



Casa abierta al tiempo

---

# UNIVERSIDAD AUTÓNOMA METROPOLITANA

---

**Programación y evaluación en un DSP del nivel físico de un módem OFDM para comunicación por la red eléctrica doméstica**

Idónea Comunicación de Resultados para obtener el grado de

MAESTRO EN CIENCIAS  
(CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN)

por

Daniel Torres Alvarado

Asesores:

Dr. Gerardo Abel Laguna Sánchez

Dr. Alfonso Prieto Guerrero

Noviembre, 2012



# Resumen

---

Las nuevas tecnologías de comunicación son constantemente estudiadas con el fin de mejorar aquellos puntos que contribuyan a maximizar las tasas de transferencia, ancho de banda, velocidad y, al mismo tiempo, se minimice el costo en infraestructura, pérdida de información o errores en los datos y, quizá el más relevante para nuestros tiempos, el consumo de energía.

A partir del año 1980, la red eléctrica dejó de ser sólo un medio de distribución de energía y comenzó a ser usada como un medio de transmisión de datos. La información que se envió inicialmente por el cableado eléctrico, fue de lecturas remotas. No fue hasta el año de 1997, cuando la red eléctrica fue presentada al mercado como una nueva tecnología de comunicación que llevaría el nombre de *Power Line Communication* (PLC)<sup>1</sup>. Desde entonces la inversión privada, instituciones de gobierno y de educación, han iniciado la explotación de los beneficios de este medio altamente viable y económico para el trayecto de la denominada última milla.

En el presente documento se presenta la comunicación idónea de resultados logrados en la investigación, diseño y evaluación en un procesador digital de señales de los bloques funcionales de un módem para la transmisión de datos por la red eléctrica doméstica. Los algoritmos propuestos en el presente trabajo de investigación, son inspirados en los modelos formales propuestos por Matthieu Crussière en [1].

En la investigación realizada, sobre los algoritmos necesarios para una transmisión bajo el esquema OFDM, se encontró la necesidad de poner en práctica algoritmos como: modulación OFDM (sigla del ingl. *Orthogonal Frequency Division Multiplexing*, 'multiplexado por división de frecuencias ortogonales'), sincronización, técnicas de estimación de canal, por mencionar algunos. Los algoritmos seleccionados después de la revisión bibliográfica, fueron programados y evaluados en un procesador digital de señales (de *DSP*, sigla del ingl. *Digital Signal Processing*) de la familia TMS032C6416T. La puesta en práctica de los algoritmos en el DSP, se llevo a cabo siguiendo las recomendaciones del fabricante para lograr el máximo desempeño.

---

<sup>1</sup>El nombre asignado a esta tecnología puede cambiar de acuerdo al país u organización privada que la investigue, desarrolle o ponga en práctica.

Durante la puesta en práctica se evaluaron algunas propuestas de mejora sobre el proceso de sincronización, tales propuestas como el diseño de una trama de calibración para la puesta a punta del sistema y una modificación en la estimación del inicio de los símbolos OFDM con base a la moda estadística. La implementación de ambas propuestas, mostró resultados satisfactorios, logrando disminuir el número de símbolos OFDM empleados en el proceso de sincronización. Los algoritmos implementados en el proceso de sincronización sólo permiten la puesta a punto a partir de 8 dB de valor de relación señal a ruido.

A lo largo del presente documento el lector podrá apreciar una revisión del estado del conocimiento sobre de la evolución y desarrollo de la tecnología PLC, así como la del estándar consolidado hasta el año 2012 sobre esta tecnología. La investigación realizada sobre el estándar obligó a la investigación y estudio de modelos formales de la capa física usados para la transmisión de datos con la tecnología PLC. Otra parte importante del documento, es el reporte sobre el desarrollo realizado a lo largo de la investigación, donde se presenta la programación y diseño de los algoritmos evaluados en el DSP. También se presenta el análisis y discusión de resultados obtenidos, los que son presentados principalmente mediante gráficas BER vs. SNR como medida de desempeño apropiada en los sistemas de comunicación digital. Finalmente se presentan las conclusiones obtenidas a lo largo del presente trabajo de investigación de acuerdo con los objetivos marcados.

---

# Agradecimientos

---

El presente trabajo esta dedicado principalmente, a la persona que ha dado todo por mí en esta vida, a quien le debo no sólo mi tesis o mi carrera, sino toda mi vida, con todo mi cariño te dedico mi éxito madre mía. Gracias mamá por apoyarme y comprenderme a cada instante durante mis años de carrera.

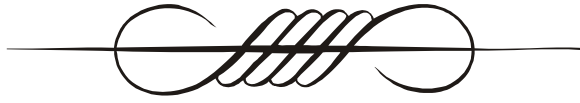
Quiero agradecer con todo mi corazón a mis hermanos, que más que ser una familia para mí, han sido el pilar que nunca me dejó caer. Gracias por el apoyo y la confianza depositada en cada una de mis decisiones, no habría logrado lo que hoy en día soy sin ustedes.

Quiero expresar un profundo agradecimiento a mis asesores Dr. Gerardo Abel Laguna Sánchez y Dr. Alfonso Prieto Guerrero que me apoyaron, guiaron y compartieron sus conocimientos durante la elaboración del presente trabajo de investigación.

Agradezco al Consejo Nacional de Ciencia y Tecnología, por el financiamiento otorgado durante los años de estudio de mi posgrado.

A la Universidad Autónoma Metropolitana unidad Iztapalapa, por las facilidades prestadas para el desarrollo de este trabajo.

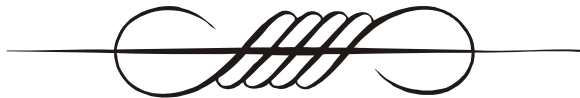
A mis profesores que me han compartido sus conocimientos durante cada clase y aún fuera de ellas. Y por supuesto, a mis compañeros de la MCyTI, con quienes compartí momentos de estudio y de diversión.



*Si buscas resultados diferentes, no hagas siempre lo mismo.  
... Albert Einstein ...*

*Naturaleza, por siempre fuente de inspiración y conocimiento, dichoso quien al mirarla  
reciba el don del saber, y comparta con aquellos que le hemos olvidado.*

*... Daniel Torres ...*



# Contenido

---

|  |          |
|--|----------|
| Resumen  | I        |
| Lista de figuras   | VIII     |
| Lista de tablas  | X        |
| Lista de acrónimos   | XII      |
| <b>1. Introducción</b>   | <b>1</b> |
| 1.1. Planteamiento del problema . . . . .                      | 1        |
| 1.2. Objetivos del proyecto . . . . .                          | 3        |
| 1.3. Metodología de investigación . . . . .                    | 3        |
| 1.4. Contribución . . . . .                                    | 4        |
| 1.5. Estructura del documento . . . . .                        | 4        |
| <b>2. Marco teórico</b>  | <b>5</b> |
| 2.1. Antecedentes . . . . .                                    | 5        |
| 2.2. Tecnología PLC . . . . .                                  | 6        |
| 2.2.1. Historia . . . . .                                      | 6        |
| 2.2.2. Modelo del canal PLC . . . . .                          | 7        |
| 2.2.3. Transmisión en un canal PLC . . . . .                   | 10       |
| 2.2.4. Estándar para la tecnología PLC . . . . .               | 10       |
| 2.3. Fundamentos básicos de OFDM . . . . .                     | 12       |
| 2.3.1. Datos históricos sobre OFDM . . . . .                   | 13       |
| 2.3.2. Modulación OFDM . . . . .                               | 15       |
| 2.3.3. Principio de ortogonalidad . . . . .                    | 15       |
| 2.3.4. Transmisión de una señal OFDM . . . . .                 | 17       |
| 2.3.5. Intervalo de guarda . . . . .                           | 20       |
| 2.3.6. Símbolo piloto . . . . .                                | 20       |
| 2.4. Proceso de sincronización y estimación de canal . . . . . | 21       |
| 2.4.1. Sincronización . . . . .                                | 21       |
| 2.4.2. Estimación del canal . . . . .                          | 24       |
| 2.5. Modelo de ruido . . . . .                                 | 25       |

|   |           |
|---|-----------|
| 2.5.1. Canal AWGN . . . . .   | 25        |
| 2.6. Kit de desarrollo para el DSP . . . . .                            | 26        |
| 2.6.1. Procesador digital de señales . . . . .                          | 26        |
| 2.6.2. Code Composer Studio (CCS) . . . . .                             | 28        |
| 2.7. Método Monte Carlo . . . . .                                       | 29        |
| <b>3. Programación y evaluación</b>                                     | <b>30</b> |
| 3.1. Arquitectura general del sistema de prueba . . . . .               | 30        |
| 3.1.1. Información fuente . . . . .                                     | 31        |
| 3.1.2. Transmisor . . . . .   | 32        |
| 3.1.3. Construcción de tramas OFDM . . . . .                            | 36        |
| 3.1.4. Canal . . . . .  | 38        |
| 3.1.5. Puesta a punto . . . . .   | 39        |
| 3.1.6. Receptor . . . . .   | 49        |
| 3.1.7. Contador de errores . . . . .                                    | 51        |
| 3.2. Simulación Monte Carlo . . . . .                                   | 52        |
| 3.3. Plataforma de implementación DSK . . . . .                         | 54        |
| <b>4. Pruebas y resultados</b>  | <b>55</b> |
| 4.1. Planteamiento experimental . . . . .                               | 55        |
| 4.2. Pruebas y resultados individuales . . . . .                        | 56        |
| 4.2.1. Generador de la información fuente . . . . .                     | 57        |
| 4.2.2. Formateo antipodal . . . . .                                     | 57        |
| 4.2.3. Modulación/demodulación QAM . . . . .                            | 58        |
| 4.2.4. Igualador . . . . .  | 59        |
| 4.2.5. Modulación OFDM . . . . .  | 59        |
| 4.2.6. Inserción del intervalo de guarda . . . . .                      | 60        |
| 4.2.7. Sincronización burda . . . . .                                   | 61        |
| 4.2.8. Sincronización fina . . . . .                                    | 62        |
| 4.3. Evaluación global . . . . .  | 63        |
| 4.3.1. Pruebas y resultados del canal AWGN . . . . .                    | 64        |
| 4.3.2. Pruebas y resultados del canal selectivo en frecuencia . . . . . | 67        |
| 4.3.3. Pruebas y resultados de la técnica DMT en un canal PLC . . . . . | 69        |
| <b>5. Conclusiones y trabajo futuro</b>                                 | <b>73</b> |
| 5.1. Conclusiones . . . . .   | 73        |
| 5.2. Trabajo futuro . . . . .   | 75        |
| <b>Referencias</b>  | <b>76</b> |
| <b>A. Glosario</b>  | <b>79</b> |
| <b>B. Secuencia compleja</b>  | <b>82</b> |

---



---

|  |           |
|--|-----------|
| <b>C. Algoritmos</b>                                     | <b>84</b> |
| C.1. Transmisor/Receptor OFDM (trama de datos) . . . . . | 84        |
| C.2. Generador de símbolos OFDM . . . . .                | 88        |
| C.3. Proceso OFDM inverso . . . . .                      | 90        |
| C.4. Modulación DMT . . . . .                            | 91        |
| C.5. Demodulación DMT . . . . .                          | 92        |
| C.6. Estimación VR . . . . .                             | 93        |
| C.7. Proceso de sincronización . . . . .                 | 95        |

---

# Lista de figuras

---

|       |  |    |
|-------|--|----|
| 2.1.  | Modelo general de un canal PLC. . . . .  | 8  |
| 2.2.  | Respuesta en frecuencia típica de un canal PLC. . . . .                          | 8  |
| 2.3.  | Esquema del sistema eléctrico. . . . .   | 9  |
| 2.4.  | Transmisión de datos en la última pulgada. . . . .                               | 10 |
| 2.5.  | Desvanecimiento selectivo en frecuencia. . . . .                                 | 13 |
| 2.6.  | Comparación entre ancho de banda requerido para (a) FDM, y (b) OFDM. . . . .     | 14 |
| 2.7.  | Interferencia en tiempo (ISI) y en frecuencia (ICI). . . . .                     | 16 |
| 2.8.  | Señales sinusoidales ortogonales. . . . .  | 16 |
| 2.9.  | Función rectángulo. . . . .  | 17 |
| 2.10. | Portadoras ortogonales. . . . .  | 18 |
| 2.11. | Inserción del intervalo de guarda. . . . .                                       | 20 |
| 2.12. | Esquema OFDM asistido por piloto. . . . .  | 21 |
| 2.13. | Correlación $\Psi(\delta_t)$ . . . . .   | 23 |
| 2.14. | Corrección de fase. . . . .  | 24 |
| 2.15. | Efecto de un canal AWGN, a) señal original, b) señal afectada por ruido. . . . . | 26 |
| 2.16. | Clasificación del formato aritmético. . . . .                                    | 27 |
|       |  |    |
| 3.1.  | Esquema a bloques del sistema de prueba. . . . .                                 | 31 |
| 3.2.  | Diagrama general del transmisor. . . . .   | 34 |
| 3.3.  | Formateo antipodal. . . . .  | 34 |
| 3.4.  | Constelación 4QAM. . . . .   | 35 |
| 3.5.  | Trama de control. . . . .  | 37 |
| 3.6.  | Trama OFDM de datos. . . . .   | 38 |
| 3.7.  | Canal selectivo en frecuencia empleado. . . . .                                  | 39 |
| 3.8.  | Esquema general del módulo 'Puesta a punto'. . . . .                             | 40 |
| 3.9.  | Diagrama a bloques de la sincronización burda. . . . .                           | 41 |
| 3.10. | Demodulación OFDM. . . . .   | 44 |
| 3.11. | Etapas de sincronización fina. . . . .   | 45 |
| 3.12. | Comparación del proceso de estimación de canal. . . . .                          | 47 |
| 3.13. | Modulos del receptor. . . . .  | 50 |
| 3.14. | Simulación Monte Carlo. . . . .  | 53 |
|       |  |    |
| 4.1.  | Escenario de simulación. . . . .   | 55 |

---

|       |   |    |
|-------|---|----|
| 4.2.  | Escenario real de evaluación. . . . .   | 56 |
| 4.3.  | Secuencia binaria de tipo "R". . . . .  | 57 |
| 4.4.  | Secuencia antipodal de tipo "A" (a) y "R" (b). . . . .                                      | 57 |
| 4.5.  | Constelación QAM y los efectos de un canal AWGN. . . . .                                    | 58 |
| 4.6.  | Igualación de la señal analítica. . . . .   | 58 |
| 4.7.  | Símbolo piloto OFDM del sistema de referencia (a) y del DSP (b). . . . .                    | 59 |
| 4.8.  | Símbolo OFDM con su intervalo de guarda. . . . .  | 60 |
| 4.9.  | Símbolo OFDM con intervalo de guarda alterado por un canal AWGN. . . . .                    | 60 |
| 4.10. | Correlación GI con canal AWGN, simulación por computadora (Matlab). . . . .                 | 61 |
| 4.11. | Correlación GI con canal AWGN en el DSP. . . . .  | 61 |
| 4.12. | Operación angular (a) y corrección de fase (b) en el DSP. . . . .                           | 63 |
| 4.13. | Adelanto de la ventana FFT. . . . .   | 63 |
| 4.14. | Retardo de la ventana FFT. . . . .  | 64 |
| 4.15. | Evaluación en el DSP vs. PC (lenguaje C) aplicando estimación LS. . . . .                   | 65 |
| 4.16. | Evaluación por simulación por computadora (Matlab y C) vs. DSP de la estimación VR. . . . . | 66 |
| 4.17. | Resultados del DSP de la estimación LS vs. VR. . . . .                                      | 67 |
| 4.18. | Comparación de resultados ante la curva teórica QAM. . . . .                                | 67 |
| 4.19. | Gráficas BER del desempeño del DSP en un canal PLC vs. AWGN usando estimación LS. . . . .   | 68 |
| 4.20. | Gráficas BER del desempeño del DSP en un canal PLC vs. AWGN usando estimación VR. . . . .   | 69 |
| 4.21. | Gráficas BER de la evaluación de un canal PLC con estimación LS vs. VR. . . . .             | 69 |
| 4.22. | Evaluación de la técnica DMT usando la estimación LS. . . . .                               | 70 |
| 4.23. | Evaluación de la técnica DMT usando la estimación VR. . . . .                               | 71 |
| 4.24. | Estimación de canal LS y VR sin técnica DMT. . . . .  | 71 |

---

# Lista de tablas

---

|   |    |
|---|----|
| 2.1. Declaración de parámetros OFDM. . . . .  | 18 |
| 3.1. Notación del sistema de prueba. . . . .  | 32 |
| 3.2. Secuencia binaria del símbolo tipo "A". . . . .  | 33 |
| 4.1. Estimación burda por correlación GI para un desfase de 666 muestras (señal analítica). . . . . | 62 |
| 4.2. Parámetros iniciales del sistema de prueba. . . . .  | 64 |
| B.1. Secuencia compleja de un canal selectivo en frecuencia. . . . .                                | 82 |

# Lista de algoritmos

---

|       |  |    |
|-------|--|----|
| 3.1.  | Construcción de la secuencia binaria del símbolo piloto "A". . . . .         | 33 |
| 3.2.  | Modulador 4QAM. . . . .  | 34 |
| 3.3.  | Simetría Hermitiana. . . . .   | 35 |
| 3.4.  | Intervalo de guarda. . . . .   | 36 |
| 3.5.  | Canal AWGN. . . . .  | 38 |
| 3.6.  | Correlación GI. . . . .  | 42 |
| 3.7.  | Actualización del vector de correlación. . . . .                             | 43 |
| 3.8.  | Estimación MC. . . . .   | 43 |
| 3.9.  | Demodulación OFDM. . . . .   | 44 |
| 3.10. | Operación ángulo. . . . .  | 46 |
| 3.11. | Estimación LLS para sincronización fina. . . . .                             | 46 |
| 3.12. | Estimación LS. . . . .   | 48 |
| 3.13. | Estimación por ventaneo rectangular de la respuesta al impulso (VR). . . . . | 48 |
| 3.14. | Máscara DMT. . . . .   | 49 |
| 3.15. | Igualador. . . . .   | 50 |
| 3.16. | Demodulación y detección QAM. . . . .  | 51 |
| 3.17. | Contador de errores. . . . .   | 52 |

# Lista de acrónimos

---

|             |   |
|-------------|---|
| <b>BER</b>  | Bit Error Rate.                             |
| <b>bps</b>  | bits por segundo.                           |
| <b>CCS</b>  | Code Composer Studio.                       |
| <b>DFT</b>  | Discrete Fourier Transform.                 |
| <b>DMT</b>  | Discrete Multitone.                         |
| <b>DSK</b>  | DSP Starter Kit.                            |
| <b>DSP</b>  | Digital Signal Processing.                  |
| <b>FDM</b>  | Frequency Division Multiplexing.            |
| <b>FFT</b>  | Fast Fourier Transform.                     |
| <b>GI</b>   | Guard Interval.                             |
| <b>ICI</b>  | Inter Channel Interference.                 |
| <b>IDE</b>  | Integrated Development Environment.         |
| <b>ISI</b>  | Inter-Symbol Interference.                  |
| <b>ISO</b>  | International Standards Organization.       |
| <b>LAN</b>  | Local Area Network.                         |
| <b>LS</b>   | Least Square.                               |
| <b>LLS</b>  | Linear Least Square.                        |
| <b>Mbps</b> | Mega bits por segundo.                      |
| <b>MC</b>   | Maximum Correlation.                        |
| <b>ML</b>   | Maximum Likelihood.                         |
| <b>MMSE</b> | Minimum Mean Squared Error.                 |
| <b>NRZ</b>  | Non Return to Zero.                         |
| <b>OFDM</b> | Orthogonal Frequency Division Multiplexing. |
| <b>OSI</b>  | Open System Interconnection.                |
| <b>PLC</b>  | Power Line Communication.                   |
| <b>QAM</b>  | Quadrature Amplitude Modulation.            |
| <b>SCs</b>  | Subcarriers.                                |
| <b>SNR</b>  | Signal to Noise Ratio.                      |
| <b>VCO</b>  | Voltage Controlled Oscillator.              |

---

# Capítulo 1

## Introducción

---

### 1.1. Planteamiento del problema

El hombre, a lo largo de la evolución, se ha visto acompañado por los sistemas de comunicación, desde los más simples como los gestos corporales, hasta las modernas técnicas de comunicación digital, que hoy en día siguen evolucionando. En la actualidad, las comunicaciones se han vuelto tan comunes que ya no se percibe como fueron cambiando nuestra vida cotidiana.

Los grandes avances en las tecnologías de comunicación, se han visto acompañados de una gran demanda y crecimiento del número de usuarios, propiciando la investigación de nuevas técnicas para satisfacer las nuevas exigencias de hoy en día, como el estar comunicados e informados en cualquier punto del planeta. No sólo se desea omnipresencia de las telecomunicaciones, sino que también se requiere que el intercambio de información sea lo más económico, que el tiempo para enlazarnos sea el mínimo posible y que la cantidad de información transmitida sea lo más cercano posible al límite teórico.

Los ingenieros e investigadores de las comunicaciones electrónicas y áreas afines, no sólo se encontraron con el problema de una gran cantidad de usuarios, sino también con las dificultades que la naturaleza, y el mismo hombre generan. Esta última alusión va dirigida al ruido, que constituye el problema central para lograr una transmisión eficiente de la información. Además se tienen nuevos retos como son: transmisión de datos a tasas elevadas, reducción en la inversión monetaria en infraestructura, incremento de ancho de banda, y sin olvidar el compromiso de garantizar un desarrollo sustentable, que busca disminuir la contaminación ambiental minimizando el consumo de energía.

El sistema de distribución de la energía eléctrica, es un medio omnipresente, a todo lo largo y ancho de cada ciudad o país. El sistema eléctrico forma una red que se compone de miles de hogares, empresa, instituciones públicas, por mencionar algunas. Esta red, que podría ser la más grande a nivel mundial, ha dado origen a una nueva tecnología de transmisión de datos conocida como *Power Line Communication (PLC)*, la cual es una área de investigación muy atractiva y potencialmente conveniente para el envío de datos a tasas

elevadas sobre una infraestructura ya existente.

En esencia, la tecnología PLC representa para los ingenieros de comunicaciones el formidable reto de transmitir datos a través de un medio que fue originalmente diseñado para la distribución de energía eléctrica y no para la transmisión de datos. Los investigadores e ingenieros deben considerar algunas características de este medio que lo hacen en cierta forma uno de los peores canales de comunicación, como: la presencia de ruido impulsivo, ser un canal selectivo en frecuencia y variante en el tiempo, esto por mencionar algunas características [2].

La tecnología PLC está en vías de desarrollo y los últimos años ha ganado terreno en el mercado. El crecimiento que ha tenido se debe a que representa una alternativa económica para los consumidores de Internet. Esta tecnología se ha probado en algunos países como España, Alemania, Francia, entre otros países comprometidos con el servicio de la telecomunicación. No sólo el sector público se ha puesto en el camino de explotar esta prometedora tecnología, también la inversión privada está interesada en el desarrollo y avance tecnológico de las comunicaciones digitales por medios alternos y económicamente atractivos.

La norma IEEE P1901.2 para tecnología PLC, propone como estándar de la capa física el uso del esquema de modulación OFDM (sigla del ingl. *Orthogonal Frequency Division Multiplexing*, 'multiplexado por división de frecuencias ortogonales'). La transmisión bajo el esquema OFDM es ideal para minimizar los efectos de atenuación selectiva, ruido, y al mismo tiempo maximizar el aprovechamiento del canal [3].

Los procesadores digitales de señales o DSP (sigla del ingl. *Digital Signal Processing*), hoy por hoy, representan una herramienta poderosa para la realización y evaluación de algoritmos de aplicación específica con gran demanda de procesamiento. En el presente proyecto se propone la puesta en práctica y evaluación de los algoritmos para la transmisión de datos bajo el esquema OFDM en un DSP TMS320C6416T de la compañía *Texas Instrument*. Los modelos formales del esquema OFDM para un canal PLC como estimación de canal, sincronización y algunos otros modelos que han sido puestos en práctica durante este proyecto de investigación, fueron propuestos por Matthieu Crussière [1], para un canal PLC utilizando el esquema OFDM.

Es importante mencionar que la investigación realizada en el presente documento, permitió proponer una mejora en el proceso de sincronización. Dicha mejora se expone en el capítulo 3, también se propone un diseño de la trama utilizada en la puesta a punto de un sistema de comunicación. El diseño de la trama de puesta a punto fue diseñado para disminuir el tiempo requerido en el proceso de sincronización.

En el presente trabajo se propone el uso de técnicas de estimación de canal y de ruido, dichas técnicas permiten compensar los daños que sufre cualquier señal sobre un canal PLC. Las técnicas propuestas son estimación por mínimos cuadrados (de *LS*, sigla del ingl. *Least*

---



---

*Square*) y eliminación de ruido por ventaneo rectangular (VR) en el dominio temporal.

## 1.2. Objetivos del proyecto

### Objetivo general

Como objetivo general, se plantea diseñar y programar en un DSP los algoritmos de la capa física de un módem OFDM para un enlace PLC.

### Objetivos específicos

- Diseñar y programar los módulos de modulación y demodulación OFDM para un DSP.
- Diseñar y programar el módulo de sincronización para un DSP.
- Diseñar y programar los módulos de estimación de canal y estimación de ruido para un DSP .
- Diseñar y programar el módulo de igualamiento para un DSP .
- Evaluar el desempeño de los algoritmos seleccionados en el DSP.

## 1.3. Metodología de investigación

Para lograr satisfactoriamente los objetivos antes mencionados se siguió una metodología de tres etapas:

- **Estudio del estado del arte.**
    - o Estudiar los antecedentes, estándares, así como investigación e inversión sobre la tecnología PLC.
    - o Investigar los algoritmos de la capa física de un módem OFDM.
    - o Investigar el funcionamiento y manejo adecuado de la tarjeta de desarrollo para el DSP.
    - o Investigar sobre las características del DSP, que permitan el óptimo uso de éste.
  - **Programación de los algoritmos básicos de la capa física del módem OFDM.**
    - o Diseño del esquema de comunicación de prueba.
    - o Simulación por computadora del esquema de comunicación.
-

- o Programar los algoritmos básicos de la capa física del módem OFDM en el DSP.
- **Evaluar el desempeño de la ejecución del conjunto de los algoritmos seleccionados en el DSP.**

## 1.4. Contribución

Después de seguir la metodología indicada arriba se tienen los siguientes aportes:

- Visión general del estado de la tecnología PLC hasta el año 2012.
- Puesta en práctica de los algoritmos de la capa física para un módem PLC para el esquema OFDM en aritmética de punto fijo en el DSP TMS032C6416T de *Texas Instruments*.
- Puesta en práctica y adecuación de la técnica de sincronización del modelo original de Matthieu Crussière.

## 1.5. Estructura del documento

El resto del documento se ha organizado de la siguiente forma: en el capítulo dos, se presenta la revisión bibliográfica sobre la que está basado el aporte del trabajo, se exponen temas como: antecedentes de la tecnología PLC, normas que definen el estándar de la capa física de un módem OFDM, modelos formales del esquema OFDM, sincronización, estimación de canal y de ruido. También se presenta de forma breve las herramientas de software y hardware empleadas para lograr satisfactoriamente los objetivos propuestos.

En el capítulo tres se describe la propuesta de este trabajo de investigación, donde se propone una arquitectura de prueba para un sistema de comunicación.

En el capítulo cuatro se presentan los resultados obtenidos de la evaluación de los algoritmos de la capa física de un módem PLC. Se encuentran también los resultados del sistema para diferentes escenarios de prueba.

Finalmente, el capítulo cinco se presentan las conclusiones del proceso de investigación y desarrollo de este proyecto, de acuerdo a los objetivos propuestos.

---

---

## Capítulo 2

# Marco teórico

---

En este capítulo se presentan algunos datos históricos, características y estándares referentes a la tecnología PLC, así como organizaciones públicas y privadas que investigan y han puesto en práctica la transmisión de datos en un canal PLC. Posteriormente se presentan los modelos formales del esquema OFDM y dos aspectos básicos para éste: la técnica de sincronización y estimación de canal. Los métodos aquí presentados fueron propuestos por Matthieu Crussière [1] y estudiados en trabajos relacionados. Finalmente, se presenta una breve descripción de las herramientas de programación e implementación, requeridas para desarrollar este proyecto.

### 2.1. Antecedentes

En el proceso de la investigación de la frontera del conocimiento, se realizó una búsqueda de textos científicos en diversos temas relacionados con este proyecto. Tales temas son: trabajos en una serie de artículos como [4–6], donde ponen en práctica en un DSP los algoritmos del esquema OFDM y son evaluados bajo una simulación de un canal AWGN (sigla del ingl. *Additive White Gaussian Noise*, 'ruido blanco Gaussiano aditivo'). En el caso del trabajo en [6] proponen una técnica para optimizar la estimación de los niveles de la SNR (sigla del ingl. *Signal to Noise Ratio*, 'razón señal a ruido').

Dado que en el presente trabajo se propuso implementar la modulación DMT (sigla del ingl. *Discrete Multitone*, 'multitono discreto'), se realizó una búsqueda corta sobre el uso de la técnica de modulación DMT en un canal PLC, esto con el fin de conocer la mejoría que aporta en una transmisión bajo el esquema OFDM de acuerdo a [7]. También se encontró investigación e implementación utilizando el DSP TMS032C6416T de las técnicas DMT con simulación de un canal PLC [8].

A pesar no de tener como objetivo en este trabajo, el proponer la mejora de alguno de los algoritmos que se encuentran en el módem para un enlace PLC, se realizó una revisión en [9, 10] donde proponen técnicas de sincronización para el esquema OFDM.

No se puede pasar por alto los puntos de vista e ideas más recientes referente a la evolución

de la tecnología PLC. Por ello buscamos escritos científicos que tratan sobre lo atractivo de esta novedosa tecnología, como lo mencionan en [11], o la tasa de transferencia que se puede lograr como lo comentan en [12] entre otras interesantes opiniones y aportes encontrados.

Los trabajos antes mencionados, fueron estudiados con el fin de conocer diversas técnicas e ideas, que pudieran ser implementadas en este trabajo de investigación. Y con ello mejorar el desempeño de los algoritmos aquí propuestos para simular de la capa física de un módem OFDM/PLC sobre un DSP.

## 2.2. Tecnología PLC

La tecnología *Power Line Communication* (PLC) o comunicación a través de la red eléctrica, es el nombre que se ha dado a la transmisión de datos en banda estrecha o en banda ancha, usando como medio físico de transmisión a la red eléctrica. El objetivo de usar un enlace PLC consiste en aprovechar la infraestructura existente de la red eléctrica para transmitir señales de baja frecuencia (50 ó 60 Hz, según el país), encargadas de la distribución de potencia, y señales de mayor frecuencia para los datos, ambas circulando simultáneamente a través del mismo cable de cobre.

De acuerdo al país, organización o compañía, el nombre de esta tecnología puede cambiar. Algunos acrónimos que hacen referencia a esta tecnología, y que puede encontrarse en la literatura son:

- BPL (*Broadband over Power Line*)
- PLC (*Power Line Communication*)
- PLT (*Power Line Telecommunication*)
- PPC (*Power Plus Communication*)

El uso de las líneas para la transmisión de datos no es algo novedoso, desde hace unos años se han usado las líneas eléctricas para la transmisión de datos. Lo novedoso de esta tecnología es la tasa de transferencia lograda hasta hoy en día de 1Gbps en una red de área local [13].

### 2.2.1. Historia

Las líneas eléctricas forman una red universal y quizá la red cableada más amplia del mundo. Su propósito original, no fue la comunicación, el primer intento de comunicación en las líneas eléctricas, fue transmitir datos de mediciones remotas. De acuerdo a N. Pavlidou en [3], en 1838 se dio la primera medición remota en este canal. Algunos otros datos históricos encontrados durante la revisión literaria de [3, 14–16] son:

---

- En 1887 se propuso en el Reino Unido la primera patente en la señalización sobre PLC.
- En 1905 algunas aplicaciones sobre el canal PLC fueron patentadas en Estados Unidos.
- En 1913 se comenzó la producción de medidores electromecánicos para el consumo energético.
- En 1980 técnicas sofisticadas de control de errores y de codificación fueron propuestas dentro de un módem de PLC con transmisión en una sola dirección.
- En 1997 se realizaron las primeras pruebas de transmisión bidireccional, a través de las líneas eléctricas por parte de Ascom (Suiza) y Norweb (Reino Unido).
- En el 2000 se realizaron pruebas en Francia por el grupo EDF R&D y Ascom, donde implementaron un sistema de control para llenado de un estanque de agua.
- En el 2005 se aprobó el proyecto para estandarizar la tecnología PLC en los Estados Unidos de América, reconociendo dos propuestas de estándar: *HomePlug* y HD-PLC.
- En el 2009 se aprobó el estándar *HomePlug*, donde estipula detalles técnicos de la capa física en el documento TSG2, como el uso de la FFT (sigla del ingl. *Fast Fourier Transform*, 'transformada rápida de Fourier') para el esquema OFDM.
- En el 2010 la IEEE (sigla del ingl. *Institute of Electrical and Electronics Engineers*, 'Instituto de ingenieros eléctricos y electrónicos') hizo público el estándar generado por HomePlug y HD-PLC en el documento: P1901-2010 IEEE '*Standard for Broadband over Power Line Networks: Medium Access Control and Physical Layer Specifications*'.

La evolución en los últimos años, de las técnicas de transmisión en canales PLC y estándares de codificación y modulación, han contribuido e impulsado el crecimiento de las comunicaciones digitales en canales PLC, lo que promete nuevos beneficios de las líneas eléctricas, como la conectividad de Internet.

Hoy en día la transmisión de datos sobre el cableado eléctrico ya es un hecho, por lo que el reto es mejorar la tasa de transmisión, la cual supera ya los 200Mbps en exterior y hasta 1Gbps en interiores.

### 2.2.2. Modelo del canal PLC

Los canales PLC representan un medio nefasto para la transmisión de datos, debido a que presentan desventajas como ser variante en el tiempo, selectivo en frecuencia y contaminado por diversas formas de ruido. Un canal PLC, de acuerdo a N. Pavlidou [3], es considerado como un canal mucho peor que el de las comunicaciones móviles.

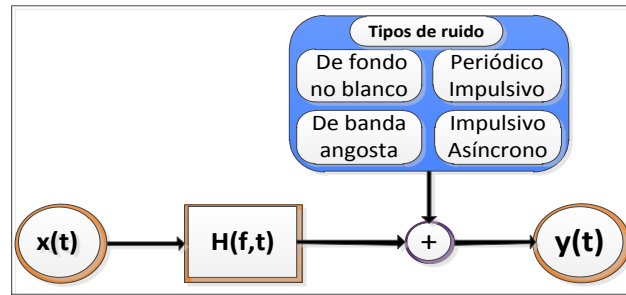


Figura 2.1: Modelo general de un canal PLC.

Al tener tantos factores adversos el modelo del canal PLC se hace muy complejo. El modelo general más usado, es el que se muestra en la figura 2.1 [3, 17, 18].

El modelo del canal eléctrico generalmente se realiza con base en estudios de la función de transferencia (empírico o determinista [18]) y del ruido aditivo. Debido a que existen diversos enfoques para modelar la función de transferencia de este canal, en [3] se realiza la siguiente clasificación:

- **Enfoque a nivel de Hardware.** Esta clasificación se realiza con base en la impedancia de los cables y la topología de la red.
- **Enfoque de comunicación.** Aquí se modela la atenuación, cambio de fase o fuentes de ruido, factores de pérdida y más.

En la figura 2.2 se observa el comportamiento típico de la función de transferencia de un canal PLC.

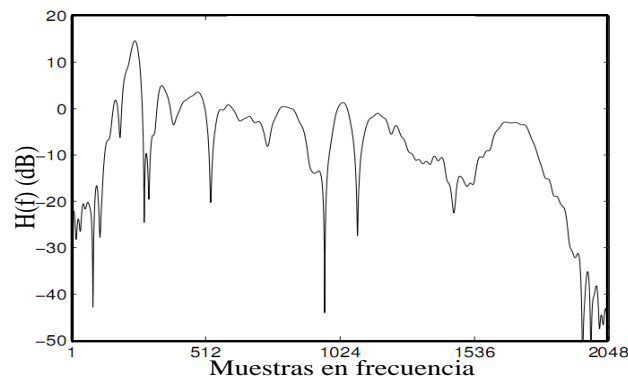


Figura 2.2: Respuesta en frecuencia típica de un canal PLC.

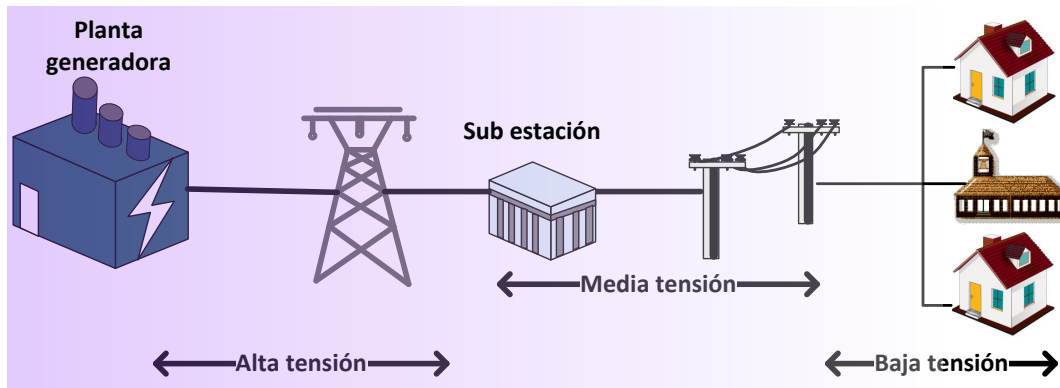


Figura 2.3: Esquema del sistema eléctrico.

Como se mencionó anteriormente, en un sistema PLC se modela el comportamiento de ruido y, de acuerdo a [3], en un canal PLC se presentan básicamente 4 tipos de ruido (además del ruido blanco Gaussiano) que son objeto de estudio, véase la figura 2.1:

- **Ruido de fondo no blanco (rosa o coloreado).** Es el resultado de sumar todas las fuentes de ruido de baja potencia.
- **Ruido periódico impulsivo.** Son señales periódicas de corta duración y gran potencia. Por ejemplo fuentes de poder conmutadas.
- **Ruido de banda angosta.** Se compone por señales sinusoidales con modulación en amplitud (estaciones de radio, señales de televisión, etc.)
- **Ruido impulsivo asíncrono.** Ruido producido en el cambio de estado de un sistema, como puede ser el poner en marcha algún aparato eléctrico, que genera señales de corta duración y alta potencia.

Existen otros modelos para representar y caracterizar a un canal PLC: métodos matemáticos, estrategias con base en teoría de líneas de transmisión y algunos otros que se realizan de forma práctica, como mediciones reales de ruido en distintos nodos para caracterizarlo, y así tratar de eliminar sus efectos mediante algoritmos de filtrado adaptativo [19].

Algunas mediciones realizadas en el canal PLC, muestran que la atenuación del canal se incrementa a altas frecuencias. El canal se describe como aleatorio y variante en el tiempo, con una frecuencia que depende del valor de SNR sobre la transmisión de banda ancha [17].

El eco resulta ser una característica más de un enlace PLC, este fenómeno produce un efecto de multi-trayectoria<sup>1</sup> y lleva consigo el problema de generar interferencia inter símbo-

<sup>1</sup>En un canal PLC, el efecto de multi-trayectoria se debe a malos acoplamientos de impedancia (desacoplamiento). En la comunicación inalámbrica, donde la señal de propagación entre emisor y receptor, no se da en línea de vista (de *LoS*, sigla del ingl. *Line of Sight*), aparece el efecto de multi-trayectoria.

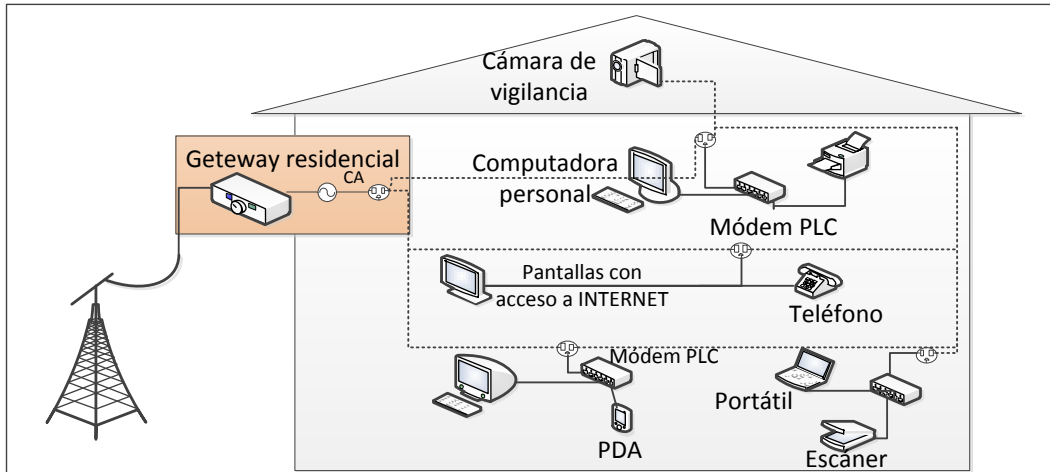


Figura 2.4: Transmisión de datos en la última pulgada.

lo (de *ISI*, sigla del en ingl. *Inter-Symbol Interference*). Como se ve más adelante en este capítulo, existen técnicas de modulación que permiten contrarrestar este efecto tanto en el canal inalámbrico como el canal PLC.

### 2.2.3. Transmisión en un canal PLC

Hoy en día se transmiten datos sobre redes de baja y media tensión del sistema eléctrico, como parte de la última milla, pero pudiera ser un enlace inalámbrico (en la figura 2.3 se aprecia el esquema del sistema eléctrico). Por otro lado, el avance tecnológico en la redes de área local (de *LAN*, sigla del ingl. *Local Area Network*), ya sean cableadas o no, dio origen al término de conexión en la "última pulgada" (*last-inch access* o *in-home networking*), el cual se aprecia en la figura 2.4 [17].

En realidad, la tecnología PLC no puede estar aislada ni en contienda con tecnologías de radio frecuencia o cableadas. Hoy en día las redes deben tener la tendencia de convivir en una gran red heterogénea.

### 2.2.4. Estándar para la tecnología PLC

Debido a que la comunicación sobre un canal PLC es aún una tecnología en vías de desarrollo, no cuenta con un estándar universalmente aceptado. Esto dio pauta a que organizaciones internacionales iniciaran el trabajo para generar un estándar adecuado. A continuación se muestran las propuestas de cada organización encargada de los estándares en comunicación.

#### a) IEEE:



- 1) **IEEE P1675.** "*Standard for Broadband over Power line Hardware*". Este trabajo contiene información relacionada con la instalación y seguridad en el hardware empleado en líneas de energía.
  - 2) **IEEE P1775.** "*Power line Communication Equipment - Electromagnetic Capability (EMC) Requirements - Testing and Measurement Methods*". Este trabajo está dedicado a los dispositivos físicos del enlace PLC, requerimientos para la compatibilidad electromagnética, así como los métodos de pruebas y mediciones.
  - 3) **IEEE P1901.** "*Draft Standard for Broadband over Power Line Networks: Medium Access Control and Physical Layer Specifications*". Este trabajo es un bosquejo de los protocolos de acceso al medio y especificaciones necesarias de la capa física de un módem OFDM para un enlace PLC [15].
- b) **OPERA** (*Open PLC European Research Alliance*), proyecto cofinanciado por el 6º Programa en el Marco de la Comisión Europea (área de tecnologías de la información, IST), que comenzó en enero de 2004 y concluyó en febrero de 2007. OPERA participó ampliamente en proceso de estandarización con organizaciones como IEEE y IEC (International Electrotechnical Commission), y ha contribuido en la norma IEEE P1901. Aportes como (por mencionar algunos): [20]
- 1) **Nuevos equipos y prototipos:** OPERA llevó a cabo las actividades relacionadas con los aspectos físicos del medio de transmisión y los componentes periféricos necesarios para desplegar una red PLC.
  - 2) **Nuevos servicios:** Estos servicios son por ejemplo acceso a Internet, servicios de telefonía (VoIP), video distribución de la señal (IPTV), la vigilancia y de gestión de energía, redes eléctricas inteligentes de vigilancia y control.

Algunas otras organizaciones también han colaborado en el proceso de estandarización como son: **POWERNET**, **UPA** (*Universal Powerline Association*), **CEPCA** (*Consumer Electronics Powerline Communications Alliance*), **ETSI PLC**, entre otras.

Una más es **HomePlug Power Line Alliance**, que se conforma por un grupo de más de 65 compañías de todo el mundo que realizan investigación y manufactura de sistemas de comunicación como Comcast, GE, Intel, Linksys, Motorola, Radio Shack, Samsung, Sharp y Sony por mencionar algunas. La tecnología de *HomePlug* se desarrolla con base en las contribuciones de cada una de estas compañías, y ha definido algunos estándares para las comunicaciones PLC como: *HomePlug 1.0*, *HomePlug AV*, *HomePlug BPL*, *HomePlug CC* [15].

El estándar *HomePlug 1.0*, detalla los aspectos relacionados con la capa física de un módem OFDM para PLC, tales como modulación OFDM, número de subportadoras, frecuencias sobre las que trabaja la señal de datos PLC entre otros aspectos técnicos. Para más información el lector podrá consultar [16].

En el año 2010 se elaboró el primer estándar oficial propuesto por la IEEE, en colaboración con las instituciones y alianzas antes mencionadas. La norma lleva el nombre: **IEEE P1901-2010**, pero aún se sigue trabajando en la definición de este estándar. En el 2012 el avance logrado con respecto a la capa física se encuentra en la norma IEEE P1901.2, la cual se caracteriza por:

- Especificar comunicaciones de baja frecuencia (menos de 500 kHz) o banda estrecha en PLC.
- Garantizar compatibilidad con las comunicaciones interiores (indoor) y exteriores (outdoor) sobre la línea de baja tensión (entre el transformador y línea de metro, a menos de 1000 V), a través del transformador de baja tensión a media tensión (1000 V hasta 72 kV).
- Asegurar la coexistencia con la línea de banda ancha de potencia ( de *BPL*, sigla del ingl. *Broadband Power Line*) dispositivos mediante la minimización de fuera de banda de las emisiones en las frecuencias superiores a 500 kHz.
- Abordar los requisitos de seguridad necesarios que garanticen la privacidad de las comunicaciones y permitir el uso de los servicios de seguridad sensibles.
- Proponer la capa de enlace de datos tal como se define por la Organización Internacional de Normalización (de *ISO*, sigla del ingl. *International Standards Organization*) en el Modelo de referencia básico OSI (sigla del ingl. *Open System Interconnection*, 'Interconexión de Sistemas Abiertos').

Si el lector desea una mayor información puede consultar el sitio oficial de la IEEE en el apartado de estándares o directamente en [21].

Con base a lo detallado en el estándar para PLC, en el presente trabajo se estudian la modulación OFDM utilizando la transformada de Fourier como núcleo de este esquema de modulación y multiplexado. En la siguiente sección se presenta una revisión general del modelo formal de OFDM y algunos otros propuesto por Matthieu Crussière en [1] para la transmisión de datos sobre un canal PLC.

## 2.3. Fundamentos básicos de OFDM

La técnica OFDM, resulta ser hoy en día una conveniente técnica de transmisión en la capa física, especialmente para ambientes propensos a efectos multi-trayectoria, variantes en el tiempo y selectivos en frecuencia [22]; como por ejemplo el canal inalámbrico Wi-Fi, o en nuestro caso el canal PLC.

En general, el uso de OFDM en un canal PLC tiene beneficios como:

---

- **Robustez ante desvanecimiento selectivo en frecuencia.** El esquema OFDM segmenta el ancho de banda en  $N$  sub-bandas y, al dividir el ancho de banda se discriminan aquellas sub-bandas altamente afectadas por atenuación. Esta ventaja permite visualizar la respuesta en frecuencia de un canal PLC, como un conjunto de  $N$  sub-canales planos. En la [figura 2.5](#) se aprecia dicha propiedad de OFDM ante el esquema de una sola portadora.
- **Eliminación de la interferencia inter símbolo.** La ISI se genera por el efecto de multi-trayectoria que sufre la señal que arriba en el receptor. El esquema OFDM agrega información redundante al inicio de cada símbolo, lo cual hace menos vulnerable a la señal transmitida con respecto a la interferencia inter-símbolo.
- **Buen aprovechamiento y uso del espectro.** El esquema OFDM surge con el propósito de aprovechar el espectro mejor que el esquema FDM (sigla del ingl. *Frequency Division Multiplexing*, 'multiplexado por división de frecuencia'), lo cual es posible a través del uso de portadoras ortogonales y, con lo cual se tiene un uso eficiente del ancho de banda.
- **Uso de técnicas de modulación adaptativa.** El esquema OFDM combinado con modulación DMT, permiten estimar el comportamiento del canal. Lo cual permite a los sistemas de comunicación OFDM, adaptar la modulación, y así mejorar el desempeño en la transmisión de datos.

El resto de esta sección, presenta los modelos formales para el esquema de transmisión OFDM estipulado en el estándar de PLC: IEEE P1901-2010. Por lo tanto, se presenta una visión general de la transmisión de datos bajo este esquema y las técnicas necesarias para lograr cumplir con los objetivos del proyecto.

### 2.3.1. Datos históricos sobre OFDM

El concepto de transmisión paralela y de multiplexado por división de frecuencia tiene sus orígenes a mediados de 1960. En un sistema clásico de transmisión en paralelo, el total de ancho de banda es dividido en  $N$  subportadoras, que se modulan de tal manera que exista

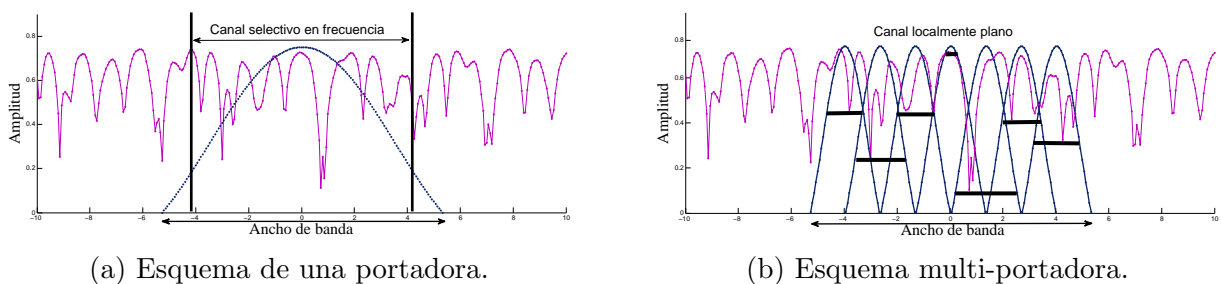


Figura 2.5: Desvanecimiento selectivo en frecuencia.

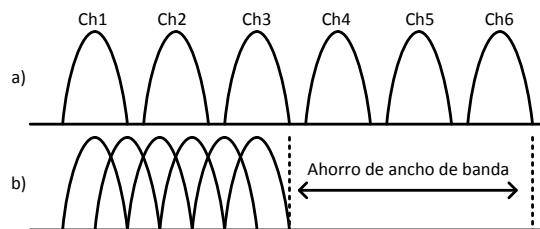
una separación entre ellas, para evitar el traslape.

El uso ineficiente de la banda de frecuencias en FDM, dio paso al desarrollo de OFDM, donde la transmisión se realiza usando portadoras con frecuencias ortogonales. Las portadoras de OFDM se caracterizan por la ortogonalidad con las adyacentes, logrando un uso eficiente de ancho de banda. En la [figura 2.6](#) se aprecia la diferencia entre el esquema FDM y OFDM.

El esquema OFDM, es un caso especial de transmisión multiportadora, el cual puede ser visto como técnica de modulación y multiplexado. En un sistema de portadora única la interferencia podría causar que el enlace falle perdiendo la conexión por completo, pero en un sistema multiportadora, el problema de interferencia genera falla en un pequeño porcentaje de portadoras pero no se pierde el enlace.

Algunas fechas relevantes encontradas durante la revisión de literatura en [\[1, 18, 23\]](#) del esquema OFDM son :

- En 1960 la técnica OFDM fue usada en múltiples sistemas militares a alta frecuencia tales como: KINEPLEX, ANDEFT y KATHARYN.
- En 1970 fue publicada una patente de OFDM.
- En 1971 Weinstein y Ebert implementan por primera vez la transformada discreta de Fourier (de *DFT*, sigla del ingl. *Discrete Fourier Transform*) para la modulación en OFDM.
- En 1980 OFDM fue incluido en módems de altas velocidades y comunicaciones móviles digitales. Por primera vez se empleó la modulación OFDM para multiplexar bandas con modulación QAM (sigla del ingl. *Quadrature Amplitude Modulation*, 'modulación de amplitud en cuadratura') usando la DFT.
- En 1990 OFDM fue usado para comunicaciones en banda ancha en enlaces de radio FM (sigla del ingl. *Frequency Modulation*, 'frecuencia modulada'), en líneas HDSL (sigla del ingl. *High data rate Digital Subscriber Line*, 'línea digital de abonado de alta velocidad') de 1.6Mbps.



[Figura 2.6](#): Comparación entre ancho de banda requerido para (a) FDM, y (b) OFDM.

- En 2002 se especifica a OFDM como técnica de modulación en el estándar IEEE 802.11g.
- En 2003 ya es ampliamente usado OFDM en sistemas PLC.
- En 2004 OFDM se considera para el estándar IEEE 802.11n, la nueva generación de comunicación inalámbrica.
- En 2009 la transmisión OFDM en un canal óptico logra transmitir datos a una velocidad de hasta 160Gbps [24].
- En 2010 se propone como estándar en 3GPP *Long Term Evolution* (LTE), donde LTE es el estándar para comunicación de datos en alta velocidad.
- En 2010 se aprueba el estándar para PLC con OFDM como esquema de modulación.

### 2.3.2. Modulación OFDM

El canal PLC presenta el fenómeno de multitrayectoria, generando que arriben al receptor señales repetidas en diferentes tiempos. Las señales repetidas interfieren con la señal original, a este fenómeno se le conoce como interferencia inter símbolo, que a su vez provoca una degradación significativa en el rendimiento de la red.

Además de los sistemas PLC, el esquema OFDM ha ganado gran terreno como técnica de transmisión, un ejemplo es su uso en el estándar IEEE 802.11g (conocido como Wi-Fi), así como en estándares de radiodifusión de televisión y radio digital [18]. Esta aprobación se debe gracias a que OFDM trabaja paralelizando el flujo de información en subportadoras en el dominio de la frecuencia, lo que proporciona ventajas al transmitir en canales con los efectos negativos del fenómeno de multitrayectorias.

En los sistemas de comunicación digital multiportadora se observan dos clases de interferencia: la ISI y la interferencia inter canal (de *ICI*, sigla del ingl. *Inter Channel Interference*), que se presentan en dominio temporal y en el dominio de la frecuencia, respectivamente, véase la [figura 2.7](#)). El esquema OFDM permite contrarrestar los efectos de ISI e ICI, como se describe en los siguientes párrafos.

### 2.3.3. Principio de ortogonalidad

El concepto que hace de OFDM una técnica ampliamente usada, es el concepto de subportadoras ortogonales. El uso de frecuencias ortogonales en OFDM, permite la superposición de múltiples portadoras en una misma señal. Cada banda se sobrepone a su adyacente sin afectar la recuperación ni generar ICI.

Dos señales son ortogonales si al evaluar el producto punto entre ambas, el resultado es igual a cero. En otras palabras, el área bajo la curva del producto punto entre ambas señales es

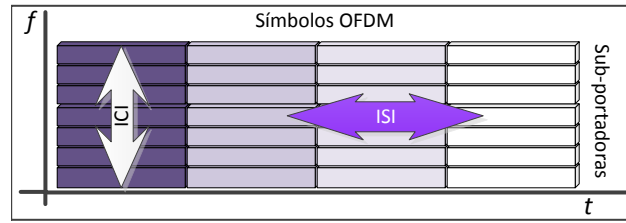


Figura 2.7: Interferencia en tiempo (ISI) y en frecuencia (ICI).

igual a cero. Para comprender aún mejor este concepto se presenta la figura 2.8 (a), donde se aprecian dos señales sinusoidales  $x_1(t)$  y  $x_2(t)$ , con frecuencias  $f_0$  y  $f_1 = 2f_0$ , respectivamente.

Si se calcula el producto punto entre ambas señales, se tiene como resultado la señal mostrada en la figura 2.8 (b). Finalmente, se calcula la integral, teniendo cero como resultado, tal y como se presenta en la ecuación (2.1).

$$\int_0^{1000} x_1(t)x_2(t) dt = 0 \quad (2.1)$$

La base de la ortogonalidad en OFDM, consiste en asignar portadoras ortogonales en cada banda. Para cada portadora se usa una señal sinusoidal finita (sólo se toma una ventana de la señal), donde la frecuencia de esta señal es múltiplo entero de una frecuencia base o central. Como se observa en la figura 2.8 (a), la frecuencia de  $x_2(t)$  es múltiplo de  $f_0$ . Por lo tanto, se considera a  $f_0$  y  $f_1$  portadoras ortogonales entre sí.

### Construcción de señales con portadoras ortogonales.

Ya introducido el concepto de ortogonalidad en OFDM, se presenta el proceso de construcción de señales con portadoras ortogonales. Antes, se debe tener claro que el uso de una señal sinusoidal para las portadoras, no se toma en el intervalo de  $[-\infty, \infty]$ . Por lo tanto,

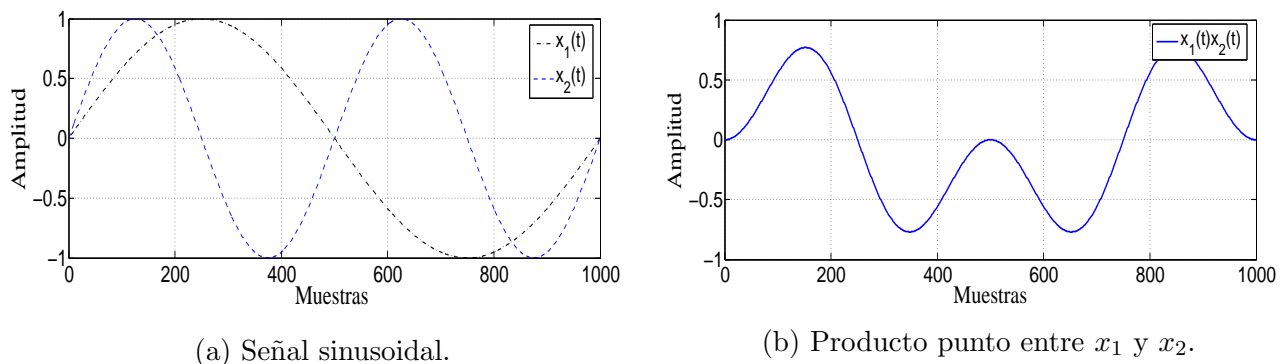


Figura 2.8: Señales sinusoidales ortogonales.

se toma una ventana en el dominio temporal de la señal sinusoidal utilizando la función rectángulo, véase la [figura 2.9 \(a\)](#), la cual tiene una respuesta en frecuencia como se aprecia en la [figura 2.9 \(b\)](#).

De hecho, el uso de la función rectángulo sobre cada una de las señales sinusoidales garantiza conservar la ortogonalidad. En la [figura 2.10](#), se presenta el resultado de mapear al dominio de la frecuencia la ventana de  $x_1$  y  $x_2$ , con ello se tiene una señal con dos portadoras ortogonales. La FFT de una señal sinusoidal evaluada en un intervalo finito de tiempo, resulta una envolvente que tiene la forma de la función seno cardinal, siendo este tipo de espectros que se encuentran en el esquema OFDM.

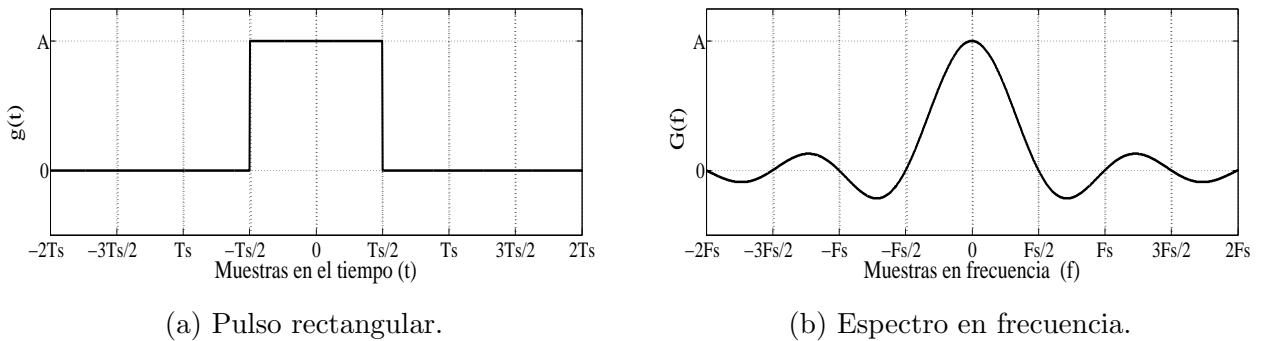
### 2.3.4. Transmisión de una señal OFDM

El principio de OFDM inicia con la conversión de un flujo de datos en serie de alta velocidad a un flujo paralelo de baja velocidad. En la [tabla 2.1](#) se encuentran los parámetros empleados en el resto del capítulo.

El modulador OFDM recibe como entrada,  $b * M$  bits en serie de alta velocidad, que son divididos en  $M$  módulos paralelos de menor velocidad, definiendo símbolos complejos de modulación con la forma  $c_m = a_m + jb_m$ , con  $m = 0, 1, \dots, M$  que pertenecen a una constelación QAM (Quadrature Amplitud Modulation) o PSK (Phase Shift-Keying) para múltiples subcanales espectrales denominados subportadoras.

Un símbolo OFDM es el resultado de aplicar un conversor a simetría Hermitiana a los  $M$  módulos paralelos, resultando un vector de longitud  $N = 2 * M$  de la forma  $X_k = [0, C_{(2:M)}, 0, C_{(M:2)}^*]$ , donde  $C_{(M:2)}^*$  representa el complejo conjugado del vector de símbolos QAM reflejado. La duración de un símbolo OFDM es igual a  $N$  veces el periodo de un bit individual ( $T_{sc}$ ), es decir  $T_s = NT_{sc}$ . El ancho de banda se obtiene dividiendo el número de subportadoras entre la duración de símbolo OFDM, esto es  $B_s = N/T_s$ .

Si se aumenta eficientemente el número de subportadoras, la duración del símbolo aumenta



**Figura 2.9:** Función rectángulo.

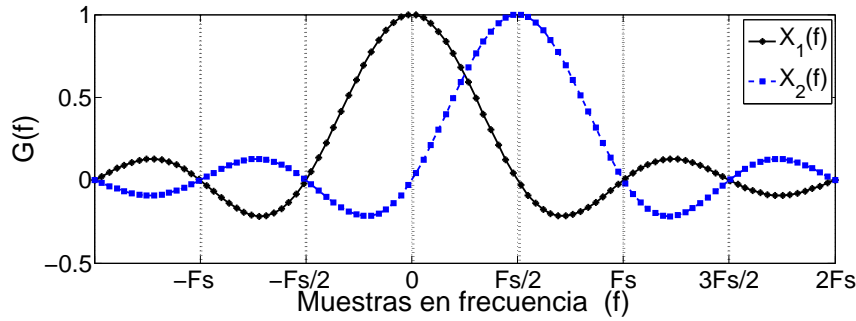


Figura 2.10: Portadoras ortogonales.

de tal manera que la respuesta al impulso se ve menos afectada por efecto de multi-trayectoria y como consecuencia se disminuye el efecto de ISI [18].

La frecuencia de cada subportadora se define como  $f_k = f_0 + \Delta_k$ ,  $k \in [0 \dots N - 1]$ , siendo  $f_0$  la frecuencia para la portadora cero. Asumiendo que el principio de ortogonalidad se cumple, entonces  $\Delta_k = 1/T_s$ . Entonces, la expresión normalizada de la señal OFDM está dada por: [18]

$$s(t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \Re \left\{ x_k \Pi(t) e^{(j2\pi(f_0 + \frac{k}{T_s})t)} \right\} \quad t \in [0 \dots T_s] \quad (2.2)$$

Donde  $\Pi(t)$  representa la función rectángulo que está definida por

Tabla 2.1: Declaración de parámetros OFDM.

| Variable     | Descripción                                    |
|--------------|--|
| $N$ :        | Longitud de símbolo OFDM; Longitud de la FFT   |
| $k$ :        | Índice del vector de muestras del símbolo OFDM |
| $M$ :        | Longitud del vector de símbolos QAM o PSK      |
| $m$ :        | Índice del vector de símbolos QAM              |
| $D$ :        | Número de muestras del intervalo de guarda     |
| $b$ :        | Bits por símbolo QAM                           |
| $T_{QAM}$ :  | Duración de símbolo QAM o PSK                  |
| $T_s$ :      | Duración de un símbolo OFDM                    |
| $T_{sc}$ :   | Duración de una portadora única                |
| $f_c$ :      | Frecuencia central                             |
| $f_s$ :      | Frecuencia del símbolo OFDM                    |
| $\Delta_k$ : | Valor incremental de la frecuencia $k$ -ésima  |
| $R_s$ :      | Tasa de transmisión de OFDM                    |
| $B_s$ :      | Ancho de banda por símbolo OFDM                |



$$\Pi(t) = \begin{cases} A & |t| < \frac{1}{2}T_s, \\ 0 & \text{Otro caso.} \end{cases} \quad (2.3)$$

y  $A$  es la amplitud de la señal. En la [figura 2.9](#) se representa la función rectángulo así como su espectro en frecuencia. Recordando el principio de ortogonalidad en el esquema OFDM, es necesario definir el valor de los múltiplos de la frecuencia que garantiza la ortogonalidad, a este valor se le conoce como frecuencia central, y se define como  $f_c = f_0 + N/(2T_s)$ , al sustituir en la ecuación (2.2) se tiene

$$s(t) = \Re \left\{ \Pi(t) e^{j2\pi f_c t} \sum_{k=0}^{N-1} \frac{x_k}{\sqrt{N}} e^{j2\pi(k-\frac{N}{2})\frac{t}{T_s}} \right\} \quad (2.4)$$

que puede expresarse como:

$$s(t) = \Re \{ \tilde{s}(t) \Pi(t) e^{j2\pi f_c t} \} \quad (2.5)$$

Donde  $\tilde{s}(t)$  es la parte compleja de la señal  $s(t)$  antes del ventaneo por la función rectángulo. Estando limitado el espectro al intervalo  $[-\frac{N}{2T_s} : \frac{N}{2T_s}]$ , la señal  $\tilde{s}(t)$  puede ser muestreada a una frecuencia  $f_m = \frac{N}{T_s}$  sin que haya superposición espectral. Las muestras obtenidas se expresan

$$\begin{aligned} s_n &= \Re \left\{ \Pi\left(n\frac{T_s}{N}\right) e^{j2\pi f_c n\frac{T_s}{N}} \sum_{k=0}^{N-1} \frac{x_k}{\sqrt{N}} e^{j2\pi(k-\frac{N}{2})\frac{n}{N}} \right\} \\ &= \Re \left\{ (-1)^n \Pi\left(n\frac{T_s}{N}\right) e^{j2\pi f_c n\frac{T_s}{N}} \underbrace{\sum_{k=0}^{N-1} \frac{x_k}{\sqrt{N}} e^{j2\pi\frac{kn}{N}}}_{DFT^{-1}} \right\} \end{aligned} \quad (2.6)$$

De la ecuación (2.6), se observa que la señal puede ser generada fácilmente utilizando una transformada discreta de Fourier inversa (de *IDFT*, sigla del ingl. *Inverse Discrete Fourier Transform*)<sup>2</sup>, por lo que el receptor tiene que realizar una DFT para recuperar los símbolos QAM o PSK según sea el caso [18, 23]

Una representación de la señal OFDM puede ser en forma matricial: donde se representa como

$$S_k = F^{-1} x_k \quad (2.7)$$

---

<sup>2</sup>En la práctica se usa la transformada rápida de Fourier inversa (de *IFFT*, sigla del ingl. *Inverse Fast Fourier Transform*), la cual tiene una complejidad numérica de  $O(N \log N)$  en función a  $N$  puntos, por otra parte, la complejidad de la IDFT es  $O(N^2)$

---

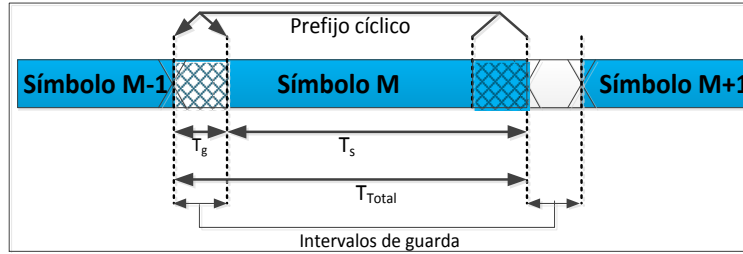


Figura 2.11: Inserción del intervalo de guarda.

siendo  $S_k$  es el vector de salida  $[S_0, S_1, \dots, S_{(k-1)}]^T$ , que representa al símbolo OFDM,  $F$  es la matriz inversa de Fourier de orden  $N \times N$  definida por:

$$A = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{-j\frac{2\pi}{N}} & \dots & e^{-j\frac{2\pi(N-1)}{N}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{2\pi(N-1)}{N}} & \dots & e^{-j\frac{2\pi(N-1)^2}{N}} \end{bmatrix} \quad (2.8)$$

Recuérdese que esta matriz es unitaria y su inversa es igual a su transpuesta Hermitiana (conjugada)  $F^{-1} = F^H$ . [18]

### 2.3.5. Intervalo de guarda

Una solución para reducir la interferencia inter símbolo es incrementar la duración del símbolo ( $T_s$ ) de manera ilimitada. Sin embargo, este método podría dificultar la implementación en términos de la estabilidad y la ortogonalidad entre las subportadoras y el tamaño de la FFT. Otra manera de limitar la ISI, conservando el principio de ortogonalidad entre portadoras, es agregar un intervalo de guarda (de *GI*, sigla del ingl. *Guard Interval*). Un GI es un lapso donde no se transmite información útil, para garantizar la ortogonalidad, el símbolo OFDM es precedido por una extensión periódica de la misma señal. A este intervalo se le conoce como prefijo cíclico y tiene una duración total de  $T_g + T_s$ , donde  $T_g$  es la duración del intervalo de guarda [18,25].

Entonces, la señal a transmitir se ve afectada por la inserción del GI, resultando  $s_n = [g_m + s_k]$ , donde  $g_m$  es una réplica de las últimas  $m$  muestras de  $s_k$ . En la figura 2.11 ilustra un símbolo OFDM con su respectivo prefijo cíclico.

### 2.3.6. Símbolo piloto

El uso de un símbolo piloto en la transmisión de OFDM y otros esquema de comunicación, se conoce como esquema asistido por piloto. La información contenida en el piloto es conocida por el receptor y juega un papel de suma importancia para la sincronización, estimación de

canal y estimación de ruido. La inserción se realiza como se ve en la [figura 2.12](#) [25], la cual se agrega al inicio de cada trama de símbolos OFDM. Una trama OFDM, es una secuencia de símbolos OFDM que portan la carga útil.

Los símbolos piloto son insertados al inicio de las tramas OFDM, estos símbolos pueden ser de los siguientes tipos:

- Los símbolos de tipo "A" se insertan al inicio de cada trama OFDM, y se usan en el proceso de sincronización y estimación de canal.
- Los símbolos de tipo "B" se transmiten sólo eventualmente, y se colocan enseguida de los de tipo "A". Éstos se emplean para la estimación de ruido.

Si el lector desea conocer más acerca del uso de los símbolos piloto se recomienda consultar [1]. En el presente trabajo, se propuso el uso de los símbolos piloto para realizar el proceso de sincronización, estimación de canal y de ruido. En la siguiente sección se describe el modelo formal de los procesos mencionados, donde es de utilidad el símbolo piloto.

## 2.4. Proceso de sincronización y estimación de canal

En esta sección se presentan los modelos formales para el proceso de sincronización y de estimación de canal, propuestos por M. Crussière en [1].

### 2.4.1. Sincronización

Se dice que dos señales con portadora múltiple están sincronizadas si la ortogonalidad entre las subportadoras se preserva una vez recibidas las señales. Lo que implica que en el receptor, la ventana de muestreo inicie en el instante preciso del inicio de los símbolos OFDM.

Para efecto de estudio de este proyecto, se presenta una técnica de sincronización que trabaja con la información que se encuentra en los símbolos piloto de tipo "A". Para realizar la sincronización, Crussière propone un esquema en dos etapas:

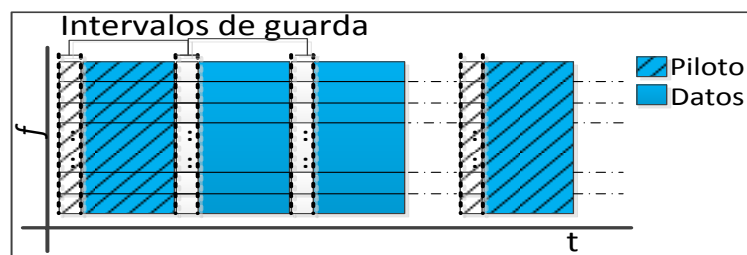


Figura 2.12: Esquema OFDM asistido por piloto.

- a) **Sincronización burda.** En la primera etapa, se estima el inicio aproximado de los símbolos, para ello se utiliza la técnica de correlación del prefijo cíclico. Como se vera más adelante, esta etapa garantiza una buena estimación del inicio de cada símbolo OFDM.
- a) **Sincronización fina.** En la segunda etapa, se realiza un ajuste más preciso del inicio del símbolo siguiente. Para ello, se emplea la técnica de mínimos cuadrados (de *LS*, sigla del ingl. *Least Square*) sobre el seguimiento de fase de los símbolos piloto tipo "A".

### Sincronización burda

La propuesta en [1] para estimación de la diferencia de tiempo o desfase ( $\delta$ ) de las muestras de los símbolos OFDM afectados por un canal gaussiano, se realiza con base al principio de máxima verosimilitud (de *ML*, sigla del ingl. *Maximum Likelihood*) en el dominio temporal.

$$\delta_t^{ML} = \frac{1}{2} \arg_{\delta_t} \max (\Psi(\delta_t) - \frac{\rho}{2} \Phi(\delta_t)) \quad (2.9)$$

donde  $\Psi(\delta_t)$ : es la métrica de correlación sobre  $D$  muestras de la ventana entrante desplazadas en  $2N$  (donde  $N$  son los puntos de la FFT) posiciones. La métrica  $\Phi(\delta_t)$ , es la potencia total de las  $2D$  muestras. Ambas métricas se definen:

$$\Psi(\delta_t) = \sum_{n=\delta_t}^{\delta_t+D-1} r_n r_{n+N'} \quad (2.10)$$

$$\Phi(\delta_t) = \sum_{n=\delta_t}^{\delta_t+D-1} r_n^2 + r_{n+N'}^2 \quad (2.11)$$

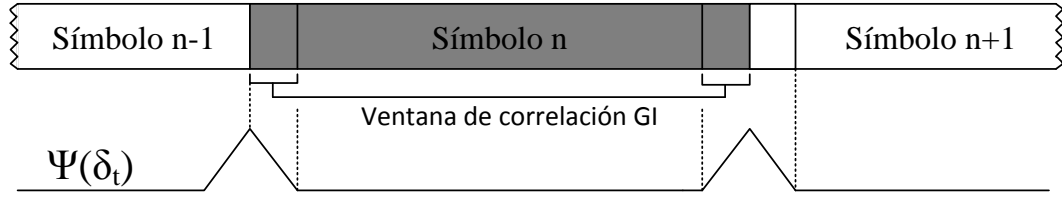
donde  $r_n$  es la ventana de muestras recibidas de tamaño  $N' = 2N$ .

Se define  $\rho = \frac{SNR}{(SNR+1)}$ , donde  $\rho$  tiene dos posibles limites:  $\rho \rightarrow 1$  o  $\rho \rightarrow 0$ , esto si los niveles de relación a señal a ruido son altos o bajos, respectivamente. Por lo tanto, se obtienen las siguientes expresiones de estimación sub-óptimas con base al nivel SNR que exista:

$$\delta_t^{MMSE} = \frac{1}{2} \arg_{\delta_t} \max (\Psi(\delta_t) - \frac{1}{2} \Phi(\delta_t)) \quad (2.12)$$

$$\delta_t^{MC} = \frac{1}{2} \arg_{\delta_t} \max (\Psi(\delta_t)) \quad (2.13)$$

En la ecuación 2.12, se realiza una estimación del mínimo error cuadrático medio (de *MMSE*, sigla del ingl. *Minimum Mean Squared Error*) para niveles altos de relación señal a ruido, mientras que para los niveles bajos se realiza una estimación por máxima correlación (de *MC*, sigla del ingl. *Maximum Correlation*), tal y como se muestra en la ecuación 2.13 y se observa en la [figura 2.13](#). En la práctica, es más común usar una estimación MC. La correlación GI sólo es calculada una vez cada inicio de la recepción, posteriormente sólo se actualiza la ventana de correlación con las muestras entrantes de la siguiente manera:

Figura 2.13: Correlación  $\Psi(\delta_t)$ .

$$\Psi(\delta_t + 1) = \Psi(\delta_t) + r_{\delta_t+1}r_{\delta_t+1+N'} - r_{\delta_t}r_{\delta_t+N'} \quad (2.14)$$

Los efectos de un canal gaussiano sobre la sincronización burda, genera una pérdida de precisión o desfase, para corregir este desfase se propone la etapa de sincronización fina, que se describe a continuación.

### Sincronización fina

El proceso de sincronización fina, permite corregir el posicionamiento de la ventana del símbolo de referencia de forma perfecta. Crussière propone un ajuste fino con base en el principio de rotación de fase. Este procesamiento se realiza en el dominio de la frecuencia, es decir, después de aplicar la FFT a una ventana de  $N$  muestras recibidas. La estimación fina ( $\hat{\epsilon}$ ) se calcula con base en el principio de mínimos cuadrados lineales (de *LLS*, sigla del ingl. *Linear Least Square*) en el dominio de la frecuencia, y se propone como:

$$\hat{\epsilon}_f^{LLS}(p) = \frac{1}{2\pi} \frac{\sum_{k=0}^{N-1} k \Delta\phi_k(p)}{\sum_{k=0}^{N-1} k^2} \quad (2.15)$$

donde  $\Delta\phi_k(p)$  es el resultado de la diferencia de fase entre el vector de datos del  $p$ -ésimo símbolo piloto transmitido y el recibido. La representación vectorial de la diferencia de fase se denota  $\Delta\phi_k = [\phi_0(p), \phi_1(p), \dots, \phi_{(N-1)}(p)]^T$ , este vector también se expresa como

$$\Delta\phi_k = \angle(X_A^T Y_p) \quad (2.16)$$

donde  $X_A^T = \text{diag}([x_0^A, x_1^A, \dots, x_{(N-1)}^A])$ .  $Y_p$  es el vector de muestras recibidas afectadas por el canal PLC.

El operador  $\angle$  representa el cálculo del ángulo de fase. En la figura 2.14 se observa el resultado de la corrección de fase [1], donde el ángulo de la pendiente de la recta sobre la corrección de fase lo determina el operador angular aplicando la siguiente condición:

$$\phi_k = \begin{cases} \phi_k - 2\pi & (|\phi_k - \phi_{k-1}| < \pi) y (\phi_k < 0), \\ \phi_k + 2\pi & (|\phi_k - \phi_{k-1}| < \pi) y (\phi_k \geq 0), \\ \phi_k & \text{Otro caso.} \end{cases} \quad (2.17)$$

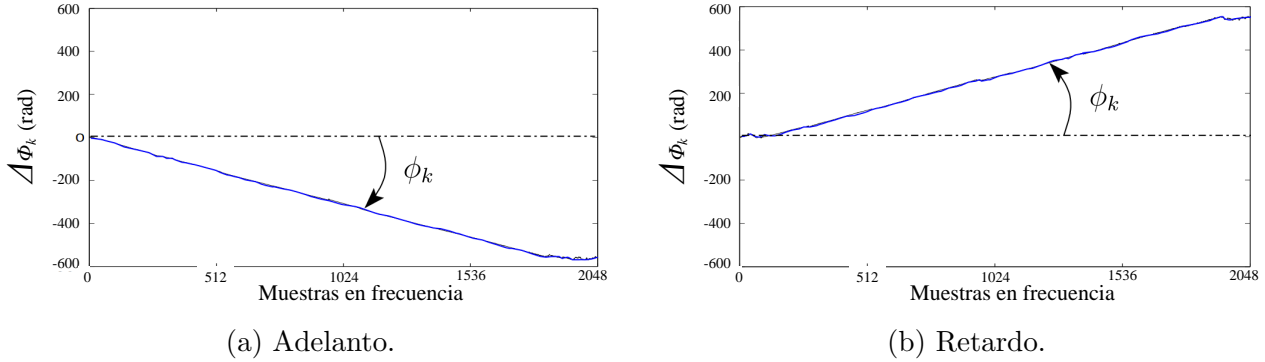


Figura 2.14: Corrección de fase.

### 2.4.2. Estimación del canal

En sistemas que utilizan el esquema OFDM, y que operan en un canal con dispersión de la respuesta al impulso, típicamente se inserta al inicio de la señal transmitida un prefijo cíclico de mayor duración que la dispersión de la respuesta al impulso del canal. Como resultado, el efecto del canal es una dispersión multiplicativa en cada sub-banda OFDM. Entonces, la estimación de canal se expresa como:

$$H_{(l,k)} = \frac{C'_{(l,k)}}{C_{(l,k)}} \quad (2.18)$$

Donde  $H_{(l,k)}$  es el estimado para el coeficiente complejo de atenuación para la subportadora  $k$  del  $l$ -ésimo símbolo y,  $C_{(l,k)}$  representa la secuencia de referencia. Y  $C'_{(l,k)}$  la señal analítica del símbolo piloto afectado por el canal. Dependiendo de la procedencia del valor de referencia, se pueden clasificar en 4 técnicas: asistidas por piloto, regidas por decisión, semi-ciegas y ciegas [26].

Para efectos de este proyecto, se ha estudiado las técnicas con estimador clásico de transmisión asistida por piloto. El uso de estos esquemas presenta ventajas sobre el proceso de sincronización, estimación de ruido y de canal. También presenta la desventaja de enviar y recibir símbolos OFDM sin información útil.

#### Estimador clásico de canal asistido por piloto

En esta técnica la señal analítica  $C_{(l,k)}$  se conoce tanto por el emisor como por el receptor, de esta manera se hace más sencillo el cálculo de la ecuación 2.18.

#### Estimación de canal por mínimos cuadrados (LS)

Los coeficientes obtenidos a través de la estimación LS son:

$$H_{(l,k)}^{LS} = \frac{C'_{(l,k)}}{C_{(l,k)}} \quad (2.19)$$

donde  $C'_{(l,k)}$  es la muestra del  $l$ -ésimo símbolo OFDM recibido sobre la  $k$ -ésima portadora y  $C_{(l,k)}$  es el símbolo original correspondiente.

### Estimación por ventaneo rectangular de respuesta al impulso en el dominio temporal (VR)

Esta técnica realiza un filtrado por ventaneo rectangular sobre la respuesta al impulso de los coeficientes de estimación. El filtro rectangular, resulta ser una técnica eficiente y sencilla de implementar para eliminar el ruido en la estimación de canal.

Para llevar a cabo esta técnica, los coeficientes  $H$  son mapeados al dominio temporal mediante la IFFT, posteriormente se colocan ceros en los valores de las muestra que se encuentran por encima de la máxima dispersión temporal del canal. Finalmente, la secuencia es regresada al dominio de la frecuencia mediante la FFT, obteniendo de forma simple una respuesta en frecuencia suavizada y una mejora en la estimación del canal.

## 2.5. Modelo de ruido

Como se mencionó anteriormente, la red eléctrica presenta más de un tipo de ruido (Gaussiano, impulsivo, de banda estrecha, entre otros), que afectan la señal transmitida, en este apartado se plantea el modelo a seguir para simular el efecto del ruido sobre nuestra señal evaluada.

### 2.5.1. Canal AWGN

Se puede definir al ruido como cualquier perturbación no deseada que altera la señal que es transmitida por un canal. El ruido aditivo gaussiano de clase A, es considerado en la literatura para representación del ruido impulsivo en un canal PLC. Este tipo de ruido es una combinación de ruido aditivo blanco gaussiano con ruido impulsivo [26].

El presente trabajo se ve limitado en tiempo, por lo que sólo se considera de manera formal el ruido aditivo blanco. El daño de una señal por ruido aditivo Gaussiano o blanco es de lo más común en las comunicaciones, ya sean digitales o analógicas. El ruido, en las comunicaciones digitales nos genera daño parcial o total de la señal que transporta la cadena binaria. En la [figura 2.15](#) se aprecia el efecto de un canal AWGN sobre una señal.

El modelo a seguir para simular la potencia de ruido Gaussiano, está dado por:

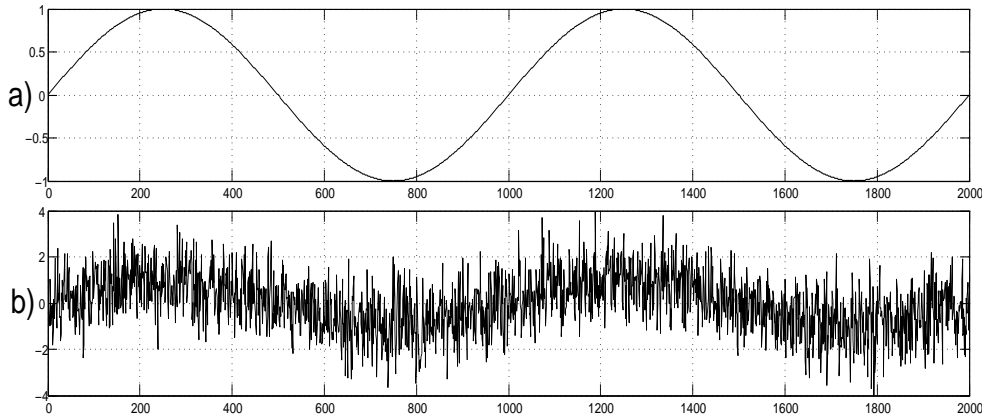


Figura 2.15: Efecto de un canal AWGN, a) señal original, b) señal afectada por ruido.

$$P_n = \frac{P_s}{10^{\left(\frac{SNR}{10}\right)}} \quad (2.20)$$

donde  $P_s$  es la potencia promedio de la señal transmitida, y definida como:

$$P_s = \frac{1}{N + D} \sum_{k=0}^{N+D-1} \hat{s}_k^2 \quad (2.21)$$

y  $\hat{s}$  representa la señal original.

Una vez conocida la potencia del ruido, se genera mediante valores pseudo-aleatorios con distribución Gaussiana ( $n_k$ ) y con estadísticas:  $\mu = 0$  y  $\sigma = \sqrt{P_n}$ . Finalmente la señal afectada se representa como:

$$\tilde{s}_k = \hat{s}_k + n_k \quad (2.22)$$

## 2.6. Kit de desarrollo para el DSP

Se denomina DSK (de *DSK*, sigla del ingl. *DSP Starter Kit*) al conjunto de periféricos de entrada/salida, memoria, microprocesador y otros dispositivos de hardware instalados en una tarjeta de desarrollo, la cual permite implementar y evaluar aplicaciones específicas de alto rendimiento. En los siguientes párrafos se describen los componentes básicos del DSK.

### 2.6.1. Procesador digital de señales

La familia del procesador digital de señales de *Texas Instrument* tiene sus inicios en el año de 1982 (TMS32010). Desde su inicio los DSP de esta familia han proliferado en el mercado



y en la industria, aumentando día a día su capacidad de procesamiento. Es por ello que, para fines de evaluación de los algoritmos de la capa física de un módem OFDM usar este microprocesador es ideal.

Algunas de las características que hacen del DSP una herramienta potente para aplicaciones de alta demanda de procesamiento son:

- **Formato de representación numérica**

Una de las características fundamentales de los DSP es el tipo de formato aritmético utilizado por el procesador. La [figura 2.16](#) muestra la estrecha relación entre formato numérico y número de bits del DSP.

Un DSP de punto flotante es generalmente más fácil de programar que los de punto fijo, pero son usualmente más caros. En los DSP de punto fijo el programador debe escalar las señales en sus programas para asegurar una adecuada precisión numérica teniendo cuidado en no rebasar el rango por tipo de variable.

- **Tamaño de palabra**

En los DSP de punto fijo, el tamaño de palabra más común es de 16 bits. El tamaño del bus de datos tiene un gran impacto en el costo, ya que influye notablemente en el tamaño del chip y en el número de terminales del encapsulado.

- **Velocidad de ejecución**

Una métrica de desempeño para saber si un DSP es o no apropiado para una aplicación es su velocidad de ejecución. Existen varias formas para medir la velocidad de un procesador, aunque quizás el parámetro más usual es el tiempo de ciclo de instrucción: tiempo necesario para ejecutar la instrucción más rápida del procesador. Su inverso dividido por un millón da lugar a la velocidad del procesador en millones de instrucciones por segundo o MIPS. El DSP de la familia TMS320C64x tiene la capacidad de realizar hasta 8000 millones de MIPS.

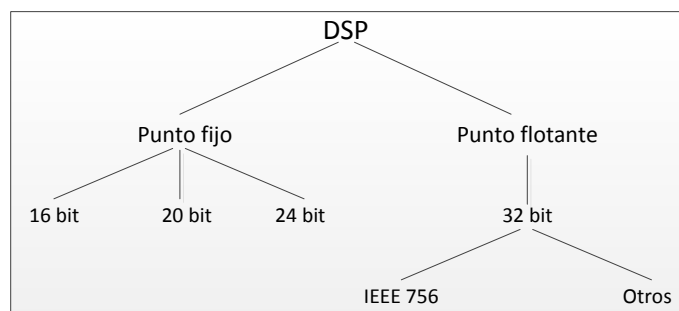


Figura 2.16: Clasificación del formato aritmético.

- **Segmentación y entubamiento**

Se cuenta con la técnica de segmentación *Pipelining*, la que es una técnica para incrementar la velocidad de un procesador, que consiste en dividir una secuencia de operaciones en otras de más sencillas y ejecutar en lo posible cada una de ellas en paralelo. En consecuencia se reduce el tiempo total requerido para completar un conjunto de operaciones. Casi todos los DSP del mercado incorporan el uso de la segmentación en mayor o menor medida.

- **Consumo de energía**

El uso masivo de esta tecnología en dispositivos móviles obligó a diseñar a los DSP con el mínimo requisito de alimentación energética. Una técnica que implementan para disminuir el consumo de energía es la modalidad *sleep* o *idle* que gestiona el consumo de acuerdo a la demanda del proceso a ejecutar o suspender.

- **Características técnicas**

Otras características que nos permiten saber la capacidad y limitantes del DSP, se pueden revisar en [27].

### 2.6.2. Code Composer Studio (CCS)

El CCS, es un ambiente de desarrollo integrado que funciona como interfaz entre la computadora y el DSK, permitiendo una programación en lenguajes de alto nivel (C o C++). El CCS incluye un conjunto de herramientas que permiten desarrollar y depurar aplicaciones con cualquier DSP de *Texas Instruments*: un compilador para cada modelo de DSP, editor de código fuente, explorador de proyectos, ensamblador, depurador, enlazador y simuladores en tiempo real, así como la opción de visualización gráfica de los resultados y muchas otras herramientas.

El CCS permite observar, en tiempo de ejecución, las variables contenidas en memoria, tanto de forma gráfica como numérica. Otra opción que resulta de gran utilidad, es la que permite colocar puntos de ruptura sobre el código (*Break point*) y puntos de prueba para periféricos I/O.

Este ambiente de desarrollo incluye una serie de bibliotecas que permiten al usuario facilitar la programación. Algunas de ellas se encuentran con más detalle en [28], además en [29, 30] se encuentra una guía completa para comenzar a trabajar con el CCS.

---

## 2.7. Método Monte Carlo

El método Monte Carlo es un método estadístico numérico que permite resolver problemas físicos y matemáticos mediante la simulación de variables aleatorias. Este método fue llamado así por el principado de Mónaco por ser "la capital del juego de azar", al tomar una ruleta como un generador simple de números aleatorios.

Este método fue empleado como herramienta de investigación durante la segunda guerra mundial en el Laboratorio Nacional de los Álamos en de los Estados Unidos de América, para el desarrollo de la bomba atómica. El nombre y el desarrollo sistemático de los métodos de Monte Carlo datan aproximadamente de 1944 con el desarrollo de la computadora electrónica. En 1947 John Von Neumann envió una carta a los laboratorios del Álamo, en la que expuso de modo exhaustivo tal vez el primer informe por escrito del método de Monte Carlo.

La importancia actual del método Monte Carlo se basa en la resolución de problemas complejos de resolver que dependen de factores aleatorios (resolución de integrales de muchas variables, minimización de funciones, etc.).

---

---

## Capítulo 3

# Programación y evaluación

---

En este capítulo se presenta el sistema de prueba, el cual permitió la evaluación de los algoritmos de la capa física de un módem para PLC. Los modelos formales programados y evaluados en este capítulo (descritos en el capítulo anterior), fueron propuestos por M. Crussière [1], también se encuentra una síntesis en español realizada por G. Laguna en [18], quien también desarrolló la simulación por computadora de la transmisión sobre PLC [26]. Este simulador se ha empleado en el presente trabajo como sistema de referencia, del cual se han tomado algunos resultados y propuestas para ser puestos en práctica con el DSP.

### 3.1. Arquitectura general del sistema de prueba

El propósito de esta sección es introducir de forma general el sistema de prueba, donde se presentan los módulos fundamentales para la transmisión de datos, usando el esquema de modulación OFDM sobre un canal PLC.

La simulación es una herramienta de uso general para cualquier disciplina tanto de investigación como en la docencia. Las comunicaciones digitales también hacen uso de esta potente herramienta para la evaluación de sistemas complejos en ambientes muy próximos a uno real.

Dado que los objetivos del presente trabajo, no contempla los últimos pasos para el desarrollo práctico de un transceptor para la evaluación de los algoritmos de la capa física de un módem PLC, se propuso la simulación Monte Carlo, con la cual se pretende evidenciar el desempeño de los algoritmos implementados en el DSP.

La [figura 3.1](#) muestra la arquitectura general del sistema de prueba por medio de un diagrama de bloques, donde se observan los módulos de: información fuente, transmisor, canal, puesta a punto, receptor, información recuperada y contador de errores. A continuación se da una descripción general de cada módulo:

- **Información fuente.** Este es el módulo inicial del sistema de prueba, la función de éste es la de propiciar una secuencia binaria de entrada al sistema. La secuencia se genera de forma pseudo-aleatoria para los datos o se usa alguna secuencia constante para los símbolos piloto.

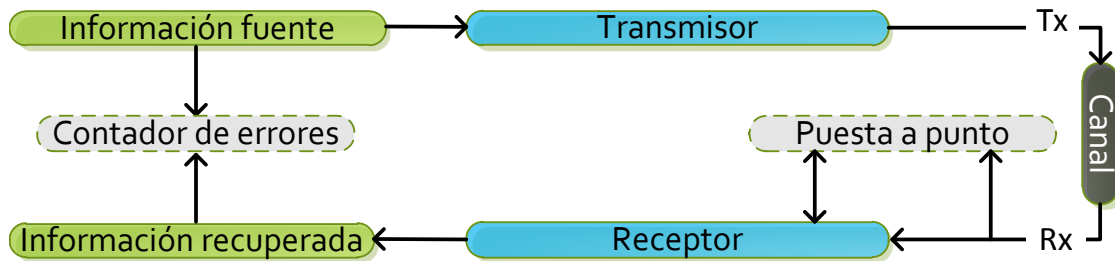


Figura 3.1: Esquema a bloques del sistema de prueba.

- **Transmisor.** En este módulo se lleva a cabo todo el cómputo necesario para la transmisión de señales OFDM a partir de secuencias binarias sin codificación.
- **Canal.** Este módulo se encarga de simular los efectos característicos de un canal PLC, como: ruido blanco Gaussiano y atenuación selectiva en frecuencia.
- **Puesta a punto.** Este módulo forma parte del receptor, y realiza procesos que han de permitir establecer la conexión entre emisor y receptor. Algunos procesos que realiza como sincronización, estimación de canal entre otros son de suma importancia para el buen funcionamiento de un módem OFDM.
- **Receptor.** Este módulo contiene los algoritmos necesarios para recuperar la información enviada por el transmisor. Es importante mencionar que la información ha sido afectada por los efectos negativos del canal. Por tanto, este módulo contiene técnicas avanzadas que permiten recuperar la información con el menor número de errores.
- **Contador de errores.** Este módulo es el responsable de contabilizar los bits de la secuencia recibida que difieren de la secuencia original. La diferencia permitirá conocer el desempeño del sistema mediante el BER (sigla del ingl. *Bit Error Rate*, 'tasa de error de bit') por cada valor de SNR evaluado.

La [tabla 3.1](#) contiene un resumen de la notación utilizada en los algoritmos de cada módulo a lo largo de este capítulo. En los siguientes apartados se presenta con mayor detalle cada uno de los módulos antes mencionados, así como algunos algoritmos involucrados en la transmisión de datos en un módem OFDM para un canal PLC.

### 3.1.1. Información fuente

Este módulo es el responsable de alimentar al sistema con secuencias binarias definidas como  $\vec{v} = [v_0, v_1, \dots, v_{b \cdot M - 1}]$  (nombrado  $\vec{v}_{on-off}$  para futuras referencias), donde  $b$  es el número de bits por símbolo QAM y  $M$  el número de símbolos QAM por símbolo OFDM. Cabe hacer mención que en [31] se desarrollan de forma paralela, al presente trabajo, los módulos de codificación y decodificación para la capa física de un módem OFDM/PLC.

Tabla 3.1: Notación del sistema de prueba.

| Expresión             | Comentario                                    | Longitud (muestras) | Módulo                  |
|-----------------------|---|---------------------|-------------------------|
| $\vec{v}_{on-off}$    | Secuencia binaria                             | $b * M$             | Información fuente      |
| $\vec{v}_{piloto}$    | Secuencia binaria constante                   | 64                  | Información fuente      |
| $\vec{v}_{on-off-A}$  | Secuencia binaria del símbolo piloto tipo "A" | $b * M$             | Información fuente      |
| $\vec{v}_{on-off-R}$  | Secuencia binaria del símbolo tipo "R"        | $b * M$             | Transmisor              |
| $\vec{v}_{antipodal}$ | Secuencia antipodal                           | $b * M$             | Transmisor              |
| $C$                   | Señal analítica original                      | $M$                 | Transmisor              |
| $X^H$                 | Matriz Hermitiana                             | $N$                 | Transmisor              |
| $\vec{s}_n$           | Símbolo OFDM original                         | $N$                 | Transmisor              |
| $\vec{s}'_{n'}$       | Símbolo OFDM original con intervalo de guarda | $N + D$             | Transmisor              |
| $\vec{s}_k$           | Trama OFDM original                           | $T$                 | Transmisor-Canal        |
| $\vec{s}'_k$          | Trama OFDM afectada por un canal AWGN         | $T$                 | Canal                   |
| $\vec{r}'_{n'}$       | Símbolo OFDM recibido con intervalo de guarda | $N + D$             | Puesta a punto/Receptor |
| $\vec{r}_n$           | Símbolo OFDM recibido sin intervalo de guarda | $N$                 | Puesta a punto/Receptor |
| $Y^H$                 | Matriz Hermitiana del símbolo entrante        | $N$                 | Puesta a punto/Receptor |
| $H$                   | Coefficientes de estimación (LS o VR)         | $M$                 | Puesta a punto/Receptor |
| $C'$                  | Señal analítica del símbolo entrante          | $M$                 | Puesta a punto/Receptor |
| $C''$                 | Estimación de la señal analítica transmitida  | $M$                 | Puesta a Receptor       |

Las secuencias binarias pueden ser constantes o pseudo-aleatorias, el uso de cada una se describe a continuación.

- **Secuencias constantes.** El sistema OFDM propuesto, emplea secuencias binarias constantes para la formación de símbolos piloto. El uso de patrones conocidos, en la secuencia empleada para símbolos piloto contribuye a una mejor estimación del canal.

La secuencia propuesta para la construcción de símbolos piloto tipo "A" ( $\vec{v}_{on-off-A}$ ), se presenta en la [tabla 3.2](#), donde se observa que la secuencia base se compone por 64 valores binarios ('0' y '1'). Es importante mencionar que la secuencia empleada en el presente trabajo fue tomada del sistema de referencia [26]. El [Algoritmo 3.1](#) muestra la construcción de la señal digital para los símbolos piloto "A".

- **Secuencias pseudo-aleatorias.** El uso de estas secuencias binarias ( $\vec{v}_{on-off-R}$ ) tiene dos utilidades: la primera es representar a los datos útiles de una transmisión real. Y la segunda es generar símbolos no correlacionados (para futuras referencias serán nombrados símbolos OFDM de tipo "R") introducidos en la trama OFDM empleada para la puesta a punto del sistema, tal y como se describe más adelante.

### 3.1.2. Transmisor

El módulo de transmisión genera una señal OFDM a partir de una secuencia binaria de entrada ( $\vec{v}_{on-off}$ ). Dicha secuencia se somete a un proceso digital hasta obtener una señal

Tabla 3.2: Secuencia binaria del símbolo tipo "A".

| Índice | Valor | Índice | Valor | Índice | Valor | Índice | Valor |
|--------|-------|--------|-------|--------|-------|--------|-------|
| 1      | 1     | 17     | 0     | 33     | 1     | 49     | 0     |
| 2      | 1     | 18     | 1     | 34     | 1     | 50     | 1     |
| 3      | 1     | 19     | 0     | 35     | 1     | 51     | 1     |
| 4      | 1     | 20     | 1     | 36     | 1     | 52     | 0     |
| 5      | 0     | 21     | 0     | 37     | 1     | 53     | 1     |
| 6      | 0     | 22     | 1     | 38     | 1     | 54     | 1     |
| 7      | 0     | 23     | 0     | 39     | 1     | 55     | 0     |
| 8      | 0     | 24     | 1     | 40     | 1     | 56     | 0     |
| 9      | 0     | 25     | 0     | 41     | 0     | 57     | 1     |
| 10     | 0     | 26     | 0     | 42     | 0     | 58     | 0     |
| 11     | 0     | 27     | 1     | 43     | 0     | 59     | 0     |
| 12     | 0     | 28     | 1     | 44     | 0     | 60     | 1     |
| 13     | 1     | 29     | 0     | 45     | 0     | 61     | 0     |
| 14     | 1     | 30     | 1     | 46     | 0     | 62     | 0     |
| 15     | 1     | 31     | 1     | 47     | 0     | 63     | 1     |
| 16     | 1     | 32     | 0     | 48     | 0     | 64     | 1     |

---

**Algoritmo 3.1** Construcción de la secuencia binaria del símbolo piloto "A".

---

**Entrada:** Longitud del vector de símbolos QAM  $M$ ,  $b$  y la secuencia constante binaria para el símbolo piloto "A" ( $\vec{v}_{piloto}$ ).

**Salida:** Secuencia binaria del símbolo piloto tipo "A" ( $\vec{v}_{on-off-A}$ ).

- 1:  $w = (b * M) / 64$
  - 2: **Para**  $i \leftarrow 0$  **to** 64 **Hacer**
  - 3:   **Para**  $j \leftarrow 0$  **to**  $w - 1$  **Hacer**
  - 4:      $\vec{v}_{on-off-A_{(i+j*w)}} \leftarrow \vec{v}_{piloto_{(i)}}$
  - 5:   **Fin Para**
  - 6: **Fin Para**
  - 7: **Regresa**  $\vec{v}_{on-off-A}$
- 

OFDM ( $s_n$ ). El proceso para generar símbolos OFDM se muestra en la [figura 3.2](#), donde se aprecian las etapas requeridas para generar dichos símbolos como son: Formateo antipodal, Modulador QAM, Simetría Hermitiana, IFFT, Intervalo de guarda y el conversor D/A. En los siguientes párrafos se describe a detalle cada una de estas etapas. En el [apéndice C.1](#) se presenta el código implementado para la transmisión OFDM.

### Formateo antipodal

Esta etapa se encarga de dar un formato antipodal a la secuencia entrante. En la [figura 3.3](#) se observa el formato antipodal, el cual consiste en mapear las muestras de la secuencia entrante con valor '0' a un valor de '-1'. La secuencia saliente se expresa como  $\vec{v}_{antipodal} = [v'_0, v'_1, \dots, v'_{b*M-1}]$ . Esta conversión permite a la siguiente etapa realizar un mapeo en cuadratura, con al menos dos bits por símbolo QAM y una buena distancia entre cada

---

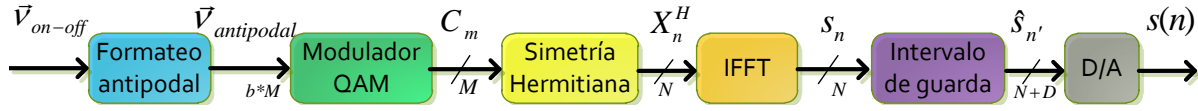


Figura 3.2: Diagrama general del transmisor.

punto de la constelación.

### Modulador QAM

Entre las técnicas de modulación aprobadas en el estándar para la tecnología PLC (IEEE P1901-2010), destaca la modulación QAM. En esta etapa se implementó la modulación QAM, la cual recibe de entrada una secuencia antipodal de  $b * M$  muestras, con un valor propuesto de  $b = 2$ , lo que implica un flujo de 2 bits por símbolo QAM. A la salida de esta etapa, se tiene un flujo paralelo compuesto por símbolos modulados en *fase* ( $I$ ) y en *cuadratura* ( $Q$ ) obtenidos de una constelación 4QAM. (Véase figura 3.4)

El vector de salida se construye agrupando en pares la secuencia antipodal, lo que resulta en una secuencia de  $M$  valores complejos de la forma  $C_m = a_m + jb_m$ . Donde  $a_m$  son las primeras  $M$  muestras y  $b_m$  representan las últimas  $M$  muestras de la secuencia entrante. El vector complejo resultante de esta etapa, es una señal analítica. El pseudocódigo para la construcción de símbolos 4QAM se presenta en el algoritmo 3.2.

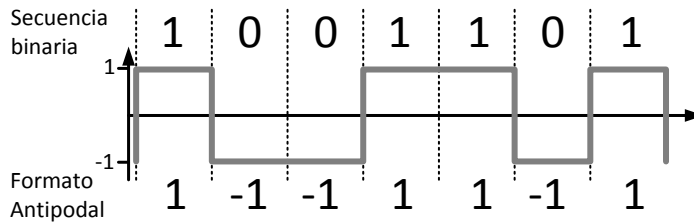


Figura 3.3: Formateo antipodal.

---

#### Algoritmo 3.2 Modulador 4QAM.

---

**Entrada:** Longitud del vector de símbolos QAM  $M$ ,  $b$  y la secuencia antipodal  $\vec{v}_{antipodal}$ .

**Salida:** Señal analítica QAM. ( $C$ ).

- 1: **Para**  $i \leftarrow 0$  **to**  $M - 1$  **Hacer**
  - 2:    $C_{(i,real)} \leftarrow \vec{v}_{antipodal}_i$
  - 3:    $C_{(i,imag)} \leftarrow \vec{v}_{antipodal}_{i+M}$
  - 4: **Fin Para**
  - 5: **Regresa**  $C$
-



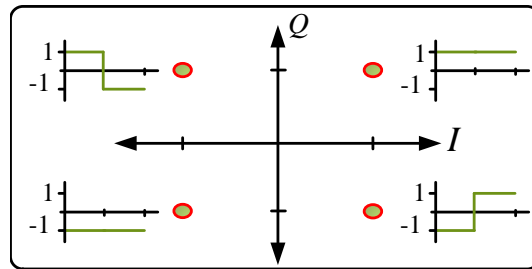


Figura 3.4: Constelación 4QAM.

### Simetría Hermitiana

En un módem PLC no resulta viable la transmisión de señales complejas. Esta etapa obtiene la simetría Hermitiana de la señal analítica, antes de calcular la IFFT, lo que resulta en una parte imaginaria de la señal prácticamente cero. Por lo tanto, el envío de la parte real es suficiente para la recuperación de la información en el receptor.

Sea  $C_M$  la secuencia entrante, al calcular la entrada con simetría Hermitiana se tiene una secuencia  $X_N^H = [0, C_{(2:M)}, 0, C_{(M:2)}^*]$ , donde  $C_{(2:M)}^*$  es el complejo conjugado de  $C_{(2:M)}$ , y  $(M : 2)$  denota la dirección de las muestras, que va de la última muestra hasta la segunda muestra (espejo). El pseudo-código mostrado en el [algoritmo 3.3](#) permite generar el vector  $X_N$ .

### IFFT

Esta etapa recibe una secuencia de  $N$  muestras complejas (vector con simetría Hermitiana), la cual se somete a la IFFT, resultando a la salida otra secuencia compleja con la parte imaginaria igual a cero. Sea  $\vec{s}_n = [s_0, s_1, \dots, s_N]$  la secuencia resultante y equivalente a un símbolo OFDM con  $N$  sub-portadoras.

En el presente trabajo, se emplea el algoritmo de la IFFT que fue diseñado por Tom

---

#### Algoritmo 3.3 Simetría Hermitiana.

---

**Entrada:** Longitud del vector de símbolos QAM  $M$ ,  $C$  y  $N$ .

**Salida:** Vector de la simetría Hermitiana.  $(X^H)$ .

- 1:  $X_0^H \leftarrow 0$
  - 2: **Para**  $i \leftarrow 0$  **to**  $M - 1$  **Hacer**
  - 3:    $X_i^H \leftarrow C_i$
  - 4:    $X_{i+M}^H \leftarrow \text{CONJ}(C_{M-i})$  //CONJ devuelve el complejo conjugado.
  - 5: **Fin Para**
  - 6: **Regresa**  $X^H$
-

Roberts en 1986 [32]. Este algoritmo realiza las operaciones aplicando el formato de punto fijo que recomienda *Texas Instruments* para el DSP TMS320C6416. Cabe mencionar que el algoritmo fue adaptado a la configuración del sistema desarrollado en este trabajo. En el [apéndice C.2](#) se presenta el código implementado para la construcción de símbolos OFDM.

### Intervalo de guarda

En esta etapa se agrega el intervalo de guarda al símbolo OFDM. Sea  $s_n$  la señal de entrada, a la salida se tiene una señal de la forma  $\vec{s}'_{n'} = [s_{N-D-1}, s_{N-D-2}, \dots, s_{N-1}, s_0, s_1, \dots, s_{N-1}]$  y una longitud  $N + D$ . La señal resultante se construye agregando de las últimas  $D$  muestras de  $\vec{s}_n$  al inicio de esta misma. En el [algoritmo 3.4](#) se presenta en el pseudo-código de esta etapa.

### Conversor D/A

En los objetivos del presente trabajo no se contempla la implementación del conversor D/A. Por tanto, esta etapa sólo se presenta con fines ilustrativos para el esquema de transmisión OFDM. En la práctica esta etapa realiza la conversión de una señal analógica a una discreta por medio del muestreo. Las especificaciones técnicas de este módulo se encuentran en el estándar para la capa física de la tecnología PLC (IEEE P1901-2010).

### 3.1.3. Construcción de tramas OFDM

En la [figura 3.2](#) no se contempla el proceso que permite armar las tramas OFDM (una trama OFDM es un conjunto de símbolos OFDM). En el presente trabajo se han diseñado dos tipos de tramas: la primera, empleada en el módulo de puesta a punto del sistema, nombrada trama de control. La segunda trama contiene los datos útiles, la cual ha sido nombrada trama de datos. A continuación se detalla el uso y construcción de ambas tramas.

---

**Algoritmo 3.4** Intervalo de guarda.

---

**Entrada:**  $\vec{s}_n$  y  $D$ .

**Salida:** Símbolo OFDM con su respectivo intervalo de guarda.  $(\vec{s}'_{n'})$ .

- 1: **Para**  $n \leftarrow 0$  **to**  $D - 1$  **Hacer**
  - 2:      $\hat{s}_n \leftarrow s_{N-D+n}$
  - 3: **Fin Para**
  - 4: **Para**  $n \leftarrow 0$  **to**  $N - 1$  **Hacer**
  - 5:      $\hat{s}_{n+D} \leftarrow s_n$
  - 6: **Fin Para**
  - 7: **Regresa**  $s_n$
-

### Trama de control

La trama de control se emplea en la puesta a punto del sistema, donde se aplican técnicas como: sincronización y estimación de canal. Por tanto, un buen diseño de la trama de control contribuye a lograr el mejor desempeño posible de un módem OFDM/PLC.

En el presente trabajo se propone una trama que beneficie a las técnicas antes mencionadas. Por lo cual, la trama aquí propuesta se divide en dos bloques diseñados de manera empírica de acuerdo a la observación experimental obtenida a lo largo del trabajo. A continuación se describen ambos bloques.

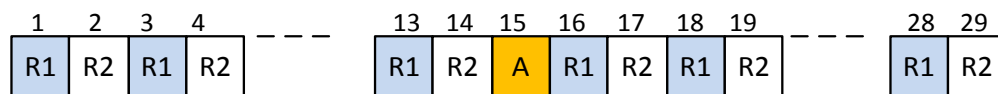
El primer bloque se utiliza para el proceso de sincronización burda y se construye intercalando 5 símbolos OFDM de tipo "R" y 5 símbolos piloto tipo "A".

El segundo bloque se diseñó para el proceso de sincronización fina y estimación de canal. Éste se construye con una secuencia de 10 símbolos piloto tipo "A".

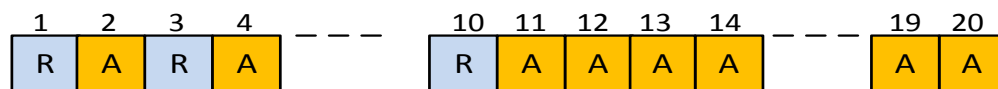
En la [figura 3.5 \(a\)](#) se ilustra el diseño de la trama propuesta para la fase de puesta a punto del sistema de prueba. La trama propuesta en el sistema de referencia desarrollado por G. Laguna en el 2011, véase [figura 3.5 \(b\)](#), y tiene un total de 29 símbolos OFDM, de los cuales uno es de tipo "A" y el resto de tipo "R". En la [figura 3.5 \(b\)](#) se presenta la trama diseñada para la etapa de puesta a punto en el presente trabajo. En la trama propuesta se construye con símbolos OFDM nombrados R1 y R2, formados de secuencias pseudo-aleatorias distintas. A pesar de ser un diseño eficiente la trama del sistema de referencia, resulta ser más complejo de construir y extenso que el alcance propuesto en el presente trabajo.

### Trama de datos

La trama OFDM para la transmisión de datos se construye de la misma manera que fue propuesta en el sistema de referencia [26]. La trama tiene la característica de llevar siempre de cabecera un símbolo piloto de tipo "A", seguido a este se colocan los símbolos que



(a) Propuesta del sistema de referencia.



(b) Propuesta del presente trabajo.

Figura 3.5: Trama de control.

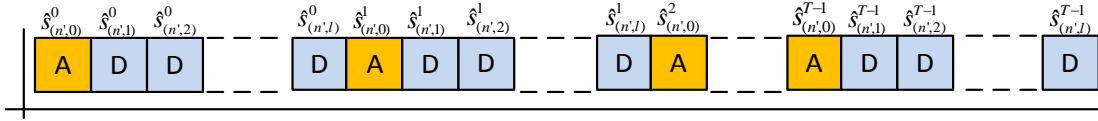


Figura 3.6: Trama OFDM de datos.

contienen la información útil (para efectos de simulación se transmiten símbolos de tipo "R").

La figura 3.6 ilustra la representación de una secuencia de  $T$  tramas OFDM, una trama se construye a partir de  $l + 1$  símbolos  $\vec{s}_{n'}$ . Siendo el primer símbolo un piloto y los  $l$  restantes de datos. Finalmente, se expresa la trama de símbolos OFDM como  $\vec{s}_{(n',l)}^T = [\vec{s}_{(n',0)}^t, \dots, \vec{s}_{(n',l-1)}^t, \vec{s}_{(n',l)}^t]$ , donde  $\vec{s}_{(n',l)}^t$  representa el  $l$ -ésimo símbolo OFDM insertado en la  $t$ -ésima trama transmitida.

### 3.1.4. Canal

El módulo del canal simula dos efectos negativos propios de un canal PLC, como el de ruido blanco Gaussiano y atenuación selectiva en frecuencia. A continuación se describe el proceso de simulación de ambos efectos.

#### Canal AWGN

Este módulo recibe de entrada una trama OFDM ( $\vec{s}_T$ ), a la cual se le agrega ruido blanco Gaussiano. El canal AWGN se simula bajo el modelo mencionado en el capítulo anterior. La señal resultante se expresa como  $\vec{s}_t = \vec{s}_t + \vec{n}_t$ , donde  $\vec{n}$  son muestras de ruido blanco Gaussiano, las cuales son una secuencia de  $T$  variables pseudo-aleatorias con distribución Gaussiana con media ( $\mu$ ) igual a cero y desviación estándar ( $\sigma$ ) igual a la raíz cuadrada de la potencia de la señal de ruido. El algoritmo 3.5 presenta el pseudo-código implementado para simular este canal.

---

#### Algoritmo 3.5 Canal AWGN.

---

**Entrada:** Trama OFDM  $\vec{s}_T$ , símbolos OFDM por trama  $T$  y valor SNR en dB.

**Salida:** Trama OFDM afectada por ruido. ( $\vec{s}_T$ ).

- 1:  $P_s \leftarrow \text{CalculaPotencia}(\vec{s}, T)$
  - 2:  $P_n \leftarrow P_s / (10^{(SNR/10)})$  // Calcula la potencia del ruido de acuerdo al valor SNR (dB)
  - 3: **Para**  $t \leftarrow 0$  **to**  $T$  **Hacer**
  - 4:  $\vec{s}_t \leftarrow \vec{s}_t + \text{RN}(0, \sqrt{P_n})$  //  $\text{RN}(\mu, \sigma)$  regresa una variable aleatoria con distribución Normal
  - 5: **Fin Para**
  - 6: **Regresa**  $\vec{s}_T$
-

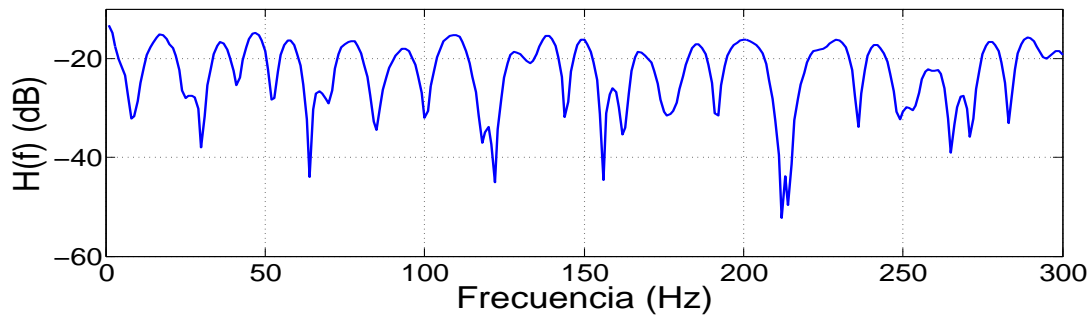


Figura 3.7: Canal selectivo en frecuencia empleado.

### Canal selectivo en frecuencia

Los alcances de este proyecto no contemplan la puesta en práctica de una transmisión real sobre un cableado eléctrico. Sin embargo, para efectos de evaluación de los algoritmos, se realizó en la simulación los efectos de un canal selectivo en frecuencia sobre la señal procesada en el DSP. Por lo tanto, en el presente trabajo se utilizó una máscara de un canal selectivo en frecuencia, obtenida de los resultados observados en el sistema de referencia [26]. Dicha máscara es un conjunto de 512 coeficientes de atenuación selectiva en frecuencia y, se muestran en la figura 3.7.

En el apéndice B.1 se encuentran los primero 153 coeficientes usados para la evaluación los algoritmos de estimación de canal, en total se usaron 512 coeficientes, el coeficiente 154 hasta el 512 tiene valor igual a cero.

En la plataforma de prueba se aplicó esta máscara en el módulo del receptor, esto con fines de evitar el cómputo de la convolución en el dominio temporal con la señal original. Con este ajuste se evitó afectar el desempeño del sistema en cuanto al tiempo de cómputo requerido durante la simulación Monte Carlo.

#### 3.1.5. Puesta a punto

Antes de describir este módulo, es importante aclarar que en el presente trabajo se da por hecho la existencia de un protocolo de comunicación, el cual permite el intercambio entre transmisor y receptor de la información para la configuración inicial de éstos, como lo es el uso o no de modulación adaptativa, la duración del símbolo OFDM, el número de símbolos OFDM por trama, la duración del intervalo de guarda, la estimación de la máscara OFDM/DMT (descrita más adelante), entre otros aspectos.

Ahora bien, el módulo de puesta a punto implementa procesos de un módem OFDM/PLC real para establecer la conexión transmisor/receptor sobre un canal simulado. Tales procesos son la sincronización, la estimación de canal y la estimación de la máscara OFDM/DMT. En

la [figura 3.8](#) se presenta el diagrama general de este módulo.

Este módulo se encuentra en el nodo receptor y, con el fin de simplificar la descripción, se presenta aquí de forma separada. La [figura 3.8](#) ilustra el seguimiento del proceso de puesta a punto en el nodo receptor, el cual se describe de forma breve en el siguiente párrafo.

El nodo receptor detecta la existencia de una señal sobre el canal, la cual después de ser digitalizada resulta en una trama OFDM contaminada con ruido y con un desfase temporal ( $\tilde{\mathbf{s}}_T$ ). El receptor toma una serie de ventanas de esta trama, para ser procesada en la etapa de sincronización burda, que realiza una estimación del desfase en tiempo aplicando la correlación del intervalo de guarda (correlación GI para futuras referencias). Una vez terminada la sincronización burda, el sistema realiza la primera corrección del inicio de la ventana FFT. Posteriormente, el sistema conmuta a la etapa de sincronización fina, la cual estima el desfase a detalle de la ventana FFT utilizando símbolos piloto como referencia. Si se encuentra desfase en la muestras, corrige la frecuencia de muestreo del oscilador local controlado por voltaje (de VCO, sigla del ingl. *Voltage Controlled Oscillator*), para garantizar la sincronía.

Hasta este punto, si el sistema no logra encontrar el inicio del símbolo OFDM adyacente, se da por finalizada la simulación. Por otro parte, si las condiciones se cumplen, el sistema pasa a la etapa de estimación de canal y finalmente a la etapa de estimación de la máscara OFDM/DMT. Los resultados de estas dos últimas etapas permiten al módulo Receptor recuperar la secuencia original con el mínimo número de errores, dicha disminución se debe, por una parte, al uso de modulación DMT, la cual que permiten discriminar las sub-bandas no viables para la transmisión (sub-bandas con alto nivel de atenuación). Por otra parte, la implementación de técnicas que intentan igualar el comportamiento del canal y con ello recuperar la señal original con el mínimo número de bits en error.

En la [figura 3.8](#) aparecen dos círculos enumerados, estos representan los puntos de conexión con el módulo Receptor. El círculo 1 otorga al módulo Receptor los coeficientes que caracterizan al canal selectivo en frecuencia, mientras que el círculo 2 da al módulo Receptor

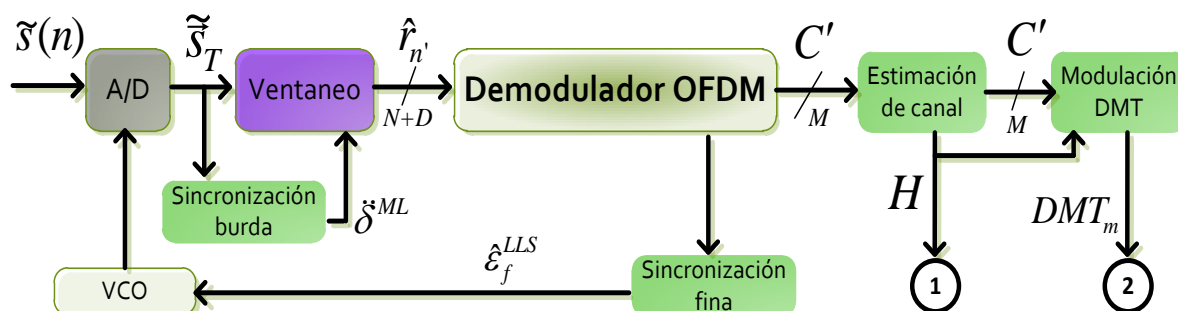


Figura 3.8: Esquema general del módulo 'Puesta a punto'.

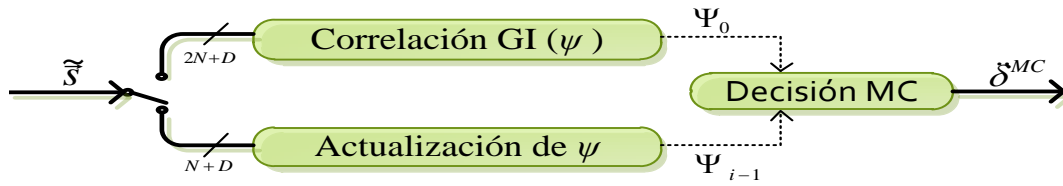


Figura 3.9: Diagrama a bloques de la sincronización burda.

la información de las bandas que contienen información útil.

Ahora que se ha explicado de forma general el funcionamiento de este módulo, en los siguientes párrafos se describe con mayor detalle la puesta en práctica de las etapas de éste. Dichas etapas son: sincronización burda, ventaneo, demodulador OFDM, sincronización fina, estimación de canal y modulación DMT. Los bloques de convertidor analógico a digital (A/D) y VCO no fueron implementados, dado que salen de los objetivos del proyecto.

### Sincronización burda

Esta etapa se encarga de estimar el inicio de cada símbolo OFDM, para lo cual, M. Crussière propone usar la correlación del intervalo de guarda. Esta técnica se basa en tomar una ventana de  $2N + D$  muestras, de las cuales se toman  $D$  para efectuar la correlación con el resto de la trama. La finalidad es hallar el valor máximo de correlación, el cual se presenta al correlacionar el intervalo de guarda con las últimas  $D$  muestras (recordando que las muestras del intervalo de guarda son una réplica de las últimas  $D$  muestras). El valor MC es una primera estimación al inicio del símbolo siguiente.

Dado que el cálculo de la correlación GI puede demandar un tiempo significativo, M. Crussière propuso que ésta se realice sólo una vez al inicio de cada trama de control y, el resto de iteraciones necesarias para una buena estimación, sólo actualice el vector de correlación con ventanas de  $N$  muestras.

Anteriormente se describió la trama de control (la cual se emplea en este módulo), de la cual, los primeros 10 símbolos se han designado a la etapa de sincronización burda. Para efectos de simulación, el índice al inicio de la trama OFDM ( $ix$ ) en el receptor, se define en forma pseudo-aleatoria.

Esta etapa se ha implementado en tres fases, véase figura 3.9, la primera calcula el primer valor para el vector de correlación GI ( $\Psi$ ). La segunda sólo actualiza los valores del vector de correlación. Finalmente, en la tercera se decide el valor máximo de correlación empleando la moda estadística. A continuación se describe la implementación de cada bloque.

- **Correlación GI.** A la entrada se recibe la señal afectada por el canal ( $\tilde{s}_t$ ), de la cual

se toma una ventana de  $2N + D$  muestras que nombraremos  $r_{2N+D}$  y sobre esta ventana se calcula el primer vector de correlación ( $\Psi_0(\delta_t)$ ). En el [algoritmo 3.6](#) se presenta el pseudo-código para obtener la correlación GI.

- **Actualización del vector de correlación ( $\Psi_i(\delta_t)$ ).** Aquí se toma una ventana  $r$  de  $N + D$  muestras de  $\widetilde{\vec{s}}_t$  con las que se actualiza el vector de correlación de manera iterativa, hasta un máximo de 7 iteraciones. El número de iteraciones depende de la longitud del primer bloque de la trama de control (el cual se ha definido en 10 símbolos OFDM), del cual se estima que los primeros 3 símbolos se ocupan en la fase anterior. Por lo tanto, se realizan un máximo de 7 actualizaciones del vector de correlación GI (valor estimado de forma experimental). El pseudo-código de esta fase se presenta en el [algoritmo 3.7](#).
- **Estimación MC.** Aquí se estima el valor de máxima correlación. Se obtiene un total de ocho vectores de correlación, por lo tanto existen ocho valores MC. El primero de los vectores se obtiene de la correlación GI, mientras que los otros siete son producto de la actualización del vector original.

En el presente trabajo se propone la moda estadística como criterio de decisión para elegir el mejor valor de estimación MC entre los ocho obtenidos. Es importante mencionar que en el sistema de referencia [26] calcula el promedio de sus valores estimados MC como la mejor estimación.

Los ocho valores MC son depositados en un vector de la forma  $\vec{\delta}_i^{MC} = [\delta_0^{MC}, \delta_1^{MC}, \dots, \delta_7^{MC}]$ , de este se ha de obtener el mejor valor MC de la forma

$$\ddot{\delta}^{MC} = Mo(\vec{\delta}_i^{MC}) \quad (3.1)$$

donde  $Mo$  es la moda estadística.

---

#### **Algoritmo 3.6** Correlación GI.

---

**Entrada:** Ventana de muestras  $\widetilde{\vec{r}}_{2N+D}$ , valor aleatorio inicial de muestreo  $ix$ ,  $N$ ,  $D$ .

**Salida:** Vector de correlación  $\Psi_0(\delta_t)$ .

- 1: **Para**  $t = ix$  **to**  $ix + D - 1$  **Hacer**
  - 2:      $\Psi_0(\delta_0) = \Psi_0(\delta_0) + r_t * r_{t+N}$
  - 3: **Fin Para**
  - 4: **Para**  $t = 0$  **to**  $N + D - 1$  **Hacer**
  - 5:      $\Psi_0(\delta_{t+1}) = \Psi_0(\delta_t) - r_{ix} * r_{ix+N} + r_{ix+D}$
  - 6: **Fin Para**
  - 7: **Regresa**  $\Psi_0(\delta_t)$
-



**Algoritmo 3.7** Actualización del vector de correlación.

**Entrada:** Ventana de muestras  $\vec{r}_{N+D}$ , vector de correlación GI  $\Psi_0(\delta_t), ix, N, D$ .

**Salida:** Vector de correlación actualizado  $\tilde{\Psi}$ .

- 1:  $\tilde{\Psi}_0 = \Psi_0$
- 2: **Para**  $i = 1$  **to** 7 **Hacer**
- 3:  $\tilde{\Psi}_i = \tilde{\Psi}_{i-1}$
- 4:  $\tilde{\Psi}_i(\delta_0) = \tilde{\Psi}_i(\delta_N + D - 1) - r_{ix} * r_{ix+N} + r_{ix+D} * r_{ix+D+N}$
- 5: **Para**  $t = 0$  **to**  $N + D - 1$  **Hacer**
- 6:  $\tilde{\Psi}_i(\delta_{t+1}) = \tilde{\Psi}_i(\delta_t) - r_{ix} * r_{ix+N} + r_{ix+D} * r_{ix+D+N}$
- 7:  $ix = ix + 1$
- 8: **Fin Para**
- 9: **Fin Para**
- 10: **Regresa**  $\tilde{\Psi}_i(\delta_t)$

Por lo tanto  $\delta^{MC}$  es la mejor aproximación para el índice de inicio para la ventana de los símbolos piloto, este valor se usa para ajustar el inicio del ventaneo de símbolos OFDM con su intervalo de guarda. El [algoritmo 3.8](#) presenta el pseudo-código de este bloque.

**Ventaneo**

Esta etapa tiene dos funciones: la primera es realizar el ventaneo de  $N + D$  muestras de la señal entrante. La segunda es ajustar dicho ventaneo por el retraso o adelanto estimado en la etapa de sincronización fina. Al final de esta etapa se tiene un vector para el símbolo recibido como  $\vec{r}_{n'} = \vec{s}_{n'} + n_{n'}$ , donde  $n_{n'}$  representa el ruido agregado al símbolo. En otras palabras, esta ventana es el equivalente a un símbolo OFDM más prefijo cíclico que es afectado por paso en el canal.

**Algoritmo 3.8** Estimación MC.

**Entrada:**  $\tilde{\Psi}_i(\delta_t), N, D$ .

**Salida:** Estimación MC  $\delta^{MC}$ .

- 1: **Para**  $t = 0$  **to** 7 **Hacer**
- 2:  $\bar{\delta}_i^{MC} = \text{MAX}(\tilde{\Psi}_i(\delta_t)) // \text{MAX}(y)$  regresa el valor máximo del vector  $y$
- 3: **Fin Para**
- 4:  $\delta^{MC} = \text{Mo}(\bar{\delta}_i^{MC}) // \text{Mo}(y)$  regresa la moda estadística del vector  $y$
- 5: **Regresa**  $\delta^{MC}$



Figura 3.10: Demodulación OFDM.

## Demodulador OFDM

En esta etapa se lleva a cabo el proceso de demodulación OFDM, véase la [figura 3.10](#), la cual se compone básicamente de 3 bloques: Exclusión GI, FFT y Simetría Hermitiana inversa. A continuación se describen los bloques antes descritos.

- **Exclusión GI.** Aquí se retira el intervalo de guarda de la ventana  $\vec{r}_{n'}$ , teniendo como salida el símbolo piloto  $\vec{r}_n$  que ha sido alterado por el canal.
- **FFT.** Una vez retirado el intervalo de guarda, se calcula la FFT sobre el símbolo recibido, lo que resulta una serie de valores complejos con simetría Hermitiana de la forma  $Y_n^H = [0, C'_m, 0, C_m^{*'}]$ , donde  $C'_m$  representa la señal analítica (secuencia de  $M$  símbolos QAM) afectada por el canal y,  $C_m^{*'}$  es el complejo conjugado de  $C'_m$ .
- **Simetría Hermitiana inversa.** El vector resultante después de computar la FFT, tiene información útil en la primera mitad. Por tanto, sólo se adquieren los  $M$  símbolos QAM y se desecha el resto. El vector resultante tiene la forma  $C'_m = a'_m + jb'_m$ , donde  $a'$  y  $b'$  son valores en formato antipodal afectados por el canal.

En el [algoritmo 3.9](#) se presenta el pseudo-código de esta etapa, en éste se incluyen los bloques antes mencionados. La implementación en el DSP se presenta en el [apéndice C.3](#).

---

### Algoritmo 3.9 Demodulación OFDM.

---

**Entrada:** Ventaneo (símbolo OFDM con intervalo de guarda)  $\vec{r}$ ,  $N$ ,  $D$ .

**Salida:** Señal analítica recibida  $C'$ .

- 1: **Para**  $n = 0$  **to**  $N - 1$  **Hacer**
  - 2:    $\vec{r}_n = \vec{r}_{n+D}$  //Exclusión GI
  - 3: **Fin Para**
  - 4:  $Y^H = \mathbf{FFT}(\vec{r})$  //Computa la FFT
  - 5: **Para**  $m = 0$  **to**  $M - 1$  **Hacer**
  - 6:    $C'_m = Y_m^H$  //Se retira la simetría Hermitiana
  - 7: **Fin Para**
  - 8: **Regresa**  $C'$
-

### Sincronización fina

En esta etapa se emplea la segunda mitad de la trama de control, la cual se construye con una secuencia de 10 símbolos piloto que permiten agilizar el proceso de sincronización fina y estimación de canal. Enseguida se describe de manera general el proceso aquí realizado.

En esta etapa se realiza la corrección del fase que aún persiste en la señal recibida, esta corrección se realiza en el dominio de la frecuencia ( $f$ ), es decir se aplica a la señal analítica ( $C'$ ). En esta etapa se realizan tres operaciones, véase [figura 3.11](#): la primera consiste en la corrección de fase de la señal analítica entrante. La segunda, se encarga de estimar el valor inicial del símbolo OFDM con base en la pendiente de la recta ajustada por mínimos cuadrados (LLS). Y finalmente, se realiza la corrección en el VCO. La implementación de estas operaciones se describe a continuación.

- **Corrección de fase.** Aquí se calcula la diferencia de fase entre el símbolo piloto transmitido y el recibido, es importante mencionar que el desfase en el dominio dual corresponde a un retraso en el dominio temporal. Para dicho cálculo se emplea la ecuación  $\Delta\varphi_k = \angle(X_A^H Y_p)$ , con  $X_A^T = C^*$  y  $Y_p = C'$ . Donde, el operador  $\angle$  representa la función angular. En el [algoritmo 3.10](#) se presenta el pseudo-código para realizar esta operación.
- **Estimación LLS.** Se realiza el cálculo del desfase que aun persiste en la señal, M. Crussière propone la estimación por mínimos cuadrados con base a la pendiente de la recta que resulta de las fases observadas en la señal analítica. De acuerdo con esto, si después de calcular la corrección de fase, la pendiente de la recta es positiva se dice que existe un adelanto en la ventana FFT. Por otro lado, si resulta una pendiente negativa se dice que hay un atraso. Y en el caso ideal, donde el valor de la recta tiende a cero, se dice que el muestreo esta sincronizado de manera perfecta. En el [algoritmo 3.11](#) se presenta el pseudo-código para realizar esta operación.

Es importante mencionar que esta técnica sólo resulta eficiente ante un canal plano, M. Crussière propone un modelo aún mas complejo para un canal PLC real. Si el sistema logra la sincronización, se procede a la etapa de estimación de canal, la cual se describe a continuación. En el [apéndice C.7](#) se presenta el código de la etapa de sincronización.

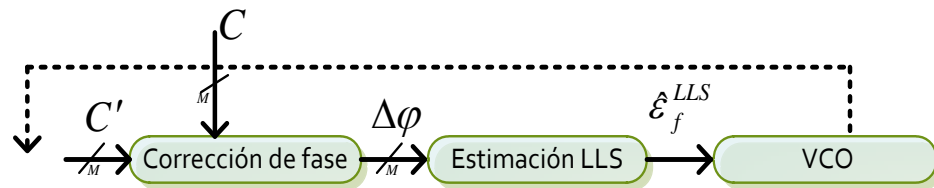


Figura 3.11: Etapa de sincronización fina.

**Algoritmo 3.10** Operación ángulo.

**Entrada:** Señal analítica original  $C$ , Señal analítica recibida  $C'$ , longitud del vector de símbolos QAM  $M$ .

**Salida:** Corrección de fase  $\Delta\varphi_k$ .

- 1:  $Z \leftarrow \text{CONJ}(C) * C'$  //Producto punto
- 2:  $\Delta\phi \leftarrow \text{ATAN}(\Im(Z)/\Re(Z))$
- 3:  $k \leftarrow 1$
- 4: **Mientras**  $k \leq M$  **Hacer**
- 5:    $sum \leftarrow |\Delta\varphi_{k-1} - \Delta\varphi_k|$  //  $|\cdot|$  representa el valor absoluto
- 6:   **Si**  $sum \geq \pi$  **Entonces**
- 7:      $\Delta\varphi_k \leftarrow \Delta\varphi_k + 2\pi$
- 8:   **Otro**
- 9:      $\Delta\varphi_k \leftarrow \Delta\varphi_k - 2\pi$
- 10:   **Fin Si**
- 11:    $k \leftarrow k + 1$
- 12: **Fin Mientras**
- 13: **Regresa**  $\Delta\varphi$

**Algoritmo 3.11** Estimación LLS para sincronización fina.

**Entrada:** vector de la corrección de fase  $\Delta\varphi_k$ , longitud del vector de símbolos QAM  $M$ .

**Salida:** Corrección de fase  $\epsilon_f^{LLS}$ .

- 1:  $aux1 \leftarrow 0$
- 2:  $aux2 \leftarrow 0$
- 3:  $aux3 \leftarrow 0$
- 4:  $aux4 \leftarrow 0$
- 5: **Para**  $k \leftarrow 1$  **to**  $M - 1$  **Hacer**
- 6:    $aux1 \leftarrow aux1 + k * \Delta\varphi_k$
- 7:    $aux2 \leftarrow aux2 + k$
- 8:    $aux3 \leftarrow aux3 + \Delta\varphi_k$
- 9:    $aux4 \leftarrow aux4 + j * j$
- 10: **Fin Para**
- 11:  $\epsilon^{LLS} \leftarrow 100 * (N/200\pi) * (M * aux1 - aux2 * aux3) / (M * aux4 - aux2 * aux2)$
- 12: **Regresa**  $\epsilon^{LLS}$

**Estimación de canal**

Una vez sincronizado el sistema se realiza la estimación de canal. En este trabajo se ha propuesto que la etapa de estimación de canal se realice después de la sincronización fina utilizando, como mínimo, un símbolo piloto y, como máximo hasta 9 símbolos pilotos. Este valor máximo y mínimo depende del número de símbolos empleados en la sincronización fina. Por ejemplo, si el proceso de sincronización fina elimina el desfase con primer símbolo piloto, los nueve restantes se emplean para estimación de canal.

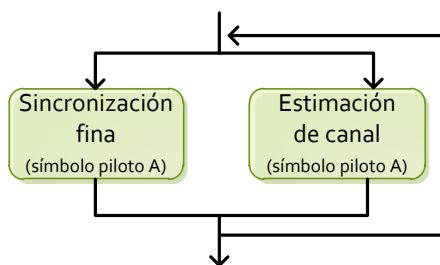
El sistema de referencia y el esquema de M. Crussière, proponen que la estimación de canal se realice después de la detección del único símbolo piloto, insertado en la trama de control (en la sincronización fina). Por tanto, se limita la estimación de canal con un sólo símbolo piloto. En la [figura 3.12](#) se presenta una comparación entre el esquema de referencia y el propuesto en el presente trabajo. La [figura 3.12 \(a\)](#) corresponde al sistema de referencia, el cual propone realizar en un ciclo el proceso de sincronización fina y la estimación de canal usando el mismo símbolo OFDM. En la propuesta propia ([figura 3.12 \(b\)](#)), se plantea el uso de dos ciclos separados, donde el primero realiza la etapa de sincronización fina. Si el sistema logra la calibración perfecta, se procede al segundo ciclo, el cual realiza la estimación de canal.

El proceso de estimación de canal tiene como objetivo, obtener una serie de coeficientes característicos de la respuesta en frecuencia del canal, con lo cual el receptor puede compensar los efectos de un canal selectivo en frecuencia sobre los símbolos OFDM.

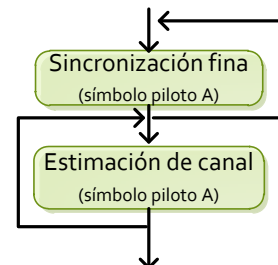
En el presente trabajo se han implementado las técnicas de estimación de canal por mínimos cuadrados (LS) y eliminación de ruido por ventaneo rectangular (VR) en el dominio temporal, ambas definidas en el capítulo anterior. Estas técnicas funcionan con base en la información del símbolo piloto tipo "A". A continuación se describe la implementación de dichas técnicas en el DSP.

- a) **Estimación LS.** La estimación por mínimos cuadrados es una técnica iterativa que se ejecuta cada inicio de trama, obteniendo información diferente entre cada iteración. Dicha información permite afinar los coeficientes de estimación. Se trata de minimizar la suma del error al cuadrado, obteniendo valores promedio para los coeficientes.

Como se mencionó anteriormente, en este proyecto se implementó el uso de símbolos piloto de tipo "A", bajo este esquema de comunicación se realiza el cálculo de los coeficientes de estimación dividiendo la señal analítica recibida  $C'_m$  entre la señal analítica original  $C_m$ . Como resultado se tiene un vector de coeficientes complejos ( $H_m$ ), con



(a) Esquema del sistema de referencia [26].



(b) Esquema propuesto.

**Figura 3.12:** Comparación del proceso de estimación de canal.

**Algoritmo 3.12** Estimación LS.

**Entrada:** Señal analítica original  $C_m$ , señal analítica recibida  $C'_m$  y longitud del vector de símbolos QAM  $M$ .

**Salida:** Coeficientes de estimación por mínimos cuadrados  $C^{LS}$ .

1: **Para**  $k \leftarrow 0$  **to**  $M - 1$  **Hacer**

2:  $H_k^{LS} \leftarrow C'_k / C_k$

3: **Fin Para**

4: **Regresa**  $H^{LS}$

$m = 0, 1 \dots M$ . En el [algoritmo 3.12](#) se presenta el pseudo-código implementado para el cálculo de la estimación LS.

- b) **Estimación VR.** El ventaneo rectangular es una técnica que contribuye a mejorar la calidad de los coeficientes, el cual consiste en pasar los coeficientes de estimación al dominio temporal, donde se sustituye por cero el valor de los coeficientes que se encuentran por encima de la máxima dispersión temporal del canal (se considera que más allá de la máxima dispersión, las muestras son de ruido). Posterior a esto, los coeficientes son regresados al dominio de la frecuencia. En el [algoritmo 3.13](#) se presenta el pseudo-código y en el [apéndice C.6](#) se presenta el código implementado.

**Modulación DMT**

La modulación DMT es otra manera de referirnos a la modulación OFDM cuando se aplica en tecnologías DSL (sigla del ingl. *Digital Subscriber Line*, 'línea de suscripción digital'). El objetivo de este módulo en el presente trabajo, es el de detectar las sub-bandas viables para la transmisión de datos sobre un canal selectivo en frecuencia.

Se considera banda viable aquella que su nivel de atenuación se encuentre por encima de un umbral calculado como

**Algoritmo 3.13** Estimación por ventaneo rectangular de la respuesta al impulso (VR).

**Entrada:** Coeficientes de estimación por mínimos cuadrados  $H^{LS}$ , longitud del vector de símbolos QAM  $M$  y  $D$ .

**Salida:** Coeficientes de estimación VR  $H^{VR}$ .

1:  $H^{VR} \leftarrow \text{IFFT}(H^{LS})$

2: **Para**  $k \leftarrow D/2$  **to**  $M - (D/2) - 1$  **Hacer**

3:  $H_k^{VR} \leftarrow 0$

4: **Fin Para**

5:  $H^{VR} \leftarrow \text{FFT}(H^{LS})$

6: **Regresa**  $H^{VR}$

**Algoritmo 3.14** Máscara DMT.

**Entrada:** Coeficientes de estimación  $H$ , y longitud del vector de símbolos QAM  $M$ .

**Salida:** Máscara DMT  $DMT\_masc$ .

```

1:  $v \leftarrow 10^{(\xi/20)}$ 
2: Para  $k \leftarrow 0$  to  $M - 1$  Hacer
3:   Si  $|H_k| \geq v$  Entonces
4:      $DMT\_masc_k \leftarrow 1$ 
5:   Otro
6:      $DMT\_masc_k \leftarrow 0$ 
7:   Fin Si
8: Fin Para
9: Regresa  $DMT\_masc$ 

```

$$v = 10^{\frac{\xi}{20}} \quad (3.2)$$

donde  $\xi$  es el nivel de atenuación (dB) máximo permitido, el valor propuesto en el sistema de referencia y adoptado en el presente trabajo, es de -35 dB. La información de cada banda se encuentra contenida en los coeficientes de estimación LS o VR, de los cuales se calcula su potencia expresada en dB y son sometidos uno a uno en un ciclo para determinar si su potencia se encuentra por encima del umbral. En este módulo se obtiene un vector o máscara DMT, que contiene valores '0' o '1' para sub-bandas no viables o viables, respectivamente.

La máscara DMT se aplica tanto en el transmisor como receptor dependiendo del número de sub-bandas viables será la información útil transportada por cada símbolo OFDM transmitido. El uso de técnicas como esta, permiten disminuir el número de bits erróneos, gracias a la discriminación de sub-bandas con alto nivel de atenuación. La implementación de la modulación DMT se presenta en el [algoritmo 3.14](#). El código implementado para la modulación y demodulación se encuentra en el [apéndice C.4](#) y [C.5](#), respectivamente.

### 3.1.6. Receptor

Antes de describir este módulo, es importante aclarar que éste es complemento del módulo de "puesta a punto". Por tanto, este módulo emplea etapas antes vistas. A continuación se describe de forma general el proceso que se lleva a cabo en este módulo.

El sistema de prueba simula un canal AWGN y selectivo en frecuencia, estos fenómenos alteran las señales OFDM. La señal recibida ( $\tilde{s}$ ) es muestreada y representa una trama OFDM con ruido ( $\tilde{s}_k$ ), en particular para este módulo la trama corresponde a una de datos. Es importante mencionar que la trama de datos se recibe después de finalizar satisfactoriamente el proceso de puesta a punto.

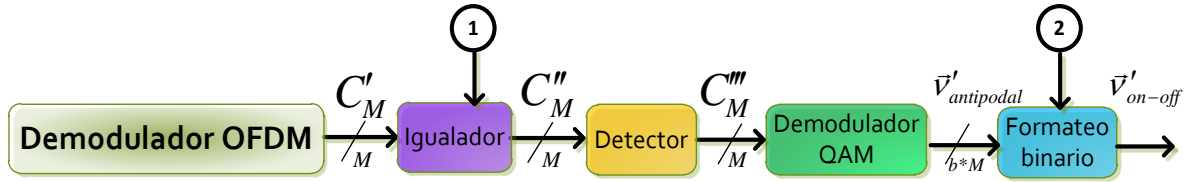


Figura 3.13: Módulos del receptor.

Dado que se conoce el inicio de cada símbolo OFDM insertado en la trama de datos, se realiza la demodulación OFDM de cada uno. El primer símbolo de cada trama de datos es un piloto de tipo "A", el cual se emplea para la sincronización fina y la estimación de canal. El resto de los símbolos son de datos y son sometidos a la etapa de igualación, detección, demodulación QAM y formateo binario, tal y como se observa en la figura 3.13. En los siguientes párrafos se describe la implementación de cada una de estas etapas.

### Igualador

Esta etapa es la responsable de compensar los efectos negativos causados por el canal selectivo en frecuencia. Para ello se emplean los coeficientes de estimación ( $H$ ) obtenidos en la estimación de canal. El objetivo de esta etapa es recuperar cada símbolo OFDM original.

Por lo tanto, a la entrada se tiene la señal analítica del símbolo entrante ( $C'$ ) y se obtiene a la salida los  $M$  coeficientes de estimación. Para compensar los efectos del símbolo entrante, se calcula

$$C''_{(l,k)} = \frac{C'_{(l,k)}}{H_{(l,k)}} \quad (3.3)$$

siendo  $C''_{(l,k)}$  la aproximación de  $l$ -ésimo símbolo sobre la  $k$ -ésima sub-portadora. Este modulo se aplica a todos los símbolos de datos. En el algoritmo 3.15 se presenta el pseudo-código.

---

#### Algoritmo 3.15 Igualador.

---

**Entrada:** Señal analítica entrante  $C'$ , coeficientes de estimación  $H$  y longitud del vector de símbolos QAM  $M$ .

**Salida:** Estimación de la señal analítica transmitida  $C''$ .

- 1: **Para**  $k \leftarrow 0$  **to**  $M - 1$  **Hacer**
  - 2:  $C''_k \leftarrow C'_k * H_k$
  - 3: **Fin Para**
  - 4: **Regresa**  $C''$
-



**Algoritmo 3.16** Demodulación y detección QAM.

**Entrada:** Estimación de la señal analítica transmitida  $C''$ , longitud del vector de símbolos QAM  $M$ ,  $b$

**Salida:** Secuencia antipodal recibida  $\vec{v}'_{antipodal}$ .

```

1: Para  $k \leftarrow 0$  to  $M - 1$  Hacer
2:   Si  $\Re(C''_k) < 0$  Entonces
3:      $\vec{v}'_{antipodal_k} \leftarrow -1$ 
4:   Otro
5:      $\vec{v}'_{antipodal_k} \leftarrow 1$ 
6:   Fin Si
7:   Si  $\Im(C''_k) < 0$  Entonces
8:      $\vec{v}'_{antipodal_k} \leftarrow -1$ 
9:   Otro
10:     $\vec{v}'_{antipodal_k} \leftarrow 1$ 
11:  Fin Si
12: Fin Para
13: Regresa  $\vec{v}'_{antipodal}$ 

```

**Detector**

En esta etapa se asignan símbolos QAM a la secuencia compleja entrante  $C''$ , se hace por medio del criterio de la distancia más corta o detección dura, asignando a cada muestra el valor complejo más probable de la constelación QAM. A la salida se tiene una serie de valores en cuadratura ( $C'''$ ).

**Demodulación QAM**

Este módulo realiza el mapeo de cada símbolo QAM ( $C'''_m$ ) a una secuencia antipodal ( $\vec{v}'_{antipodal}$ ), donde la parte real de los símbolos QAM representan los primero  $M$  valores y la parte imaginaria la otra mitad. El [algoritmo 3.16](#) contiene el pseudo-código de la demodulación QAM y la detección.

**Formateo binario**

En esta etapa se realiza un mapeo de la secuencia antipodal ( $\vec{v}'_{antipodal}$ ) a formato binario, es decir las muestras con valor '-1' toman el valor de '0' y las muestras con valor '1' se conservan igual. La secuencia resultante ( $\vec{v}'_{on-off}$ ) permite el calculo de la razón de error por bit (BER) en el siguiente módulo.

**3.1.7. Contador de errores**

Este módulo tiene la función de contar los bits recibidos que difieren de la secuencia original durante la ejecución del sistema de prueba. La información obtenida de este proceso

---

**Algoritmo 3.17** Contador de errores.

---

**Entrada:** Secuencia binaria original  $\vec{v}_{on-off}$ , secuencia binaria entrante  $\vec{v}'_{on-off}$ , longitud del vector de símbolos QAM  $M$  y  $b$ .

**Salida:** Razón de error por bit  $BER$ .

```

1:  $\epsilon \leftarrow 0$ 
2: Para  $i \leftarrow 0$  to  $b * M$  Hacer
3:   Si  $\vec{v}_{on-off_i} \neq \vec{v}'_{on-off_i}$  Entonces
4:      $\epsilon \leftarrow \epsilon + 1$ 
5:   Fin Si
6: Fin Para
7:  $BER \leftarrow \epsilon / (b * M)$ 
8: Regresa  $BER$ 

```

---

permite conocer el desempeño del sistema, para lo cual se ha propuesto como métrica de desempeño la curva BER vs. SNR. En los siguientes párrafos se describe la implementación de este módulo.

Como se mencionó anteriormente, este módulo tiene la tarea de contar el número de errores producidos durante la transmisión. Por lo tanto, se tiene como entrada la secuencia binaria original ( $\vec{v}_{on-off}$ ) y la recibida ( $\vec{v}'_{on-off}$ ) después de la detección y mapeo. Los bits de cada secuencia son comparados uno a uno de cada  $l$ -ésimo símbolo OFDM contenido en la trama. Al final de este conteo, el valor obtenido permite conocer la razón de error por bit de la trama. Esto se calcula por cada trama transmitida y se obtiene el promedio, lo que nos da una mejor estimación de la probabilidad de error existente sobre el canal simulado. El [algoritmo 3.17](#) presenta el pseudo-código para calcular el BER por símbolo OFDM.

## 3.2. Simulación Monte Carlo

Como se mencionó anteriormente, en el presente trabajo se implementaron los algoritmos de un módem OFDM/PLC en un DSP. Pero, dada la necesidad de evaluar el desempeño de la implementación, se ha empleado la simulación Monte Carlo. Dicha simulación permite evaluar el sistema de prueba descrito en la sección dos. En esta sección se presenta la descripción del proceso de simulación que permitió la evaluación de los algoritmos.

El sistema de prueba se compone de etapas que podrán aplicarse en un módem real y otras que solo son simuladas, en la [figura 3.14](#) se aprecia de forma general el proceso de simulación propuesto para la evaluación de los algoritmos de la capa física de un módem OFDM/PLC.

En la [figura 3.14](#) se presenta en forma de flujo de tiempo, los módulos descritos en la sección anterior como: transmisor, canal, puesta a punto y receptor. También se presenta de forma gráfica el envío de tramas de control y de datos.

---

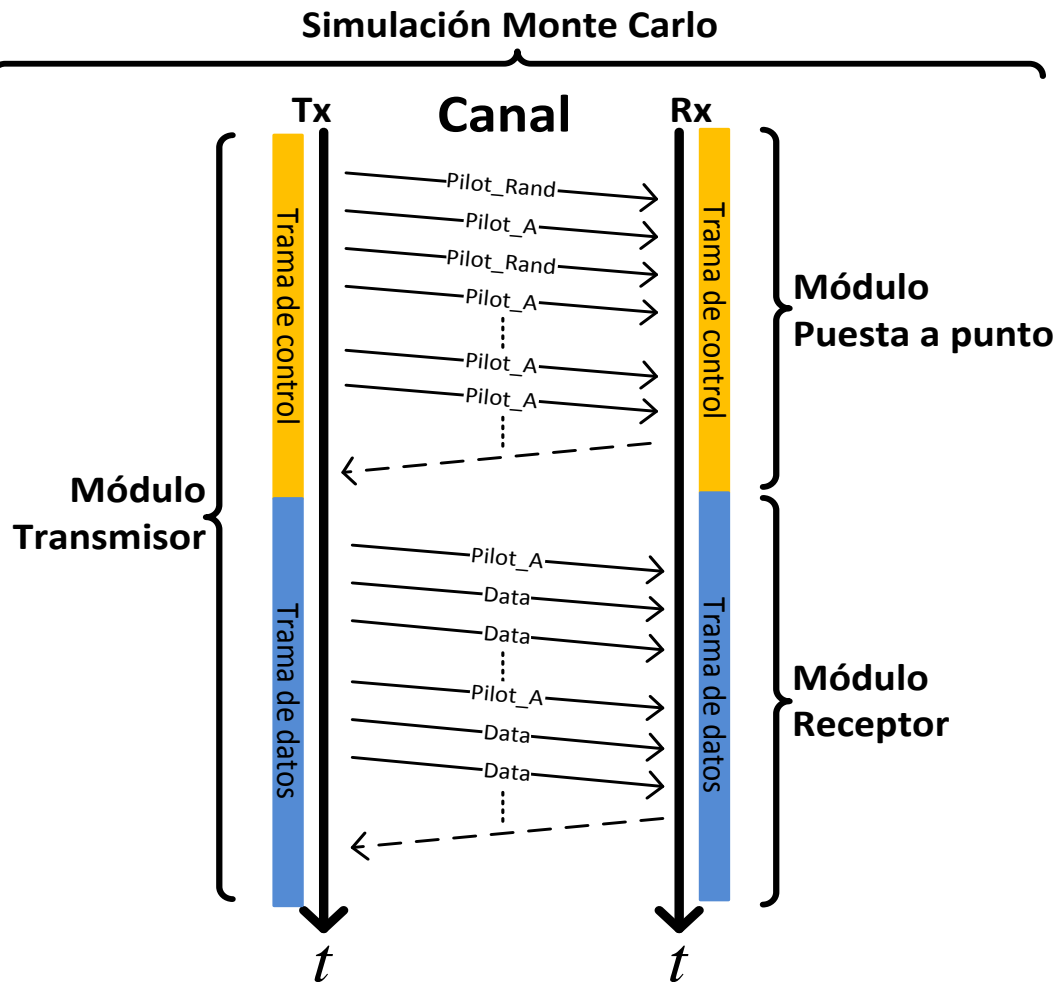


Figura 3.14: Simulación Monte Carlo.

El proceso de simulación se realiza en dos etapas importantes que se describen a continuación.

- a) En la primera interviene el módulo de transmisor, canal y puesta a punto. En esta etapa se simula la transmisión de la trama de control, a la cual se adiciona ruido y finalmente es recibida y procesada. Posterior a esta etapa se realiza un intercambio de información entre nodo emisor y receptor para establecer los parámetros de comunicación (modulación DMT, duración de tramas de datos, etc.), este intercambio se realiza con algún protocolo de conexión que se asume resuelto en el módem OFDM/PLC.
- b) La segunda etapa de simulación involucra los mismos módulos que la anterior, así como la información obtenida del protocolo de conexión. El objetivo general de esta etapa es

la transmisión, recepción y procesamiento de tramas de datos, en particular esta etapa permite evaluar el desempeño de los algoritmos del módem receptor, los cuales deben tener la capacidad de mitigar los efectos propios de una canal PLC. La información que resulta de esta etapa se expresa en la gráficas de la curva BER vs. SNR.

### 3.3. Plataforma de implementación DSK

En el presente trabajo de investigación se ha considerado usar el kit inicial de procesamiento digital de señales o DSK desarrollado conjuntamente por *Spectrum Digital* y *Texas Instrument* para la implementación de los métodos anteriormente descritos. El kit fue seleccionado con el objetivo de acelerar la implementación de los algoritmos de la capa física de un módem OFDM, partiendo de la tarjeta de desarrollo que utiliza como núcleo un DSP TMS320C6416T de la compañía *Texas Instrument*.

La tarjeta de desarrollo, utiliza comunicación USB para su conexión con la computadora sin necesidad de alguna configuración previa (*plug and play*). El DSK permite descargar y ejecutar en el DSP, el código de la aplicación de forma rápida, así como visualizar los datos en tiempo real.

Para iniciar el desarrollo en el kit empleado en este proyecto, se requiere conocer los elementos al DSP (de la familia TMS320C64T) y el ambiente de programación conocido como IDE Code Composer Studio, ambos proporcionados por *Texas Instrument*.

En el siguiente capítulo (capítulo 4), se presentan las pruebas y resultados de la plataforma de prueba presentada en este capítulo, donde se pone en práctica el simulador Monte Carlo para evaluar el sistema de prueba de forma modular y global.

---

## Capítulo 4

# Pruebas y resultados

---

En este capítulo se presentan las pruebas realizadas al sistema antes descrito. Las pruebas realizadas tienen el objetivo de demostrar el buen funcionamiento del sistema propuesto, por tanto se han desarrollado algunos experimentos que permitan evaluar la simulación de la transmisión sobre un canal PLC.

Los resultados obtenidos han sido comparados con el sistema de referencia [26] y con los resultados obtenidos en la simulación por computador (desarrollada en lenguaje C antes de la implementación en el DSP).

### 4.1. Planteamiento experimental

En la [figura 4.1](#) se presenta un escenario de transmisión de datos por un canal PLC entre dos nodos, conectados de extremo a extremo, estos se conectan a un módem OFDM/PLC para transmitir sobre el canal PLC. En el sistema de prueba descrito en el capítulo anterior, simula la transmisión/recepción de ambos módem OFDM/PLC de la [figura 4.1](#). Los resultados de la implementación en el DSP han sido comparados con los del "Simulador Monte Carlo para el esquema básico de comunicación OFDM sobre la red eléctrica", descrito en (G. Laguna et. al.) [26], donde se presenta la simulación de la transmisión de datos por un módem OFDM/PLC, dicho simulador fue desarrollado en MATLAB. El trabajo de G. Laguna, es

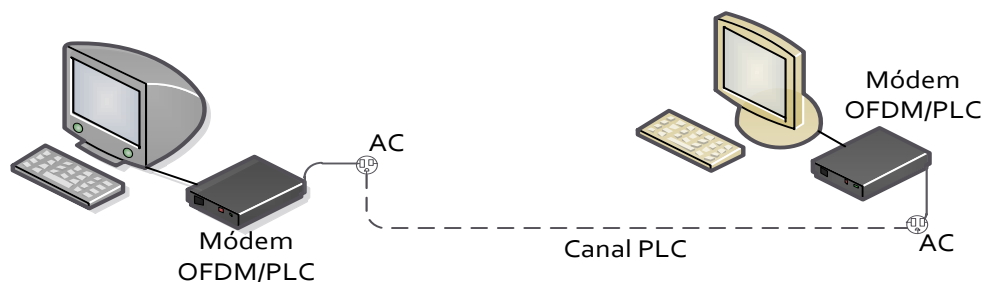


Figura 4.1: Escenario de simulación.

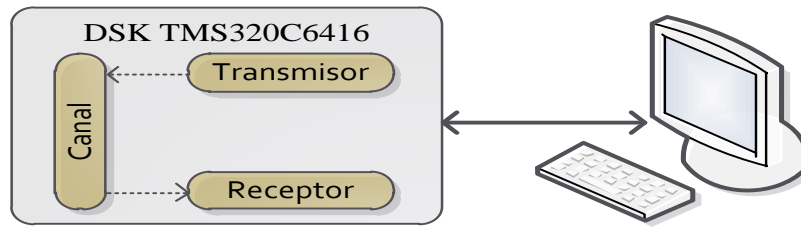


Figura 4.2: Escenario real de evaluación.

tomado como sistema de referencia dado que de dicho trabajo son tomados parámetros como la máscara de subred, número de sub-portadoras y otros. Además, del sistema de referencia se toma la idea de evaluar el sistema de prueba utilizando el método Monte Carlo.

La evaluación del sistema de prueba se llevó a cabo tras una serie de experimentos que permiten medir el funcionamiento y desempeño por módulo. Por lo tanto, la evaluación de los algoritmos de la capa física de un módem OFDM/PLC se realizó por módulo y de forma global, donde para la evaluación global se realizó una serie de experimentos con simulación Monte Carlo. A continuación se presentan las pruebas realizadas a los algoritmos implementados en el DSP, y posteriormente, se presentan los resultados de la evaluación global del sistema.

## 4.2. Pruebas y resultados individuales

En este apartado se presentan las pruebas y resultados de los algoritmos de forma individual, el objetivo de estas pruebas es ver el comportamiento y desempeño de algunos módulos del sistema de prueba. Las pruebas se realizaron a los siguientes módulos y/o algoritmos: generador de información fuente, formateo antipodal, modulador/demodulador QAM, igualador, modulador OFDM (usando la FFT), inserción del intervalo de guarda, sincronización burda y sincronización fina. En esta etapa la mayoría de las pruebas se han realizado en un canal libre de ruido, con el objetivo de comprobar el buen funcionamiento de cada algoritmo, y sólo algunas se realizaron con diferentes niveles de SNR.

En los siguientes párrafos se presentan las pruebas y resultados para los ya mencionados módulos. Es importante mencionar que las gráficas mostradas fueron obtenidas con ayuda del ambiente de desarrollo integrado CCS<sup>1</sup>, el cual permite visualizar en modo gráfico los datos generados por el DSP.

---

<sup>1</sup>La plataforma de desarrollo utilizada fue Code Composer Studio (CCS) versión 3.1, de Texas Instruments. El software va incluido en el DSK.

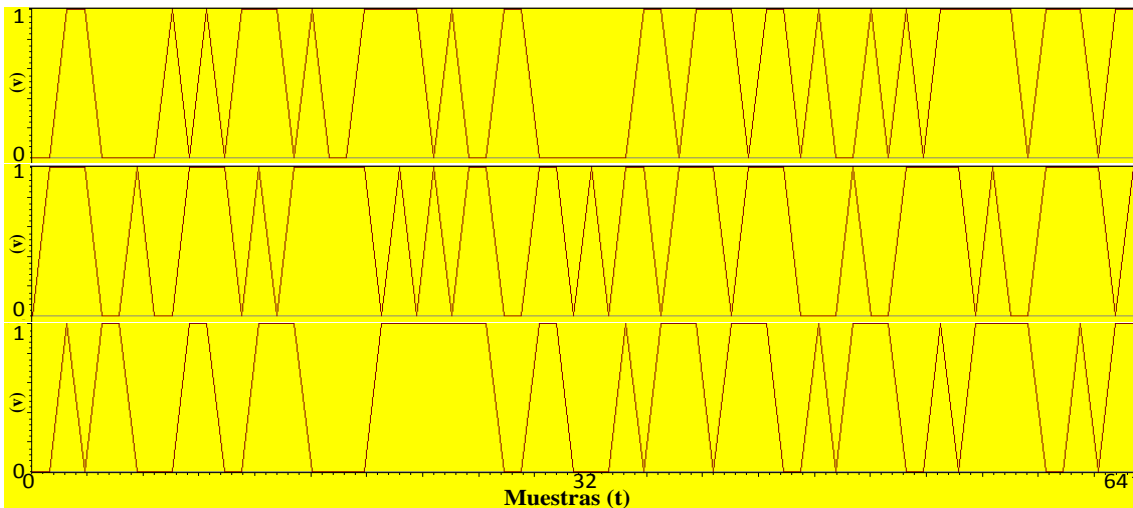


Figura 4.3: Secuencia binaria de tipo "R".

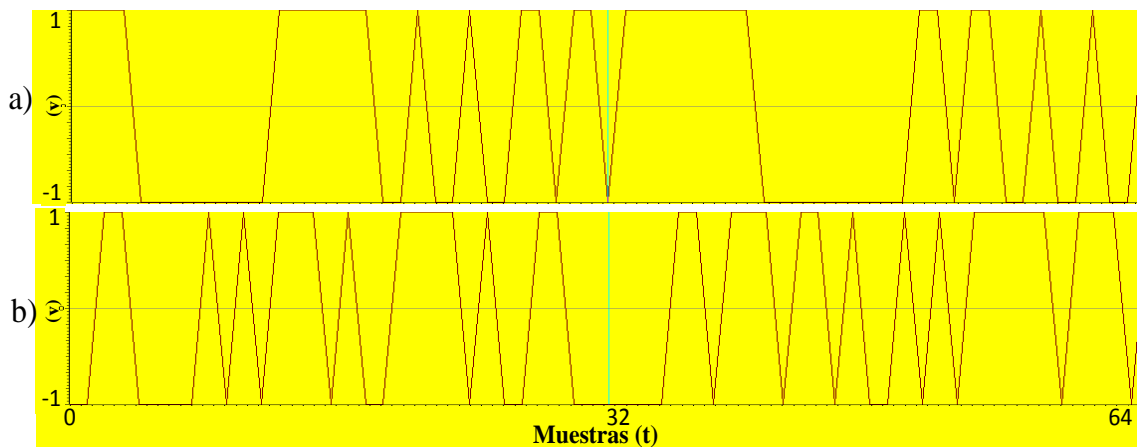


Figura 4.4: Secuencia antipodal de tipo "A" (a) y "R" (b).

#### 4.2.1. Generador de la información fuente

El sistema propuesto trabaja con dos tipos de secuencias binarias, una constante ("A") y la otra pseudo-aleatoria ("R"). En la [figura 4.3](#) se presentan 3 secuencias pseudo-aleatorias de 64 muestras binarias.

#### 4.2.2. Formateo antipodal

En la [figura 4.4](#) se presentan dos secuencias en formato antipodal, la superior corresponde a una secuencia del piloto y la inferior a una de tipo "R".

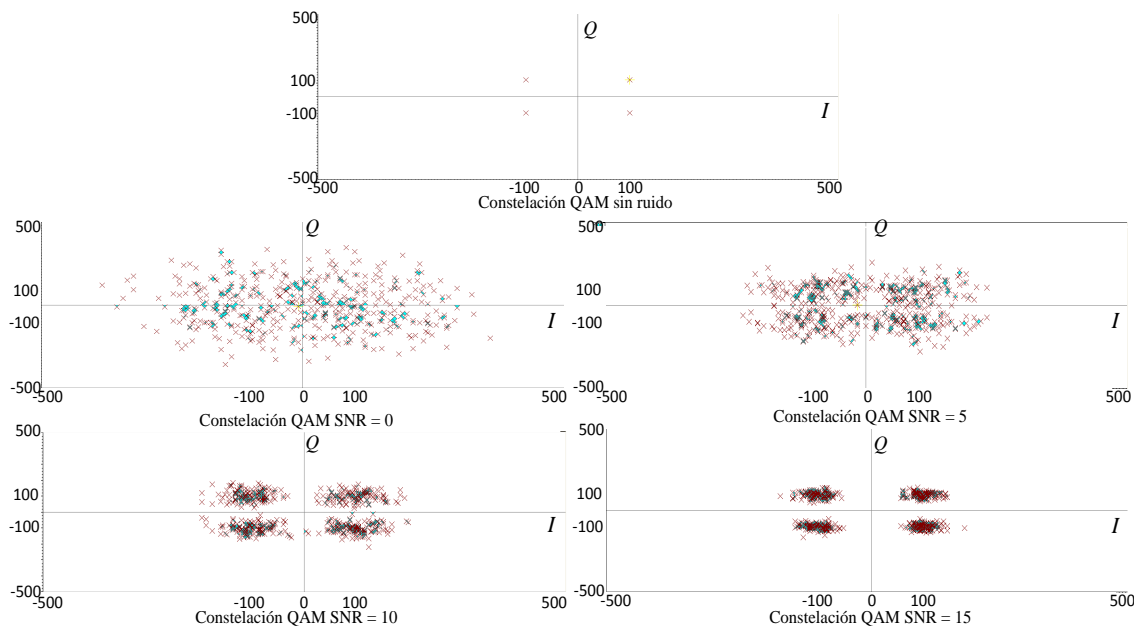


Figura 4.5: Constelación QAM y los efectos de un canal AWGN.

### 4.2.3. Modulación/demodulación QAM

El algoritmo de modulación/demodulación primero fue evaluado en un ambiente sin ruido, esto con el fin de asegurar el buen funcionamiento. Posteriormente, los símbolos QAM fueron afectados simulando a un canal AWGN. En la figura 4.5 se aprecia una secuencia de una constelación QAM afectada por diferentes niveles de SNR (0, 5, 10 y 15 dB). La constelación de símbolos afectados por un nivel de SNR = 0, presenta una gran variación en los símbolos recibidos. Se aprecia como disminuye la variación conforme aumenta el nivel de la relación señal a ruido.

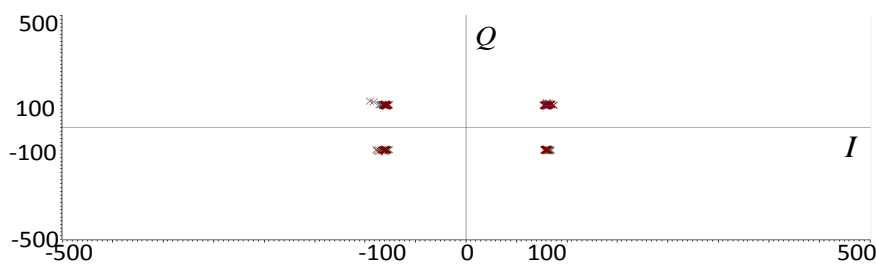


Figura 4.6: Igualación de la señal analítica.



#### 4.2.4. Igualador

El proceso de igualación intenta compensar, a pesar del ruido, la distorsión sufrida por la señal analítica. En la [figura 4.6](#) se presenta el resultado de dicho proceso sobre la constelación de la [figura 4.5](#), sometida a un valor de  $\text{SNR} = 0$ . Es importante aclarar que la variación en los símbolos recibidos se debe tanto a distorsión como al ruido, en la práctica no es trivial separar los efectos para compensarlos en forma aislada.

#### 4.2.5. Modulación OFDM

La modulación OFDM emplea los algoritmos de simetría Hermitiana y de IFFT, con los cuales se generan los símbolos OFDM. En la [figura 4.7](#) se gráfica el símbolo piloto de tipo "A" del sistema de referencia y del DSP. A pesar de contar con la misma secuencia binaria, se tiene diferencia con respecto a la escala, esto se debe a que en el DSP los algoritmos fueron programados utilizando aritmética de punto fijo con una escala de 1 por 100 (es decir un valor de 1 se mapea a un valor de 100), mientras que en la PC (MATLAB) se utilizó aritmética flotante. La implementación de la transformada rápida de Fourier con aritmética de punto fijo, produce una pérdida en la precisión de la señal resultante pero dicha pérdida no fue significativa para desempeño global del sistema.

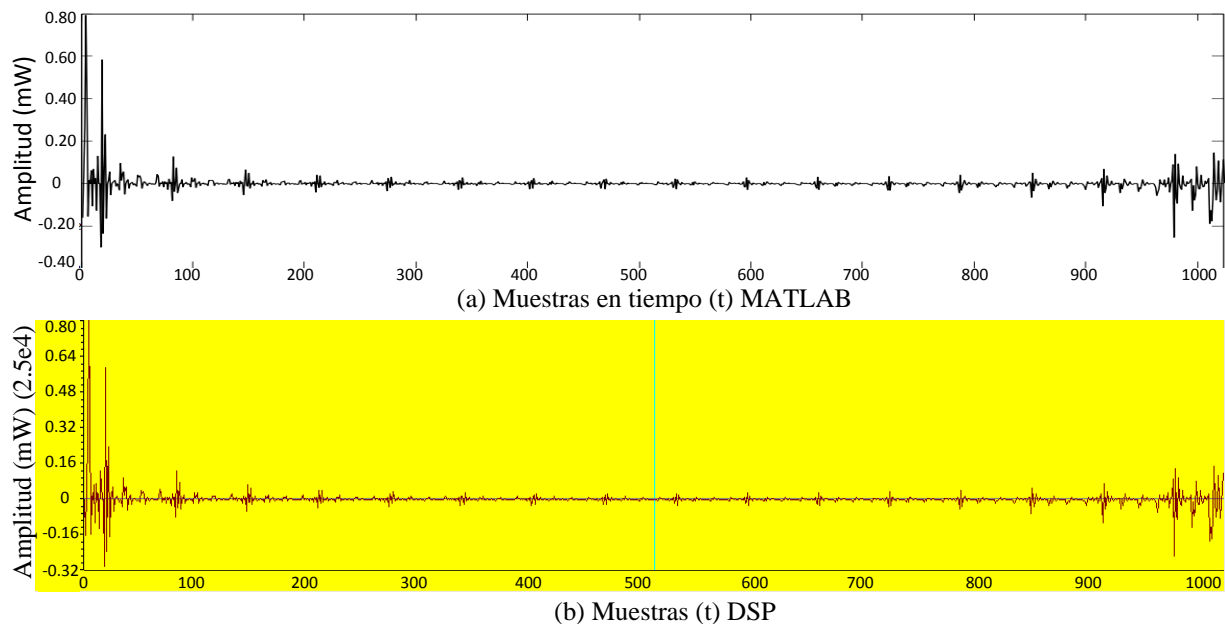


Figura 4.7: Símbolo piloto OFDM del sistema de referencia (a) y del DSP (b).

### 4.2.6. Inserción del intervalo de guarda

En la [figura 4.8](#) se presenta el símbolo piloto con su intervalo de guarda, esta prueba se realizó para comprobar el buen funcionamiento de este algoritmo. En la [figura 4.9](#) se presenta el efecto del canal AWGN sobre el símbolo piloto simulado en el DSP empleando diferentes valores de SNR (0, 5, 10 y 15 dB).

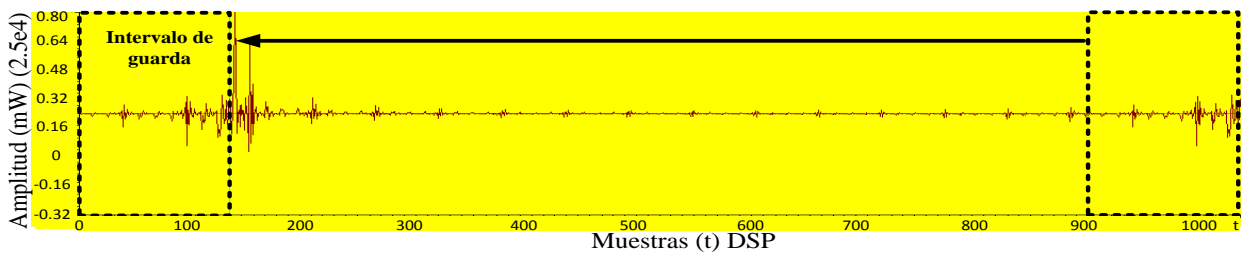


Figura 4.8: Símbolo OFDM con su intervalo de guarda.

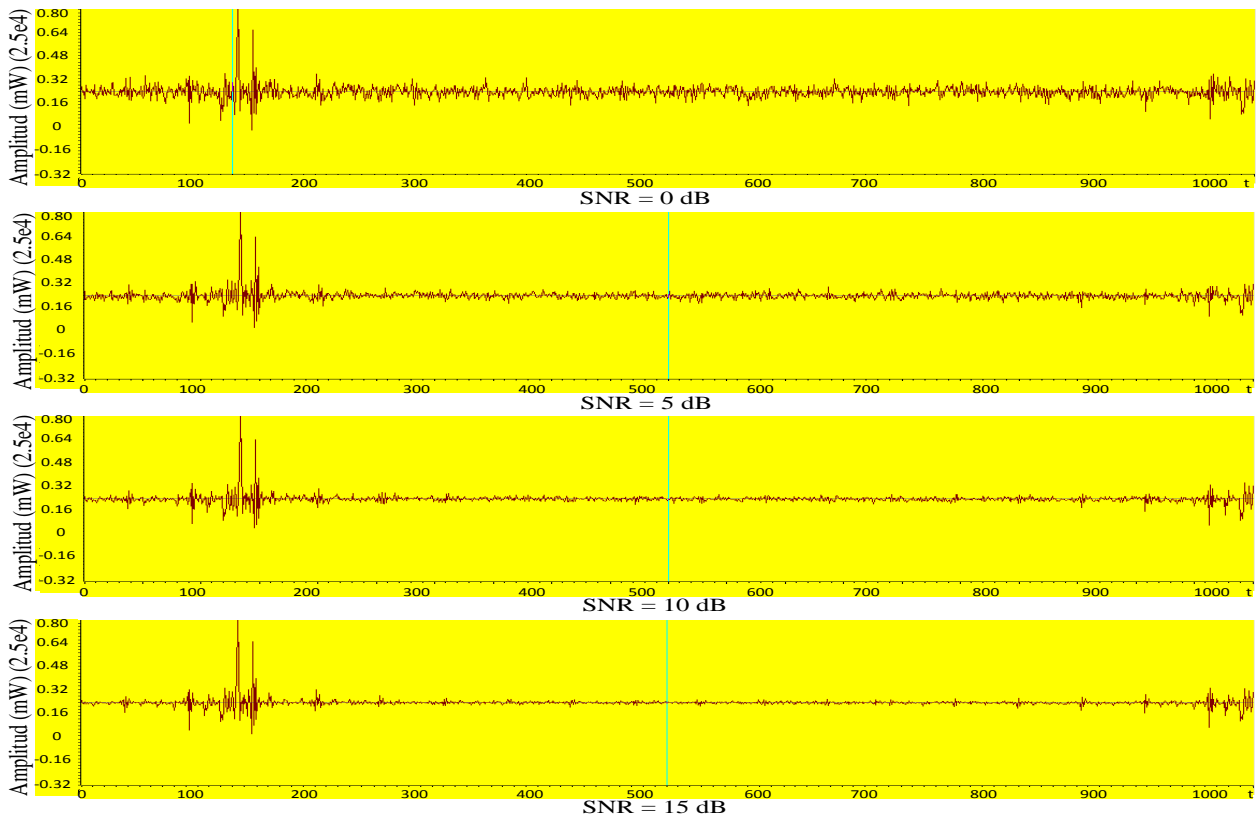


Figura 4.9: Símbolo OFDM con intervalo de guarda alterado por un canal AWGN.

### 4.2.7. Sincronización burda

Los resultados de la etapa de sincronización burda fueron comparados con los del simulador de G. Laguna. Recuérdese que, en el presente proyecto se propuso una trama de control distinta a la empleada en el sistema de referencia, así como el cambio en cuanto al modo de obtener el mejor valor del desfase de los símbolos OFDM.

En la [figura 4.10](#) se presentan los resultados del sistema de referencia, el cual realiza una simulación por computadora usando Matlab y en la [figura 4.11](#) se muestran los resultados generados por el DSP. Ambas gráficas presentan los resultados de la sincronización burda por correlación del intervalo de guarda en un canal AWGN (se evaluaron los valores de SNR 0, 5, 10 y 15 dB). Es importante mencionar que en este ejemplo, la señal evaluada tenía un

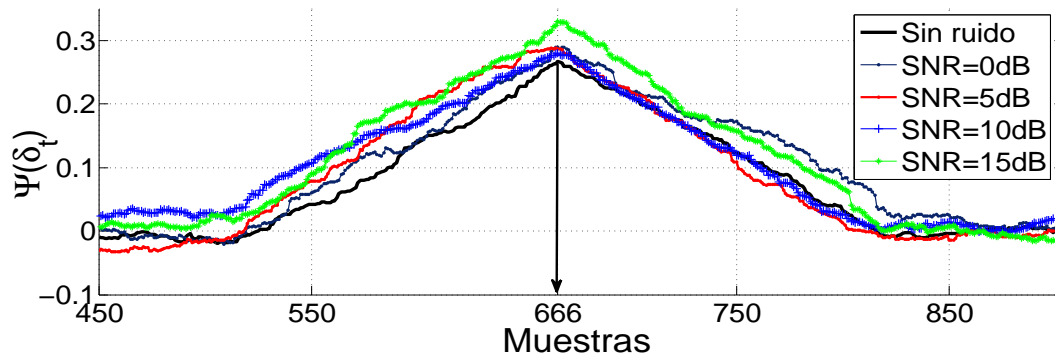


Figura 4.10: Correlación GI con canal AWGN, simulación por computadora (Matlab).

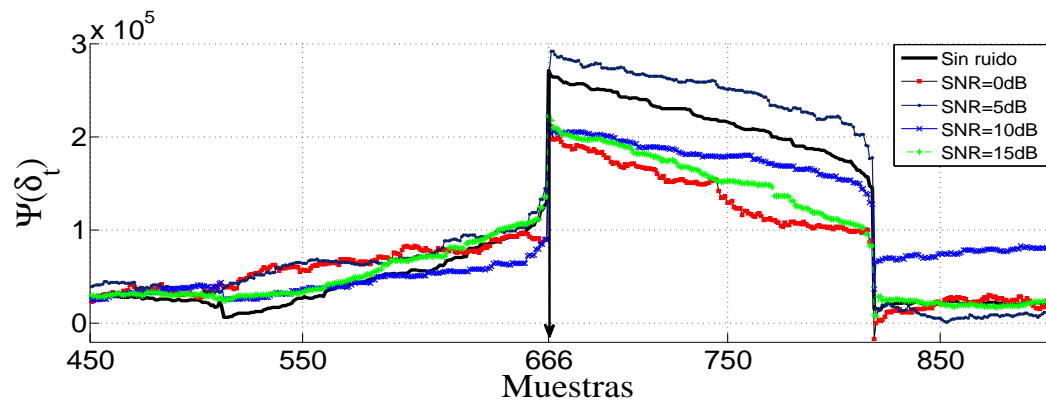


Figura 4.11: Correlación GI con canal AWGN en el DSP.

desfase de 666 muestras. Los resultados del DSP presentan ciertas diferencias gráficas con respecto al sistema de referencia, lo cual se debe a los ajustes propuestos y a la implementación bajo aritmética de punto fijo. Pero cabe mencionar, que el desempeño del algoritmo de sincronización burda en el DSP es similar al de Matlab, esto con respecto a la precisión al estimar el valor de MC.

Cabe mencionar que los resultados que se muestran en la [figura 4.11](#) presentan una escala en el valor de MC, superior con respecto a los valores de la [figura 4.10](#), esto se debe al uso de aritmética en punto fijo con escala de 1 por 100.

Los valores de estimación para el retraso (desfase)  $\delta^{ML}$  del sistema de referencia y del DSP se observan en la [tabla 4.1](#), donde se aprecia el buen funcionamiento del sistema implementado en el DSP. El diseño de la trama de control dispone de 10 símbolos para la correlación burda, pero los resultados prácticos muestran que al usar sólo 6 símbolos OFDM (intercalando símbolos de tipo "A" y "R") se tiene el mismo desempeño que al usar 10 como inicialmente se había propuesto. Por lo tanto, se tiene un ahorro de 4 símbolos en compración con el sistema de referencia.

#### 4.2.8. Sincronización fina

Las pruebas de la etapa de sincronización fina se realizaron empleando una sesión con señales sin ruido y otra con señales contaminadas por ruido blanco Gaussiano. Los resultados obtenidos muestran que el sistema tiene la capacidad de sincronización sin desfase alguno en un ambiente libre de ruido. Por otra parte, en un canal AWGN el sistema permite la sincronización si el valor de SNR se encuentra por encima de los 8 dB. En la [figura 4.12](#) se muestran los resultados obtenidos en el DSP, si se observa la [figura 4.12 \(b\)](#), se puede ver una pendiente positiva, este comportamiento se describe en el siguiente parrafo.

Es importante mencionar que existe una relación entre el corrimiento de fase y temporal.

[Tabla 4.1](#): Estimación burda por correlación GI para un desfase de 666 muestras (señal analítica).

| SNR<br>(dB) | $\tilde{\delta}^{ML}$ (Estimación del retraso) |                  |
|-------------|--|------------------|
|             | Matlab (10 Símbolos)                           | DSP (6 Símbolos) |
| 0           | 665  | 667              |
| 5           | 665  | 667              |
| 10          | 666  | 666              |
| 15          | 666  | 666              |
| Sin ruido   | 666  | 666              |

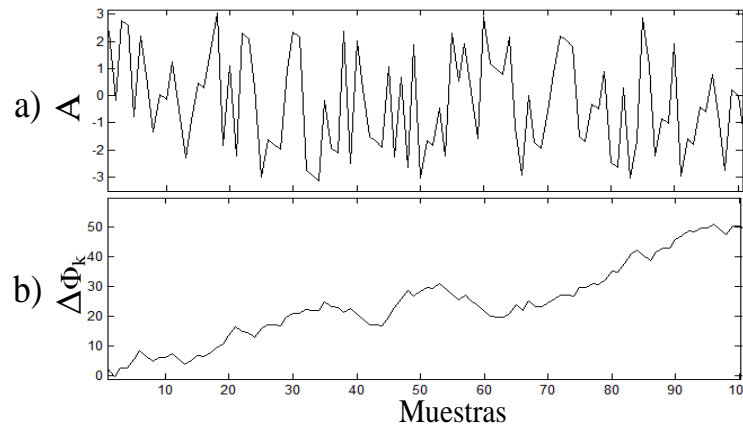


Figura 4.12: Operación angular (a) y corrección de fase (b) en el DSP.

Es decir, si en el dominio temporal se presenta un adelanto, esto se verá reflejado como una pendiente negativa en la corrección de fase, véase figura 4.13. Por otra parte, si se presenta un retardo en el dominio temporal, la recta resultante en la corrección de fase tendrá una pendiente positiva, véase figura 4.14.

### 4.3. Evaluación global

Una vez probados los módulos por separado, se evaluó el sistema en forma global. Las pruebas realizadas se hicieron con el objetivo de analizar el desempeño de los algoritmos en conjunto usando la curvas BER vs. SNR. Se realizó una evaluación por simulación utilizando el método Monte Carlo (descrito en el capítulo 3), en la tabla 4.2 se presentan los parámetros iniciales de configuración del experimento. En total se procesaron 10,240,000 bits, esto con la finalidad de tener una mejor aproximación de la media estadística para la probabilidad de

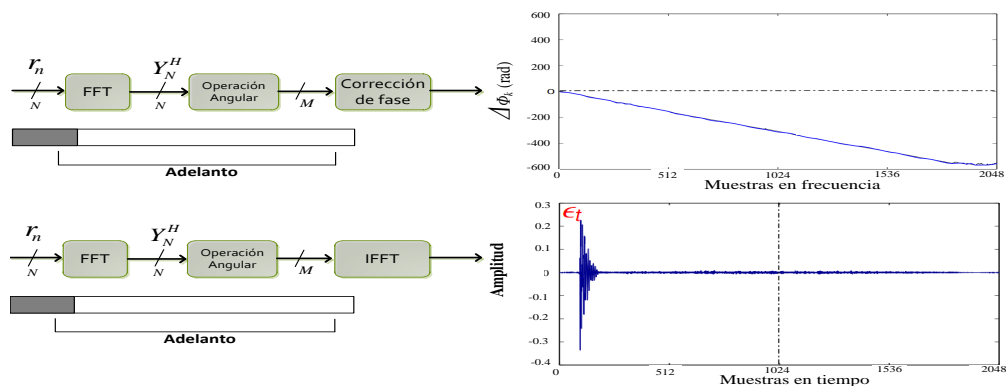


Figura 4.13: Adelanto de la ventana FFT.

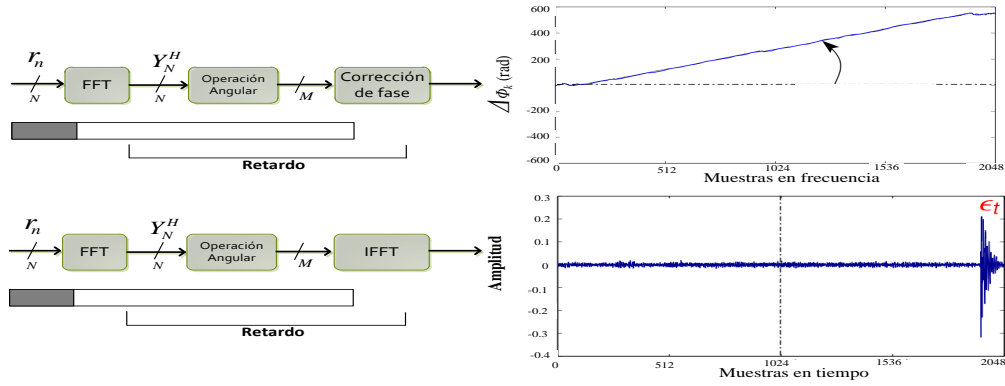


Figura 4.14: Retardo de la ventana FFT.

error.

La funcionalidad del sistema se ha puesto a prueba tanto con un canal AWGN como con uno selectivo en frecuencia. En el resto de esta sección se presentan los resultados obtenidos por las técnicas de estimación de canal por mínimos cuadrados (LS) y estimación por ventaneo rectangular de la respuesta al impulso (VR). La presentación de resultados se ha separado en dos apartados, uno para las pruebas y resultados del canal AWGN y el otro para los resultados del canal selectivo en frecuencia.

### 4.3.1. Pruebas y resultados del canal AWGN

Los resultados que se presentan en este apartado, son los resultados de la simulación del canal AWGN. La potencia de la señal de ruido se determinó según el valor de SNR deseado (ver el cálculo para obtener la potencia del ruido en el capítulo 3).

Tabla 4.2: Parámetros iniciales del sistema de prueba.

| Parámetro                               | Valor   |
|---|---------|
| Sub-portadoras (Puntos FFT)             | 1024    |
| Símbolos QAM                            | 512     |
| Bits por símbolo QAM                    | 2       |
| Muestras GI                             | 153     |
| SNR                                     | 0-15 dB |
| Símbolos OFDM por trama OFDM            | 100     |
| Tramas OFDM por experimento Monte Carlo | 10      |
| Experimentos Monte Carlo                | 10      |

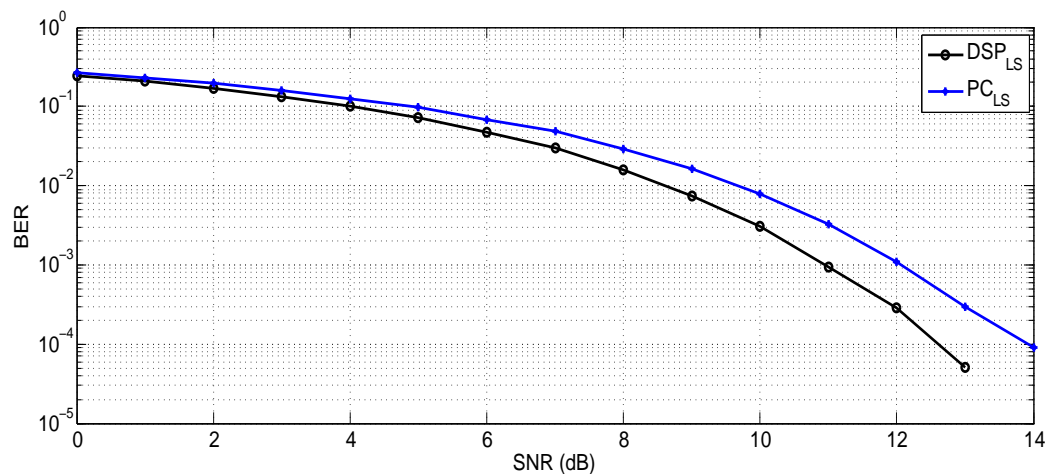
### Evaluación del sistema con estimación LS

Los resultados logrados en el DSP al evaluar el sistema de prueba usando estimación LS, han sido comparados con los obtenidos en simulación por computadora (usando lenguaje C), aplicando aritmética de punto flotante. La comparación se aprecia en la [figura 4.15](#), donde se observa el buen desempeño de la puesta en práctica en el DSP.

Los resultados obtenidos con el DSP usando aritmética de punto fijo, presentan un mejor desempeño sobre los resultados de la simulación por computadora que emplea lenguaje C, con representación de punto flotante. Al evaluar el desempeño de ambos sistemas en 12 dB de SNR, se aprecia que la simulación con lenguaje C presenta un bit en error de cada mil y, en el DSP se tienen tres errores de cada 10 mil. El DSP presenta un mejor desempeño al implementar la técnica LS, esto se debe al uso de representación en punto fijo, lo cual funciona como un filtro para valores pequeño que se agregan en el canal AWGN.

### Evaluación del sistema de prueba con estimación VR

Las pruebas realizadas al sistema en el DSP, usando el ventaneo rectangular a la respuesta al impulso de la señal, para la eliminación de ruido en la estimación de canal, muestra resultados muy parecidos a los de la simulación por computadora (en C y Matlab ambos en representación flotante), tal y como se aprecia en la [figura 4.16](#). La gráfica de los resultados muestra el mismo desempeño de los tres sistemas comparados. Por lo tanto, el desempeño de la puesta en práctica en el DSP de la estimación por ventaneo rectangular fue el deseado.



[Figura 4.15](#): Evaluación en el DSP vs. PC (lenguaje C) aplicando estimación LS.

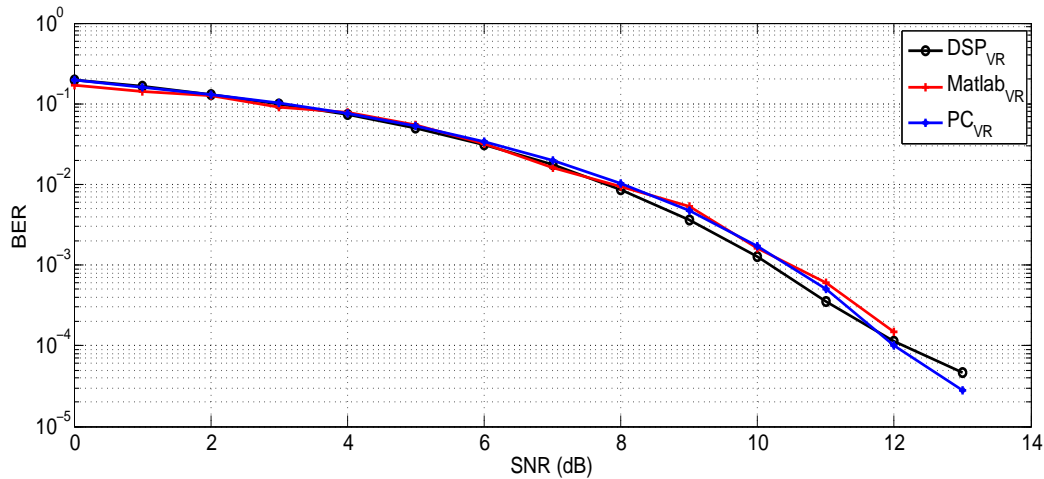


Figura 4.16: Evaluación por simulación por computadora (Matlab y C) vs. DSP de la estimación VR.

### Análisis de resultados de estimación LS vs. VR en un canal AWGN

De acuerdo a la literatura revisada, la técnica de estimación de canal por mínimos cuadrados resulta altamente vulnerable a niveles altos de ruido. Por otra parte la estimación de canal por ventaneo rectangular en el dominio temporal permite limpiar de ruido los coeficientes de estimación de canal, lo cual, presenta un mejor desempeño. La técnica VR, requiere de información a priori sobre la dispersión de la respuesta al impulso del canal, con ello tiene la capacidad de limpiar (poner a cero) la región de la respuesta al impulso que se sabe es ruido. En la figura 4.17 se puede observar la certeza de la teoría sobre el uso de técnicas para disminuir el ruido, mostrando un mejor desempeño la estimación VR ante LS.

### Evaluación y comparación teórica

Una forma de saber si un sistema funciona de manera adecuada, es comparar los resultados experimentales obtenidos con los resultados teóricos esperados. Como referencia, los resultados del DSP han sido comparados con la curva teórica QAM. La equivalencia QAM pero con OFDM no es exacta, pero da una idea del desempeño logrado. En la figura 4.18 se pueden comparar los resultados logrados con el DSP de las técnicas de estimación de canal LS y VR con respecto de la curva teórica de QAM. El hecho de que presente un mejor comportamiento el algoritmo VR ante la curva teórica, se debe al uso de estimación y eliminación de ruido con información adicional que proporciona la dispersión temporal del canal, dicha información se conoce antes de la transmisión. En el presente trabajo se tomo como referencia para la dispersión del canal la presentada por G. Laguna en [26].



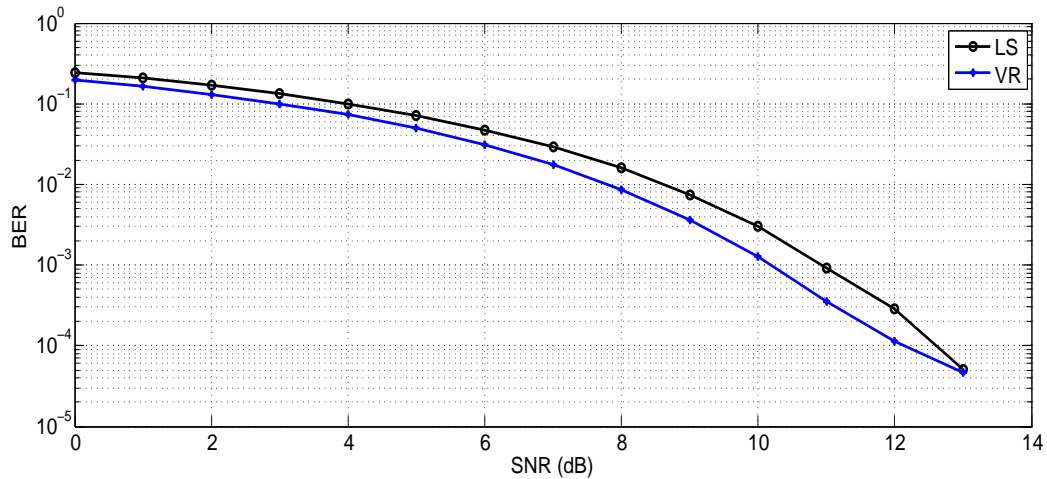


Figura 4.17: Resultados del DSP de la estimación LS vs. VR.

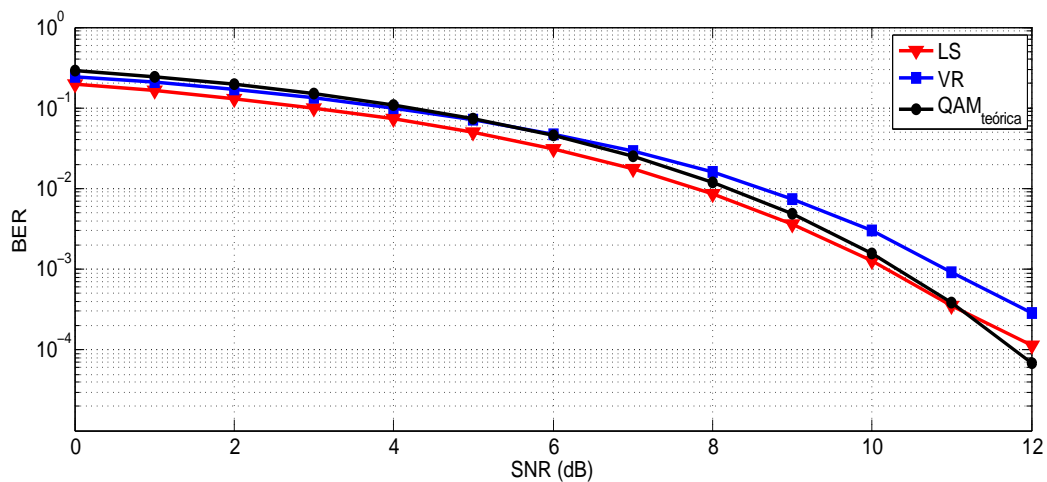


Figura 4.18: Comparación de resultados ante la curva teórica QAM.

### 4.3.2. Pruebas y resultados del canal selectivo en frecuencia

En este apartado se presentan los resultados obtenidos en el DSP, con la simulación de un canal selectivo en frecuencia más ruido blanco Gaussiano (que en lo futuro, dica combinación de canales, se referirá como canal PLC), así como la evaluación de la técnica DMT. Los coeficientes complejos del canal selectivo en frecuencia empleado, se pueden encontrar en la [tabla B.1](#). En los siguientes párrafos se presenta la comparación de resultados experimentales del canal PLC y el AWGN en el DSP.

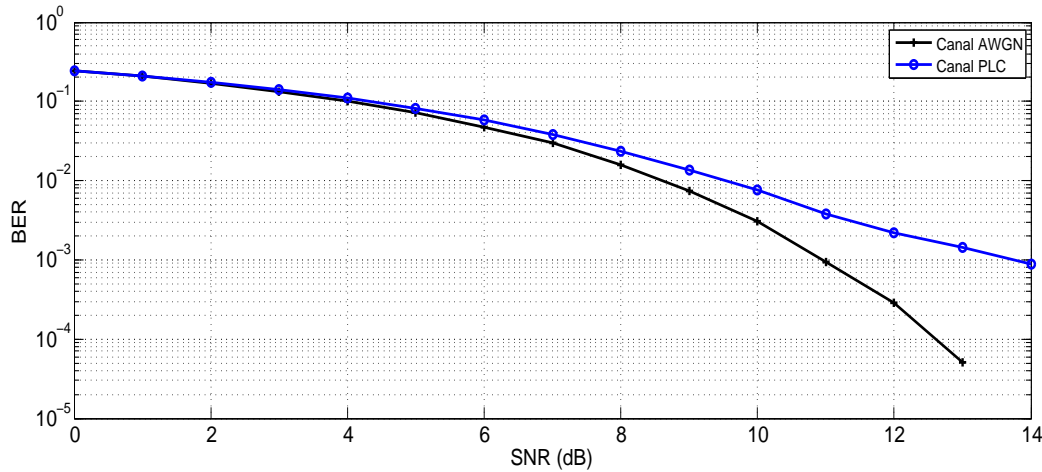


Figura 4.19: Gráficas BER del desempeño del DSP en un canal PLC vs. AWGN usando estimación LS.

### Evaluación de un canal PLC usando estimación LS

En la figura 4.19 se gráfica el desempeño del DSP en un canal PLC usando la estimación LS. En la figura 4.19 también se gráfica la curva de la estimación LS en un canal AWGN, donde se observa que después de los 3 dB de valor SNR comienzan a separarse ambas gráficas. Como se sabe, una señal afectada por atenuación presenta una probabilidad de error superior que en un canal Gaussiano. Si se analiza el desempeño en 12 dB de valor SNR, se aprecia que un canal AWGN tiene 3 bits en error de cada 10 mil, por otra parte, en el canal PLC resultan 2 bits en error por cada mil. La gráfica del canal PLC presenta una tendencia asintótica al nivel irreductible del error, debido a la distorsión del canal.

### Evaluación de un canal PLC usando estimación VR

En la figura 4.20 se encuentran los resultados obtenidos con el DSP simulando un canal PLC y uno AWGN usando la estimación VR. Ambas gráficas tienen una separación visible, la cual conforme aumenta el valor SNR, ésta se incrementa. Si se analiza el desempeño en los 12 dB de valor SNR, se aprecia que un canal AWGN genera 12 bits en error de cada 100 mil, por otra parte, en el canal PLC resultan 19 bits en error por cada mil. Al igual que la gráfica de estimación LS, la gráfica de la estimación VR en un canal PLC presenta una tendencia asintótica al nivel irreductible del error, conforme aumenta el valor SNR.

### Comparación de los resultados del canal PLC con estimación LS vs. VR

En la figura 4.21 se presentan los resultantes de evaluar los algoritmos implementados en el DSP usando estimación LS y VR ante un canal PLC. A diferencia de la comparación de resultados de ambas estimaciones en un canal AWGN, la estimación LS presenta un mayor

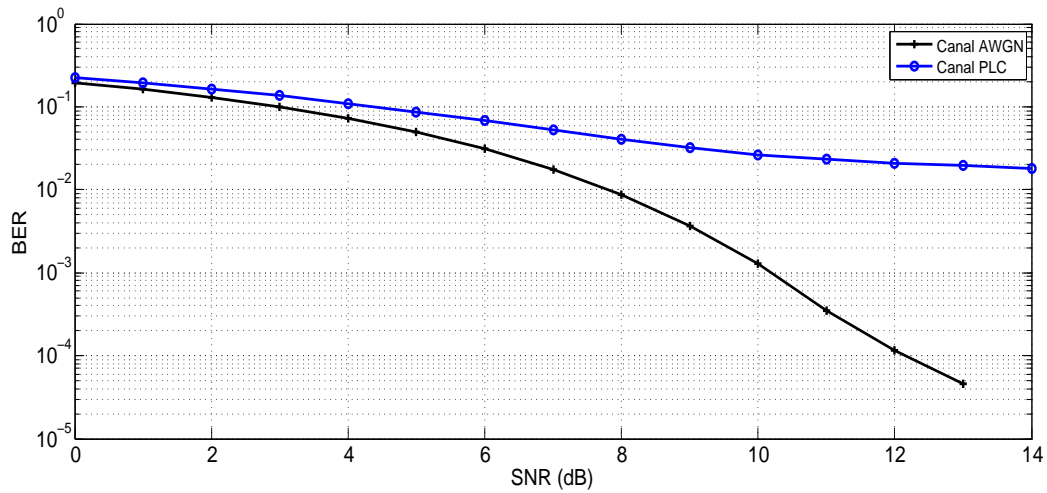


Figura 4.20: Gráficas BER del desempeño del DSP en un canal PLC vs. AWGN usando estimación VR.

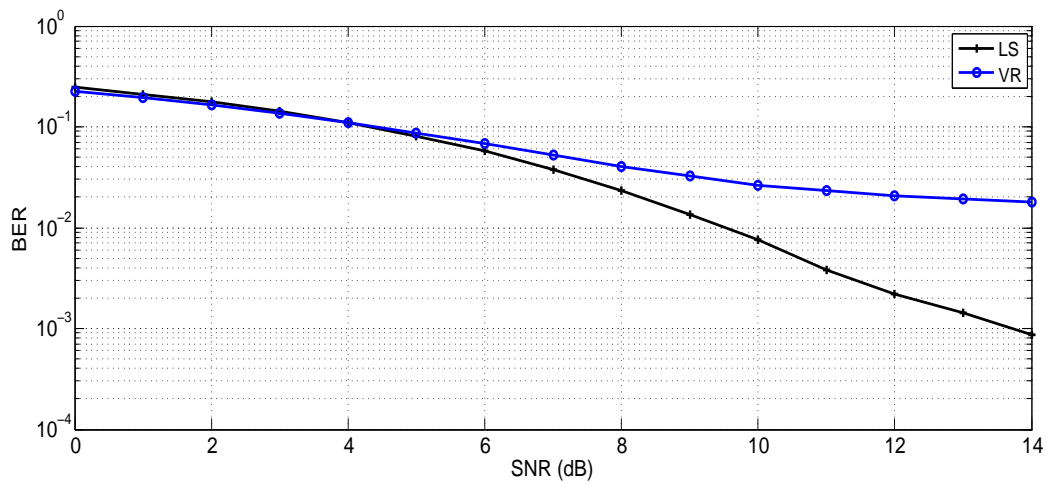


Figura 4.21: Gráficas BER de la evaluación de un canal PLC con estimación LS vs. VR.

desempeño ante un canal PLC. Por ejemplo, al analizar la tasa de bits en error a los 12 dB de valor SNR, se tienen 2 bits en error de cada mil con estimación LS y 19 bits en error usando estimación VR.

### 4.3.3. Pruebas y resultados de la técnica DMT en un canal PLC

En los siguientes párrafos se presentan los resultados de la evaluación de la técnica DMT. Donde, la técnica DMT permite discriminar las sub-bandas afectadas significativamente por

la atenuación. Por lo tanto, se realizó la evaluación empleando las técnicas de estimación de canal seleccionadas (LS y VR) y los resultados de esta prueba se presentan en los siguientes párrafos.

### Evaluación de la técnica DMT usando la estimación LS

En la [figura 4.22](#) se grafican los resultados de la estimación LS usando la técnica DMT, así como la gráfica de evaluar el sistema de prueba sin la técnica DMT. Los resultados muestran que el sistema funciona adecuadamente, mostrando un mejor desempeño la gráfica donde se usó la técnica DMT. Por ejemplo, al evaluar los resultados bajo un valor de 12 dB de SNR, se tiene que al usar DMT se presentan dos bits en error de cada mil y sin DMT se tienen 8 bits en error de cada 100. Estos resultados se deben al uso de la técnica DMT con OFDM, la cual permite discriminar del canal, las sub-bandas que presentan altos niveles de atenuación. Por lo tanto, esta técnica contribuye a disminuir el número de errores en la transmisión y se puede observar en la tendencia asintótica al nivel irreductible del error de la [figura 4.22](#).

### Evaluación de la técnica DMT usando la estimación VR

En la [figura 4.23](#) se grafican los resultados de la estimación VR usando la técnica DMT, así como la gráfica de los resultados de evaluar al sistema de prueba sin la técnica DMT. Los resultados muestran que el sistema funciona adecuadamente, mostrando un mejor desempeño la gráfica donde se usó la técnica DMT. Por ejemplo, al evaluar los resultados bajo un valor de 12 dB de SNR, se tiene que al usar DMT se presentan 19 bits en error de cada mil y sin DMT se tienen 9 bits en error de cada 100. Los resultados de esta evaluación se deben a la misma situación que el caso de la estimación LS. Es decir, la gráfica de la evaluación DMT

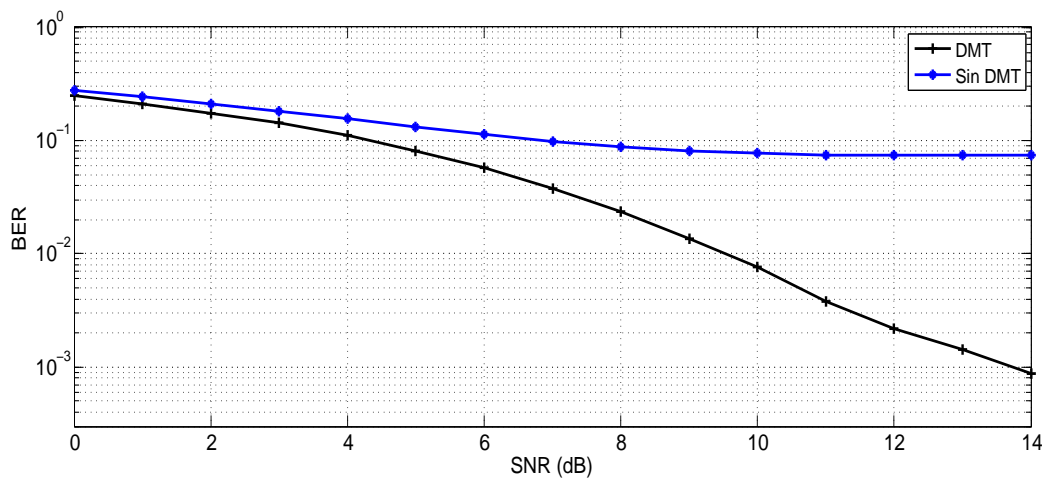


Figura 4.22: Evaluación de la técnica DMT usando la estimación LS.

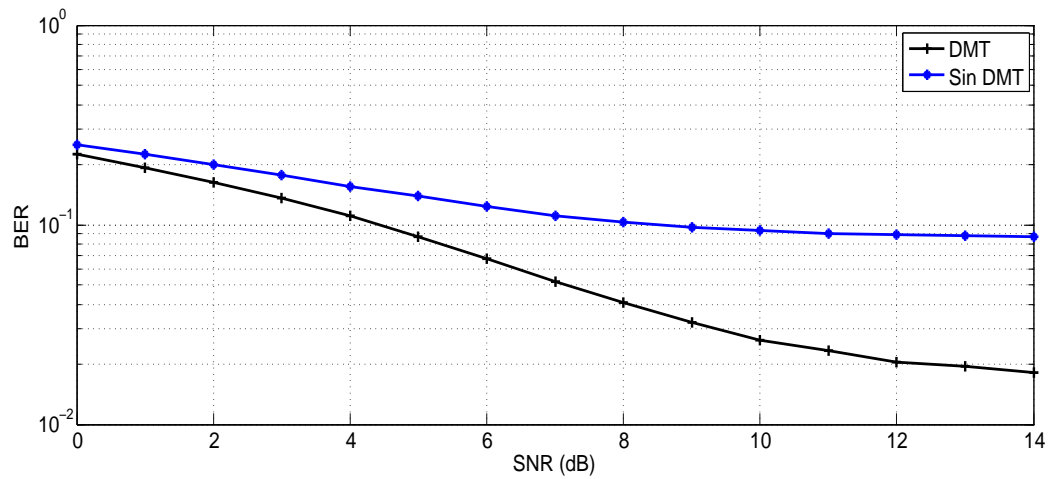


Figura 4.23: Evaluación de la técnica DMT usando la estimación VR.

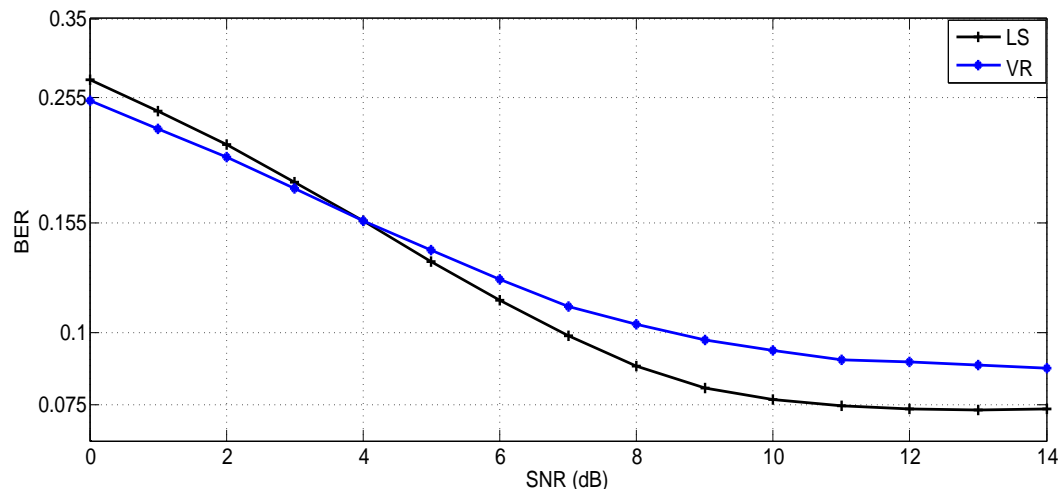


Figura 4.24: Estimación de canal LS y VR sin técnica DMT.

presenta un mejor desempeño que cuando no se usa DMT, debido a que evita transmitir información sobre sub-bandas afectadas con niveles altos de atenuación.

#### Comparación de las técnicas de estimación de canal LS y VR sin técnica DMT

En la [figura 4.24](#) se observan las gráficas resultantes de evaluar el sistema de prueba con estimación de canal LS y VR sin uso de la técnica DMT. Nuevamente, la estimación LS presenta un mejor desempeño comparado con VR. La diferencia en errores es muy baja, por ejemplo, teniendo un valor de SNR de 12 dB, se tiene 8 y 9 bits en error para LS y

ZF, respectivamente. La técnica LS muestra un mejor desempeño por arriba de los 4 dB de valor SNR, esto se debe a que VR es sólo una técnica de estimación por eliminación de ruido con conocimiento a priori de la dispersión de la respuesta al impulso del canal, lo cual no le permite compensar los efectos de distorsión introducidos por un canal selectivo en frecuencia.

Es importante destacar que por debajo de los 4 dB de SNR, la técnica VR presenta un mejor desempeño. Lo cual tiene justificación en lo antes mencionado, es decir, VR elimina el ruido Gaussiano introducido a la señal usando el conocimiento de la dispersión del canal. Por otra parte, la estimación LS es altamente susceptible a niveles altos de ruido.

---

# Conclusiones y trabajo futuro

---

## 5.1. Conclusiones

En el presente trabajo de investigación se presentó la implementación en un DSP, de los algoritmos del nivel físico para la transmisión con un esquema OFDM para un módem PLC. Este trabajo ha permitido analizar y evaluar el comportamiento de los modelos formales propuestos por M. Crussière [1] en un ambiente de recursos limitados que simula los efectos de un canal Gaussiano y selectivo en frecuencia sobre una señal OFDM.

La programación de los algoritmos se hizo usando representación de punto fijo, con el fin de aprovechar el máximo desempeño del procesador digital de señales, de acuerdo a las especificaciones del fabricante. Podemos concluir respecto al kit de desarrollo (DSK), que son una potente herramienta para el diseño, implementación y evaluación de aplicaciones que demandan un costo computacional elevado con el mínimo de recursos.

Las pruebas realizadas de manera individual a los algoritmos, permitieron evaluar el funcionamiento modular del sistema y así, realizar un análisis comparativo respecto de los resultados obtenidos en la simulación por computadora. Los resultados obtenidos en el DSP, demostraron el funcionamiento esperado de cada uno de los módulos evaluados. Algunas propuestas puestas en práctica en el DSP que podrían ser consideradas en la construcción de un prototipo de módem OFDM/PLC son:

- a) El diseño de la trama de control o preámbulo que permite disminuir el tiempo requerido en el proceso de sincronización. También se aumenta la eficiencia en la detección del inicio de cada símbolo. La trama de control propuesta en el presente trabajo, se diseñó con el objetivo de disminuir el tiempo de sincronización, conservar la eficacia en la sincronización del sistema de referencia [26] y obtener mayor información para la estimación de canal, logrando satisfactoriamente dicho objetivo.
- b) Se implementó una modificación al modelo propuesto por M. Crussière [1] para el proceso de sincronización burda, dicha modificación se realizó sobre la elección del valor máximo de correlación (MC), donde, en el presente trabajo se propuso una elección con base en la moda estadística. La propuesta mostró resultados satisfactorios, logrando

disminuir el número de símbolos OFDM empleados en la sincronización burda de 10 a sólo 6 símbolos. Estos resultados contribuyen a disminuir el tiempo requerido en el proceso de sincronización.

- c) Los resultados revisados en la tesis de M. Crussiere, sólo muestran la sincronización en niveles por encima de 10 dB de SNR. Los resultados obtenidos del DSP con las modificaciones antes mencionadas (punto a y b), permiten sincronizar la comunicación a partir de los 8 dB.

La evaluación general del sistema usando simulación de Monte Carlo, permitió evaluar el desempeño del sistema de prueba, tomando como medida de desempeño la gráfica de la curva BER vs. SNR. Con base en el análisis realizado de los tres experimentos llevados a cabo (canal AWGN, canal PLC y evaluación DMT), podemos concluir lo siguiente:

- a) La evaluación del sistema de prueba usando las técnicas de estimación de canal por mínimos cuadrados (LS) y eliminación de ruido por ventaneo rectangular (VR), en un canal AWGN, dio como resultado un funcionamiento adecuado con respecto a los resultados del sistema de referencia (MATLAB) [26] y a la simulación por computadora realizada en lenguaje C.

De acuerdo a la discusión de resultados presentados en el capítulo 4, la estimación VR presentó un mejor desempeño comparado con la estimación por LS en un canal Gaussiano. Por lo tanto, la estimación de canal con eliminación de ruido por ventaneo rectangular resulta ser una alternativa eficiente en una transmisión de datos sobre un canal afectado por altos niveles de ruido Gaussiano.

- b) La evaluación del sistema de prueba en un canal PLC mostró resultados satisfactorios. Se evaluaron las técnicas de estimación LS y VR, y los resultados analizados permiten concluir que ante un canal con niveles de SNR por debajo de los 4 dB resulta ser más conveniente el uso de la estimación VR. Por otro parte, con niveles por encima de los 4 dB, la estimación LS mostró un mejor desempeño. Por lo tanto, es importante conocer el estado del canal para poder determinar cual de estas técnicas resulta más apropiada de implementar.
  - c) Finalmente, la evaluación de la técnica DMT permitió conocer la importancia de la implementación de dicha técnica en un canal no plano. Los resultados analizados muestran que el uso de DMT disminuye el número de errores de una transmisión. Nuevamente, al evaluar las técnicas de estimación de canal en un canal selectivo en frecuencia con ruido Gaussiano, la estimación LS presenta un mayor desempeño que VR.
  - d) En conclusión sobre las técnicas LS y VR, podemos decir que la estimación LS es una técnica ampliamente usada por su simplicidad y eficiencia en medios de transmisión con bajos niveles de ruido. Por otro lado, de acuerdo al análisis de los resultados, LS no es funcional en una transmisión por un canal altamente afectado por ruido. La estimación VR es una alternativa para la transmisión de datos por un canal altamente afectado
-



por ruido, esta técnica sólo requiere información a priori de la dispersión del canal para su buen funcionamiento y, mostrando como desventaja su bajo desempeño en un canal selectivo en frecuencia.

Hoy en día vemos el gran uso de los DSP como núcleo en módems para medios inalámbricos o cableados. En este trabajo, el uso de uno de estos procesadores, el cual se encuentra integrado en un kit de desarrollo, permitió evaluar los modelos formales de M. Crussière [1], así como algunas propuestas de mejora sobre la implementación de los algoritmos de transmisión OFDM del sistema de referencia de G. Laguna [26].

La tecnología PLC resulta ser una alternativa muy atractiva para satisfacer la demanda para estar conectados siempre y en todo lugar. Esta tecnología, aceptada principalmente por contar con la infraestructura que, me atrevo a decir, es la más grande existente en el mundo, y que lo seguirá siendo hasta el día que los dispositivos electrónicos sean autosuficientes energéticamente y no dependan de un cableado para su alimentación eléctrica. El avance que está alcanzando la tecnología PLC y los beneficios que presenta como el uso de una infraestructura existente, altas tasas de transferencia de datos, etc., me permiten recomendar y hasta sugerir a empresas como Comisión Federal de Electricidad (CFE), tomen la iniciativa de la puesta en práctica de esta nueva tecnología de comunicación, la cual podría ser accesible por cualquier usuario de la red eléctrica sin importar su ubicación geográfica.

## 5.2. Trabajo futuro

El alcance de este proyecto se ve acotado por el tiempo y financiamiento, sin embargo, se propone como continuación de este trabajo de investigación, la puesta en práctica de un prototipo de módem OFDM para la transmisión por un canal PLC.

Por otra parte, es conveniente la implementación de técnicas que impulsen el desarrollo sustentable desde la capa física de todo módem, siendo hoy en día un tópico de gran interés el ahorro de energía. Queda pendiente investigar e implementar técnicas como modulación adaptativa, potencia adaptativa, técnicas de detección y corrección de errores, entre otras que contribuyan a la transmisión eficiente de información.

Se espera que este proyecto sirva a nuevas generaciones de investigadores, como herramienta de simulación y evaluación de nuevas propuestas de aplicaciones de comunicación digital implementadas en dispositivos de aplicación específica como el DSP.

# Referencias

---

- [1] M. Crussière, *Tesis: Étude et Optimisation de Communications à Haut-débit sur lignes d'énergie: exploitation de la combinaison OFDM/CDMA*. Institut National des Sciences Appliquées de Rennes, Rennes, Francia, 2005.
- [2] E. Biglieri, "Coding and Modulation for a Horrible Channel," *IEEE Communications Magazine*, vol. 41, pp. 92–98, Mayo 2003.
- [3] N. Pavlidou, "Power Line Communication: State of the art and future trends," *IEEE Communications Magazine*, pp. 34–40, Abril 2003.
- [4] H. Yan, S. Zhou, *et al.*, "DSP Implementation of SISO and MIMO OFDM Acoustic Modems," *Conference OCEANS 2010 IEEE Sydney, Australia*, pp. 89–92, 2010.
- [5] V. Duc *et al.*, "Implementation of an OFDM system based on the TMS320C6416," *International Conference on Advance Technologies for Communications. Hai Phong, Vietnam*, pp. 74–77, 2009.
- [6] A. Doukas, A. Kotsopoulos, and G. Kalivas, "DSP Implementation of SNR Estimation Algorithm for OFDM Systems," *Communications, Circuits and Systems Proceedings, International Conference. Bucharest, Romania*, pp. 583–587, 2006.
- [7] S. Baig and N. Gohar, "A Discrete Multitone Transceiver at the Heart of the PHY Layer of an In-Home Power Line Communication Local Area Network," *IEEE Communications Magazine*, pp. 48–53, Abril 2003.
- [8] J. L. Carmona, F. J. Cañete, *et al.*, "A DMT Modem Prototype for Broadband PLC," *Power Line Communications and Its Applications, IEEE International Symposium. Orlando, Florida, USA*, pp. 56–61, 2006.
- [9] J. Wang *et al.*, "A New OFDM blind synchronization algorithm suitable for PLC," *Image and Signal Processing. CISP '09. 2nd International. Nankai District, Tianjin, China*, pp. 1–5, Octubre 2009.
- [10] H. Wang *et al.*, "A New Synchronization Scheme for PLC," *Future Information Technology and Management Engineering. FITME '08. Leicestershire, Reino Unido*, pp. 604–607, Noviembre 2008.

- 
- [11] A. K. Aminuddin Bin and M. Y. Bin Alias, "Performance of OFDM in the Variation of the Path Quantity in Power Line Communications," *International Journal of Network and Mobile Technologies*, pp. 44–48, Enero 2011.
- [12] J. Y. Baudais and J. F. Helard, "Increase in Multicast OFDM Data Rate in PLC Network using Adaptive LP-OFDM," *International Journal of Network and Mobile Technologies*, pp. 384–389, Enero 2009.
- [13] A. Maiga, J. Y. Baudais, and J. F. Helard, "Very high bit rate power line communications for home networks," *Power Line Communications and Its Applications. ISPLC '09. Dresden, Alemania*, pp. 313–318, Abril 2009.
- [14] M. S. Yousuf and M. El-Shafei, "Power Line Communication: An Overview - Part I," *Innovations in Information Technology. IIT '07. Dubai, Emiratos Árabes Unidos*, pp. 218–222, Noviembre 2007.
- [15] M. S. Yousuf and M. El-Shafei, "Power Line Communication: An Overview - Part II," *Information and Communication Technologies: From Theory to Applications. ICTTA '08. Damasco, Siria*, pp. 1–6, Abril 2008.
- [16] Home Plug Power line Alliance, "White Papers HomePlug 1.0." <https://www.homeplug.org/tech/whitepapers>. [Consulta: Noviembre 2011].
- [17] A. Majumder and J. J. Caffery, "Power Line Communication: an overview," *IEEE Communications Magazine*, pp. 4–13, Noviembre 2004.
- [18] G. Laguna and R. Barrón, *Reporte técnico: Especificación de referencia para sistema de comunicación por la línea eléctrica (PLC)*. Instituto Politécnico Nacional, Centro de Investigación en Computación, ISBN 978-607-414-048-4, Enero 2009.
- [19] G. Qiang and Z. Jun, "LMS Algorithms for noise Offset in Medium-Voltage Power Line Communication," *Intelligent Systems and Applications, ISA '09. International Workshop*, pp. 1–4, Mayo 2009.
- [20] Open PLC European Research Alliance, "PLC Technology license." <http://www.ipcf.org> [Consulta: Febrero 2011].
- [21] IEEE Standards Association, "P1901.2 - Standard for Low Frequency (less than 500 khz) Narrow Band Power Line Communications for Smart Grid Applications." <http://standards.ieee.org/develop/project/1901.2.html> [Consulta: Junio 2012].
- [22] R. W. Chang and R. A. Gibby, "A theoretical study of performance of an orthogonal multiplexing data transmission scheme," *IEEE Transactions on Communication Technology*, pp. 529–540, Agosto 2003.
- [23] R. Prasad, *OFDM for wireless communication*. Boston London: Artech House Publishers, 2004.
-

- 
- [24] X. Liang *et al.*, “160 Gb/s OFDM Transmission Utilizing An All-optical Symbol Generator Based on PLC,” *Asia Communications and Photonics Conference and Exhibition (ACP)*. Shanghai, China, vol. 41, pp. 1–7, Noviembre 2009.
- [25] H. Harada and R. Prasad, *Simulation and software radio for mobile communications*. Boston London: Artech House Publishers, 2002.
- [26] G. Laguna and A. Prieto, *Reporte técnico: Simulador Monte Carlo para el esquema básico de comunicación OFDM sobre la red eléctrica*. México, DF.: Universidad Autónoma Metropolitana, ISBN 02.0201.II.15.003.2011, 2011.
- [27] Texas Instrument Incorporated, *TMS320C6414, TMS320C6415, TMS320C6416 Fixed-point Digital Signal Processors*. Literature Number: SPRS146N, Febrero 2001.
- [28] Texas Instrument Incorporated, *TMS320C64x DSP Library Programmer’s Reference*. Literature Number: SPRU565B, Octubre 2004.
- [29] Texas Instrument Incorporated, *TMS320C6000 Code Composer Studio Tutorial*. Literature Number: SPRU301C, Febrero 2000.
- [30] Texas Instrument Incorporated, *Code Composer Studio v3.1 Getting Started Guide*. Literature Number: SPRU509E, Septiembre 2004.
- [31] L. I. Fabian *et al.*, “Programación y evaluación en un DSP de un esquema de corrección de errores unidireccional (Forward Error Correction FEC).”.
- [32] T. Roberts and M. Slaney, “FFT using fixed point arithmetic in C.” [www.jjj.de/fft/fftpage.html](http://www.jjj.de/fft/fftpage.html) [Consulta: Febrero. 2012].
-

---

## Apéndice A

# Glosario

---

### A

**Amplitud.** Valor de una señal, normalmente se mide en voltios (tensión), amperios (corriente), o vatios (potencia).

**Analógico.** Entidad que varía de forma continua.

**Atenuación.** Pérdida en la energía de una señal debido a la resistencia del medio.

### B

**Banda ancha.** Se refiere a una tecnología en la que la señal comparte el ancho de banda de un medio.

**Banda Base.** Se refiere a una tecnología en la que la señal se transmite directamente a través de un canal sin modular una portadora.

**Bit.** Dígito binario; unidad más pequeña de información; 1 o 0.

### C

**Canal.** Un camino utilizado para las comunicaciones.

**Comunicación de datos.** Intercambio de información entre dos o más entidades.

**Constelación.** Representación gráfica de la fase y la amplitud de combinaciones de bits diferentes en modulación combinada digital y

analógica.

**Corriente continua (DC).** Señal con frecuencia cero y amplitud constante.

### D

**Decibelio (dB).** Medida de la energía relativa entre dos puntos de una señal.

**Demodulación.** Proceso de separación de la señal portadora de la señal de información.

**Demodulador.** Dispositivo que realiza la demodulación.

**Detección de errores.** Proceso que determina si algunos bits se han cambiado durante la transmisión.

**Desplazamiento de fase.** Cambio de fase de una señal.

**Distorsión.** Cualquier cambio en una señal como consecuencia del ruido, atenuación u otras influencias.

### E

**Emisor.** Sistema que origina un mensaje.

**Enlace.** Camino de comunicación físico que transfiere datos de un dispositivo a otro.

**Error.** Daño producido en la transmisión de datos.

**Espectro.** Rango de frecuencias de una señal.

**Estándar.** Base o modelo en el que todo el

mundo se ha puesto de acuerdo.

## F

**Fase.** Posición relativa en el tiempo de una señal.

**Frecuencia.** Numero de ciclos por segundo de una señal periódica.

## H

**Hercio (Hz).** Unidad utilizada para medir la frecuencia.

## I

**Instituto de ingenieros eléctricos y electrónicos (IEEE).** Grupo formado por ingenieros profesionales que tiene sociedades especializadas cuyos comités preparan estándares en las áreas de la especialidad de los miembros.

**Internet.** Conjunto de redes conectadas por dispositivos de interconexión como encaminadores o pasarelas.

**I/O.** Input/Output; Entrada/Salida.

## K

**Kbps.** Kilobits por segundo.

## L

**Línea de abonado digital (DSL).** Tecnología que utiliza las redes de telecomunicaciones existentes para conseguir la entrega a alta velocidad de datos, voz, video y datos multimedia.

## M

**Módem.** Dispositivo para modular y demodular. Convierte una señal digital en una

analógica (modulación) y viceversa (demodulación).

**Modulación.** Modificación de una o más características de una onda portadora por una señal de información.

**Modulación por amplitud en cuadratura (QAM).** Método de modulación digital o analógica en el que la fase y la amplitud de una señal portadora varía con la señal que modula.

**Multiplexado por división en frecuencia (FDM).** Combinación de señales analógicas en una única señal.

**Multiplexado por división de frecuencias ortogonales (OFDM).** Combinación de señales analógicas en una múltiples señales portadoras espaciadas en frecuencias ortogonales.

## N

**Nivel físico.** Primer nivel del modelo OSI, responsable de la especificaciones eléctricas y mecánicas del medio.

## P

**Protocolo.** Reglas para la comunicación.

**Puesta a punto.** Etapa preliminar necesaria para una conexión lógica antes de transferir datos.

## R

**Receptor.** Punto destino de una transmisión.

**Ruido.** Señal electrónica aleatoria que puede afectar a un medio de transmisión y dar lugar a la degradación o distorsión de los datos.

## S

---

**Señal.** Ondas electromagnéticas propagadas a través de un medio de transmisión.

**Símbolo OFDM.** Conjunto de detos modulados con portadoras ortogonales.

**Sincronización.** Proceso que permite la transmisión y recepción entre dos dispositivos simultaneamente.

## T

**Tasa de transmisión.** El número de bits enviados por segundo.

**Técnica multitono discreta (DMT).** Método de modulación que combina elementos QAM y FDM.

**Trama.** Grupo de bits que representan un grupo de datos.

**Trama de control.** Trama utilizada en el proceso de puesta a punto del sistema.

**Trama de datos.** Trama utilizada para representar los datos útiles en una transmisión.

**Transformada de Fourier.** Técnica matemática que reduce una señal periódica en una serie de señales seno más sencillas.

---

---

## Apéndice B

# Secuencia compleja

---

Tabla B.1: Secuencia compleja de un canal selectivo en frecuencia.

| Índice | Valor (re, im)      | Índice | Valor (re, im)      | Índice | Valor (re, im)      |
|--------|---------------------|--------|---------------------|--------|---------------------|
| 1      | ( 0.0003 , 0 )      | 52     | ( 0.0014 ,-0.0002 ) | 103    | (-0.0048 , 0.0015 ) |
| 2      | ( 0.0011 , 0 )      | 53     | (-0.0001 , 0 )      | 104    | (-0.0004 , 0.0001 ) |
| 3      | (-0.0003 , 0 )      | 54     | ( 0.0027 ,-0.0005 ) | 105    | (-0.0007 , 0.0002 ) |
| 4      | ( 0.0009 , 0 )      | 55     | (-0.0054 , 0.0009 ) | 106    | (-0.0006 , 0.0002 ) |
| 5      | ( 0.0001 , 0 )      | 56     | ( 0.0333 ,-0.0057 ) | 107    | (0.0009 ,-0.0003 )  |
| 6      | ( 0.0010 , 0 )      | 57     | ( 0.0139 ,-0.0024 ) | 108    | (-0.0007 , 0.0002 ) |
| 7      | (-0.0003 , 0 )      | 58     | (-0.0067 , 0.0012 ) | 109    | ( 0.0022 ,-0.0008 ) |
| 8      | ( 0.0013 , 0 )      | 59     | ( 0.0045 ,-0.0008 ) | 110    | (-0.0040 , 0.0014 ) |
| 9      | (-0.0002 , 0 )      | 60     | (-0.0048 , 0.0009 ) | 111    | ( 0.0119 ,-0.0042 ) |
| 10     | ( 0.0005 , 0 )      | 61     | ( 0.0036 ,-0.0007 ) | 112    | ( 0.0135 ,-0.0048 ) |
| 11     | (-0.0002 , 0 )      | 62     | (-0.0042 , 0.0008 ) | 113    | (-0.0047 , 0.0017 ) |
| 12     | ( 0.0011 , 0 )      | 63     | ( 0.0034 ,-0.0007 ) | 114    | ( 0.0025 ,-0.0009 ) |
| 13     | (-0.0004 , 0 )      | 64     | (-0.0040 , 0.0008 ) | 115    | (-0.0030 , 0.0011 ) |
| 14     | ( 0.0015 ,-0.0001 ) | 65     | ( 0.0033 ,-0.0007 ) | 116    | ( 0.0013 ,-0.0005 ) |
| 15     | (-0.0002 , 0 )      | 66     | (-0.0059 , 0.0012 ) | 117    | (-0.0019 , 0.0007 ) |
| 16     | ( 0.0017 ,-0.0001 ) | 67     | ( 0.0062 ,-0.0013 ) | 118    | ( 0.0008 ,-0.0003 ) |
| 17     | (-0.0005 , 0 )      | 68     | (-0.0304 , 0.0063 ) | 119    | (-0.0008 , 0.0003 ) |
| 18     | ( 0.0014 ,-0.0001 ) | 69     | (-0.0073 , 0.0015 ) | 120    | ( 0.0008 ,-0.0003 ) |
| 19     | (-0.0009 , 0 )      | 70     | ( 0.0052 ,-0.0011 ) | 121    | (-0.0018 , 0.0007 ) |
| 20     | ( 0.0018 ,-0.0001 ) | 71     | (-0.0005 , 0.0001 ) | 122    | (-0.0006 , 0.0003 ) |
| 21     | (-0.0004 , 0 )      | 72     | ( 0.0019 ,-0.0004 ) | 123    | ( 0.0036 ,-0.0014 ) |
| 22     | ( 0.0017 ,-0.0001 ) | 73     | (-0.0000 , 0 )      | 124    | ( 0.0064 ,-0.0025 ) |
| 23     | (-0.0012 , 0.0001 ) | 74     | ( 0.0006 ,-0.0001 ) | 125    | (-0.0038 , 0.0015 ) |
| 24     | ( 0.0022 ,-0.0002 ) | 75     | ( 0.0000 , 0 )      | 126    | ( 0.0018 ,-0.0007 ) |
| 25     | (-0.0011 , 0.0001 ) | 76     | ( 0.0015 ,-0.0004 ) | 127    | (-0.0028 , 0.0012 ) |



---

| Índice | Valor (re, im)       | Índice | Valor (re, im)       | Índice | Valor (re, im)       |
|--------|----------------------|--------|----------------------|--------|----------------------|
| 26     | ( 0.0027 , -0.0002 ) | 77     | ( -0.0003 , 0.0001 ) | 128    | ( 0.0015 , -0.0006 ) |
| 27     | ( -0.0019 , 0.0002 ) | 78     | ( -0.0122 , 0.0029 ) | 129    | ( -0.0032 , 0.0013 ) |
| 28     | ( 0.0034 , -0.0003 ) | 79     | ( 0.0022 , -0.0005 ) | 130    | ( 0.0013 , -0.0005 ) |
| 29     | ( -0.0023 , 0.0002 ) | 80     | ( -0.0005 , 0.0001 ) | 131    | ( -0.0032 , 0.0014 ) |
| 30     | ( 0.0044 , -0.0004 ) | 81     | ( 0.0024 , -0.0006 ) | 132    | ( 0.0040 , -0.0017 ) |
| 31     | ( -0.0038 , 0.0004 ) | 82     | ( -0.0002 , 0 )      | 133    | ( -0.0184 , 0.0079 ) |
| 32     | ( 0.0076 , -0.0007 ) | 83     | ( 0.0021 , -0.0005 ) | 134    | ( 0.0006 , -0.0002 ) |
| 33     | ( -0.0087 , 0.0009 ) | 84     | ( 0.0001 , 0 )       | 135    | ( -0.0135 , 0.0059 ) |
| 34     | ( 0.0289 , -0.0029 ) | 85     | ( 0.0021 , -0.0005 ) | 136    | ( -0.0136 , 0.0060 ) |
| 35     | ( 0.0466 , -0.0049 ) | 86     | ( 0.0001 , 0 )       | 137    | ( 0.0096 , -0.0043 ) |
| 36     | ( -0.0139 , 0.0015 ) | 87     | ( 0.0030 , -0.0008 ) | 138    | ( -0.0014 , 0.0006 ) |
| 37     | ( 0.0065 , -0.0007 ) | 88     | ( -0.0007 , 0.0002 ) | 139    | ( 0.0049 , -0.0022 ) |
| 38     | ( -0.0054 , 0.0006 ) | 89     | ( 0.0047 , -0.0013 ) | 140    | ( -0.0006 , 0.0003 ) |
| 39     | ( 0.0037 , -0.0004 ) | 90     | ( 0.0203 , -0.0057 ) | 141    | ( 0.0037 , -0.0017 ) |
| 40     | ( -0.0030 , 0.0004 ) | 91     | ( -0.0011 , 0.0003 ) | 142    | ( 0.0002 , -0.0001 ) |
| 41     | ( 0.0028 , -0.0003 ) | 92     | ( 0.0002 , -0.0001 ) | 143    | ( 0.0039 , -0.0018 ) |
| 42     | ( -0.0022 , 0.0003 ) | 93     | ( 0.0001 , 0 )       | 144    | ( -0.0018 , 0.0008 ) |
| 43     | ( 0.0018 , -0.0002 ) | 94     | ( -0.0002 , 0 )      | 145    | ( 0.0198 , -0.0094 ) |
| 44     | ( -0.0011 , 0.0002 ) | 95     | ( 0.0014 , -0.0004 ) | 146    | ( 0.0025 , -0.0012 ) |
| 45     | ( 0.0016 , -0.0002 ) | 96     | ( 0.0000 , 0 )       | 147    | ( -0.0007 , 0.0004 ) |
| 46     | ( -0.0006 , 0.0001 ) | 97     | ( 0.0021 , -0.0006 ) | 148    | ( 0.0001 , 0 )       |
| 47     | ( 0.0011 , -0.0002 ) | 98     | ( -0.0013 , 0.0004 ) | 149    | ( -0.0005 , 0.0002 ) |
| 48     | ( -0.0003 , 0 )      | 99     | ( 0.0049 , -0.0015 ) | 150    | ( 0.0006 , -0.0003 ) |
| 49     | ( 0.0009 , -0.0001 ) | 100    | ( 0.0030 , -0.0009 ) | 151    | ( 0.0004 , -0.0002 ) |
| 50     | ( 0.0001 , 0 )       | 101    | ( 0.0055 , -0.0017 ) | 152    | ( 0.0011 , -0.0006 ) |
| 51     | ( 0.0004 , -0.0001 ) | 102    | ( 0.0297 , -0.0095 ) | 153    | ( -0.0008 , 0.0004 ) |

---

---

## Apéndice C

# Algoritmos

---

### C.1. Transmisor/Receptor OFDM (trama de datos)

```
1  /*****
2  * Función que simula al transmisor/canal/receptor de una
3  * transmisión OFDM/DMT.
4  *****/
5  *****/
6  Programador: Daniel Torres Alvarado
7  Fecha: 20/09/2012
8  *****/
9
10 /*****
11 Parámetros de entrada
12 * SNR: Nivel de relación señal a ruido deseado
13 * noise: bandera de activación de ruido
14 *   - 0 Sin ruido
15 *   - 1 Con ruido
16 * CE: Tipo de estimación de canal:
17 *   - 0 Estimación LS
18 *   - 1 Estimación VR
19 * co_re: Ventana real requerida para la FFT
20 * co_im: Ventana imaginaria requerida para la FFT
21 *****/
22 *****/
23 Parámetros de salida
24 * regresa: el valor BER de la simulación
25 *****/
26 float uncoded_OFDM_CE_frame_PA_XI(int SNR,int noise ,int CE,
27     fixed *co_re ,fixed *co_im){
28     /*****
29     Parámetros iniciales
30     *****/
31     int i,j, errores , total_errores ,capacidad_simbolo ,
32         total_bits ,escala;
33     COMPLEX Y_analitica [QAMsimbolos] ,
34     xAofdm [Long_tram_XA_D]={0}, X_analitica [QAMsimbolos] ,
```

```

35     h_en [ QAMsimbolos ];
36     short Yr [ Nbits_simbolo ] = { 0 }, Y [ Nbits_simbolo ],
37           XA [ Nbits_simbolo ], bit_DMT [ Nbits_simbolo ], X_ft [ Nptos_fft ];
38
39     /*****
40     *****/
41     Transmisión/Recepción del símbolo piloto tipo A
42     *****/
43     *****/
44
45
46     /*****
47     Creación de la secuencia binaria del símbolo piloto A,
48     que se almacena en XA
49     *****/
50     bit_simbolA2 (XA);
51
52     /*****
53     Creación de: símbolo OFDM tipo A, que se almacena en
54     X_ft, señal analítica regresada en X_analitica
55     *****/
56     escala=fofdmqamtx03 (XA, X_ft, X_analitica, co_re, co_im);
57
58     /*****
59     Se agrega intervalo de guardia de longitud D (fading)
60     *****/
61     for (i=0; i<D; i++){
62         xAofdm [ i ]. real = X_ft [ Nptos_fft -D+i ];
63         xAofdm [ i ]. imag = 0;
64     }
65     for (i=0; i<Nptos_fft; i++){
66         xAofdm [ D+i ]. real = X_ft [ i ];
67         xAofdm [ D+i ]. imag = 0;
68     }
69
70     /*****
71     FIN DE TRANSMISOR (piloto)
72     *****/
73     Se simula el inicio de transmisión, la trama de datos
74     con se deposita en xAofdm
75     *****/
76     A continuación, se simula el paso por el canal de la
77     trama de datos
78     *****/
79     *****/
80     /*****
81     Simulación de canal AWGN
82     *****/
83     if (noise)
84         gen_noise (xAofdm, Long_tram_XA_D, SNR);
85
86     /*****

```

```

86      *****
87      INICIO DE RECEPTOR de simbolo piloto
88      *****
89      *****/
90
91      /*****
92      Recepción del simbolo piloto A
93      *****/
94      // inicio de símbolo (En principio el símbolo A):
95      Recep_Simbolo_OFDM(xAofdm, Y_analitica , co_re , co_im , escala);
96      /*****
97      Estimacion de canal
98      Observación del canal PLC con símbolo A
99      mediante expresión  $H=Y/X$ 
100     *****/
101     for (i=0;i<QAMsimbolos;i++)
102         h_en[i] = divi_com(Y_analitica[i], X_analitica[i]);
103     /*****
104     Estimación del canal:
105     *****/
106     if(CE)
107         ChannelEstimation_bs(h_en);
108     /*****
109     Mapeo y deteccion de bits
110     *****/
111     Recep_Map_Detec(Y_analitica , h_en);
112     /*****
113     Detección de datos de símbolo A
114     *****/
115     detect_bits(Y_analitica ,Y);
116     /*****
117     Detección de errores de símbolo A
118     *****/
119     errores = 0;
120     for (i=0;i<Nbits_simbolo;i++)
121         if(XA[i] != Y[i]){
122             errores+=1;
123         }
124
125     /*****
126     *****
127     Transmisión/Recepción de la trama de datos
128     *****
129     *****/
130
131     //Actualización de coeficientes de estimación de canal
132     for (i=0;i<QAMsimbolos;i++)
133         h_en[i]= suma_com(multi_com_const(h_en[i], epsilon) ,
134             multi_com_const(hen[i],(1-epsilon)));
135
136     //capacidad de bits transmisibles por símbolo OFDM/DMT

```

```

136     capacidad_simbolo = mapped_bit_count(mask) * (QAMsize>>1);
137     total_errores = 0;
138     total_bits = 0;
139
140     //*****
141     //Construye tramas OFDM
142     //*****
143     //ciclo para transmisión/recepcion de la trama de datos
144     // 100 simbolos OFDM por trama
145     for(i=0;i<100;i++){
146     //Construccion de simbolos "R1" para datos (pseudo-aleatorios)
147         for(j=0;j<capacidad_simbolo;j++){
148             if(UN(0,1)>0.5)
149                 XA[j]=1;
150             else
151                 XA[j]=0;
152             // Mapeo binario DMT
153             DMT_map_bits2(XA,mask,bit_DMT);
154             // Pasamos los bits de formato on-off a antipodal
155             onoff_2_antipodal(bit_DMT);
156             // creación del simbolo OFDM con FFT
157             escala=fofdmqamtx03(bit_DMT,X_ft,X_analitica,co_re,co_im
158                 );
159             // Inserción prefijo ciclico
160             for(j=0;j<D;j++){
161                 xAofdm[j].real = X_ft[Nptos_fft-D+j];
162                 xAofdm[j].imag = 0;
163             }
164             for(j=0;j<Nptos_fft;j++){
165                 xAofdm[D+j].real = X_ft[j];
166                 xAofdm[D+j].imag = 0;
167             }
168             //*****
169             //*****
170             FIN DE TRANSMISOR (piloto)
171             Se simula el inicio de transmisión, la trama de datos
172             con se deposita en xAofdm
173             *****
174             A continuacion, se simula el paso por el canal de la
175             trama de datos
176             *****
177             //*****
178             //*****
179             Simulación de canal AWGN
180             *****
181             if(noise)
182                 gen_noise(xAofdm, Long_tram_XA_D, SNR);
183
184             //*****
185             //*****

```

```

186      INICIO DE RECEPTOR de simbolos de datos
187      *****/
188      *****/
189
190      // inicio del primer símbolo de datos
191      Recep_Simbolo_OFDM(xAofdm,Y_analitica , co_re , co_im ,escala);
192      //Estimación del canal:
193      if(CE)
194          ChannelEstimation_bs(h_en);
195      //Mapeo y deteccion de bits
196      Recep_Map_Detec(Y_analitica , h_en);
197      //Detección de datos de símbolo A
198      detect_bits(Y_analitica ,Y);
199
200      /******
201      Inicia Detección de errores de símbolo
202      *****/
203
204      //pasamos los datos a valores on-off
205      antipodal_2_onoff(Y);
206      //para recuperar la secuencia codificada mapeada en las
207      //portadoras habilitadas con DMT
208      DMT_unmap_bits(Y,Yr,mask);
209
210      // contador de errores
211      errores = 0;
212      for(j=0;j<capacidad_simbolo;j++)
213          if(XA[j] != Yr[j])
214              errores+=1;
215      // se obtiene la tasa de error por símbolo
216      total_errores = total_errores + errores;
217      total_bits = total_bits + capacidad_simbolo;
218  }//fin de ciclo de simulacion de transmisión
219  // de simbolos OFDM de datos.
220
221  // Regresa el BER total de la transmisión
222  return ((float)total_errores/(float)total_bits);
223  }

```

## C.2. Generador de símbolos OFDM

```

1
2  /******
3  * Script para generar en el DSP símbolos piloto para el
4  * OFDM usando la FFT. Los símbolos se generan a partir de una
5  * secuencia de símbolos complejo quivalentes a
6  * símbolos QAM.
7  *****/

```

```

8  *****
9  * Programador: Daniel Torres Alvarado
10 * Fecha:05/05/11
11 *****/
12
13 /*****
14 * Parámetros de entrada:
15 * x_bin: Secuencia anipodal de longitud 2*M
16 * w_re: Ventana real requerida en la IFFT
17 * w_im: Ventana imaginaria requerida en la IFFT
18 *****
19 *****
20 * Parámetros de salida:
21 * X_ft: Secuencia real resultante de la FFT con simetría Hermitiana
22 * X_analitica: Secuencia compleja que contiene la señal analítica
23 * escala: Valor de escala sobre cada valor resultante de la FFT
24 *****/
25
26 int fofdmqamtx03(short *x_bin,short *X_ft,COMPLEX *X_analitica ,
27                 short *w_re , short *w_im){
28
29     /*****
30     Parámetros iniciales
31     *****/
32     int i ,escala;
33     // Vector auxiliar para la simetría Hermitiana
34     short imag[Nptos_fft];
35
36     /*****
37     Modulación 4QAM (Señal analítica):
38     Se genera la secuencia compleja empleada como señal analítica
39     *****/
40     for (i=0;i<QAMsimbolos; i++)
41     {
42         X_analitica[i].real = x_bin[i];
43         X_analitica[i].imag = x_bin[i+QAMsimbolos];
44     }
45
46     /*****
47     Simetría Hermitiana:
48     Se crea el vector con simetría Hermitiana a partir de la
49     señal analítica.
50     *****/
51     X_ft[0]= X_ft[QAMsimbolos]= 0;
52     imag[0]= imag[QAMsimbolos]= 0;
53     for (i=1;i<QAMsimbolos; i++)
54     {
55         // Parte real
56         X_ft[i] = X_analitica[i].real;
57         // Parte real espejo
58         X_ft[QAMsimbolos*2-i] = X_analitica[i].real;

```

```

59         // Parte imaginaria
60         imag[i] = X_analitica[i].imag;
61         // Parte imaginaria conj espejo
62         imag[QAMsimbolos*2-i] = -X_analitica[i].imag;
63     }
64     /*****
65     Símbolo OFDM:
66     Se procesa el vector con simetría Hermitiana usando la FFT,
67     como resultado se obtiene una señal con parte imaginaria
68     casi cero, por lo tanto se elimina la parte imaginaria y
69     se regresa sólo la parte real de longitud N
70     *****/
71     //Transmormada inverza de fpurier a 2*M puntos
72     escala = fix_fft(X_ft,imag, LOG2_N_WAVE-1,w_re,w_im, 1,
73         LOG2_N_WAVE);
74     /*****
75     FFT punto fijo:
76     El algoritmo empleado para la FFT realiza las operaciones con
77     aritmética en punto fijo, la variable "escala" almacena la
78     escala a la cual se encuentran las muestras del vector real
79     *****/
80     return escala;
}

```

### C.3. Proceso OFDM inverso

```

1  /*****
2  * Función que permite recuperar la señal real transmitida,
3  * para ello calcula la FFT, teniendo como salida un vector
4  * con simetría Hermitiana. Posteriormente, toma la mitad de
5  * muestras de dicho vector y las regresa como la
6  * señal entrante.
7  *****/
8  *****/
9  Programador: Daniel Torres Alvarado
10 Fecha: 20/09/2012
11 *****/
12
13 /*****
14 Parámetros de entrada
15 * s_ofdm: Ventana OFDM entrante sincronizada en forma fina
16 * w_re: Ventana real requerida para la FFT
17 * w_im: Ventana imaginaria requerida para la FFT
18 *****/
19 *****/
20 Parámetros de salida
21 * Y: Se regresa el vector complejo obtenido en la FFT.
22 *****/

```



```

23
24 void Recep_Simbolo_OFDM(COMPLEX *s_ofdm,COMPLEX *Y,
25     short *w_re, short *w_im, int escala)
26 {
27     /*****
28     Parámetros iniciales
29     *****/
30     int i;
31     short real[Nptos_fft], imag[Nptos_fft];
32
33     /*****
34     Inicio de símbolo, se retira intervalo de guardia y
35     se separa en un vector real y otro imaginario
36     *****/
37     for(i=0;i<Nptos_fft;i++){
38         real[i] = s_ofdm[i+D].real;
39         imag[i] = s_ofdm[i+D].imag;
40     }
41     /*****
42     Aplicamos FFT y se guarda en Y.
43     *****/
44     fix_fft(real,imag, LOG2N_WAVE-1,w_re, w_im,
45         0, LOG2N_WAVE);
46     /*****
47     Se recupera señal analítica a partir de la señal con
48     simetría hermitiana resultante de la FFT
49     *****/
50     for(i=0;i<QAMsimbolos;i++){
51         //Se recupera la escala de cada valor
52         Y[i].real = real[i]<< escala;
53         Y[i].imag = imag[i]<< escala;
54     }
55 }

```

## C.4. Modulación DMT

```

1 /*****
2 * Función que permite colocar la secuencia binaria a
3 * transmitir sobre las bandas viables para modulación
4 * DMT
5 *****/
6 *****/
7 Programador: Daniel Torres Alvarado
8 Fecha: 20/09/2012
9 *****/
10
11 /*****
12 Parámetros de entrada

```

```

13  * bits: Vector con la secuencia binaria a transmitir
14  * Mascara: Vector con la máscara de sub-bandas útiles
15  ****
16  ****
17  Parámetros de salida
18  * bits_DMT: la secuencia de datos por transmitir que corresponde
19  * a la máscara con utiles: el número de bits útiles transmitidos.
20  ****/
21
22  void DMT_map_bits2(short *bits,short *Mascara,
23                  short *bits_DMT)
24  {
25      /*****
26      Parámetros iniciales
27      *****/
28      int si,ix,i,j;
29      si=ix=0; //source index
30      /*****
31      Ordenamiento de la secuencia binaria sobre posiciones
32      utiles
33      *****/
34      for(j=0;j<2;j++)
35          //se recorre la secuencia binaria segun la longitud
36          // de la máscara
37          for(i=0;i<QAMsimbolos;i++){
38              if(Mascara[i]==1) //si la posición es viable
39                  {//bit asignado a la subportadora habilitada
40                      bits_DMT[ix] = bits[si];
41                      ix++;
42                      si++;
43                  }
44              else
45                  {//dehabilita el bit
46                      bits_DMT[ix] = 0;
47                      ix++;
48                  }
49          }
50  }

```

## C.5. Demodulación DMT

```

1  /*****
2  * Función que permite recuperar la secuencia binaria
3  * transmitida sobre las bandas viables para demodulación
4  * DMT
5  ****
6  ****
7  Programador: Daniel Torres Alvarado

```

```

8 | Fecha: 20/09/2012
9 | *****/
10 |
11 | /*****
12 | Parámetros de entrada
13 | * bits: Vector con la secuencia binaria a recibida
14 | * Mascara: Vector con la máscara de sub-bandas útiles
15 | *****
16 | *****
17 | Parámetros de salida
18 | * regresa: la secuencia de datos por transmitir que corresponde
19 | * a la máscara con utiles.
20 | *****/
21 |
22 | void DMT_unmap_bits(short *bits ,short *regresa ,
23 |                 short *Mascara)
24 | {
25 |     /*****
26 |     Parámetros iniciales
27 |     *****/
28 |     int di ,i ,j;
29 |     di=0; //source index
30 |     j = 0;
31 |     /*****
32 |     Recuperación de la secuencia binaria sobre posiciones
33 |     utiles
34 |     *****/
35 |     for (i=0;i<Nbits_simbolo ; i++)
36 |         if (j<QAMsimbolos) {
37 |             if (Mascara[j]==1) { //si la máscara es de
38 |                 //bit asignado a la subportadora
39 |                 //habilitada
40 |                 regresa[di] = bits[i];
41 |                 di++; //actualizamos índice
42 |             }
43 |             j++;
44 |         }
45 |         else {
46 |             j = 0;
47 |             i--;
48 |         }
49 |     }

```

## C.6. Estimación VR

```

1 | /*****
2 | * Script para estimación por ventaneo rectangular (VR)

```

```

3  * de la respuesta al impulso de la señal entrante
4  * aplicando la IFFT y FFT.
5  *****/
6  * Programador: Daniel Torres Alvarado
7  * Fecha:20/09/12
8  *****/
9
10 /******/
11 * Parámetros de entrada:
12 * H_en: Secuencia compleja de coeficientes de igualación H
13 * X_analitica: Secuencia compleja que contiene la señal analítica
14 *****/
15 *****/
16 * Parámetros de salida:
17 * X_analitica: La estimación de ruido se aplica directamente
18   a la secuencia de entrada.
19 *****/
20 void ChannelEstimation_bs(COMPLEX *H_en,COMPLEX *X_analitic){
21   /******/
22   Parámetros iniciales
23   *****/
24   int i, escala, L_halfspread;
25   short real[QAMsimbolos] = {0}, imag[QAMsimbolos]= {0};
26   /******/
27   Se extrae la parte real y la imaginaria del vector de
28   de coeficientes de estimación H_en
29   *****/
30   for (i=0;i<QAMsimbolos;i++){
31       real[i] = H_en[i].real;
32       imag[i] = H_en[i].imag;
33   }
34   /******/
35   Primero: pasar la señal analitica al dominio temporal
36   *****/
37   //Se realiza la transformada rápida de Fourier inversa
38   escala = fix_fft(real,imag, LOG2_N_WAVE-2,cofre ,
39                   cofim , 1, LOG2_N_WAVE-1);
40
41   // Se calcula la longitud de dispersión máxima
42   L_halfspread=(int)(D/2);
43   /******/
44   Segundo: Aislamiento del ruido en dominio temporal
45   de acuerdo a de longitud de dispersión máxima:
46   *****/
47   for (i=L_halfspread ; i<QAMsimbolos-L_halfspread -1;i++)
48   {
49       real[i] = 0;
50       imag[i] = 0;
51   }
52   /******/
53   Tercero: Se regresan los coeficientes la dominio de la frecuencia

```



```

33     Parámetros iniciales
34     *****/
35     int i , j , ix , escala , aux1 , aux2 , aux3 , aux4 , delta ;
36     short XA[Nbits_simbolo] , X_ft [ Nptos_fft ] , DeltaPhi [ QAMsimbolos ] ;
37     COMPLEX X_analitica [ QAMsimbolos ] , xAofdm [ Long_tram_XA_D ] = {0} , aux [
        Long_tram_XA_D ] , xBofdm [ Long_tram_XA_D ] , Y_analitica [ QAMsimbolos
        ] ;
38
39     /******
40     Se genera el inicio de la trama de forma aleatoria
41     *****/
42     ix = (ran*Long_tram_XA_D) ;
43
44     /******
45     Se generan los símbolos R1 (pseudo-aleatorios)
46     *****/
47     XA[0] = 0;
48     for (j=1;j<Nbits_simbolo;j++)
49         if (UN(0,1) > 0.5)
50             XA[j]=1;
51         else
52             XA[j]=0;
53     //Simbolo OFDM R1
54     fofdmqamtx03(XA, X_ft , X_analitica , co_re , co_im);
55     //Adición de intervalo de guarda
56     for (i=0;i<D;i++){
57         xBofdm[i].real = X_ft [ Nptos_fft -D+i ] ;
58         xBofdm[i].imag = 0;
59     }
60     for (i=0;i<Nptos_fft;i++){
61         xBofdm[D+i].real = X_ft [ i ] ;
62         xBofdm[D+i].imag = 0;
63     }
64
65     /******
66     Se generan los símbolos A
67     *****/
68     //Secuencia binaria A
69     bit_simbolA2(XA);
70     //Simbolo OFDM tipo A
71     escala=fofdmqamtx03(XA, X_ft , X_analitica , co_re , co_im);
72     //se agrega intervalo de guarda de longitud D (fadding)
73     for (i=0;i<D;i++){
74         xAofdm[i].real = X_ft [ Nptos_fft -D+i ] ;
75         xAofdm[i].imag = 0;
76     }
77     for (i=0;i<Nptos_fft;i++){
78         xAofdm[D+i].real = X_ft [ i ] ;
79         xAofdm[D+i].imag = 0;
80     }
81     /******

```

```

82      *****
83      Etapa de sincronización burda
84      *****
85      *****/
86
87      /*****
88      Se generan las trama de control para la sincronización
89      burda, la cual consta de 5 simbolos R1 y 5 A intercalados
90      *****/
91      for (i=0;i<Trama_sincro ; i+=Long_tram_XA_D){
92          // Insertamos en posiciones pares pilotos A
93          for (j=0;j<Long_tram_XA_D ; j++)
94              aux[j] = xAofdm[j];
95
96          ////Simulación de ruido AWGN
97          if(noise)
98              gen_noise(aux, Long_tram_XA_D, SNR);
99
100         for (j=0;j<Long_tram_XA_D ; j++)
101             sn[j+i] = aux[j];
102
103         i=i+aux1;
104
105         // Insertamos en posiciones nones pilotos B
106         for (j=0;j<Long_tram_XA_D ; j++)
107             aux[j] = xBofdm[j];
108
109         ////Simulación de ruido AWGN
110         if(noise)
111             gen_noise(aux, Long_tram_XA_D, SNR);
112
113         for (j=0;j<Long_tram_XA_D ; j++)
114             sn[j+i] = aux[j];
115     }
116     /*
117
118     *****
119     funcion de Sincronizacion burda
120     *****
121     */
122     ix = Vector_PSI(ix, sn, Nbits_simbolo, D, psi);
123     // Obtenemos indice del maximo valor de correlación
124     //y busca el inicio del simbolo:
125     aux1 = 0;
126     for (j=0;j<Long_tram_XA_D ; j++){
127         if(psi[j] > aux1){
128             i = j; //indice del valor maximo
129             aux1 = psi[j];
130         }
131     }
132     }
133     /*****

```

```

130      *****
131      Etapa de sincronización fina
132      *****
133      *****/
134
135      /*****
136      Se generan las trama de control para la sincronización
137      fina , la cual consta de 10 simbolos A
138      *****/
139      aux1 = (Nptos_fft+D)*5;
140      ix= abs(aux1-(ix+i));
141      // Realiza el proceso de sincronización fina con
142      // solo 6 simbolos OFDM
143      for(i=0;i<6;i++){
144          //Simulamos el inicio de los simbolos
145          //con un retardo ix
146              for(j=0;j<Long_tram_XA_D;j++)
147                  xAofdm[j] = sn[j+ix];
148          //Recupera la señal analitica transmitida
149          Recep_Simbolo_OFDM(xAofdm,Y_analitica , co_re , co_im ,
150                          escala);
151          // Operacion angular
152              for(j=0;j<QAMsimbolos;j++)
153                  DeltaPhi[j] = angle_pha (X_analitica[j] ,
154                                          Y_analitica[j]);
155          // funcion del comando unwrap de matlab , corrige la fase
156          // angular
157          //compensando a 2pi
158          //Correccion de fase
159          j = 2;
160          aux1 = 100 * pi;
161          aux2 = 2 * pi *100;
162          while(j < QAMsimbolos){
163              if(abs(DeltaPhi[j-1]-DeltaPhi[j]) > aux1)
164                  if(DeltaPhi[j-1]>0)
165                      DeltaPhi[j] = DeltaPhi[j] + aux2;
166                  else
167                      DeltaPhi[j] = DeltaPhi[j] - aux2;
168              else
169                  j++;
170          }
171          aux1 = aux2 = aux4 = aux3 = 0;
172          for(j=1;j<QAMsimbolos;j++){
173              delta = DeltaPhi[j];
174              aux1 = aux1 + j * delta;
175              aux2 = aux2 + j;
176              aux3 = aux3 + delta;
177              aux4 = aux4 + j * j;
178          }
179          //lls realiza el ajuste fino usando minimos cuadrados

```



```
178         delta = (Nbits_simbolo/100 * ((1/(2*pi))*(QAMsimbolos*
179         (double)aux1 - (double)aux2 * (double)aux3)/
180         (QAMsimbolos * (double)aux4 - (double)aux2 *
181         (double)aux2)));
182     //Simula el ajuste del VCO
183     ix=ix+Nbits_simbolo+D+delta;
184     j = 1;
185     }
186 }
```

---





Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

# ACTA DE EXAMEN DE GRADO



No. 00020

Matrícula: 210383436

PROGRAMACION Y EVALUACION EN UN DSP DEL NIVEL FISICO DE UN MODEM OFDM PARA COMUNICACION POR LA RED ELECTRICA DOMESTICA

En México, D.F., se presentaron a las 15:00 horas del día 30 del mes de noviembre del año 2012 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

- DR. RICARDO BARRON FERNANDEZ
- DR. GERARDO ABEL LAGUNA SANCHEZ
- DR. ENRIQUE RODRIGUEZ DE LA COLINA

DANIEL TORRES ALVARADO  
ALUMNO

Bajo la Presidencia del primero y con carácter de Secretario el último, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:


MAESTRO EN CIENCIAS (CIENCIAS Y TECNOLOGIAS DE LA INFORMACION)

DE: DANIEL TORRES ALVARADO

y de acuerdo con el artículo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

*aprobar*


REVISÓ



LIC. JULIO CESAR DE LARA ISASSI  
DIRECTOR DE SISTEMAS ESCOLARES

Acto continuo, el presidente del jurado comunicó al interesado el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

DIRECTOR DE LA DIVISIÓN DE CBI



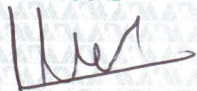
DR. JOSE ANTONIO DE LOS REYES  
HEREDIA

PRESIDENTE



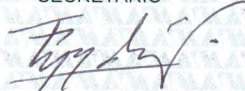
DR. RICARDO BARRON FERNANDEZ

VOCAL



DR. GERARDO ABEL LAGUNA SANCHEZ

SECRETARIO



DR. ENRIQUE RODRIGUEZ DE LA COLINA