



Universidade Federal do ABC

Trabalho de Graduação

Pedro Ivo da Cruz

Projeto de um Sistema OFDM Codificado empregando a
Técnica de Rádio Definido por Software

Santo André - SP
2014

Pedro Ivo da Cruz

Projeto de um Sistema OFDM Codificado empregando a
Técnica de Rádio Definido por Software

Trabalho de Graduação

Relatório de Trabalho de Graduação apresentado ao Curso de Engenharia de
Informação como requisito parcial para obtenção do grau de Engenheiro de
Informação.

Orientador: Prof. Dr. Ivan Roberto de Santana Casella

SANTO ANDRÉ

2014

Pedro Ivo da Cruz

Projeto de um Sistema OFDM Codificado empregando a Técnica de Rádio Definido por Software

Esse relatório de Trabalho de Graduação foi julgado e aprovado para a obtenção do grau em Bacharel em Engenharia da Informação no curso de Engenharia da Informação na Universidade Federal do ABC.

Santo André – SP, 18 de Dezembro de 2014

Prof. Dr. Ricardo Suyama
Coordenador do Curso

BANCA EXAMINADORA

Prof. Dr. Ivan de Santana Casella
Orientador

Prof. Dr. Ricardo Suyama
UFABC

Prof. Dr. Rodrigo Muñoz
UFABC

Agradecimentos

Gostaria de agradecer à Universidade Federal do ABC, que me concedeu a oportunidade de estudar uma área do conhecimento que é uma paixão para mim.

Gostaria também de agradecer a todos os professores que me deram aula ao longo dessa jornada na UFABC, principalmente aos do curso de Engenharia de Informação.

Entre eles, gostaria de agradecer principalmente ao professor Ivan de Santana Casella, que me orientou ao longo de todos esse trabalho, prestando ajuda desde os conceitos até a implementação final.

Gostaria também de expressar meus agradecimentos à empresa National Instruments, que possibilitou a execução desse trabalho ao me emprestar uma USRP.

Também gostaria de agradecer aos meus pais pelo apoio e compreensão que têm me dado e que é, de certa forma, muito importante para mim.

À minha namorada, Karen, pelo carinho, apoio e paciência, que foram fundamentais não só durante esse ano, mas nos últimos três.

Resumo

O principal objetivo deste trabalho é implementar um sistema de comunicação digital OFDM (*Orthogonal Frequency Division Multiplexing*) codificado empregando a técnica de *Software Defined Radio* e analisar seu desempenho para diferentes condições de propagação. A análise de desempenho será feita através da taxa de erro de bits (BER), primeiramente para um sistema simulado, e em seguida para um sistema utilizando uma USRP (Universal Software Radio Peripheral). Ambas análises levarão em conta um sistema OFDM sem codificação, e sistemas com codificação Reed-Solomon para diferentes capacidades de correção de erros.

Abstract

The main goal of this work is to implement a coded OFDM (Orthogonal Frequency Division Multiplexing) digital communication system employing the Software Defined Radio and analyze its performance for different conditions of propagation. The performance analysis will be performed through the bit error rate (BER), first for a simulated system, and then for a system using the USRP (Universal Software Radio Peripheral). Both analyses will consider a OFDM system with no code, and systems using the RS codes with different error correction capacity.

SUMÁRIO

1. Introdução.....	1
1.1. Comunicações Sem Fio	2
1.2. Modelos de Canais.....	3
1.2.1. Caminhos Múltiplos e Atenuação em Pequena Escala	4
1.2.2. Modelo de Canal Seletivo em Frequência com Distribuição de Atenuação de Rayleigh.....	5
1.3. OFDM.....	7
1.3.1. Ortogonalidade	10
1.3.3. Intervalo de Guarda	13
1.3.4. Prefixo Cíclico.....	14
1.3.5. Zero Padding	16
1.3.6. Sincronização usando o Prefixo Cíclico.....	18
1.3.7. Estimação e Sincronização por Máxima Verossimilhança.....	19
1.3.8. Estimação de Canal.....	22
1.4. Codificação Reed-Solomon.....	25
1.4.1. Propriedades do Código Reed-Solomon	25
1.4.2. Processo de Codificação Reed Solomon	27
1.4.3. Processo de Decodificação Reed-Solomon	29
1.5. Contribuições e Publicações Geradas	32
2. Técnicas Usadas no Desenvolvimento de um Modem OFDM-RS	33
2.1. Radio Definido Por Software	33
2.2. USRP	34
3. Simulação de um Modem OFDM-RS	36
3.1. Descrição do Modelo.....	36
3.2. Análise de Desempenho	37
4. Implementação do Modem OFDM-RS Utilizando a USRP	42
4.1. Descrição do Sistema Implementado.....	42
4.2. Blocos Funcionais	43
4.2.1. Fonte de Dados	43
4.2.2. Cálculo da BER	44
4.2.3. Outras funções	44
4.2.4. Configuração da USRP	46

4.2.5. Transmissor e Receptor OFDM-RS.....	47
4.3. Análise de Desempenho do Modem OFDM-RS.....	49
5. Conclusão e Futuros Trabalhos.....	55
6. Referências Bibliográficas	56
Anexo A - Cálculo de E_b/N_0 a partir dos valores de potência medidos	58
Anexo B - Valores Obtidos nas Medições	60
Anexo C - Apresentação do Modelo em LabVIEW do modem OFDM-RS	63

1. Introdução

A comunicação tem sido uma das mais profundas necessidades da humanidade em toda a história. Ela é necessária para educar os jovens, manter e propagar a cultura de uma região e para formar laços sociais entre as pessoas [1].

É comum dizer-se que o coração dos princípios de comunicação foi desenvolvido no *Math Center* na *Bell Laboratories*. Talvez a mais importante contribuição desse centro foi a criação da Teoria da Informação, por Claude Shannon, em 1948 [1]. Por cerca de 25 anos, foi considerada uma teoria muito bonita, mas não como um guia central para o desenvolvimento de sistemas de comunicação. Mas após um tempo, foi essa mesma teoria que possibilitou o desenvolvimento das comunicações digitais.

Duas ideias vindas dessa teoria são muito importantes no projeto de qualquer sistema de comunicação digital. A primeira, é a da informação sendo representada por sequências binárias. A segunda é a de criar um sistema de comunicação que primeiro converte a fonte de informação em uma sequência binária e depois converte essa sequência em uma forma adequada para transmissão em meios como cabos, par trançados, fibras ópticas, ou radiação eletromagnética através do espaço [1].

Sistemas de comunicação digital, por definição, são sistemas que usam uma sequência digital como interface entre a fonte e o canal de transmissão. Como pode ser visto no esquema de blocos mostrado na Figura 1, o codificador de fonte transforma a informação em uma sequência binária e o modulador processa a sequência binária para transmissão pelo canal de forma eficiente. O demodulador detecta e recria a sequência binária e o decodificador de fonte recria a informação no seu formato original [1].

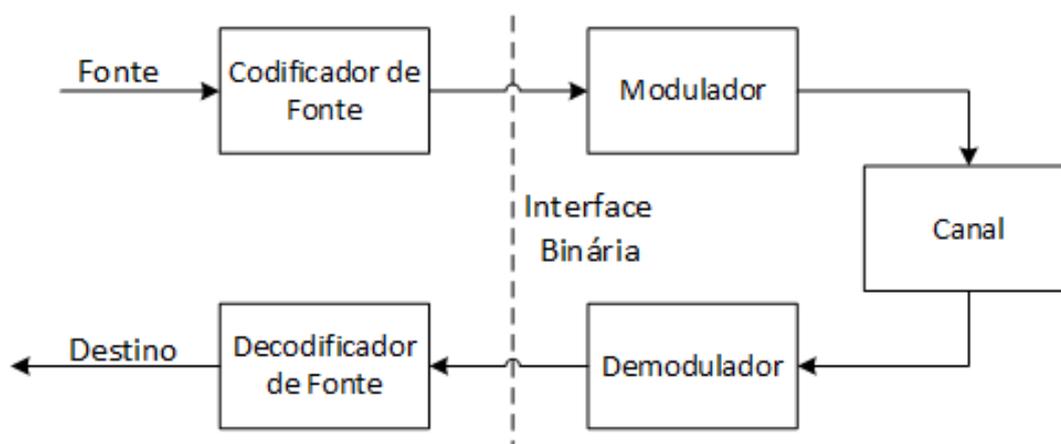


Figura 1 - Interface Binária.

Existem alguns motivos para os sistemas de comunicação digital terem se tornado um padrão. O barateamento de componentes eletrônicos, diminuição da escala e aumento da confiabilidade do hardware, a simplicidade da implementação e entendimento do sistema, e a padronização de uma interface binária entre fonte e canal são alguns desses motivos [1].

1.1. Comunicações Sem Fio

Nos últimos 10 anos a indústria de comunicações via rádio cresceu em ordem de grandeza, alimentada pela melhoria na fabricação de circuitos digitais e de radiofrequência (RF), bem como pela integração de circuitos em grande escala e outras tecnologias de miniaturização que tornaram os equipamentos de rádio ainda menores, mais baratos e mais confiáveis. As técnicas de comutação digital permitiram a implantação de redes de comunicação de rádio em grande escala, fáceis de usar e a preços acessíveis [2].

Devido ao crescimento da importância desse meio de transmissão e recepção de informação, diversos estudos foram realizados nessa área e diversas tecnologias foram desenvolvidas ao longo do tempo, sempre com o objetivo de obter uma maior qualidade e maior velocidade e desempenho dos sistemas de comunicação sem fio.

É nesse contexto que esse trabalho está inserido, focando em um dos sistemas de comunicação digital mais utilizados hoje em dia em sistemas sem fio, que é o sistema *Orthogonal Frequency Division Multiplexing* (OFDM), e na implementação de um modem (modulador/demodulador) juntamente com um

codificador e decodificador Reed-Solomon (RS), no intuito de melhorar o seu desempenho.

Uma das grandes vantagens de um sistema OFDM, como veremos a seguir, é a diminuição dos efeitos de um canal sem fio com multipercurso. Para elucidar melhor esse problema, a próxima seção irá descrever como um canal de comunicação pode ser modelado. Em seguida, serão descritos os principais tipos de modelos de canais sem fio que iremos utilizar nesse trabalho, como por exemplo o canal com desvanecimento seletivo em frequência. Na seção seguinte a essa, fala-se sobre o sistema OFDM, sua sincronização e estimação de canal através de portadoras pilotos. Em seguida, explica-se os códigos RS, abordando assim todos os fundamentos teóricos para a realização desse trabalho. Em seguida, será abordado a técnica de rádio definido software na prototipagem de sistemas de comunicações, focando na plataforma USRP. Na seção seguinte desse trabalho, estarão os resultados de algumas simulações desse sistema, e após isso a descrição da implementação do sistema OFDM-RS na USRP e sua análise.

1.2. Modelos de Canais

Em geral, o canal é visto como a parte em um sistema de comunicação que fica entre o transmissor e o receptor, do qual o projetista não possui nenhum controle.

A princípio, em um canal sem ruído, é possível transmitir um número infinito de dígitos binários, transmitindo um único número real. Mas Shannon, em 1948, percebeu que o ruído provê uma limitação fundamental na capacidade de transmissão de um canal.

A forma mais comum de canal envolve um sinal de entrada $x(t)$ um ruído aditivo $z(t)$ e um sinal de saída $y(t) = x(t) + z(t)$, como mostra a Figura 2. Cada um desses sinais é tratado como aleatório. O ruído $z(t)$ é usualmente modelado como um ruído Branco Gaussiano Aditivo (AWGN).

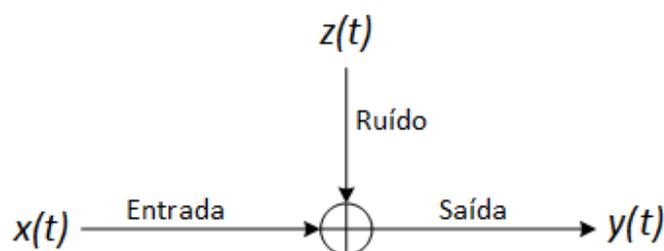


Figura 2 - Canal Com Ruído AWGN.

Em um modelo de canal mais geral, chamado de Canal Linear Gaussiano, a entrada passa por um filtro linear de resposta ao impulso $h(t)$ e então um ruído independente Gaussiano é adicionado ao sinal resultante, de forma que a saída do canal é dada por:

$$y(t) = x(t) * h(t) + z(t) \quad (1)$$

Onde $*$ denota a operação de convolução. Na Figura 3 pode ser visto o diagrama de blocos que representa esse canal.

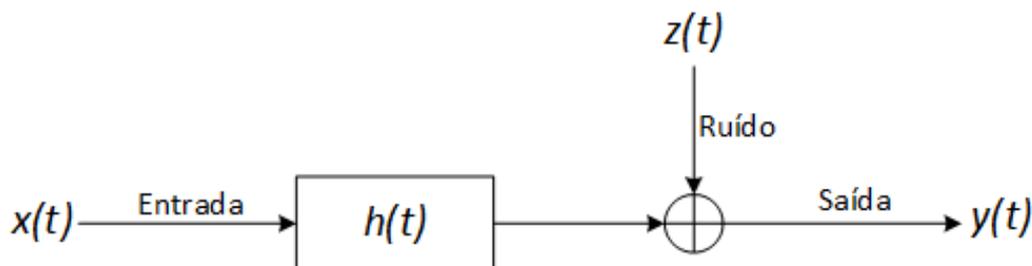


Figura 3 - Canal Linear Gaussiano.

O canal linear Gaussiano é frequentemente um bom modelo de canal para sistemas sem fio (wireless) onde não há obstruções [1].

1.2.1. Caminhos Múltiplos e Atenuação em Pequena Escala

No entanto, no contexto de comunicações móveis, esse pode ser um modelo bem pobre se a resposta ao impulso do canal for invariante no tempo. Nesse tipo de comunicações, o canal geralmente possui obstruções e existe multipercurso entre a fonte e o destino do sinal transmitido. A mobilidade da fonte, do destino e as reflexões dos obstáculos faz com que os percursos variem com o tempo de uma forma que pode ser melhor modelada de forma aleatória.

Dessa forma, a resposta ao impulso do canal $h(t)$ pode ser substituída por um filtro linear e que varie no tempo aleatoriamente $h(t, \tau)$.

O multipercurso faz com que o receptor receba variações do sinal transmitido com atrasos, e quanto maior esses atrasos, maior será a dispersão temporal dos símbolos, introduzindo interferência intersimbólica (ISI) [3]. Esse espalhamento temporal pode ocorrer em toda a faixa de frequências do sinal, configurando o chamado Desvanecimento Plano, ou poderá afetar somente uma determinada faixa de frequências, configurando o Desvanecimento Seletivo em Frequência. A ocorrência de desvanecimento plano ou seletivo em frequência pode ser determinado através da Largura da Banda de Coerência do canal, que fornece uma medida estatística da faixa de frequência pela qual o canal pode ser considerado plano. A Largura da Banda de Coerência é inversamente proporcional ao Espalhamento Temporal do canal e o seu valor exato depende da definição de um valor para a correlação entre as amplitudes das componentes espectrais, não existindo, portanto, uma fronteira nítida que separe um canal seletivo daquele que possa ser considerado plano [4].

A movimentação que ocorre no transmissor e no receptor ocasiona na variação dos caminhos tomados pelo sinal. Isso faz com que ocorra uma variação da fase no sinal recebido que pode ser visto como uma variação de frequência, formando o Espalhamento Doppler. O Tempo de Coerência é uma medida estatística que informa o intervalo de tempo em que a resposta em frequência $h(t, \tau)$ do canal pode ser considerada constante. O Tempo de Coerência é inversamente proporcional ao Espalhamento Doppler e, de maneira análoga à anterior, não fornece uma fronteira nítida entre um canal que varia rapidamente, configurando um Desvanecimento Rápido, e aquele que varia lentamente, configurando um Desvanecimento Lento. Na prática assume-se que um canal pode ser considerado lento se suas características não se alteram entre dois intervalos de sinalização consecutivos do sinal transmitido [4].

1.2.2. Modelo de Canal Seletivo em Frequência com Distribuição de Atenuação de Rayleigh

O desvanecimento em pequena escala é o tipo de desvanecimento que geralmente está associado ao multipercurso. Em canais de comunicação

móveis, geralmente a distribuição de Rayleigh é usada para descrever a natureza estatística e variante no tempo do canal [2].

A função de distribuição de probabilidades que descreve a amplitude do sinal para o desvanecimento em pequena escala é a função de Rayleigh, mostrada a seguir:

$$P_r = \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} \quad (2)$$

Onde σ^2 é a potência média do sinal. Dessa forma, um canal com desvanecimento em pequena escala, onde há multipercurso, pode ser modelado com a envoltória de $h(t, \tau)$ respeitando a distribuição Rayleigh. Esse canal é chamado de canal de desvanecimento Rayleigh [5].

O modelo de Rayleigh pode ser usado em conjunto com ganhos variáveis e tempos de atraso para produzir efeitos de desvanecimento seletivo em frequência. Isso pode ser visto na Figura 4.

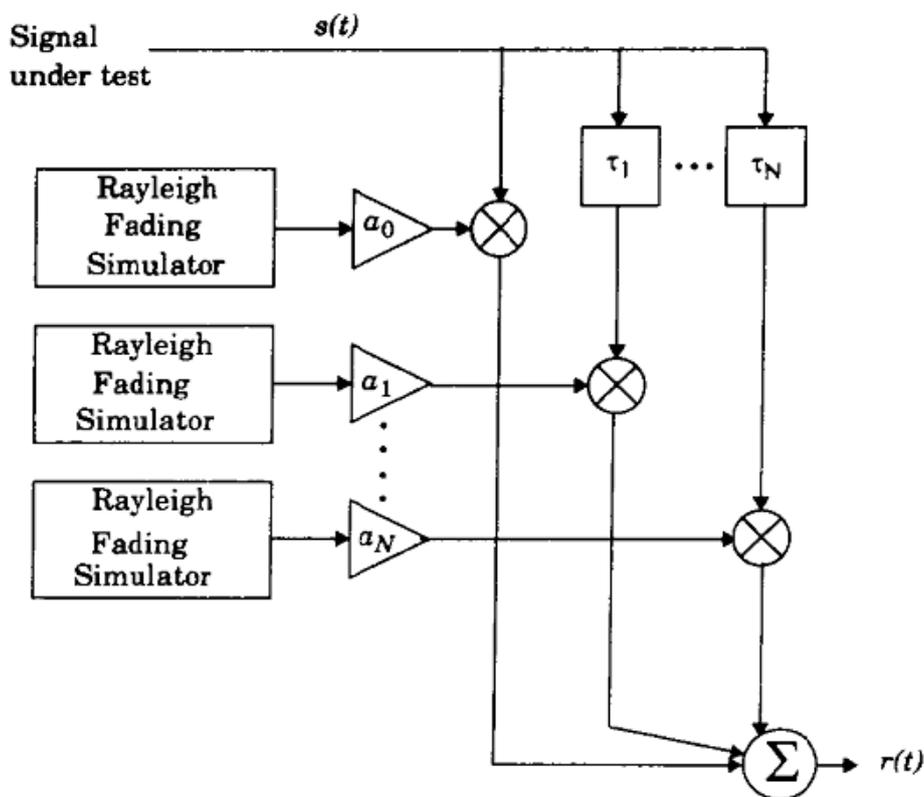


Figura 4 - Simulação de um canal seletivo em frequência usando a função de distribuição Rayleigh [2].

1.3. OFDM

Sistemas de uma única portadora são mais simples de serem implementados e, para suportarem uma taxa de símbolos R_s símbolos por segundo, a largura de banda mínima requerida é a largura de banda de Nyquist, dada por $R_s/2[Hz]$ em banda passante e por R_s em banda passante [6]. Portanto, para uma maior taxa de transmissão de símbolos, é necessária uma banda maior em um sistema de única portadora. Quando a largura de banda do sinal ultrapassa a banda de coerência de um canal sem fio, o sinal sofre com desvanecimento por multipercurso, ficando sujeito a ISI (Interferência intersimbólica). Geralmente, equalizadores adaptativos são o suficiente para lidarem com interferência a ISI, porém, quanto maior a taxa de transmissão, mais complexos esses equalizadores se tornam. Disso, conclui-se que um sistema de transmissão de portadora única e de alta taxa, pode não ser praticável, devido à complexidade desses equalizadores no receptor.

Para superar os efeitos de um canal seletivo em frequência, múltiplas portadoras podem ser utilizadas, cada uma com uma taxa de transmissão menor, resultando, ao todo, em um sistema com alta taxa de transmissão.

Em sistemas multicanal, a banda total é dividida entre N subcanais de banda estreita, os quais possuem cada um uma portadora em frequências diferentes. Sistemas multiportadoras são uma forma específica dos sistemas multicanal, onde diferentes símbolos são transmitidos em canais ortogonais [6].

O esquema de transmissão OFDM é outro sistema multicanal que faz o uso de múltiplas portadoras. Enquanto sistemas como FMT, mostrado na Figura 5 utilizam filtros de banda limitada e osciladores para cada sub-canal e tem sua banda totalmente dividida em N sub-canais ortogonais, o OFDM tem o espectro de suas subportadoras sobrepostos, obtendo uma maior eficiência espectral. Na prática, a transformada discreta de Fourier (DFT) e a transformada discreta de Fourier inversa (IDFT) são úteis para obter esses sinais ortogonais. Algoritmos como *Fast Fourier Transform* (FFT) e *Inverse Fast Fourier Transform* (IFFT) são usados para implementar eficientemente a DFT e a IDFT.

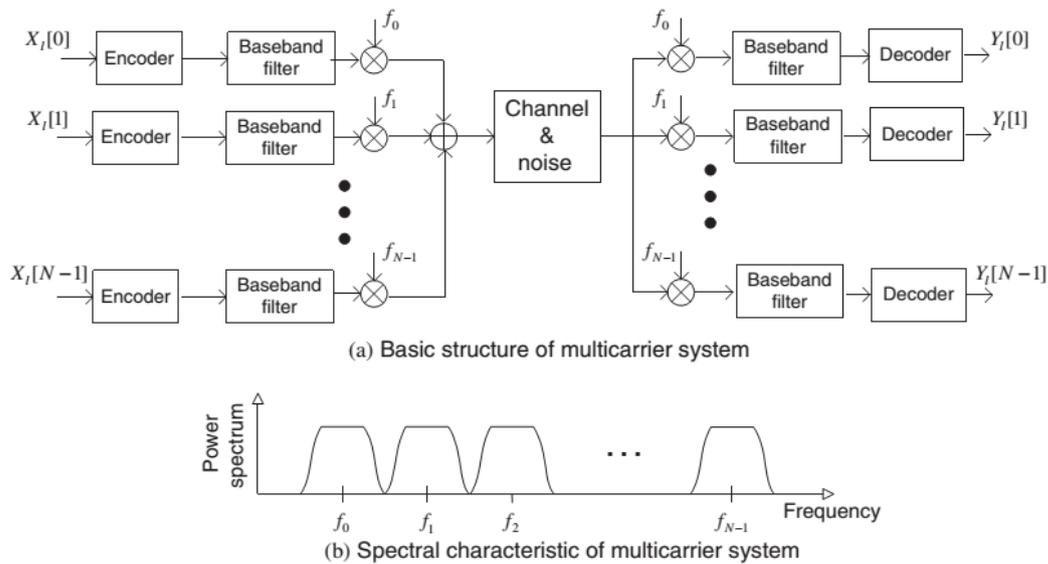


Figura 5 - Estrutura básica de um sistema multi-portadora e suas características espectrais [6].

No sistema de transmissão OFDM é calculada a IFFT de N pontos, gerando os símbolos OFDM. No receptor, o símbolo OFDM contendo ruído do canal, passa pela operação de FFT de N pontos. Esse esquema de transmissão pode ser visto na Figura 6 através de um diagrama de blocos.

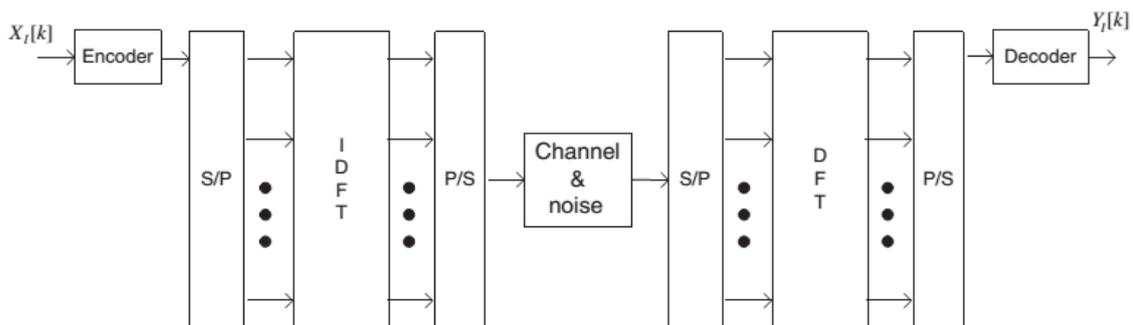


Figura 6 - Esquema de transmissão OFDM utilizando DFT e IDFT [6].

Como todas as portadoras possuem tempo de duração finito T , o espectro do sinal OFDM pode ser considerado como a soma de funções sinc deslocadas no domínio da frequência, onde o espaço entre elas é de $1/T$, como pode ser observado na Figura 7.

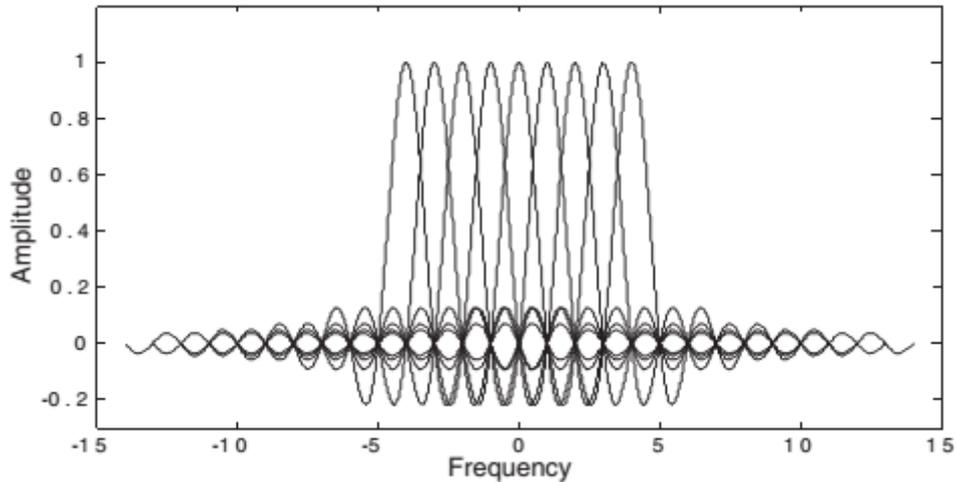


Figura 7 - Espectro de um sinal OFDM em escala linear [6].

Como o sinal de cada subportadora é limitado no tempo para cada símbolo, a banda do sinal não é limitada. Isso quer dizer que um sinal OFDM pode produzir radiação fora da banda utilizada, o que causa interferência de canal adjacente. Isso é visto claramente na Figura 8, que representa o espectro de potência do sinal OFDM com a escala em dB, onde a potência do primeiro lóbulo lateral não é tão pequeno quando comparado com a dos lóbulos principais.

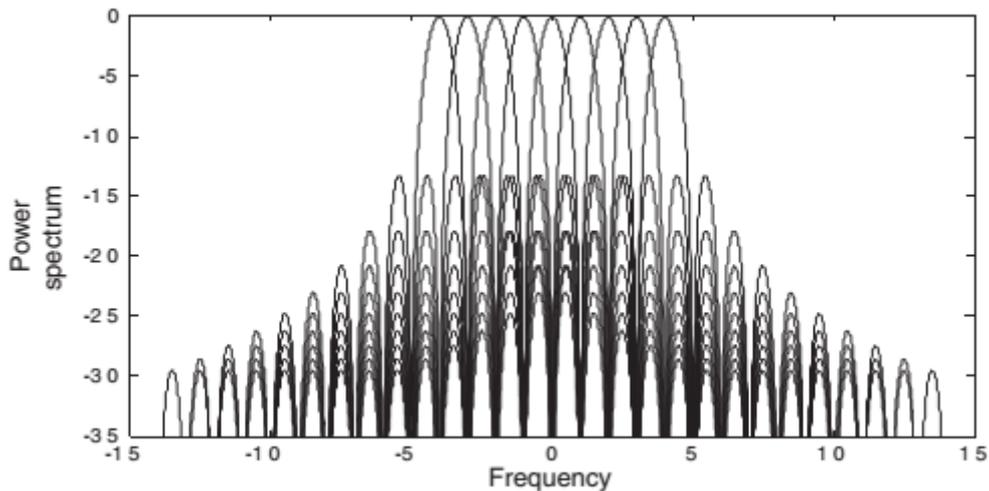


Figura 8 - Espectro de potência de um sinal OFDM em dB [6].

Por isso, o esquema de transmissão OFDM adiciona uma banda de guarda nas subportadoras externas, chamadas de portadoras virtuais (ou *virtual carriers* – VC) em volta da banda principal para reduzir a radiação fora da banda.

O transmissor OFDM também insere um intervalo de guarda no domínio do tempo, chamado de prefixo cíclico (CP), no intuito de mitigar a interferência intersimbólica entre os símbolos OFDM. A seguir serão detalhados esses assuntos.

1.3.1. Ortogonalidade

Seja o sinal exponencial complexo limitado no tempo $\{e^{j2\pi f_k t}\}_{k=0}^{N-1}$, que representa cada subportadora em $f_k = k/T_{sym}$ no sinal OFDM, onde $0 \leq t \leq T_{sym}$. Esses sinais são ortogonais se o produto entre períodos comuns (fundamentais) é zero, como pode ser visto a seguir.

$$\begin{aligned} \frac{1}{T_{sym}} \int_0^{T_{sym}} e^{j2\pi f_k t} e^{-j2\pi f_i t} dt &= \frac{1}{T_{sym}} \int_0^{T_{sym}} e^{j2\pi \frac{k}{T_{sym}} t} e^{-j2\pi \frac{i}{T_{sym}} t} dt \\ &= \frac{1}{T_{sym}} \int_0^{T_{sym}} e^{j2\pi \frac{(k-i)}{T_{sym}} t} dt \\ &= \begin{cases} 1, & \forall \text{ inteiro } k = i \\ 0, & \text{caso contrário} \end{cases} \end{aligned}$$

Amostrando nos instantes $t = nT_s = nT_{sym}/N$, essa equação pode ser representada no domínio discreto por:

$$\begin{aligned} \frac{1}{N} \sum_{n=0}^N e^{j2\pi \frac{k}{T_{sym}} nT_s} e^{-j2\pi \frac{i}{T_{sym}} nT_s} &= \frac{1}{N} \sum_{n=0}^N e^{j2\pi \frac{k}{T_{sym}} \frac{nT_{sym}}{N}} e^{-j2\pi \frac{i}{T_{sym}} \frac{nT_{sym}}{N}} \\ &= \frac{1}{N} \sum_{n=0}^N e^{j2\pi \frac{(k-i)}{T_{sym}} n} \\ &= \begin{cases} 1, & \forall \text{ inteiro } k = i \\ 0, & \text{caso contrário} \end{cases} \end{aligned}$$

Essa ortogonalidade é uma condição essencial para o funcionamento do sistema OFDM com zero de interferência entre portadoras (ICI).

1.3.2. Transmissão e Recepção OFDM

A Figura 9 mostra um esquema de modulação e demodulação do esquema OFDM.

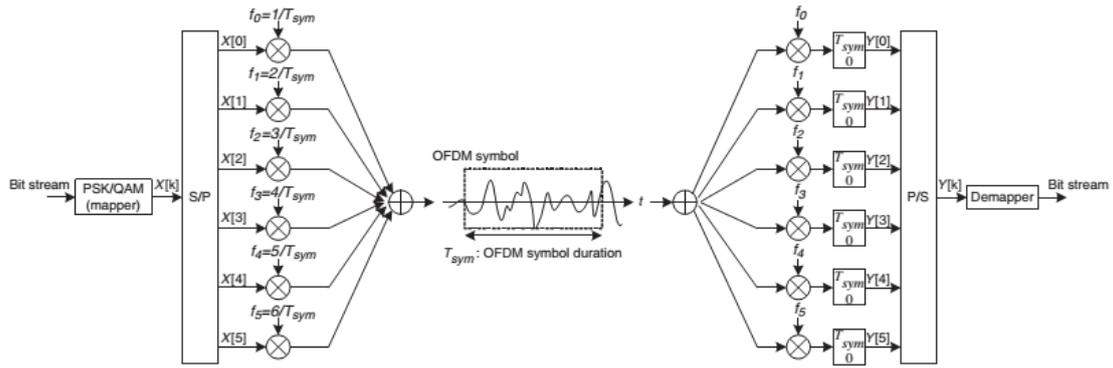


Figura 9 - Modulação e Demodulação OFDM [6].

O transmissor OFDM mapeia a mensagem, formada por uma sequência de bits, em uma sequência de símbolos *Phase Shift Key* (PSK) ou *Quadrature Amplitude Modulation* (QAM). Em seguida, esses símbolos são convertidos em N sequências paralelas. Cada N símbolos em paralelo será transmitido por uma subportadora diferente. Seja $X_l[k]$ o l -ésimo símbolo na k -ésima subportadora, onde $k = 0, 1, 2, \dots, N - 1$. Devido a conversão serial-paralelo, a duração da transmissão de N símbolos é de NT_s , que forma um único símbolo OFDM com tamanho T_{sym} ($T_{sym} = NT_s$).

Seja o l -ésimo sinal OFDM na k -ésima subportadora dado por:

$$\Psi_{l,k}(t) = \begin{cases} e^{j2\pi f_k(t-lT_{sym})}, & 0 \leq t \leq T_{sym} \\ 0, & \text{caso contrário} \end{cases} \quad (3)$$

Os sinais OFDM em banda passante e em banda base em tempo contínuo podem ser escritos, respectivamente, como:

$$x_l(t) = \text{Re} \left\{ \frac{1}{T_{sym}} \sum_{l=0}^{\infty} \left\{ \sum_{k=0}^{N-1} X_l[k] \Psi_{l,k}(t) \right\} \right\} \quad (4)$$

e

$$x_l(t) = \sum_{l=0}^{\infty} \sum_{k=0}^{N-1} X_l[k] e^{j2\pi f_k(t-lT_{sym})} \quad (5)$$

Amostrando a equação do sinal OFDM banda base nos instantes $t = lT_{sym} + nT_s$, com $T_s = T_{sym}/N$ e $f_k = k/T_{sym}$, o símbolo OFDM em tempo discreto pode ser escrito como:

$$x_l[n] = \sum_{k=0}^{N-1} X_l[k] e^{j2\pi kn/N} \quad (6)$$

Para $n = 0, 1, \dots, N - 1$.

Essa equação em tempo discreto nada mais é do que a IDFT de N pontos dos símbolos PSK ou QAM, e pode ser realizada eficientemente através do algoritmo de IFFT.

O sinal banda base OFDM recebido pode ser escrito como:

$$Y_l[k] = \sum_{k=0}^{N-1} X_l[k] e^{j2\pi f_k(t-lT_{sym})}, lT_{sym} < t \leq lT_{sym} + nT_s \quad (7)$$

O símbolo transmitido $X_l[k]$ pode ser reconstruído a partir da ortogonalidade entre as subportadoras, sem levar em consideração os efeitos do canal e o ruído, chega-se em [6]:

$$Y_l[k] = \sum_{n=0}^{N-1} \sum_{i=0}^{N-1} X_l[i] e^{j2\pi(i-k)n/N} = X_l[k] \quad (8)$$

Essa equação, de fato, é a DFT de N pontos do sinal recebido e pode ser eficientemente computada através do algoritmo de FFT.

Dessa forma, ao invés de implementar um sistema OFDM como ilustrado na Figura 6, pode-se usar os algoritmos de IFFT e FFT no transmissor e no receptor, respectivamente, como pode ser visto na Figura 10.

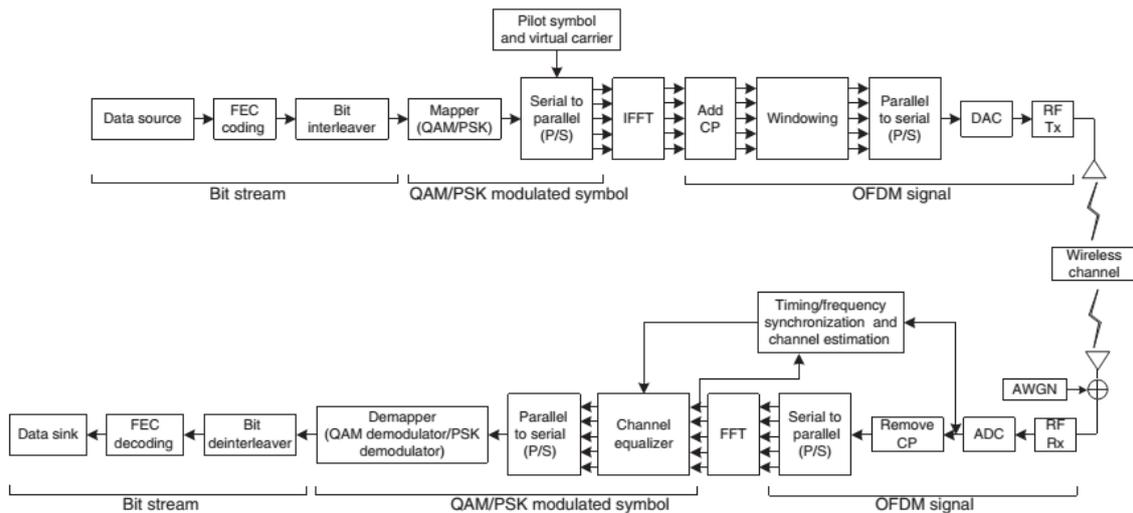


Figura 10 - Diagrama de blocos de um sistema OFDM utilizando os algoritmos de IFFT e FFT [6].

Cada um dos blocos como o de adição de CP, serão descritos nos tópicos a seguir.

1.3.3. Intervalo de Guarda

Para um canal com uma resposta ao impulso $h_l(t)$ e $z_l(t)$ ruído aditivo Gaussiano, o sinal recebido pode ser dado por:

$$y_l(t) = x_l(t) * h_l(t) + z_l(t) \quad (9)$$

No tempo discreto pode ser escrito como:

$$y_l[n] = x_l[n] * h_l[n] + z_l[n] \quad (10)$$

Onde $x_l[n] = x_l(nT_s)$, $y_l[n] = y_l(nT_s)$, $h_l[n] = h_l(nT_s)$, e $z_l[n] = z_l(nT_s)$

Ao se estender a duração efetiva do símbolo OFDM (T_{sub}), em N vezes ($T_{sub} = NT_s$) o efeito de desvanecimento multipercurso do canal é bastante reduzido no símbolo OFDM [6]. No entanto, esse efeito ainda permanece como um fator prejudicial à condição de ortogonalidade entre as subportadoras do sistema OFDM. Como pode ser visto na Figura 11, o primeiro símbolo OFDM (plotado em linha sólida) é misturado com o segundo símbolo OFDM (plotado em linha pontilhada), o que implica na ocorrência de ISI. Portanto, as subportadoras não são mais ortogonais durante a duração do símbolo OFDM.

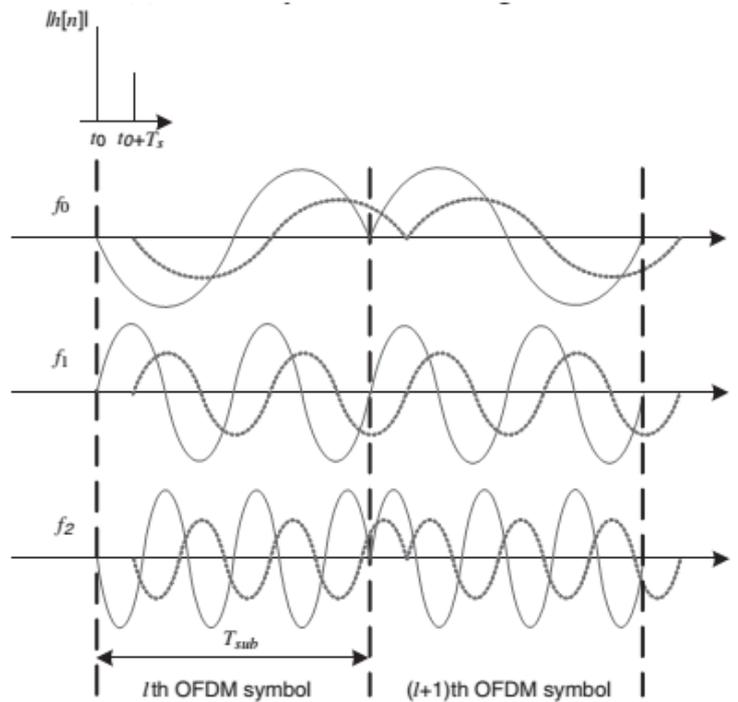


Figura 11 - ISI ocasionada por um canal multipercorso no sinal recebido [6].

Para garantir a performance do sistema OFDM, existem alguns meios de lidar com a ISI ocasionada pelos efeitos do canal multipercorso. Uma dessas maneiras, é a utilização de um intervalo de guarda entre dois símbolos OFDM consecutivos, como será visto a seguir.

1.3.4. Prefixo Cíclico

O intervalo de guarda pode ser inserido de duas formas. Uma é inserindo um intervalo com valores zeros entre os símbolos OFDM, chamada de *Zero Padding* (ZP). A outra é a extensão cíclica, usando o CP ou o Sufixo Cíclico (CS).

O CP insere o intervalo de guarda copiando as últimas amostras do símbolo OFDM na frente do símbolo. Seja T_G o tamanho do CP em termos de amostras, o símbolo OFDM resultante terá tamanho $T_{sym} = T_{sub} + T_G$, a construção do símbolo OFDM com CP pode ser visto na Figura 12.

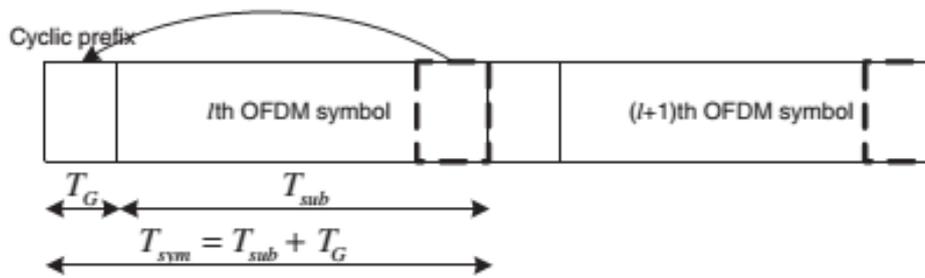


Figura 12 - Símbolo OFDM com CP [6].

Na Figura 13 pode ser visto que se o tamanho do CP for igual ou maior que ao atraso máximo do canal multipercurso, o efeito de ISI fica contido dentro do intervalo de guarda, não influenciando o cálculo da FFT do próximo símbolo OFDM. Isso quer dizer que, escolher um CP nessas condições, permite que a condição de ortogonalidade entre as portadoras seja mantida. Já a Figura 14 mostra o que acontece quando escolhe-se um CP menor. A parte final do símbolo OFDM afeta o início do próximo símbolo, resultando em ISI.

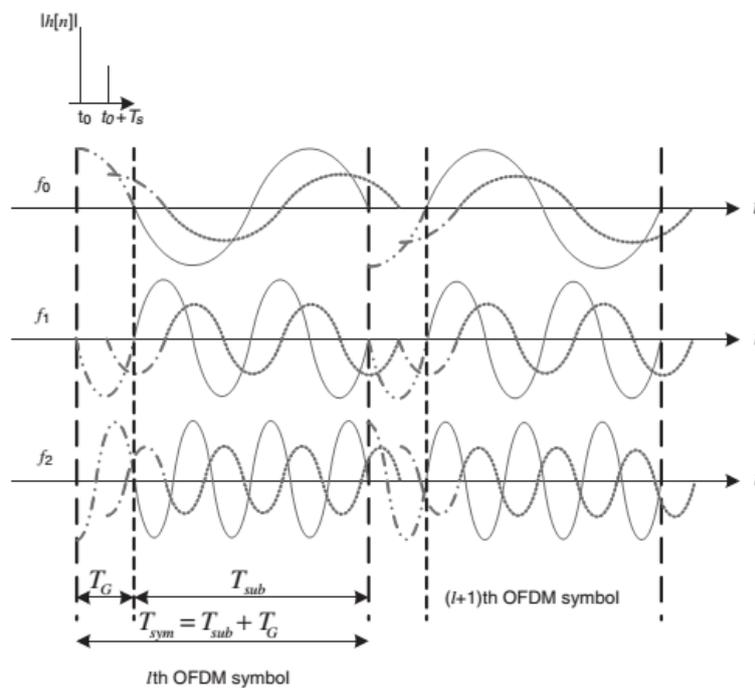


Figura 13 - Efeito de um canal multipercursos em símbolos OFDM com CP [6].

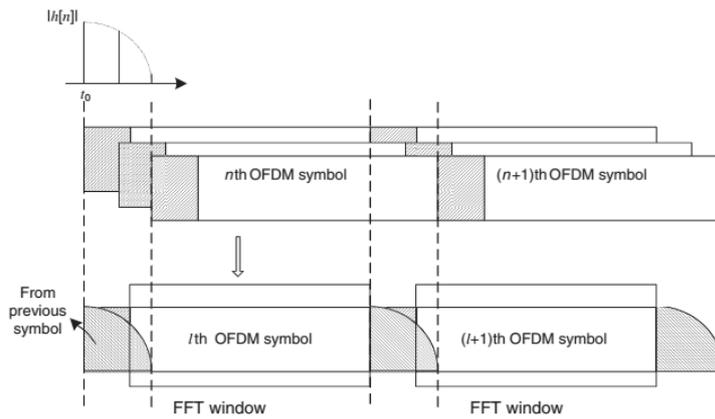


Figura 14 - ISI em um símbolo OFDM com CP menor que o atraso máximo do canal [6].

Assumindo que o tamanho de P não é menor que o atraso do canal, o receptor OFDM realiza a FFT das amostras recebidas $\{y_l[n]\}_{n=0}^{N-1}$ resultando em [6]:

$$Y_l[k] = H_l[k]X_l[k] + Z_l[k] \quad (11)$$

Onde $X_l[k]$ é a componente em frequência da k -ésima portadora do l -ésimo símbolo transmitido, $Y_l[k]$ é o símbolo recebido, $H_l[k]$ é a resposta em frequência do canal e $Z_l[k]$ é o ruído no domínio da frequência. Sem a dição de ruído, temos apenas $Y_l[k] = H_l[k]X_l[k]$, o que faz com que o símbolo transmitido possa ser detectado através de equalização *one-tap* [7], que divide o símbolo recebido pelo canal ($X_l[k] = Y_l[k]/H_l[k]$) Note que sem o CP, temos $Y_l[k] \neq H_l[k]X_l[k]$. De fato, essa igualdade só vale quando temos uma convolução circular no domínio do tempo discreto, que só pode ser obtida graças à inserção do prefixo cíclico no símbolo OFDM.

1.3.5. Zero Padding

Nós podemos inserir zeros dentro do intervalo de guarda. A Figura 15 mostra como fica um símbolo OFDM com Zero Padding (ZP) e a Figura 16 mostra o efeito da ISI em um canal com multipercurso.

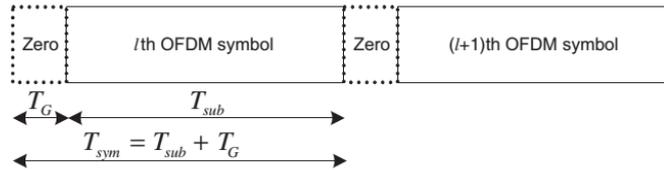


Figura 15 - Símbolo OFDM com ZP [6].

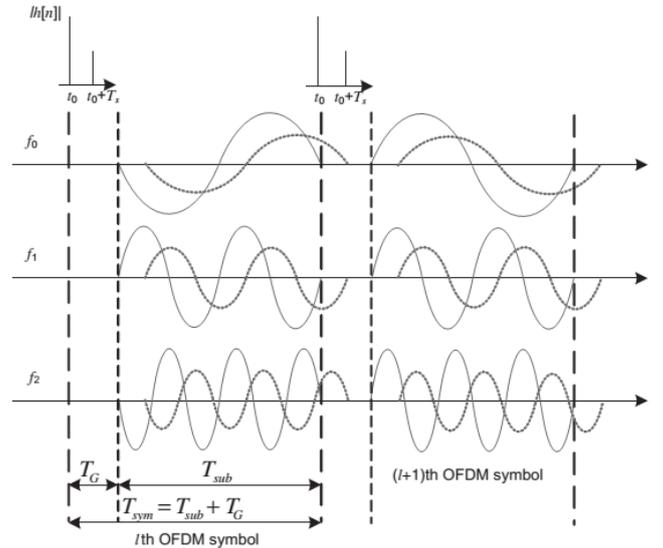


Figura 16 - Efeito de um canal multipercursos em símbolos OFDM com ZP [6].

Mesmo com um tamanho de ZP maior que o maior atraso de um canal multipercurso, um pequeno offset no tempo no símbolo (STO) pode fazer com que o símbolo OFDM de uma certa duração possua uma descontinuidade dentro da janela de FFT e portanto, o intervalo de guarda do próximo símbolo OFDM é copiado e adicionado ao início do símbolo OFDM atual, prevenindo assim a ocorrência de ICI, como mostra a Figura 17.

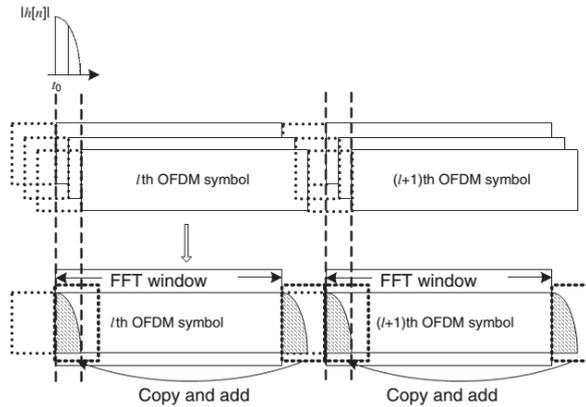


Figura 17 - Copiando e adicionando o Intervalo de Guarda do símbolo seguinte no início do símbolo OFDM para prevenir a ocorrência de ICI [6].

1.3.6. Sincronização usando o Prefixo Cíclico

Normalmente considera-se duas incertezas no receptor de um símbolo OFDM. A primeira, é a incerteza do tempo de chegada do símbolo, e é modelada como um atraso na resposta ao impulso do canal, ou seja, $\delta(k - \theta)$ onde θ é um número inteiro desconhecido que representa o tempo de chegada do símbolo. A segunda é a da frequência da portadora. Essa incerteza se dá por conta da diferença entre os osciladores locais no transmissor e receptor, que desloca o sinal no domínio da frequência. Este comportamento é modelado como uma distorção multiplicativa complexa no domínio do tempo, $e^{j2\pi\epsilon k/N}$, dos dados recebidos, onde ϵ denota a diferença entre os osciladores do transmissor e do receptor. Dessa forma, os dados recebidos podem ser descritos por:

$$r(k) = s(k - \theta)e^{j2\pi\epsilon k/N} + n(k) \quad (12)$$

O sinal transmitido $s(k)$ é a DFT dos símbolos mapeados da primeira modulação, que assumimos ser um processo branco, $s(k)$ é uma combinação linear de variáveis aleatórias independentes e identicamente distribuídas. Se o número de subportadoras é grande o suficiente, pelo teorema central do limite $s(k)$ é aproximadamente um processo Gaussiano complexo, no qual a parte real e imaginária são independentes. Esse processo, no entanto, não é branco, desde que a adição de um Prefixo Cíclico ocasiona uma certa correlação entre alguns pares de amostras, espaçadas em N amostras. Portanto, $r(k)$ também não é um processo Branco, mas contém informações sobre o offset no tempo θ e do offset na frequência da portadora ϵ através da sua estrutura probabilística.

Essa é uma informação crucial para que se possa estimar ao mesmo tempo esses parâmetros através de $r(k)$ [8]. Nas sessões seguintes, será mostrado como essa ideia é formalmente modelada e elaborada.

1.3.7. Estimação e Sincronização por Máxima Verossimilhança

Assume-se que se observa $2N + L$ amostras consecutivas de $r(k)$, conforme mostra a Figura 18, e que essas amostras contém um símbolo OFDM $(N + L)$ completo.

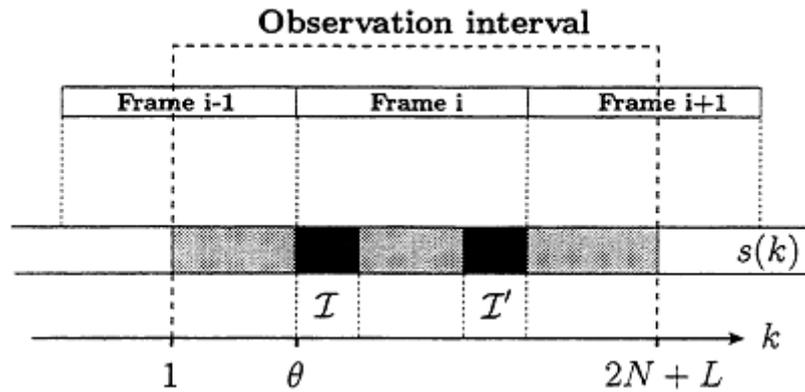


Figura 18 - Estrutura de um símbolo OFDM com prefixo cíclico e o intervalo de observação [8].

A posição desse símbolo, dentro do bloco de amostras observadas, é desconhecida, já que o atraso do canal θ é desconhecido. Seja os conjuntos de índices $I \triangleq [\theta, \theta + L - 1]$ e $I' \triangleq [\theta + N, \theta + N + L - 1]$, como mostra a Figura 18. O conjunto I' contém os índices das amostras de dados que são copiados na extensão cíclica, Enquanto I contém os índices dessa extensão. Colocando as amostras observadas em um vetor de dimensão $(2N + L) \times 1$ $\mathbf{r} \equiv [r(1) \dots r(2N + L)]^T$. As amostras no prefixo cíclico e suas réplicas são correlacionadas, isto é, seja $\sigma_s^2 = E\{|s(k)|^2\}$ e $\sigma_n^2 = E\{|n(k)|^2\}$, temos que para $\forall k \in I$:

$$E\{r(k)r^*(k + N)\} = \begin{cases} \sigma_s^2 + \sigma_n^2 & m = 0 \\ \sigma_s^2 e^{j2\pi\epsilon} & m = N \\ 0 & otherwise \end{cases} \quad (13)$$

Enquanto as amostras restantes ($k \notin I \cup I'$) permanecem não correlacionadas. Explorando essa propriedade dessa correlação, pode-se obter

o estimador *Maximum Likelihood* (ML), ou estimador por máxima verossimilhança, simultâneo de θ e ε .

A função logarítmica de verossimilhança para θ e ε é o logaritmo da função densidade de probabilidade das amostras $2N + L$ observadas em \mathbf{r} , dado o tempo de chegada θ e o offset em frequência da portadora ε . A estimação por máxima verossimilhança de θ e ε , é o argumento que maximiza essa [8]. Sobre a hipótese de que \mathbf{r} é um vetor Gaussiano, temos que a função log-likelihood pode ser mostrada por:

$$\Lambda(\theta, \varepsilon) = 2|\gamma(\theta)| \cos\{2\pi\varepsilon + \angle\gamma(\theta)\} - \rho\mathcal{E}(\theta) \quad (14)$$

Pode-se mostrar então que a estimação ML simultânea de θ e ε se torna:

$$\hat{\theta}_{ML} = \arg \max\{2|\gamma(\theta)| - \rho\mathcal{E}(\theta)\} \quad (15)$$

$$\hat{\varepsilon}_{ML} = -\frac{1}{2\pi} \angle\gamma(\hat{\theta}_{ML}) \quad (16)$$

Onde \angle denota o argumento de um número complexo e os termos de correlação e energia são dados respectivamente por:

$$\gamma(m) \equiv \sum_{k=m}^{m+L-1} r(k)r^*(k+N) \quad (17)$$

$$\mathcal{E}(m) \equiv \sum_{k=m}^{m+L-1} |r(k)|^2 + |r(k+N)|^2 \quad (18)$$

E ρ é a magnitude da correlação entre $r(k)$ e $r(k+N)$, sendo dado por:

$$\rho \equiv \frac{\sigma_s^2}{\sigma_s^2 + \sigma_n^2} \quad (19)$$

O diagrama de blocos desse estimador é mostrado na Figura 19.

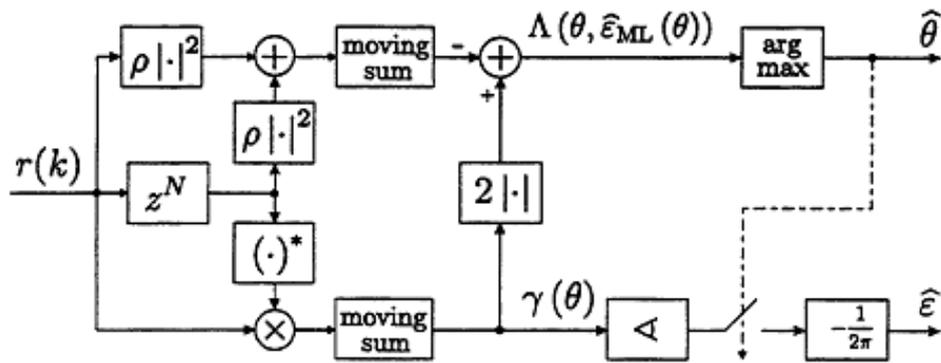


Figura 19 - Visão Geral do Estimador ML [8].

O processamento é realizado continuamente e dois sinais são gerados. O primeiro é $\Lambda(\theta, \varepsilon)$, cujos argumentos maximizados resultam na estimação dos instantes de tempo $\hat{\theta}_{ML}$. O segundo é o sinal $\gamma(\hat{\theta}_{ML})$, cujo argumento complexo no instante $\hat{\theta}_{ML}$ é proporcional à frequência estimada $\hat{\varepsilon}_{ML}$. Esses dois sinais estão representados na Figura 20.

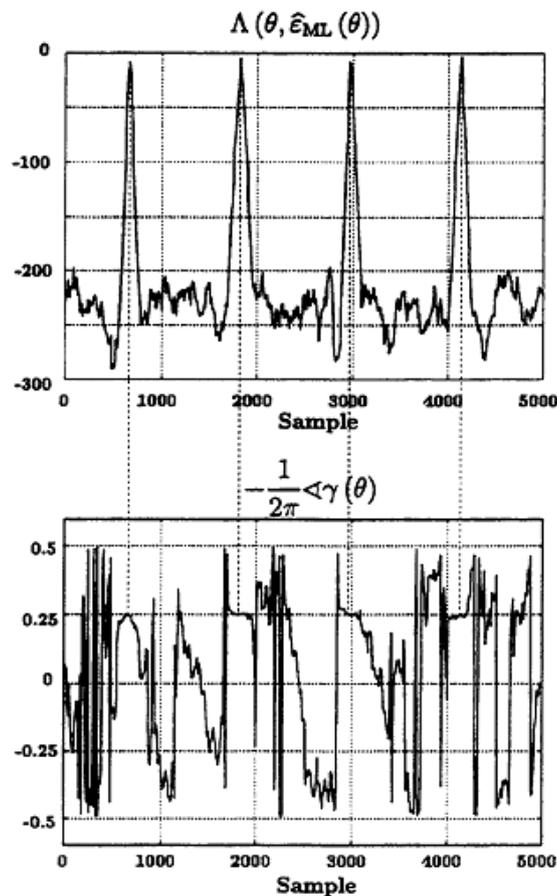


Figura 20 - Sinais que geram a estimação por ML [8].

1.3.8. Estimação de Canal

Como discutido anteriormente, o sinal recebido normalmente é distorcido pelo canal, de forma que, para realizar uma correta recuperação da informação, é necessário que a resposta do canal seja estimada e compensada através de equalização.

No sistema OFDM, o canal pode ser estimado em cada uma das subportadoras. Em geral, isso é feito usando-se *preambles* ou símbolos pilotos. Esses símbolos pilotos são símbolos conhecidos pelo estimador de canal, de forma que ele possa comparar o símbolo recebido e o original, conseguindo assim determinar o efeito do canal. Nessa seção, trataremos a estimação de canal baseada nos símbolos pilotos.

Os símbolos pilotos podem ser arranjados de diversas maneiras dentro da estrutura do sinal OFDM que é transmitido. Uma das maneiras é você transmitir um símbolo OFDM inteiro de tempos em tempos, isto, é, colocar um símbolo piloto em todas as portadoras do sistema. A Figura 21 mostra como isso ocorre na frequência e no tempo. Essa estrutura é chamada de *block-type* [6].

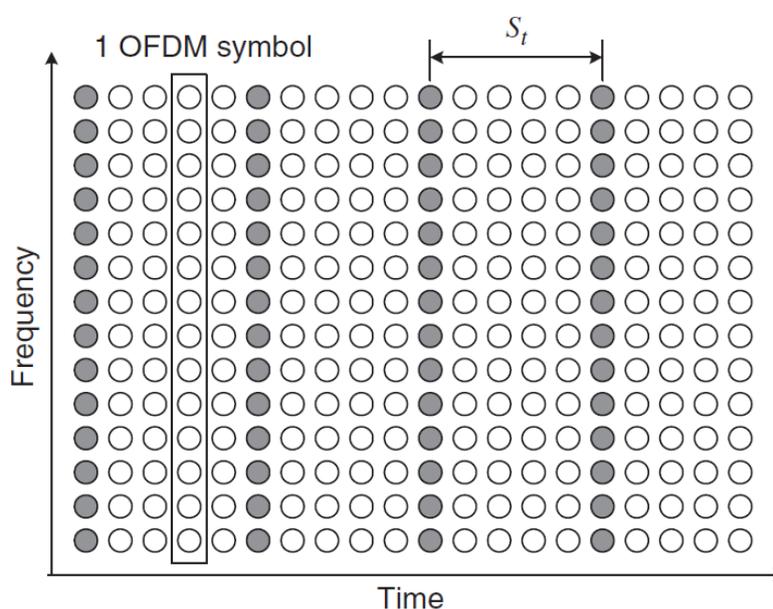


Figura 21 - Arranjo tipo Bloco [6].

Usando essa estrutura, é feita uma interpolação no domínio do tempo para estimar o canal ao longo do tempo. Para que essa estimação seja boa, o

intervalo de tempo entre o envio de pilotos deve ser menor que o tempo de coerência, ou seja:

$$S_t \leq \frac{1}{f_{Doppler}} \quad (20)$$

Essa estrutura é boa para canais seletivos em frequência, dado que os símbolos piloto são inseridos em todas as portadoras, porém, para um canal com decaimento muito rápido, ou que varia muito no tempo, o desempenho se reduz, devido a necessidade de se transmitir os símbolos pilotos com mais frequência.

Uma segunda maneira de organizar os símbolos pilotos é o tipo Comb. Nesse tipo, os símbolos pilotos são transmitidos durante todo o tempo, em subportadoras específicas, chamadas de tons piloto. A partir desses tons piloto é feito uma estimação de canal no domínio da frequência. Os símbolos são arranjados como mostrado na Figura 22 [6].

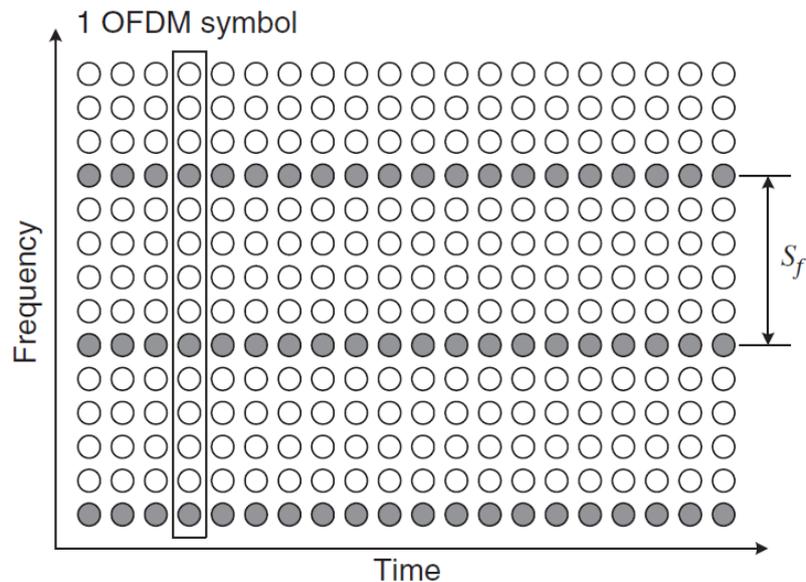


Figura 22 - Arranjo tipo Comb [6].

Para conseguir estimar o canal de forma satisfatória, a distância na frequência entre os tons pilotos deve ser menor que a largura de coerência, Sendo σ_{max} o *delay spread* máximo do canal, a seguinte condição deve ser satisfeita:

$$S_f \leq \frac{1}{\sigma_{max}} \quad (21)$$

Esse arranjo é bom para canais que variam rapidamente ao longo do tempo, mas não para canais com seletividade em frequência muito grande.

Nesse trabalho, a estimação do canal é feita no domínio da frequência, ou seja, após o processo de FFT no receptor, os símbolos pilotos são extraídos (Y_p). Seja X_p os símbolos pilotos conhecidos, inseridos no transmissor pode-se seguir os processos indicados na Figura 23.

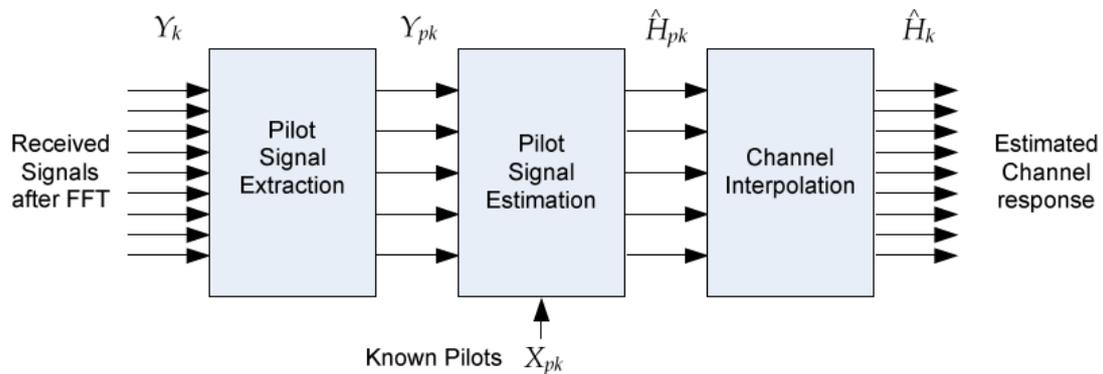


Figura 23 - Estimação de canal no domínio da frequência [9].

Como estamos trabalhando no domínio da frequência, a resposta do canal nos símbolos pilotos pode ser estimado através da equação abaixo, onde k é o índice do símbolo piloto [9]:

$$\hat{H}_p[k] = \frac{Y_p[k]}{X_p[k]} \quad (22)$$

Entre as amostras de $\hat{H}_p[k]$ existem amostras dos símbolos OFDM, como pode ser visto na Figura 24. A resposta do canal nesses pontos $\hat{H}[k]$ é obtida através da interpolação linear das pilotos vizinhas. Embora a interpolação linear seja limitada quando se fala em redução do ruído nos pontos estimados, é extremamente atrativa a sua implementação devido à sua simplicidade [10].

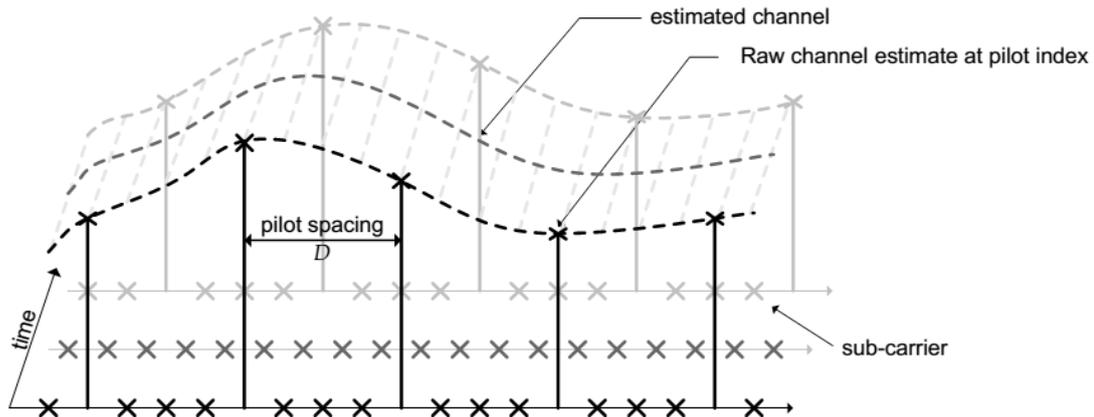


Figura 24 - Estimação de canal no domínio da frequência [9].

Uma vez feita a estimação do canal, inverte-se a sua resposta e multiplica-se pelos símbolos OFDM, de forma a compensar o efeito causado pelo canal através dessa resposta em frequência. Esse tipo de equalização é conhecido como equalização *one-tap*.

1.4. Codificação Reed-Solomon

Muitas técnicas de transmissão digital utilizam técnicas de detecção e correção de erros *Forward Error Correction* (FEC). Essa técnica consiste em adicionar informação redundante no sinal original, de forma que possa ser usada no receptor para identificar a ocorrência de erros e, eventualmente, realizar sua correção.

Existem diversos tipos de códigos corretores de erro, entre eles está o código de blocos Reed-Solomon, que discutiremos na seção seguinte.

Os códigos RS surgiram em 1960, com a publicação de um artigo por Irving Reed e Gus Solomon, que descrevia uma nova classe de códigos corretores de erro [11]. Hoje, eles são usados em diversas aplicações, entre elas CD's, comunicações sem fio, TV Digital, comunicações via satélite e modems banda larga [12].

1.4.1. Propriedades do Código Reed-Solomon

Os códigos RS são conhecidos por ter um bom compromisso entre eficiência (a proporção de informação redundante requerida) e complexidade (dificuldade em codificar e decodificar) [13].

Existem muitas categorias de códigos corretores de erros, sendo os principais os códigos de blocos e os códigos convolucionais. O código RS é um código de bloco, ou seja, a mensagem a ser transmitida é dividida em blocos de dados [13]. A cada bloco é adicionada informação de paridade para formar uma palavra de código. A codificação não altera os símbolos originais e os símbolos de paridade são adicionados como uma parte separada do bloco de mensagem. A Figura 25 mostra a organização do bloco de código e os parâmetros utilizados na sua construção, que serão discutidos a seguir.

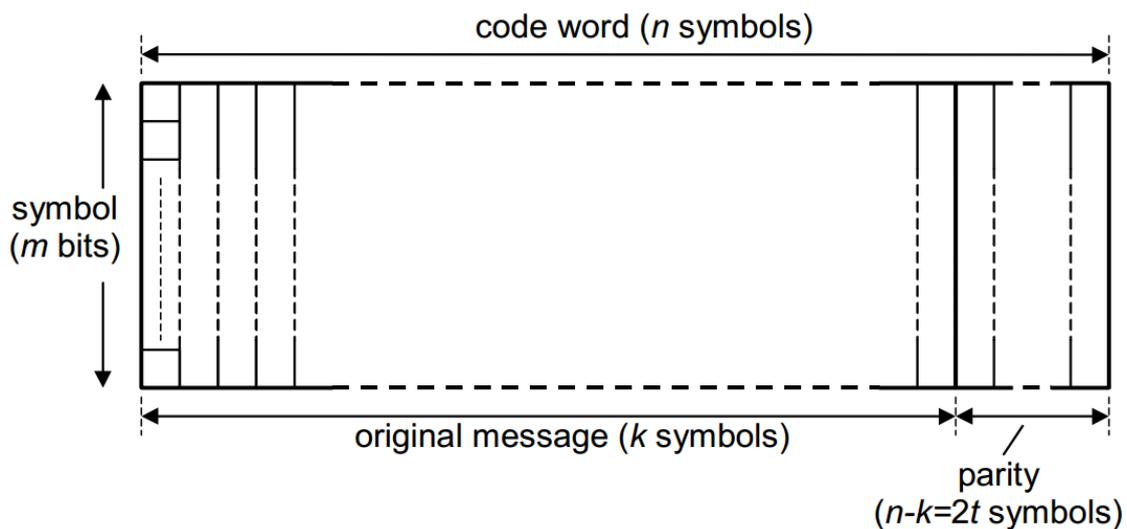


Figura 25 - Organização e parâmetros do bloco de código RS [14].

O código RS é um código linear. Isso significa que somando duas palavras de código obtém-se outra palavra de código. Também é cíclico, o que significa que ao deslocar os símbolos de uma palavra de código, obtém-se outra palavra de código. O código RS pertence à família de códigos Bose-Chaudhuri-Hocquenghem (BCH), diferenciando-se dos outros por possuir símbolos multi-bit. Isso faz com o que o código seja particularmente bom com erros de rajadas porque, embora um símbolo tenha todos os seus bits errados, conta-se apenas como um erro em termos da capacidade de correção do código [14].

O código RS pode ser descrito como um código (n, k) , onde n é o tamanho do bloco em símbolos e k é o tamanho do bloco de mensagem em símbolos. Escolhendo diferentes parâmetros para o código pode variar a complexidade da sua implementação e de proteção contra erro.

Os parâmetros devem respeitar algumas regras, como:

$$n \leq 2^m - 1 \quad (23)$$

Onde m é o número de bits em um símbolo. Quando essa condição não é igualitária, o código é conhecido como uma versão mais curta. Haverão $n - k$ símbolos de paridade. Com isso, é definida a grandeza t , que representa a capacidade de correção do código, e é dada por:

$$t = \frac{(n-k)}{2}, \text{ para } n - k \text{ par} \quad (24)$$

Ou

$$t = \frac{(n-k-1)}{2}, \text{ para } n - k \text{ ímpar} \quad (25)$$

Os códigos RS possuem uma base matemática complicada. A seguir será descrito o processo de codificação e depois o de decodificação, nos quais todas as operações aritméticas estão no campo de Galois, na qual os códigos são baseados.

1.4.2. Processo de Codificação Reed Solomon

Os valores da mensagem e dos símbolos de paridade do código RS são elementos do campo de Galois. Portanto, um código baseado em símbolos de m -bits possui 2^m elementos.

Um código RS (n,k) é construído formando um polinômio gerador $g(x)$ composto por $n - k = 2t$ fatores, do qual as raízes são elementos consecutivos do campo de Galois [14]. O polinômio gerador toma, então, a seguinte forma:

$$g(x) = (x + \alpha^b)(x + \alpha^{b+1}) \dots (x + \alpha^{b+2t-1}) \quad (26)$$

Os k símbolos de informação que formam a mensagem a ser codificada como um bloco podem ser representados por um polinômio $M(x)$ de ordem $k - 1$, de forma que:

$$M(x) = M_{k-1}x^{k-1} + \dots + M_1x + M_0 \quad (27)$$

Onde cada um dos coeficientes M_{k-1}, \dots, M_1, M_0 é um símbolo de mensagem contendo m bits, isto é, um elemento de $GF(2^m)$. M_{k-1} é o primeiro símbolo da mensagem.

Para codificar a mensagem, o polinômio de mensagem é primeiro multiplicado por x^{n-k} e o resultado é dividido pelo polinômio gerador $g(x)$, produzindo um coeficiente $q(x)$ e um resto $r(x)$, como mostrado a seguir:

$$\frac{M(x)x^{n-k}}{g(x)} = q(x) + \frac{r(x)}{g(x)} \quad (28)$$

Uma vez encontrado o resto $r(x)$ através da divisão, pode-se montar o código $T(x)$ combinando os polinômios $M(x)$ e $r(x)$ [14]:

$$T(x) = M(x)x^{n-k} + r(x) \quad (29)$$

$$T(x) = M_{k-1}x^{n-1} + \dots + M_0x^{n-k} + r_{n-k-1}x^{n-k-1} + \dots + r_0 \quad (30)$$

Que mostra que a palavra de código é produzida de forma sistemática.

Ao somar o resto, é garantido que o polinômio da mensagem codificada seja sempre divisível pelo gerador polinomial sem produzir resto [14]. Isso pode ser visto através da manipulação matemática abaixo:

$$\frac{M(x)x^{n-k}}{g(x)} g(x) = \left[q(x) + \frac{r(x)}{g(x)} \right] g(x) \quad (31)$$

$$M(x)x^{n-k} = q(x)g(x) + r(x) \quad (32)$$

Rearranjando a equação acima (lembrando que estamos trabalhando no campo de Galois), temos:

$$M(x)x^{n-k} + r(x) = q(x)g(x) \quad (33)$$

Do lado esquerdo temos a palavra de código e o lado direito possui o polinômio gerador como fator. Como o polinômio gerador é escolhido para conter um certo número de fatores, cada um desses fatores é também um fator do polinômio da mensagem codificada e irá dividir sem produzir resto. Portanto, se essa condição não for verificada no decodificador, fica claro que ocorrerão um ou mais erros no processo de decodificação.

1.4.3. Processo de Decodificação Reed-Solomon

Os erros podem ser adicionados ao polinômio de mensagem $T(x)$ utilizando um polinômio de erro, sendo o polinômio do código recebido dado por:

$$R(x) = T(x) + E(x) \quad (34)$$

Onde $E(x)$ é o polinômio de erros e segue o formato $E(x) = E_{n-1}x^{n-1} + \dots + E_1x + E_0$ e cada um dos coeficientes é um valor de erro de m bits, representados por um elemento de $GF(2^m)$ e sendo as posições dos erros ocorridos determinadas pelo grau de x daquele termo.

A decodificação RS segue os processos mostrados na Figura 26 abaixo [14]:

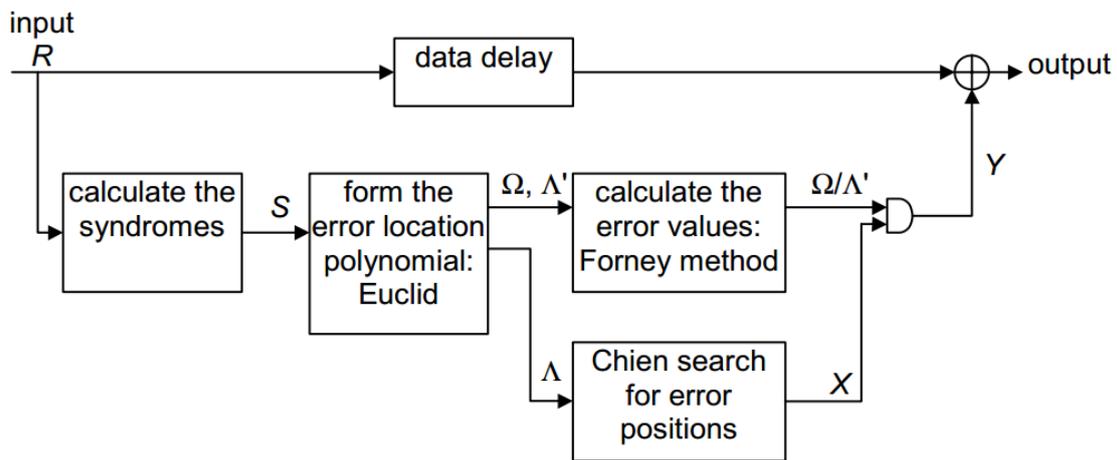


Figura 26 - Processo de decodificação RS [14].

O primeiro passo é dividir o polinômio recebido por cada um dos fatores $(x + \alpha^i)$ do polinômio gerador, produzindo um quociente e um resto. O resto é conhecido como síndrome, e pode ser calculada através de:

$$S_i = Q_i(x) \times (x + \alpha^i) + R(x) \quad (35)$$

Quando $x = \alpha^i$, essa equação é reduzida a:

$$S_i = R(\alpha^i) = R_{n-1}(\alpha^i)^{n-1} + R_{n-2}(\alpha^i)^{n-2} + \dots + R_1\alpha^i + R_0 \quad (36)$$

Onde cada um dos coeficientes são os símbolos recebidos da palavra de código. Isso significa que cada um dos valores da síndrome pode ser obtido realizando essa simples substituição no polinômio recebido [14].

Fazendo essa mesma substituição na equação do polinômio recebido, temos:

$$R(\alpha^i) = T(\alpha^i) + E(\alpha^i) \quad (37)$$

No qual $T(\alpha^i) = 0$, pois $x + \alpha^i$ é um fator de $g(x)$, que é fator de $T(x)$. Daí, tem-se que $R(\alpha^i) = E(\alpha^i) = S_i$, o que significa que a síndrome depende apenas do padrão de erros e não é afetada pelos valores dos dados [14]. Quando nenhum erro ocorre, os valores das síndromes dão zero.

Para produzir um conjunto de equações simultâneas para encontrar os erros, podemos reescrever o polinômio de erro para incluir os termos que correspondem a eles. Então assumindo que ν erros ocorreram, e que $\nu \leq t$, podemos escrever:

$$E(x) = Y_1x^{e_1} + Y_2x^{e_2} + \dots + Y_\nu x^{e_\nu} \quad (38)$$

Onde e_1, \dots, e_ν identificam as localizações dos erros na palavra de código na correspondente potência de x e os coeficientes Y representa o valor do erro naquela localização. Podemos então escrever:

$$S_i = Y_1\alpha^{ie_1} + Y_2\alpha^{ie_2} + \dots + Y_\nu\alpha^{ie_\nu} \quad (39)$$

$$S_i = Y_1X_1^i + Y_2X_2^i + \dots + Y_\nu X_\nu^i \quad (40)$$

Onde $X_1 = \alpha^{e_1}$ e assim por diante, e é conhecido como localizador de erro.

Portanto, as equações de síndrome podem ser escritas na seguinte forma matricial:

$$\begin{bmatrix} S_0 \\ S_1 \\ \vdots \\ S_{2t-1} \end{bmatrix} = \begin{bmatrix} X_1^0 & X_2^0 & \dots & X_\nu^1 \\ X_1^1 & X_2^1 & \dots & X_\nu^2 \\ \vdots & \vdots & \ddots & \vdots \\ X_1^{2t-1} & X_2^{2t-1} & \dots & X_\nu^{2t-1} \end{bmatrix} \times \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_\nu \end{bmatrix} \quad (41)$$

Uma das maneiras de escrever o polinômio localizador de erros é a seguinte:

$$\Lambda(x) = 1 + \Lambda_1x + \dots + \Lambda_{v-1}x^{v-1} + \Lambda_vx^v \quad (42)$$

$$\Lambda(x) = (1 + X_1x)(1 + X_2x) \dots (1 + X_vx) \quad (43)$$

Aplicando o método direto, que pode ser encontrado em [14], chega-se em:

$$S_{i+v} + \Lambda_1S_{i+v-1} + \dots + \Lambda_vS_i = 0 \quad (44)$$

Ainda pelo método direto, pode-se resolver essa equação para os coeficientes Λ montando-se uma matriz e calculando a sua determinante para verificar se há solução e depois inverter a matriz. Porém, existe um algoritmo muito mais eficiente que é o algoritmo de Berlekamp [14], cuja descrição pode ser vista em [15]. Trata-se de um algoritmo iterativo que parte de uma aproximação para o polinômio localizador de erros, começando com $\Lambda(x) = 1$. Depois, a cada iteração um valor de erro é formado substituindo os coeficientes aproximados nas equações correspondentes àquele valor de v . Os erros são então usados para aprimorar a aproximação de $\Lambda(x)$.

Outro método muito utilizado é o algoritmo euclidiano, cuja descrição pode ser vista em [14].

Uma vez calculados os coeficientes Λ , é possível encontrar as raízes do polinômio localizador de erros. Elas são encontradas por meio de tentativa e erro através de um algoritmo chamado de Busca de Chien [14], no qual todos os possíveis valores de raiz são substituídos na equação $\Lambda(x)$, e se ela for reduzida a zero com essa substituição, o valor de x é uma raiz e identifica a posição do erro.

Quando a localização do erro é substituída na equação da síndrome, as primeiras v equações podem ser resolvidas por inversão de matriz. Esse é conhecido como o método direto.

Uma maneira alternativa é o uso do algoritmo de Forney que encontra a solução a partir do polinômio localizador de erros e pode ser visto em [14]. Como

com os passos anteriores já determinou-se a localização dos erros, é utilizado o algoritmo de Forney para se obter o valor dos erros, permitindo assim a decodificação.

Uma vez encontrados os símbolos que contém erros, identificado X_j e encontrado os valores dos erros Y_j , os erros podem ser corrigidos adicionando o polinômio de erros $E(x)$ ao polinômio recebido $R(x)$.

1.5. Contribuições e Publicações Geradas

Publicações e submissões geradas até o presente em decorrência o desenvolvimento desse Trabalho de Graduação:

- [A] CRUZ, P. I.; CASELLA, I. R. S. Análise de um Sistema OFDM Empregando Codificação Reed-Solomon. Boletim Técnico da Faculdade de Tecnologia de São Paulo, v. 1, p. 95, 2014.
- [B] CRUZ, P. I. FERNANDES, T. M. MUNOZ, R. R. CASELLA, I. R. S. Projeto de um Codificador Reed-Solomon em FPGA Usando VHDL. Boletim Técnico da Faculdade de Tecnologia de São Paulo, v. 1, p. 123, 2014.
- [C] CRUZ, P. I. CASELLA, I. R. S. Um Analisador de Espectro Empregando a Técnica de Software Defined Radio. Boletim Técnico da Faculdade de Tecnologia de São Paulo, v. 1, p. 134, 2014.

Futuramente pretende-se submeter um artigo considerando os resultados obtidos através das medidas no sistema utilizando a USRP.

2. Técnicas Usadas no Desenvolvimento de um Modem OFDM-RS

O desenvolvimento do Modem OFDM-RS aqui descrito usará uma *plataforma de Software Definido por Rádio* (SDR), a USRP. As seções seguintes descrevem o conceito de SDR e a plataforma USRP.

2.1. Radio Definido Por Software

Um sistema SDR é um sistema de comunicação no qual alguns componentes, tradicionalmente construídos utilizando-se componentes eletrônicos, são implementados em *software*. Esses componentes incluem filtros, amplificadores, moduladores, e demoduladores. Como esses componentes são definidos por *software*, pode-se ajustar o sistema sem que seja necessário alterar o *hardware* [16].

O SDR não é um conceito novo, mas recentemente, com os avanços da eletrônica digital, os dispositivos SDR se tornaram muito práticos e baratos para a realização de prototipagem de sistemas de comunicação, pois possibilita implementação e a realização de testes de forma fácil, apenas trocando reprogramando o dispositivo. Com o crescimento das comunicações sem fio e a constante mudança nos padrões e sistemas usados para testes e prototipagens desses padrões, indústria e pesquisadores da área estão fazendo uso desse tipo de plataforma.

Um sistema SDR é composto basicamente por um hardware genérico programável, ou que se comunica com um computador, como por exemplo, um Processador Digital de Sinais (DSP). Através da programação desse hardware são definidos os componentes do sistema de comunicação a ser projetado ou testado. Em seguida, o código é embarcado no *hardware*, e esse se comporta de acordo com o que está definido no *software*. Outra modalidade existente de SDR é a em que o *hardware* se comunica com o computador durante todo o processo. Nesse caso, o *hardware* condiciona o sinal RF recebido e todo o processamento de sinais é realizado no computador *host*, ou a geração do sinal e seu processamento é realizado no computador, sendo o sinal resultante enviado para o *hardware* e esse gera o sinal RF. A comunicação entre e o

computador *host* e o *hardware* em questão é geralmente feita através de uma conexão *Gigabit Ethernet* ou através da placa de áudio do computador *host*, ou de uma porta USB. Existem *hardwares* que são embarcados e também recebem e geram sinal RF, sendo tanto a recepção e transmissão, quanto o processamento do sinal, realizado no próprio hardware.

2.2. USRP

A *Universal Software Radio Peripheral (USRP)* é uma plataforma de SDR que possui tanto modelos baseados em computadores *hosts*, como a NI USRP-2920, que será utilizada nesse trabalho [17], quanto modelos embarcáveis. A USRP NI-2920 se conecta ao computador *host* através de conexões de alta velocidade, como a *Gigabit Ethernet*, ou USB.

A USRP é responsável por receber ou enviar ao computador *host* um sinal em frequência intermediária e fazer a *up-conversion* ou a *down-conversion*, respectivamente, desses sinais. Para que isso seja feito corretamente, o sinal passa por diversas etapas de processamento antes que o sinal RF seja gerado [18]. A Figura 27 mostra o diagrama de blocos básico da USRP.

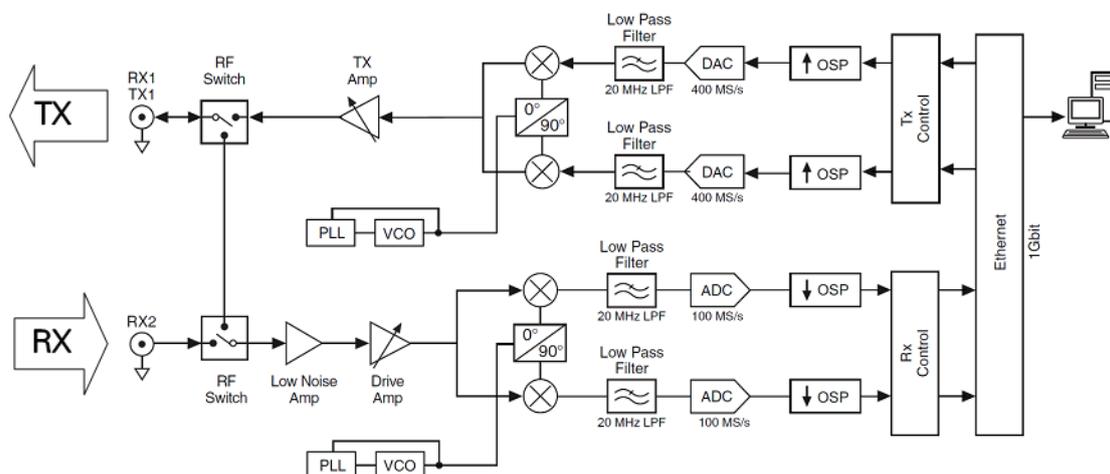


Figura 27 - Diagrama de blocos da USRP [17].

Os sinais RF recebidos passam por um estágio de ganho ajustável (0-30dB), e são convertidos nos sinais banda base I e Q através de um processo de *down-conversion*. Em seguida, os sinais I e Q são amostrados por um conversor analógico-digital (ADC) de 2 canais, 100MS/s, e 14-bit.

Os dados digitais I e Q percorrem um *Onboard Signal Processor* (OSP) que aplica um sinal de correção de offset com o algoritmo CORDIC [19] e realiza uma decimação do sinal digital. As amostras convertidas são então passadas ao computador *host* a uma taxa IQ de até 20MS/s através da conexão *Gigabit Ethernet* com o computador *host*.

Para transmissão, o computador *host* sintetiza o sinal IQ banda-base, e transfere as suas amostras para a USRP a uma taxa de 20MS/s através da conexão *Gigabit Ethernet*. O OSP da USRP realiza uma sobreamostragem dos sinais I e Q a uma taxa de 400MS/s e o sinal é então convertido para analógico através de um conversor analógico-digital (DAC). Em seguida, é realizado o processo de *up-conversion* do sinal para a faixa de RF. O algoritmo CORDIC é, então, usado para corrigir a frequência do sinal para garantir a frequência central especificada. Um estágio de ganho ajustável (0-31dB) amplifica o sinal de RF resultante para a transmissão [17].

3. Simulação de um Modem OFDM-RS

A análise do modem OFDM-RS se dará através de uma simulação do modelo proposto, comparando seu desempenho com os valores teóricos sem codificação RS e com codificação RS.

Em seguida, com o uso de uma plataforma USRP, o mesmo modelo será implementado e testado, comparando os resultados com os da simulação.

3.1. Descrição do Modelo

O modelo aqui proposto consiste em um sistema OFDM tradicional, utilizando CP e *Interleaving*. O modelo utiliza uma modulação 16-QAM. No transmissor, antes da modulação, foi adicionado um codificador RS, e no receptor, um decodificador foi adicionado após a demodulação. O número de subportadoras utilizado é de 128, e a equalização no receptor é a *One-Tap*. O modelo pode ser resumido no diagrama de blocos da Figura 28.

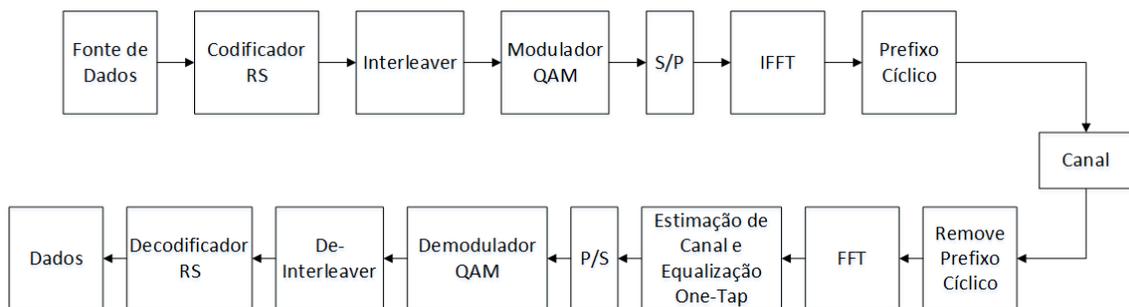


Figura 28 - Diagrama de blocos do modelo usado na simulação.

Utilizando o Matlab, foi desenvolvido um script para a simulação do modelo proposto. Primeiramente foi executada uma simulação sem a codificação RS, analisando o desempenho de acordo com a BER (*Bit Error Rate*)

O modelo de canal utilizado foi o seletivo em frequência com desvanecimento do tipo Rayleigh e ruído do tipo AWGN. Para simular a ocorrência de ISI entre os símbolos OFDM, é montado um bloco de transmissão, transmitindo um certo número de símbolos OFDM a cada realização. Essa quantidade de símbolos OFDM é então colocada em série (alocada em um vetor linha) e em seguida passa-se essa sequência pelo filtro FIR que representa o canal, com 5 coeficientes que mudam a cada transmissão de acordo com uma

distribuição de Rayleigh, seguindo um perfil de potência com fator de decaimento exponencial igual a 2. Não são usados símbolos pilotos. A equalização é feita a partir do conhecimento dos símbolos que foram transmitidos, realizando-se assim uma equalização de forma a obter o melhor desempenho possível. A Tabela 1 mostra um resumo dos parâmetros do sistema utilizado na simulação.

Tabela 1 - Resumo dos parâmetros usados na simulação.

Modulação	16-QAM
Número de Portadoras	128
Tamanho do Prefixo Cíclico	32 símbolos
Símbolos Pilotos	Nenhum
Código Reed Solomon	RS(10,8), RS(10,6) e RS(10,4)
Número de bits por símbolo Reed-Solomon	4 bits
Modelo de Canal	Seletivo em frequência com desvanecimento Rayleigh
Número de Coeficientes do Canal	5
Fator de Decaimento Exponencial de Rayleigh	2
Tipo de Ruído	Aditivo Gaussiano

3.2. Análise de Desempenho

A primeira simulação realizada foi a do sistema sem codificação RS. O resultado obtido pode ser visto no gráfico da Figura 29.

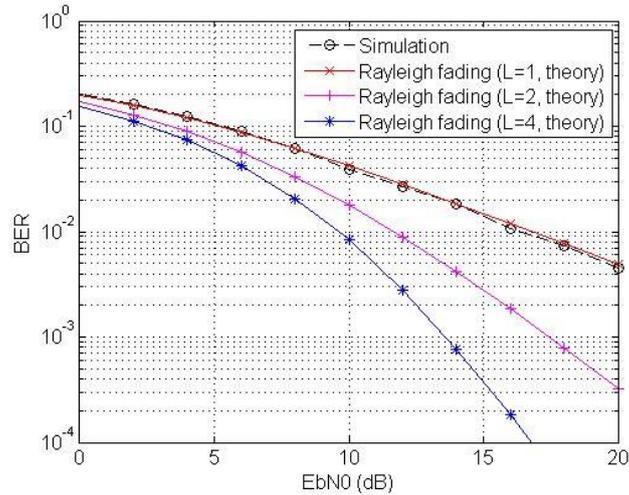


Figura 29 - BER em função da SNR do canal.

Como pode ser observado, O sistema OFDM simulado apresentou um desempenho semelhante ao do esperado para um canal de desvanecimento Rayleigh com diversidade 1. Vale ressaltar que nesse caso, a ISI entre os símbolos OFDM foi praticamente eliminada devido ao tamanho do prefixo cíclico utilizado (32 símbolos de CP). A ideia de utilizar uma codificação é diminuir a taxa de erro para uma mesma SNR, ou seja, apresentar uma curva com uma queda maior que a dessa simulação.

Na Figura 30, podemos ver o resultado de uma simulação para um código RS(10,8).

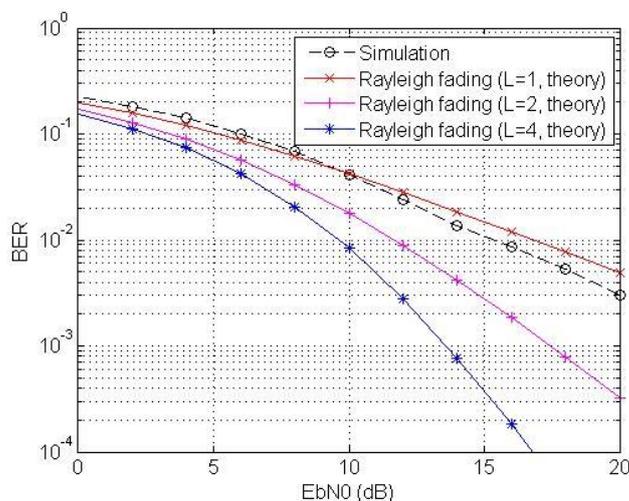


Figura 30 - Simulação de um sistema OFDM com codificação Reed-Solomon.

Pode ser observado uma melhora no desempenho do sistema para SNRs mais altas (a partir de 10dB), embora para SNRs mais baixas esteja ocorrendo uma pequena piora.

Na Figura 31, pode-se ver o resultado da simulação onde o bloco de Interleaving e a codificação RS estão ligados.

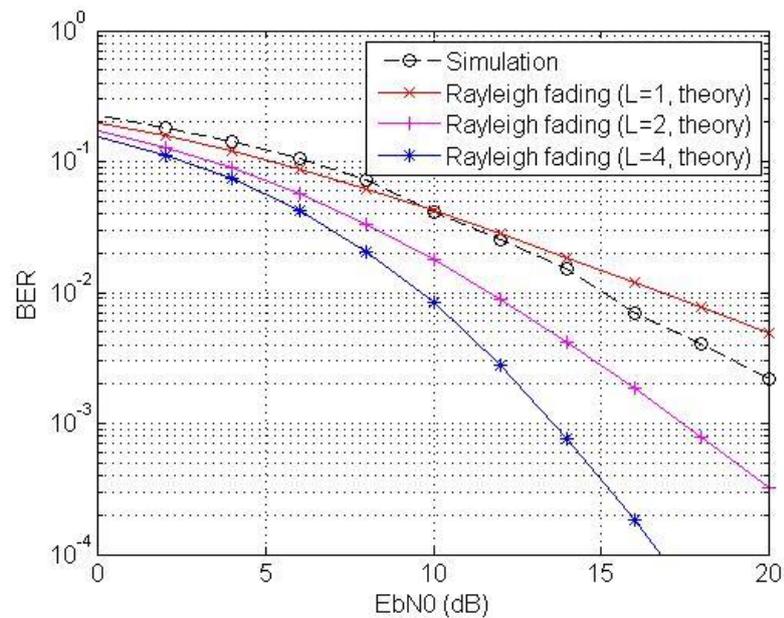


Figura 31 - Simulação de um sistema OFDM com Interleaving e RS.

Pode-se observar uma melhora ainda maior no desempenho para SNRs muito altas, embora a diferença em relação à simulação anterior não seja muito grande em SNRs médias e baixas.

A Figura 32 abaixo mostra uma comparação entre diferentes tipos de código RS, cuja principal diferença é sua capacidade de correção. Na figura também pode ser vista a curva do desempenho de um sistema OFDM sem codificação.

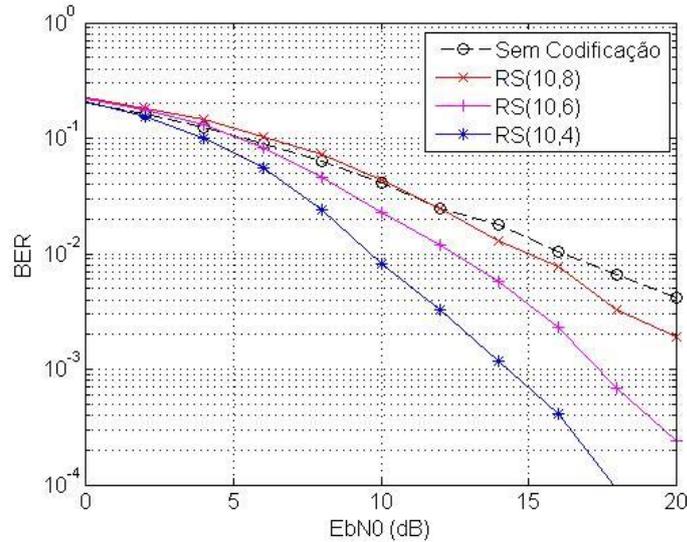


Figura 32 - Comparação entre diferentes códigos RS com diferentes capacidades de correção em um sistema OFDM e um sistema sem codificação.

Pode-se observar claramente que, com o aumento da capacidade de correção do código RS, melhor é o desempenho do sistema. Por exemplo, para E_b/N_0 de aproximadamente 10dB podemos ver que o código RS(10,8) (capacidade de correção de um símbolo RS) possui um desempenho semelhante ao do sistema sem codificação, que corresponde a uma BER de 0,7. Já o código RS(10,6), que possui uma capacidade de correção maior (2 símbolos RS) e o código RS(10, 4) (corrige até 3 símbolos RS), possuem respectivamente uma BER de 0,9 e 0,03, para E_b/N_0 de 10 dB. Conforme E_b/N_0 aumenta, os valores de BER ficam cada vez menores, sendo a que para os códigos RS com maior capacidade de correção, a BER cai mais rapidamente. No entanto, esse ganho obtido através de um código com maior capacidade de correção se dá através de um aumento na complexidade do sistema. Um fenômeno interessante, que pode ser observado também na Figura 30, é que para baixos valores de E_b/N_0 , existe uma performance mais baixa dos sistemas com código RS em relação aos sistemas sem código. Isso ocorre porque os códigos RS possuem uma capacidade de correção de erros fixada. A medida que mais erros vão acontecendo, devido a uma redução da relação sinal-ruído baixa, o decodificador só permanece corrigindo um mesmo número de erros. Isso estabelece um ponto em que a performance do sistema com código é igual à performance de um sistema sem código, que é o ponto em que suas curvas de erro de bits se cruzam. Valores de E_b/N_0 abaixo do valor no ponto de

cruzamento é onde o sistema com código produz resultados piores que o sistema sem código. Essa degradação da performance pode ser interpretada como um gasto de energia causado pelos bits redundantes, sem que esses tragam algum real benefício ao sistema [20].

4. Implementação do Modem OFDM-RS Utilizando a USRP

4.1. Descrição do Sistema Implementado

O sistema OFDM com codificação RS foi implementado de acordo com o diagrama de blocos mostrado na Figura 33.

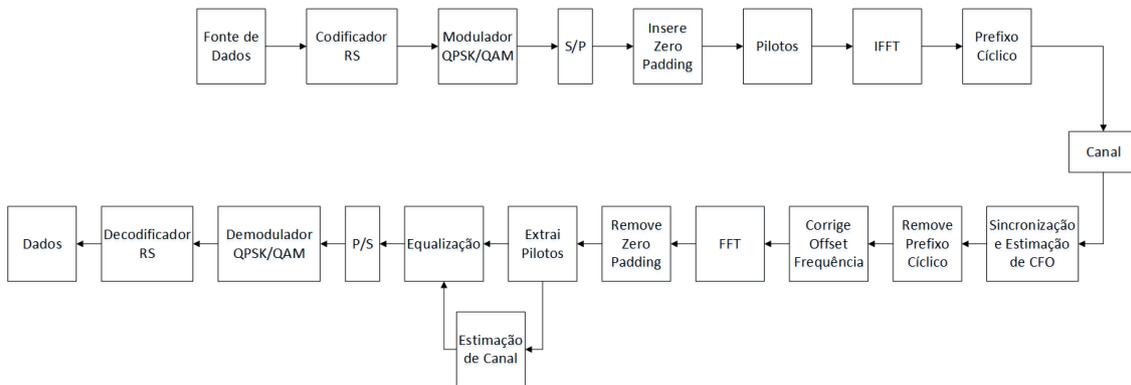


Figura 33 - Diagrama de blocos que representa o Modem OFDM implementado.

Como pode ser observado, toda a parte relacionada a processamento de sinais, como a codificação e decodificação RS, modulação e demodulação QPSK ou QAM, algoritmos de IFFT e FFT, adição de CP e demais, são realizadas via *software*, portanto seus parâmetros podem ser alterados a qualquer momento. A USRP fica responsável por receber o sinal resultante desse processamento e transformá-lo em um sinal de rádio, recebendo-o em seguida. A própria USRP trata esse sinal como descrito anteriormente e passa para o computador, onde todo o processamento de sinal do receptor é realizado via *software*.

Foi implementado um sistema OFDM de 256 portadoras, sendo 25 delas contendo símbolos pilotos e 106 símbolos ZP, resultando 125 portadoras para os símbolos de informação. Os símbolos ZP são adicionados da seguinte forma: 53 zeros antes do início dos símbolos de informação, um no centro, e 52 no final dos símbolos de informação. As portadoras pilotos são adicionadas da seguinte forma: o primeiro símbolo é um piloto; a cada cinco símbolos de informação insere um símbolo piloto. O sistema RS foi implementado usando-se 5 bits de tamanho de símbolo do bloco RS (m). Foram testados sistemas com códigos RS(25,21), RS(25,19) e RS(25,17), que são sistemas com capacidade de

correção de 2, 4 e 6 símbolos respectivamente. A Tabela 2 mostra um resumo dos parâmetros utilizados na implementação do sistema.

Tabela 2 - Resumo dos parâmetros usados na implementação do sistema OFDM.

Modulação	QPSK e 16-QAM
Frequência Central	910 MHz
Taxa IQ	1MS/s
Número de Portadoras	256
Tamanho do Prefixo Cíclico	64 símbolos
Símbolos Pilotos	25 símbolos
Zero Padding	106 símbolos
Código Reed Solomon	RS(25,21), RS(25,19) e RS(25,17)
Número de bits por símbolo Reed-Solomon	5 bits
Equalização	One-Tap através de estimação linear de canal.

4.2. Blocos Funcionais

Os principais blocos funcionais mostrados na Figura 33 já foram descritos teoricamente nas seções do capítulo 1 desse texto. Nessa seção será dada uma ênfase maior nos blocos funcionais mais críticos sob o ponto de vista de implementação no LabVIEW, levando em consideração funções especiais conhecidas como *Virtual Instruments* ou VIs. Abaixo segue a explicação desses blocos funcionais.

4.2.1. Fonte de Dados

A nossa fonte de dados na implementação é dada através do *VI MT Generate Bits*. Essa função gera uma sequência de bits pseudoaleatória de Galois de acordo com certos parâmetros fornecidos pelo programador. Nessa implementação, foi fornecido a ordem da sequência de bits.

Sequências pseudoaleatórias, apesar de sua natureza determinística, satisfazem diversas propriedades de sequências aleatórias, como por exemplo auto correlação e correlação cruzada.

A Figura 34 mostra essa função como é vista no LabVIEW:

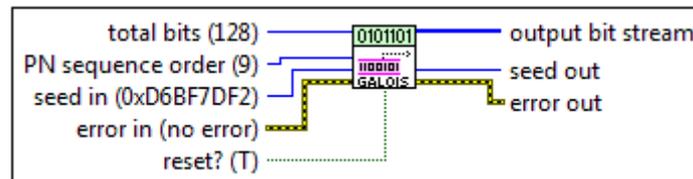


Figura 34 - VI MT Generate Bits.

4.2.2. Cálculo da BER

O cálculo da BER no receptor é feito através da função *MT Calculate BER After Trigger*, que pode ser vista na Figura 35.

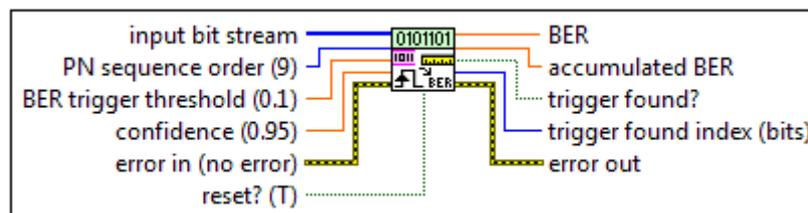


Figura 35 - MT Calculate BER After Trigger.

Ela recebe como entrada a sequência de bits identificada pelo receptor, a ordem da sequência, que deve ser a mesma usada na no *VI MT Generate Bits*, e valores de limiar do trigger e de confiança. Este VI procura por um trigger baseado nesses dois últimos parâmetros tentando sincronizar a sequência de bits na entrada. O VI apenas calcula a BER após realizada essa sincronia e o trigger ser encontrado.

4.2.3. Outras funções

Algumas outras funções que já tiveram seus conceitos explicados aqui, merecem ainda uma atenção em relação à sua utilização no programa de uma forma geral.

Uma delas é o VI *Data Reference Interleave*, que pode ser vista na Figura 36. No transmissor é adicionado um bloco que realiza a adição de símbolos

pilotos no modo *Comb*, ou seja, são criados tons pilotos para que no receptor seja feita a estimação do canal no domínio da frequência.



Figura 36 - VI Data Reference Interleave.

Outra dessas funções é a de sincronização através do Algoritmo de Van de Beek, que se encontra no receptor e pode ser vista na Figura 37. Essa função retorna os índices em que ocorrem os valores máximos da função $\Lambda(\theta, \varepsilon)$, que é onde ocorre o início do símbolo OFDM com o CP. Ela também retorna o Offset em frequência estimado.

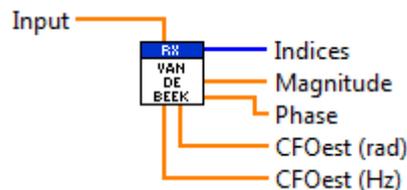


Figura 37 - Sincronização através do algoritmo de Van de Beek.

Os valores do índice seguem para a função *Remove CP* que separa os símbolos OFDM e retira o prefixo cíclico de cada um deles. Após a remoção do CP, o offset em frequência é corrigido através de um VI chamado *CFO Correction*. Após a retirada dos símbolos pilotos no VI *Zero Pad Removal and Deinterleave*, a estimação de canal é feita através da função *Channel Estimation Linear*, que pode ser vista na Figura 38. Essa função realiza a estimação do canal através de interpolação linear.



Figura 38 - VI Channel Estimation Linear.

Em seguida, a resposta do canal estimada é multiplicada pelos símbolos restantes, compensando o efeito do canal nesses símbolos.

A função de decodificação RS do LabVIEW utiliza os algoritmos de Berlekamp, Forney e busca de Chien para realizar a decodificação.

4.2.4. Configuração da USRP

A USRP possui parâmetros que devem ser configurados para que a transmissão e a recepção possam ser realizadas. Para isso, o driver da USRP para o LabVIEW possui uma série de Vis que realizam essas configurações. Esses VIs podem ser vistos na Figura 39, divididos entre os utilizados para transmissão e para recepção.



Figura 39 - Conjunto de VIs usados para realizar a configuração da USRP através do LabVIEW.

O *niUSRP Open Rx Session* e o *niUSRP Open Tx Session* cria uma seção para recepção e transmissão de um sinal, respectivamente. Deve-se fornecer para esses Vis o endereço IP da USRP que está sendo utilizada.

O *niUSRP Configure Signal* configura os principais parâmetros de transmissão e recepção da USRP. Esses parâmetros estão explicados na Tabela 1. Não são aceitos todos os valores para esses parâmetros, por isso, esse VI fornece os valores aproximados, caso seja fornecido um valor não aceito.

Tabela 3 - Parâmetros do VI niUSRP Configure Signal.

Parâmetro	Descrição
IQ Rate	Ajusta a taxa de transmissão ou recepção das amostras IQ. Reflete diretamente na largura de banda.
Carrier Frequency	Frequência central a ser utilizada na transmissão.
Gain	Define o ganho utilizado no estágio de ganho da USRP antes da transmissão ou após a recepção.
Active Antenna	Define qual porta será utilizada para transmitir ou receber.

Após a configuração de uma seção de Rx, usa-se o VI *niUSRP Initiate* para iniciar a aquisição de dados IQ. O VI *niUSRP Fetch Rx Data* retorna as amostras IQ obtidas em uma seção RX para que sejam trabalhadas, enquanto o VI *niUSRP Write Tx Data* permite que você envie amostras IQ à USRP.

O VI *niUSRP Abort* para uma aquisição de sinal RF que esteja em andamento, e o VI *niUSRP Close Session* encerra tanto uma seção Rx como uma seção Tx.

Mais informações sobre a utilização desses VIs podem ser obtidas em [21].

4.2.5. Transmissor e Receptor OFDM-RS

A Figura 40 abaixo mostra a interface geral do transmissor enquanto a Figura 41 mostra a interface geral do receptor.

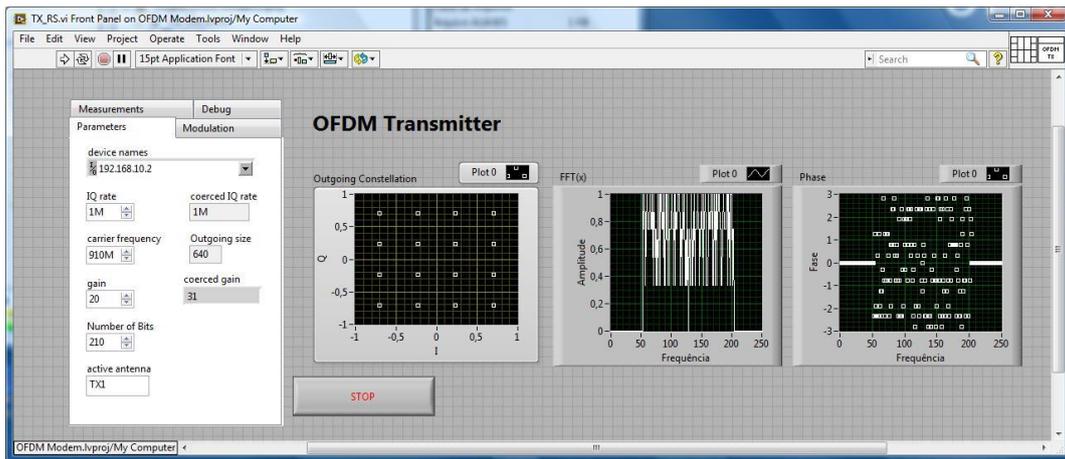


Figura 40 - Interface do transmissor OFDM.

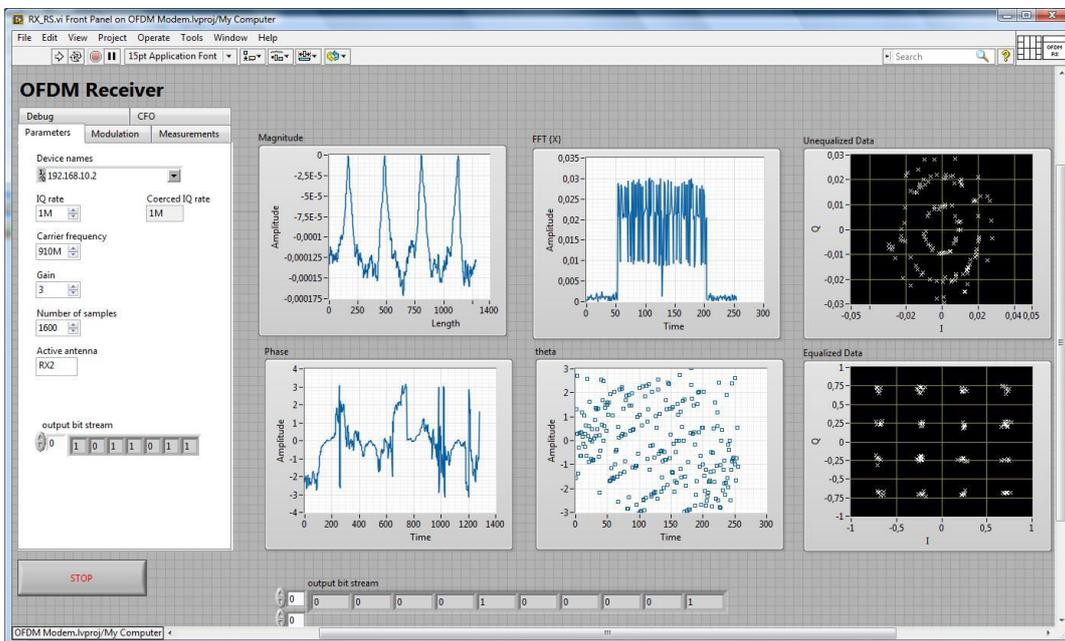


Figura 41 - Interface do receptor OFDM.

Nas figuras acima está exibida a aba de configuração da USRP, onde são configurados os parâmetros que discutimos na seção 3.2.4 desse texto. A Figura 42 mostra a aba onde são configuradas a modulação e o código RS, tanto no transmissor quanto no receptor.

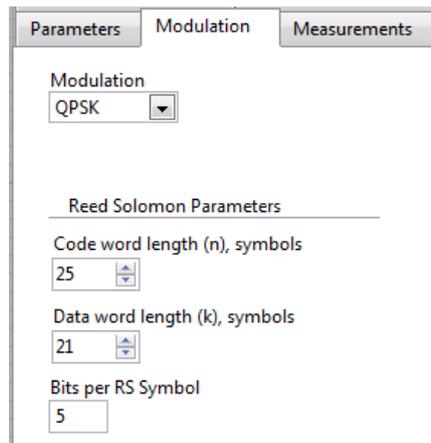


Figura 42 - Aba de configuração do tipo de modulação e código RS a serem utilizados.

No receptor há ainda a aba onde são exibidas as medições como a BER, e pode ser vista na Figura 43.

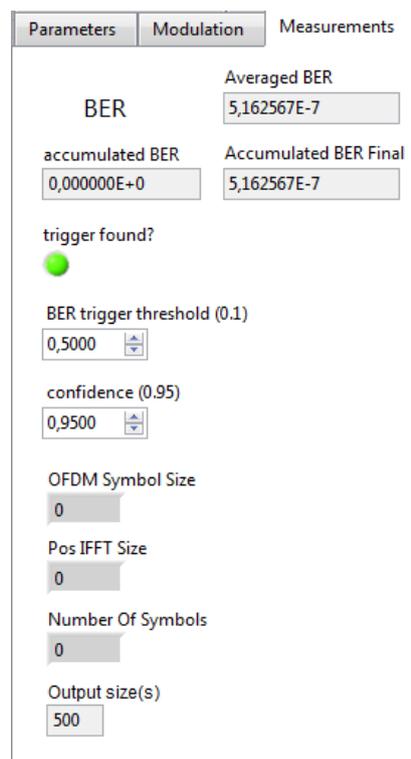


Figura 43 - Aba de medições encontrada no receptor.

4.3. Análise de Desempenho do Modem OFDM-RS

A transmissão e a recepção ocorrem na mesma USRP, através de um link sem-fio. As antenas de transmissão e recepção estão separadas por cerca de 2 cm, nesse primeiro momento. A portadora central foi ajustada na USRP para 910

MHz e a taxa de transmissão de símbolos foi de 1MS/s. A Figura 44 mostra uma foto do sistema, mostrando que as antenas estão posicionadas de modo a reduzir a intensidade do sinal recebido.



Figura 44 - Configuração usada nos testes.

O primeiro teste realizado utilizou um mapeamento QPSK em cada subportadora do sistema OFDM-RS. Nos testes realizados, variou-se o ganho no transmissor e mediu-se a potência recebida no receptor e a BER obtida.

Em relação a potência, levantou-se a curva que mostra a relação entre o ganho do transmissor e a potência do sinal recebida no receptor. Mediu-se a potência recebida no receptor quando nenhum sinal estava sendo transmitido, sendo essa então a potência do ruído. O valor obtido foi $P_N = 2,309 \times 10^{-22}W$. Como a potência recebida no transmissor é a potência do sinal (P_S) mais a potência do ruído, bastou subtrai-la da potência recebida para obter a potência do sinal, chegando assim na curva que pode ser vista na Figura 45.

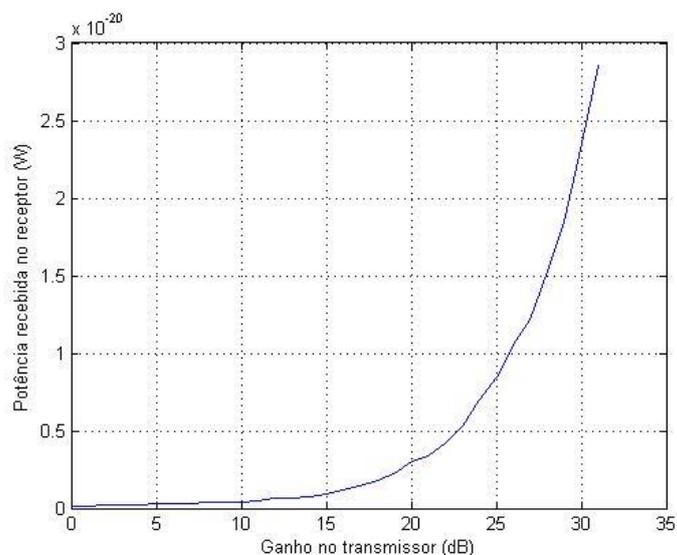


Figura 45 - Relação entre ganho ajustado no transmissor e a potência recebida no receptor.

Uma vez com esses dados em mão, foi possível calcular a SNR e E_b/N_0 para o sistema OFDM empregando mapeamento QPSK através da seguinte relação, cuja dedução está mostrada no anexo A:

$$\frac{E_b}{N_0} = \frac{P_S}{P_N} \frac{1}{2} \frac{1320}{150} f_c \quad (45)$$

Nota-se que no cálculo é utilizado um fator de correção. Esse fator é usado devido ao fato de que a resposta ao impulso dos filtros de entrada e saída da USRP não são conhecidos, e portanto podem haver diferenças entre o valor da potência do sinal incidente no receptor e o valor de potência de fato medido. O uso desse fator de correção, faz com que os valores de BER se aproximem bastante dos valores teóricos em todos os valores de E_b/N_0 medidos, e o valor obtido foi de $f_c = 7/2$.

Realizados esses cálculos, e tendo medido os valores de BER para cada valor de ganho, para um sistema sem codificação RS e para sistemas com codificações RS(25,21), RS(25,19) e RS(25,17), obteve-se as curvas de BER mostradas na Figura 46.

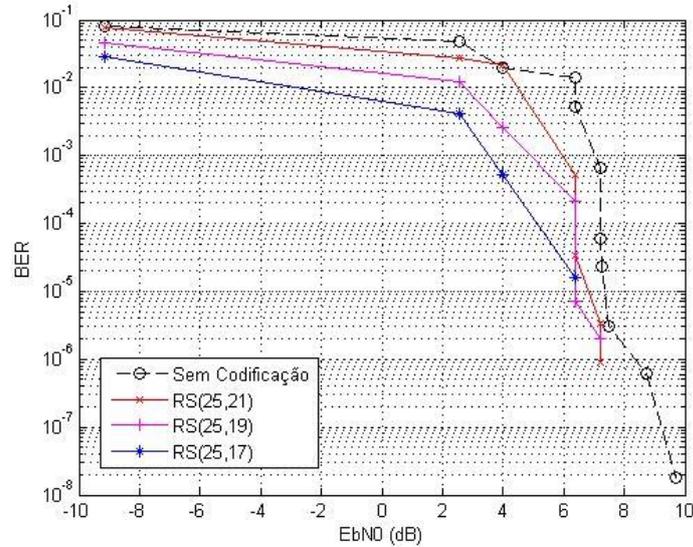


Figura 46 -Comparação da BER em um sistema OFDM utilizando mapeamento QPSK para diferentes tamanhos de código RS.

Pode ser observado no gráfico da Figura 46 que o uso do código RS melhora o desempenho do sistema OFDM. Para um E_b/N_0 de cerca de 2,54 dB, tem-se uma BER de 0,04758 para um sistema sem codificação. Com código RS(25,21) tem-se uma BER de 0,0273, enquanto com código RS (25,19) resulta em uma BER de 0,01231, e um código RS(25,17) resulta em uma BER de 4,212e-03. Para um valor de E_b/N_0 de aproximadamente 6,382 dB, tem-se uma BER de 0,01399 para o sistema sem codificação, 0,514e-03 para um sistema com codificação RS(25,21), 0,218e-03 para um sistema com codificação RS(25,19) e 1,60e-05 para um sistema com codificação RS(25,17). Com base nesses valores, pode-se concluir que quanto maior a capacidade de correção do código, mais robusto fica o sistema, e essa melhora fica mais evidente para valores mais altos de E_b/N_0 . Vale lembrar que para que o código comece a demonstrar uma melhora, o valor da SNR deve passar por um limiar, devido a capacidade de correção fixada.

Foi realizado o mesmo procedimento para mapeamento 16-QAM, porém, a relação de SNR agora é:

$$\frac{E_b}{N_0} = \frac{P_S}{P_N} \frac{1}{4} \frac{1320}{150} f_c \quad (46)$$

Na Figura 47 podemos ver as curvas de BER resultantes.

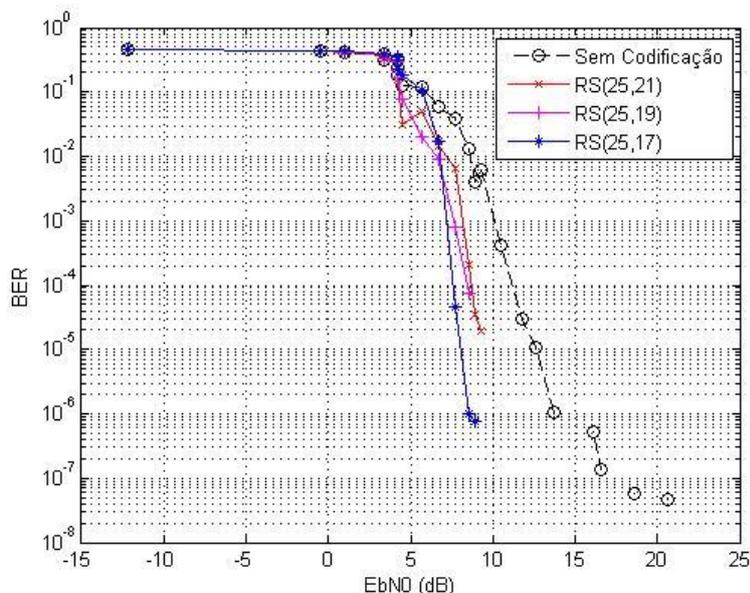


Figura 47 - Comparação da BER em um sistema OFDM utilizando mapeamento 16-QAM para diferentes tamanhos de código RS.

Pode-se observar, da mesma forma que no sistema com mapeamento QPSK, houve uma melhora no sistema com o uso dos códigos, e quanto maior a capacidade de correção do código, maior a robustez do sistema. Para um $Eb/N0$ de aproximadamente 5,736 dB, tem-se uma BER de 0,1179 no sistema sem codificação, enquanto para o sistema com codificação RS(25,21), tem-se uma BER de 0,05136 e o sistema com RS(25,19), apresentou uma BER de 0,01994. Já o sistema com RS(21,17) apresentou uma BER de 0,2047, um valor não muito abaixo da BER obtida pelo sistema sem codificação. Isso ocorre pois o sistema com código RS(25,17) ainda não ultrapassou o limiar onde começa a trazer real benefício ao sistema. Nesse ponto, apenas está gastando energia inserindo redundância no sinal.

Esses valores de BER são altos e indicam quem ainda está ocorrendo muitos erros. Para o sistema RS(25,19), que teve um menor valor de BER, aproximadamente 2% dos bits estão apresentando erros.

Para um valor de $Eb/N0$ mais elevado, de aproximadamente 8,589 dB, o sistema sem codificação apresentou uma taxa de erro de 0,01291, enquanto o sistema com codificação RS(25,21) apresentou uma taxa de erro de 0,000207.

Já os sistemas com codificação RS(25,19) e RS(25,17) apresentaram taxas de erro, respectivamente, de $7,388e-05$ e $9,923e-07$.

Novamente, essa melhora no desempenho do sistema utilizando códigos com maior capacidade de correção de erros, se dá as custas de um aumento da complexidade do transmissor e do receptor.

5. Conclusão e Futuros Trabalhos

Este trabalho apresenta o detalhamento teórico básico para o entendimento do modelo de sistema de comunicação proposto (OFDM com codificação RS). As simulações realizadas mostraram que o modelo OFDM opera como o esperado quando utilizado sem codificação com a codificação RS. Pode-se notar também a melhora no desempenho do sistema OFDM quando utilizando um código RS mais robusto, isto é, com uma capacidade de correção de erros maior.

O sistema implementado na USRP permite observar a melhor do desempenho do sistema em função do aumento da potência do transmissor, permitindo selecionar entre um mapeamento 16-QAM e um mapeamento QPSK. Foi adicionado também opções que permitem a escolha do tamanho do código RS.

O sistema é capaz de calcular a BER no receptor. Um programa foi desenvolvido para medir a potência do sinal incidente, de forma a se obter a relação E_b/N_0 . Dessa forma, pôde-se construir uma curva de BER em função de E_b/N_0 , permitindo uma comparação mais adequada das diferentes possibilidades e medir de forma melhor o desempenho do sistema. Com esses dados, ficou claro que o sistema utilizando codificação RS apresentou uma performance superior ao sistema que não utiliza nenhuma codificação.

Através dos resultados aqui obtidos, pode-se aferir que a utilização de códigos Reed-Solomon no sistema OFDM traz um benefício considerável, mesmo quando se leva em conta o aumento da complexidade do sistema oriundo da sua utilização.

A implementação aqui realizada poderá ser usada futuramente para realização de outros testes, como o uso de diferentes tipos de código ou modulação, desde que o sistema a ser testado seja OFDM. Também poderá ser usado na implementação de um modem comercial.

6. Referências Bibliográficas

1. GALLAGER, R. **Principles of Digital Communication**. Cambridge: Cambridge University Press, 2008.
2. RAPPAPORT, T. S. **Comunicações Sem Fio: Princípios e Práticas**. 2nd. ed. São Paulo: Pearson, 2009.
3. NETO, A. D. C. TV Digital I: Entrelaçadores Temporais para Sistemas de Transmissão. **Teleco Inteligência em Telecomunicações**, 2011. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialtvdentr1/pagina_3.asp>. Acesso em: 25 Março 2014.
4. GUIMARÃES, D. A. Introdução às Comunicações Móveis. **Wireless Brasil**, 1998. Disponível em: <<http://www.wirelessbrasil.org/wirelessbr/colaboradores/dayani/intro02.html>>. Acesso em: 7 Abril 2014.
5. MACHADO, R. Tópicos Avançados em Telecomunicações. Disponível em: <<http://coral.ufsm.br/gpscom/professores/Renato%20Machado/TopicosAvancados/TopAvanTelecom03.pdf>>. Acesso em: 20 Março 2014.
6. CHO, Y. S. et al. **MIMO-OFDM Wireless Communication With MATLAB**. Singapore: Wiley, 2010.
7. LATHI, B. P. **Modern digital and analog communication systems**. 4th. ed. New York: Oxford University Press, 2009.
8. VAN DE BEEK, J. J. . S. M. . B. P. O. On Synchronization in OFDM Systems Using the Cyclic Prefix.
9. FIRST R. HAJIZADEH, S. K. M. A. T. M. R. T. Channel Estimation in OFDM System Based on the Linear Interpolation, FFT and Decision Feedback. **18th Telecommunications forum TELFOR 2010**, Belgrade, 2010.
10. AKRAM, M. S. **Pilot-based Channel Estimation in OFDM Systems**. Copenhagen. 2007.
11. REED, I.; SOLOMON, G. Polynomial codes over certain finite fields. **SIAM Journal of Applied Math**, v. 8, 1960.
12. CORRÊA, F. **Implementação em Software da Codificação de Canal em ADSL: Um Estudo de Caso Sobre os Efeitos do Ruído Impulsivo na Transmissão de Vídeo**. Universidade Federal do Pará. [S.l.]. 2011.
13. LIN, S.; COSTELLO, D. J. **Error Control Coding**. 2nd. ed. London: Pearson: Prentice Hall, 2004.

14. CLARKE, C. K. P. **Reed-Solomon Error Correction**. [S.l.]. 2002.
15. KERL, J. The Berlekamp algorithm, 2009. Disponível em:
<<http://www.johnkerl.org/doc/iw2009/berlekamp.pdf>>. Acesso em: 13 dez. 2014.
16. LIMA, A. G. M. Rádio Definido por Software: O Próximo Salto no Mundo das Telecomunicações e Computação. **Revista Digital: Comunicações Digitais & Tópicos Relacionados**, 2014. Disponível em:
<http://revdigonline.com/artigos_download/art_13.pdf>. Acesso em: 07 Abril 2014.
17. NATIONAL INSTRUMENTS. Global Synchronization and Clock Disciplining with NI USRP-293x Software Defined Radio. **National Instruments**, 2014. Disponível em:
<<http://www.ni.com/tutorial/14705/en/>>. Acesso em: 15 Março 2014.
18. HANZA, F. A. The USRP. **Scribd**, 2008. Disponível em:
<<http://pt.scribd.com/doc/9688095/USRP-Documentation>>. Acesso em: 31 Março 2014.
19. ANDRAKA, R. The CORDIC Algorithm. **Andraka Consulting Group, Inc.** Disponível em:
<<http://www.andraka.com/files/crdcsrvy.pdf>>. Acesso em: 13 dez. 2014.
20. SKLAR, B. **Digital Communications: Fundamentals and Applications**. 2. ed. Upper Saddle River: Prentice Hall, 2001.
21. NATIONAL INSTRUMENTS. A Hands-on Introduction to Software Defined Radio with the NI USRP and NI LabVIEW. **National Instruments**, 2014. Disponível em:
<<http://www.ni.com/white-paper/14518/en/>>. Acesso em: 07 Dezembro 2014.
22. VAN DE BEEK, J. J. . S. M. . I. M. . A. B. P. O. Low-complex frame synchronization in OFDM systems. **IEEE ICUPC**, p. 982–986, 1995.
23. VAN DE BEEK, J. J. . S. M. . A. B. P. O. ML estimation of time and frequency offset in OFDM systems. **IEEE Trans. Commun.**, p. 1800–1805, 1997.
24. VAN DE BEEK, J. J. . E. O. . S. M. On channel estimation in OFDM systems. **IEEE VTC'95**, v. 2, p. 815–819, 1995.
25. TUFVESSON, F. A. M. T. Pilot assisted channel estimation for OFDM in mobile cellular systems. **IEEE VTC'97**, v. 3, p. 1639–1643, 1997.
26. COLERI, S. . E. M. . P. A. . A. B. A. Channel estimation techniques based on pilot arrangement in OFDM systems. **IEEE Trans. on Broadcasting**, p. 223–229, 2002.

Anexo A - Calculo de E_b/N_0 a partir dos valores de potência medidos

Como foi dito durante o texto, foram realizadas medidas do valor da potência do ruído P'_N e do valor da potência do sinal incidente P_I . Como sabemos que a potência do sinal incidente contém a parcela correspondente ao sinal OFDM P_S somado ao ruído, temos que:

$$P_I = P_S + P'_N \quad (47)$$

$$P_S = P_I - P'_N \quad (48)$$

A potência P'_N se encontra em toda a banda, incluindo o ZP. Porém, ao realizar o cálculo de P_S/P_N , devemos considerar somente a potência dentro da banda útil do sinal, ou seja, a banda onde estão as 150 portadoras contendo informação e símbolos piloto. Sendo assim, temos que:

$$P'_N = P_N \frac{150}{256} \quad (49)$$

$$SNR = \frac{P_S}{P_N} = \frac{P_S}{P'_N} \frac{256}{150} \quad (50)$$

Dessa forma pôde-se obter a relação P_S/P_N . Em seguida, sabemos que:

$$SNR = \frac{E_b W_I}{N_0 R_b} \quad (51)$$

Onde W_I é a banda de interesse. Em um sistema OFDM, temos símbolos de ZP, o qual devemos excluir da banda total e obter a banda de interesse. Portanto:

$$W_I = W \frac{150}{256} \quad (52)$$

Sendo assim, temos que:

$$\frac{E_b}{N_0} = \frac{P_S}{P_N} \frac{256}{150} W \frac{150}{256} \frac{1}{R_b} \quad (53)$$

$$\frac{E_b}{N_0} = \frac{P_S}{P_N} \frac{W}{R_b} \quad (54)$$

Como o tamanho do frame OFDM usado foi de 320 símbolos, a taxa de transmissão de bits pode ser calculada da seguinte forma:

$$R_b = WN_{bits} \frac{150}{320} \quad (55)$$

Sendo assim, podemos calcular a relação E_b/N_0 através da equação:

$$\frac{E_b}{N_0} = \frac{P_S}{P_N} \frac{1}{N_{bits}} \frac{320}{150} \quad (56)$$

Esses cálculos foram feitos assumindo que o filtro de entrada da USRP possui resposta em frequência retangular, o que não é verdade. Por tanto, incluiu-se na equação um fator de correção f_c , obtido através da comparação entre os resultados teóricos e os resultados obtidos, de forma a aproximar os resultados obtidos ao resultado teórico, compensando os efeitos causados por esse filtro. Dessa forma, a equação fica da seguinte forma:

$$\frac{E_b}{N_0} = \frac{P_S}{P_N} \frac{1}{N_{bits}} \frac{320}{150} f_c \quad (57)$$

O fator de correção obtido foi de $f_c = 7/2$.

Anexo B - Valores Obtidos nas Medições

Tabela 4 - Valores de potência medidos.

Ganho (dB)	Pn	Ps + Pn	Ps
0	2,31E-22	2,35E-22	4,39E-24
1	-	2,96E-22	6,5E-23
2	-	3,22E-22	9,1E-23
3	-	3,88E-22	1,58E-22
4	-	3,89E-22	1,58E-22
5	-	4,22E-22	1,91E-22
6	-	4,22E-22	1,91E-22
7	-	4,23E-22	1,92E-22
8	-	4,33E-22	2,02E-22
9	-	5,02E-22	2,72E-22
10	-	5,68E-22	3,37E-22
11	-	6,59E-22	4,28E-22
12	-	7,55E-22	5,24E-22
13	-	7,93E-22	5,62E-22
14	-	8,43E-22	6,12E-22
15	-	1,05E-21	8,14E-22
16	-	1,33E-21	1,1E-21
17	-	1,55E-21	1,32E-21
18	-	1,94E-21	1,71E-21
19	-	2,35E-21	2,11E-21
20	-	3,16E-21	2,93E-21
21	-	3,51E-21	3,28E-21
22	-	4,32E-21	4,09E-21
23	-	5,44E-21	5,2E-21
24	-	7,14E-21	6,91E-21
25	-	8,56E-21	8,33E-21
26	-	1,07E-20	1,05E-20
27	-	1,23E-20	1,2E-20
28	-	1,54E-20	1,52E-20
29	-	1,86E-20	1,84E-20
30	-	2,37E-20	2,35E-20
31	-	2,86E-20	2,84E-20

Tabela 5 - Valores de BER para mapeamento QPSK.

QPSK				
Ganho (dB)	Sem FEC	Com FEC RS(25,21)	Com FEC RS(25,19)	Com FEC RS(25,17)
0	8,10E-02	7,79E-02	4,57E-02	2,92E-02
1	4,76E-02	2,73E-02	1,23E-02	4,21E-03
2	1,97E-02	2,17E-02	2,61E-03	5,30E-04
3	1,40E-02	5,14E-04	2,18E-04	1,60E-05
4	5,27E-03	3,45E-05	7,01E-06	0,00E+00
5	6,51E-04	3,37E-06	2,01E-06	0,00E+00
6	6,08E-05	8,96E-07	0,00E+00	0,00E+00
7	2,35E-05	0,00E+00	0,00E+00	0,00E+00
8	3,09E-06	0,00E+00	0,00E+00	0,00E+00
9	6,13E-07	0,00E+00	0,00E+00	0,00E+00
10	1,84E-08	0,00E+00	0,00E+00	0,00E+00
11	0,00E+00	0,00E+00	0,00E+00	0,00E+00
12	0,00E+00	0,00E+00	0,00E+00	0,00E+00
13	0,00E+00	0,00E+00	0,00E+00	0,00E+00
14	0,00E+00	0,00E+00	0,00E+00	0,00E+00
15	0,00E+00	0,00E+00	0,00E+00	0,00E+00
16	0,00E+00	0,00E+00	0,00E+00	0,00E+00
17	0,00E+00	0,00E+00	0,00E+00	0,00E+00
18	0,00E+00	0,00E+00	0,00E+00	0,00E+00
19	0,00E+00	0,00E+00	0,00E+00	0,00E+00
20	0,00E+00	0,00E+00	0,00E+00	0,00E+00
21	0,00E+00	0,00E+00	0,00E+00	0,00E+00
22	0,00E+00	0,00E+00	0,00E+00	0,00E+00
23	0,00E+00	0,00E+00	0,00E+00	0,00E+00
24	0,00E+00	0,00E+00	0,00E+00	0,00E+00
25	0,00E+00	0,00E+00	0,00E+00	0,00E+00
26	0,00E+00	0,00E+00	0,00E+00	0,00E+00
27	0,00E+00	0,00E+00	0,00E+00	0,00E+00
28	0,00E+00	0,00E+00	0,00E+00	0,00E+00
29	0,00E+00	0,00E+00	0,00E+00	0,00E+00
30	0,00E+00	0,00E+00	0,00E+00	0,00E+00
31	0,00E+00	0,00E+00	0,00E+00	0,00E+00

Tabela 6 - Valores de BER para mapeamento 16-QAM

16-QAM				
Ganho (dB)	Sem FEC	Com FEC RS(25,21)	Com FEC RS(25,19)	Com FEC RS(25,17)
0	4,51E-01	4,51E-01	4,53E-01	4,59E-01
1	4,31E-01	4,31E-01	4,37E-01	4,42E-01
2	4,06E-01	4,09E-01	4,04E-01	4,45E-01
3	3,84E-01	3,65E-01	3,83E-01	3,95E-01
4	3,12E-01	3,28E-01	3,32E-01	3,92E-01
5	3,13E-01	2,83E-01	3,02E-01	3,50E-01
6	2,41E-01	2,34E-01	2,20E-01	3,09E-01
7	1,82E-01	1,55E-01	1,66E-01	2,23E-01
8	1,29E-01	3,17E-02	7,57E-02	1,80E-01
9	1,18E-01	5,14E-02	1,99E-02	1,05E-01
10	5,92E-02	1,48E-02	8,87E-03	1,69E-02
11	3,82E-02	6,52E-03	7,97E-04	4,62E-05
12	1,29E-02	2,07E-04	7,39E-05	9,92E-07
13	3,94E-03	3,41E-05	0,00E+00	7,48E-07
14	6,06E-03	1,92E-05	0,00E+00	0,00E+00
15	4,06E-04	0,00E+00	0,00E+00	0,00E+00
16	2,90E-05	0,00E+00	0,00E+00	0,00E+00
17	1,05E-05	0,00E+00	0,00E+00	0,00E+00
18	1,07E-06	0,00E+00	0,00E+00	0,00E+00
19	0,00E+00	0,00E+00	0,00E+00	0,00E+00
20	5,35E-07	0,00E+00	0,00E+00	0,00E+00
21	1,33E-07	0,00E+00	0,00E+00	0,00E+00
22	0,00E+00	0,00E+00	0,00E+00	0,00E+00
23	5,91E-08	0,00E+00	0,00E+00	0,00E+00
24	0,00E+00	0,00E+00	0,00E+00	0,00E+00
25	4,60E-08	0,00E+00	0,00E+00	0,00E+00
26	0,00E+00	0,00E+00	0,00E+00	0,00E+00
27	0,00E+00	0,00E+00	0,00E+00	0,00E+00
28	0,00E+00	0,00E+00	0,00E+00	0,00E+00
29	0,00E+00	0,00E+00	0,00E+00	0,00E+00
30	0,00E+00	0,00E+00	0,00E+00	0,00E+00
31	0,00E+00	0,00E+00	0,00E+00	0,00E+00

Anexo C - Apresentação do Modelo em LabVIEW do modem OFDM-RS

As figuras a seguir mostram trechos dos programas do transmissor e receptor OFDM-RS feitos no LabVIEW em linguagem gráfica.

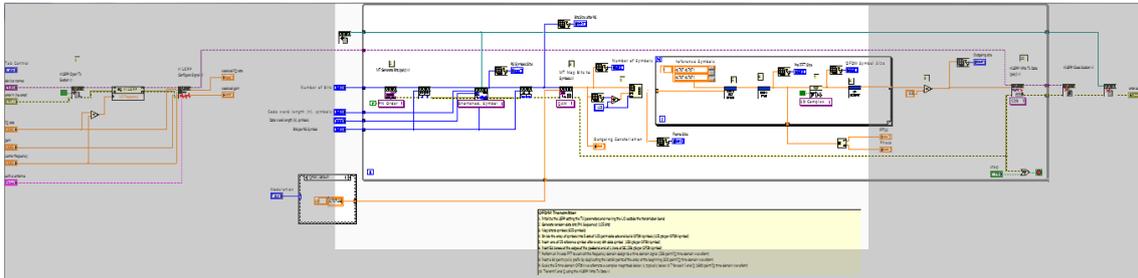


Figura 48 - Visão geral da implementação do transmissor OFDM-RS.

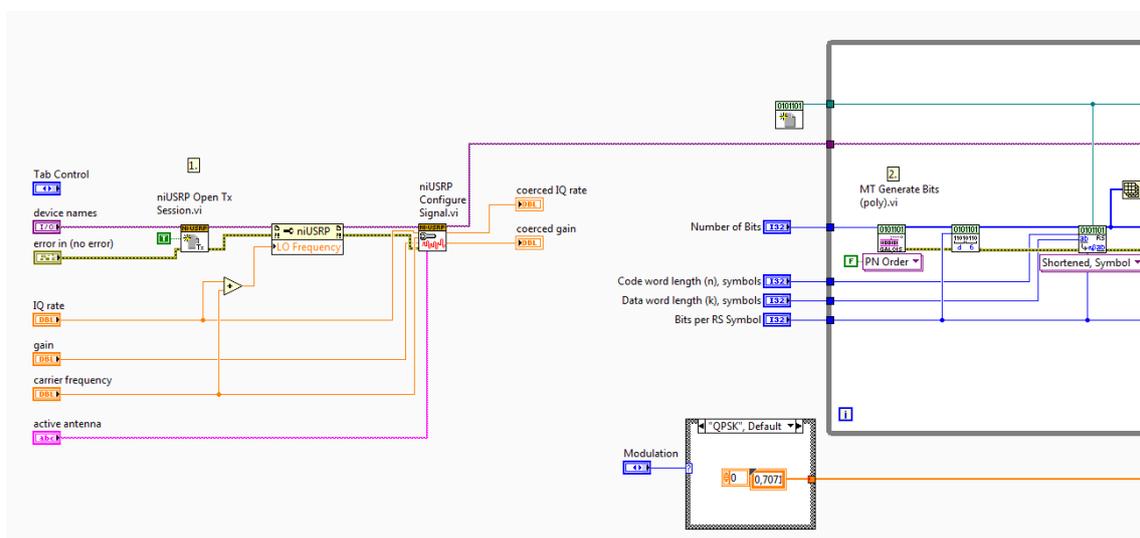


Figura 49 - Configuração de Tx da USRP.

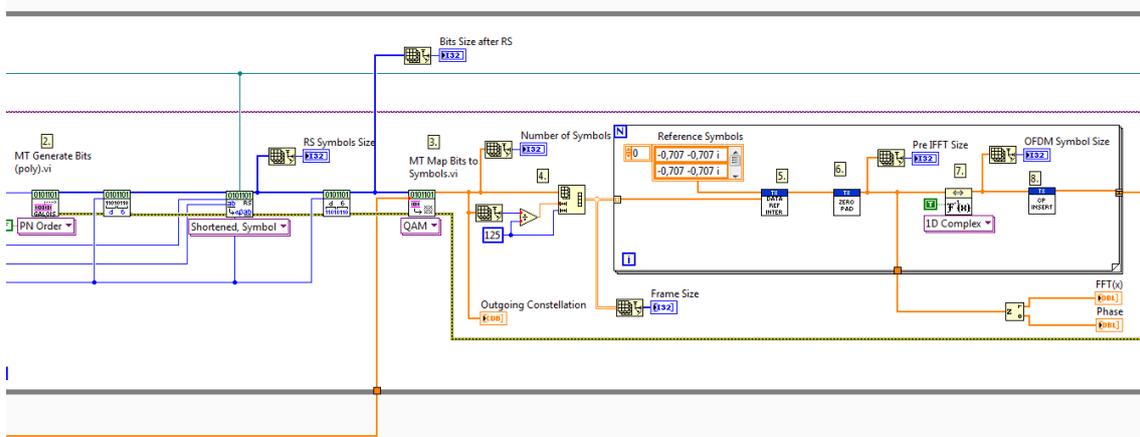


Figura 50 - Codificação RS e Modulação OFDM.

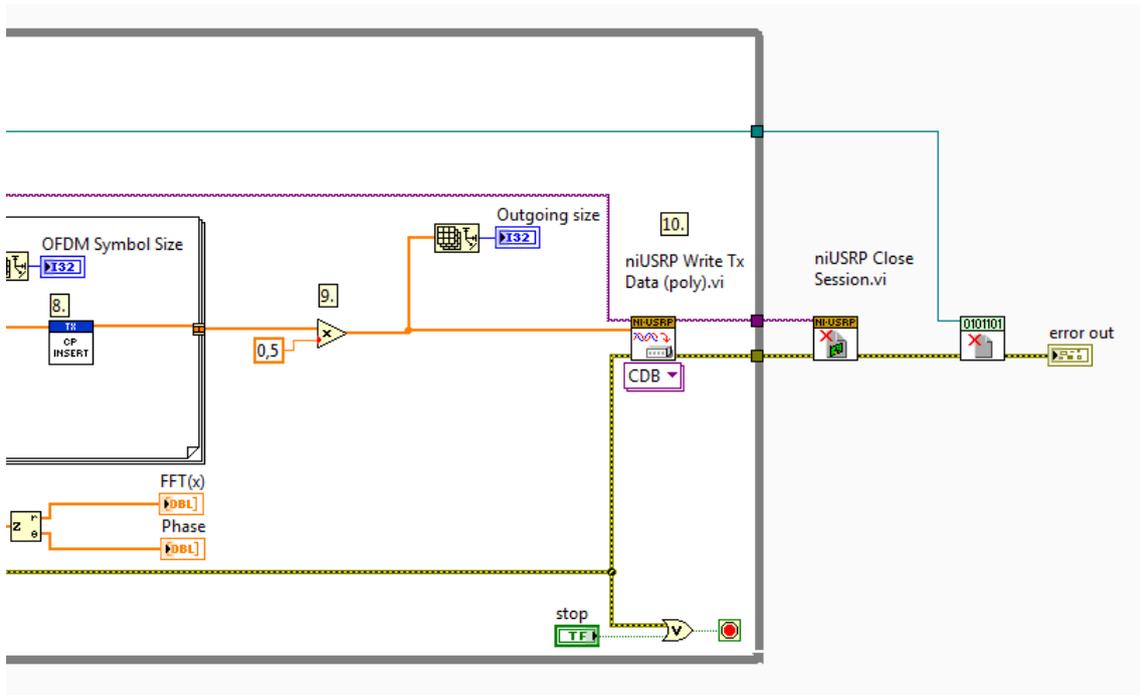


Figura 51 - Escrita dos símbolos resultantes no buffer da USRP e encerramento da sessão de Tx ao parar o programa.

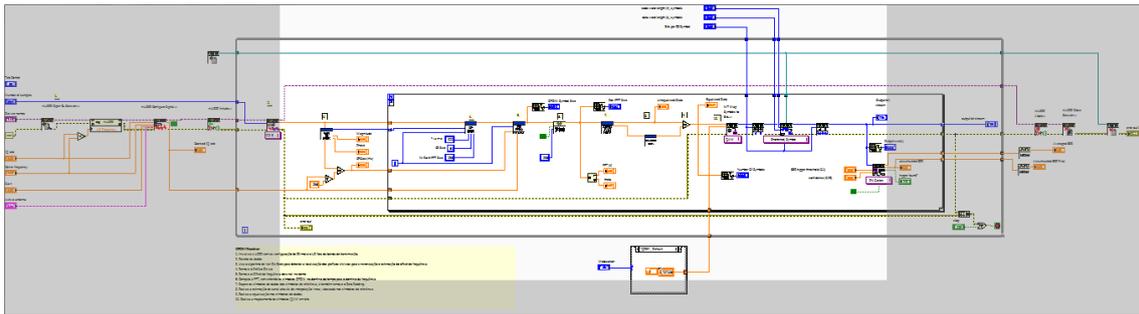


Figura 52 - Visão geral do receptor OFDM-RS.

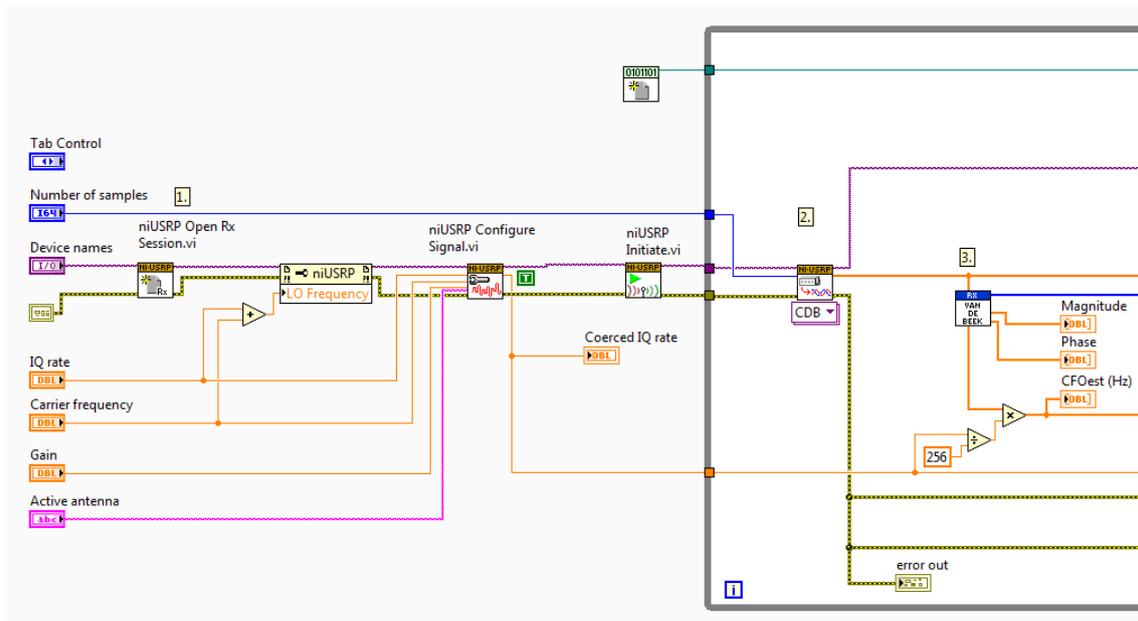


Figura 53 - Configuração da USRP como RX, leitura dos valores do Buffer na USRP e sincronismo.

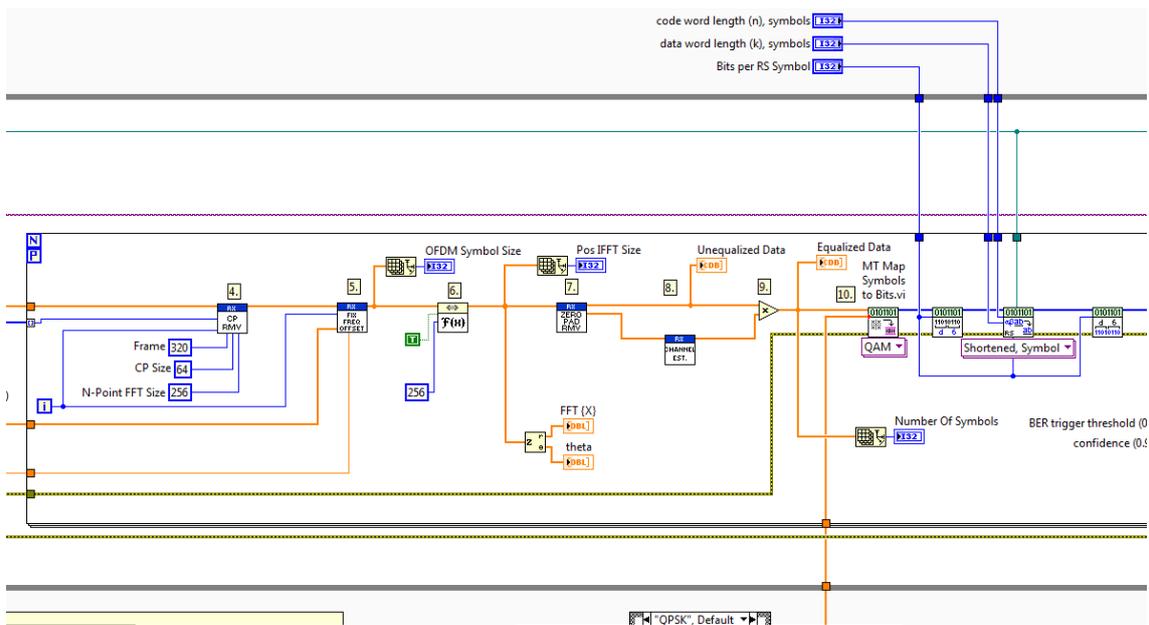


Figura 54 - Demodulação OFDM e decodificação RS.

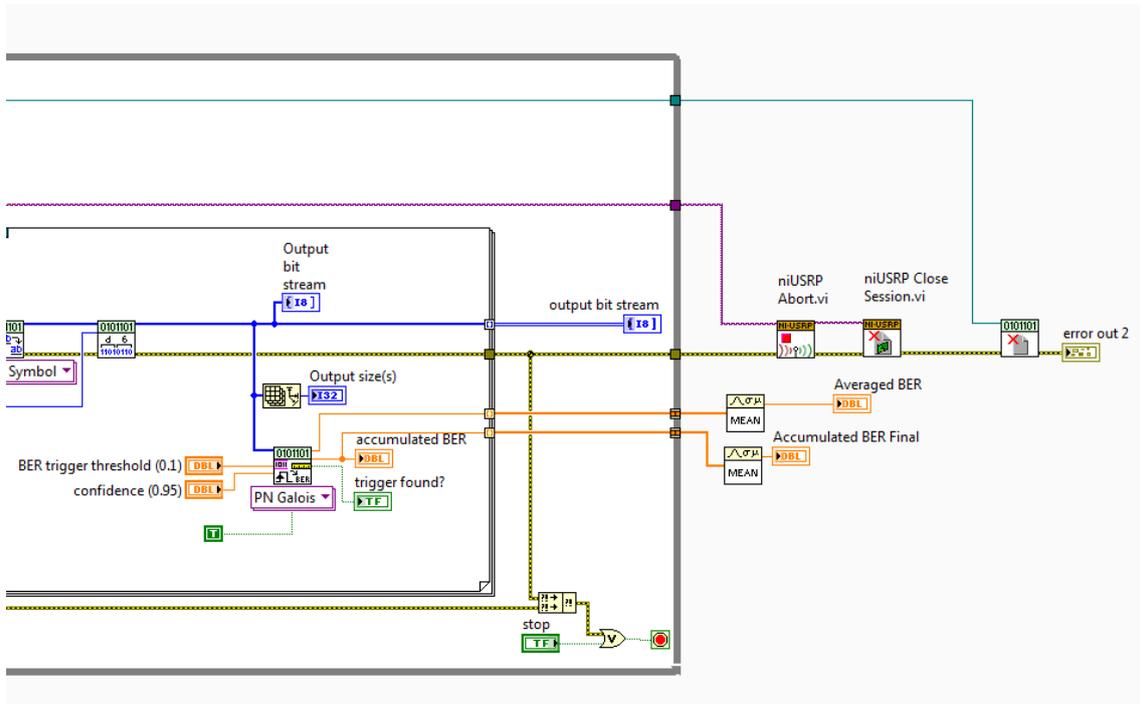


Figura 55 - Cálculo da BER e encerramento da seção de Rx caso o programa seja parado.