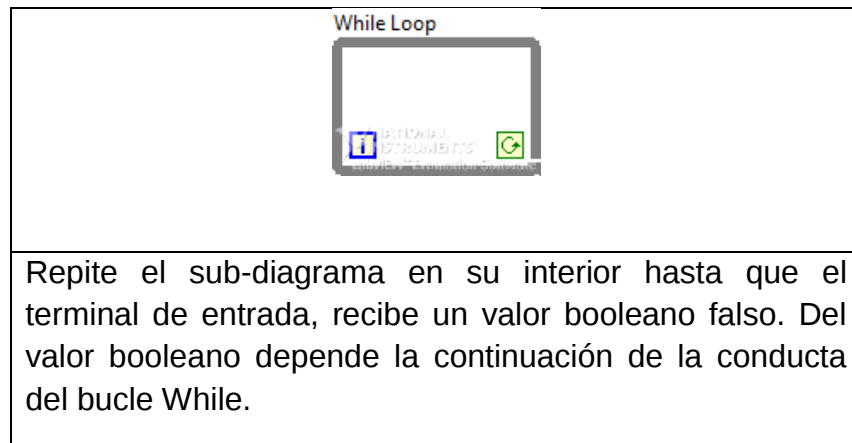


Fig. 4.7. Ciclo While Loop

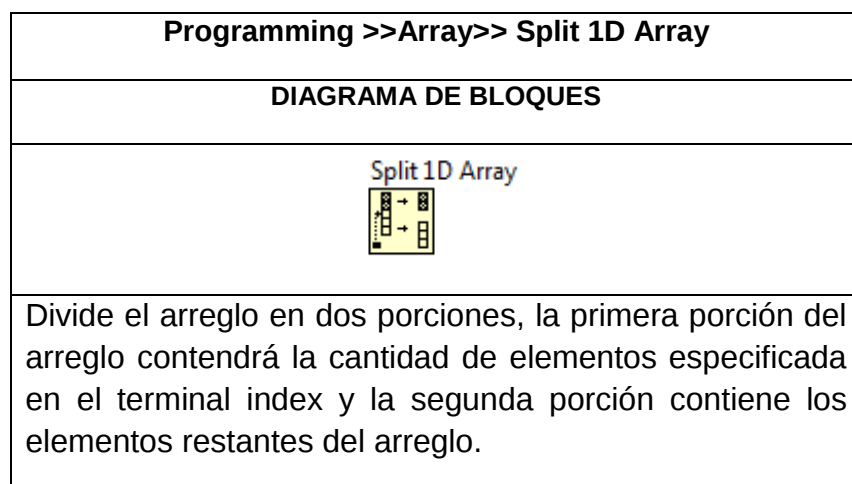
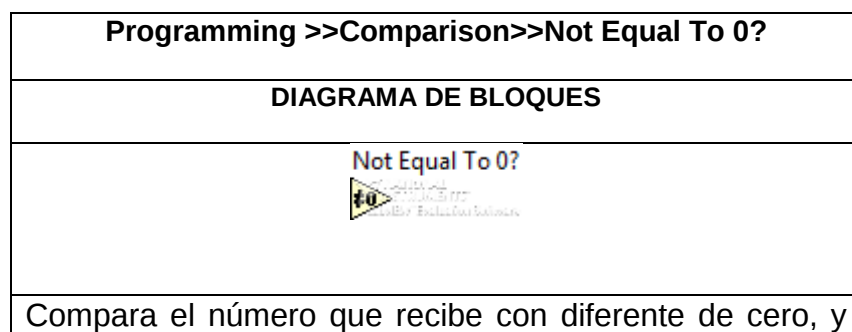


Fig. 4.8. Spilt 1D Array



retorna 0 o 1.

Fig. 4.9. Not Equal To 0?

Programming >>Array>>Array Size
DIAGRAMA DE BLOQUES

Retorna el tamaño de un arreglo.

Fig. 4.10. Array Size

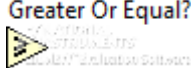
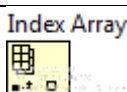
Programming >>Comparison >>Greater Or Equal?
DIAGRAMA DE BLOQUES

Compara si el número que ingresa en el terminal x de la función es mayor o igual que el que se ingresa en el terminal y, dependiendo de eso retorna un valor booleano 0 o 1.

Fig. 4.11. Greater Or Equal?

Programming >>Array>>Index Array.
DIAGRAMA DE BLOQUES


Retorna un elemento del arreglo, especificando el índice del arreglo del cual se desea extraer el elemento.

Fig. 4.12. Index Array

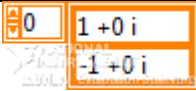
Programming >>Array >>Array Constant.
DIAGRAMA DE BLOQUES

Crea un arreglo de constantes numéricos complejos.

Fig. 4.13. Array Constant

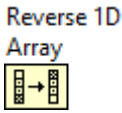
Programming >>Array >> Reverse 1D Array
DIAGRAMA DE BLOQUES

Revierte el orden de los elementos en el arreglo.

Fig. 4.14. Reverse 1D Array

EXPRESIONES MATEMATICAS
Programming >> Numeric >> Square Root
Programming >> Numeric >> Divide
DIAGRAMA DE BLOQUES

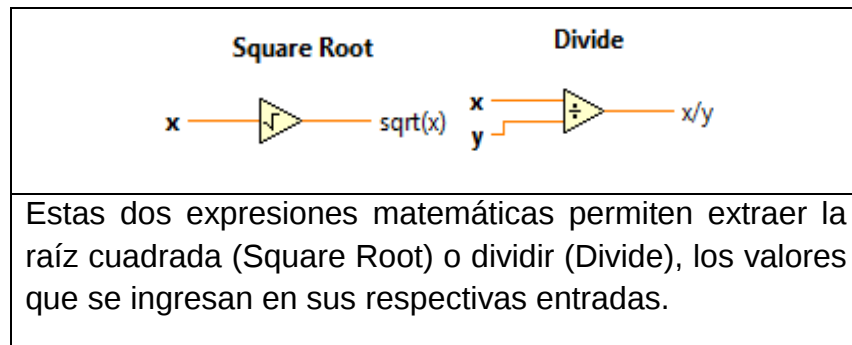


Fig. 4.15. Expresiones Square Root y Divide

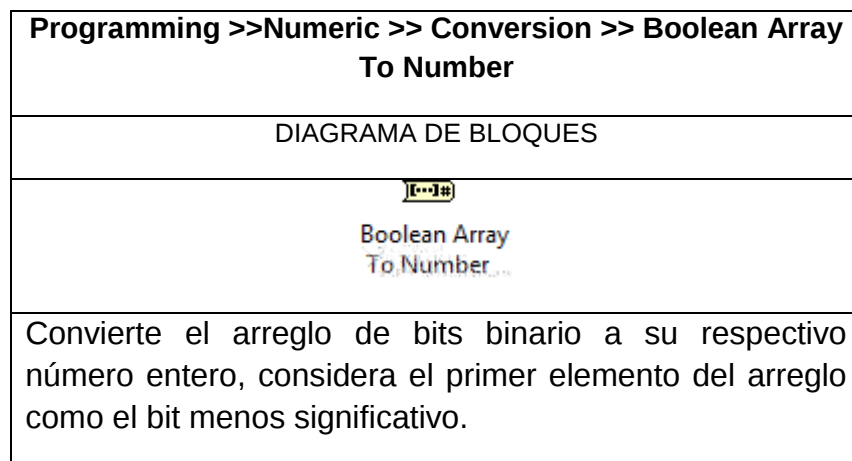


Fig. 4.16. Boolean Array To Number

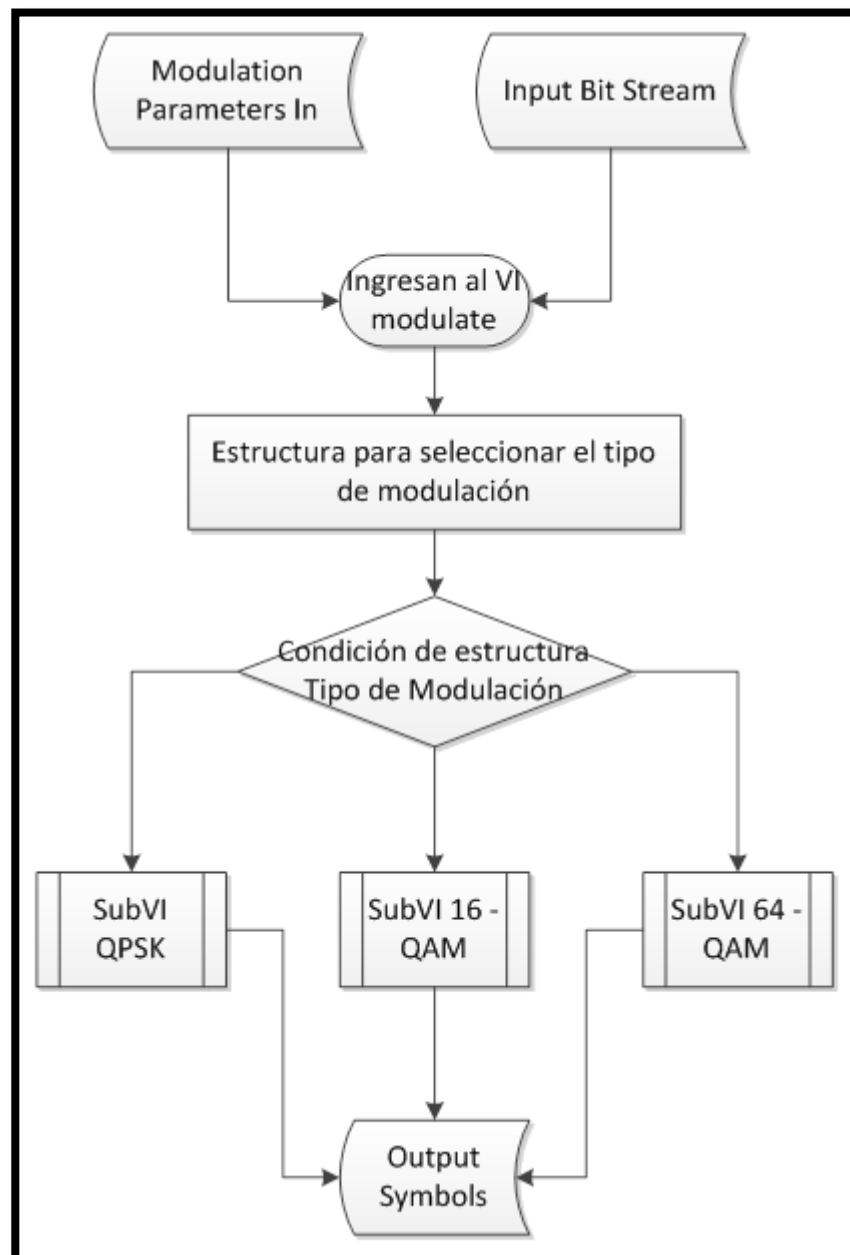


Fig. 4.17. Diagrama de flujo VI Modulate

Sub VI Modulador QPSK:

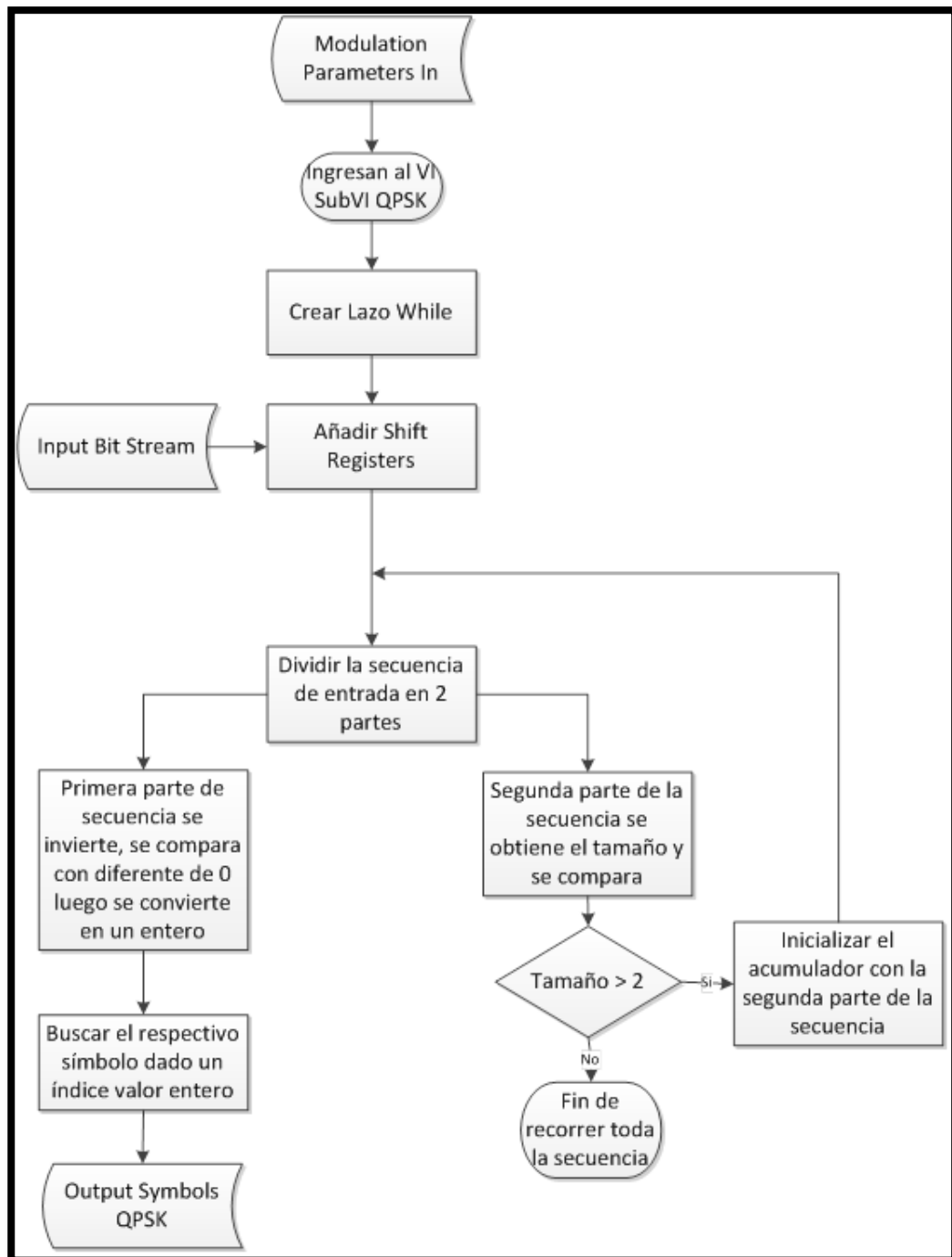


Fig. 4.18. Diagrama de flujo SUBVI QPSK

Para el diseño del esquema de modulación QPSK se tiene que seguir el siguiente procedimiento y en el orden establecido que se describe a continuación:

1. Creamos un VI, siguiendo los pasos de inicialización de un VI.
2. Se procede a insertar un lazo While que se encuentra en la **paleta de funciones/Structures/While Loop**, este lazo se ejecuta mientras el tamaño del arreglo sea mayor o igual a 2, dado que es una modulación QPSK. Vamos a detener la ejecución del lazo While cuando la condición sea falsa, por lo tanto para cambiar este valor damos clic derecho en **Loop Condition ==>> Continue if True**, como se muestra en la Figura 3.19.

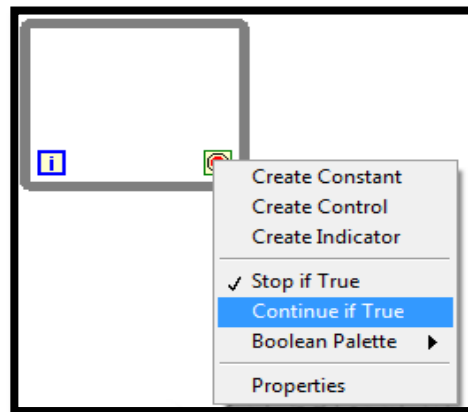


Fig. 4.19. Condition While Loop

3. Luego creamos Shift Registers en los bordes del lazo While, estos se utilizan como acumuladores para que en cada iteración del lazo se almacene el arreglo restante de la secuencia de entrada. Para añadir estos acumuladores clic derecho en el borde **Add Shift Register**.

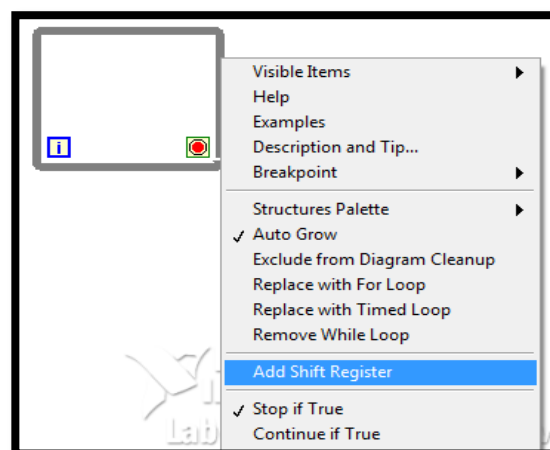


Fig. 4.20. Add Shift Registers

4. El arreglo de la secuencia de entrada, lo dividimos en dos porciones: la primera porción contiene los dos primeros elementos del arreglo de entrada y la segunda porción contiene los elementos restantes del arreglo **input bit string**. Para dividir el arreglo de entrada utilizamos la función **Split 1D Array**, ubicado en: **paleta de funciones/ Array**, en el terminal *index* de la función clic derecho **Create ==> Constant de valor 2**.

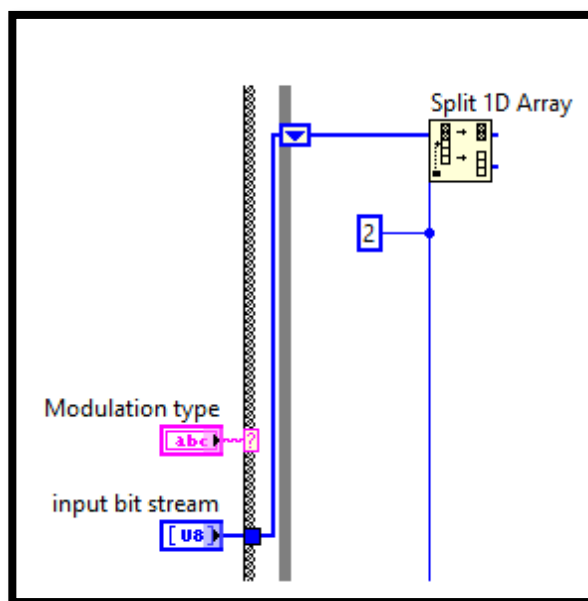


Fig. 4.21. Dividir la secuencia de entrada QPSK

5. La primera porción de la secuencia de entrada, que son dos bits para el caso de modulación QPSK, se invierten mediante la

función **Reverse 1D Array** que se localiza en la **paleta de funciones/ Array**

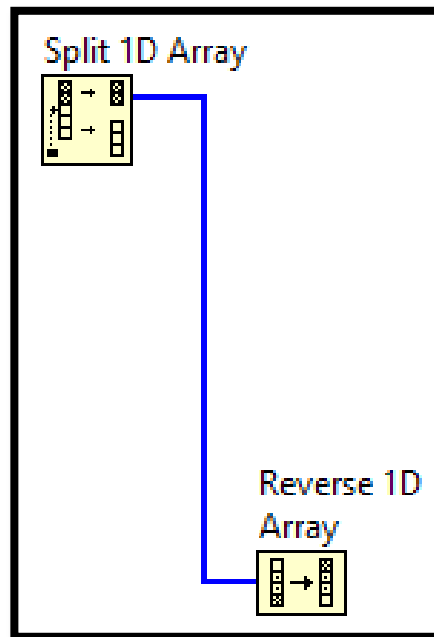


Fig. 4.22. Invertir los bits a ser modulados

6. Cada di-bit previamente invertido se compara con diferente de cero para de esta manera tener un arreglo de dos bits booleanos 00, 01,10 o 11 como se muestra en la Figura 3.23. La función de comparación se encuentra en la **paleta de funciones/ Comparison/ Not Equal To 0?**



Fig. 4.23. Conversión de arreglo entero a booleano

7. La función **Boolean Array To Number** que se encuentra en la **paleta de funciones/ Numeric/ Conversion**, esta función convierte un arreglo booleano a entero como se muestra en la Figura 3.24, LabVIEW interpreta el arreglo como complemento a 2, por tal motivo previamente se hizo la inversión del arreglo. Como resultado de esta conversión si el arreglo de entrada es 11 a la salida tendremos 3.

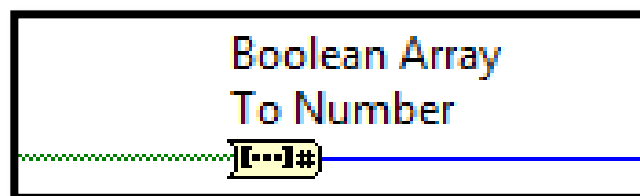


Fig. 4.24. Conversión de booleano a entero

8. A su vez con cada dato entero se manda a buscar un símbolo en el arreglo de constantes, se extrae elementos dependiendo del índice mediante la función **Index Array**, que se encuentra en la **paleta de funciones/ Array**, para crear el arreglo de

constante **paleta de funciones/ Array/ Array Constant** una vez creado el arreglo hay que insertar constantes numéricas y cambiar la representación a CDB para lo cual ir a la **paleta de funciones/ Numeric/ DBL Numeric Constant** insertar la constante en el arreglo creado previamente, posteriormente dar clic derecho y seleccionar **Representation ==> CDB**; como es una modulación QPSK los díbits son modulados mediante cuatro símbolos, por lo tanto se expande el arreglo para que muestre cuatro posiciones, donde la primera posición del arreglo contendrá la constante $0.7+0.7i$, segunda posición $0.7-0.7i$, tercera posición $-0.7+0.7i$ y cuarta posición $-0.7-0.7i$, dichas constantes representan a los bits 00, 01, 10 y 11 respectivamente.

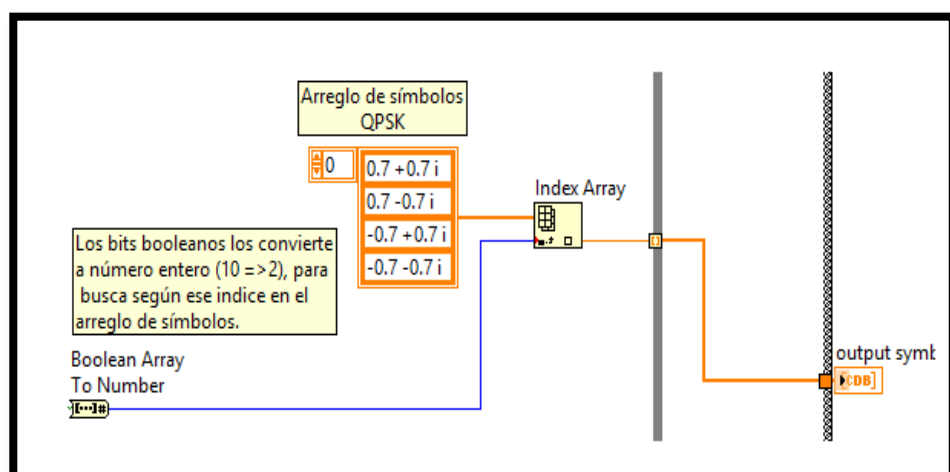


Fig. 4.25. Bits mapeados a sus respectivos símbolos QPSK

9. Finalmente para determinar la condición de stop del lazo While se compara el tamaño de la segunda porción de la secuencia de entrada con mayor o igual a 2 ya que se necesitan 2 bits para modular en QPSK, para obtener el tamaño del arreglo se utiliza la función **Array Size** que se encuentra en la **paleta de funciones/ Array** y para hacer la comparación se utiliza la función **Greater Or Equal?** que se localiza en la **paleta de funciones/ Comparison** como se muestra en la Figura 3.26.

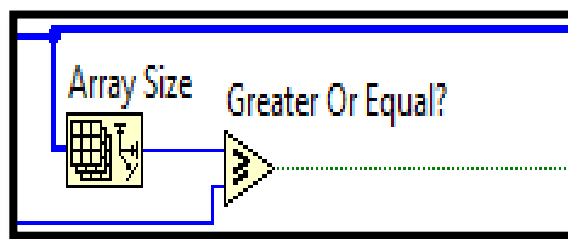


Fig. 4.26. Condición de Stop del Lazo While

Diagrama de bloques de modulación QPSK.

La figura 3.27 muestra la programación completa del diagrama de bloques de la modulación QPSK, con sus respectivos controles e indicadores que permiten manipular los parámetros de entrada y visualizar el comportamiento del arreglo de salida frente a los valores que se hayan ingresado.

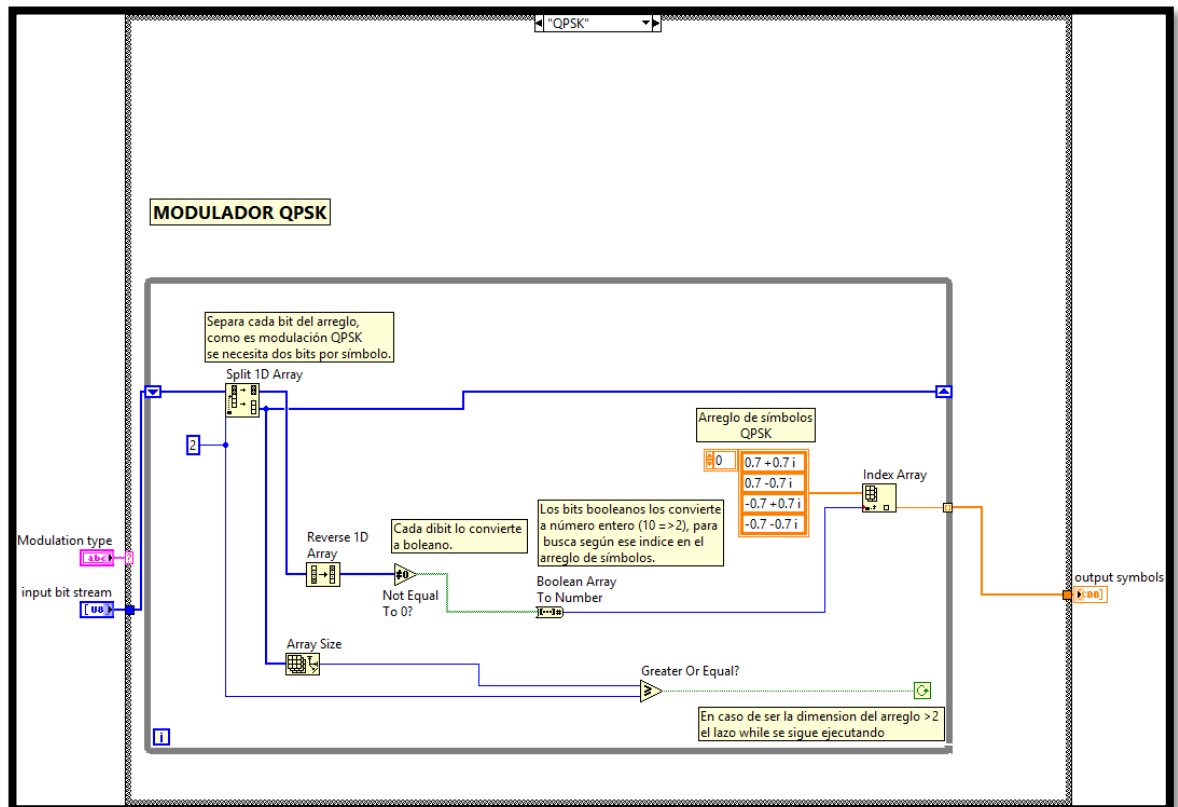


Fig. 4.27. Diagrama de bloques del Sub VI - QPSK

Sub VI Modulador 16 - QAM:



Fig. 4.28. Diagrama de flujo del Sub VI 16-QAM

1. Repetir paso 1, 2 y 3 que se describen en el Sub – VI modulación QPSK.
2. El arreglo de la secuencia de entrada, lo dividimos en dos porciones: la primera porción contiene los cuatro primeros elementos del arreglo de entrada y la segunda porción contiene los elementos restantes del arreglo **input bit string**. Para dividir el arreglo de entrada utilizamos la función **Split 1D Array**, ubicado en: **paleta de funciones/ Array**, en el terminal *index* de la función clic derecho **Create ==> Constant** de valor 4.

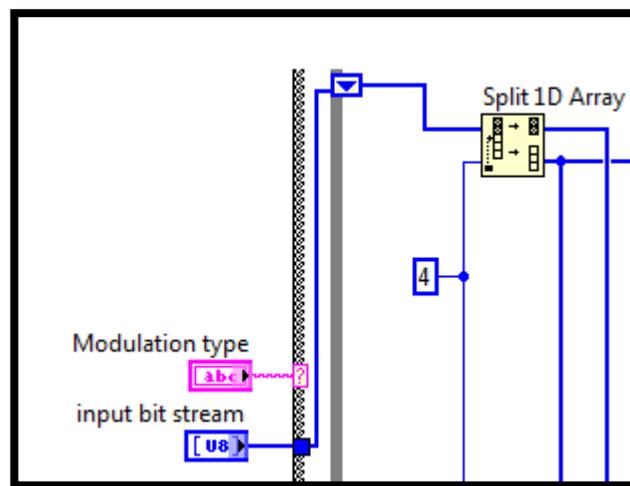


Fig. 4.29. Dividir la secuencia de entrada 16 - QAM

3. La primera porción de la secuencia de entrada, que son cuatro bits para el caso de modulación 16-QAM, se invierten mediante la función ***Reverse 1D Array*** que se localiza en la ***paleta de funciones/ Array*** como se muestra en la Figura 3.22.

4. El arreglo de cuatro bits previamente invertidos se compara con diferente de cero como se muestra en la Figura 3.23, para de esta manera tener un arreglo donde cada elemento este representado por 4 bits booleanos 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110 o 1111. La función de comparación se encuentra en la ***paleta de funciones/ Comparison/ Not Equal To 0?***

5. Como resultado de esta conversión si el arreglo de entrada es 1111 a la salida tendremos 16, para realizar esta conversión repetir el paso 7 del Sub – VI modulación QPSK.

6. A su vez con cada dato entero se manda a buscar un símbolo en el arreglo de constantes, se extrae elementos dependiendo del índice mediante la función ***Index Array***, que se encuentra en la ***paleta de funciones/ Array***, para crear el arreglo de

constante ***paleta de funciones/ Array/ Array Constant*** una vez creado el arreglo hay que insertar constantes numéricas y cambiar la representación a CDB para lo cual ir a la ***paleta de funciones/ Numeric/ DBL Numeric Constant*** insertar la constante en el arreglo creado previamente, posteriormente dar clic derecho y seleccionar ***Representation ==> CDB***; como es una modulación 16-QAM los cuatro bits son modulados mediante ocho símbolos, por lo tanto se expande el arreglo para que muestre ocho posiciones, donde la primera posición del arreglo contendrá la constante $1+1i$, segunda posición $1+3i$, tercera posición $1-1i$, cuarta posición $1-3i$, quinta posición $3+1i$, sexta posición $3+3i$, séptima posición $3-1i$, octava posición $3-3i$, novena posición $-1+1i$, décima posición $-1+3i$, décima primera posición $-1-1i$, décima segunda posición $-1-3i$, décima tercera posición $-3+1i$, décima cuarta posición $-3+3i$, décima quinta posición $-3-1i$ y décima sexta posición $-3-3i$.

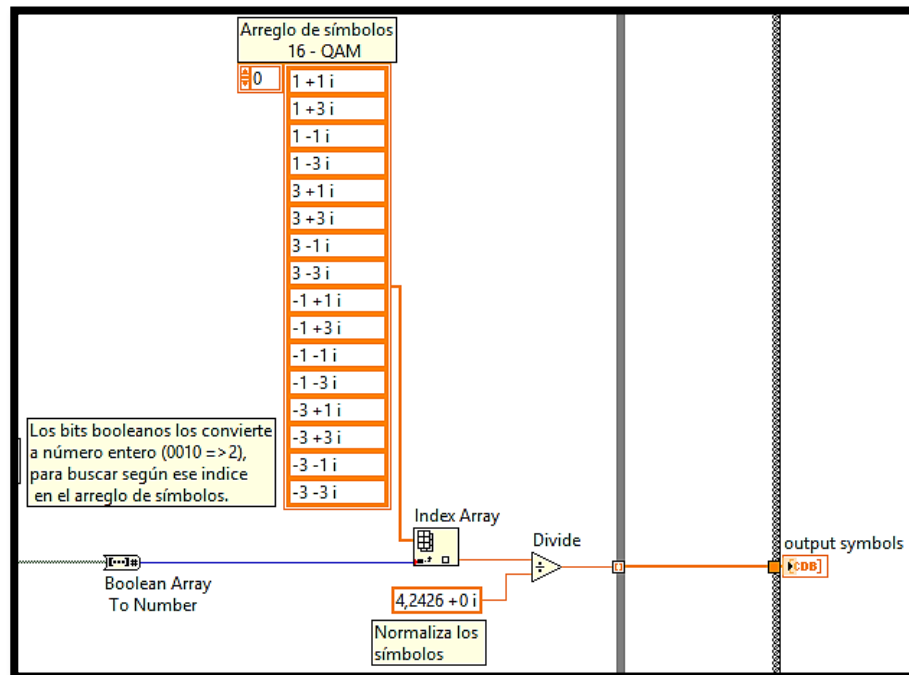


Fig. 4.30. Bits mapeados a sus respectivos símbolos 16 - QAM

7. Una vez que obtenemos los símbolos hay que normalizarlos, esta normalización se logra dividiendo para la mayor magnitud de la constelación 16-QAM que en este caso es $\sqrt{18} = 4,2426$.

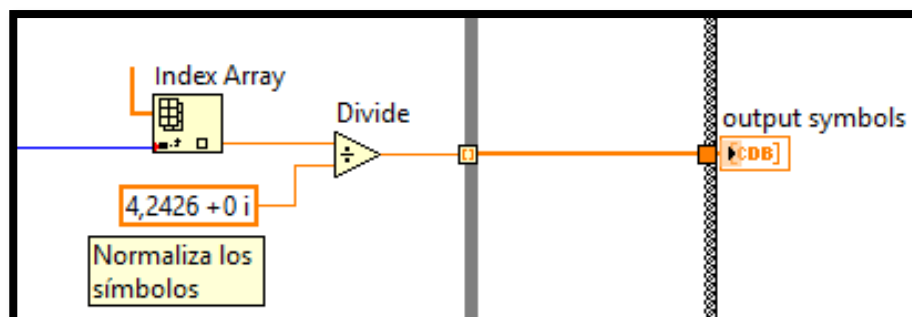


Fig. 4.31. Normalizar los símbolos 16 - QAM

8. Finalmente para determinar la condición de stop del lazo While se compara el tamaño de la segunda porción de la secuencia de entrada con mayor o igual a 4 ya que se necesitan 4 bits para modular en 16-QAM, como se puede observar en la Figura 3.26.

Diagrama de bloques de modulación 16-QAM.

La figura 3.32 muestra el diagrama de bloques correspondiente a la modulación 16-QAM.

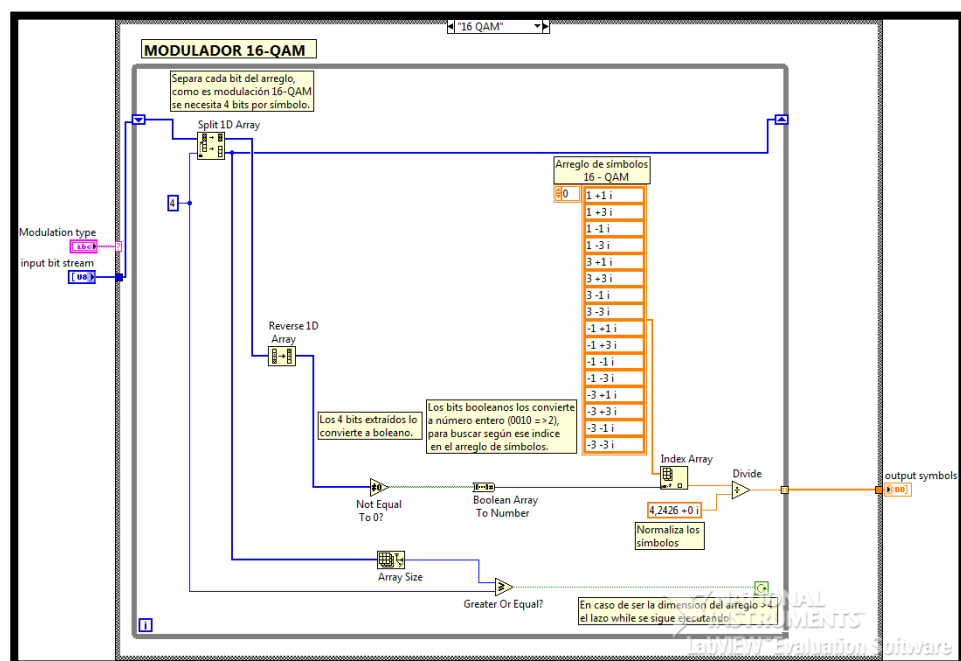


Fig. 4.32. Diagrama de bloques del Sub VI 16 - QAM

Sub VI Modulador 64 - QAM:

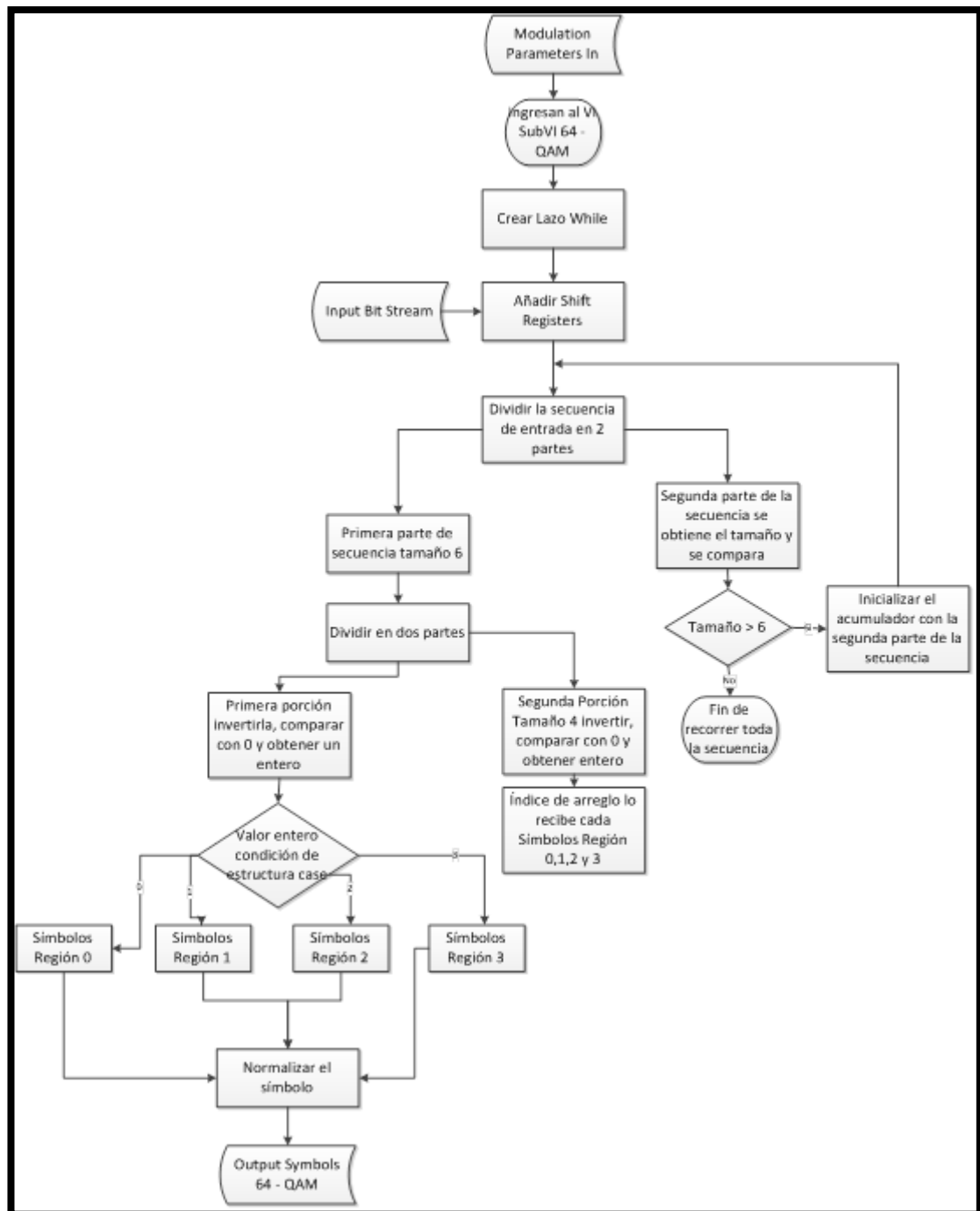


Fig. 4.33. Diagrama de flujo Sub VI 64 - QAM

1. Repetir paso 1, 2 y 3 que se describen en el Sub – VI modulación QPSK.
2. El arreglo de la secuencia de entrada, lo dividimos en dos porciones: la primera porción contiene los seis primeros elementos del arreglo de entrada y la segunda porción contiene los elementos restantes del arreglo **input bit string**. Para dividir el arreglo de entrada utilizamos la función **Split 1D Array**, ubicado en: **paleta de funciones/ Array**, en el terminal *index* de la función clic derecho **Create ==> Constant** de valor 6.

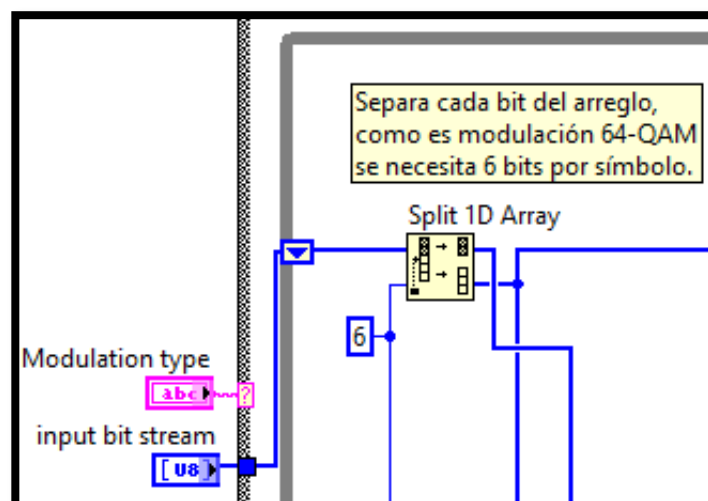


Fig. 4.34. Dividir la secuencia de entrada 64 - QAM

3. Los 6 bits extraídos previamente los dividimos en dos porciones: la primera porción define la zona en la que se encuentra el símbolo, dependiendo de esos bits ingreso a uno de los sub-diagramas de la estructura Case y la segunda porción me define la posición que mando a buscar en el arreglo de símbolos creado en cada sub-diagrama de la estructura.

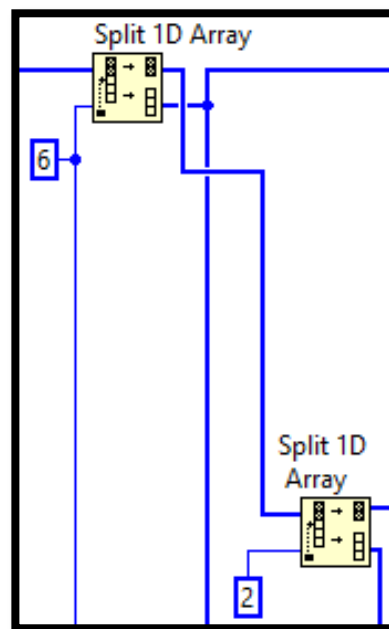


Fig. 4.35. Dividir los 6 bits de 64 - QAM

4. La primera y segunda porción de los 6 bits se invierten mediante la función **Reverse 1D Array** que se localiza en la **paleta de funciones/ Array**.

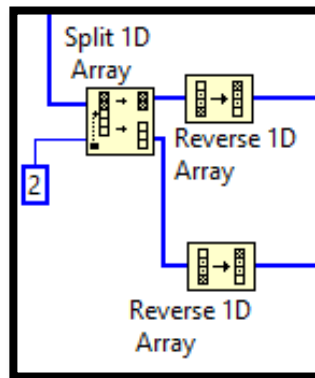


Fig. 4.36. Invertir los bits a ser modulados 64 - QAM

5. El arreglo de 2 y 4 bits previamente invertidos se compara con diferente de cero para de esta manera tener dos arreglos booleanos 00, 01, 10 o 11 y 000, 001, 010, 011, 100, 101, 110 o 111 respectivamente. La función de comparación se encuentra en la **paleta de funciones/ Comparison/ Not Equal To 0?** como se muestra en la figura 3.23.

6. La función **Boolean Array To Number** que se encuentra en la **paleta de funciones/ Numeric/ Conversion**, esta función convierte un arreglo booleano a entero, LabVIEW interpreta el arreglo como complemento a 2, por tal motivo previamente se hizo la inversión del arreglo.

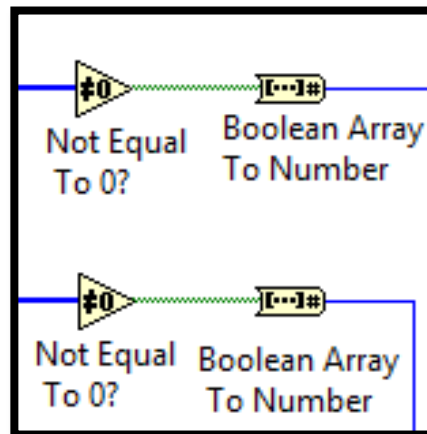


Fig. 4.37. Crear datos enteros dado un arreglo booleano

7. A su vez cada dato entero generado de los 2 bits, será la condición de selección de la estructura case, y los demás 4 bits también convertidos a enteros nos dan el índice mediante el cual se manda a buscar un símbolo en el arreglo de constantes.
8. Los dos bits que son la condición de la estructura case define una región y mediante eso seleccionamos un símbolo. Si es 00 nos encontramos en el sub-diagrama 0 con el siguiente arreglo de símbolos:

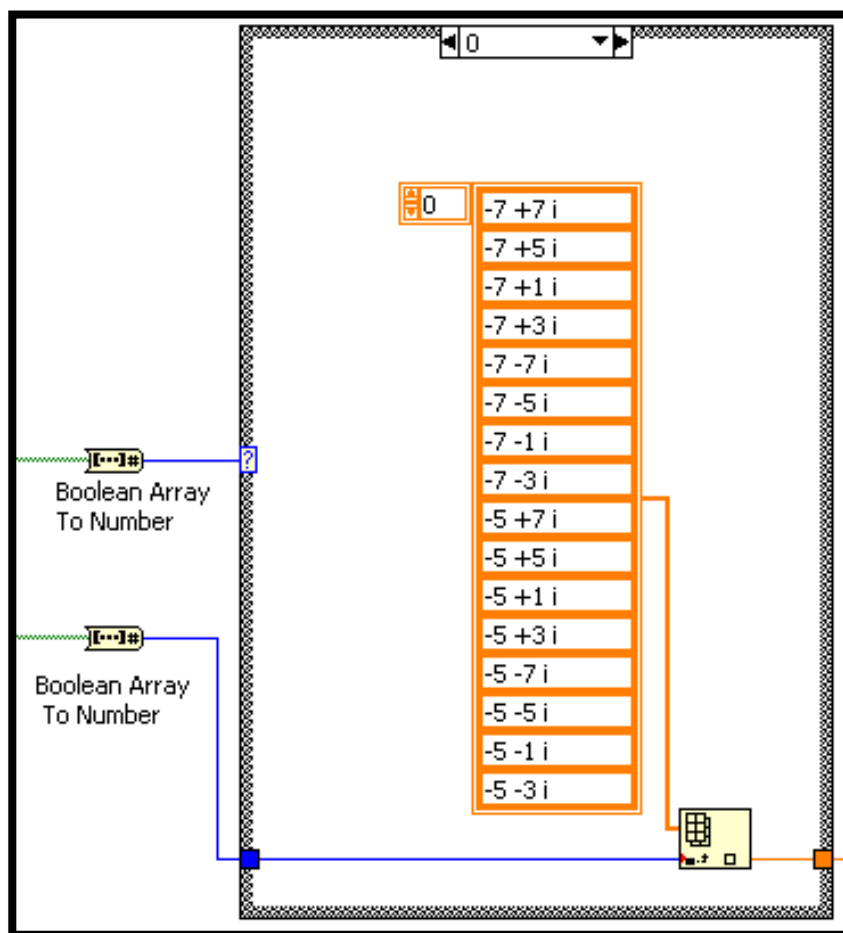


Fig. 4.38. Arreglo de símbolos 64 – QAM región 0

Si es 01 nos encontramos en el sub-diagrama 1 con el siguiente arreglo de símbolos:

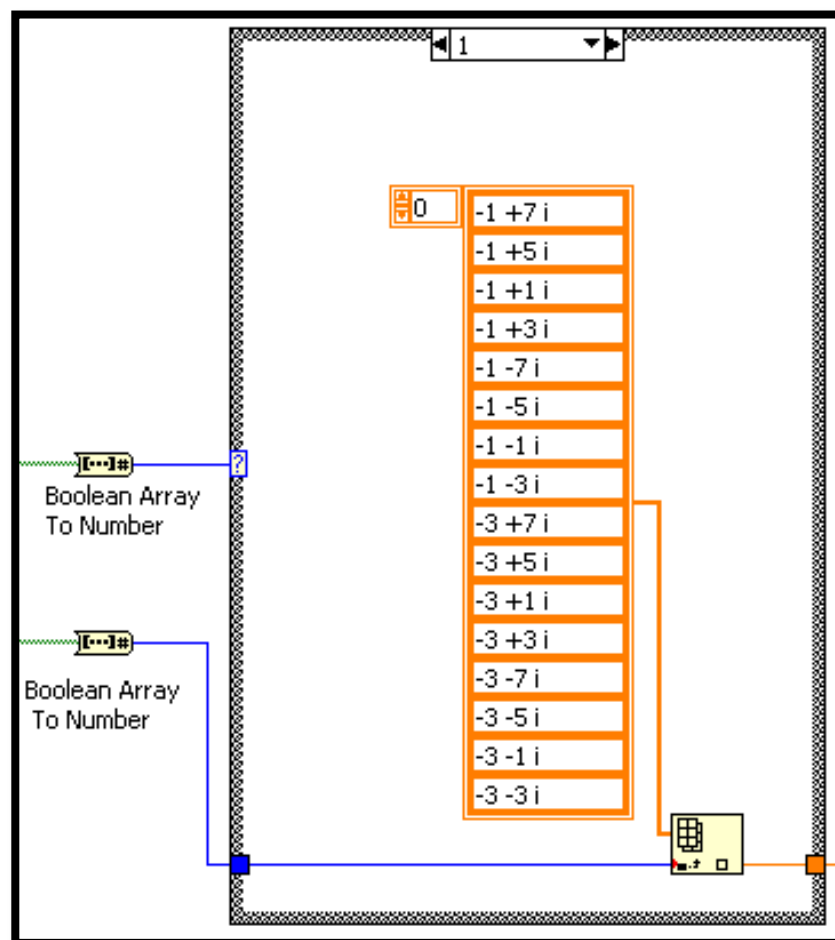


Fig. 4.39. Arreglo de símbolos 64 – QAM región 1

Si es 10 nos encontramos en el sub-diagrama 2 con el siguiente arreglo de símbolos:

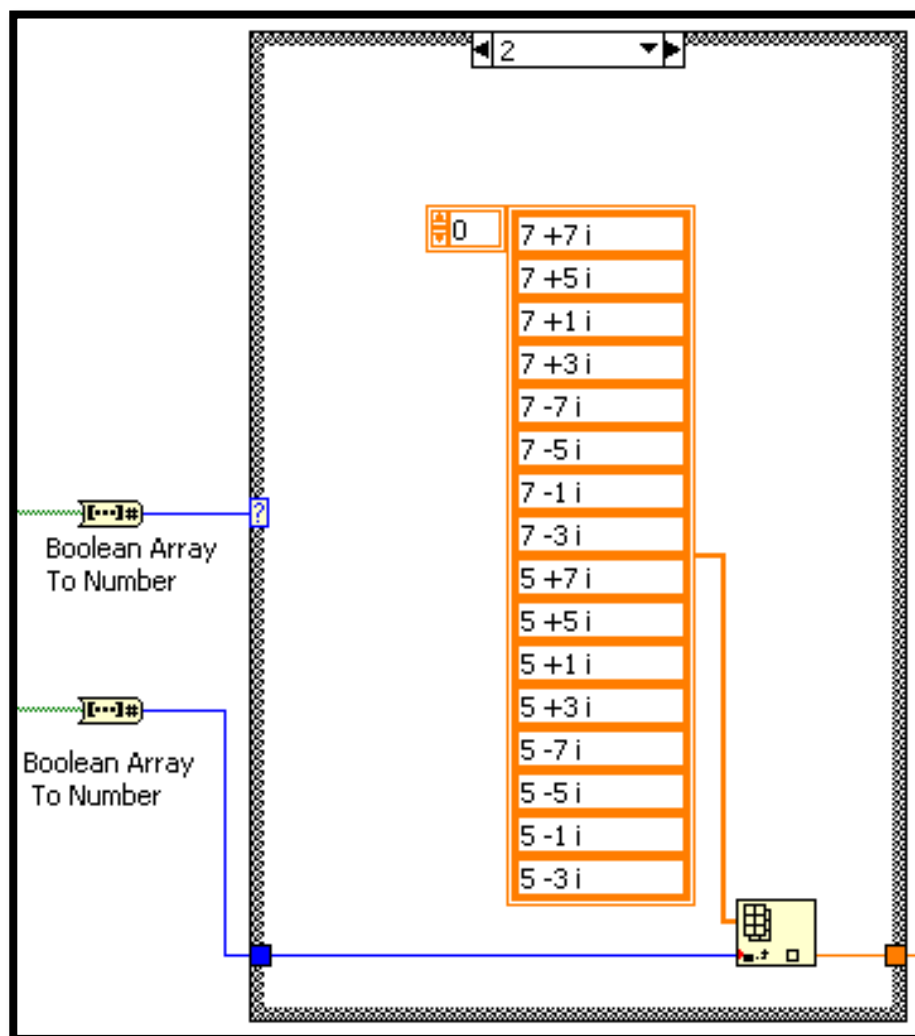


Fig. 4.40. Arreglo de símbolos 64 – QAM región 2

Si es 11 nos encontramos en el sub-diagrama 3 con el siguiente arreglo de símbolos:

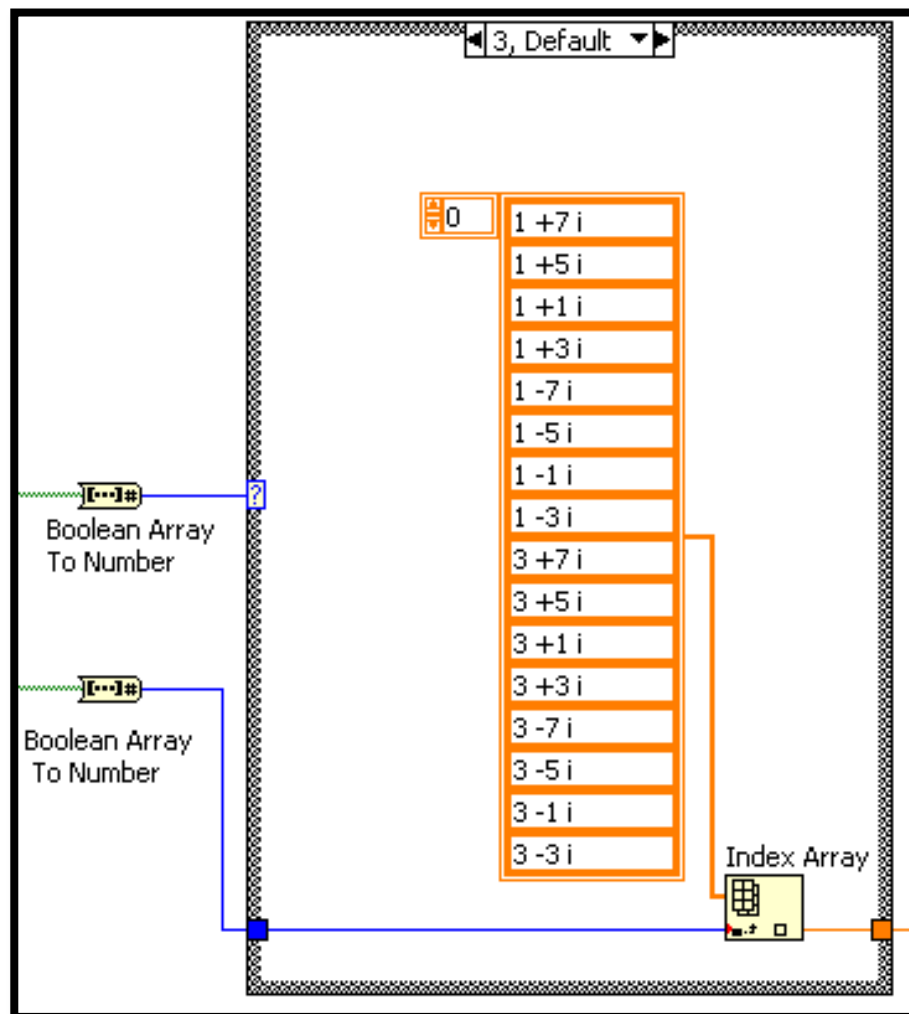


Fig. 4.41. Arreglo de símbolos 64 – QAM región 3

9. Una vez que obtenemos los símbolos hay que normalizarlos, esta normalización se logra dividiendo para la mayor magnitud de la constelación 64-QAM que en este caso es $\overline{98} = 9,8994$.

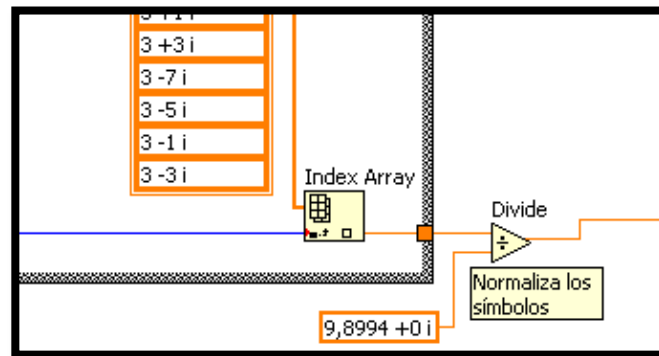


Fig. 4.42. Normalizar los símbolos 64 – QAM

Finalmente para determinar la condición de stop del lazo While se compara el tamaño de la segunda porción de la secuencia de entrada con mayor o igual a 6 ya que se necesitan 6 bits para modular en 64-QAM, para obtener el tamaño del arreglo se utiliza la función **Array Size** que se encuentra en la **paleta de funciones/ Array** y para hacer la comparación se utiliza la función **Greater Or Equal?** que se localiza en la **paleta de funciones/ Comparison** como se muestra en la figura 3.27.

Diagrama de bloques de modulación 64-QAM.

La figura 3.43 muestra el diagrama de bloques correspondiente a la modulación 64-QAM.

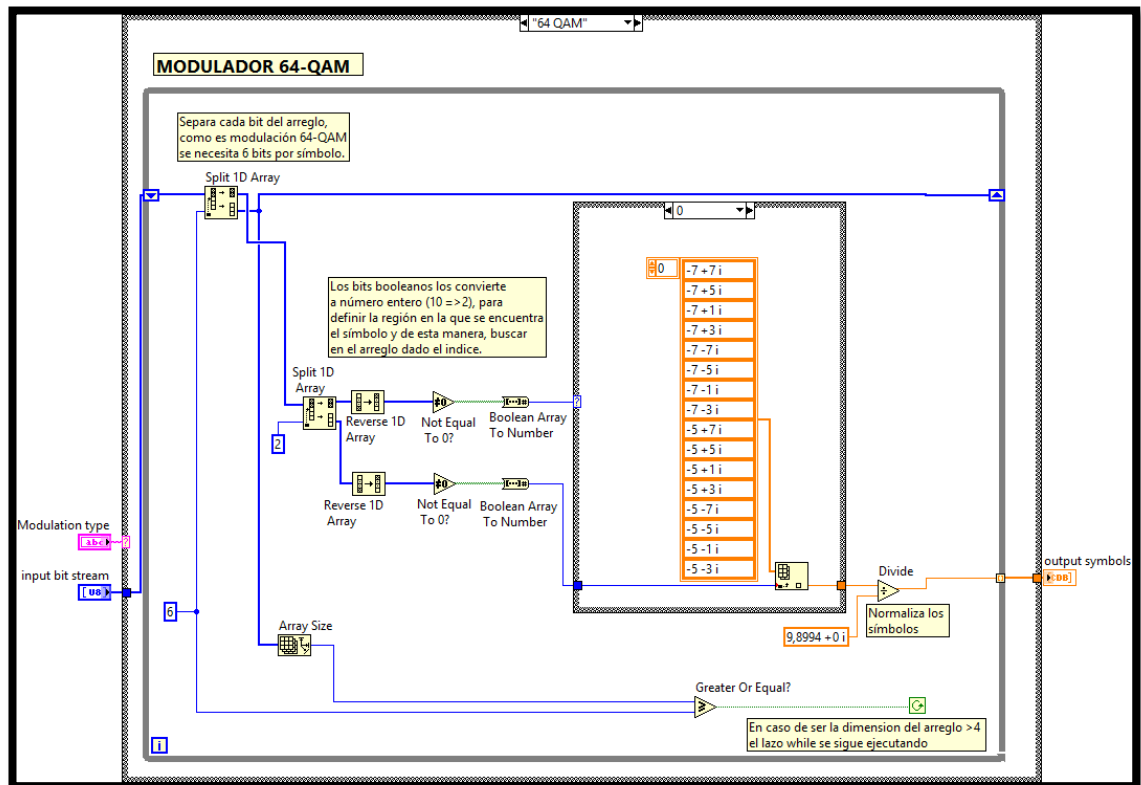


Fig. 4.43. Diagrama de bloques del Sub VI 64 – QAM

3.1.2 ADD CONTROL (Secuencia de Entrenamiento)

En este Sub-VI se añadió una secuencia de entrenamiento (Secuencia de Willard), además de modular las secuencias de acuerdo al tipo de modulación del transmisor.

Panel Frontal del Instrumento Virtual: El panel frontal del Sub-VI muestra controles e indicadores de arreglos tipo numérico flotante, también clusters que contienen los parámetros de modulación, a

continuación se describen brevemente todos los elementos que contiene dicho panel frontal:

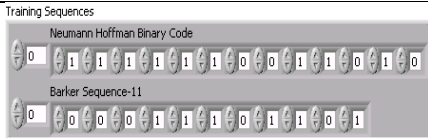
Pestaña de Controles >> Array, Matrix & Cluster>> Cluster >> Array, Matrix & Cluster>> Array >> Numeric >> Numeric Control	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Control que permite ingresar las secuencias de entrenamiento Neuman Hoffman y Barker de tamaño 11.	

Fig. 4.44. Control Cluster Training Sequences

Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Trama de símbolos de entrada al Sub-VI Add control, estos son los símbolos modulados.	

Fig. 4.45. Control Array Input

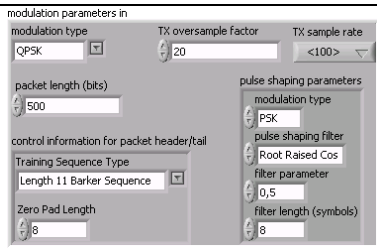
Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Dentro de este cluster tenemos los parámetros de modulación del programa principal tales como: tipo de modulación, cantidad de bits a modular, factor de muestreo, velocidad de muestreo, parámetros del pulse shaping y tipo de secuencia de entrenamiento.	

Fig. 4.46. Control Cluster Modulation Parameters In

Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Trama de símbolos añadida una secuencia de entrenamiento de cabecera y una cola zero pad.	

Fig. 4.47. Indicador tramas con secuencia de entrenamiento

Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Secuencia de entrenamiento modulada.	

Fig. 4.48. Indicador Símbolos de secuencia de entrenamiento

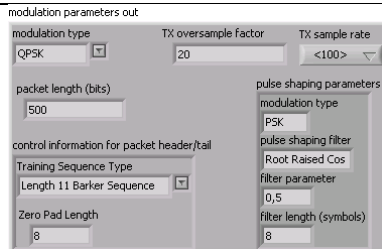
Pestaña de Controles >>Array, Matrix & Cluster>> Array	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Dentro de este cluster mostramos los parámetros de modulación del programa principal tales como: tipo de modulación, cantidad de bits modulados, factor de muestreo, velocidad de muestreo, parámetros establecidos del pulse shaping y tipo de secuencia de entrenamiento seleccionada.	

Fig. 4.49. Indicador Cluster Modulation Parameters Out

Diagrama de bloques del Instrumento Virtual: A continuación se describe las funciones utilizadas para modificar el Sub-VI Add Control.

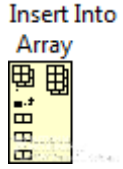
Programming>> Array >> Insert Into Array
DIAGRAMA DE BLOQUES

Inserta al arreglo de entrada un subarreglo de n-dimensiones en un índice especificado.

Fig. 4.50. Insert Into Array

Programming >>Array>> Interleave 1D Array
DIAGRAMA DE BLOQUES

Alterna cada elemento de los arreglos de entrada.

Fig. 4.51. Interleave 1D Array

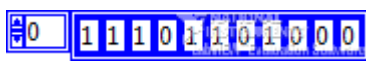
Programming >>Array >>Array Constant
DIAGRAMA DE BLOQUES

Inserta un arreglo de constantes enteras que representan los bits de la secuencia de Willard.

Fig. 4.52. Array Constant (Willard Sequence)

Diseño del VI para el esquema Add Control:

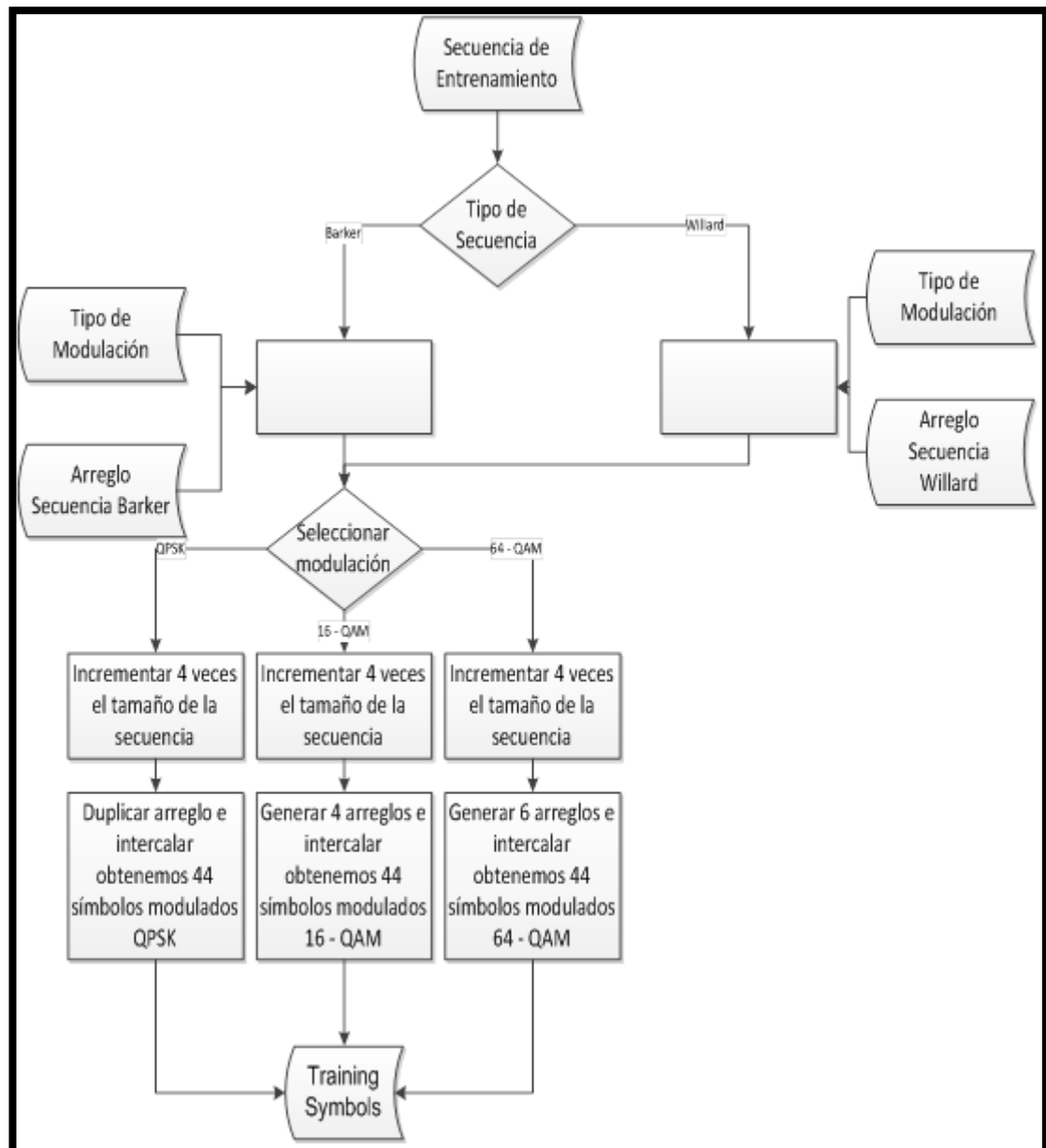


Fig. 4.53. Diagramas de flujo VI Add Control

Para modificar el VI *Add Control* se siguieron los siguientes pasos que se muestran a continuación.

1. Situar en el sub-diagrama de la estructura case “*Barker Sequence-11*”, luego ir al sub-diagrama “*QPSK*” colocarse en el **selector label** clic derecho **Add Case After**.

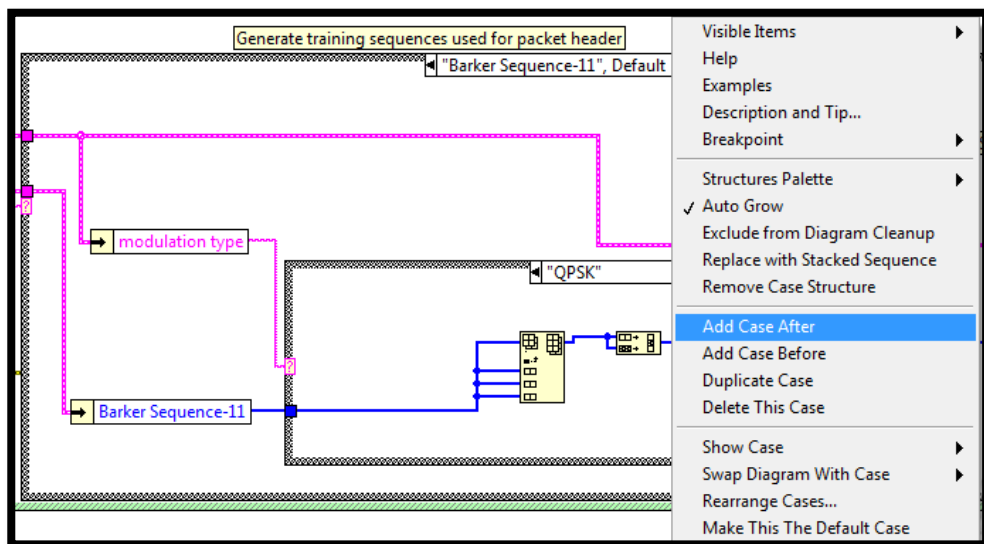


Fig. 4.54. Add Case After

2. En el **label** del nuevo caso escribimos “*16 QAM*”, en este subdiagrama insertamos la función ***Insert Into Array*** y ***Interleave 1D Arrays*** que se encuentra en la **paleta de funciones/ Array**. Lo mismo hacemos para los otros casos de

modulación QPSK y 64 QAM , para cada caso en particular necesitamos formar una secuencia de entrenamiento de 44 bits por tal motivo empleamos la función ***Insert Into Array*** con 4 entradas para los nuevos sub-arreglos añadidos; además según la cantidad de bits que se necesiten para modular se agregan nuevas entradas a las función ***Interleave 1D Arrays***, por ejemplo en el caso 64 QAM se necesitan 6 bits para modular por lo tanto se hace una copia 6 veces cada bit de la secuencia de entrenamiento mediante la función mencionada previamente.

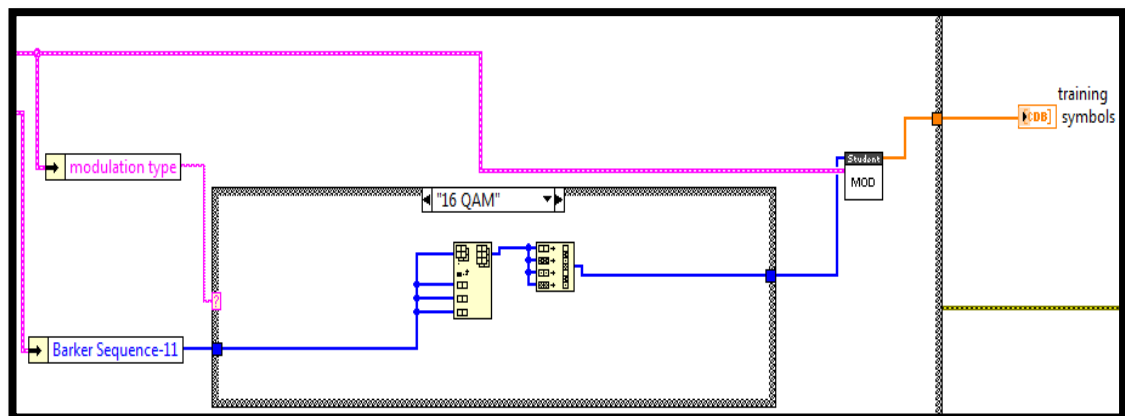


Fig. 4.55. Secuencia Barker 16 – QAM

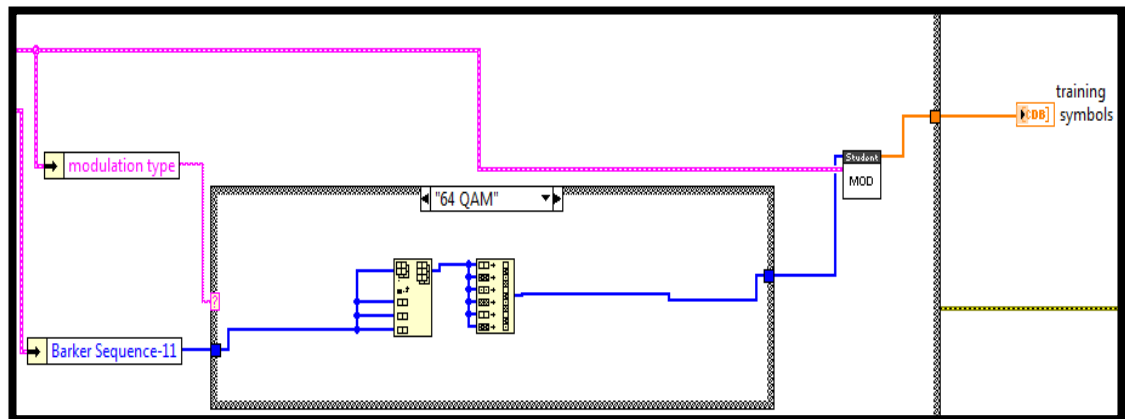


Fig. 4.56. Secuencia Barker 64 - QAM

3. Implementamos una nueva secuencia llamada Willard la misma que será modulación de acuerdo al tipo de modulación del programa principal, como se modulo la Secuencia Barker; como tenemos un procedimiento igual al anterior lo que vamos a hacer es colocarnos en el en el subdiagrama de la estructura case "Barker Sequence-11" clic derecho en el label **Duplicate Case** y colocamos como label "Willard Sequence-11".

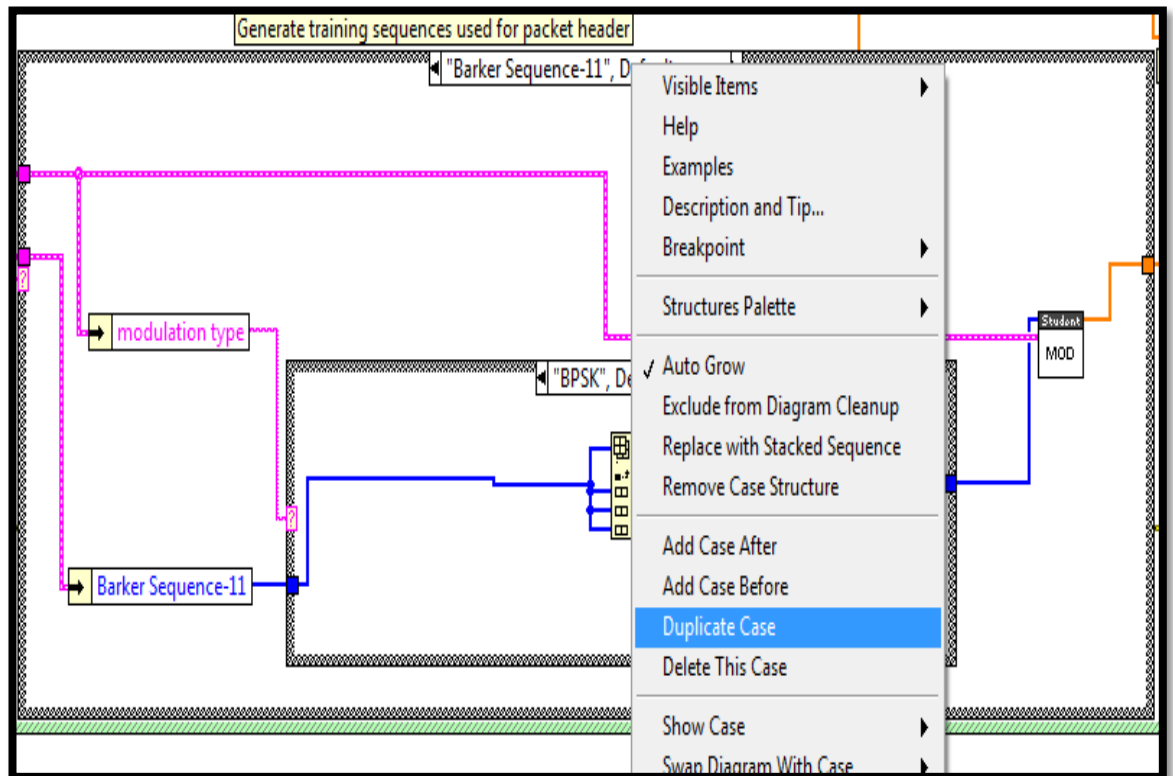


Fig. 4.57. Duplicate Case Barker Sequence

4. En el nuevo sub-diagrama lo que modificamos es la secuencia de entrada, borramos el cluster **Barker Sequence-11**, y creamos un arreglo de constantes para lo cual vamos a la **paleta de funciones/ Array/ Array Constant** insertamos dentro del arreglo un dato tipo entero **paleta de funciones/ Numeric/ Numeric Constant**.

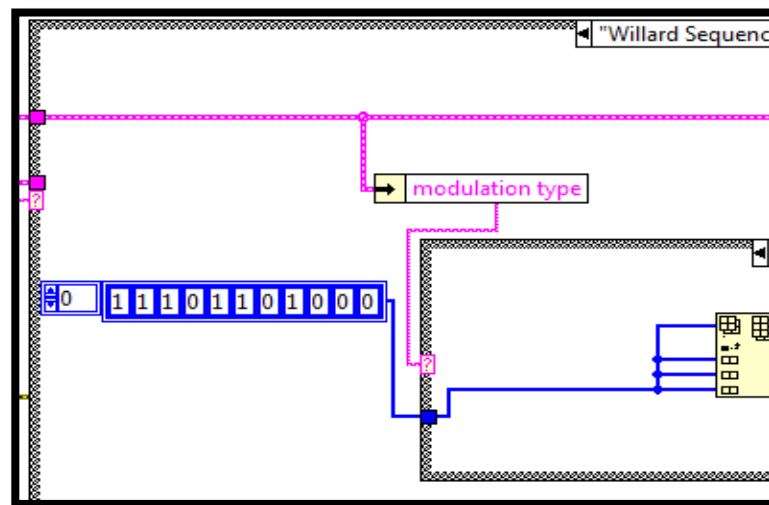


Fig. 4.58. Arreglo de Bits Willard Sequence

3.1.3 CANAL

Para analizar cómo afecta el canal a nuestra transmisión era necesaria la implementación de canales selectivos en frecuencia como Rayleigh (sin línea de vista); la implementación de este Sub VI se describe posteriormente.

Panel Frontal del Instrumento Virtual: Los controles e indicadores utilizados para el Sub-VI del canal Rayleigh se muestran a continuación:

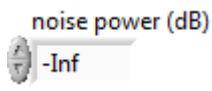
Pestaña de Controles >> Numeric >> Numeric Control	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Control que permite ingresar un valor de ruido en dB.	

Fig. 4.59. Control Noise Power

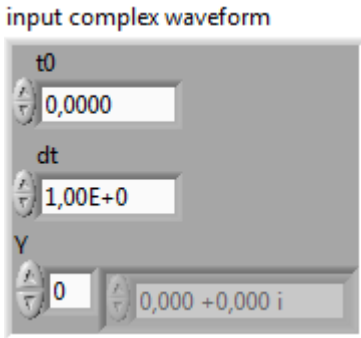
Pestaña de Controles >>Array, Matrix & Cluster>> Cluster	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Señal que viene del transmisor, la cual contiene parámetros como dt y el arreglo de símbolos.	

Fig. 4.60. Input Complex Waveform

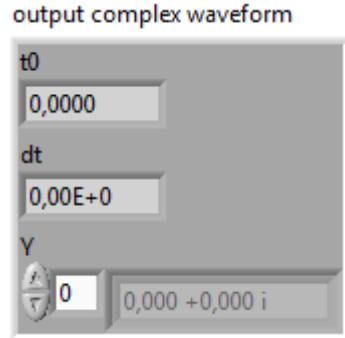
Pestaña de Controles >>Array, Matrix & Cluster>> Cluster	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Señal que sale del canal, la cual contiene parámetros como dt y el arreglo de símbolos.	

Fig. 4.61. Output Complex Waveform

Diagrama de bloques del Instrumento Virtual: A continuación se describe las funciones utilizadas para crear el Sub-VI del canal Rayleigh, algunas funciones ya fueron empleadas anteriormente en la implementación de los Sub VI's como las que se describieron en las figuras 3.10, 3.12, 3.13, 3.15 y 3.46.

Programming>> Cluster, Class & Variant >> Unbundle By Name y Bundle By Name	
DIAGRAMA DE BLOQUES	
	
Descompone un cluster, muestra todos los elementos con sus respectivos nombres contenidos en él. Bundle by	

name reemplaza en el cluster de entrada el arreglo Y.

Fig. 4.62. Unbundle y Bundle By Name

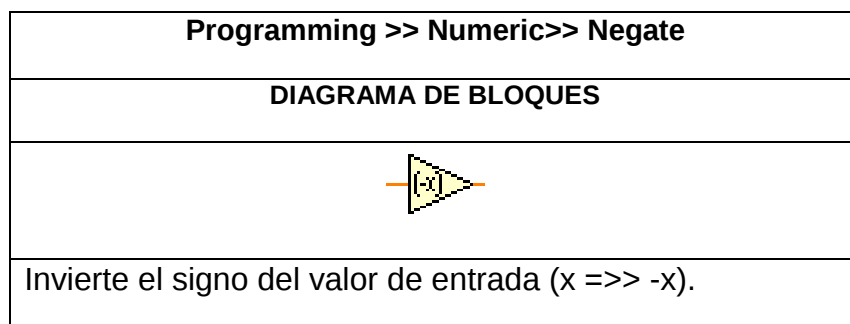


Fig. 4.63. Negate

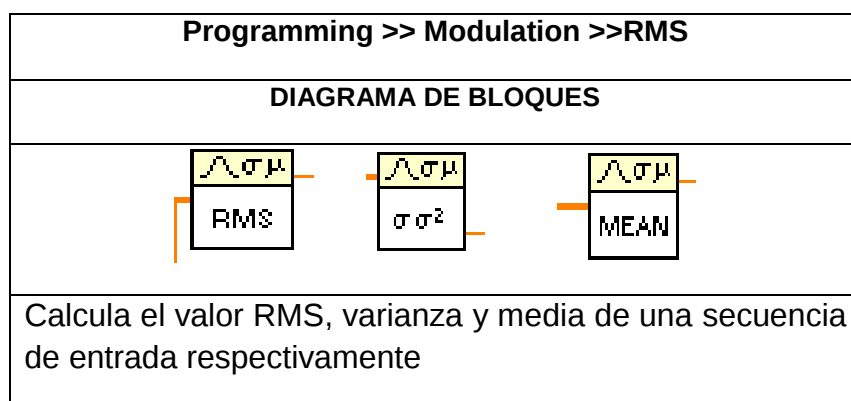
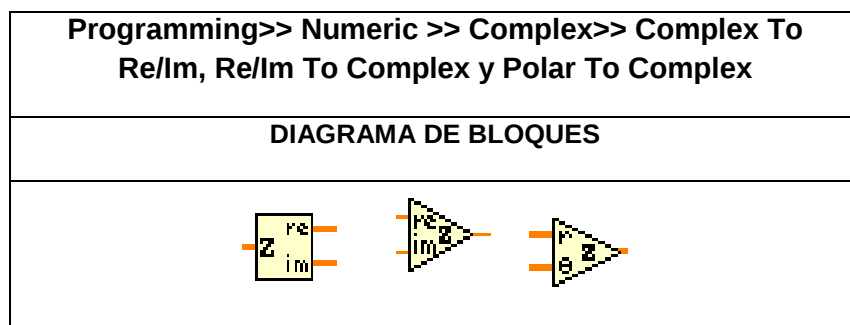


Fig. 4.64. RMS, Desviation & Variance y Mean

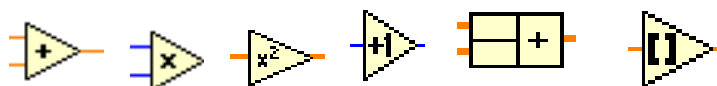


Descompone un número complejo en sus componentes rectangulares, componentes rectangulares forma un número complejo y componentes polares las convierte a rectangulares respectivamente.

Fig. 4.65. Funciones Complex

Programming >> Numeric >> Add, Multiplique, Square, Increment, Compound Aritmetic y Round To Nearst

DIAGRAMA DE BLOQUES

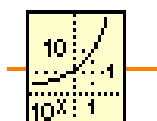


Suma dos variables, multiplica, eleva al cuadrado, incrementa en 1, suma varias variables y redondea al entero más cercano respectivamente.

Fig. 4.66. Funciones Numeric

Programming >> Mathematics>> Elementary>> Exponential>> 10^x

DIAGRAMA DE BLOQUES



Calcula la potencia 10 de la variable de entrada x.

Fig. 4.67. 10^x

Programming >> Signal Processing>> Signal Generation>> Gaussian White Noise
DIAGRAMA DE BLOQUES

Genera una distribución gaussiana con media 0 y desviación estándar s.

Fig. 4.68. Gaussian White Noise

Programming >> Mathematics >> Elementary>> Trigonometric Functions>> Inverse Tangent
DIAGRAMA DE BLOQUES

Calcula la tangente inversa de las dos variables de entrada.

Fig. 4.69. Inverse Tangent

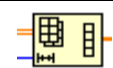
Programming >> Array>> Reshape Array
DIAGRAMA DE BLOQUES

Cambia la dimensión de un arreglo acorde al valor entero que se cablee en la entrada dimensión.

Fig. 4.70. Reshape Array

Diseño del VI para el esquema RayChannel:

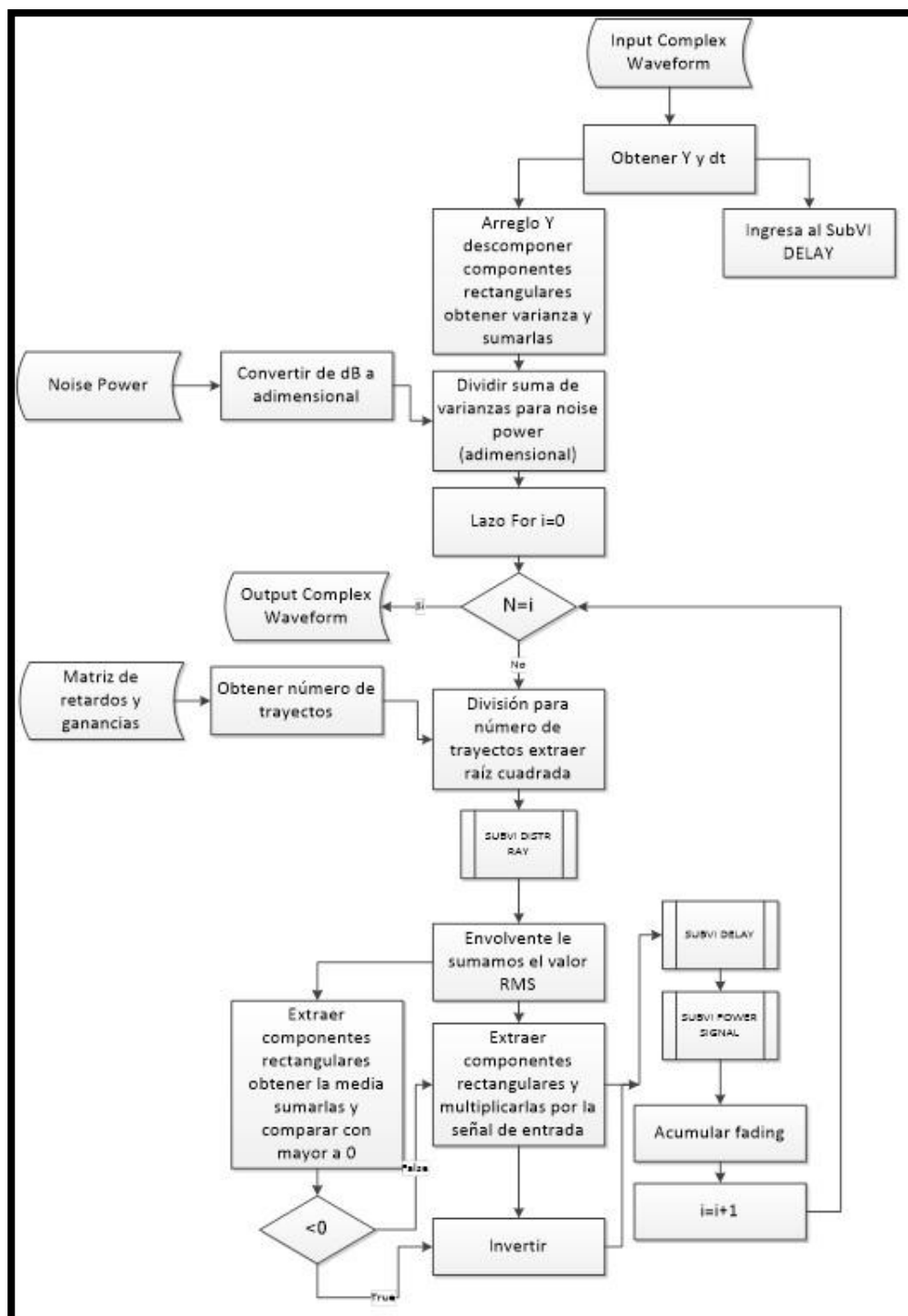


Fig. 4.71. Diagrama de flujo VI RayChannel

SUBVI DISTR RAY

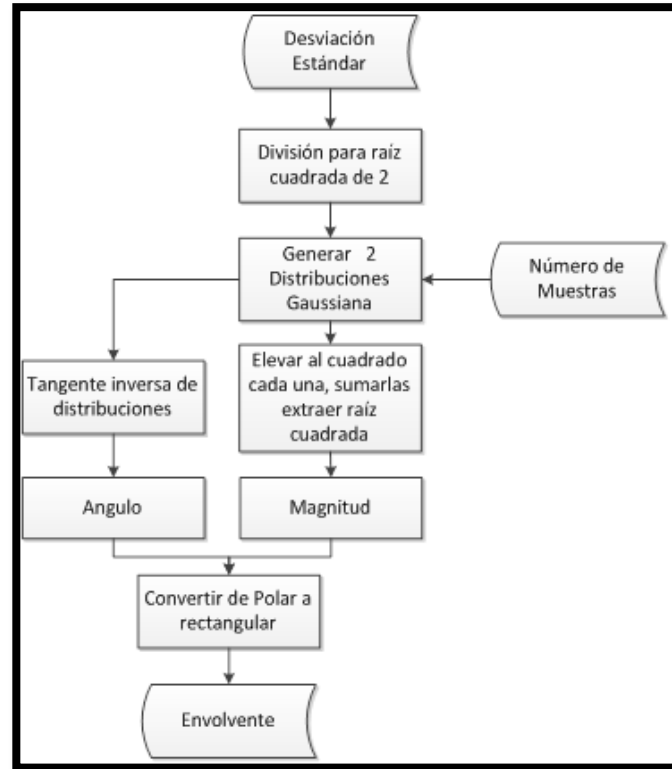


Fig. 4.72. Diagrama de flujo SUB VI Distr Ray

SUBVI POWER SIGNAL

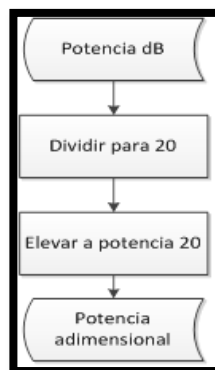


Fig. 4.73. Diagrama de flujo SUB VI Power Signal

SUBVI DELAY

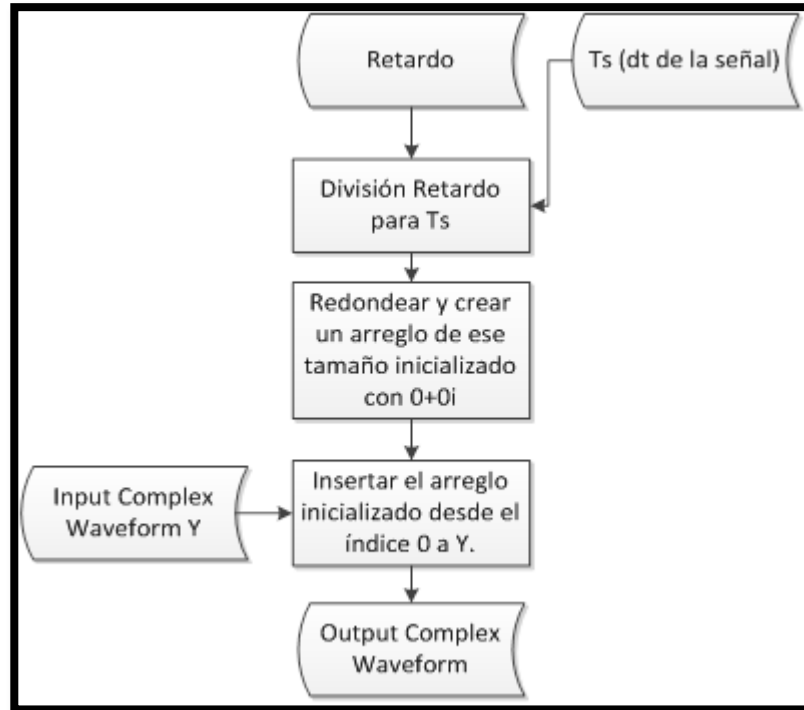


Fig. 4.74. Diagrama de flujo SUB VI Delay

1. Una vez ingresado el control que se muestra en la figura 3.55, se cablea a la entrada de la función **Unbundle By Name** la cual separa el arreglo de entrada Y y el dt de la señal, del arreglo Y obtenemos su dimensión **Size Array** que será el número de muestras.

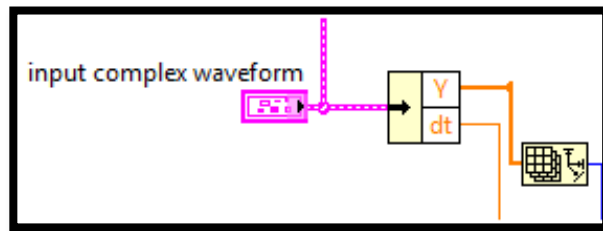


Fig. 4.75. Obtener dt y número de muestras

2. Del arreglo Y que son los símbolos complejos descomponemos en su parte real e imaginaria **Complex To Re/Im**, de cada parte extraemos la varianza mediante la función **Deviation and Variance** y las sumamos utilizando **Add**.

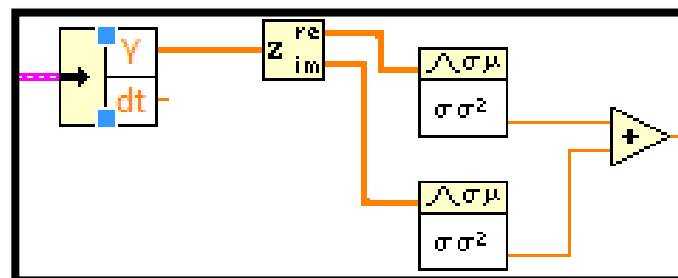


Fig. 4.76. Suma de varianza real e imaginaria de Y

3. La suma de varianzas la dividimos **Divide** para el Noise Power que es un valor que ingresa el usuario en el respectivo control que se muestra en la figura 3.54, como este valor está en dB se hace la conversión a Watts; para esta conversión se invierte el

signo de Noise Power **Negate** luego se divide para 20 **Divide** y se aplica potencia en base 10 **10^x**.

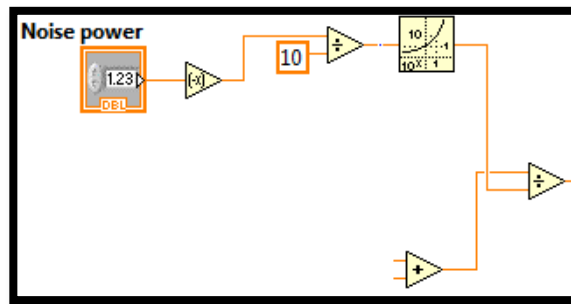


Fig. 4.77. Suma de varianzas dividida para Noise Power

4. El resultado de la división anterior la dividimos **Divide** para el número de trayectos, este número de trayectos lo obtenemos de nuestra matriz de ganancias y retardos; para lo cual vamos a utilizar la función **Size Array** como ingresa una matriz me devolverá el tamaño de las filas y columnas en un vector para extraer solo el tamaño ya sea de las filas o columnas se utiliza la función **Index Array** en la entrada index crear una constante de valor 0 este número de trayectos también será el número de fading que se generen mediante una lazo for **N** , una vez obtenida la división para el número de trayectos extraemos la raíz cuadrada **Square Root** que será el valor que ingrese al Sub VI **DISTR RAY** como desviación estándar.

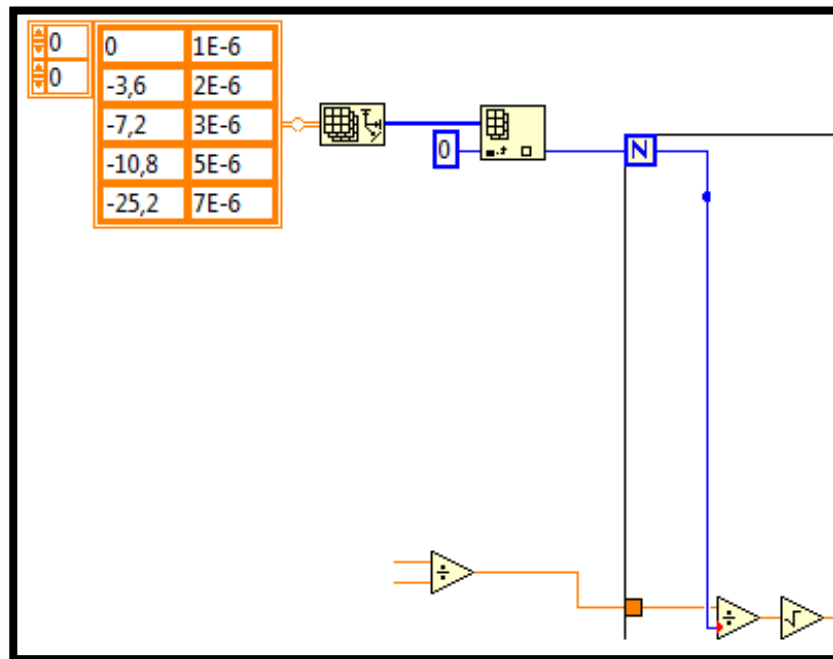


Fig. 4.78. Varianza que ingresa a DISTR RAY

5. Creamos el VI DISTR RAY colocamos dos controles uno para el número de muestras y para la desviación estándar y un indicador tipo Array para la envolvente, la desviación estándar la dividimos **Divide** para raíz de 2 **Square Root** con esta nueva desviación generamos dos distribuciones gaussianas **Gaussian White Noise** las elevamos al cuadrado **Square** sumamos las dos distribuciones **Add** y extraemos la raíz cuadrada **Square Root** de esta manera tenemos la magnitud de la envolvente, para el ángulo usamos la función **Inverse Tan**; finalmente convertimos a complejos con la magnitud y fase mediante **Polar**

to Complex y este es el resultado que se cablea al indicador envolvente. Guardamos como Sub VI asignando sus conectores respectivamente como se explicara en el **ANEXO 1**.

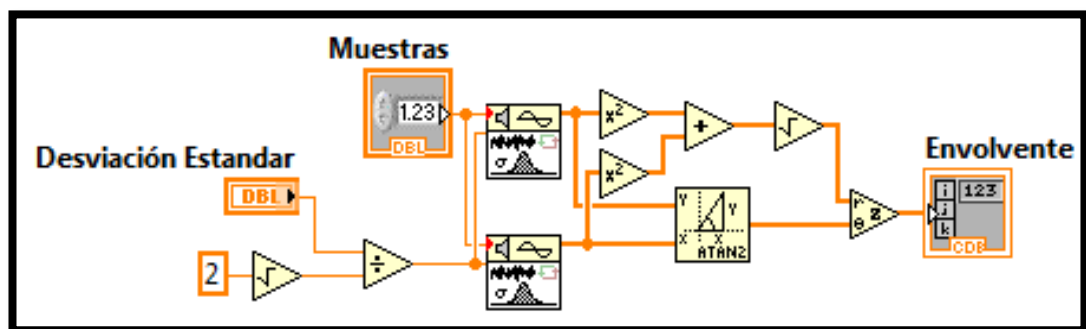


Fig. 4.79. VI DISTR RAY

6. En el diagrama de bloques del VI RayChannel clic derecho **Select a VI** seleccionar el Sub VI DISTR RAY colocarlo dentro del lazo for y cablear desviación estándar, número de muestras obtenidas anteriormente.

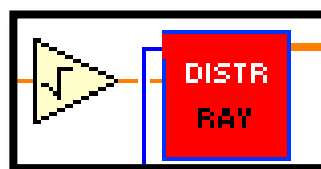


Fig. 4.80. Insertar DISTR RAY

7. Debido a que la media de la envolvente es aproximadamente cero le sumamos un valor RMS dividido para 10 factor que fue determino experimentalmente.

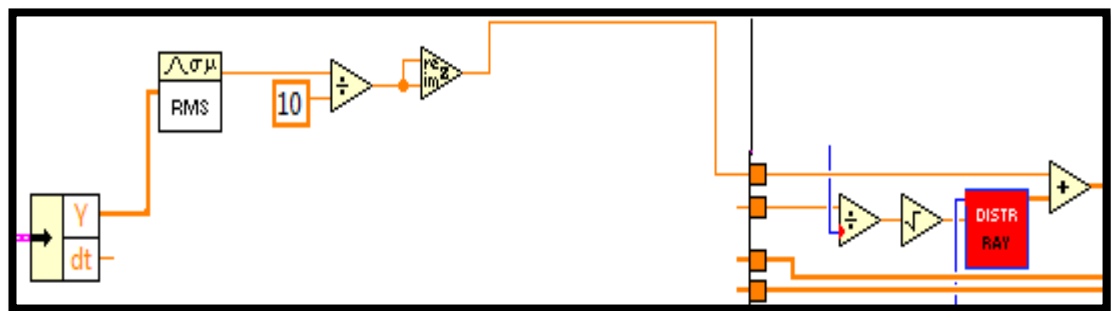


Fig. 4.81. Sumar un valor RMS a la envolvente

8. Luego de sumar el valor RMS descomponer cada envolvente generada en el lazo for en su parte real e imaginaria **Complex To Re/Im** obtener de cada componente la media **Mean** y sumarlas **Add** comparar ese valor con menor a cero **Less To 0?** esta será la condición de **Case Structure** al borde de esta estructura llega la multiplicación de las componentes reales de la envolvente sumada el RMS con la parte real de la señal de entrada **Input Complex Waveform** los mismo se hace con las componentes imaginarias, en el subdiagrama True se invierte el signo de este valor y en el caso False se pasa el valor sin invertir el signo como se muestra en la siguiente figura.

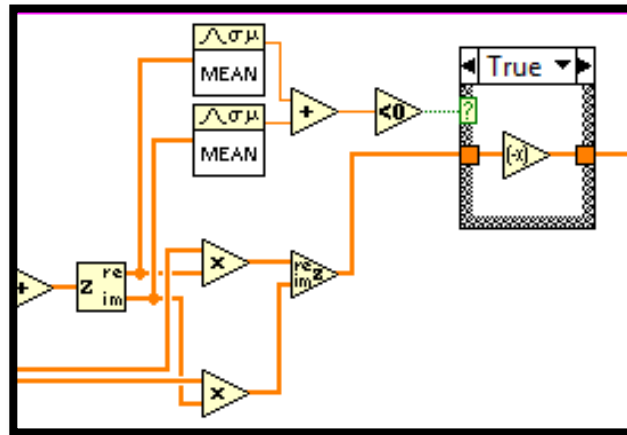


Fig. 4.82. Comparar la media de la envolvente con menor a 0

9. Una vez generado el fading en cada iteración vamos a aplicar la respuesta del canal ganancia y retardo, formar la señal con el fading para eso utilizar función **Bundle By Name** la cual recibe el cluster **Input Complex Waveform** y se reemplaza el arreglo Y por el fading, esa señal recibe el Sub VI DELAY además de recibir el retardo y el dt de la señal que fue calculada anteriormente, el retardo se lo extrae de la matriz de retardos y ganancias para esto primero se divide la matriz utilizando la función **Reshape Array** y con la variable de iteración **i** se extrae cada retardo.

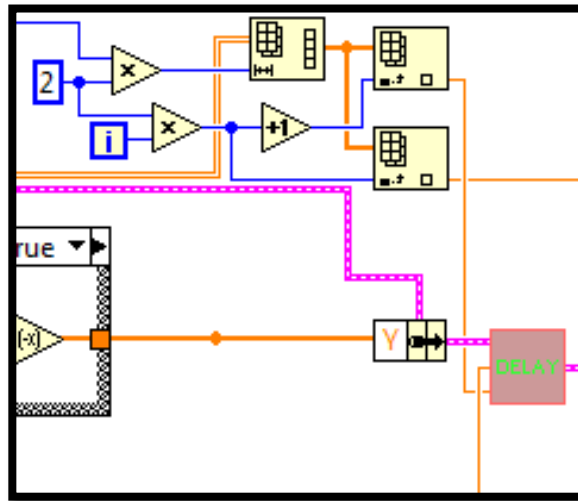


Fig. 4.83. Extraer retardo

10. Crear el VI DELAY dos controles numéricos, tipo cluster señal de entrada y un indicador tipo cluster para la señal de salida, cada retardo se lo divide para el dt luego la división se redondea al entero más cercano **Round To Nearest** este valor será el tamaño de un arreglo de símbolos 0+0i **Initialize Array**, finalmente se inserta este arreglo inicializado desde la posición 0 al arreglo de la señal de entrada mediante la función **Insert Into Array**.

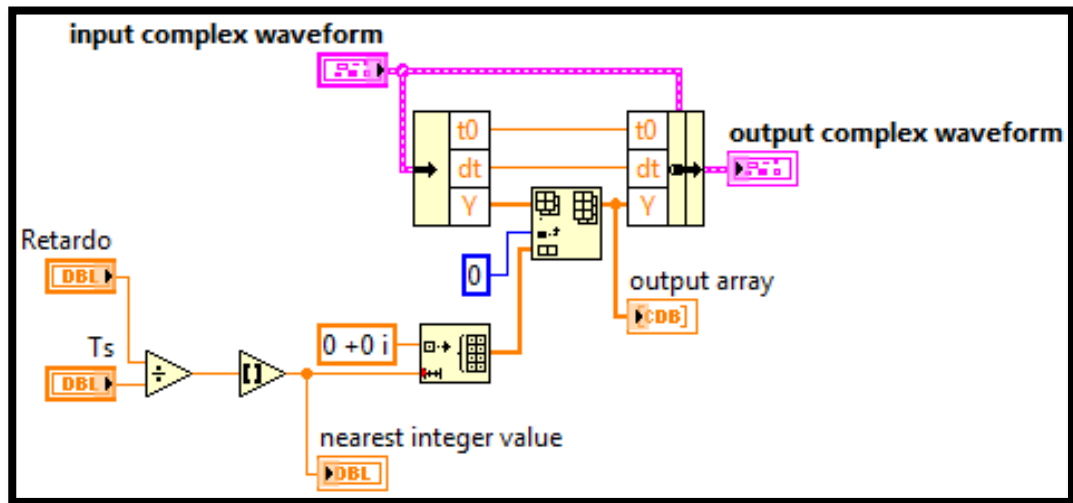


Fig. 4.84. VI DELAY

- 11.** Con la señal fading con retardos extraemos el arreglo Y del cluster mediante **Unbundle By Name** y la multiplicamos por la salida del Sub VI POWER SIGNAL luego la sumamos **Compound Arithmetic** con el Shift Registers de la izquierda que fue previamente inicializado con 0+0i de tamaño (número de muestras por número de trayectos), de esta manera almacenamos cada valor en el lado derecho del Shift Register que se crea en el lazo for.

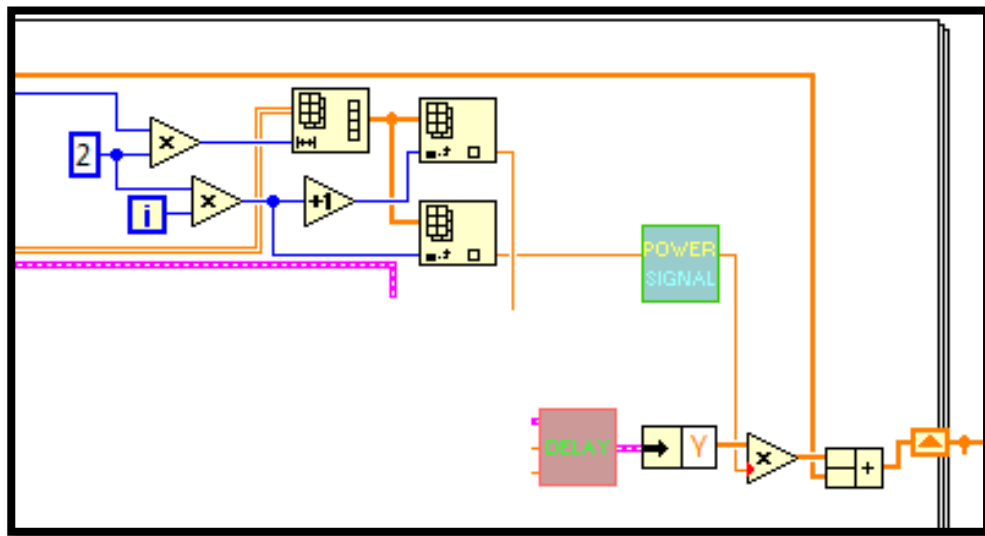


Fig. 4.85. Añadiendo Potencia del trayecto a la señal

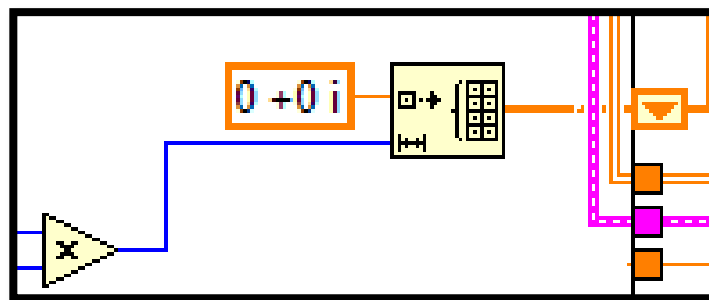


Fig. 4.86. Shift Register inicializado

- 12.** El VI POWER SIGNAL lleva el valor de dB a adimensional y recibe cada valor de ganancia de la matriz. Crearlo como Sub VI e insertarlo en el programa RayChannel.

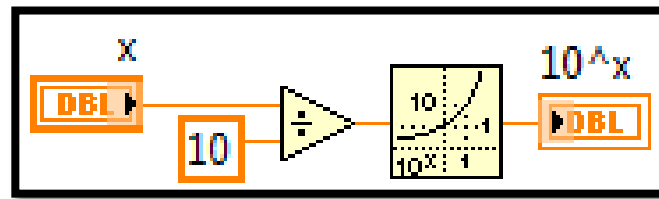


Fig. 4.87. VI POWER SIGNAL

13. El arreglo ya generado el fading y aplicado las ganancias y retardos se forma con el dt de la señal de entrada por lo tanto se utiliza la función **Bundle By Name** que recibe el cluster de entrada **Input Complex Waveform** para formar la onda de salida. Este VI **Ray Channel** añadirlo como Sub VI al bloque **APPLY CHANNEL**.

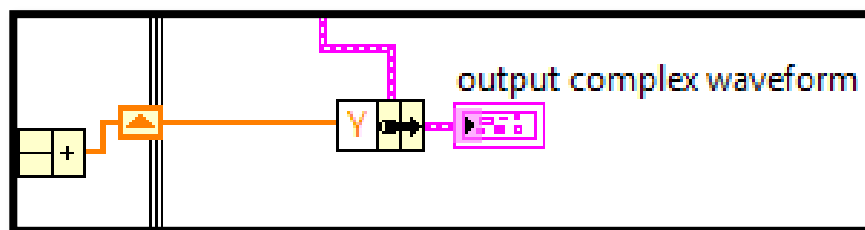


Fig. 4.88. Formar la señal de salida de una canal Rayleigh

Diagrama de bloques del VI RayChannel.

La figura 3.89 muestra el diagrama de bloques correspondiente al canal Rayleigh.

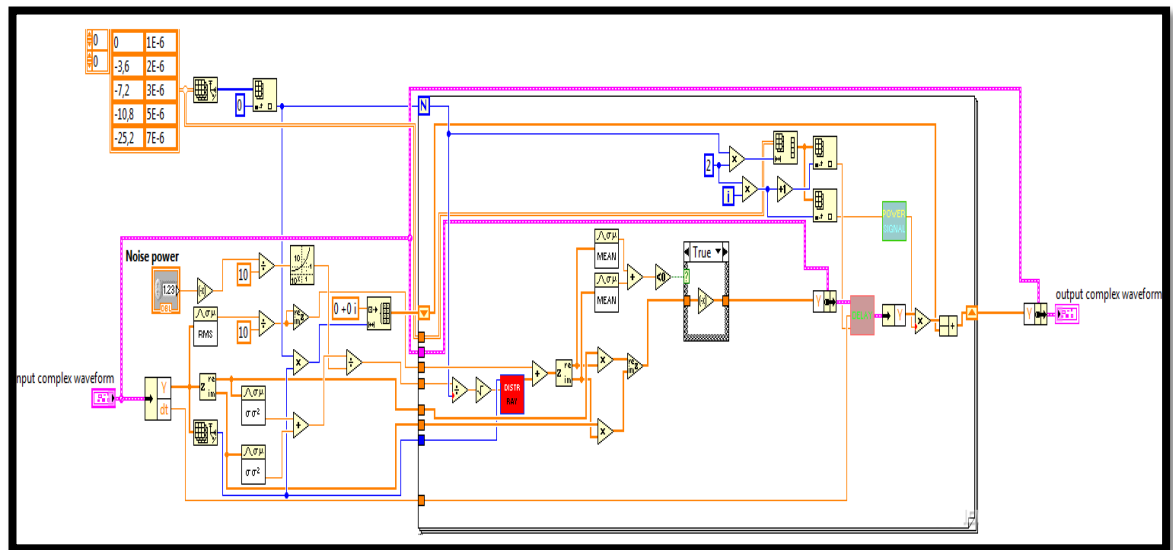


Fig. 4.89. Diagrama de bloques del Sub VI RayChannel

3.1.4 SINCRONIZADOR DE TRAMA SCHMILD & COX

Para encontrar el inicio de la trama utilizamos el algoritmo Schmild & Cox adaptado a portadora única, la implementación de este Sub VI se describe posteriormente.

Panel Frontal del Instrumento Virtual: Los controles e indicadores utilizados para el Sub-VI del Sincronizador de Trama se muestran a continuación, además de crear los que se muestran en las figuras 3.6, 3.42 y 3.55:

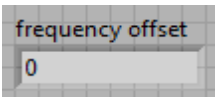
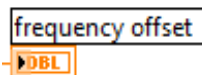
Pestaña de Controles >> Numeric >> Numeric Indicator	
PANEL FRONTAL	DIAGRAMA DE BLOQUES
	
Indicador numérico creado tanto para visualizar frequency offset y frame offset.	

Fig. 4.90. Indicadores Numéricos

Diagrama de bloques del Instrumento Virtual: A continuación se describe las funciones utilizadas para crear el Sub-VI del Sincronizador de Trama, algunas funciones ya fueron empleadas anteriormente en la implementación de los Sub VI's como las que se describieron en las figuras 3.10, 3.11, 3.15, 3.57, 3.60 y 3.61.

Programming>> Stucture>> Stacked Sequence Structure	
DIAGRAMA DE BLOQUES	
	
Consiste en uno o más sub-diagramas que se ejecutan secuencialmente.	

Fig. 4.91. Stacked Sequence Structure

Programming >> Numeric>> Decrement y Subtract
DIAGRAMA DE BLOQUES

Decrementa en 1 y resta dos variables respectivamente.

Fig. 4.92. Decrement y Subtract

Programming >> Numeric >> Absolute Value
DIAGRAMA DE BLOQUES

Calcula el valor absoluto de la entrada

Fig. 4.93. Absolute Value

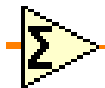
Programming>> Numeric >> Add Array Elements
DIAGRAMA DE BLOQUES

Suma todos los elementos del arreglo de entrada.

Fig. 4.94. Add Array Elements

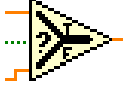
Programming >> Comparison>> Select
DIAGRAMA DE BLOQUES

De acuerdo al valor booleano a la salida tendremos el valor cableado en true or false.

Fig. 4.95. Select

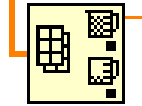
Programming >> Array>> Max & Min Array
DIAGRAMA DE BLOQUES

Retorna el mínimo valor de un arreglo.

Fig. 4.96. Max & Min Array

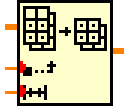
Programming >> Array >> Array Subset
DIAGRAMA DE BLOQUES

Retorna una porción del arreglo de entrada dado un índice y la longitud del nuevo subarreglo.

Fig. 4.97. Array Subset

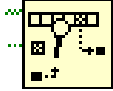
Programming >> Array >> Search Array
DIAGRAMA DE BLOQUES

Busca en un arreglo un elemento específico.

Fig. 4.98. Search Array

Programming >> Numeric>> Complex>> Complex Conjugate
DIAGRAMA DE BLOQUES

Retorna la conjugada de un número complejo ($x+yi \gg x-yi$).

Fig. 4.99. Complex Conjugate

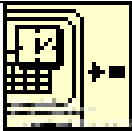
Programming >> Timing>> Get Date/ Time in Seconds
DIAGRAMA DE BLOQUES

Retorna la hora actual del sistema.

Fig. 4.100. Get Date/ Time in Seconds

Diseño del VI para el esquema Schmild & Cox:

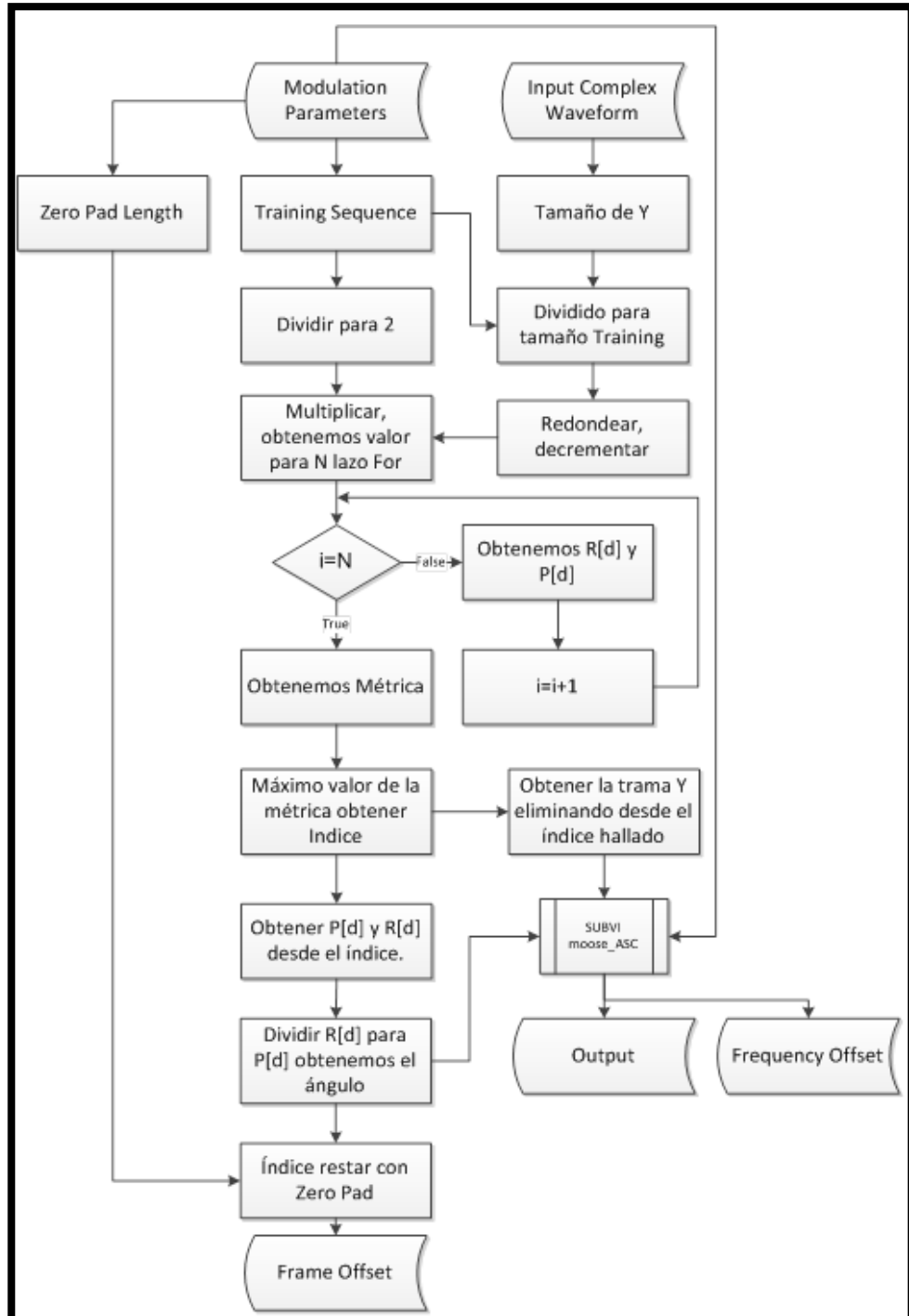


Fig. 4.101. Diagrama de flujo VI Schmild & Cox

1. Insertar la función descrita en la figura 3.83 clic derecho sobre el borde la estructura **Add Case After** esta estructura deberá contener 3 sub-diagramas. En el subdiagrama 0 insertar la función de la figura 3.93 dar clic derecho en el borde del subdiagrama seleccionar Add Sequence Local y cablear a la función que retorna el tiempo actual del sistema. Una vez ingresado el control que se muestra en la figura 3.55.

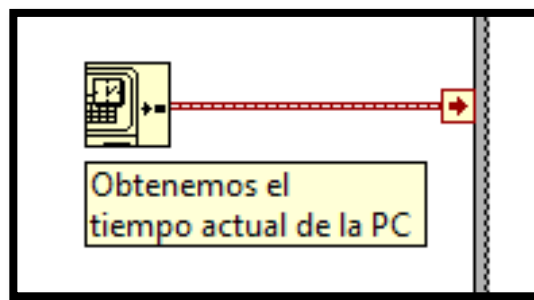


Fig. 4.102. Obtener tiempo actual de la PC

2. Descomponer el cluster **Input Complex Waveform** mediante la función **Unbundle By Name** para obtener el arreglo Y y del cluster modulation parameters extraer la secuencia de entrenamiento, de cada arreglo obtener el tamaño **Array Size**.

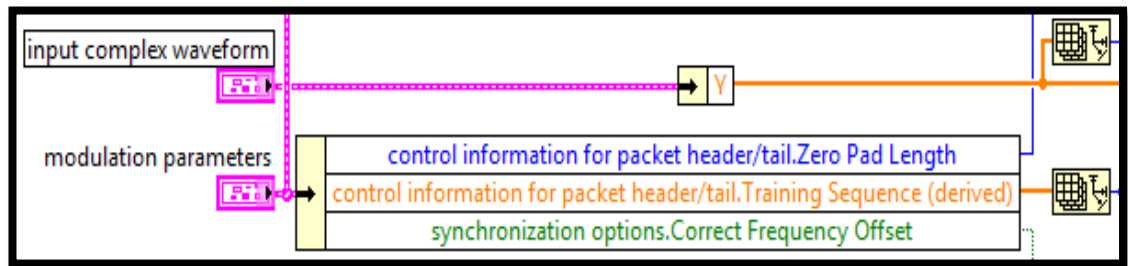


Fig. 4.103. Descomponer cluster Y y secuencia de entrenamiento

3. Dividimos el tamaño de la señal de entrada para el tamaño de la secuencia de entrenamiento, redondeamos ese valor **Round To Nearest** hacemos un decremento en -1 **Decrement** y multiplicamos por el tamaño de la secuencia de entrenamiento divide para 2 (22 símbolos), ese valor será el número de iteraciones N del lazo for.

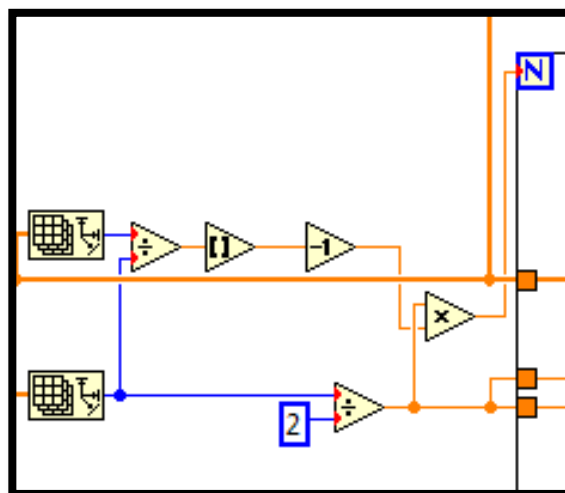


Fig. 4.104. Valor de iteración N

4. Obtenemos $y[n+d]$ para lo cual se utiliza **Array Subset** la cual recibe como índice la variable de iteración i del lazo for y como longitud la mitad del tamaño de la secuencia de entrenamiento y como arreglo de entrada Y y finalmente para obtener $y[n+d+L]$ nuevamente utilizamos **Array Subset** en este caso como índice recibe la variable i mas la mitad del tamaño de la secuencia y como longitud la mitad del tamaño de la secuencia de entrenamiento.

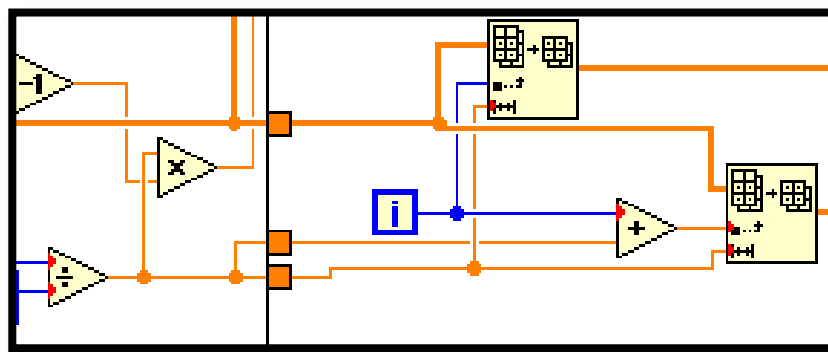
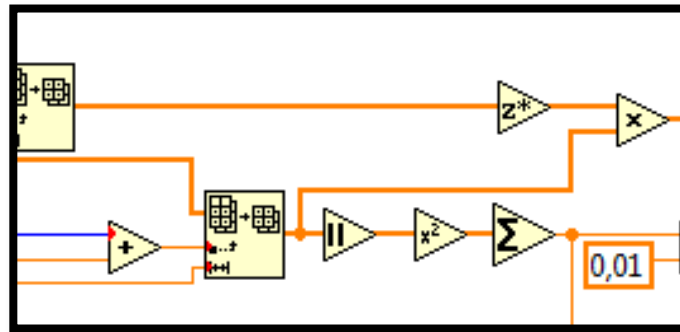
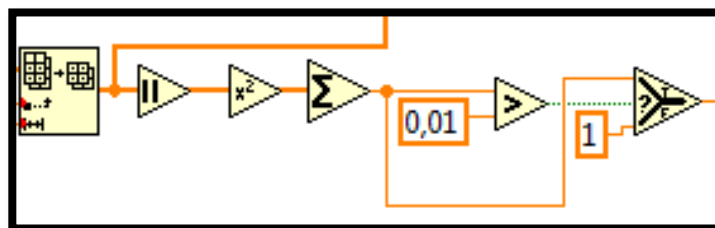


Fig. 4.105. Generando $y[n+d]$ y $y[n+d+L]$

5. Aplicamos conjugada **Conjugate Complex** a $y[n+d]$ y la multiplicamos por $y[n+d+L]$ de esta manera se obtiene $R[d]$.

Fig. 4.106. Generando $R[d]$

6. A $y[n+d+L]$ sacamos el valor absoluto ***Absolute Value*** elevamos al cuadrado ***Square*** realizamos la sumatoria de todos los elementos del arreglo ***Add Array Elements*** $P[d]$ este valor lo comparamos con mayor que 0,01 esa será la condición de la función ***Select*** en la entrada True recibirá $P[d]$ en False se crea una constante de valor 1; esto valida que $P[d]$ este entre $[0,01 - 1]$.

Fig. 4.107. Generando $P[d]$

7. Para obtener la métrica $M[d]$ elevamos al cuadrado **Square** $R[d]$ sacamos el valor absoluto **Absolute Value** y la dividimos **Divide** para el cuadrado de $P[d]$.

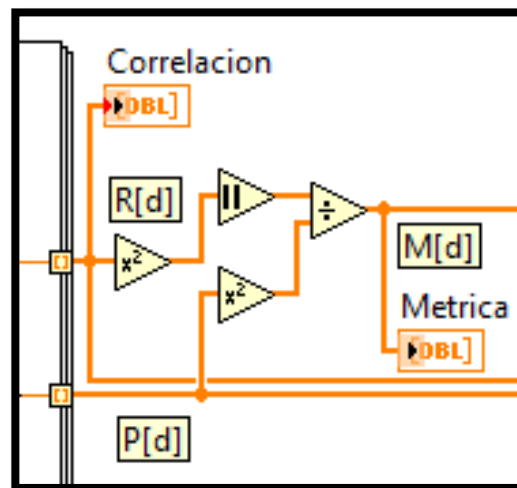


Fig. 4.108. Obtener Métrica $M[d]$

8. Obtenemos el máximo valor de $M[d]$ con la función **Max & Min Array** ese valor máximo lo comparamos con cada uno de los elementos del arreglo $M[d]$ por lo tanto tendremos un arreglo booleano de 0 y 1 en el caso de que el elemento del arreglo sea mayor o igual que el máximo $M[d]$ se llenara el arreglo con un True, luego mandamos a buscar ese valor True en el arreglo booleano mediante **Search 1D Array** la cual retornara el índice que representa el inicio de Trama.

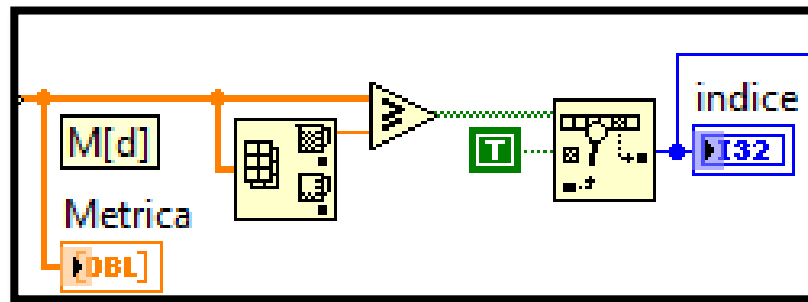


Fig. 4.109. Obtener el Índice del inicio de Trama

9. Como la trama tiene una cabecera Zero Pad Length al índice le restamos el tamaño de esa cabecera que la obtenemos al desagrupar el cluster modulation parameters que se hizo en el paso 1, al arreglo Y que también se obtuvo en ese paso remover los elementos desde el índice que indica el inicio de trama estos elementos no deseados se eliminan utilizando ***Array Subset***.

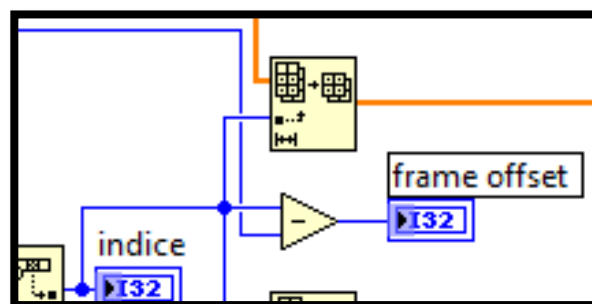


Fig. 4.110. Frame Offset

- 10.** Obtenemos los elementos de $R[d]$ y $P[d]$ desde el inicio de trama mediante ***Index Array***, dividimos esos arreglos como son símbolos complejos extraemos el ángulo ***Complex To Polar***.

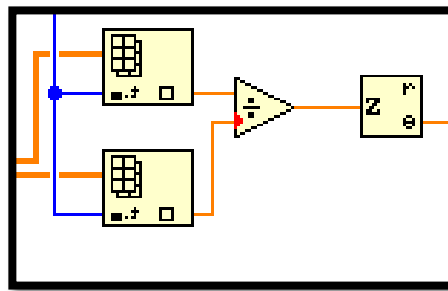


Fig. 4.111. Angulo $R[d]/P[d]$

- 11.** Insertar una estructura Case en el subdiagrama True insertar el Sub VI Moose_ASC que se describirá posteriormente, la condición de la estructura case será un booleano ***Correct Frequency Offset*** que se encuentra en el cluster modulation parameters, en el subdiagrama True el Sub VI recibe el cluster modulation parameters, la trama Y sin elementos no deseados y el ángulo y a la salida cableamos el indicador frequency offset y la trama ya corregida esa frecuencia de desplazamiento y en el subdiagrama False enviamos directamente la trama solo detectada el inicio sin corrección de frecuencia.

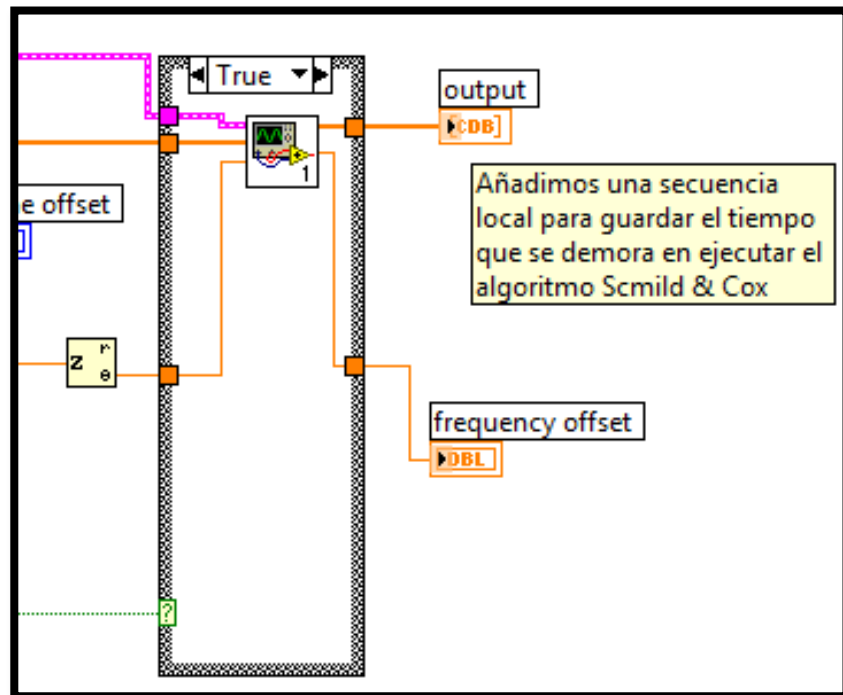


Fig. 4.112. Insertar moose_ASC

12. Obtenemos el tiempo actual del sistema ese valor lo restamos de la **Sequence Local** que almacena el tiempo en que se llevó a cabo el proceso de corrección de trama y frecuencia de desplazamiento, como nos da el valor en segundos lo multiplicamos por 100 para obtener el tiempo de ejecución del VI en milisegundos.

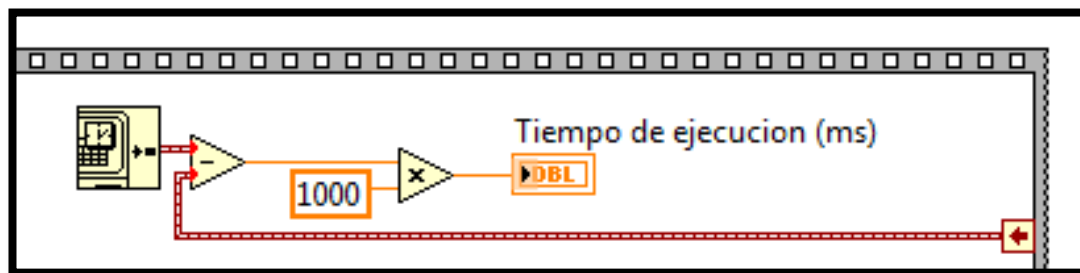


Fig. 4.113. Tiempo de ejecución (ms)

Diagrama de bloques del VI SAC.

La figura 3.114 muestra el diagrama de bloques correspondiente al sincronizador de trama este VI se lo guarda como un Sub VI para insertarlo en el Sub VI **frame detect**.

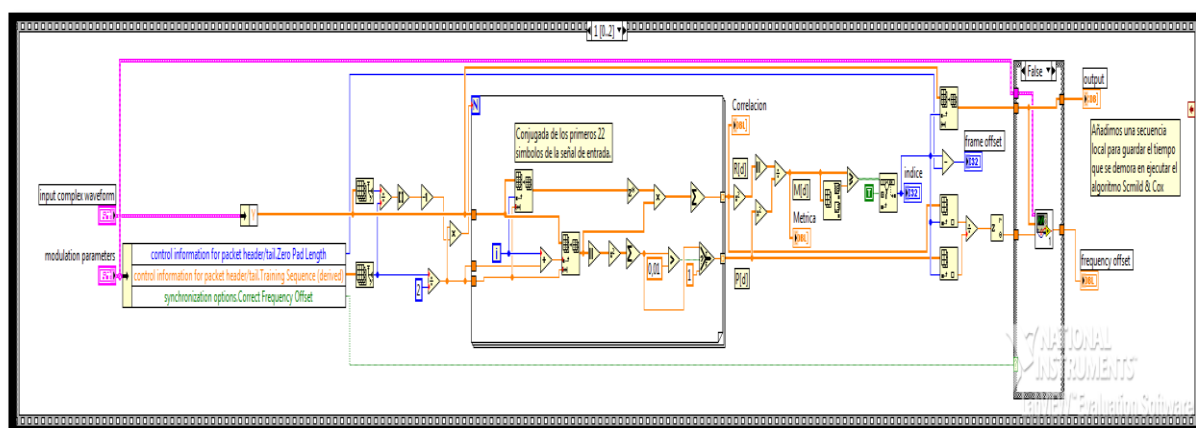


Fig. 4.114. Diagrama de bloques del Sub VI SAC

3.1.5 CORRECCIÓN DE FRECUENCIA Moose_ASC

Para corregir los problemas de rotación de la constelación que son provocados por el desplazamiento de frecuencia de portadora se implementó el siguiente VI utilizando ya la trama previamente sincronizada.

Panel Frontal del Instrumento Virtual: Los controles e indicadores utilizados para el Sub-VI de Corrección de Frecuencia ya se describieron en las figuras 3.6 y 3.42.

Diagrama de bloques del Instrumento Virtual: A continuación se describe las funciones utilizadas para crear el Sub-VI de Corrección de Frecuencia, algunas funciones ya fueron empleadas anteriormente como las que se muestra en las figuras 3.10, 3.15, 3.58, 3.60 y 3.61.

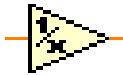
Programming>> Numeric>> Reciprocal
DIAGRAMA DE BLOQUES

Obtiene el inverso de la variable de entrada.

Fig. 4.115. Reciprocal

Diseño del VI para el esquema Moose_ASC:

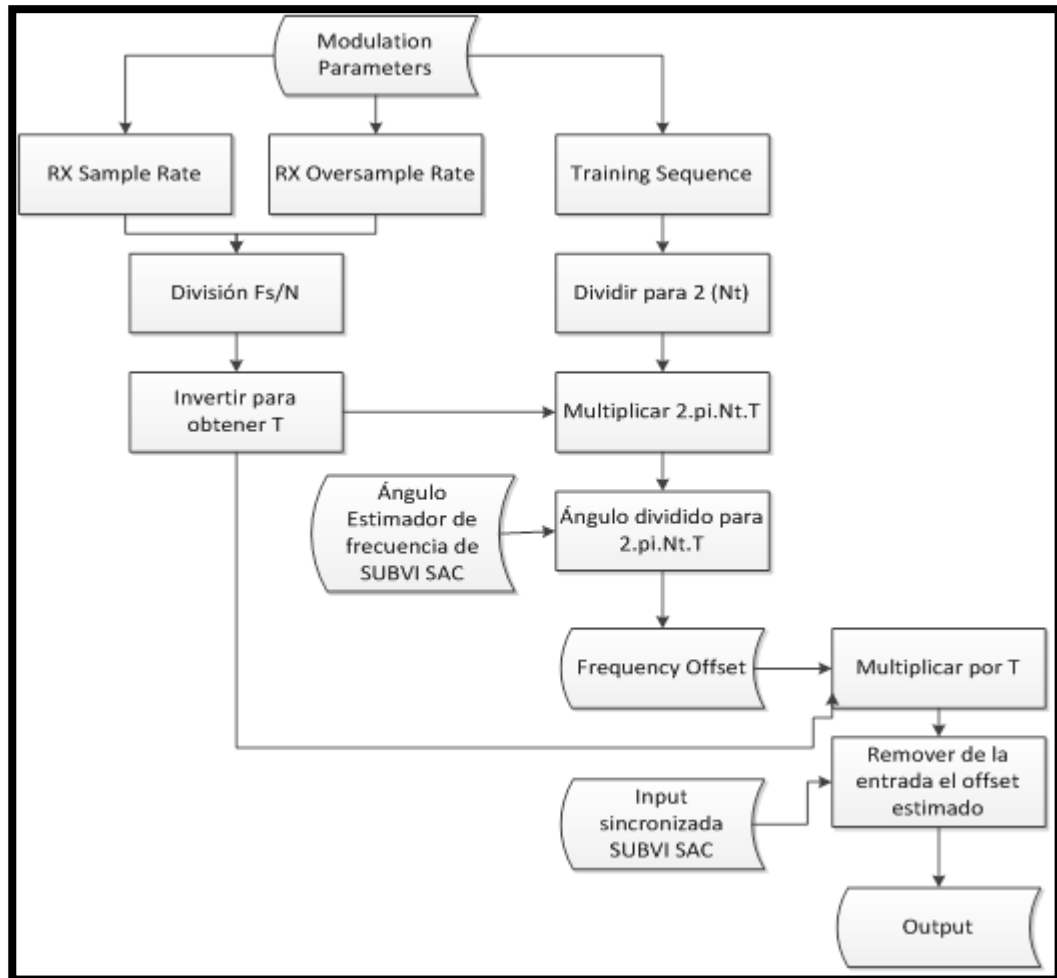


Fig. 4.116. Diagrama de flujo VI Moose_ASC

1. Del cluster **modulation parameters** desagregamos mediante **Unbundle By Name** para obtener RX Sample Rate, RX Oversample factor y secuencia de entrenamiento, del arreglo de la secuencia obtenemos su tamaño y lo dividimos para 2 y

finalmente para obtener f dividimos Sample Rate para Oversample Factor.

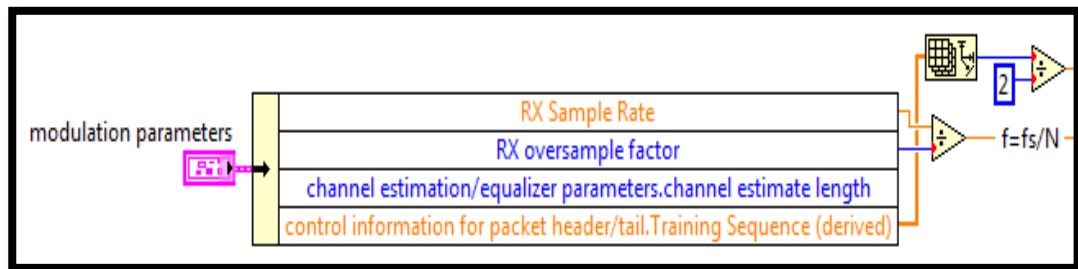


Fig. 4.117. Obteniendo f

2. El tamaño de la secuencia dividida para 2 la llamaremos N_t , para obtener T invertimos f mediante **Reciprocal** y utilizando **Compound Arithmetic** clic derecho a la función **Change Mode** >> **Multiply** estiramos dicha función y multiplicamos $2 \cdot \pi \cdot N_t \cdot T$, la constante π la extraemos de la paleta **Numeric** >> **Math Constant**.

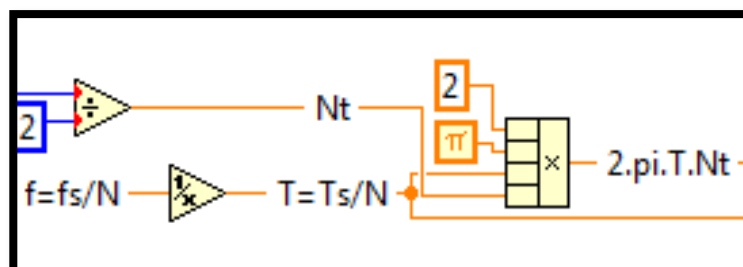


Fig. 4.118. Obtener parámetro $2\pi TN_t$

3. El ángulo estimado con el algoritmo SAC lo recibe el VI de corrección de frecuencia lo dividimos para $2\pi N_t T$ y obtenemos la frecuencia de desplazamiento. Al multiplicar esta frecuencia con T obtenemos $E^\wedge$.

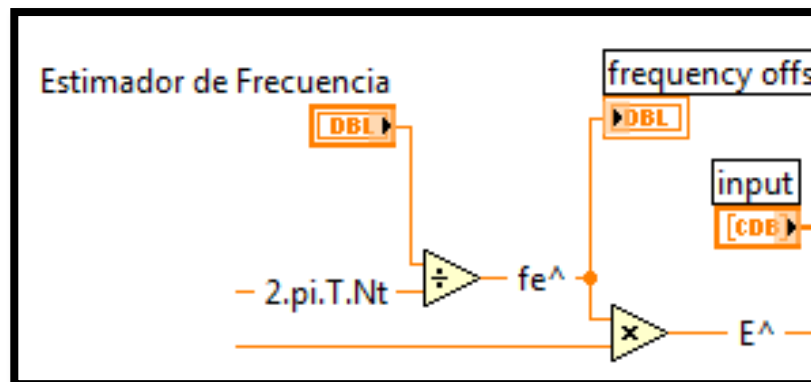


Fig. 4.119. Obteniendo Frequency Offset y $E^\wedge$

4. De la trama ya previamente sincronizada utilizando SAC se obtiene el tamaño que será el número de iteraciones N del lazo for, se remueve el offset para lo cual cada valor del arreglo trama sincronizada se multiplica por el desfase que se obtiene multiplicando la variable i por $E^\wedge$ negado **Negate** luego a este resultado se lo multiplica por la constante 2π y por último se convierte de polar a complejo con magnitud 1 **Polar To Complex**.

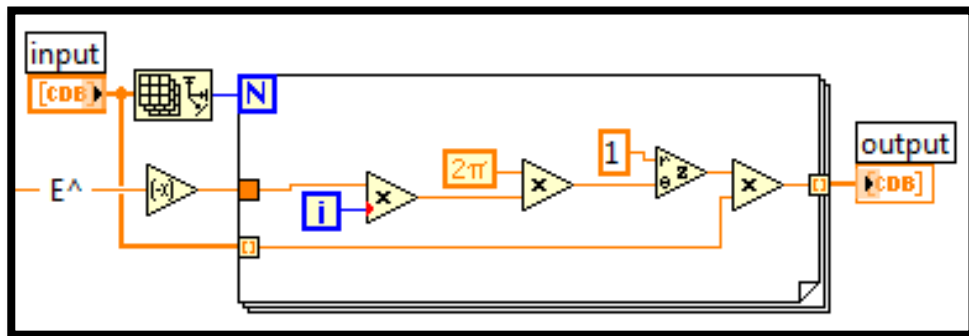


Fig. 4.120. Trama corregida frecuencia de desplazamiento

Diagrama de bloques del VI Moose_ASC.

La figura 3.121 muestra el diagrama de bloques correspondiente al corrector de frecuencia de desplazamiento este VI se lo guarda como un Sub VI para insertarlo en el Sub VI **SAC**.

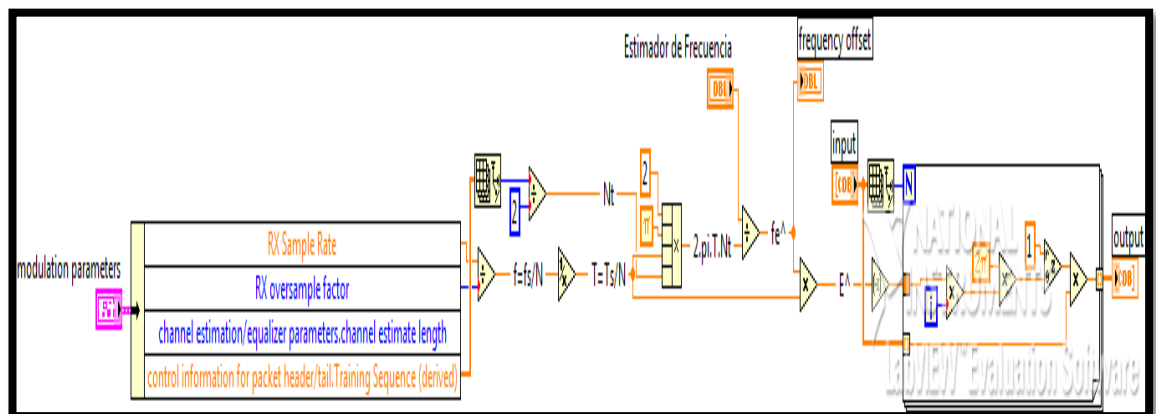


Fig. 4.121. Diagrama de bloques del Sub VI Moose_ASC

CAPÍTULO 4

ANÁLISIS Y RESULTADOS

Se analiza mediante graficas el comportamiento de la sincronización de trama, comparando métodos y verificando que cumpla con su objetivo que es el de ubicar el inicio de la trama y proceder a la eliminación de los símbolos posteriores al punto encontrado y además realizando la respectiva corrección de frecuencia de portadora.

Se observara imágenes de la constelación antes y después del bloque sincronizador de trama. También las gráficas de correlación cuando se cambia la secuencia de entrenamiento, además de las respectivas gráficas de SNR vs. BER para determinar qué sistema es más robusto frente a los cambios de un ambiente de propagación emulados.

Finalmente se realizara el análisis de Corrección de Frecuencia encontrando los puntos de ruptura tanto para el algoritmo Moose propio de la práctica como para el Moose utilizando SAC.

4.1 FUNCIONAMIENTO DEL SINCRONIZADOR SCHMILD AND COX.

Las gráficas de constelaciones presentadas nos muestran el efecto de la sincronización de trama, el nuevo algoritmo Schmild and Cox cumplió con las misma expectativa que el algoritmo Sliding Correlator.

La figura 4.1 muestra la constelación y transiciones de una trama modulada y enviada a través de los equipos USRP. Esta imagen está compuesta de zero pad, la trama de entrenamiento, los datos y nuevamente el zero pad tal como se describe en la figura 2.8.

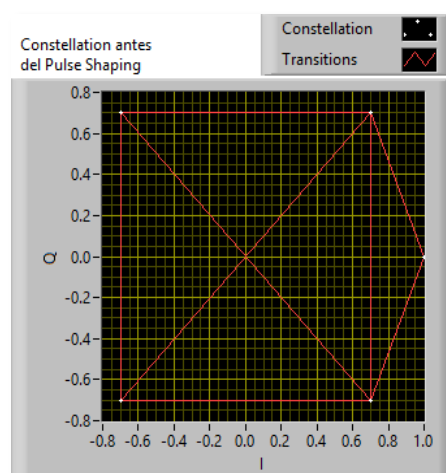


Fig. 5.1. Constelación Modulada en QPSK

Los puntos fuera de la constelación corresponden al zero pad, ya que estos están modulados en BPSK y corresponden a símbolos ceros.

La sincronización elimina los primeros símbolos ubicados antes de la secuencia de entrenamiento. Tal y como se muestra en la figura 4.2.

La eliminación se refleja como la desaparición de transiciones en la constelación así como los puntos que representan al zero pad. La pequeña dispersión que se muestra, entre los puntos encerrados, es debida al desplazamiento que presenta el enlace de transición entre los equipos, el cual es corregido por el ecualizador (caso que no se estudia en esta tesina).

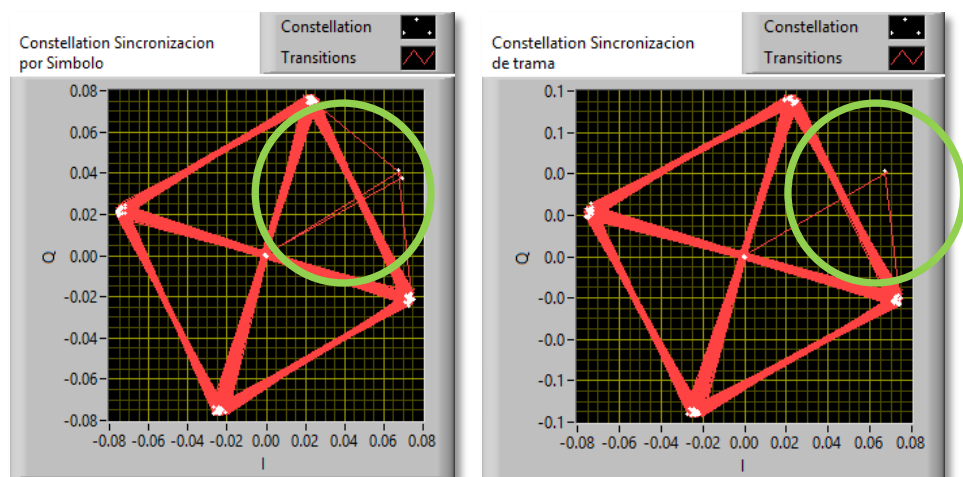


Fig. 5.2. Constelación antes y después de la Sincronización de Trama SC

Sin este efecto del sincronizador el ecualizador y demás bloques siguientes del sistema no trabajan correctamente.

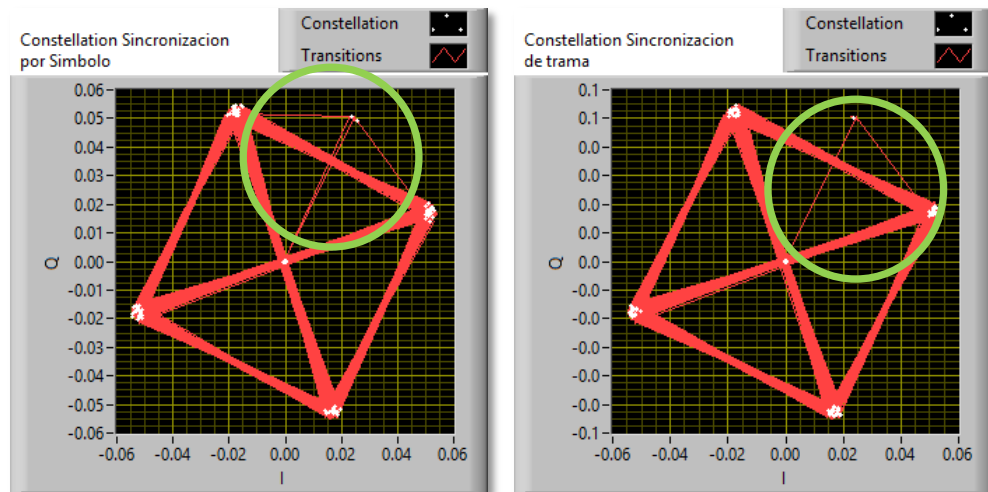


Fig. 5.3. Constelación antes y después de la Sincronización de Trama SAC

La figura 4.3 muestra el funcionamiento del algoritmo Schmid and Cox. Se aprecia como elimina los símbolos del zero pad. Lo que nos muestra que el algoritmo es efectivo en la búsqueda. Esto se puede corroborar con el punto el índice inicio de trama que ambos encuentra como solución:



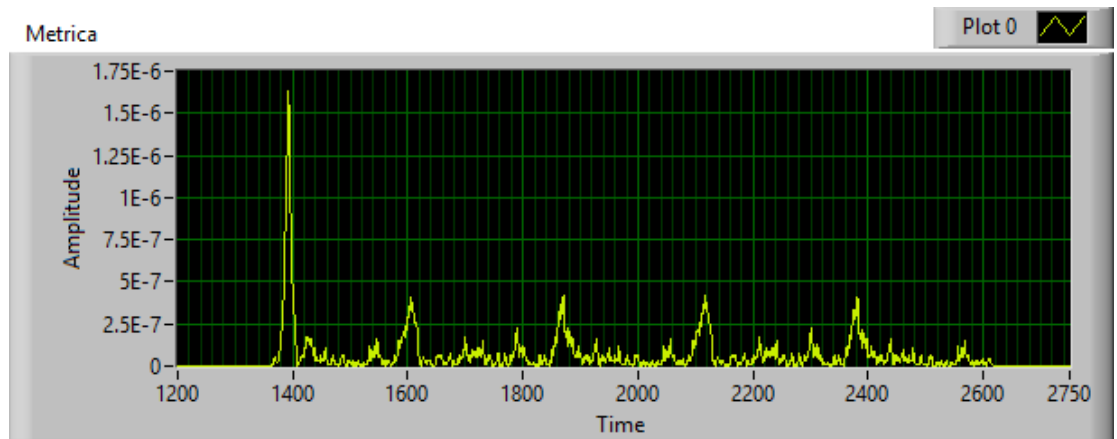
Fig. 5.4. Inicio de Trama de SC y SAC respectivamente

El hecho que encuentre el mismo punto corrobora a que el nuevo método fue correctamente adaptado. Esto también se puede verificar con el BER que muestra el sistema en el receptor:

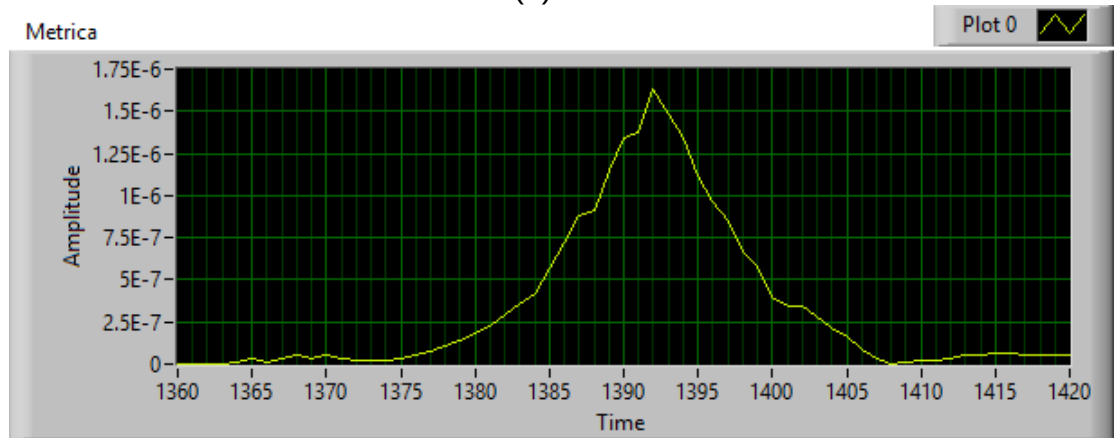


Fig. 5.5. BER de los algoritmos SC y SAC respectivamente

Estos resultados son obtenidos sin utilizar un canal implementado solo transmisión real. Los efectos de la constelación desplazada son producidos por el canal.



(a)



(b)

Fig. 4.6 Métrica de correlación a) Grafica Completa de la Métrica b) Pico de la Métrica.

Inicio de la trama

LA figura 6 muestra la forma que toma la métrica $M[d]$, clara mente se puede ver el pico que muestra el máximo valor de la correlación, lugar que indica el inicio de la trama de datos. Muchas veces el algoritmo presenta más de un pico y ubicados seguidamente del anterior. Lo que provoca ligera alteraciones del sistema.

4.2 COMPORTAMIENTO DEL SISTEMA FRENTE DIFERENTES CANALES.

Se presenta el comportamiento del sistema ante la presencia de los diferentes canales que se utilizan para la realización de este proyecto, utilizando la configuración del transmisor y receptor por defecto dada.

Se efectúa dos análisis, uno en simulación y el otro en transmisión real usando los USRP. Debido a que en simulación ciertos parámetros no son tomados en cuenta (como ganancia de la antena y frecuencia portadora) se espera observar diferencias significativas en ambas situaciones.

A los canales se les aplica la señal antes de ser transmitidos (tanto para la simulación como para la transmisión con los USRP), por lo que se consideran CANALES EMULADOS.

Los canales emulados son AWGN, ISI y RAYLEIGH. En donde los dos primeros son los que vienen incluidos en el manual y el Rayleigh es el desarrollado por nosotros. Los parámetros del canal ISI son basados en la estructura del laboratorio de simulación descrito en el capítulo 2.

Para esta graficas se trabajó con los valores por defectos de la práctica y las variables a manipular, que son:

Sample rate TX y Sample rate RX: 4MHZ

Oversample Factor: 20

Noise power: desde -60 dB a 0 dB

Modulation: QPSK, 16 QAM, 64 QAM

En las gráficas solo se muestra el SNR desde 0 a 10 ya que estos son los rangos que se utilizaran para comparar con las gráficas basadas en el documento [22].

4.2.1 CANAL AWGN

En Las siguientes gráficas muestran el funcionamiento del canal AWGN que viene implementado en el sistema en que se trabaja. Este análisis nos permite observar cómo afecta el canal y el comportamiento del sistema ante las diferentes modulaciones y potencias de ruido deseados.

Las gráficas se basan en la siguiente figura:

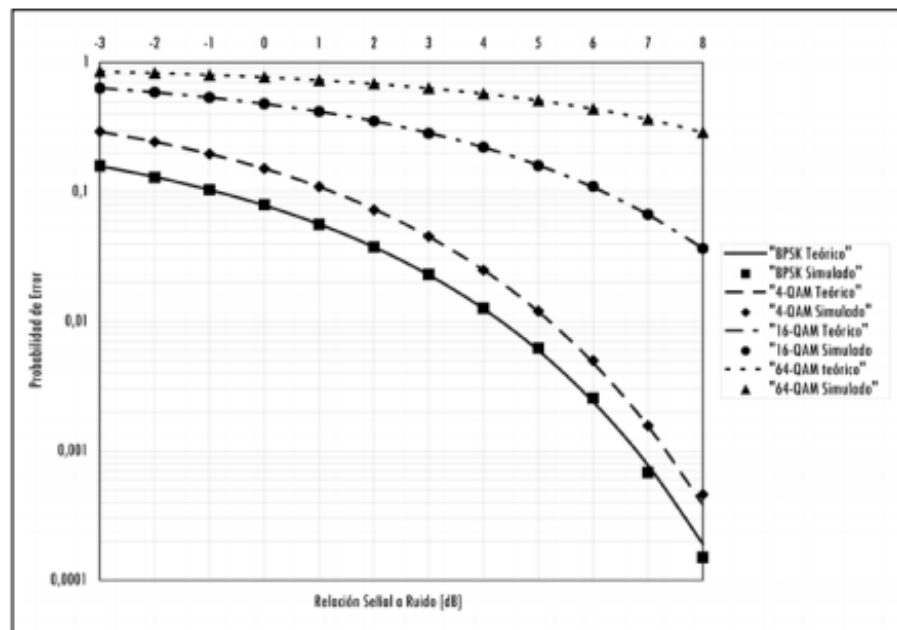


Fig. 5.7. SNR vs. BER de un canal AWGN [22]

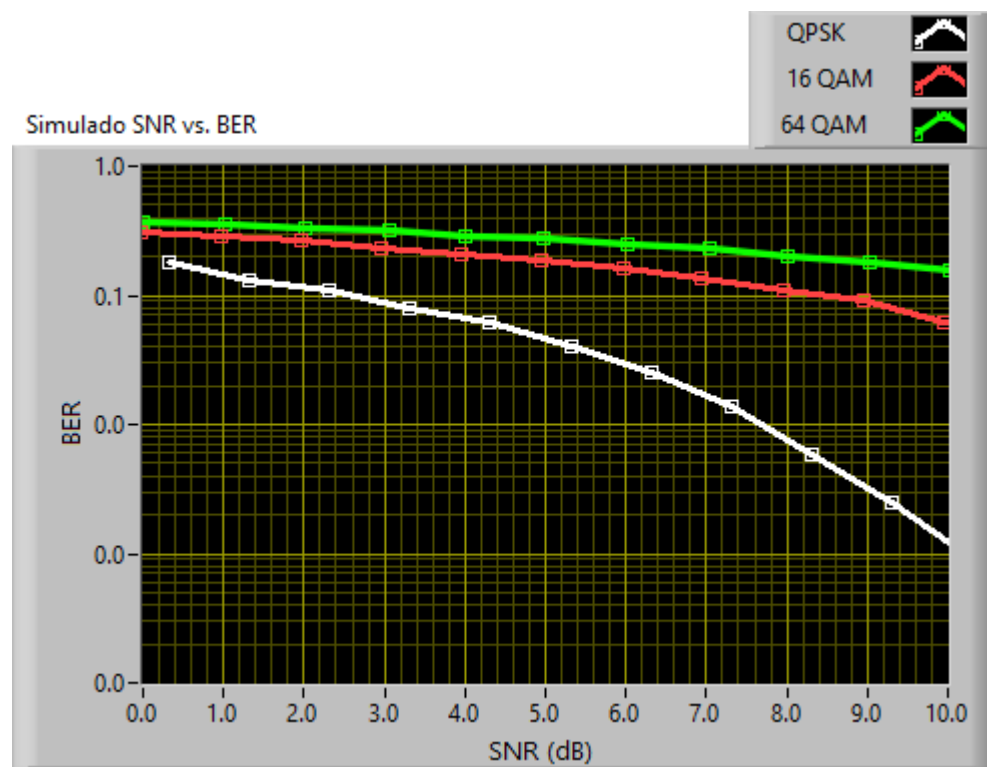


Fig. 5.8. SNR vs. BER de un canal AWGN Simulación

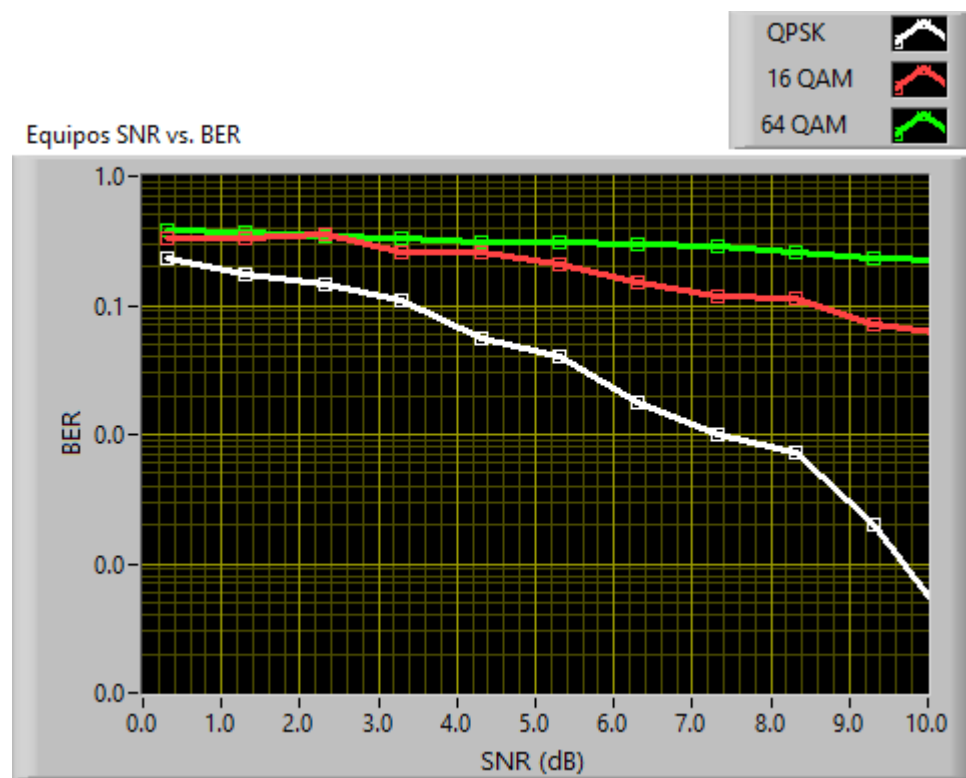


Fig. 5.9. SNR vs. BER de un canal AWGN Transmisión Real

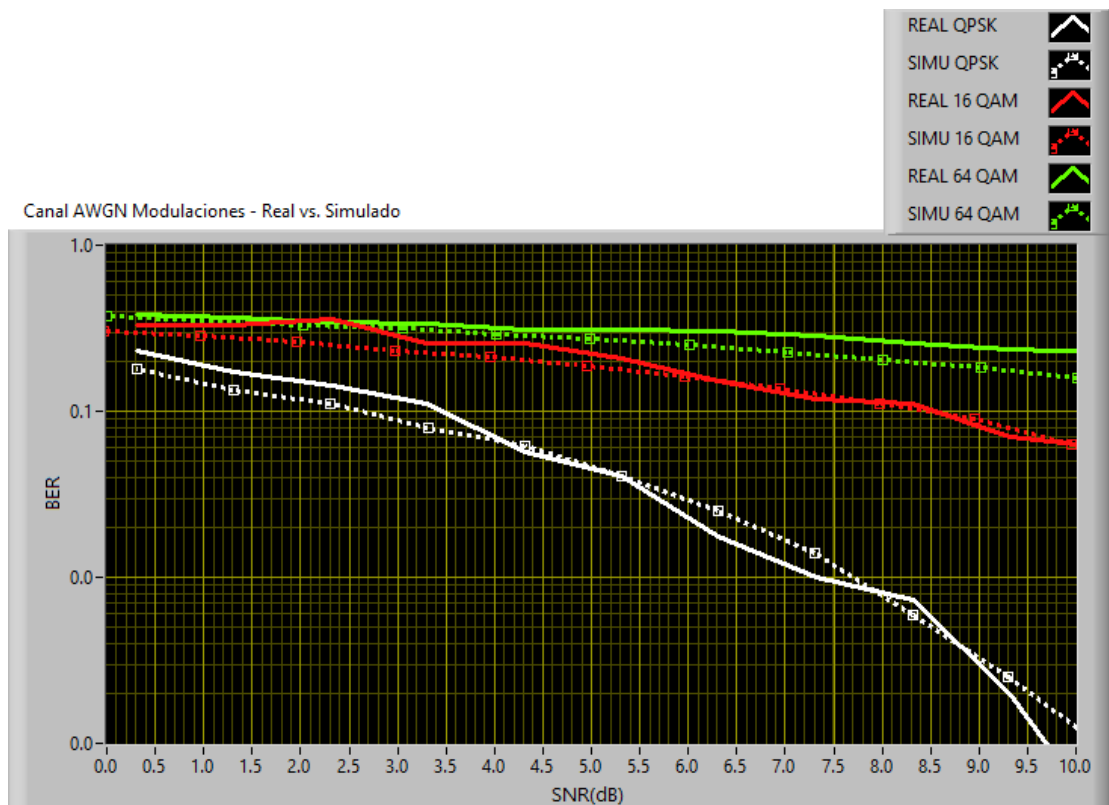


Fig. 5.10. SNR vs. BER de un canal AWGN Transmisión Real y Simulación

La figura 4.10 muestra las diferencias entre las modulaciones aplicadas tanto para el caso de simulación (líneas punteadas) y el caso de transmisión real (líneas continuas).

La diferencia se debe a que en simulación no se toma en consideración las ganancias de las antenas del transmisor ni de los impedimentos producidos por la distancia entre los equipos siendo la potencia de la señal transmitida la misma que la señal recibida.

Para QPSK:

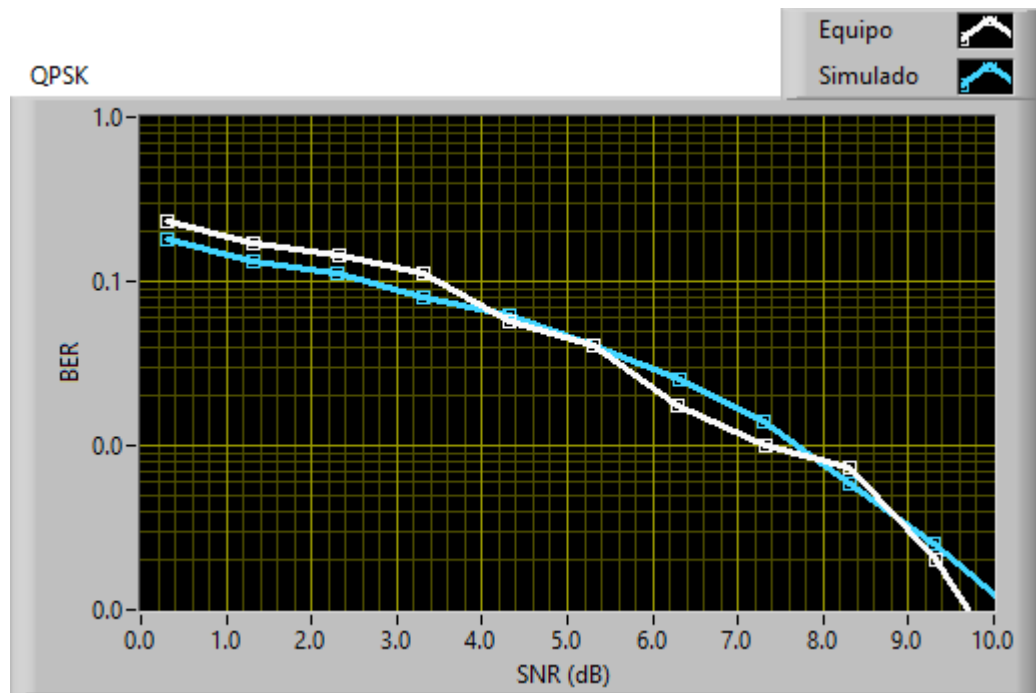


Fig. 5.11. SNR vs. BER Canal AWGN Modulación QPSK Transmisión Real y Simulación

Para 16 - QAM:

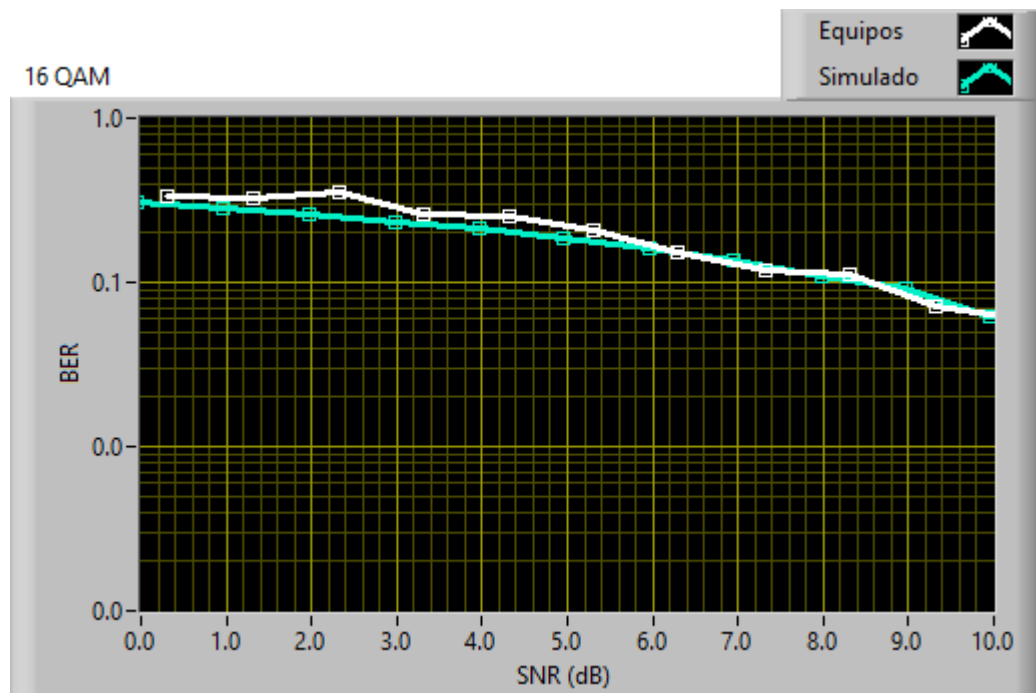


Fig. 5.62. SNR vs. BER Canal AWGN Modulación 16 - QAM Transmisión Real y Simulación

Para 64 - QAM:

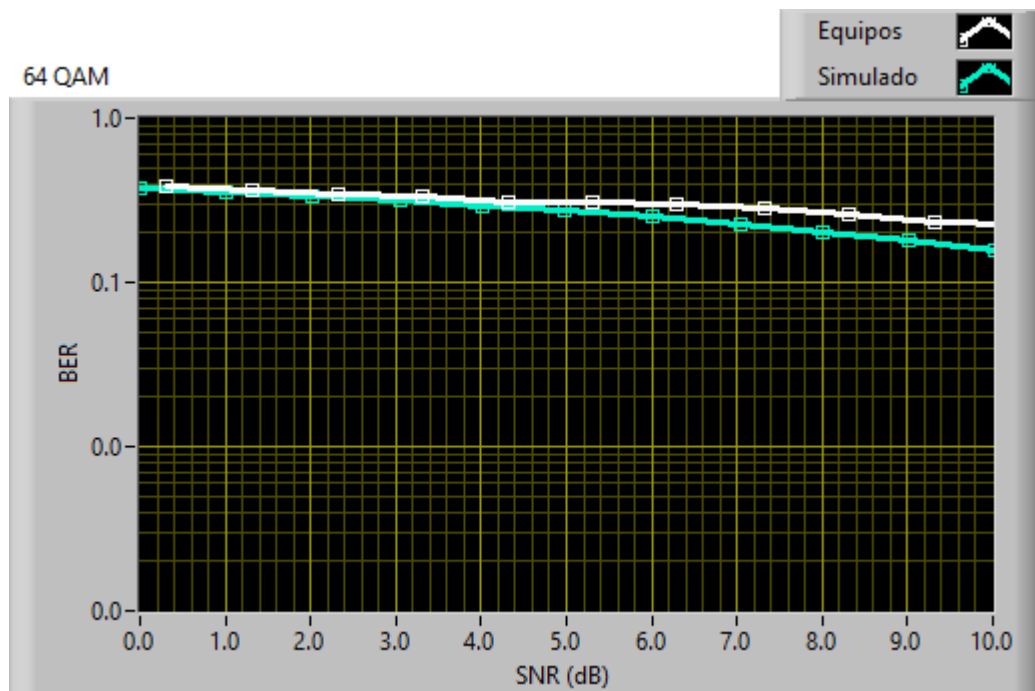


Fig. 5.73. SNR vs. BER Canal AWGN Modulación 64 - QAM Transmisión Real y Simulación

4.2.2 CANAL ISI

El canal ISI es el desarrollado en el capítulo 2 basado en las dimensiones del laboratorio de simulaciones de telecomunicaciones utilizando la técnicas de tres rayos, el sistema mostro el siguiente comportamiento en simulación y transmisión real.

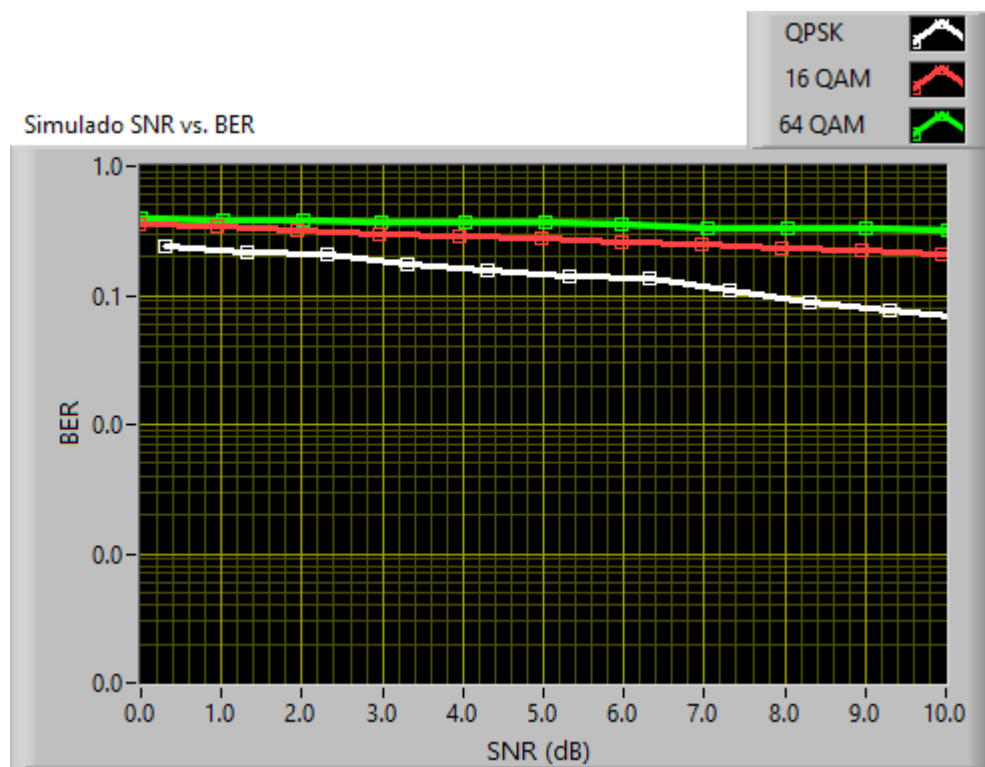


Fig. 5.84. SNR vs. BER de un canal ISI Simulación

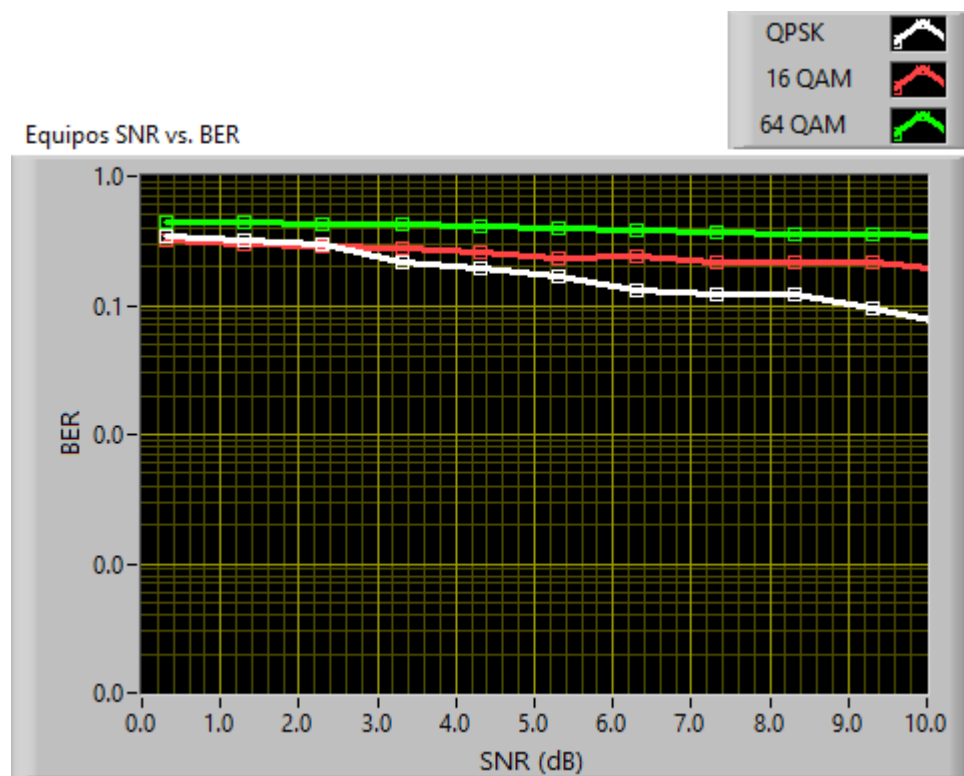


Fig. 5.15. SNR vs. BER de un canal ISI Transmisión Real

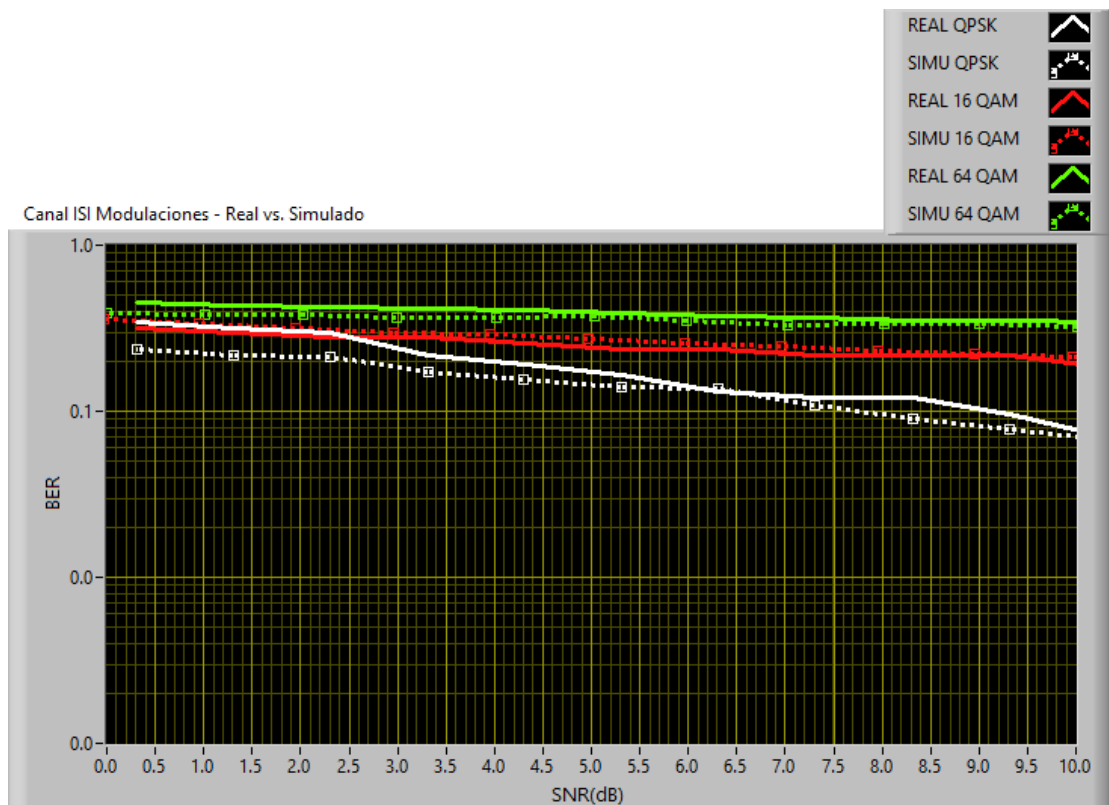


Fig. 5.16. SNR vs. BER de un canal AWGN Transmisión Real y Simulación

Como el vector es diseñado basado en el laboratorio presenta el comportamiento que se muestra en las gráficas 4.14, 4.15 y 4.16. Esta última muestra la diferencia entre las modulaciones, igualmente QPSK, 16-QAM, 64-QAM respectivamente. En este caso la separación pareciera que nos es tan visible pero si se fijan puntos se puede ver que existe diferencia de aproximadamente 4 dB entre las líneas de QPSK.

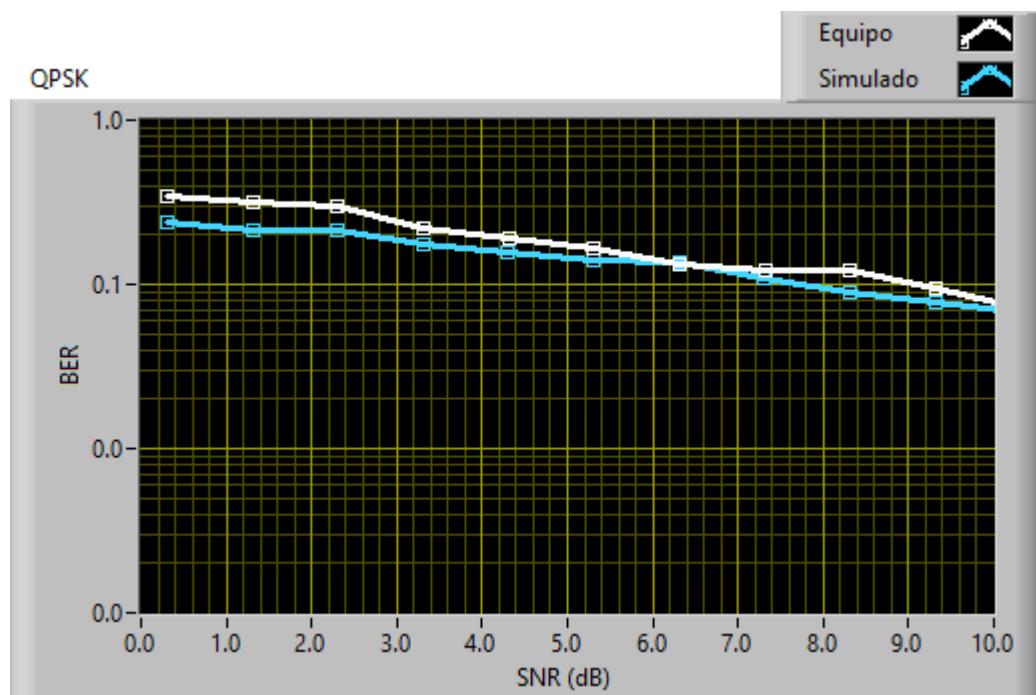


Fig. 5.97. SNR vs. BER Canal ISI Modulación QPSK Transmisión Real y Simulación

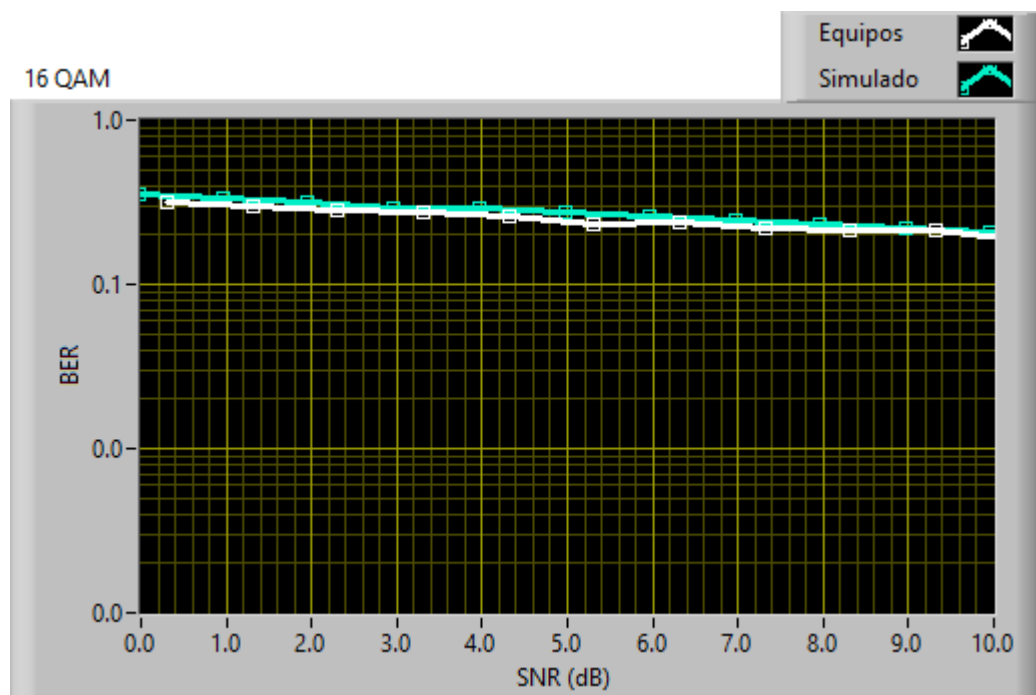


Fig. 5.1810. SNR vs. BER Canal ISI Modulación 16 - QAM Transmisión Real y Simulación

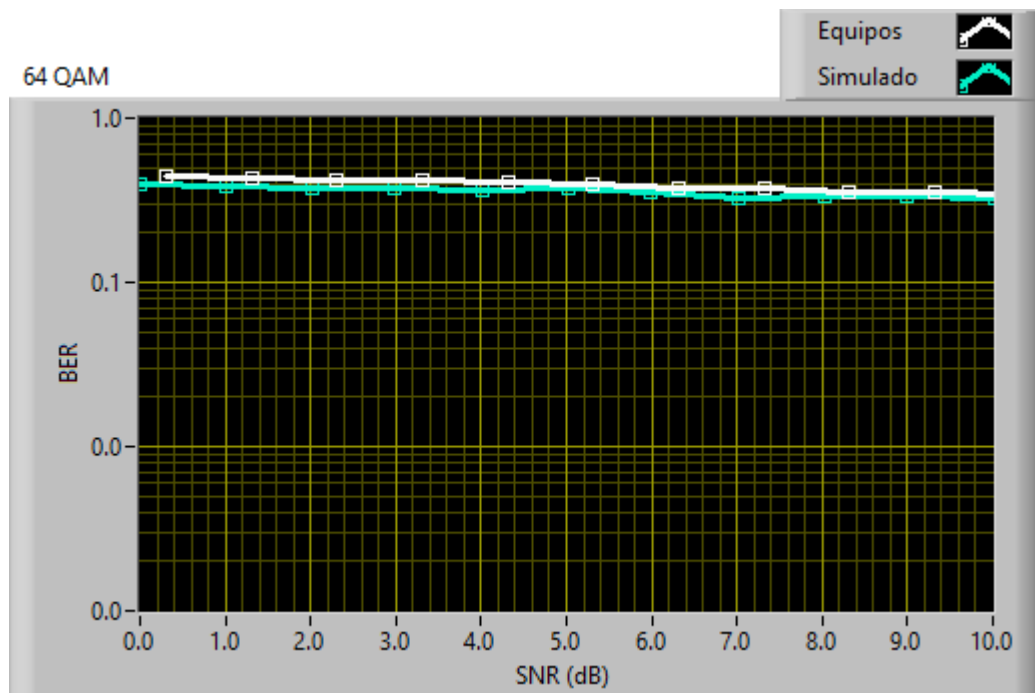


Fig. 5.119. SNR vs. BER Canal ISI Modulación 64 - QAM Transmisión Real y Simulación

Para este caso el canal ISI de tres vectores afecta a la señal, generando que estas tengan valores similares, sobretodo en el caso de 16-QAM, figura 4.18, otro caso especial es el hecho de la figura 4.19 en donde la simulación está más baja que la real.

4.2.3 CANAL RAYLEIGH

Este canal fue implementado tal y como se redactó en el capítulo 3 por lo que es diferente al canal ISI y AWGN que viene en el simulador, por lo que se espera un comportamiento variado e inesperado del mismo.

Las gráficas se basan en la siguiente figura:

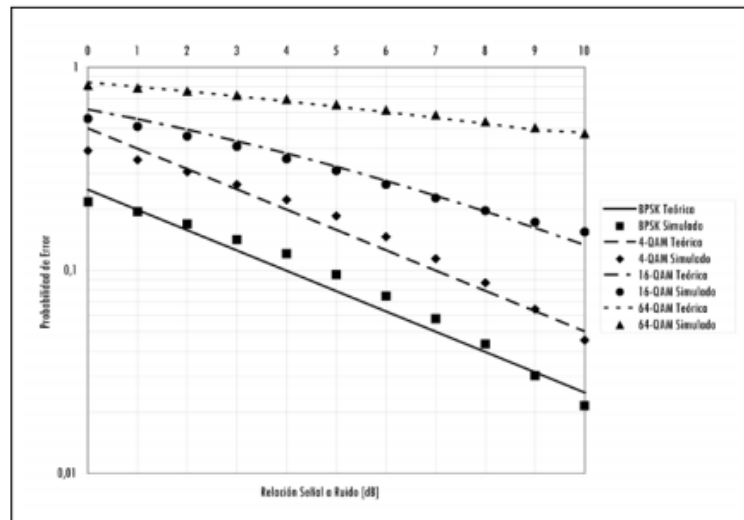


Fig. 5.20. SNR vs. BER de un canal RAYLEIGH [22]

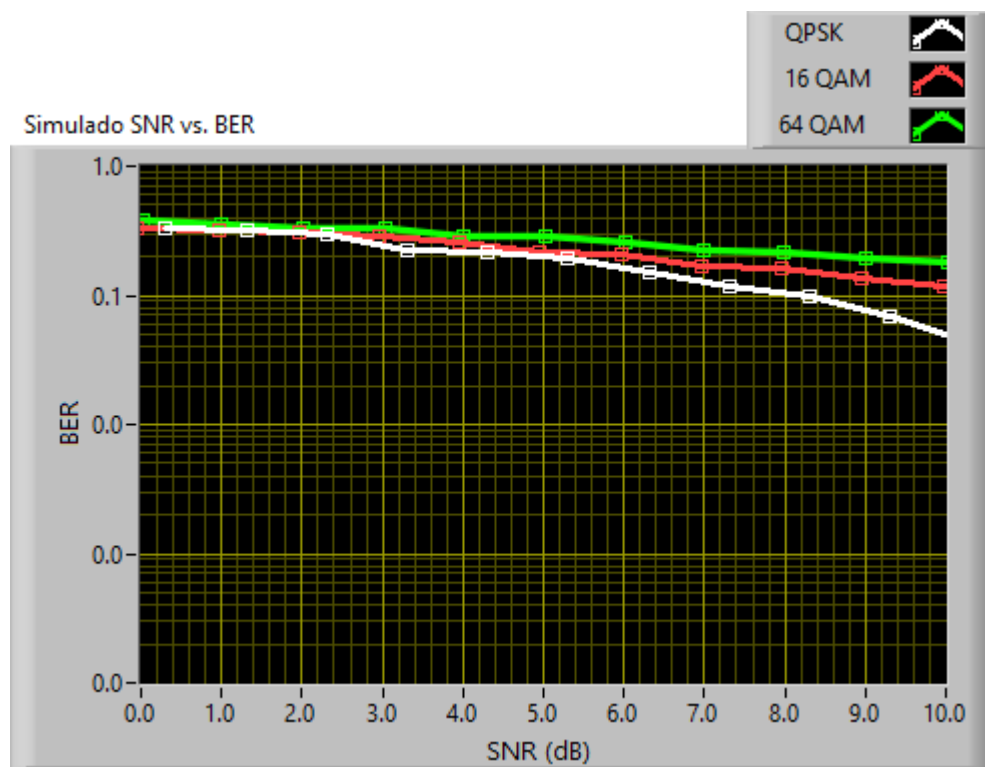


Fig. 5.121. SNR vs. BER de un canal RAYLEIGH Simulación

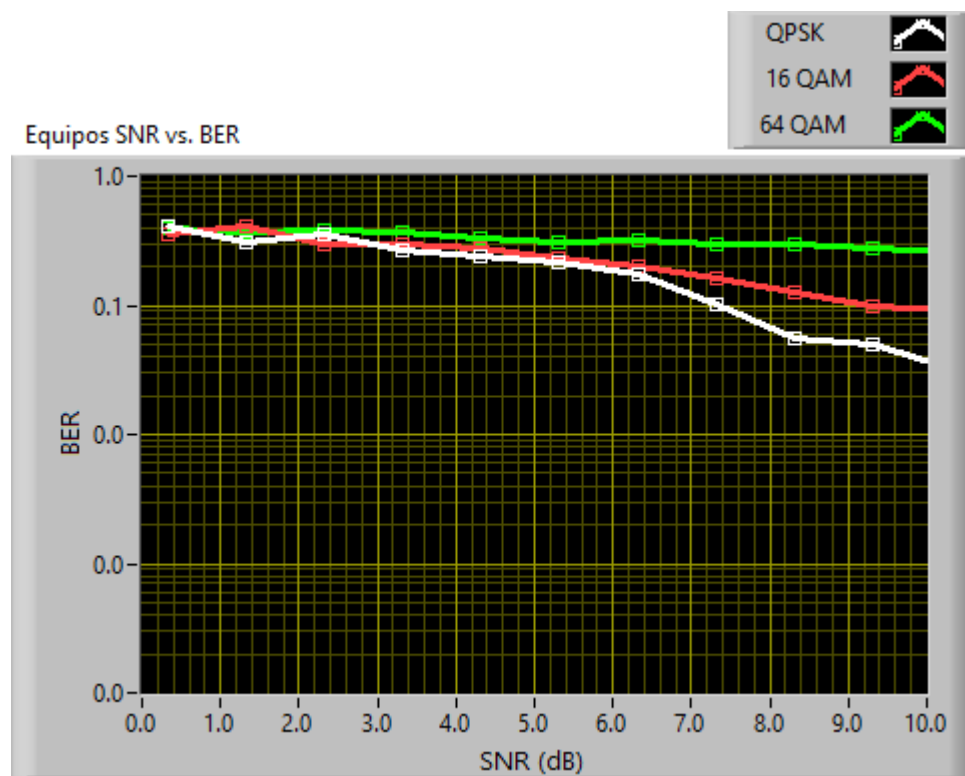


Fig. 5.132. SNR vs. BER de un canal RAYLEIGH Transmisión Real

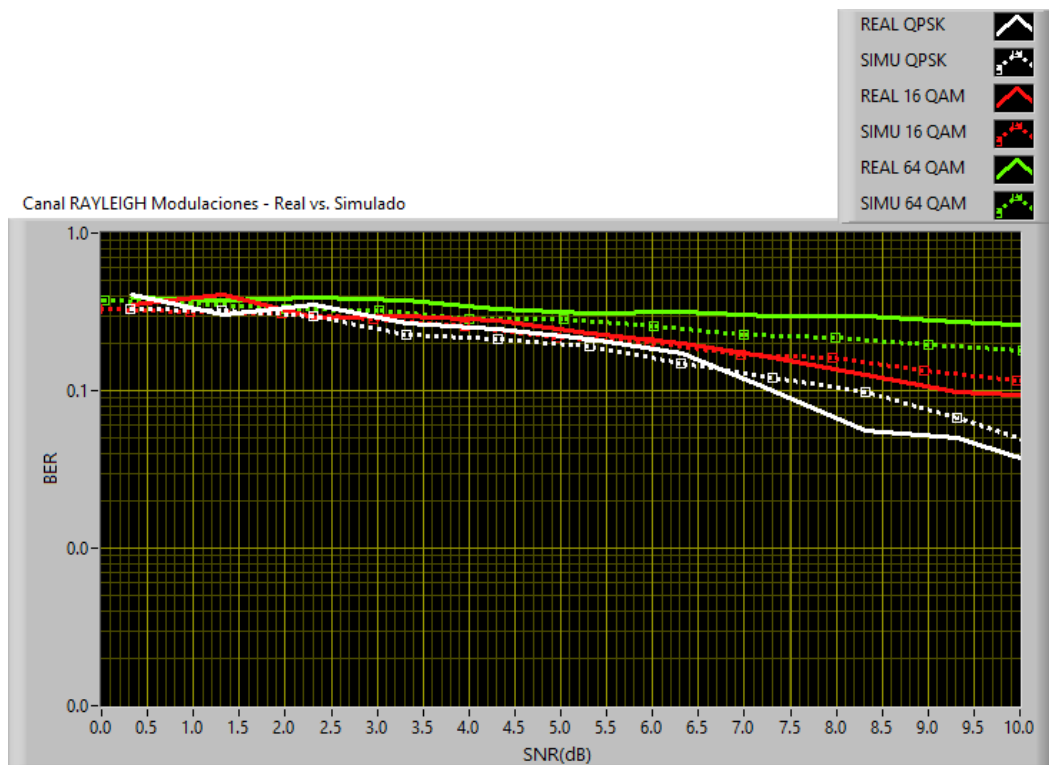


Fig. 5.143. SNR vs. BER de un canal RAYLEIGH Transmisión Real y Simulación

En las gráficas se observa cómo se comporta el sistema ante el desvanecimiento RAYLEIGH. Se puede apreciar que en simulación el sistema tiene niveles de BER más elevado que en transmisión real. Dichos niveles se los puede apreciar mejor en la gráfica 4.23.

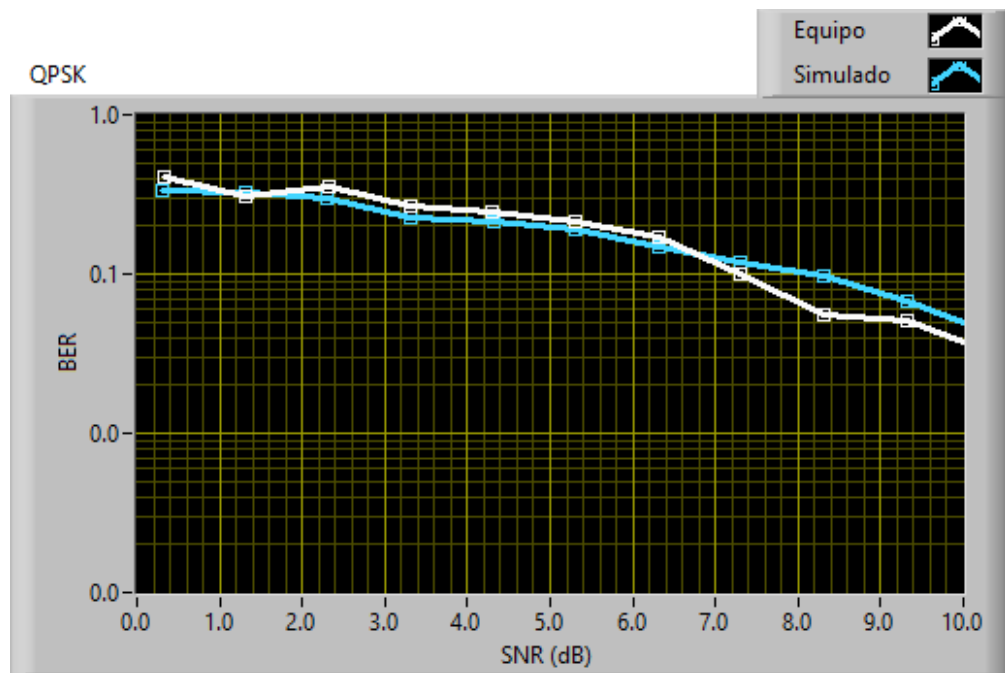


Fig. 5.24. SNR vs. BER Canal RAYLEIGH Modulación QPSK Transmisión Real y Simulación

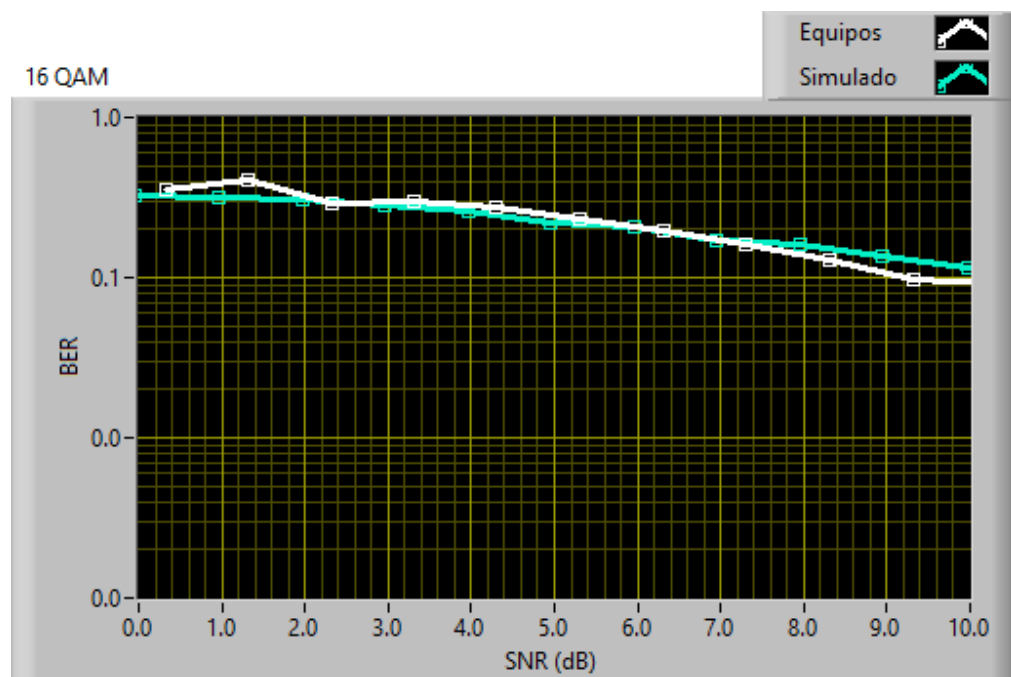


Fig. 5.25. SNR vs. BER Canal RAYLEIGH Modulación 16 - QAM Transmisión Real y Simulación

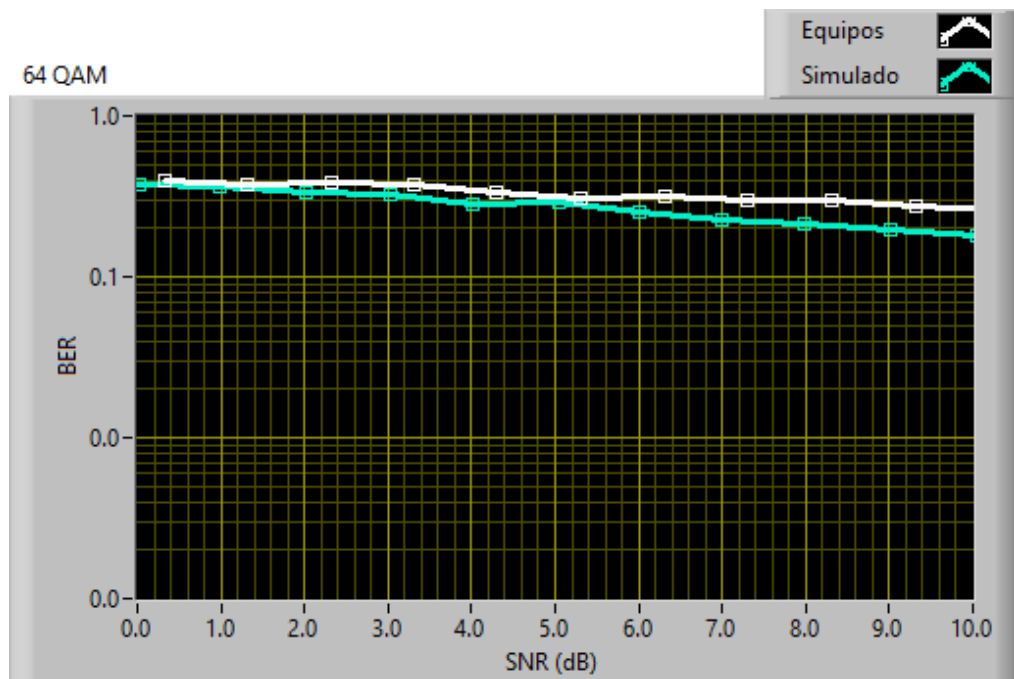


Fig. 5.26. SNR vs. BER Canal RAYLEIGH Modulación 64 - QAM Transmisión Real y Simulación

Las gráficas muestra el comportamiento del sistema ante las diferente modulaciones en simulación y en transmisión en la presencia de RAYLEIGH, recordemos que este canal genera interferencia intersimbólica desvanecimientos a la vez, por lo que ante mayor sea la modulación tanto en real como en simulación el sistema se afecta más, en el caso de 64-QAM es mayor en equipo que en simulación.

Se ve una clara diferencia en el comportamiento del sistema entre los datos obtenidos durante la simulación y las pruebas con los USRP. Esta diferencia se debe a que durante la simulación no se toma en consideración la ganancia de las antenas ni la influencia de la frecuencia portadora.

Una de las sorpresas es que en cada canal cuando se trabaja con la modulación 64-QAM el sistema presenta mejores valores de BER que en transmisión real. Lo que nos indica que el sistema es más sensible respecto al incremento de tasa de bits, también que durante la transmisión, esta modulación, es más propensa a perturbaciones que QPSK o 16-QAM.

Para el caso de AWGN y RAYLEIGH se puede ver una gran similitud con las gráficas de referencia, en el caso del canal ISI este no tiene referencia debido a que está basado a las dimensiones y asunciones que se tomaron respecto al laboratorio de simulación de telecomunicaciones, detallados en el capítulo 2. Por lo que sus curvas son bases y referencias para este proyecto planteado.

4.3 COMPORTAMIENTO DE LOS ALGORITMOS DE SINCRONIZACIÓN FRENTE DIFERENTES CANALES Y SECUENCIAS DE ENTRENAMIENTO.

Mostramos como los algoritmos funcionan ante la presencia de los canales tanto en simulación y transmisión real. Se toma como base al algoritmo Sliding Correlator de la práctica y se lo compara con el algoritmo implementado y adaptado de Schmild and Cox. También veremos cómo estos algoritmos son influenciados al cambiar la secuencia de entrenamiento.

Se toma como referencia el comportamiento del algoritmo SC para todos los casos ya que este es el algoritmo implementado por la práctica.

4.3.1 CANAL AWGN

Aquí vemos las gráficas comparativas de los dos algoritmos, el Sliding Correlator (algoritmo de referencia) y Schmild and Cox (algoritmo implementado). Las gráficas son de SNR vs. BER, cada una por cada modulación utilizada anteriormente.

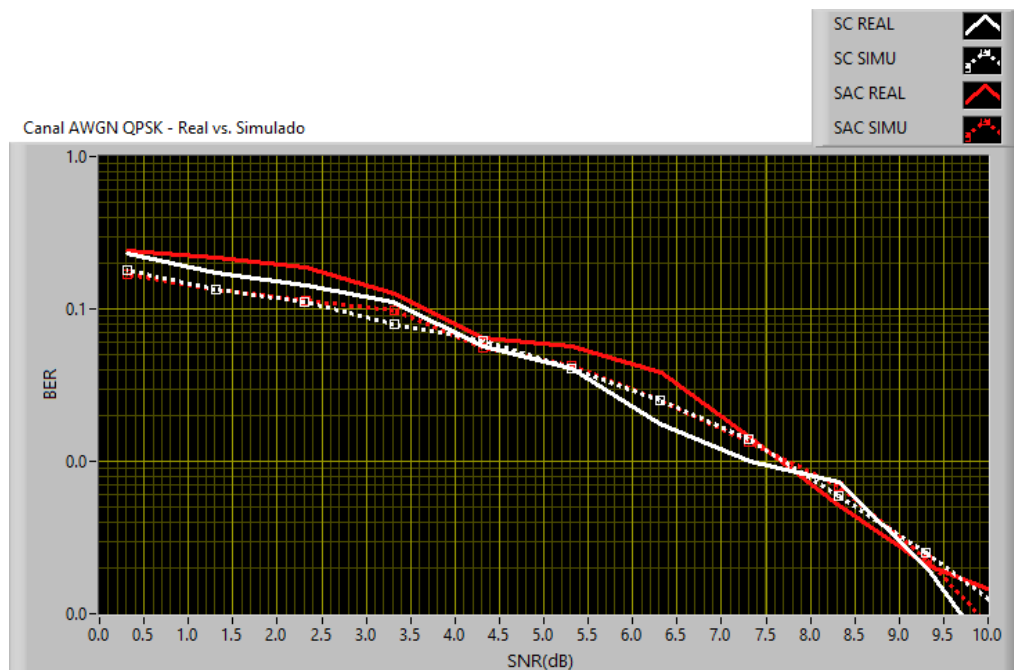


Fig. 5.27. SNR vs. BER QPSK canal AWGN Simulación- Transmisión Real SC y SAC
Secuencia Barker

Al igual que la sección anterior se puede ver claramente la diferencia que existe entre la simulación y la transmisión real. También es claro que el algoritmo SAC tiene un mejor comportamiento que el SC en transmisión real (línea más suavizada).

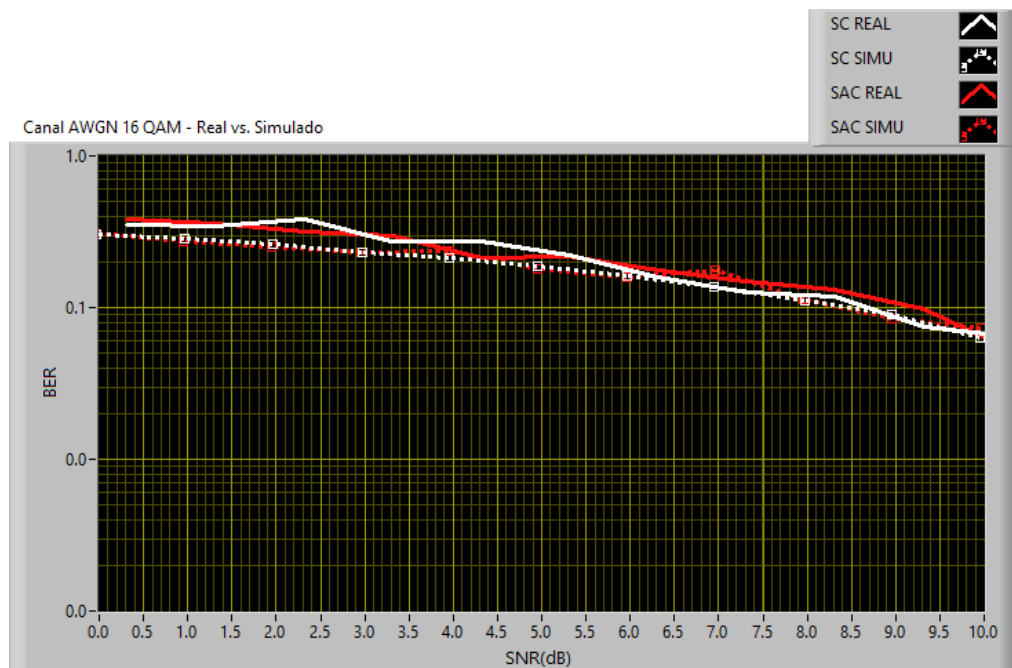


Fig. 5.28. SNR vs. BER 16 –QAM canal AWGN Simulación- Transmisión Real SC y SAC
Secuencia Barker

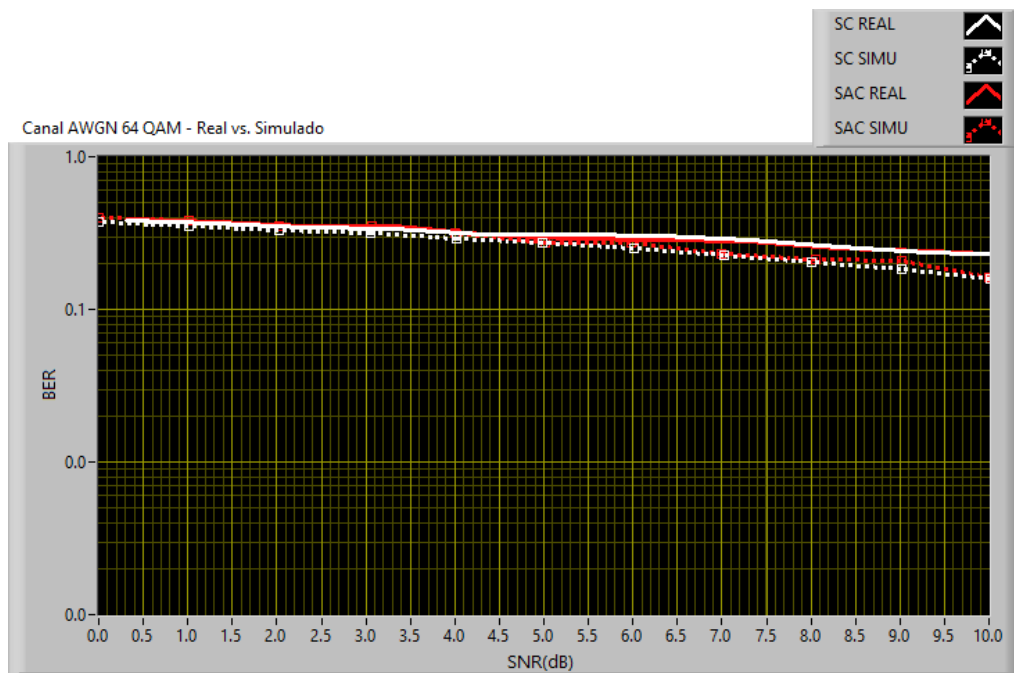


Fig. 5.29. SNR vs. BER 64 - QAM canal AWGN Simulación- Transmisión Real SC y SAC
Secuencia Barker

Se puede observar entre las figuras como el algoritmo SAC con secuencia Barker presenta mejores valores de BER que SC. Aunque en el caso de 16 QAM y 64 QAM existe una gran similitud en el comportamiento y valores obtenidos.

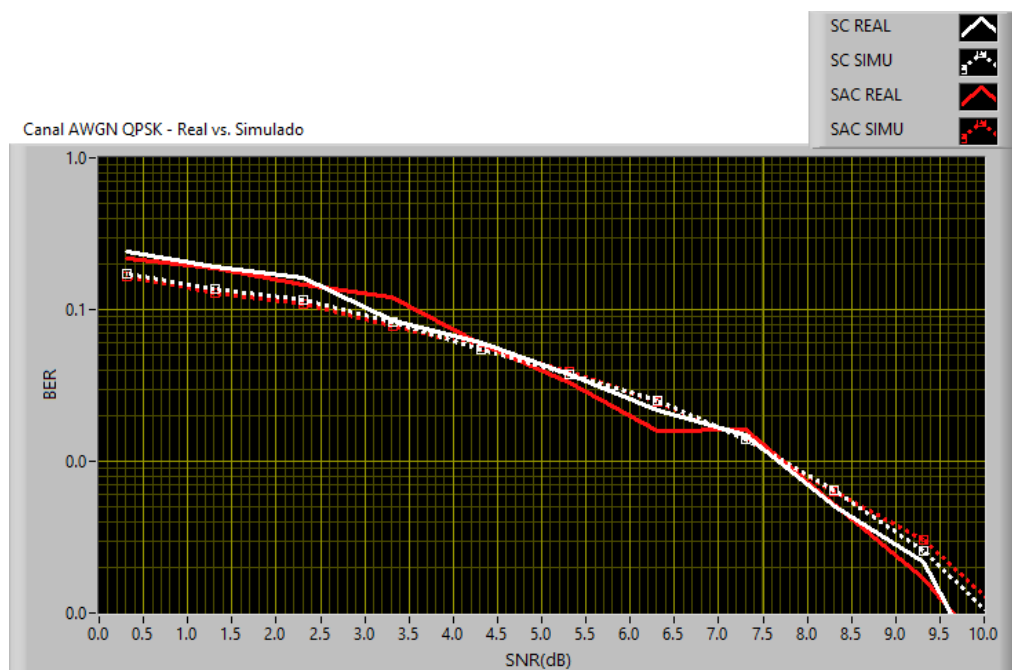


Fig. 5.30. SNR vs. BER QPSK canal AWGN Simulación- Transmisión Real SC y SAC

Secuencia Willard

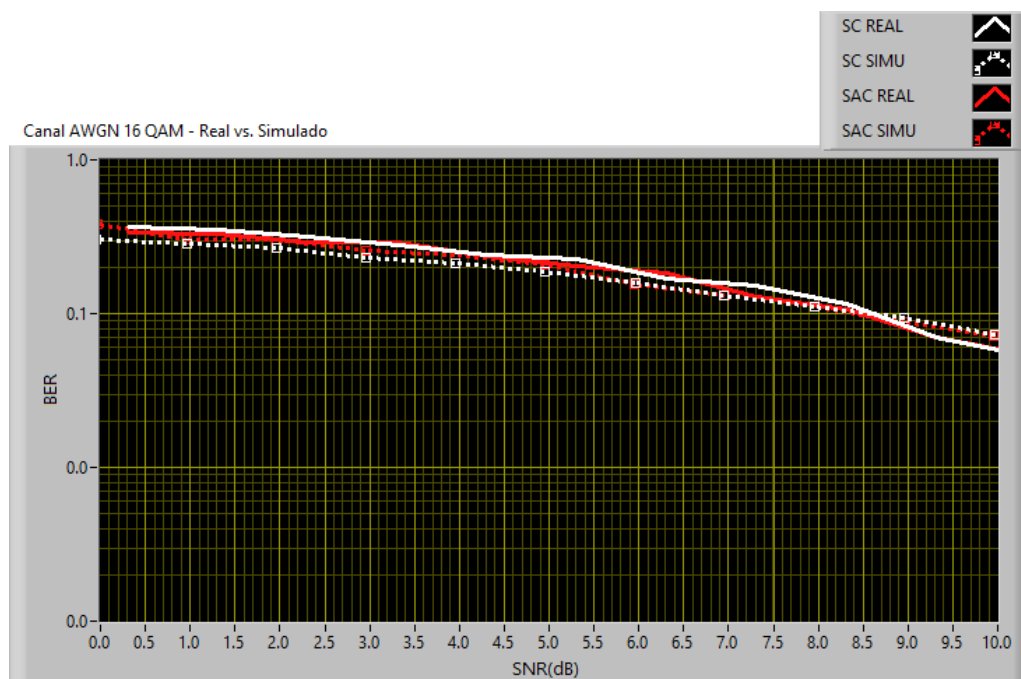


Fig. 5.31. SNR vs. BER 16 –QAM canal AWGN Simulación- Transmisión Real SC y SAC

Secuencia Willard

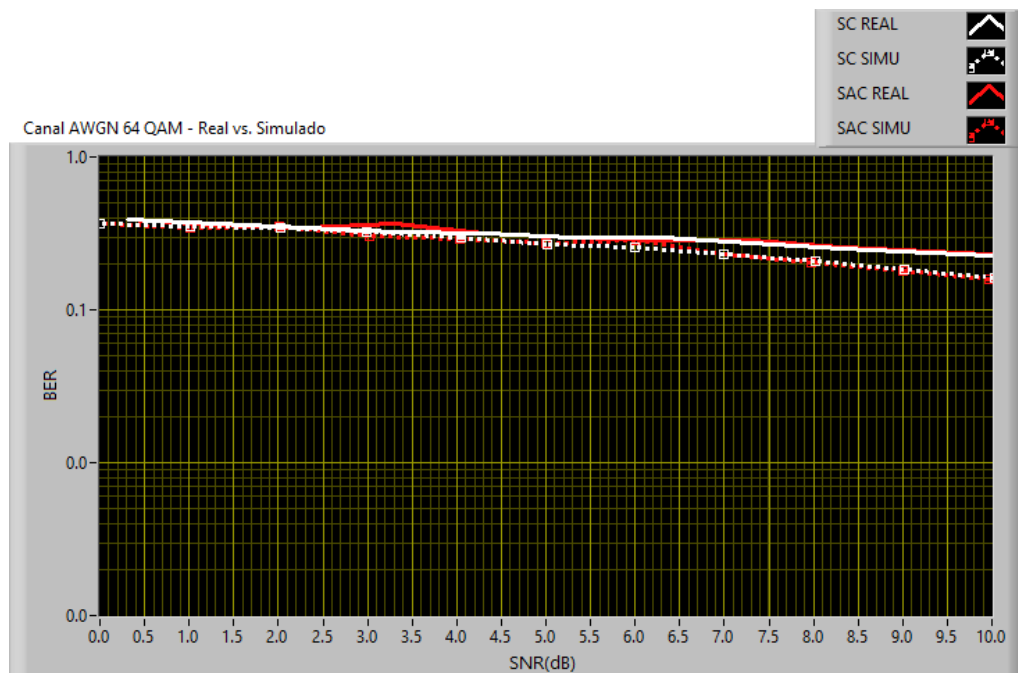


Fig. 5.32. SNR vs. BER 64 - QAM canal AWGN Simulación- Transmisión Real SC y SAC

Secuencia Willard

Los algoritmos presentaron mejor comportamiento es sus valores al cambiar la secuencia de entrenamiento, teniendo comportamiento más estables.

4.3.2 CANAL RAYLEIGH

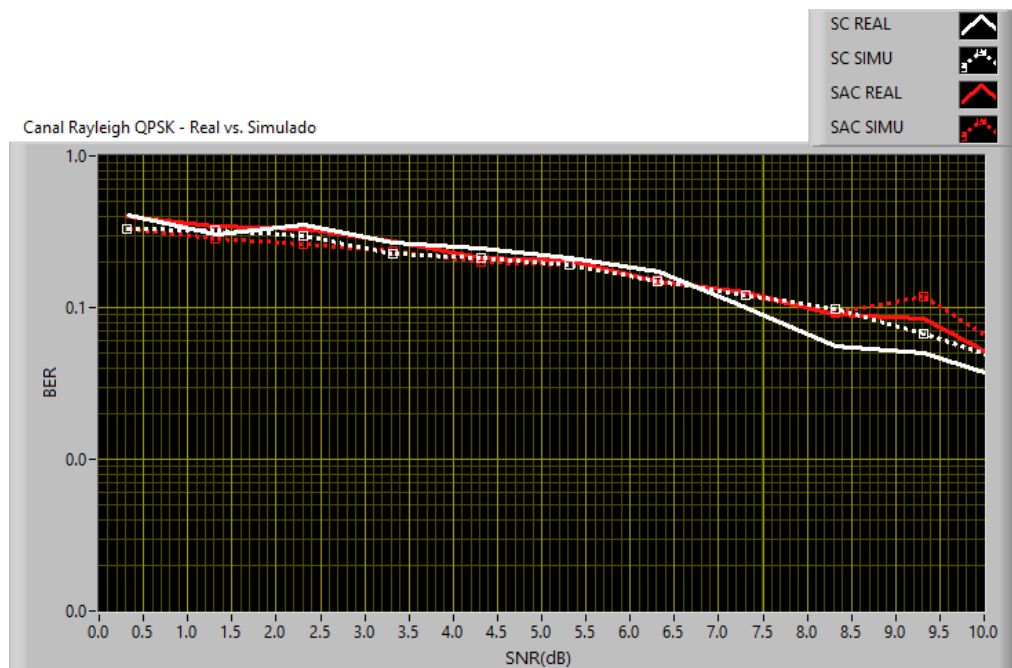


Fig. 5.33. SNR vs. BER QPSK canal RAYLEIGH Simulación- Transmisión Real SC y SAC
Secuencia Barker

Utilizando el canal Rayleigh igual se aprecia la diferencia de Potencia recibida debido a los equipos, pero en este canal es más estable la curva del sincronizador SC tanto en simulación como en transmisión real.

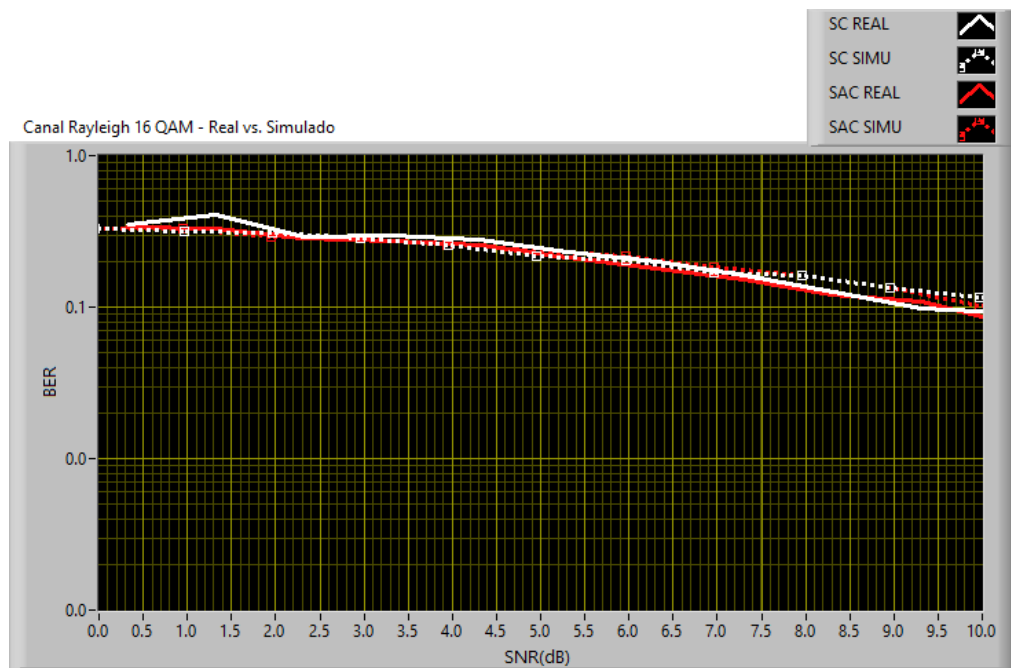


Fig. 5.34. SNR vs. BER 16 –QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC
Secuencia Barker

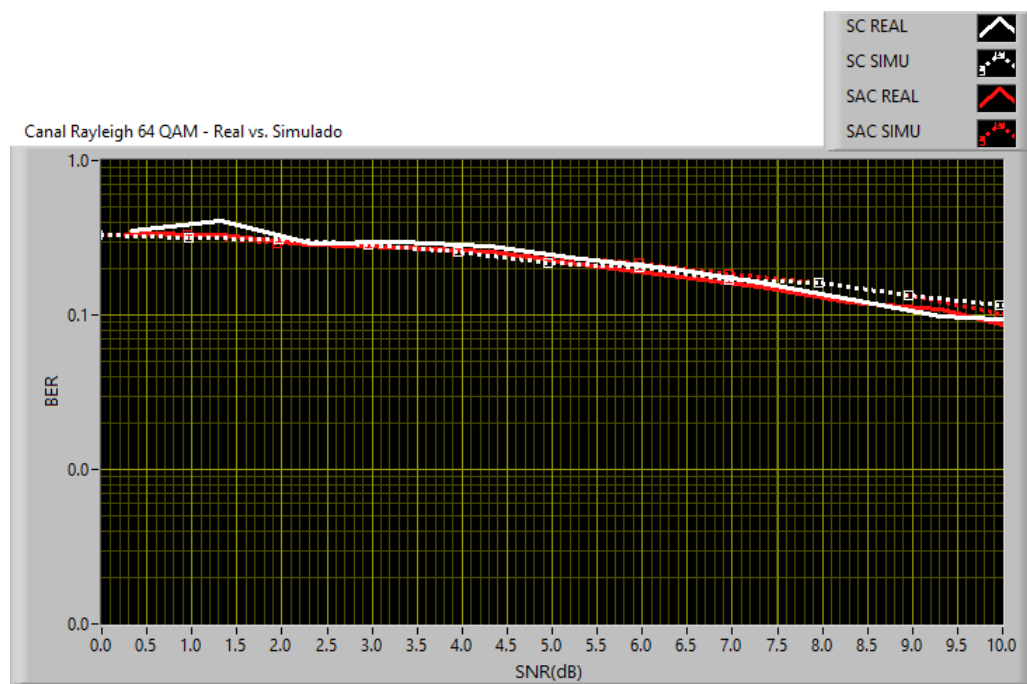


Fig. 5.35. SNR vs. BER 64 - QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC

Secuencia Barker

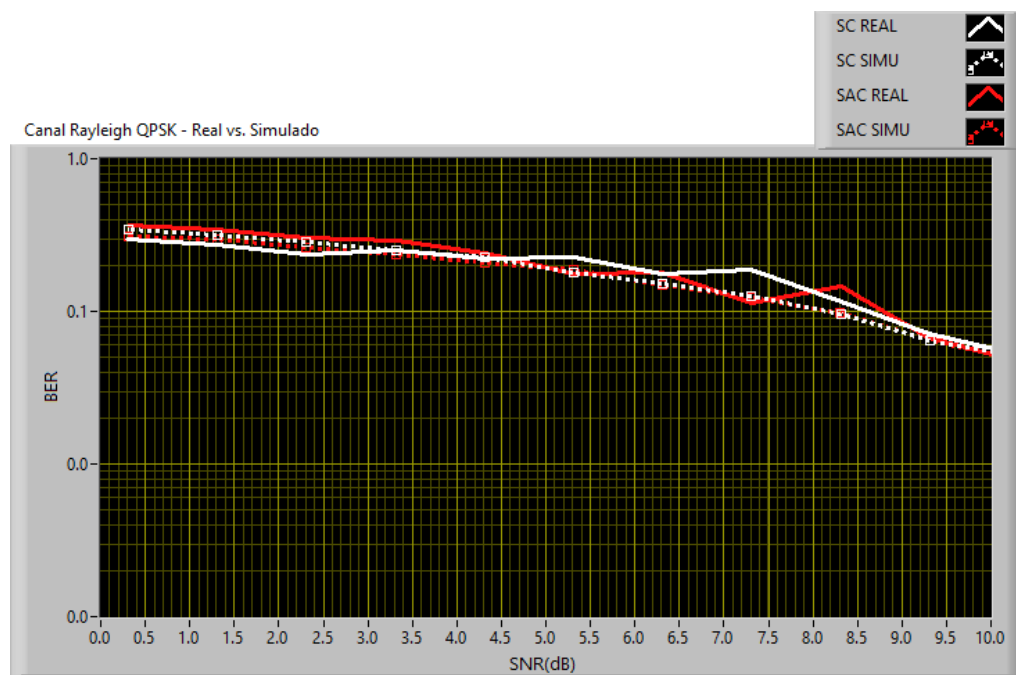


Fig. 5.36. SNR vs. BER QPSK canal RAYLEIGH Simulación- Transmisión Real SC y SAC

Secuencia Willard

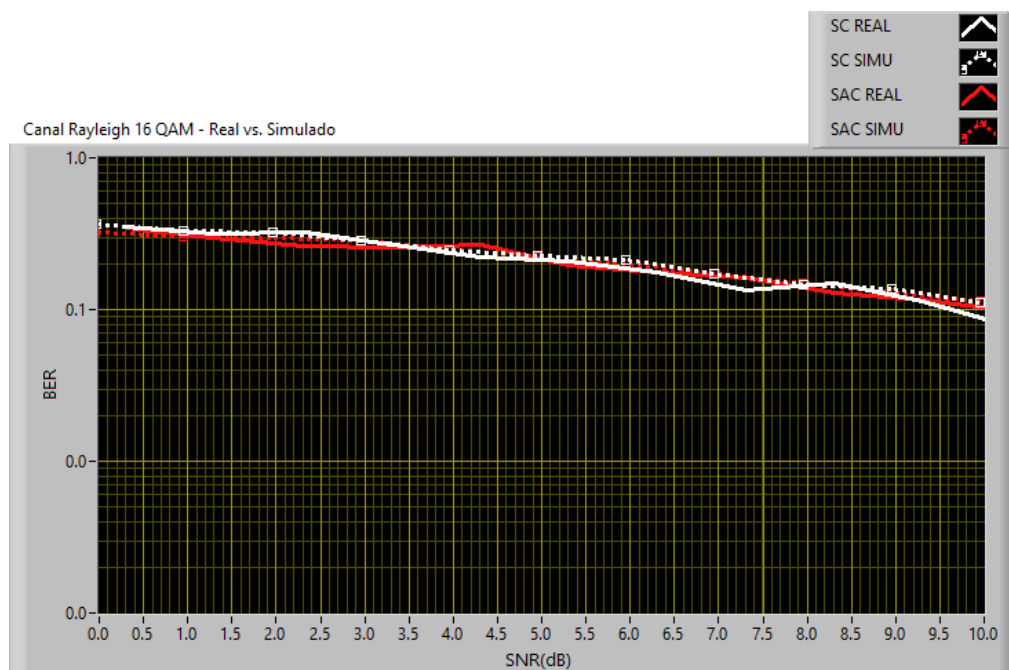


Fig. 5.37. SNR vs. BER 16 –QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC

Secuencia Willard

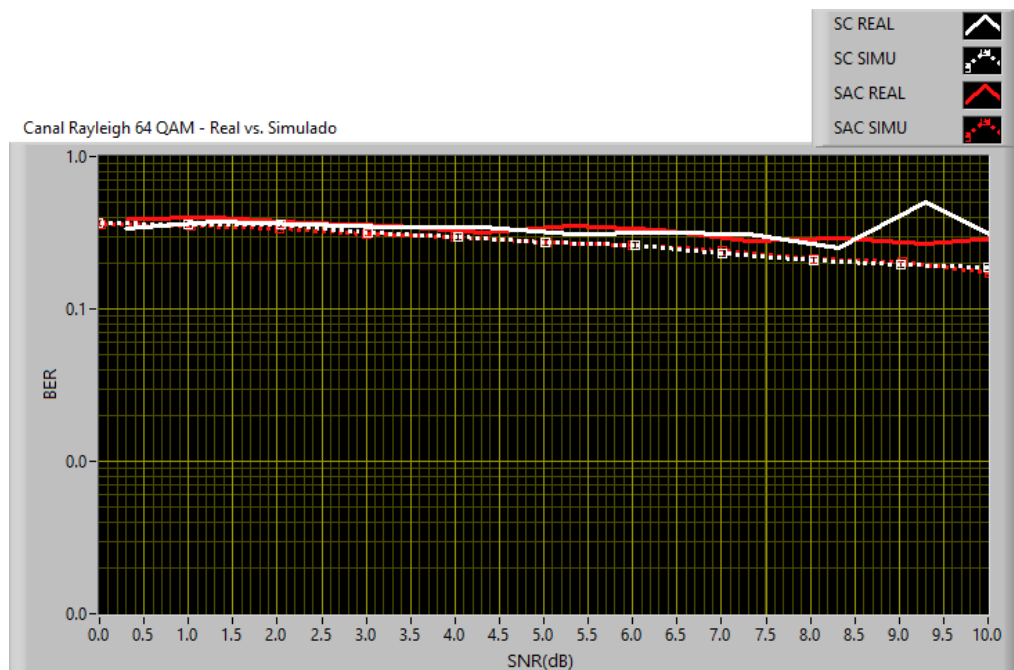


Fig. 5.15. SNR vs. BER 64 - QAM canal RAYLEIGH Simulación- Transmisión Real SC y SAC
Secuencia Willard

Al cambiar el tipo de secuencia de entrenamiento se observa que las gráficas son más estables al utilizar el sincronizador SAC sin importa el tipo de modulación, por lo tanto se determinó que el algoritmo adaptado a una portadora es más eficiente al emplear secuencia Willard.

También se puede observar que tan propensos cada algoritmo puede generar errores durante su funcionamiento. El algoritmo dado por la práctica es mucho más estable cuando se utiliza

secuencia Barker, esto se comprobó mediante las gráficas anteriores.

Por lo que esto demuestra que la adaptación del algoritmo fue correcta y el sistema tiene la misma funcionalidad como si se trabajara solo con el algoritmo SC.

Las gráficas están lo más linealizadas posibles, pero aun así se ve la presencia de grandes deformaciones que se deben a algún dato erróneo durante la captura.

4.4 DESEMPEÑO DEL SISTEMA AL INSERTAR UN DELAY AL CANAL

En esta prueba analizaremos como afecta un delay ingresado al canal. Las pruebas se basan en las que vienen en el manual de prácticas, tomando nuevamente como referencia al algoritmo SC.

Se configuran los parámetros en el Simulador para demostrar el funcionamiento al insertar un delay en un canal ISI con una trayectoria $0.247+0.247i$, el delay es $10T_s$ donde T_s es el tiempo de símbolo que en este caso es el inverso del parámetro TX Sample Rate; obteniendo un delay $1E-6$ para una Sample Rate de 10MHz.

SINCRONIZADOR DE TRAMA SLIDING CORRELATOR Y SCHMILD & COX

Utilizando el Sincronizador Basado en Correladores (Sliding Correlator) y Schmid & Cox tanto con la Secuencia de entrenamiento Barker y Willard de longitud 11 se emula un canal ISI con una trayectoria y como impedimento del canal un delay de $1\text{E-}6$, este delay es una fracción del periodo de símbolo por lo tanto se tiene un efecto de error en la sincronización, estos offsets crean un error entre los índices de los símbolos transmitidos y recibidos debido a esto es necesario un sincronizador de trama ya que el delay es un múltiplo del periodo de símbolo estos retrasos no son posibles corregir con el sincronizador de símbolo.

En las Figuras 4.39, 4.40, 4.41 y 4.42 se puede observar que se produce una buena estimación de delay del canal utilizando secuencia de entrenamiento Barker y Willard, esto también se puede observar en el parámetro *Error Statistic* que da un valor de 0, ya que este parámetro se calcula haciendo la diferencia entre el delay ingresado como parámetro al

canal y el delay estimado por el sincronizador de símbolo más el sincronizador de trama.

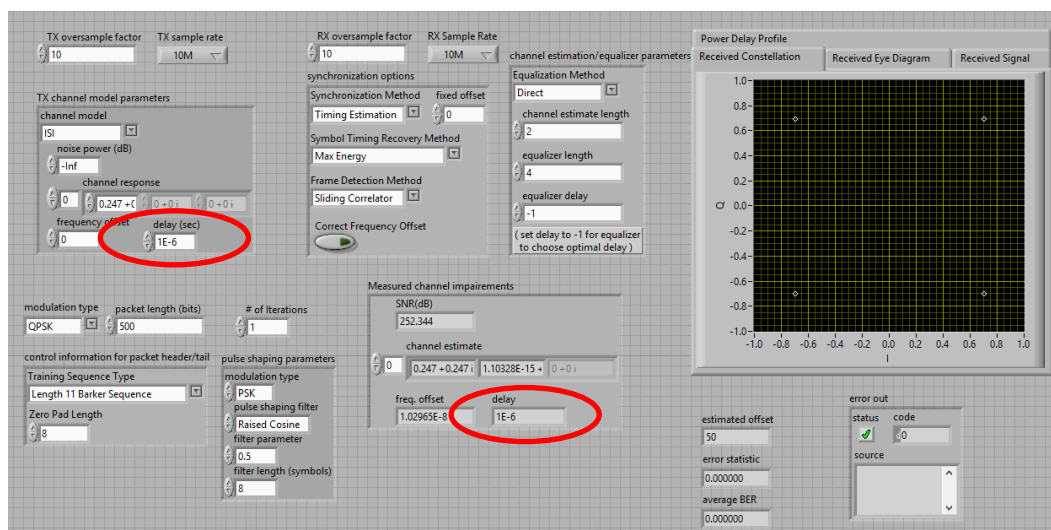


Fig. 5.169. Parámetros de Configuración del Simulador Sliding Correlator – Barker Sequence

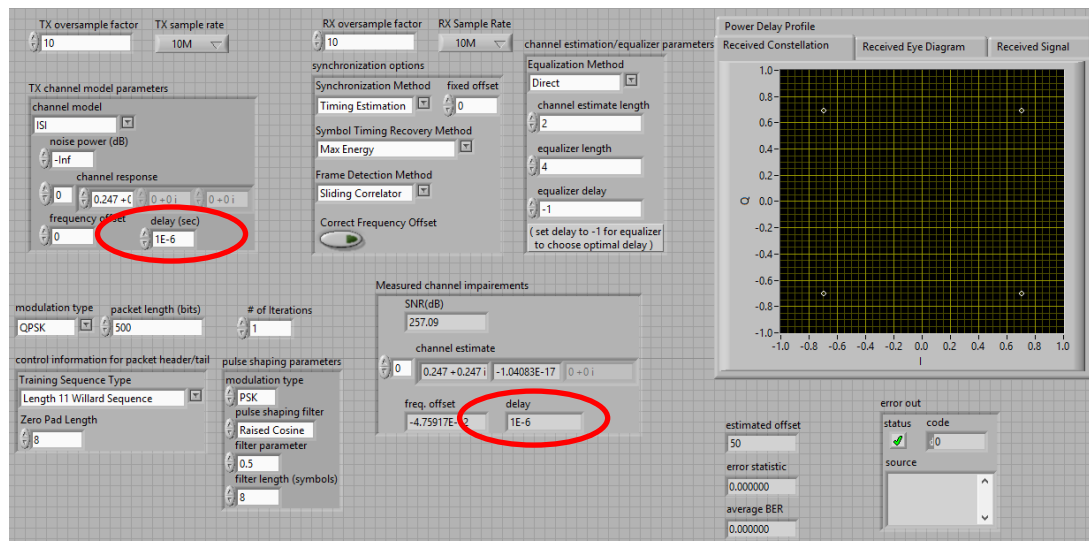


Fig. 5.40. Parámetros de Configuración del Simulador Sliding Correlator – Willard Sequence

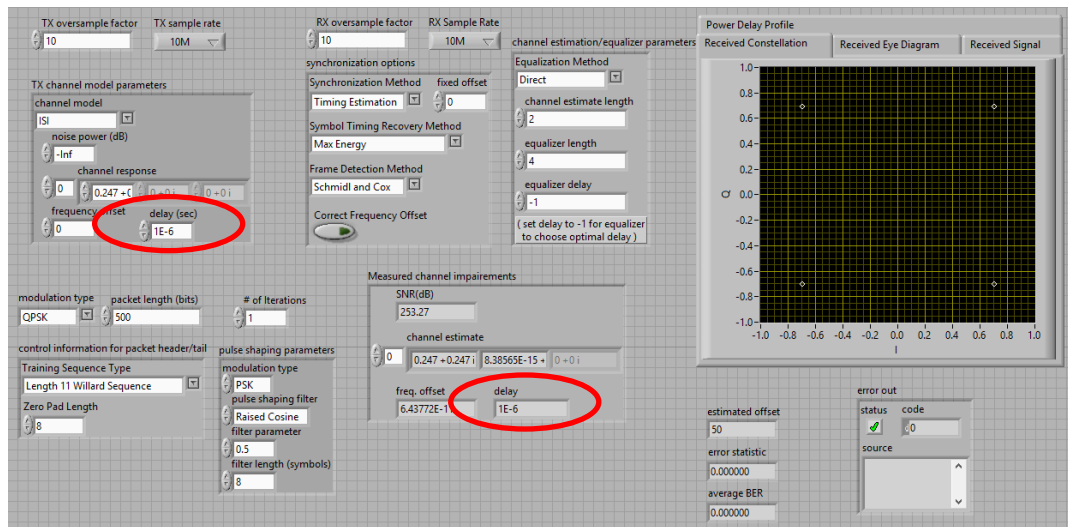


Fig. 5.41. Parámetros de Configuración del Simulador Schmid & Cox – Barker Sequence

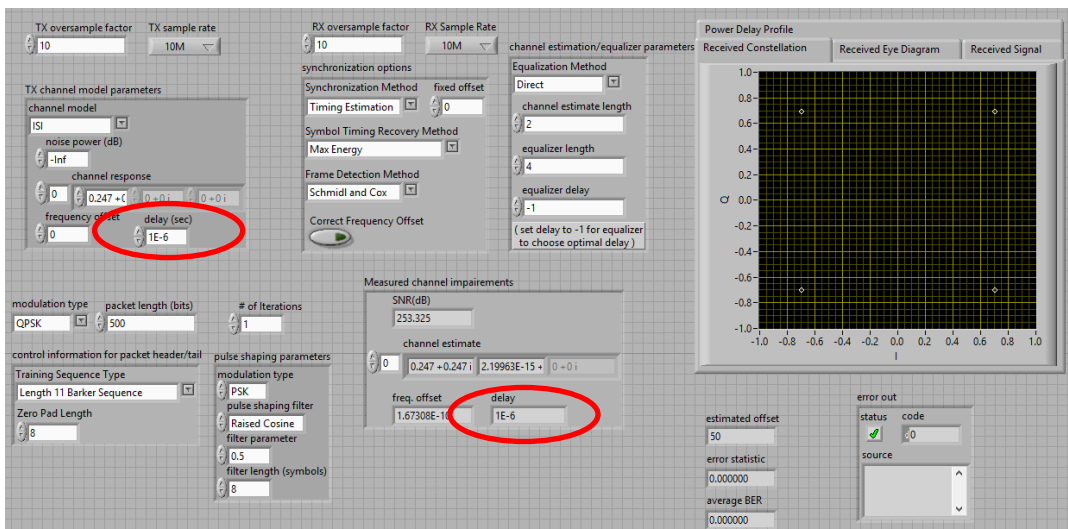


Fig. 5.42. Parámetros de Configuración del Simulador Schmid & Cox – Willard Sequence

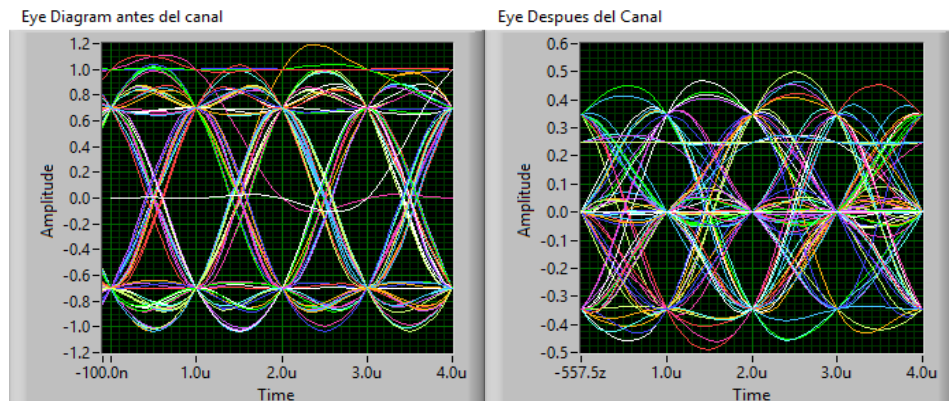


Fig. 5.43. Diagrama de Ojo antes y después del canal ISI

En la Figura 4.43 se puede observar los efectos de la interferencia intersímbolos que produce un fenómeno llamado Jitter en el cual los pulsos que componen la señal tendrán un desplazamiento, el diagrama de Ojo después de aplicar el canal ISI comienza a cerrarse por lo que el receptor tendrá más problemas al tratar de recuperar la señal.

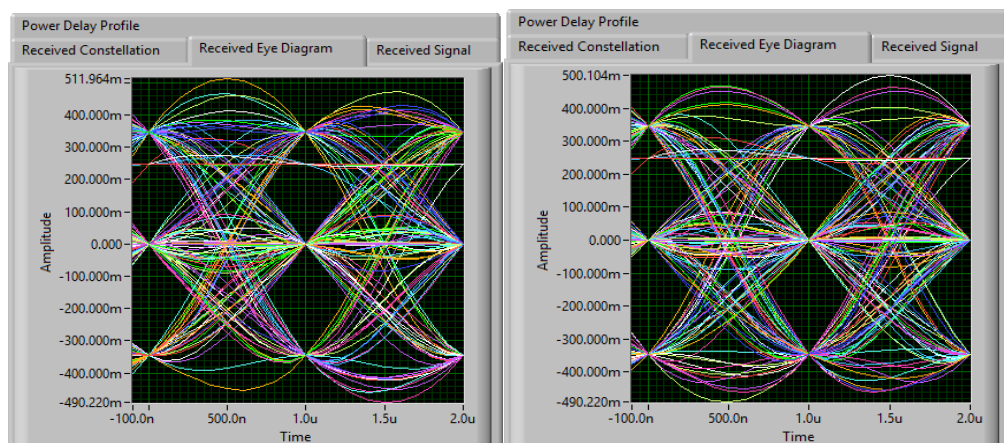


Fig. 5.44. Diagrama de Ojo Barker Sequence SC y SAC respectivamente

En la Figura 4.44 comparamos los Diagramas de Ojo de los dos Sincronizadores de Trama para secuencia Barker y con delay $1E-6$ y notamos que en el caso de Schmild & Cox tenemos líneas más juntas que en Sliding Correlator por lo tanto se puede observar una mayor apertura del ojo y menor interferencia intersímbolos.

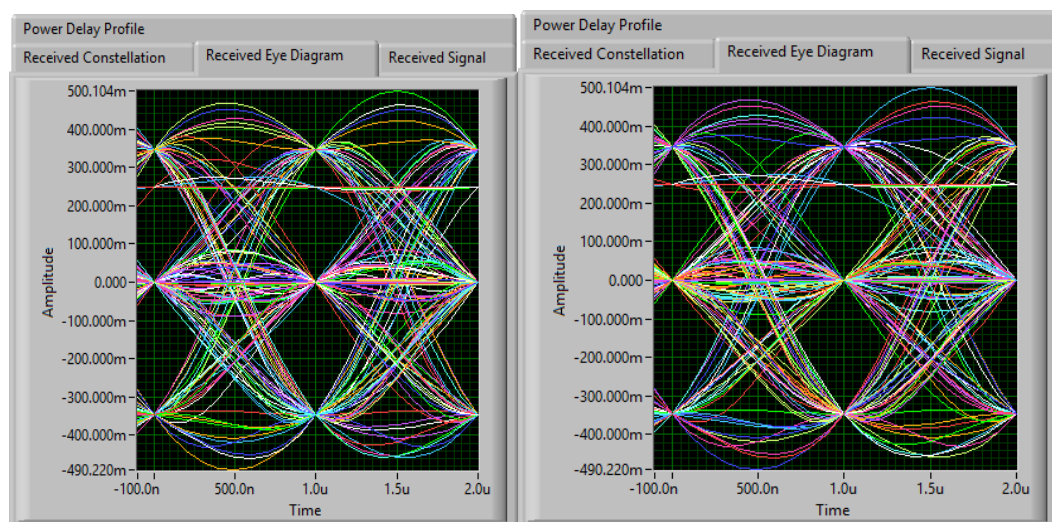


Fig. 5.45. Diagrama de Ojo Willard Sequence SC y SAC respectivamente

En la figura 4.45 se puede observar el mismo efecto que en la anterior gráfica, con lo cual se demostró que a pesar de añadir un impedimento al canal en este caso un delay los dos sincronizadores funcionan correctamente, pero el de Schmild & Cox es mucho más robusto antes estos impedimentos ya sea utilizando secuencia Barker o Willard.

4.5 DESEMPEÑO DEL SISTEMA AL INSERTAR UN DELAY Y FRECUENCIA DE DESPLAZAMIENTO

Se configuran los parámetros en el Simulador para demostrar el funcionamiento al insertar un delay 1E-6 y una frecuencia de desplazamiento 201 Hz en un canal ISI con una trayectoria $0.247+0.247i$.

SINCRONIZADOR DE TRAMA SLIDING CORRELATOR SIN Y CON CORRECCION DE FRECUENCIA DE DESPLAZAMIENTO (Frequency Offset).

La frecuencia de desplazamiento añadida al control Frequency offset = 201 Hz es la diferencia que existe entre la frecuencia de portadora entre el transmisor y receptor $f_o = f_c - f'_c$, donde $f_c = \text{frecuencia de portadora transmisor}$ y $f'_c = \text{frecuencia de portadora receptor}$, por lo tanto tendremos una rotación en la constelación de la señal recibida dada por $e^{j2\pi f_o t}$, como se puede apreciar en la Figura 4.7 y 4.8 tanto cuando el sincronizador utiliza una secuencia Barker o Willard.

Este desplazamiento de frecuencia de portadora complica mucho más el procesamiento en el receptor tanto para el sincronizador como para el ecualizador, por lo que una mala corrección de offset provocara errores en la recepción de la trama enviada.

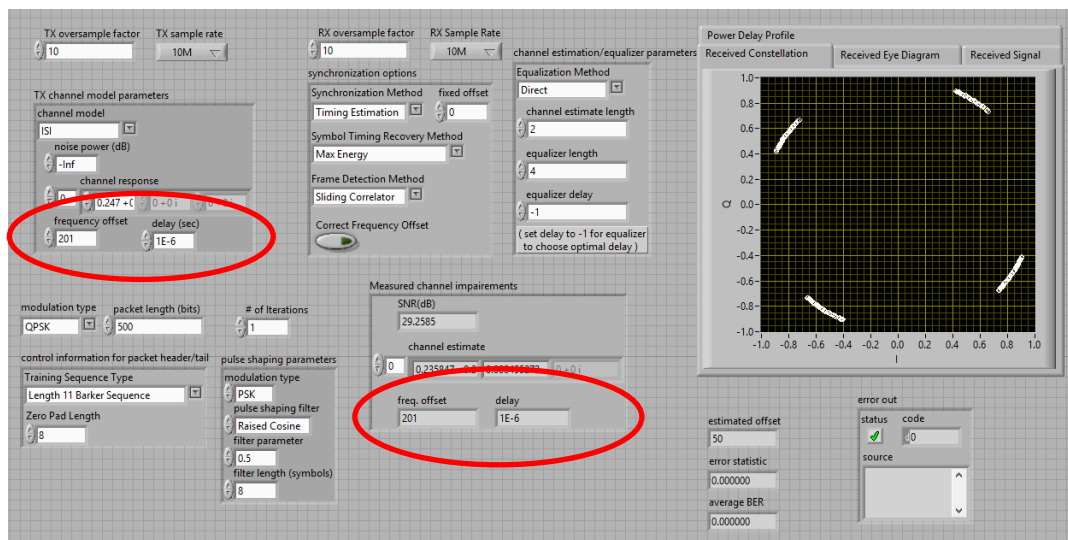


Fig. 5.46. Parámetros de Configuración del Simulador Sliding Correlator – Barker Sequence

Sin Correct Frequency Offset

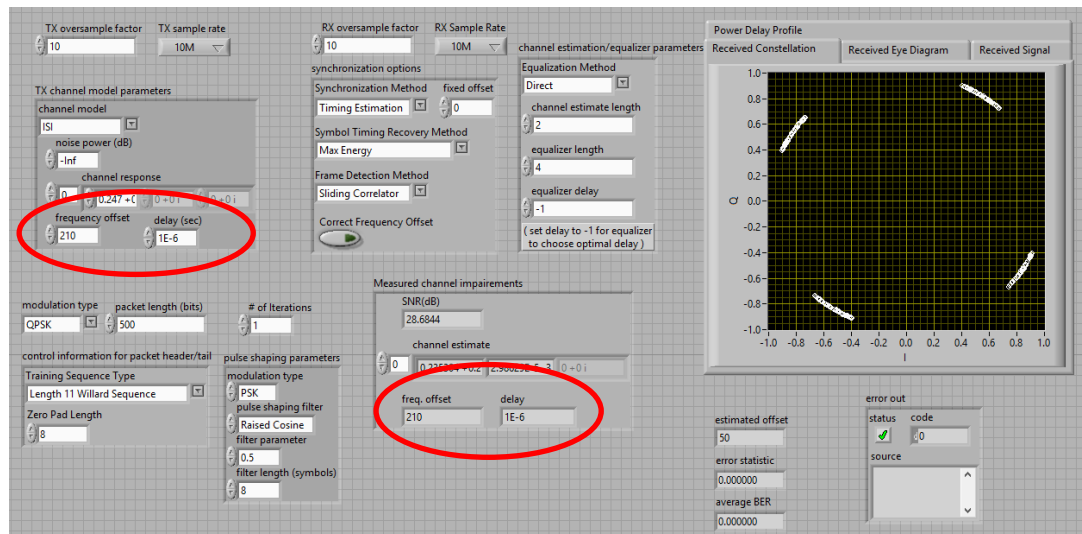


Fig. 5.47. Parámetros de Configuración del Simulador Sliding Correlator – Willard Sequence
Sin Correct Frequency Offset

Al activar el Control Booleano *Correct Frequency Offset* se logra corregir el desfase en la constelación, para lograr esta corrección se utilizó el algoritmo Moose en el cual se estima un valor de frecuencia de desplazamiento, una vez estimada esta frecuencia se remueve la rotación de la trama que llega del sincronizador Sliding Correlator.

En la figura 4.48 y 4.49 se puede observar la corrección de la rotación producida por frequency offset, según las prácticas realizadas el algoritmo de Moose corrige una frecuencia de desplazamiento mayor cuando se utiliza la secuencia de entrenamiento Willard.

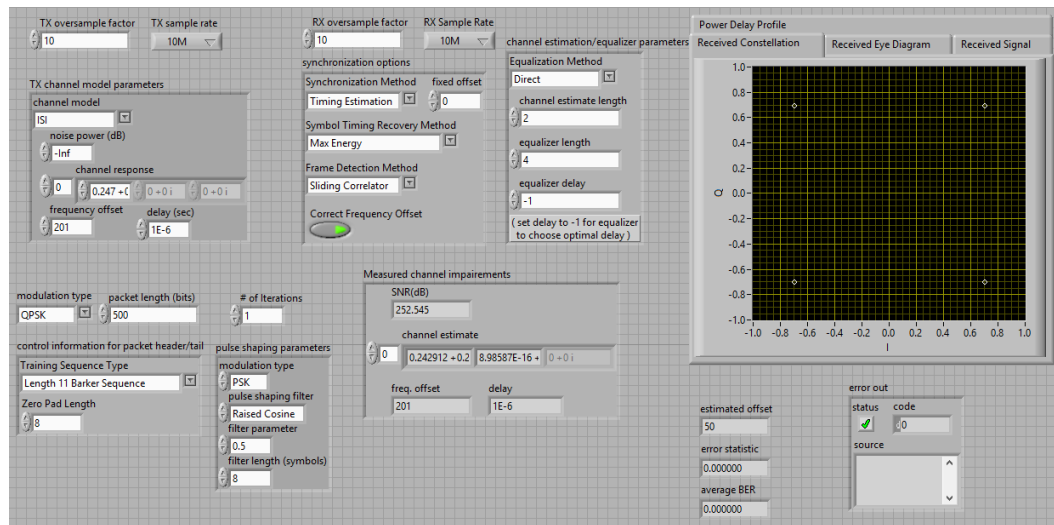


Fig. 5.48. Parámetros de Configuración del Simulador Sliding Correlator – Barker Sequence

Correct Frequency Offset

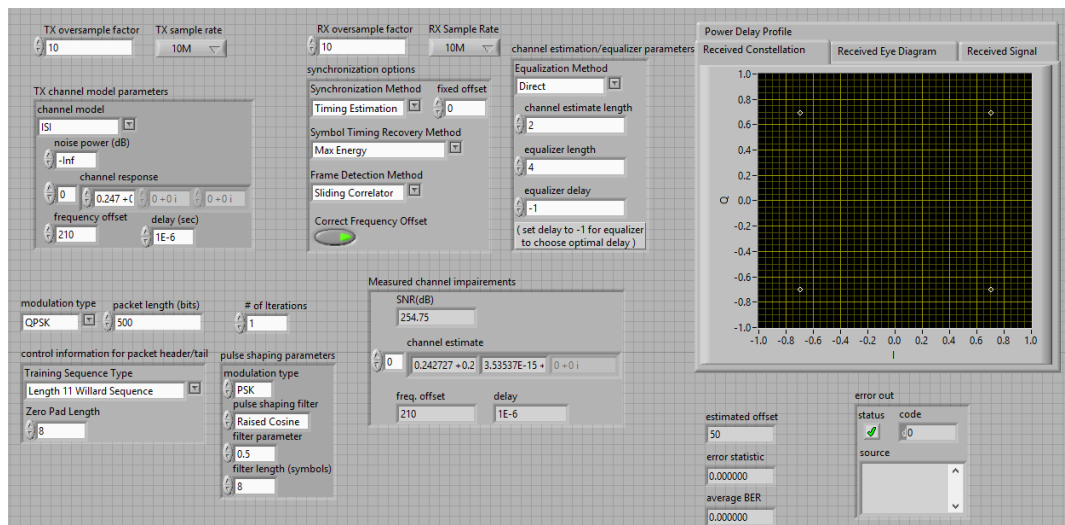


Fig. 5.49. Parámetros de Configuración del Simulador Sliding Correlator– Willard Sequence

Correct Frequency Offset

SINCRONIZADOR DE TRAMA SCHMILD & COX SIN Y CON CORRECCION DE FRECUENCIA DE DESPLAZAMIENTO (Frequency Offset).

La frecuencia de desplazamiento lo que provoca es una rotación de la constelación como se puede observar en la Figura 4.49 y 4.50 tanto cuando el sincronizador utiliza una secuencia Barker o Willard.

En este caso mostramos la constelación cuando el control booleano *Correct Frequency Offset* esta seteado en false para visualizar el efecto de rotación

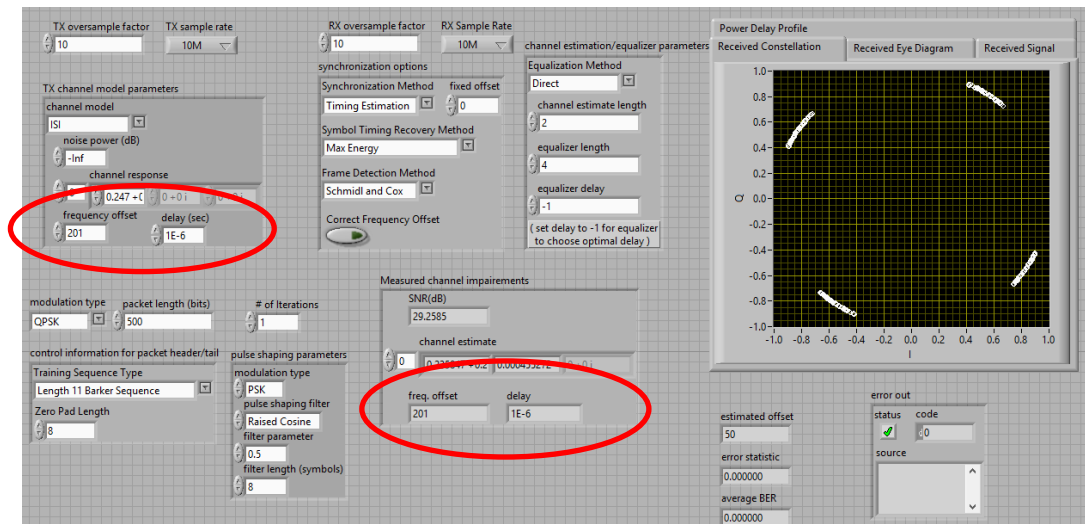


Fig. 5.50. Parámetros de Configuración del Simulador Schmid & Cox – Barker Sequence Sin

Correct Frequency Offset

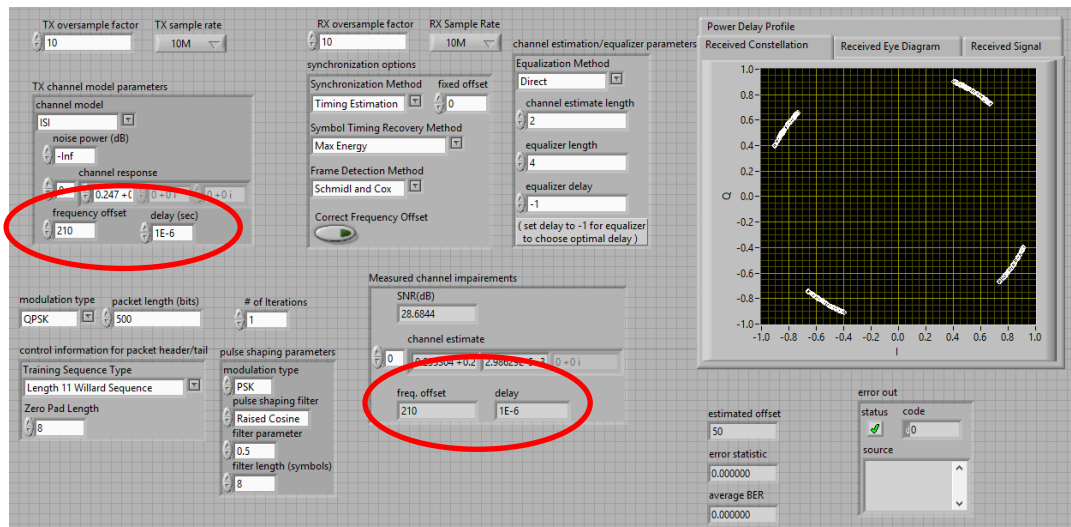


Fig. 5.51.. Parámetros de Configuración del Simulador Schmild & Cox– Willard Sequence Sin Correct Frequency Offset

Al activar *Correct Frequency Offset* se observa que ya no rota la constelación, el sincronizador de trama Schmild and Cox da como parámetro el ángulo para estimar la frecuencia de desplazamiento y eliminarlo de la trama este desplazamiento.

Schmild & Cox es más robusto que Moose, corrige frecuencias mayores tanto al trabajar con Secuencia Barker o Willard.

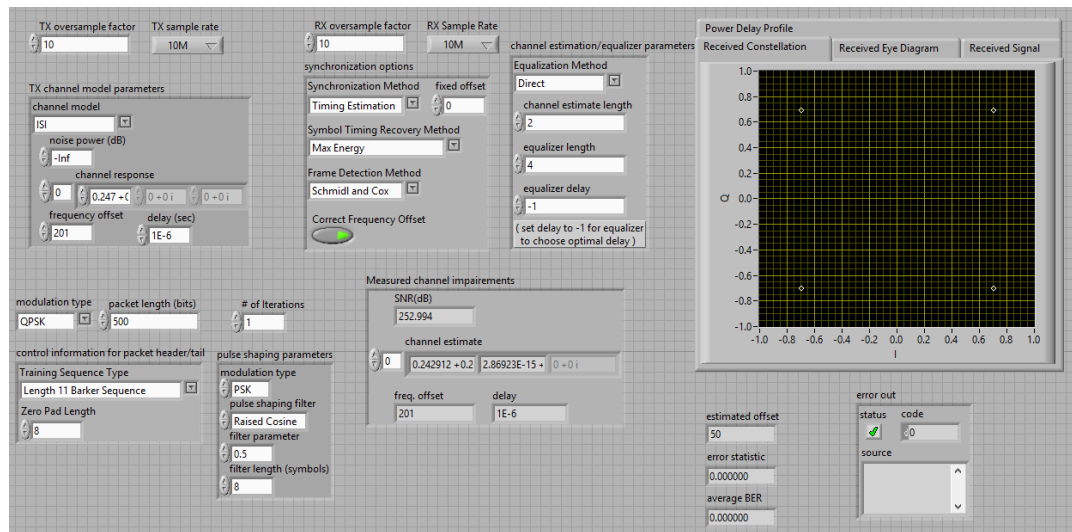


Fig. 5.52. Parámetros de Configuración del Simulador Schmid & Cox – Barker Sequence

Correct Frequency Offset

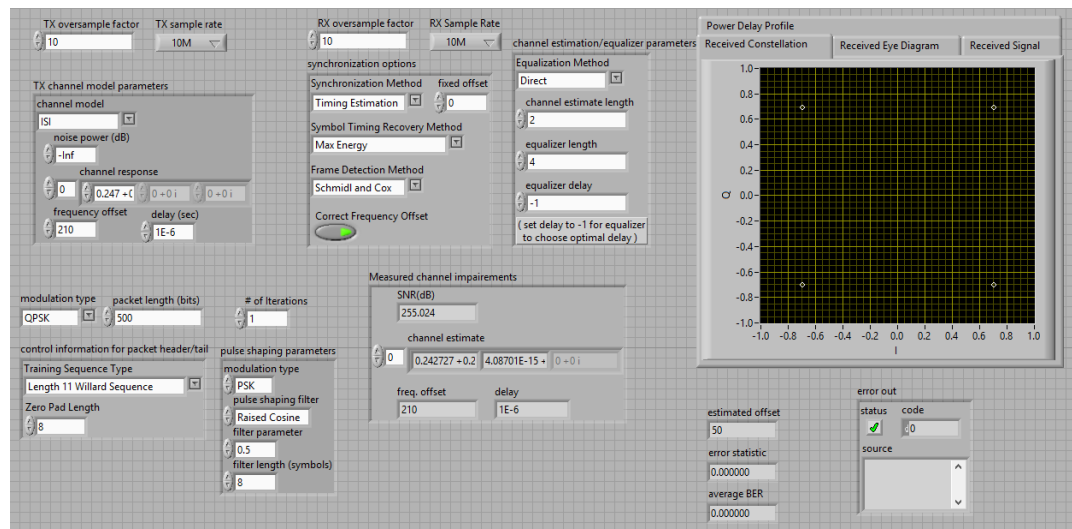


Fig. 5.53. Parámetros de Configuración del Simulador Schmid & Cox– Willard Sequence

Correct Frequency Offset

Al ingresar un valor de frecuencia de desplazamiento 201 Hz realizamos la prueba con el algoritmo SAC utilizando transmisión

real para ver como se ve afectados parámetros tales como BER y Sincronizador de Trama.

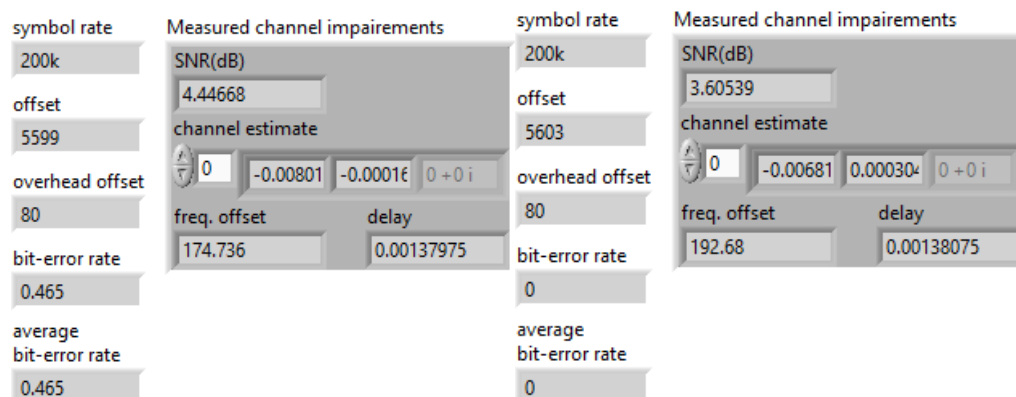


Fig. 5.54. Antes y Después de Correct Frequency Offset

Observamos como el BER mejora al activar el corrector de frecuencia de esta manera comprobamos la eficiencia de nuestro algoritmo trabajando sobre un enlace real con una frecuencia de portadora 915 MHz.

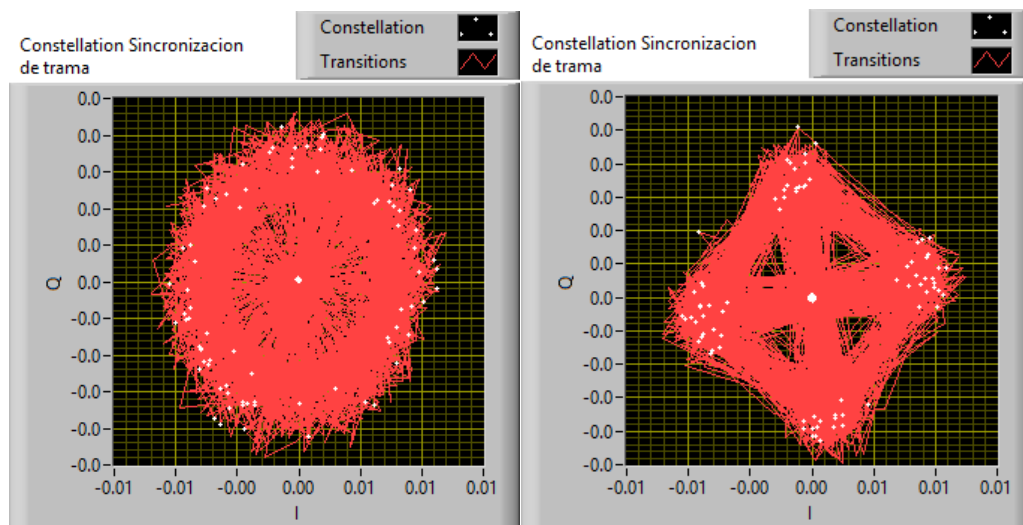


Fig. 5.55. Antes y Después de Correct Frequency Offset Constelación

Observamos que el sincronizador de trama no puede detectar el inicio de la misma sino se corrige la frecuencia de desplazamiento.

De acuerdo a todas las pruebas realizadas se determinó que en transmisión real el algoritmo Moose_ASC corrige un valor de frecuencias mayor al Moose de la práctica tanto para secuencia Barker como Willard.

En simulación empleando secuencia Barker el algoritmo SC alcanzó un máximo punto de corrección de frecuencia de 10924 Hz a diferencia de SAC que alcanzó un máximo de 22727 Hz. Ya sea que tuvieran o no configurado el delay.

Cambiando la secuencia a Willard en SC se obtuvo una mejora donde el algoritmo extendió su capacidad de corregir el desplazamiento en el rango de 10924 Hz a 11100 Hz, donde el análisis nos permitió observar que se produce un error de corrección por cada 10 muestras que se toma por debajo de los 11000. Mientras que estando en el límite, 11100 si tiene 5 errores por cada 10 muestras. En SAC no presento mejora respecto a los valores obtenidos con la secuencia Barker.

Posteriormente se analizó solo los sincronizadores, es decir se busca en qué valor de frecuencia de desplazamiento máximo el algoritmo comete errores al buscar el inicio de trama. Para esta práctica no se toma en cuenta el BER sino el valor del inicio de trama. Se varió el ingreso de desplazamiento de frecuencia y se observa el valor de Frame Begin para el caso de SAC u INICO en el caso de SC. En SC se observó que tiene el mismo punto de ruptura que cuando se analizó en la prueba de corrección de frecuencia, es decir que el algoritmo presento errores en la búsqueda de trama a partir del desplazamiento de frecuencia de 10924 Hz. En cambio el SAC soporto hasta 40000 Hz de frecuencia desplazada mostrando a lo máximo 2 errores por cada 10 muestras.

Al cambiar la secuencia a Willard SC alcanzo hasta 12000 Hz, de desplazamiento de frecuencia con 4 errores por cada 10 muestras, lo que representa una mejoría considerable del algoritmo al simple hecho de cambiar la secuencia de entrenamiento. SAC soporto hasta 220000 Hz de desplazamiento obteniendo 2 errores por cada 10 muestras lo cual es un comportamiento normal del algoritmo.

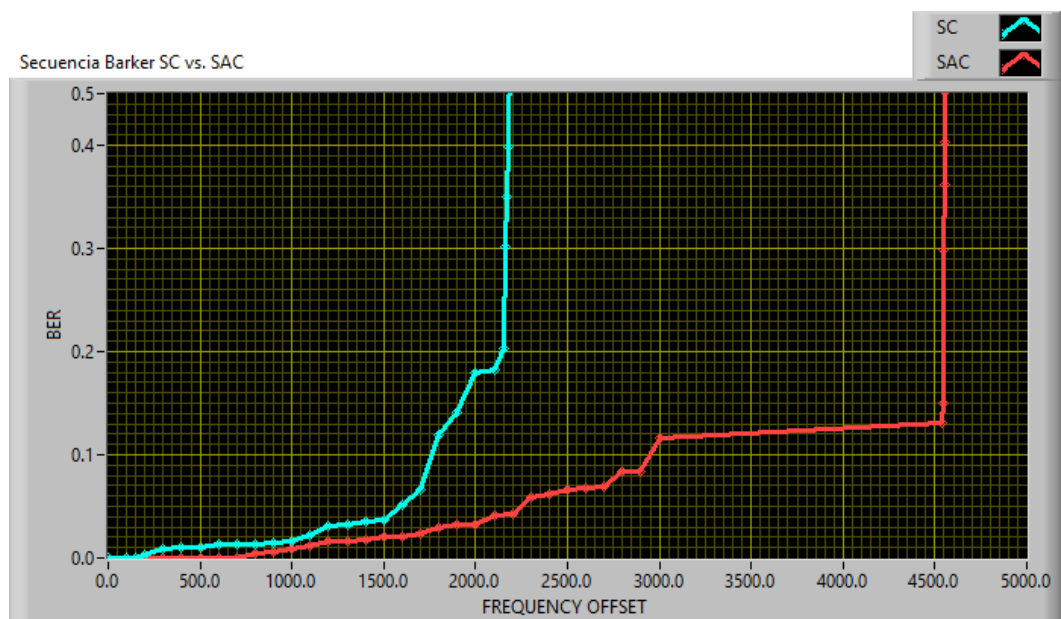


Fig. 5.56. Frequency Offset Vs BER en transmisión real Sliding Correlator vs Schimdl and Cox con secuencia Barker

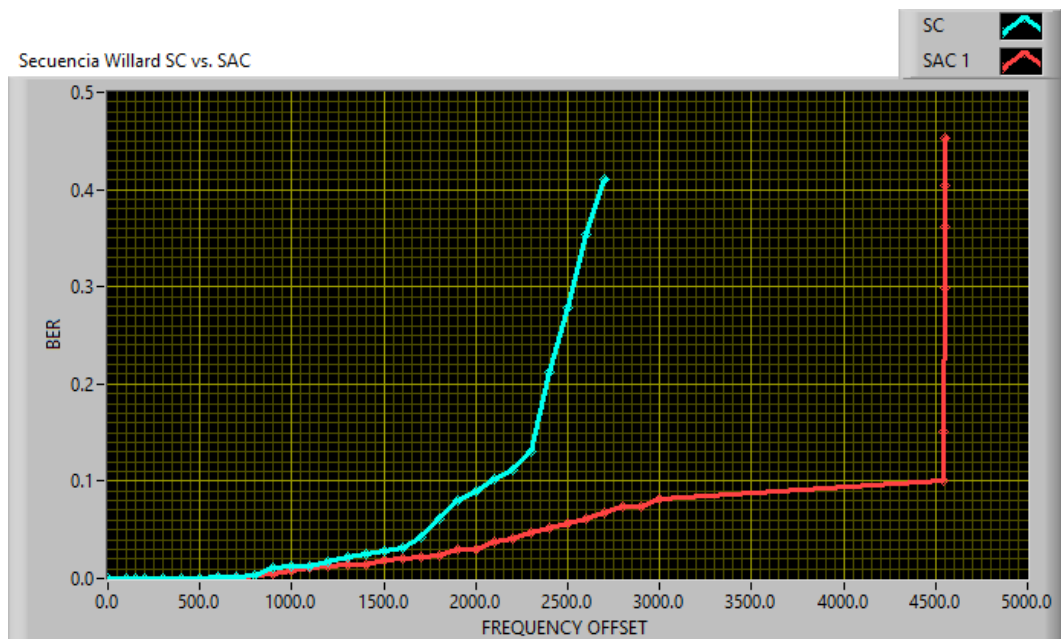


Fig. 5.57. Frequency Offset Vs BER en transmisión real Sliding Correlator vs Schimdl and Cox con secuencia Willard

Las figuras 4.56 y 4.57 muestran el comportamiento de los dos algoritmos durante la transmisión real ante diferentes secuencias de entrenamiento.

En fig. 4.56 se aprecia como el algoritmo SAC presenta un máximo valor es de 4550 Hz con un BER 0.13 respecto al SC con 2150 Hz con un BER de 0.2.

Al cambiar la secuencia, fig. 4.57, el algoritmo SAC presenta como máximo valor 4550Hz con un BER 0.13 mientras que el SC presenta una frecuencia de 2400Hz con un BER de 0.21.

Corroborar al hecho de que se incrementa la robustez de un algoritmo al cambiar la secuencia para el caso de SC. Mientras que para SAC no presento cambios en el valor máximo pero la curva es más suavizada. Estos cambios se deben a que las propiedades de correlación de la secuencia Willard son mejores respecto a la secuencia Barker.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. La elaboración de este proyecto permitió llevar a cabo la creación de una herramienta de estudio, que ofrece las bases necesarias para que los estudiantes que dispongan de ella, puedan conocer un poco más acerca de las experiencias y situaciones prácticas que envuelven el estudio teórico de determinadas materias relacionadas con el ámbito de las Telecomunicaciones.
2. La sincronización de trama en un sistema de comunicación tiene como propósito resolver retardos que no pueden ser corregidos por el sincronizador de símbolo, ya que este no es capaz de corregir retardos, que son múltiplos del periodo de símbolo, por tal motivo se utilizan los sincronizadores de trama: el sincronizador basado en Correladores necesita en el receptor conocer una secuencia patrón lo que lo hace ineficiente frente al sincronizador Schmild & Cox que solo trabaja con la señal que llega al receptor.
3. Al realizar las pruebas en el capítulo 4 tenemos que ante diferentes modulaciones el BER aumenta debido a que las regiones de los símbolos

se encuentran más cercanas, lo mismo sucede al cambiar el ambiente de propagación en el caso de una Canal Rayleigh tenemos una mayor tasa de BER que en ISI y AWGN, se produce más degradación de la señal en el receptor.

4. La frecuencia de portadora por lo general, en la práctica, se debe a un oscilador local que no siempre es el mismo en el transmisor y receptor, lo que provoca una diferencia mostrándose como una rotación en la constelación. Debido a las pruebas realizadas se tiene una mayor eficiencia utilizando el algoritmo Schmild & Cox con secuencia de entrenamiento Willard o Barker con las cuales se logra hacer un corrección hasta 22727 Hz, lo cual se logra superar a la frecuencia máxima corregida por Sliding Correlator, este valor máximo de corrección de frecuencia depende de varios parámetros tales como: tipo de canal, Oversample factor, sample rate y pulse shaping.
5. El cambio de frecuencia genera mejoras en los algoritmos según la situación que se analiza. Mostrando que, sin realizar cambios en la matemática del algoritmo, el solo hecho de cambiar una mejor palabra (con mejores propiedades de auto-correlación) incrementa la robustez del algoritmo, siendo unos KHz (en el caso de SC) hasta más de 100KHz (en el caso De SAC). Dejando como indicio de futuro temas de investigación

la búsqueda de una palabra que incremente la robustez y mejore los algoritmos dentro de un sistema.

6. El SAC es un algoritmo que es provisto de su propio método de corrección, mostro estar correctamente adaptado, presentado un buen funcionamiento ante altos valores de desplazamientos y la dificultades de los canales aplicados, tanto para simulación como en transmisión real. Su robustez se mostró más en la resolución del inicio de la trama, ante grandes desplazamientos de frecuencias y retardos, pero cabe recalcar que el algoritmo es más propenso a fallos que el SC en valores bajos de SNR (alta potencia de ruido y desvanecimientos de canal). Por lo que se deja a consideración para futuras investigaciones.
7. La presencia de varios picos continuos de correlación en el algoritmo SAC ante valores bajos de SNR (alta potencia de ruido y efectos producidos por el canal), genera mayor probabilidad de error en su funcionamiento generando altas tasas de BER.

RECOMENDACIONES

1. Al realizar transmisión real observar en el VI Top TX cuanto esta la duración del paquete (packet duration) ya que esto varia con la cantidad de bits a transmitir, esta duración se debe colocar en el Top RX en Capture Time; si este tiempo no es el correcto se observara errores en constelación recibida y por lo tanto un bit error rate alto.

2. En el transmisor al añadir la secuencia de entrenamiento se recomienda modularla con el mismo tipo de modulación que los símbolos a transmitir ya que generara errores en los puntos de la constelación ubicándolos fuera de las regiones.

3. Dada la relación $f_e \leq \frac{1}{2TN_t}$, para lograr una mayor corrección de frecuencia de desplazamiento se debe seleccionar N_t pequeño, pero esto a su vez reducirá la estimación de trama por lo tanto lo que se debe hacer es utilizar múltiples repeticiones de una secuencia de entrenamiento corta.

4. Para una correcta toma de datos, se debe realizar varias muestras y obtener un promedio de las mismas (para las muestras presentes se

tomaros de 5 a 10 muestras, según el caso), esto nos ayuda a tener una mejor percepción del funcionamiento del sistema y de nuestros algoritmos

5. Se espera que con la elaboración de este proyecto, de pie para la elaboración de proyectos similares, siendo un mecanismo de ayuda y motivación para futuros ingenieros y profesionales del área.

REFERENCIAS

- [1] Robert w. Heat Jr. Ph.D., PE. *"Digital Communications: Physical Layer Lab using the NI USRP Platform"*, National Technology and Science Press
- [2] B.P.Lathi, *"Modern Digital and Analog Communication System"*, Third Edition, Oxford University Press, 1998.
- [3] LeonCouch W., II, *"Sistema de Comunicación Digitales y Analógicos"* Septima, Edicion en Español, Pretince Hall, 2008
- [4] Simon Haykin, MichealMoher, *"Introduction to Analog and Digital Communications"*, Second Edition, Jhon Wiley and Sons, Inc, 2007
- [5] Simon Haykin, *"Communication System"*, Fourth Edition, Jhon Wiley and Sons, Inc, 2001
- [6] *"Libro de Comunicaciones digitales"*, Capitulo 11: Sincronización, pág. 686 - 759 en: www.tsc.uc3m.es/~antonio/libro_comunicaciones
- [7] Scott L. Miller, Donald G. Childers *"Probability and Random Processes with applications to signal processing and communications"*, Elsevier Academic Press 2004
- [8] Bernard Sklar, *"Digital Communications: Fundamentals and Applications"*, Second Edition, Prentice Hall.
- [9] Timothy M. Schmidl and Donald C. Cox, *"Robust Frequency and Timing Synchronization for OFDM"*, IEEE TRANSACTIONS ON COMMUNICATIONS, Vol. 45, No. 12, December 1997
- [10] Simon R. Saundres, Alejandro Aragon Zavala, *"Antennas and Propagation for Wireless Communication System"*, Second Edition, Jhon Wiley and Sons, Inc, 2007.
- [11] Byron Floreano, Santiago Jácome, *"Mitigación del ISI (Interferencia Intersimbólica) sobre un Sistema WiMAX Fijo"*, Tesina de Seminario para obtener título de Ingeniero en Electrónica y Telecomunicaciones, FIEC- ESPOL 2010.

- [12] Eneraldo Castellanos, José Talero *“Análisis de propagación electromagnética en espacios cerrados: Herramienta Software en Matlab para predicción y simulación”*, Proyecto de grado para el título de Ingeniero Electrónico, Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones, Universidad Industrial de Santander, 2005.
- [13] Paulina Yáñez Arias, *“Estudio sobre los canales de desvanecimientos sobre redes fijas y móviles en sistemas de radio comunicación”* proyecto para la obtención del título de Ingeniero en Electrónica y Telecomunicaciones, Facultad de Ingeniería eléctrica y Electrónica, Escuela Politécnica Nacional 2009.
- [14] Jhon S. Saybold. PhD. *“Introduction to RF Propagation”*, Jhon Wiley and Sons, Inc, 2005
- [15] Gary Breed, *“Analyzing signal using the eye diagram”*, High Frequency Electronic, November 2005.
- [16] Kayeh Pahlavan, Allen H. Leveque, *“Wireless Information Networks”*, Second Edition, Jhon Wiley and Sons, Inc, 2005
- [17] Cory L. Clark, *“LABVIEW Digital Signal Processing with Digital Communication”* McGraw-Hill Companies, Inc
- [18] Jalal Jamal Hamad-Ameen, *“Proposed sets of Polyphase Spreading Sequences for DS-CDMA System”*, Proceedings of the 5th WSEAS International Conference on Telecommunications and Informatics, Istanbul, Turkey, May 27-29, 2006
- [19] José Luis Camargo Olivares, *“Modelo de cobertura para redes inalámbricas de interiores”*, proyecto de fin de Carrera Universidad de Sevilla, Capítulo 6 paginas 147 – 150.
- [20] Kaveh Pahlavan, Allen H. Levesque, *“Wireless Information Networks”*, Segunda edición, JhonWiley and Son’s 2005, capítulo 3, pag 57 – 64.
- [21] RECOMMENDATION ITU-R M.1225
- [22] Camilo Torres Zambrano, Carlos Iván Páez Rueda, *“Análisis de un sistema de comunicaciones afectado por los desvanecimientos plano y lento tipo Rayleigh”*. Trabajo de grado para optar al título de Maestría en Ingeniería Electrónica, Pontificia Universidad Javeriana, Bogotá, Colombia, 2008.

ANEXOS A

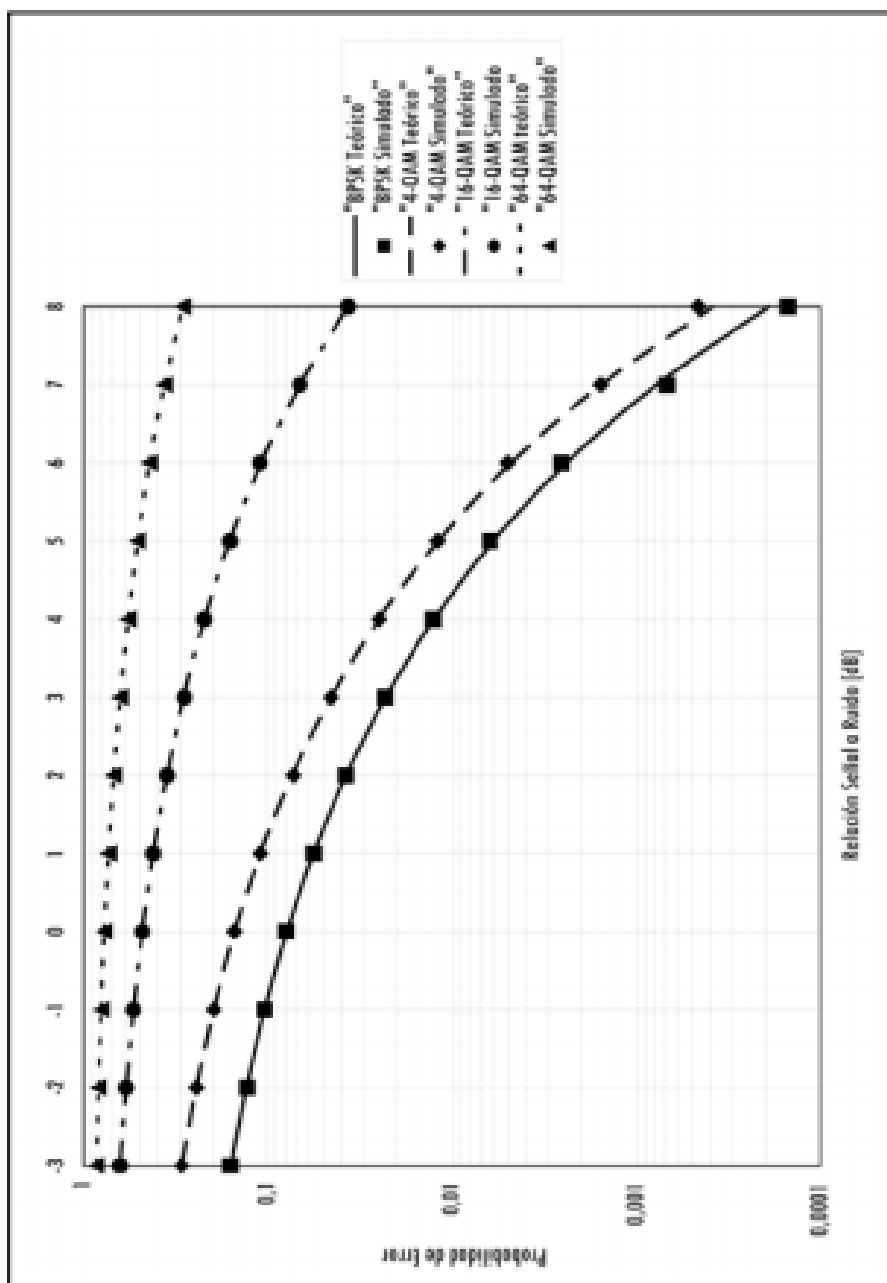
Tabla de parámetros de canal RECOMMENDATION ITU-R M.1225

Tap	Channel A		Channel B		Doppler spectrum
	Relative delay (ns)	Average power (dB)	Relative delay (ns)	Average power (dB)	
1	0	0	0	0	Flat
2	50	-3.0	100	-3.6	Flat
3	110	-10.0	200	-7.2	Flat
4	170	-18.0	300	-10.8	Flat
5	290	-26.0	500	-18.0	Flat
6	310	-32.0	700	-25.2	Flat

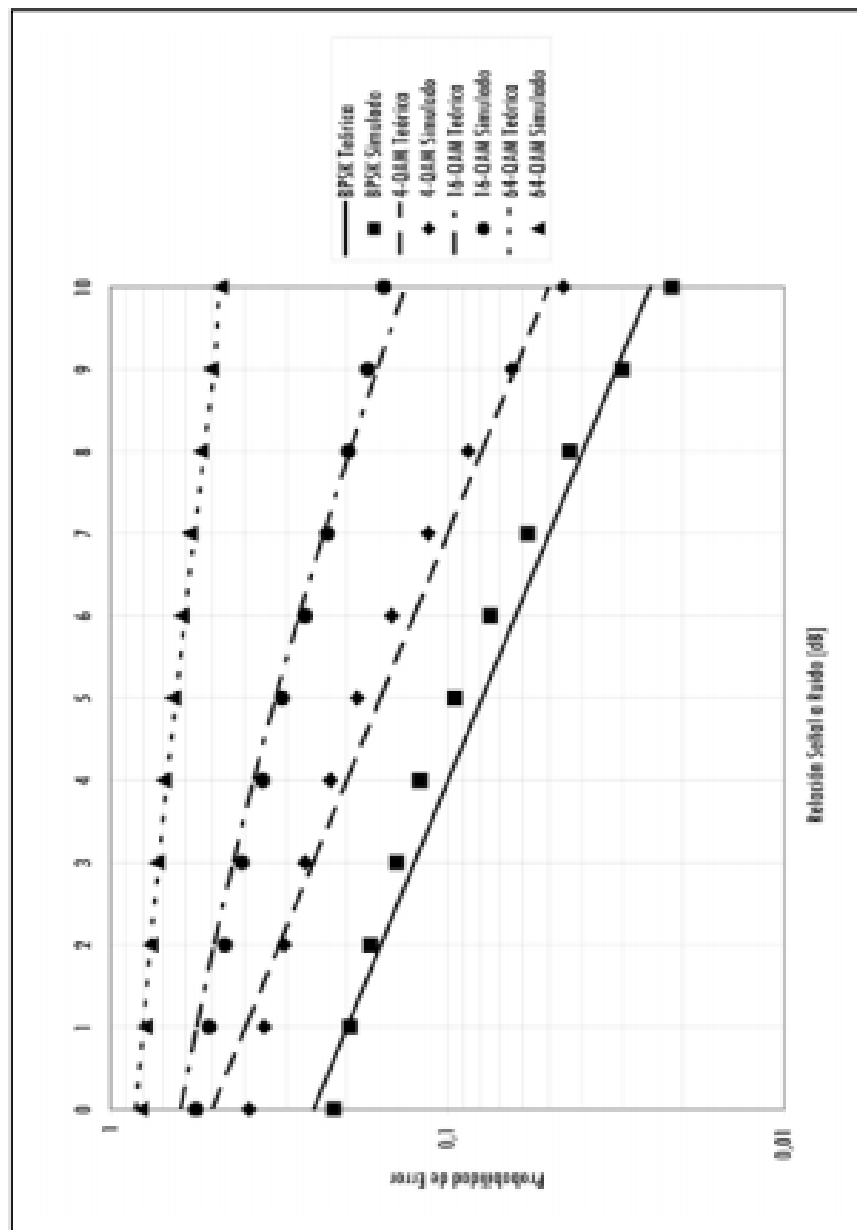
Parámetros utilizados para la implementación de canal Rayleigh.

ANEXOS B

Graficas de SNR vs BER de los canales AWGN y RAYLEIGH



B.1 SNR vs. BER de un canal AWGN [22]



B.2 SNR vs. BER de un canal RAYLEIGH [22]