



UNIVERSIDADE ESTADUAL DE CAMPINAS
Faculdade de Engenharia Elétrica e de Computação

Segundo Gerardo Gamarra Quispe

**Análise e Implementação de um Sistema de
Comunicações Sem-Fio por Portadora Única
Utilizando Rádio Definido por Software**

**Analysis and Implementation of a Wireless
Communication System on an unique carrier using a
Software-defined Radio**

CAMPINAS

2015

Segundo Gerardo Gamarra Quispe

**Análise e Implementação de um Sistema de
Comunicações Sem-Fio por Portadora Única
Utilizando Rádio Definido por Software**

**Analysis and Implementation of a Wireless
Communication System on an unique carrier using a
Software-defined Radio**

Dissertação apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica, na Área de Telecomunicações e Telemática.

Dissertation presented to the School of Electrical and Computer Engineering of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Electrical Engineering in the area of Telecoms and Telematics

Orientador: Prof. Dr. Luís Geraldo Pedroso Meloni

Este exemplar corresponde à versão final da dissertação defendida pelo aluno Segundo Gerardo Gamarra Quispe, e orientada pelo Prof. Dr. Luís Geraldo Pedroso Meloni

CAMPINAS

2015

Agência(s) de fomento e nº(s) de processo(s): Não se aplica.

ORCID: <http://orcid.org/0000-0002-4438-8647>

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Luciana Pietrosanto Milla - CRB 8/8129

G14a Gamarra Quispe, Segundo Gerardo, 1980-
Análise e implementação de um sistema de comunicações sem-fio por portadora única utilizando rádio definido por software / Segundo Gerardo Gamarra Quispe. – Campinas, SP : [s.n.], 2015.

Orientador: Luís Geraldo Pedroso Meloni.

Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Software rádio. 2. Sistemas de comunicação sem fio. 3. Comunicações digitais. 4. Modulação de fase. I. Meloni, Luís Geraldo Pedroso, 1958-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Analysis and implementation of a wireless communication system on an unique carrier using a software defined radio

Palavras-chave em inglês:

Radio software

Wireless communication systems

Digital communications

Phase modulation

Área de concentração: Telecomunicações e Telemática

Titulação: Mestre em Engenharia Elétrica

Banca examinadora:

Luís Geraldo Pedroso Meloni [Orientador]

Paulo Batista Lopes

Gustavo Fraidenaich

Data de defesa: 27-11-2015

Programa de Pós-Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - DISSERTAÇÃO DE MESTRADO

Candidato: Segundo Gerardo Gamarra Quispe RA: 123260

Data da Defesa: 27 de novembro de 2015

Título da Tese:

“Análise e Implementação de um Sistema de Comunicações Sem-Fio por Portadora Única Utilizando Rádio Definido por Software”

“Analysis and Implementation of a Wireless Communication System on an unique carrier using a Software-defined Radio”

Prof. Dr. Luís Geraldo Pedroso Meloni (Presidente, FEEC/UNICAMP)

Prof. Dr. Paulo Batista Lopes (/Mackenzie)

Prof. Dr. Gustavo Fraidenraich (FEEC/UNICAMP)

Ata de defesa, com as respectivas assinaturas dos membros da Comissão Julgadora, encontra-se no processo de vida acadêmica do aluno.

Dedico esta dissertação para minha musa inspiradora

Agradecimentos

Agradeço,

ao Prof. Luis Geraldo P. Meloni pela importante orientação e amizade.

aos colegas de trabalho mais próximos: Jackelyne Tume, Elvis Jara, Jaime Luque, Jose Luis Hinostroza, Júlio Cesar e Gabriel Trevisam pela amizade e conselhos durante todo o meu mestrado.

aos demais colegas do Departamento de Comunicações pela ótima convivência.

para minhas amigas e amigos Sheila, Jacqueline, Ana, Elvis, Pedro e Willian, que ajudaram a melhorar a qualidade da redação final da dissertação aos professores da FEEC, pelos ótimos cursos oferecidos.

Para meus pais brasileiros Ronivaldo Martins e Silvana Celia, para meu irmão Dennys Martins e minha irmã Danielle Cristina, que me acolheram em sua casa como um filho, muito obrigado.

aos membros da banca examinadora pelos comentários, sugestões e contribuições, que ajudaram a melhorar a qualidade e a redação final do manuscrito.

à FEEC/UNICAMP pela ótima estrutura e equipamentos que oferece aos estudantes e pesquisadores.

à CAPES pelo portal de periódicos eletrônicos, que permite o acesso rápido e eficiente ao conhecimento científico.

a todos que de alguma forma contribuíram com o meu progresso como aluno e como ser.

ao Brasil pela maravilhosa experiência.

Resumo

Este trabalho visa estudar a tecnologia de rádio definido por software (SDR) e utilizá-la na implementação de um sistema de comunicações sem fio em portadora única. Visa ainda sua aplicação no ensino dos sistemas de telecomunicações. SDR é uma tecnologia em desenvolvimento na qual todas ou algumas das funções da camada física do sistema são implementadas em software, o que provê vantagens em relação à tecnologia de rádio convencional, tais como flexibilidade de configuração e redução do tempo de desenvolvimento. Diz-se que o sistema é flexível porque permite que um único equipamento possa ser reconfigurado via software de acordo com o sistema de rádio desejado, sem precisar modificar o hardware. Reduz-se o tempo de desenvolvimento, porque pequenas ou grandes mudanças no sistema podem ser feitas no programa e verificadas rapidamente. Na presente dissertação, analisa-se e implementa-se um sistema de comunicação sem fio em portadora única, utilizando rádio definido por software (SDR). A plataforma de desenvolvimento de software empregada é a do Simulink/Matlab e o hardware é o periférico N210 desenvolvida pelo fabricante *Ettus Research*, uma empresa pertencente à *National Instrument*. Como parte do trabalho implementou-se o padrão europeu EN 300-421 de transmissão de vídeo digital por satélite (DVB-S).

Finalmente, mostra-se a implementação do sistema SDR em dois cenários, primeiramente mostra-se a simulação e seus respectivos gráficos BER para medir o desempenho, em seguida, mostra-se a implementação em um cenário mais real, i.e com os equipamentos físicos de SDR e transmitindo pelo ar de um computador a outro.

Palavras-chaves: Rádio definido por software; transmissão de fluxo de transporte de dados por satélite; Modulação QPSK; Reed Solomon.

Abstract

This work aims at studying the software defined radio (SDR) technology and use it to implement a wireless communication system on single carrier for telecommunications systems teaching purposes. SDR is a developing technology in which all or some of the functions of the physical layer are implemented in software, which provides advantages over conventional radio technology such as configuration flexibility and reduction in development time. It is said that the system is flexible because it allows a single device to be reconfigured via software according to the desired radio system without modifying the hardware; and reduces development time because small or large changes in the system can be made in the program and verified quickly. In this work, a wireless communication system in single carrier is analyzed and implemented by using software-defined radio (SDR). The software used in this system is Simulink / Matlab and the hardware is the N210 peripheral from Ettus Company. As part of the work, the European standard EN 300-421 digital video broadcasting satellite (DVB-S) was implemented.

Finally, two cases of the SDR system implementation are shown. The first one is about our simulation and BER graphics is shown to measure the performance. The second one shows a more real case, i.e. using the Ettus devices and wireless transmission using two computers.

Keywords: Software defined radio; Digital Video Broadcasting-Satellite; Modulation QPSK; Reed Solomon.

Lista de ilustrações

Figura 1 – Diagrama de um sistema de comunicação digital	18
Figura 2 – Diagrama funcional de um SDR	20
Figura 3 – Diagrama de blocos de um USRP.	21
Figura 4 – Tipo de software usados no SDR.	24
Figura 5 – Fluxo de dados em blocos de GNU Radio.	25
Figura 6 – Diagrama de blocos do transmissor DVB-S.	26
Figura 7 – Estrutura do pacote de transporte MPEG-2	27
Figura 8 – Diagrama do randomizador/De-randomizador.	28
Figura 9 – Pacotes de TS mostrados na saída do PRBS.	29
Figura 10 – Diagrama de blocos do codificador Reed-Solomon.	29
Figura 11 – Um <i>interleaver</i> e um <i>deinterleaver</i> de 3x4 (MOON, 2002).	31
Figura 12 – Exemplo do interleaving.	31
Figura 13 – Diagrama de funcionamento <i>interleaver</i> e <i>deinterleaver</i> , adaptado de (FISCHER, 2010).	32
Figura 14 – Diagrama de bloco do codificador interno (CCSDS, 2012).	33
Figura 15 – Funcionamento do <i>puncturing</i> para todas as taxas	34
Figura 16 – Configuração básica de um esquema de modulação em quadratura.	36
Figura 17 – Diagrama de blocos funcional do sistema de recepção DVB-S.	38
Figura 18 – De-interleaving (FISCHER, 2010)	40
Figura 19 – Diagrama de blocos do decodificador Reed-Solomon.	40
Figura 20 – Estrutura básica de um PLL em tempo discreto (a) e seu correspondente PLL equivalente de fase (b) (RICE, 2009).	42
Figura 21 – Laço de captura de fase em tempo contínuo de segunda ordem com filtro <i>proportional-plus-integrator</i> (a) e PLL equivalente de fase linearizada (b).	43
Figura 22 – Laço de fase bloqueada em tempo discreto de segunda ordem com filtro <i>proportional-plus-integrator</i> (a) e PLL equivalente de fase linearizada em tempo discreto (b).	44
Figura 23 – No domínio da frequência o sinal de banda base (a) é deslocado ao redor frequência da portadora gerando o sinal de banda passante (b)	45
Figura 24 – Sistema de comunicações passabanda ideal.	46
Figura 25 – Sistema de comunicações passabanda mais realista, adaptado de (EVERETT, 2013)	46
Figura 26 – Etapas do processo sincronismo.	47
Figura 27 – Funcionamento do sincronismo de fase	48
Figura 28 – Funcionamento do sincronismo do relógio (RICE, 2009)	50
Figura 29 – Interpolação (RICE, 2009)	51

Figura 30 – Contador de Controle de interpolação (RICE, 2009)	52
Figura 31 – Relação entre as amostras disponíveis, as amostras geradas da interpoladas e o conteúdo do contador de módulo-1 (RICE, 2009)	53
Figura 32 – Correlação da palavra de sincronismo (SW) com a data recebida.	54
Figura 33 – Periférico USRP-N210 da Ettus (ETTUS, 2014).	56
Figura 34 – Diagrama de blocos do transmissor, canal e receptor implementado no cenário controlado no computador	58
Figura 35 – Diagrama de blocos no Simulink	59
Figura 36 – BER versus Eb/No. Comparação entre QPSK teórico, QPSK simulado, e DVB-S simulado com <i>puncturing</i> 1/2.	60
Figura 37 – BER versus Eb/No, para QPSK e o sistema DVB-S, com <i>puncturing</i> em uma taxa de código 3/4.	61
Figura 38 – Equipamento para teste do sistema de comunicação.	61
Figura 39 – Diagrama de componentes usados no cenário real.	62
Figura 40 – Diagrama do transmissor.	62
Figura 41 – Diagrama do receptor.	63
Figura 42 – Constelação obtida na entrada do sincronismo de frequência	63
Figura 43 – Constelação obtida na saída do sincronismo de frequência.	64
Figura 44 – Constelação obtida na saída do sincronismo de fase.	64
Figura 45 – Constelação obtida na saída do sincronismo de relógio.	65
Figura 46 – Diagrama de blocos simplificado da cadeia de recepção da placa filha WBX e o ADC da placa mãe USRP.	72

Lista de tabelas

Tabela 1 – Características de alguns USRP	22
Tabela 2 – Relação de fases e sinais QPSK	35
Tabela 3 – Especificações do USRP N210.	56

Sumário

1	Introdução	14
1.1	Organização da Dissertação	16
2	Conceitos Básicos	17
2.1	Modelo de um Sistema de Comunicações Digitais	17
2.2	Rádio Definido por Software (SDR)	18
2.2.1	Antecedentes	19
2.2.2	Funcionamento de um Sistema SDR	20
2.2.2.1	Modo Transmissão	21
2.2.2.2	Modo Recepção	21
2.3	Componentes de Hardware	22
2.3.1	Periférico de Rádio de Software Universal (USRP)	22
2.4	Software Utilizado no SDR	23
2.4.1	Software de Controle	24
2.4.2	Software de Processamento do Sinal	24
3	Sistema de Radiodifusão de Vídeo Digital por Satélite (DVB-S)	26
3.1	Transmissor	26
3.1.1	Codificador de Fonte	27
3.1.2	Codificador de Canal	27
3.1.3	Modulação por Chaveamento Quadrifásico (QPSK)	35
3.2	Receptor	37
3.2.1	Demodulador QPSK	39
3.2.2	Decodificador de Canal	39
4	Sincronismo do Sistema	41
4.1	Malha de Captura de Fase (PLL) em Tempo Discreto	41
4.2	Recuperação da Portadora	45
4.2.1	Sincronismo Grosseiro em Frequência	47
4.2.2	Sincronismo de Frequência Residual e da Fase da Portadora	48
4.2.3	Sincronismo do Relógio	49
4.3	Sincronismo de Quadro	52
5	Implementação do Sistema e Resultados	55
5.1	Equipamentos Empregados na Implementação	55
5.2	Implementação	56
5.3	Testes	56

5.3.1	Cenário Controlado no Computador	57
5.3.2	Cenário Real	61
6	Considerações finais	66
6.1	Trabalhos Futuros	68
	Referências	69
	Anexos	71
	ANEXO A Placa filha WBX.	72

1 Introdução

No último século o desenvolvimento tecnológico nas mais variadas áreas do conhecimento cresceu exponencialmente, sendo muito evidente na área de telecomunicações. É impressionante observar como dispositivos eletrônicos ficam obsoletos em apenas um ou dois anos após fabricação. Embora essa situação represente um avanço na tecnologia, pode se converter em ameaças aos sistemas correntes, necessitando pesquisas constantes tanto pela academia quanto pela própria indústria do setor de telecomunicações.

Para superar estes desafios, os sistemas de comunicações que operem independentemente das tecnologias de hardware apresentam diversas vantagens quando comparados às tecnologias convencionais. Tais sistemas permitem a operação em diversos padrões e mesmos em diversas faixas de frequências, aumentando-se desta forma o ciclo de vidas destes equipamentos.

Um sistema de rádio convencional usa, principalmente, dispositivos de hardware associados à uma configuração de software mínima ou inexistente. A rápida evolução da tecnologia implica curto tempo de vida para este tipo de sistemas. A terminologia "Software Defined Radio" (SDR) foi utilizada pela primeira vez em 1992, por Mitola (MITOLA J., 1992), em resposta ao problema da rápida obsolescência dos sistemas de rádio convencional. O intuito do rádio software tem sido a criação de esquemas que sejam reconfiguráveis por meio do software e com uma quantidade de dispositivos de hardware mínima. Dessa maneira, o mesmo dispositivo de hardware pode ser usado para atender diferentes aplicações configuradas via software. Este fato é enfatizado em uma das definições para o SDR: "Rádio na qual algumas ou todas as funções da camada física são definidas por software" citado no *Wireless Innovation Forum* (FORUM, 2014). O Rádio definido por software (SDR) é uma tecnologia que nos últimos anos está recebendo grande interesse tanto da academia quanto da indústria, devido à sua rápida evolução, representando um avanço importante nas telecomunicações, sendo uma tecnologia flexível e de custo acessível.

O ensino das disciplinas de comunicações fundamentam-se em conceitos de processamento de sinais, técnicas de modulação analógica e digital, estudo e análise de filtros e teoria da informação e codificação. Tais conteúdos de natureza matemática em sua essência apresentam desafios ao ensino e aprendizagem pelos alunos. Tais dificuldades advêm da diversidade de tópicos teóricos e muitas equações sobre o tema, somadas ao fato de que a maioria dos alunos não têm familiaridade às experiências práticas ou realizações físicas destes sistemas. A tecnologia em estudo pode ser usada como uma valiosa ferramenta educacional auxiliando os alunos em um aprendizado mais eficiente dos diferentes aspectos dos sistemas de comunicações. O uso do SDR em experiências de laboratório vai permitir a exibição de conceitos avançados de uma

forma mais simples (ZIVKOVIC *et al.*, 2014).

Esta dissertação busca contribuir com este importante problema de interesse didático. Para isso, buscou-se primeiramente o aprofundamento dos conhecimentos em um sistema de comunicações sem fio em portadora única, por ser mais simples que os sistemas baseados em múltiplas portadoras. Também procurou-se buscar aderência aos padrões reais de comunicações. Para tal foi escolhido o sistema de radiodifusão satelital *Digital Video Broadcasting by Satellite* - DVB-S, com base no padrão EN 300 421. Por se tratar de um software matemático amplamente utilizado na área acadêmica, também definiu-se o uso do Simulink no ambiente Matlab. Apesar de sua relativa simplicidade, cada módulo do sistema de transmissão e recepção de sinais é fruto do trabalho de muitos pesquisadores da área especializada detalhados ao longo desta dissertação.

Uma importante contribuição de nossa pesquisa é a implementação em SDR do sistema de comunicações DVB-S com base no padrão EN 300 421. A grande quantidade de algoritmos e parâmetros que conformam os processos de transmissão e recepção existentes aumenta a complexidade do estudo do tema. Isto representou o desafio da implementação e depuração de diversos métodos e algoritmos. Outro grande desafio que foi enfrentado é quanto ao tempo de teste destes módulos, pois envolvem grandes quantidades de dados para diversos cenários de simulação.

A motivação principal deste trabalho foi o estudo da tecnologia de rádio definido por software visando sua aplicação no ensino das telecomunicações. A tecnologia em estudo pode ser usada como uma valiosa ferramenta educacional que permite a inclusão de novos conteúdos teóricos, além de aplicações práticas, mais flexível quando comparados aos cursos clássicos de telecomunicações. O SDR abre as possibilidades de fazer demonstrações em sala de aula que ilustrem os conceitos teóricos, motivando o aluno e ajudando na compreensão (KATZ; FLYNN, 2009) dos conteúdos.

Com respeito à metodologia, neste trabalho realizou-se o análise e implementação do padrão europeu EN 300 421 (ETSI, 1993) de transmissão de vídeo digital por satélite (DVB-S), em SDR. A escolha deste padrão foi feita por este apresentar uma complexidade moderada, já que possui uma portadora única com modulação QPSK e emprega códigos para detecção e correção de erros de uso comum. A baixa complexidade foi motivada por questões didáticas de ensino.

O software empregado para a implementação do sistema é Matlab/ Simulink e o periférico de rádio de software universal (USRP) empregado é o modelo N210 do fabricante *Ettus Research* (ETTUS, 2014), empresa de *National Instrument* (INSTRUMENTS, 2014). A análise de operação e desempenho do sistema DVB-S é feita por meio de métricas de taxa de erros de bits (BER) em função da relação energia de bit sobre densidade de ruído espectral (E_b/N_0) em canal com *Additive white Gaussian noise* (AWGN).

Este trabalho tem como objetivos o estudo da tecnologia de rádio definido por software (SDR) e sua utilização na implementação de um sistema de comunicações sem fio em portadora única para a aplicação ao ensino dos sistemas de telecomunicações. Teve ainda como escopo a implementação do padrão de radiodifusão por satélite, o DVB-S.

Como objetivos específicos, podemos citar:

- Estudo sobre SDR, DVB-S, USRP N210, entre outros.
- Implementação do sistema de transmissão e recepção SDR baseado no padrão DVB-S.
- Planejamento de uma metodologia para a realização dos experimentos com o intuito de avaliar o sistema implementado.
- Desenvolvimento de experimentos para obter métricas da taxa de erro de bit (BER).

1.1 Organização da Dissertação

Este documento está organizado em seis capítulos conforme descrito a seguir.

O Capítulo 2 apresenta os principais conceitos que são usados no decorrer do estudo. Neste capítulo primeiramente é apresentado um sistema geral de comunicações digitais. A seguir, examinam-se as principais características de SDR assim como o seu funcionamento na transmissão e recepção. Posteriormente é descrito o periférico USRP que é o principal componente de hardware na comunicação. Finalmente, apresenta-se o software de controle e de processamento de sinal.

O Capítulo 3 apresenta o padrão europeu EN 300 421 do DVB-S, onde se aprofunda em mais os detalhes o funcionamento dos blocos do transmissor, explicando-se os conceitos de cada etapa do processo de transmissão e de recepção do sistema. Cabe destacar que o padrão somente especifica os detalhes do funcionamento do transmissor e descreve o receptor em alto nível. Os detalhes do funcionamento do receptor no padrão são deixados em aberto para diversas soluções de implementação.

Posteriormente, o Capítulo 4 trata do estudo e fundamentação do sincronismo de fase e do sincronismo do relógio do sistema em análise.

O Capítulo 5 apresenta uma descrição do equipamento usado na implementação e mostra os resultados obtidos.

Finalmente, o Capítulo 6 conclui este trabalho, apresentando as considerações finais sobre a pesquisa realizada, bem como sugestões de continuidade em trabalhos futuros da pesquisa.

2 Conceitos Básicos

Esta seção apresentamos conceitos gerais da área de Rádio Definido por Software (SDR). Primeiramente é apresentado um modelo de um sistema de comunicações digitais com os subsistemas do transmissor e receptor e dos seus blocos internos. Em seguida, apresenta-se o sistema de rádio definido por software e suas principais características. Na sequência, descreve-se o periférico de rádio de software universal (USRP), detalhando brevemente sua composição interna. Finalmente é apresentado o software usado para o processamento de sinal e outras possíveis opções.

2.1 Modelo de um Sistema de Comunicações Digitais

O termo “comunicações digitais” envolve uma ampla área de técnicas de comunicações, incluindo a transmissão digital e o rádio digital. A transmissão digital é a transmissão de pulsos digitais entre dois ou mais pontos de um sistema de comunicações. O rádio digital é a transmissão de portadoras analógicas moduladas a partir de sinal de informação digital entre dois ou mais pontos de um sistema de comunicações.

Em uma transmissão entre transmissor e receptor, as informações são enviadas através de um meio físico de transmissão, chamado *canal físico* ou simplesmente *canal*. Um canal físico entre o transmissor e o receptor pode ser um cabo de par trançado de condutores, um cabo coaxial ou um cabo de fibra ótica. Nos sistemas de rádio digital, o meio de transmissão é o espaço livre ou a atmosfera da Terra. (TOMASI, 2003). Os canais físicos apenas podem transmitir sinais reais e analógicos.

Na Figura 1, pode-se observar o diagrama de um sistema de comunicações digital. A informação inicia seu percurso saindo da fonte. Estes dados são recebidos pelo codificador de fonte que se encarrega de retirar possíveis redundâncias do sinal. Em seguida, o codificador de canal protege os dados da interferência de canal, em geral, com bits de redundância em forma de códigos corretores de erro.

O modulador, em seguida, recebe os dados digitais do codificador de canal e os mapeia em sinais analógicos limitados e fixos. Esses sinais analógicos são então transmitidos pelo canal, o qual via de regra, adiciona ruído e distorce o sinal. Esse sinal será a entrada do demodulador, o qual o converterá novamente para o formato digital, sendo a sua saída uma sequência discreta de números.

O demodulador recebe o sinal vindo do canal. Um demodulador ótimo para canais AWGN possui um filtro casado, ou um correlator acompanhado de um amostrador. Os dados recebidos, os quais podem ter partes corrompidas pelo canal, são entregues ao decodificador

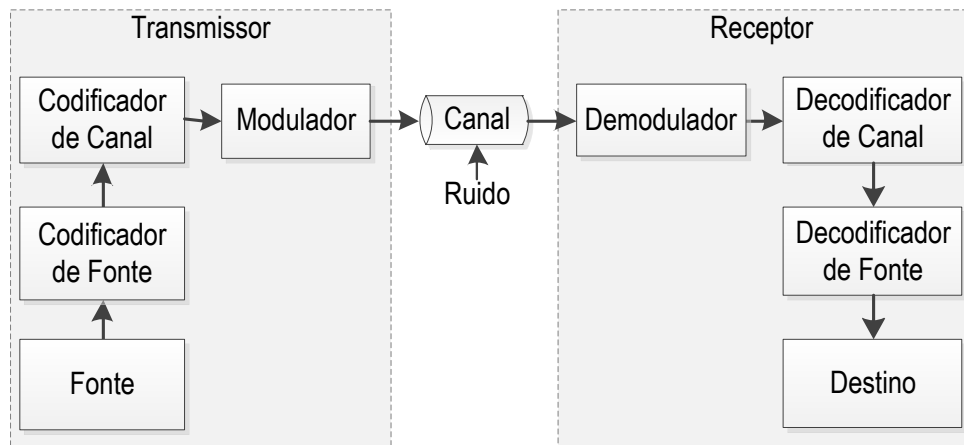


Figura 1 – Diagrama de um sistema de comunicação digital

de canal que, usando o código corretor de erro adicionado aos dados gerados pela fonte pelo codificador de canal, busca detectar e corrigir erros, reduzindo a probabilidade de erro dos dados recebidos. O decodificador de fonte visa reconstruir o sinal aplicado ao codificador de fonte, permitindo com que se possa reproduzir adequadamente os dados emitidos pela fonte de informação. Finalmente, esses dados reproduzidos chegam ao destino.

2.2 Rádio Definido por Software (SDR)

O *Software Defined Radio* *Software Defined Radio* é uma tecnologia em evolução composta de software e hardware com aplicações possíveis em diversas áreas, criada para melhorar a interoperabilidade entre diferentes serviços. Nele, algumas ou todas as funções das camadas físicas são definidas por software (FORUM, 2014), enquanto no rádio convencional as funções da camada física são implementadas em hardware específico, portanto quando se requer uma pequena mudança, o dispositivo de hardware inteiro precisa ser reprojeto e implementado novamente. Este reprojeto acarreta perdas de tempo e recursos. Dessa forma, a tecnologia de SDR é, portanto, mais flexível e reconfigurável que a tecnologia de rádio convencional e permite criar dispositivos sem fio e sistemas de comunicações que podem ser dinamicamente reconfiguráveis mediante atualizações de software ou reconfigurações de hardware (REIS *et al.*, 2012).

O conceito de SDR vem evoluindo com o tempo, mas em linhas gerais o diagrama básico estrutural na Figura 2 ilustra bem o conceito.

Algumas das principais características do SDR (REED, 2002) são descritas a seguir:

1. **Fácil de projetar:** A implementação de rádio por software reduz os ciclos de desenvolvimento para novos produtos. Desta forma, economiza-se um número de etapas necessárias associadas com o desenvolvimento de hardware analógico. É possível projetar uma variedade de equipamentos de rádio usando uma interface de RF (*Radio Frequency*) comum.

2. **Fácil de fabricar:** Partindo do fato que dois componentes analógicos não têm precisamente o mesmo desempenho, o hardware analógico precisa de rigoroso controle de qualidade e teste durante o processo de manufatura. Entretanto dois processadores digitais rodando o mesmo software reproduzem, em geral, saídas equivalentes. Portanto, a mudança para o hardware digital reduz os custos associados à fabricação e testes.
3. **Operação multimodo:** O explosivo crescimento dos sistemas de comunicação *wireless* tem levado a uma proliferação de padrões de transmissão e, em muitos casos, é desejável que o rádio opere com mais de um padrão. Tradicionalmente, a operação multimodo requer múltiplos conjuntos completos de hardware, aumentando o tamanho e custo do rádio. Um SDR, por sua vez, pode ter carregado em memória uma variedade de padrões de comunicações, mudando a maneira de funcionar pela simples reconfiguração de software.
4. **Uso de técnicas avançadas de processamento de sinal:** A disponibilidade de processamento de sinal de alta velocidade na placa de rádio permite a implementação de novas estruturas de receptores e técnicas de processamento digital. Técnicas como o uso de antenas adaptativas, rejeição de interferência e criptografia, anteriormente consideradas muito complexas, estão agora ocupando espaço em sistemas comerciais visto que o aumento do desempenho de processadores digitais de sinais continua crescendo. O impacto será a melhora na qualidade de serviço para o consumidor enquanto se reduz o custo total da infraestrutura para o provedor de serviço.
5. **Poucos componentes discretos:** Um único processador de alta velocidade deve ser capaz de implementar várias funções de rádio tradicionais como sincronização, demodulação, detecção de erro e decodificação de dados. Sendo assim, há uma redução do número de componentes necessários e com isso há uma diminuição do tamanho e do custo de um equipamento de rádio.
6. **Flexibilidade para adicionar funcionalidades extras:** O SDR pode ser modificado em campo para corrigir problemas imprevistos ou para a atualização do sistema. Tal opção não existe em rádios convencionais e caso houvesse a necessidade de adição de uma nova característica, tal processo levaria a uma reavaliação completa do projeto. Além da possibilidade de se adicionar facilmente funcionalidades extras, a modularidade dos sistemas SDR também contribui para a facilidade de criação de ferramentas de diagnósticos e correção de falhas de hardware.

2.2.1 Antecedentes

Com o objetivo de facilitar e aperfeiçoar as comunicações e trocas de informação entre dispositivos fisicamente distantes, umas das primeiras implementações conhecidas de SDR foi o projeto SpeakEasy (LACKEY, 1995) do exército dos Estados Unidos. Foi o primeiro

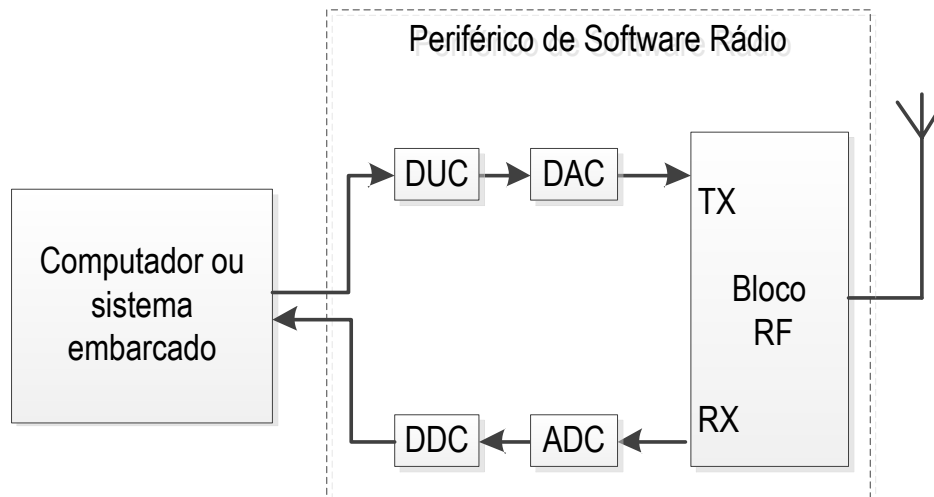


Figura 2 – Diagrama funcional de um SDR

projeto conhecido a utilizar uma FPGA para processamento de dados digitais. O principal objetivo era implementar em um equipamento programável mais de 10 das tecnologias de comunicações sem fio mais utilizadas. O equipamento tinha que funcionar entre as bandas 20 MHz e 200 MHz e atualizar seu código caso futuras atualizações no software fossem desenvolvidas

O projeto teve uma duração de quatro anos (1991-1995) e conseguiu cumprir com seus objetivos iniciais. Foi iniciada uma nova fase do projeto para incluir melhoras significativas, tais como incremento de sua capacidade de processamento e simultaneidade de comunicações (na primeira fase só era permitida uma comunicação por vez). O design foi baseado em arquitetura de software livre e trabalhou nas bandas de 4 MHz a 400 MHz.

Atualmente as pesquisas e implementações de SDR são de grande interesse no mundo inteiro e sua importância é tal que foi criado o SDR Forum (atualmente com o nome de *Wireless Innovation Forum*), formado por mais de 100 empresas, organizações e centros de pesquisa e investigação, cuja finalidade é a pesquisa e desenvolvimento mundial em SDR, o incentivo à utilização do SDR e apresentação de suas vantagens no ambiente civil, militar e industrial. A divulgação é feita tanto em eventos quanto no site do próprio fórum, no qual são publicados os avanços e notícias relacionadas ao SDR.

2.2.2 Funcionamento de um Sistema SDR

O funcionamento do sistema no modo de transmissão e recepção é apresentado na Figura 2 e descrito nas seções a seguir. Chamaremos de Periférico de Software Rádio, o conjunto formado pelos blocos *Digital Up-Converter* (DUC), *Digital to Analog Converter* (DAC), *Digital Down-Converter* (DDC), *Analog to Digital Converter* (ADC) e os blocos de transmissão e recepção RF, conforme pode ser observado na Figura 2.

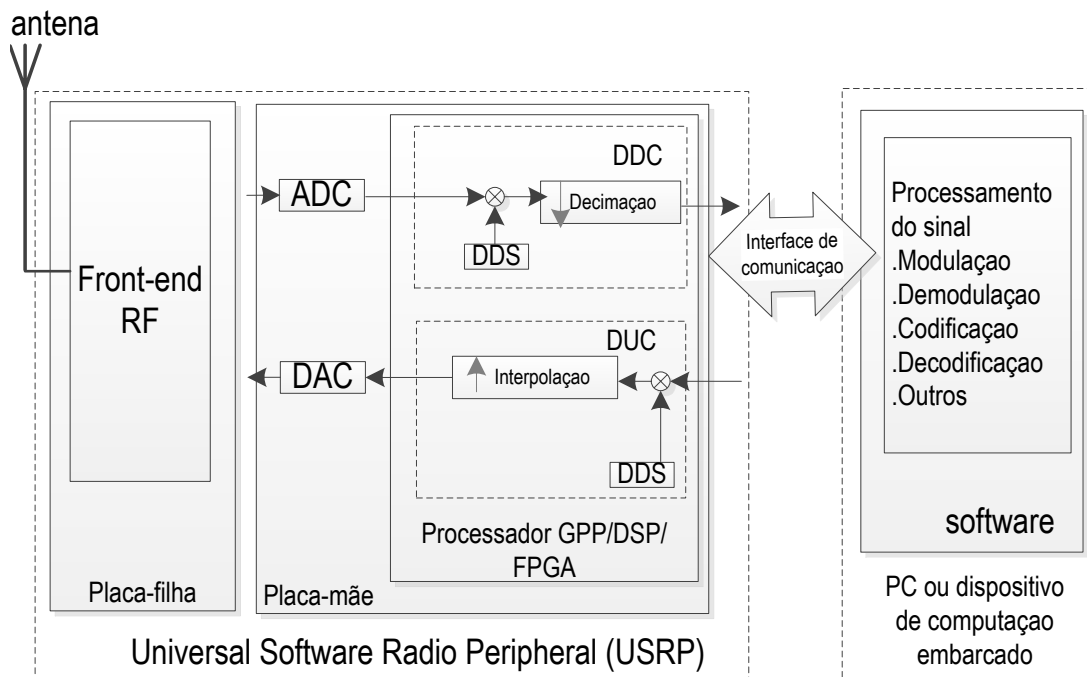


Figura 3 – Diagrama de blocos de um USRP.

2.2.2.1 Modo Transmissão

1. O processo de transmissão inicia no computador ou sistema embarcado. O software instalado no computador ou o *firmware* inserido no sistema embarcado realiza as operações de processamento tais como modulação, demodulação, codificação, decodificação, entre outros, gerando dados adequados para serem transmitidos.
2. Esses dados processados são enviados ao periférico de *software radio* por um protocolo de comunicação serial.
3. No periférico de *software radio* ocorre um incremento da taxa de amostragem no bloco DUC *Digital Up-Converter* (*Digital Up-Converter* (DUC)).
4. Posteriormente, o sinal digital é convertido em sinal analógico no DAC *Digital to Analog Converter* (*Digital to Analog Converter* (DAC)) e enviado para o bloco RF.
5. O bloco RF desloca o espectro do sinal que vem do DAC para a frequência de transmissão (radiofrequência) com o intuito de enviar o sinal pelo ar. Geralmente esta etapa se implementa usando o conceito de *Direct-Conversion* ou *zero-IF* (SUN; XU, 2000). O *Direct-Conversion* consiste em deslocar o espectro de banda base para a frequência de transmissão e vice-versa em um único processo.

2.2.2.2 Modo Recepção

O funcionamento do sistema no modo de recepção acontece da seguinte forma:

1. O processo de recepção inicia quando o sinal RF é capturado pela antena e conduzido para o bloco RF.
2. O espectro do sinal RF recebido é deslocado para a banda base no bloco RF, o sinal analógico em banda base é digitalizado com um ADC *Analog to Digital Converter* (*Analog to Digital Converter*), em seguida diminui-se a taxa de amostragem no bloco DDC.
3. Posteriormente, os dados obtidos são enviados ao computador usando uma interface de comunicação serial.
4. No computador os dados recebidos passam por diversos processos (Demodulação, decodificação, entre outras) para recuperar o sinal enviado pelo transmissor.

2.3 Componentes de Hardware

Como se mostra na Figura 2 a tecnologia SDR usa um computador ou sistema embarcado e um periférico de *software radio*. Atualmente existem vários modelos de periféricos de *software radio* que tem uma ampla gama de frequências. A Tabela 1 mostra alguns modelos que podem ser encontrados no mercado.

Tabela 1 – Características de alguns USRP

	B100	B200	B210
Faixa do Espectro	50MHz- 2.2 GHz	70MHz - 6GHz	
Largura de Banda	16 MHz	61.44 MHz	
Duplex	Full	Full	2x2 MIMO
Numero de bits (ADC/DAC)	12/14 Bits	12 Bits	
Taxa de Amostragem (ADC/DAC)	64 Msps/ 128 Msps	61.44 Msps	
Interface (Speed)	USB 2 HS (480 Megabit)	USB 3 (5 Gigabit)	
FPGA Logic Elements	25k	75k	150k
Microcontroller	Cypress FX2	Cypress FX3	
Open Source	HDL + Code Sch.	Host Code	

2.3.1 Periférico de Rádio de Software Universal (USRP)

O USRP foi desenvolvido pelo fabricante Ettus Research (ETTUS, 2014), uma empresa pertencente à *National Instrument* (NI) desde 2010. O USRP é um dispositivo de custo acessível que possibilita a implementação de sistemas SDR em tempo real e foi desenvolvido para ser utilizado com GNU Radio mas também pode ser utilizado com outros softwares. O

USRP é uma plataforma de hardware que, junto com um computador, serve para a implementação de SDR. Eles se comunicam via cabo de rede ou *Universal Serial Bus* (USB), segundo o modelo USRP usado.

O USRP foi concebido para criar uma ligação entre o mundo dos sinais RF e o computador ou dispositivo de computação embarcada. A Ettus tem vários modelos desenvolvidos até agora, que trabalham em uma ampla gama de frequências. A Figura 3 exibe um USRP genérico. Um USRP é composto principalmente por um módulo de *Front-end RF*, conversores ADC e DAC, e módulos de DDC e DUC que estão implementados em uma *Field Programmable Gate Arrays* (FPGA).

Os modelos USRP mais integrados são: O modelo B200 e B210 que consistem em uma única placa. As características deles são apresentadas na Tabela 1. Os modelos USRP: B100, USRP1, E100, E110, N200 e N210, no entanto, apresentam dois componentes: a placa mãe ou *Motherboard* e a placa filha ou *Daughterboard*, como se mostra na Figura 3.

- **Motherboard:** A placa-mãe encarrega-se de levar os dados provenientes do computador até o bloco *front-end RF* e vice versa. Composta por um processador, conversores ADC e DAC, interfaces seriais de alta velocidade para comunicação com outras placas, recursos de relógio e uma interface de comunicação com o dispositivo de computação.

Os processadores podem ser *General Purpose Processors* (GPP), *Digital Signal Processors* (DSP), *Field Programmable Gate Arrays* (FPGA), entre outros. De modo geral, o processador implementa: o bloco DDC, o processo de decimação, o processo de interpolação e o bloco DUC como mostra-se na Figura 3.

A placa-mãe se comunica com uma placa filha, descrita a seguir.

- **Daughterboard:** Existem 13 tipos de placas-filhas disponíveis, com diferentes especificações como largura de banda, potência, fator de ruído, entre outras. Existem diversos tipos de placas-filhas, tais como transmissoras, receptoras ou transceptores de tipo homo-dino e outras heteródino. Um transceptor homo-dino é conhecido como *Zero-IF transceiver* (SUN; XU, 2000) e também é chamado de transceptor de conversão direta, porque desloca o espectro de banda base para frequência de transmissão. No entanto, um transceptor heteródino desloca o espectro de banda base para uma ou várias FI até chegar na frequência de transmissão. As placas filhas possuem diferentes faixas de frequência de acordo com o modelo.

2.4 Software Utilizado no SDR

A tecnologia SDR tem como base o uso de software, destacando-se os tipos principais: de controle e de processamento do sinal (Figura 4). O software de controle serve para configurar os parâmetros do USRP tais como ganho da antena, frequência RF desejada, taxa

de interpolação e decimação. O software de processamento do sinal, através do software de controle, modifica os parâmetros do USRP. Esses dados são passados para o processador que é o encarregado de fazer a interpolação, decimação e configurar os elementos RF. O software de processamento do sinal a ser usado vai depender do fabricante da plataforma de hardware.

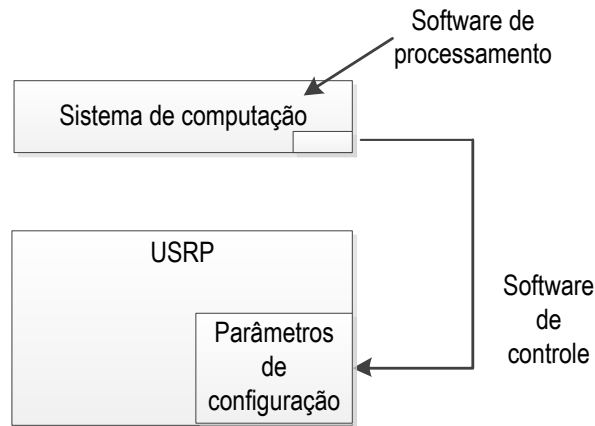


Figura 4 – Tipo de software usados no SDR.

2.4.1 Software de Controle

O USRP emprega como software de controle o *USRP Hardware Driver* (UHD), que é um driver composto por um conjunto de pacotes escritos em linguagem C++. O USRP é de natureza multi-plataforma e pode ser usado pelos sistemas operacionais Linux, Windows e Mac OS. O UHD foi concebido para ter controle sobre os produtos da *Ettus Research* e pode ser configurado de forma independente, ou usando algum dos software de processamento de sinal.

2.4.2 Software de Processamento do Sinal

Existem diversas opções de software para processamento do sinal feito para trabalhar com a tecnologia SDR. Entre as opções mais difundidas encontramos os itens a seguir.

- **GNU GNU *radio***: O projeto *General Public License* (GNU) *radio* (GNURADIO, 2015) começou em 2001. Trata-se de um software livre e de código aberto que permite criar aplicações visuais de processamento de sinais a partir da união de blocos. O GNU *radio* tem um conjunto de bibliotecas tais como: gerador de sinais, filtros, moduladores, demoduladores, entre outros. O GNU *radio* pode ser usado com um hardware externo RF para criar SDR, ou sem hardware para trabalhar em um ambiente de simulação.

Para construir um sistema de comunicações com o GNU *radio*, usam-se dois níveis de software: linguagem de programação C++ para programar cada bloco, visando um alto desempenho; e para a interconexão dos blocos usa-se um *script* escrito na linguagem de programação interpretada Python, como se apresenta na Figura 5.

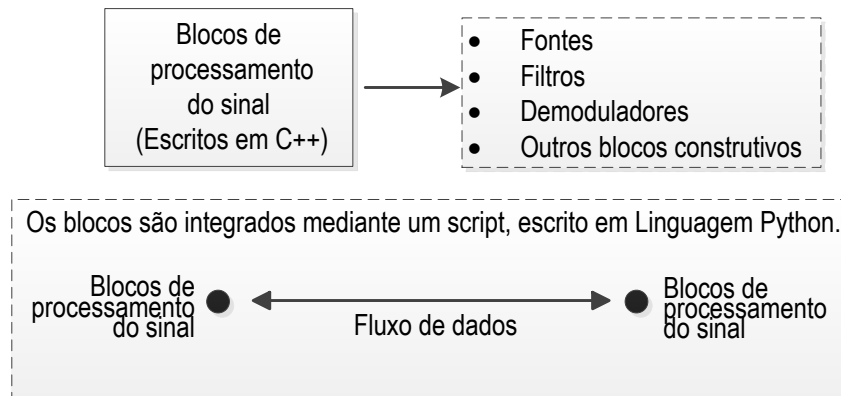


Figura 5 – Fluxo de dados em blocos de GNU Radio.

- **Matlab:** *Matrix Laboratory* (MATLAB) O Matlab é uma ferramenta de programação para fazer cálculos com matrizes. O matlab foi criado por Cleve Moler nos anos 70 na universidade de Novo México. No ano de 1984, junto com Steve Bangert e Jack Little dão vida à empresa Mathworks (MATHWORKS, 2014). O Matlab é uma linguagem de alto nível e um ambiente interativo para computação numérica, visualização e programação. Pode-se analisar dados, desenvolver algoritmos, e criar modelos e aplicações nele. Para tal intuito, pode-se programar em um script (código fonte) ou usar um ambiente visual similar com o GNU radio, chamado simulink.

O Simulink é um ambiente de diagrama de blocos para a simulação em vários domínios, que foi projetado para tratar cada bloco como um modelo. Ele suporta simulação, geração automática de código, e teste contínuo e verificação de sistemas embarcados (MATHWORKS, 2015).

O Matlab está presente no mercado há mais de 30 anos no ramo de simulação de sistemas de comunicações. Os *toolboxes* e bibliotecas de funções especializadas em comunicações digitais em Matlab têm evoluído ao decorrer do tempo.

- **Labview:** O *Laboratory Virtual Instrument Engineering Workbench Laboratory Virtual Instrument Engineering Workbench* (LABVIEW) é um ambiente gráfico de programação criado pela empresa *National Instrument*, chamado “Linguagem G”. Os programas não são escritos, mas sim desenhados. O Labview tem uma grande quantidade de blocos pré-desenhados. O Labview pode se comunicar através da interface *General Purpose Interface Bus* (GPIB), VXI, RS-232, RS-485 e cartões de aquisição de dados; além disso, ele incorpora bibliotecas para padrões de software como *Transmission Control Protocol/Internet Protocol* TCP/IP e ActiveX.

3 Sistema de Radiodifusão de Vídeo Digital por Satélite (DVB-S)

A radiodifusão de vídeo digital por satélite ou DVB-S, do inglês *Digital Video Broadcasting by Satellite*, é um padrão para transmissão de sinais de televisão digital por meio de satélite. As especificações técnicas do DVB-S encontram-se no padrão europeu EN 300 421 (ETSI, 1993), desenvolvido pelo *European Telecommunications Standards Institute* (ETSI), e contemplam tanto o transmissor quanto o receptor de um sistema DVB-S, entre outros módulos. Nesta seção, aborda-se os principais aspectos desse padrão.

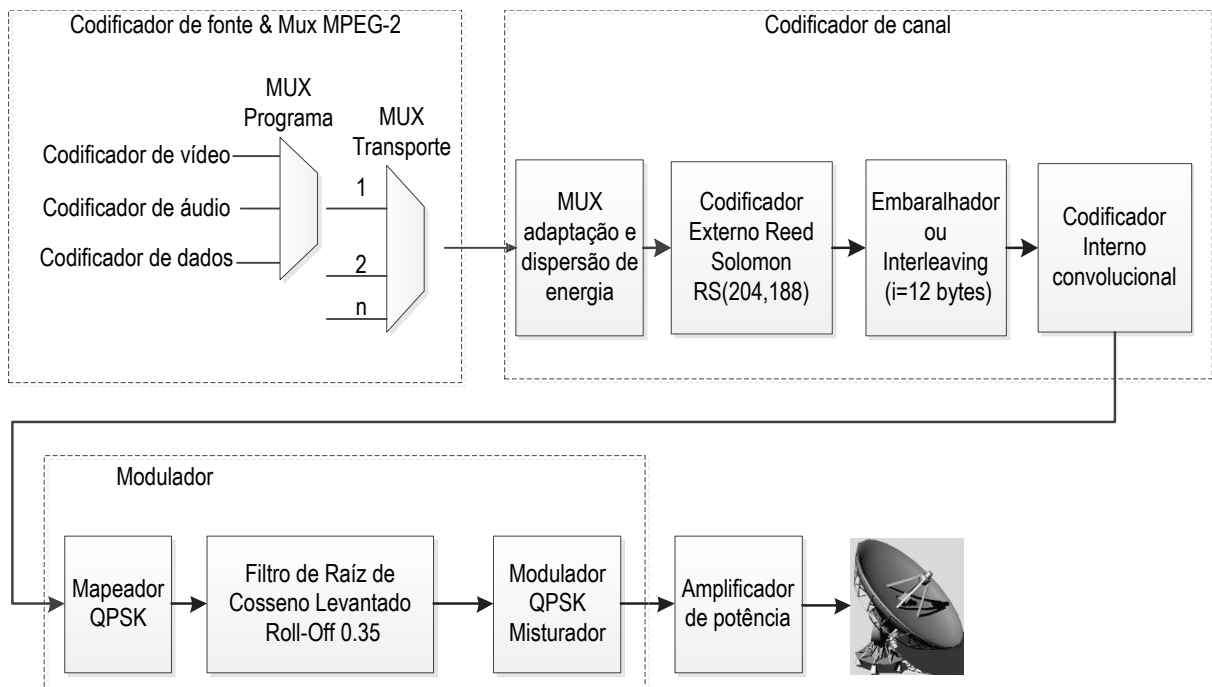


Figura 6 – Diagrama de blocos do transmissor DVB-S.

3.1 Transmissor

Os principais blocos funcionais que compõem um transmissor DVB-S são: codificador de fonte, codificador de canal (adaptação e dispersão de energia, codificador Reed-Solomon, embaralhador e codificador Viterbi) e modulador (filtro de Nyquist, modulador QPSK e etapa de amplificação de potência). A forma como cada um desses subsistemas se relaciona pode ser vista na Figura 6, que representa uma visão geral de um transmissor DVB-S.

3.1.1 Codificador de Fonte

Com o intuito de reduzir a quantidade de bits transmitidos, o padrão DVB-S emprega o codificador *Moving Picture Experts Group 2* (MPEG-2) (ISO/IEC-DIS-13818-1, June 1994) como técnica de formatação de dados. Para enviar o fluxo de bits, criou-se o conceito de *Transport Stream* (TS) ou fluxo de transporte cuja função é transportar áudio, vídeo e dados digitais através de meios ruidosos, imprevisíveis e instáveis.

O TS MPEG-2 consiste de pacotes de comprimento constante de 188 bytes, em que os primeiros 4 bytes são destinados ao cabeçalho e os 184 bytes restantes para o *payload*, que é a informação útil do pacote TS. Tais pacotes podem conter informações de áudio, de vídeo ou de dados. O cabeçalho contém vários parâmetros importantes para a transmissão dos pacotes, como pode ser visto na Figura 7. O primeiro byte do cabeçalho é o byte de sincronismo (0x47 hexadecimal). Outro componente do TS são os 13 bits de comprimento dedicados ao *Packet Identification*, (PID) cuja função é identificar sequências distintas de pacotes TS enviadas. O nono bit do cabeçalho é o indicador do erro de transporte.

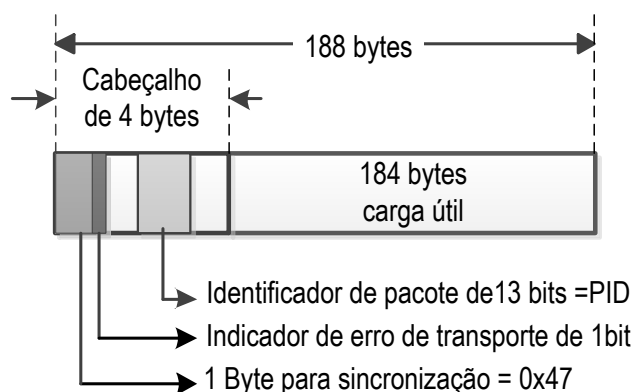


Figura 7 – Estrutura do pacote de transporte MPEG-2

3.1.2 Codificador de Canal

O intuito da codificação de canal é aumentar a robustez da comunicação através da inserção de códigos corretores e detectores de erro (MOON, 2002). No padrão DVB-S, a codificação de canal é realizada em cinco etapas:

1. **Adaptação e dispersão de energia:** Nessa etapa, todos os bytes do pacote TS, com exceção do byte de sincronismo (0x47, em hexadecimal), são transformados em uma sequência pseudoaleatória. Tal aleatorização elimina sequências longas de zeros ou uns e visa dispersar a energia do espectro irradiado. Para efetuar o processo de aleatorização, utiliza-se um gerador de sequência binária pseudoaleatória ou *Pseudo Random Binary Sequence* (PRBS) com polinômio gerador dado pela equação (3.1) e cuja implementação serial é mostrada na Figura 8. No receptor, utiliza-se o mesmo processo para realizar a desaleatorização dos dados recebidos.

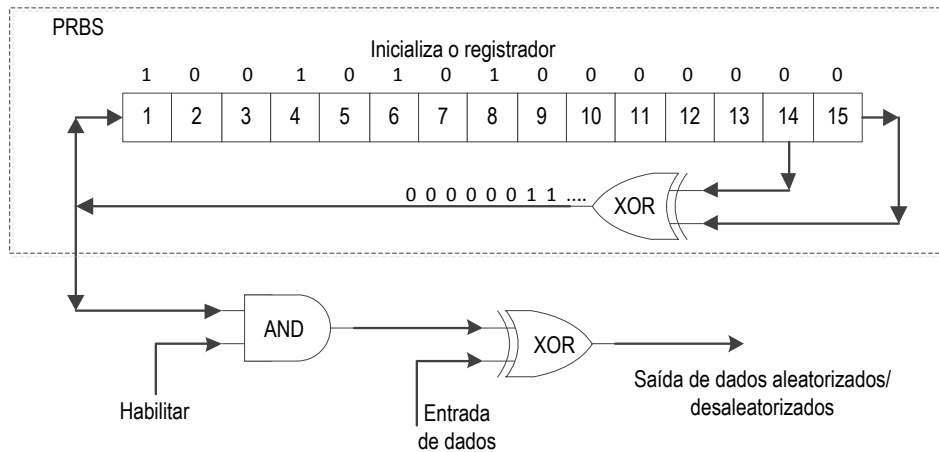


Figura 8 – Diagrama do randomizador/De-randomizador.

$$1 + x^{14} + x^{15} \tag{3.1}$$

O processo de aleatorização inicia-se com a inversão do byte de sincronismo e com a inicialização do gerador PRBS com a sequência "10010101000000". Esse processo se repete a cada oito pacotes TS, isto é, a cada oito pacotes TS há uma inversão do byte de sincronismo e o gerador PRBS é reinicializado. O intuito da inversão do byte de sincronismo é facilitar a detecção do início de um pacote TS por parte do receptor.

O PRBS deve gerar uma sequência periódica de 1503 bytes. O primeiro bit de saída do gerador PRBS opera com o nono bit do pacote TS que contém o byte de sincronismo invertido (0xB8, em hexadecimal). O último bit de saída do gerador PRBS opera com o último bit do oitavo pacote TS, sendo o byte seguinte (byte de sincronismo invertido 0xB8 hex) o início de um novo período. O PRBS gera bits de forma contínua, então, com a chegada dos bytes de sincronismo, a saída do PRBS é desabilitada para que estes não sofram alterações. A Figura 9 mostra a disposição dos 8 blocos TS na saída do PRBS.

2. **Codificador Externo ou Codificador de Reed-Solomon:** O codificador Reed-Solomon (RS) e o codificador cíclico binário (BCH) são codificadores cíclicos corretores de erros comumente utilizados. Os códigos RS baseiam-se em corpos de Galois ou campos finitos (CLARKE, 2002).

O diagrama de blocos do codificador RS é mostrado na Figura 10. O funcionamento do codificador RS será explicado através de um exemplo. Considere um codificador RS(15, 11), o que significa que a saída do codificador (sinal codificado) é de $n = 15$ bytes e a entrada do codificador é de $k = 11$ bytes (a mensagem). Portanto, os bytes de redundância são $m = n - k = 4$ e a máxima quantidade de bytes que esse codificador pode corrigir é $t = m/2 = 2$.

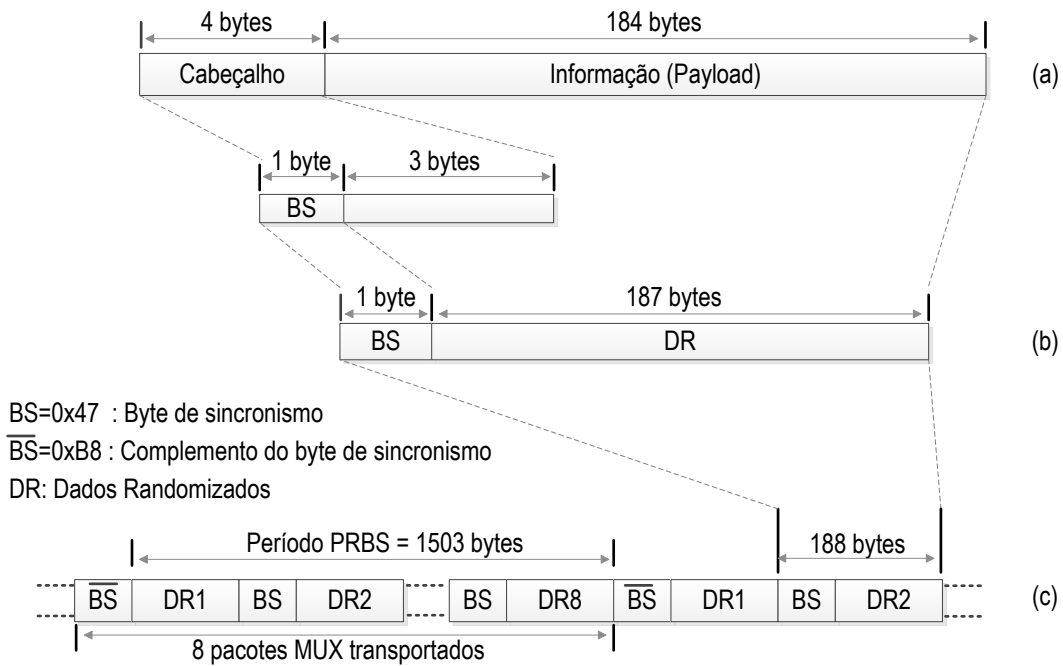


Figura 9 – Pacotes de TS mostrados na saída do PRBS.

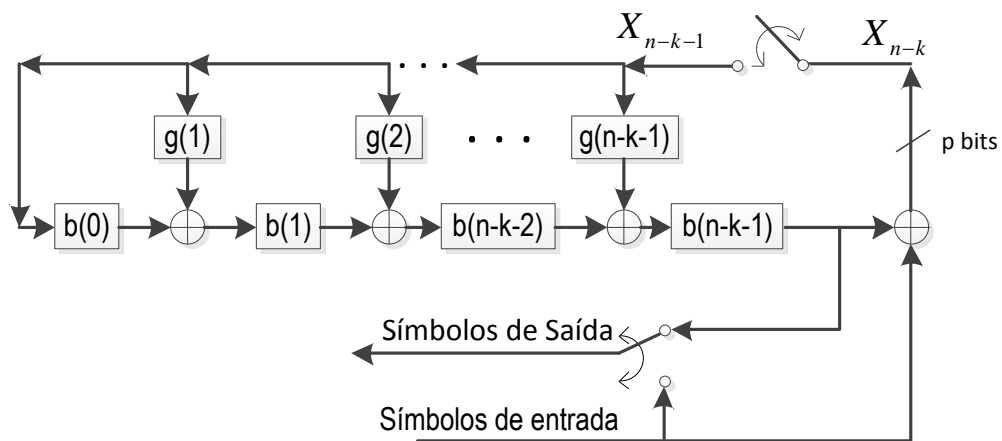


Figura 10 – Diagrama de blocos do codificador Reed-Solomon.

O primeiro passo no codificador RS consiste em encontrar o polinômio primitivo $p(x)$ de acordo com o número de bytes redundantes m . Para o exemplo em questão, o número de bits de redundância é 4 e o polinômio primitivo é dado pela equação (3.2).

$$m = 4 \Rightarrow p(x) = 1 + x + x^4 \tag{3.2}$$

O segundo passo consiste em calcular o polinômio gerador de campo $g(x)$ de acordo com o número de bytes redundantes m e o parâmetro α . Como exemplo, para $\alpha = 2$, o

polinômio $g(x)$ é dado pela equação (3.3).

$$\alpha = 2 \Rightarrow g(x) = (x+1)(x+2)(x+4)(x+8) = x^4 + 15x^3 + 3x^2 + x + 12 \quad (3.3)$$

No terceiro passo, os bytes de entrada são convertidos em polinômio mensagem, conforme a equação (3.4):

$$k[i] = [k_i = i] / i \in [1, 11] \Rightarrow k(x) = \sum_{i=1}^{11} ix^{11-i} \quad (3.4)$$

No quarto passo, aumenta-se a ordem do polinômio mensagem segundo o número de bytes redundantes m , o que dá origem ao polinômio $n(x)$:

$$m = 4 \Rightarrow n(x) = k(x) * x^4 = \sum_{i=1}^{11} ix^{11-i} \quad (3.5)$$

No quinto e último passo do codificador Reed Solomon, calcula-se os bytes redundantes, que são os coeficientes do polinômio resíduo resultante da divisão de $n(x)$ por $p(x)$.

$$n(x) = p(x)q(x) + r(x) \Rightarrow r(x) = 3x^3 + 3x^2 + 12x + 12 \quad (3.6)$$

Portanto, a mensagem codificada pelo algoritmo RS(15,11) é:

$$k[11] = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11] \Rightarrow n[15] = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 3, 3, 12, 12] \quad (3.7)$$

No padrão DVB-S, as mensagens de entrada e saída do codificador RS têm $k = 188$ e $n = 204$ bytes, respectivamente. O codificador RS adiciona $2t = 16$ bytes redundantes, portanto o receptor é capaz de detectar e corrigir até $t = 8$ bytes corrompidos. Este codificador é normalmente apresentado como RS(204,188,t=8). Assim, os pacotes passam a ter 204 bytes (no lugar de 188) o que permite corrigir até 8 bytes errados no receptor. Se houver mais erros, a proteção de erros desta etapa falha e o demodulador marca os pacotes com erros escrevendo um valor lógico alto no bit de indicação de erro presente no cabeçalho dos pacotes TS.

Erros do tipo *burst* (conjunto de erros sobre bytes consecutivos) podem ocorrer com frequência durante uma transmissão. Caso haja mais de 8 bytes errados em um único pacote protegido pela codificação Reed-Solomon, o bloco de proteção de erro falhará. Para diminuir e possivelmente evitar esse tipo de erro, os dados (consequentemente os possíveis erros em *burst*) são embaralhados e distribuídos sobre um conjunto de pacotes de fluxo de transporte. O bloco responsável por esse embaralhamento é denominado *interleaver*.

3. **Embaralhamento Externo ou *Interleaving***: Este procedimento aumenta a eficiência do codificador RS na correção de erros. O *interleaving* permuta a posição dos bytes de uma sequência de símbolos como mostrado na Figura 11 (MOON, 2002). Com isso, envia-se uma versão embaralhada dos bytes do pacote TS e caso ocorram mais de 8 erros consecutivos (*burst*), o receptor ainda poderá recuperar os dados originais (AMADO, 2015).

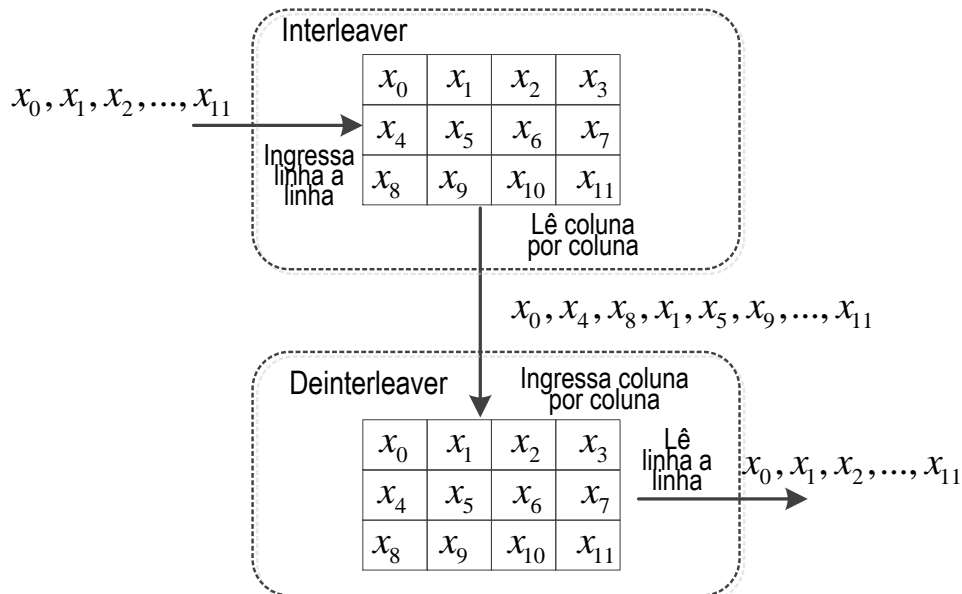


Figura 11 – Um *interleaver* e um *deinterleaver* de 3x4 (MOON, 2002).

Um exemplo deste processo é apresentado na Figura 12 onde, dada uma sequência de entrada, o *interleaver* encarrega-se de embaralhar essa sequência e os dados produzidos são enviados pelo canal de comunicação. No receptor do exemplo da Figura 12, a parte dos dados recebidos que é corrompida é representada por *xxxx*. Como os dados estavam embaralhados, o *Deinterleaver* reposicionará os símbolos em sua posição original e os erros ficarão isolados, facilitando, assim, a sua detecção e correção por meio de outros módulos do sistema, como o bloco Reed-Solomon.

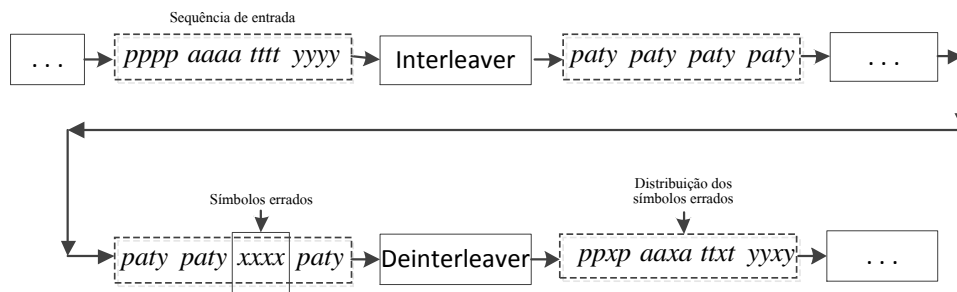


Figura 12 – Exemplo do interleaving.

No DVB-S, o *interleaving* é feito usando o *Forney Interleaver* que é mostrado na Figura 13. O *Forney Interleaver* é composto por duas chaves roteadoras e um conjunto de registradores de deslocamento. Essa estrutura garante que os dados serão embaralhados

e distribuídos aleatoriamente. O *interleaving* máximo acontece sobre 11 pacotes TS do fluxo de transporte. A posição do byte de sincronismo não é alterada pelo interleaver, portanto sua posição original dentro do pacote de transporte é mantida. Outra característica do *interleaving* utilizado no DVB-S é que a velocidade de rotação das chaves corresponde a um múltiplo exato do comprimento do pacote e que o interleaver e de-interleaver são síncronos em relação ao fluxo de transporte MPEG-2 (FISCHER, 2010).

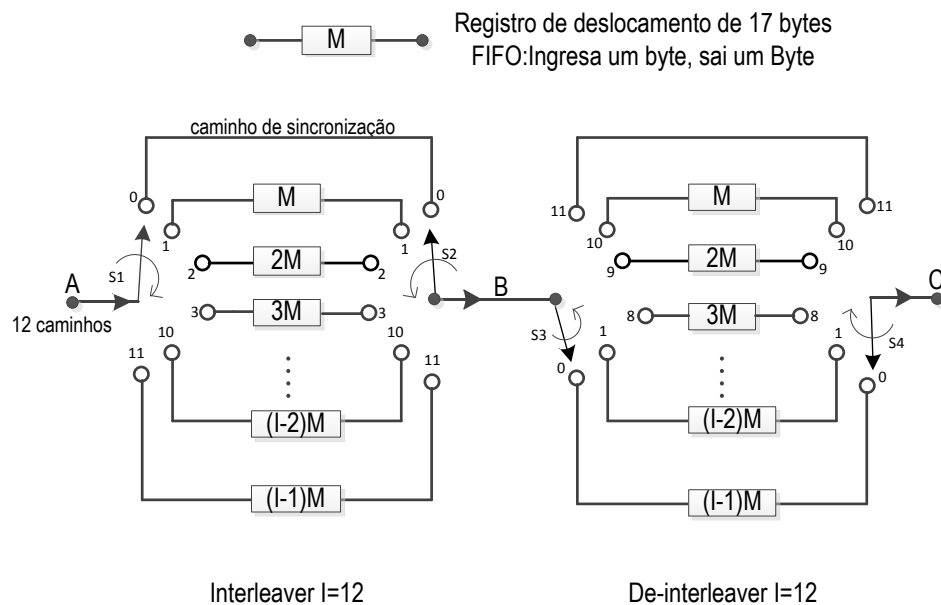


Figura 13 – Diagrama de funcionamento *interleaver* e *deinterleaver*, adaptado de (FISCHER, 2010).

4. **Codificador Interno ou Codificador Convolutacional:** O codificador convolutacional é uma estrutura matemática empregada para correção de erros em tempo real. Os bits codificados dependem tanto dos bits atuais quanto dos bits anteriores.

O codificador convolutacional consiste em um registrador de deslocamento de seis estados e dois caminhos pelos quais o sinal de entrada é misturado com o conteúdo do registrador de deslocamento como se observa na Figura 14. O fluxo de dados de entrada é dividido em três. Os dados primeiramente passam no registrador de deslocamento onde influenciam o fluxo de dados superior e inferior do codificador convolutacional mediante o uso de uma operação XOR que opera durante 6 ciclos de relógio. Este processo permite dispersar a informação de 1 bit sobre 6 bits. Em determinados pontos do codificador, como no ramo de dados superior e no ramo de dados inferior, há portas XOR que misturam o fluxo de dados com o conteúdo do registrador de deslocamento. Assim, na saída do codificador convolutacional, há dois fluxos de dados ($C1, C2$) com a mesma taxa de dados do sinal de entrada. A taxa total dos dados de saída é, então, o dobro a taxa dos dados de entrada. Com isso, um *overhead* de 100% é adicionado aos dados.

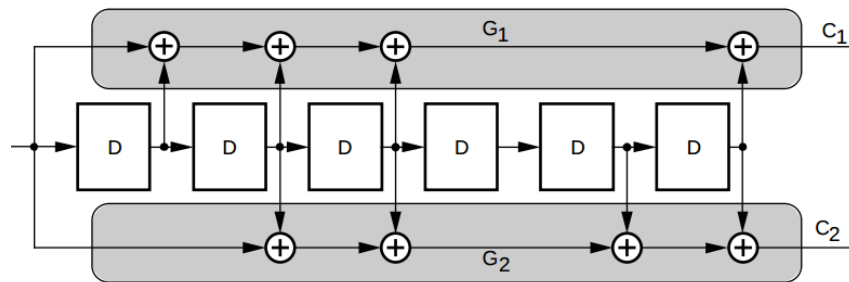


Figura 14 – Diagrama de bloco do codificador interno (CCSDS, 2012).

5. **Puncturing:** O puncionamento (do inglês *puncturing*) é a ultima etapa do codificador de canal.

Segundo o padrão DVB-S (ETSI, 1993), utiliza-se um código convolucional com a comprimento de $K = 7$ e uma taxa $1/2$. A taxa de código pode ser incrementada utilizando um padrão de perfuração sendo possível aumentar a eficiência da largura de banda. Na Figura 15 pode-se observar o processo de perfuração que consiste na remoção de alguns bits codificados antes da transmissão, o aumentando a taxa de código com a diminuição a largura de banda do código original, mas com perda do rendimento do corretor de erros. (CCSDS, 2012).

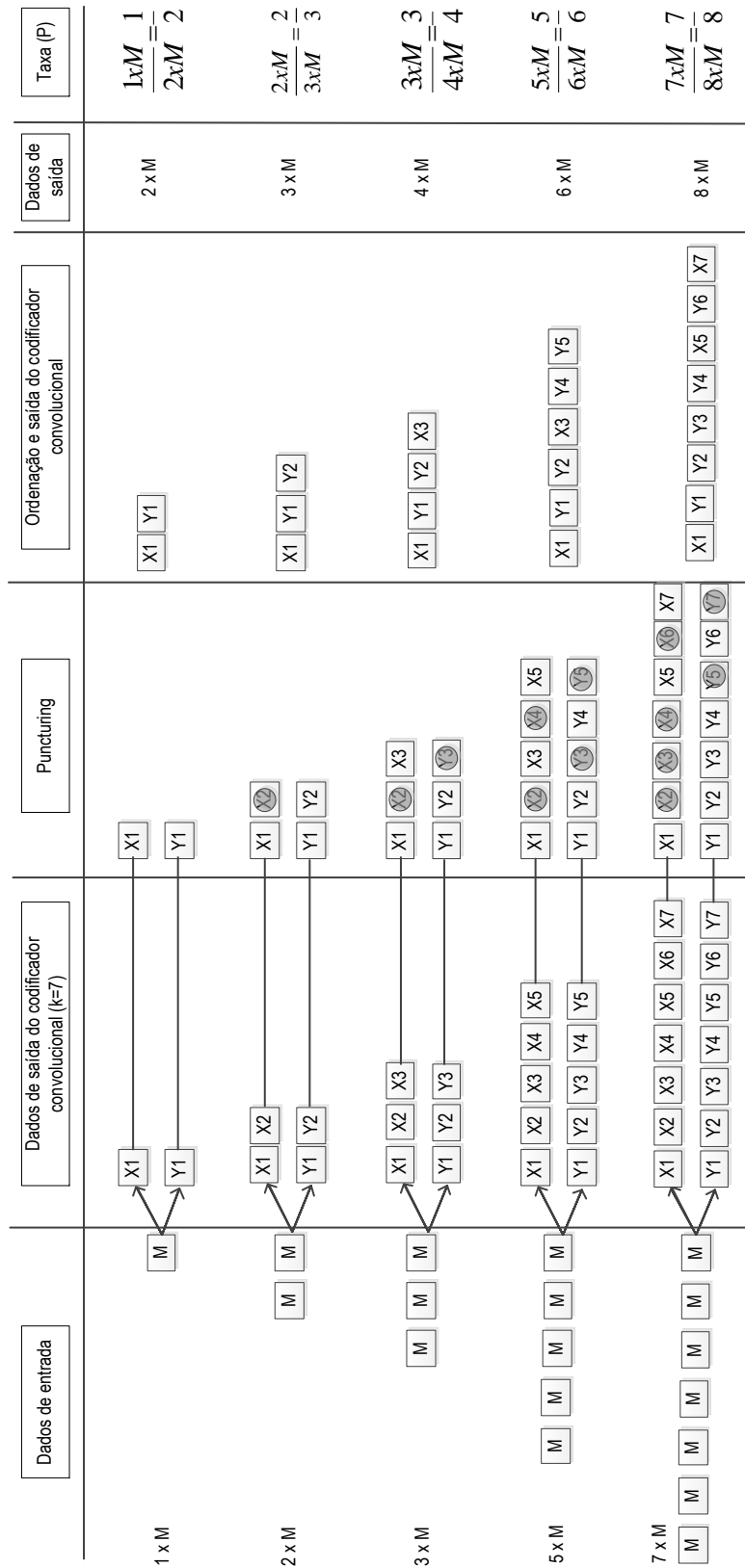


Figura 15 – Funcionamento do *puncturing* para todas as taxas

3.1.3 Modulação por Chaveamento Quadrifásico (QPSK)

No chaveamento quadrifásico (QPSK), a informação portada pelo sinal transmitido está contida na fase. Para nosso caso, a fase da portadora θ_n assume um dentre quatro valores igualmente espaçados segundo a equação (3.8) onde $n = 1, 2, 3, 4$ como é mostrado na Tabela 2.

$$\theta_n = (2n - 1)\pi/4 \quad (3.8)$$

O sinal transmitido é definido conforme a equação abaixo, onde E_s é a energia por símbolo e T_s é a duração do símbolo, e a frequência da portadora f_c é igual a n_c/T_s para algum número inteiro fixo n_c diferente de zero

$$s_n(t) = \begin{cases} \sqrt{\frac{2E_s}{T_s}} \cos(2\pi f_c t + \theta_n), & 0 \leq t \leq T_s \\ 0, & \text{caso contrário.} \end{cases} \quad (3.9)$$

Segundo a propriedade trigonométrica do coseno da soma de dois argumentos, a equação (3.9) pode ser reescrita de acordo com a equação (3.10), onde $\phi_1(t)$ e $\phi_2(t)$ são os portadores em quadratura e d_{In} e d_{Qn} representam os dados em fase e quadratura respectivamente. Cabe destacar que d_{In} e d_{Qn} geram a constelação QPSK (MOON, 2002).

$$s_n(t) = \underbrace{\sqrt{E_s} \cos(\theta_n)}_{d_{In}} \underbrace{\sqrt{\frac{2}{T_s}} \cos(2\pi f_c t)}_{\phi_1(t)} - \underbrace{\sqrt{E_s} \sin(\theta_n)}_{d_{Qn}} \underbrace{\sqrt{\frac{2}{T_s}} \sin(2\pi f_c t)}_{\phi_2(t)} \quad (3.10)$$

A Tabela 2 apresenta θ_n, d_{In}, d_{Qn} , código Gray para diferentes valores de n . O código Gray representa os *dibit* associados a cada fase.

Símbolo n	θ_n	d_{In} (Em fase)	d_{Qn} (Em Quadratura)	Código Gray
1	$\pi/4$	$\sqrt{E_s/2}$	$\sqrt{E_s/2}$	00
2	$3\pi/4$	$-\sqrt{E_s/2}$	$\sqrt{E_s/2}$	01
3	$5\pi/4$	$-\sqrt{E_s/2}$	$-\sqrt{E_s/2}$	11
4	$7\pi/4$	$\sqrt{E_s/2}$	$-\sqrt{E_s/2}$	10

Tabela 2 – Relação de fases e sinais QPSK

Um canal em que $\phi_1(t)$ é usado como um sinal portador é geralmente chamado de um canal em fase, ou I, e um canal em que $\phi_2(t)$ é usado como um sinal portador é geralmente chamado de canal em quadratura, ou Q. Portanto, d_{In} e d_{Qn} são os dados em fase e quadratura, respectivamente (KAZAZ *et al.*, 2013).

Esquemas de modulação que usam I e Q são chamados de esquemas de modulação em quadratura. A configuração básica é mostrada na Figura 16.

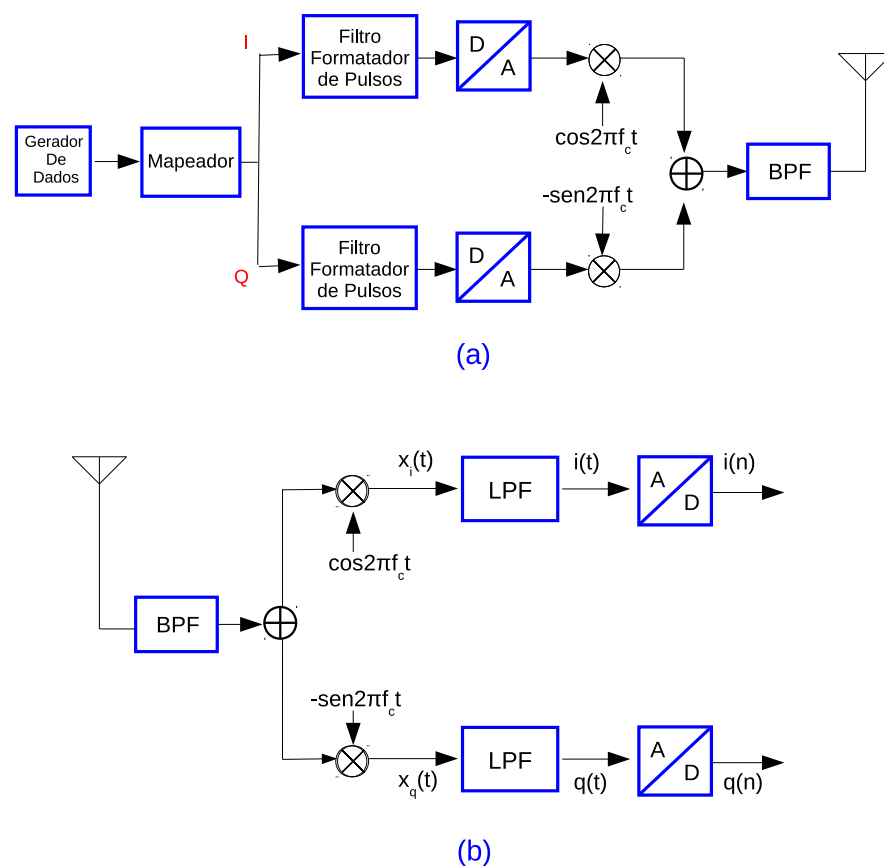


Figura 16 – Configuração básica de um esquema de modulação em quadratura.

No transmissor, os bits de informação são mapeados em símbolos QPSK pelo bloco mapeador. Esses símbolos são números complexos e sua parte real e imaginária correspondem, respectivamente, aos dados em fase, I, e quadratura, Q. Os dados em I são filtrados por um filtro formatador de pulso e então este pulso formatado é convertido em um sinal analógico por um DAC e multiplicado por um sinal portador $\cos(2\pi f_c t)$. Os dados em Q também são filtrados usando um filtro formatador de pulsos cuja saída é convertida em um sinal analógico por um DAC para, em seguida, ser multiplicada por um sinal portador $-\sin(2\pi f_c t)$. Por fim, os sinais I e Q são misturados e transmitidos.

No receptor, o sinal recebido primeiro passa por um *Bandpass Filter* (BPF) para em seguida ser transformado em um sinal banda-base através de um processo de demodulação. Para isso, multiplica-se o sinal separadamente com $\cos(2\pi f_c t)$ e $-\sin(2\pi f_c t)$ gerando sinais em fase e quadratura, respectivamente. Estes sinais passam por um *Low-pass Filter* (LPF), transformando-se em sinal em banda-base, depois é amostrado pelo ADC gerando dados em fase e quadratura.

3.2 Receptor

Nesta seção, descreveremos os principais blocos que compõem um receptor DVB-S. Note que o padrão EN 300 421 descreve, em termos gerais, o sistema para a transmissão de TV digital via satélite, mas a parte do receptor é deixada relativamente livre para acomodar diferentes soluções de implementação (ETSI, 1993). Pode-se considerar a parte da recepção como o processo inverso da transmissão, mas com algumas considerações importantes no demodulador, decodificador de canal e decodificador de fonte (ver Figura 1). Assim, cada bloco no transmissor tem seu bloco correspondente no receptor.

A Figura 17 mostra o diagrama de blocos funcional do sistema de recepção DVB-S. A entrada deste subsistema vem do canal do satélite RF e a saída correspondente é o dado útil. O processo de sincronismo de frequência e fase da portadora é uma parte própria do receptor. Este processo de sincronismo é de relevância vital no processo de recepção e o seu estudo é detalhado neste capítulo.

A seguir, detalharemos o demodulador QPSK e o decodificador do canal que formam parte do receptor conforme mostrado na Figura 1.

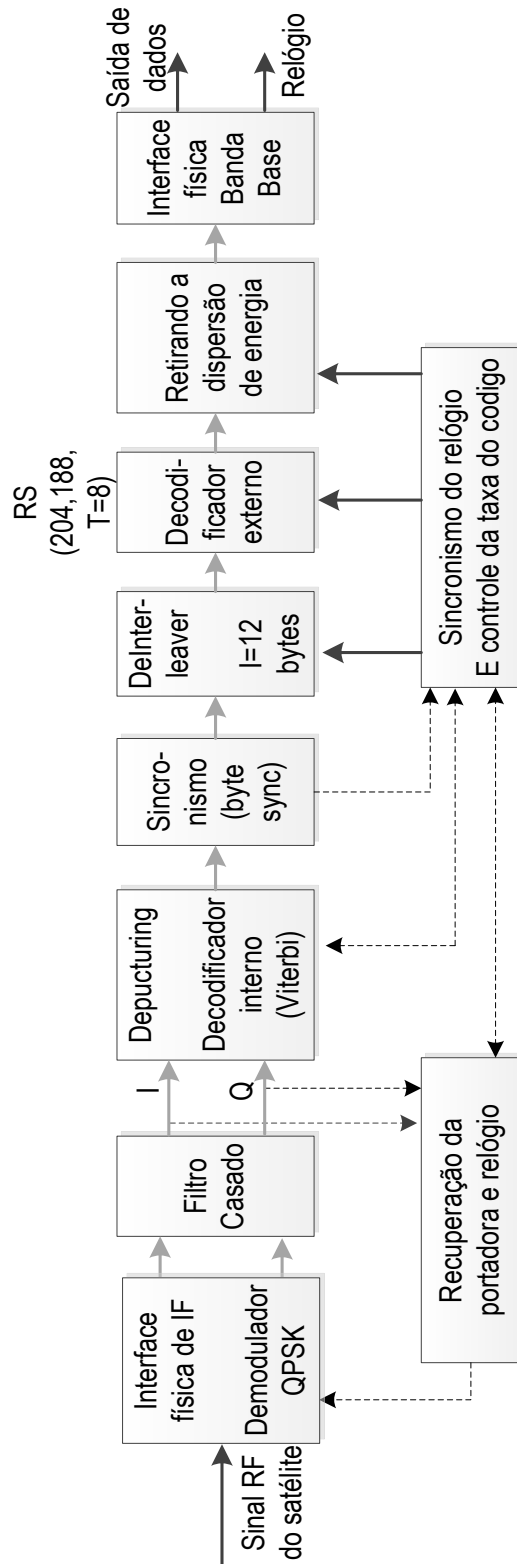


Figura 17 – Diagrama de blocos funcional do sistema de recepção DVB-S.

3.2.1 Demodulador QPSK

Um das primeiras tarefas do receptor são realizar a filtragem e demodulação do sinal recebido. O sinal recebido deve ser filtrado (por um filtro casado) para eliminar ruído e possíveis interferências fora da banda de interesse. Para minimizar os efeitos de ISI, o filtro casado deve obedecer ao critério de Nyquist. Após a demodulação e a amostragem, o sinal recebido, antes real, é representado por uma sequência complexa cuja parte real e imaginária estão associadas, respectivamente, aos dados em fase e quadratura que foram mapeadas pelo receptor em uma constelação QPSK. Cabe ao demodulador mapear os símbolos QPSK em bits.

3.2.2 Decodificador de Canal

No decodificar de canal realiza-se as seguintes operações:

- **De-puncturing:** O *De-puncturing* é o processo de adequar a taxa de bits do sinal recebido à taxa de bits que o decodificador de Viterbi precisa para realizar a decodificação convolucional.
- **Decodificador interno ou decodificador de Viterbi :** O decodificador de Viterbi é um método de correção de erros altamente eficiente. Mesmo quando não consegue corrigir os erros, o decodificador de Viterbi os minimiza, deixando assim a tarefa mais leve para o decodificador RS (VITERBI, 1967; FORNEY G., 1973).
- **De-embaralhamento externo o *Deinterleaver*:** O *deinterleaver* é o complemento do *interleaver*. O *deinterleaver* permuta o conjunto de bytes para obter a ordem original dos símbolos enviados no transmissor, como mostrado na Figura 11. Se existirem erros consecutivos na transmissão dos símbolos, estes erros serão distribuídos como pode ser observado na Figura 12.

Qualquer erro *burst* que tenha acontecido na transmissão é dividido no receptor no processo de *de-interleaving* conforme apresentado na Figura 18. Assim o erro é distribuído sobre um número de pacotes de transporte.

Assim, facilita-se a correção de um erro do tipo *burst*, que afeta a vários bytes consecutivos, quando esses erros consecutivos são distribuídos em bits de erro não consecutivos e isolados ao longo de vários pacotes de transporte.

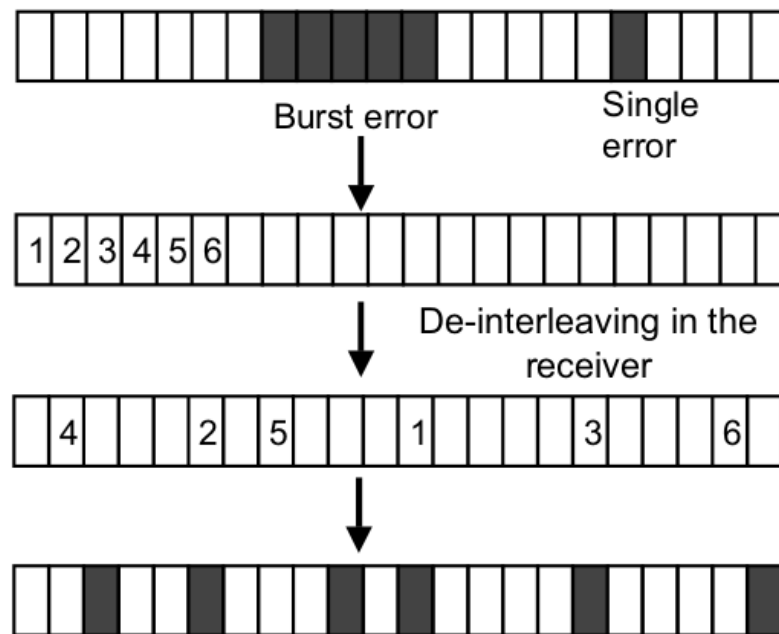


Figura 18 – De-interleaving (FISCHER, 2010)

- **Decodificador exterior ou decodificador RS** : O Decodificador RS completa a tarefa de correção de erros de um receptor DVB-S.

O diagrama de blocos do decodificador RS é mostrado na Figura 19.

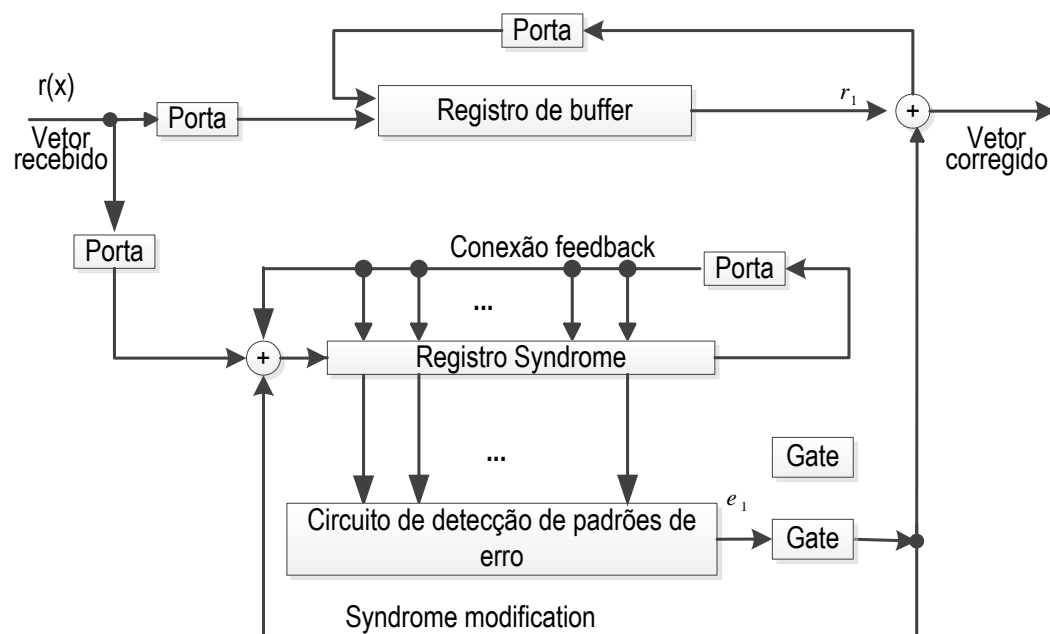


Figura 19 – Diagrama de blocos do decodificador Reed-Solomon.

Note que com uma entrada de 204 bytes e fazendo uso dos bytes de redundância, pode-se corrigir até 8 bytes consecutivos.

4 Sincronismo do Sistema

No presente capítulo são apresentadas as técnicas empregadas para a correta sincronização entre o transmissor e o receptor de um sistema de comunicação, requisito indispensável para a recuperação confiável da informação transmitida.

Como foi mencionado no capítulo anterior, o padrão DVB-S estabelece as características do transmissor relacionadas à codificação de fonte e de canal, ao esquema de modulação digital, etc. O receptor, por sua vez, deve desfazer cada uma destas etapas. No entanto, para que todo esse processamento reproduza a informação digital transmitida com um nível de confiabilidade aceitável é necessário sincronizar a operação do transmissor e do receptor com relação sua frequência, fase, e instantes de amostragem. As seguintes seções apresentam as técnicas utilizadas para sincronizar cada um destes parâmetros.

4.1 Malha de Captura de Fase (PLL) em Tempo Discreto

A malha de captura de fase (do inglês: *Phase-Locked Loop*, PLL) é um sistema de controle que produz um sinal de saída cuja fase é síncrona à fase do sinal de entrada. Basicamente, um PLL consiste de um oscilador de frequência variável e um detector de fase. O oscilador gera um sinal periódico de referência, enquanto o detector de fase compara a fase desse sinal de referência com a fase do sinal periódico de entrada e gera um sinal de erro que ajusta o oscilador para manter as fases em correspondência. Esse ajuste é feito continuamente através de um sistema de malha fechada.

Os PLLs são muito utilizados em rádio, computadores e outras aplicações eletrônicas. Dentre as diversas aplicações do PLL, pode-se citar: demodulação analógica, síntese de frequência, remoção de *jitter* de pulsos de relógio em circuitos lógicos, etc. Os PLLs podem ser de dois tipos: analógicos e discretos. Na discussão que se segue, nos limitaremos aos PLLs em tempo discreto.

A estrutura de um PLL em tempo discreto é essencialmente a mesma que a de um PLL em tempo contínuo. Um exemplo da arquitetura de um sistema básico é mostrado na Figura 20. Na malha de realimentação ou *feedback* há um detector de fase em tempo discreto, um filtro em tempo discreto e um sintetizador digital DDS (do inglês *direct digital synthesizer*) (RICE, 2009). O sinal de entrada de um PLL é formado por amostras senoidais com frequência Ω_0 radianos/segundo e fase $\theta(nT)$, onde $e(nT)$ é o erro de fase em tempo discreto, $g(\cdot)$ é uma função da diferença de fases entre o sinal de entrada do PLL $x_o(nT)$ e o sinal de saída do DDS $y_o(nT)$. Este erro é filtrado por um filtro digital e logo em seguida entra no DDS. O DDS é uma versão em tempo discreto do oscilador controlado por tensão (VCO) e seu funcionamento é

detalhado a seguir. A relação entre a entrada e a saída do bloco DDS é descrita na equação (4.1).

$$y_o(nT) = \cos(\Omega_o n + \hat{\theta}(nT)) \quad (4.1)$$

onde

$$\hat{\theta}(nT) = K_0 \sum_{k=-\infty}^{n-1} v(kT). \quad (4.2)$$

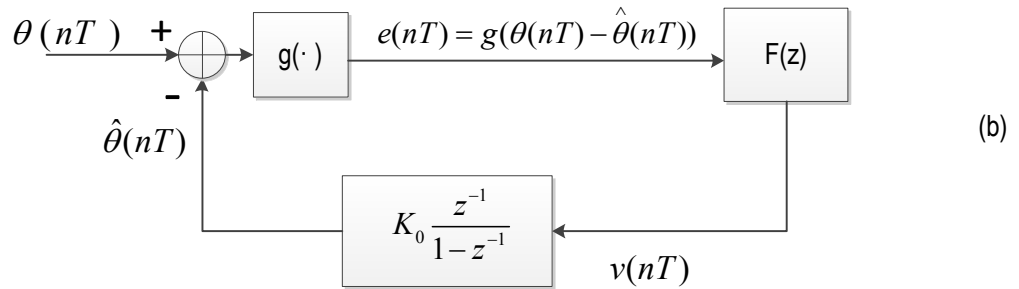
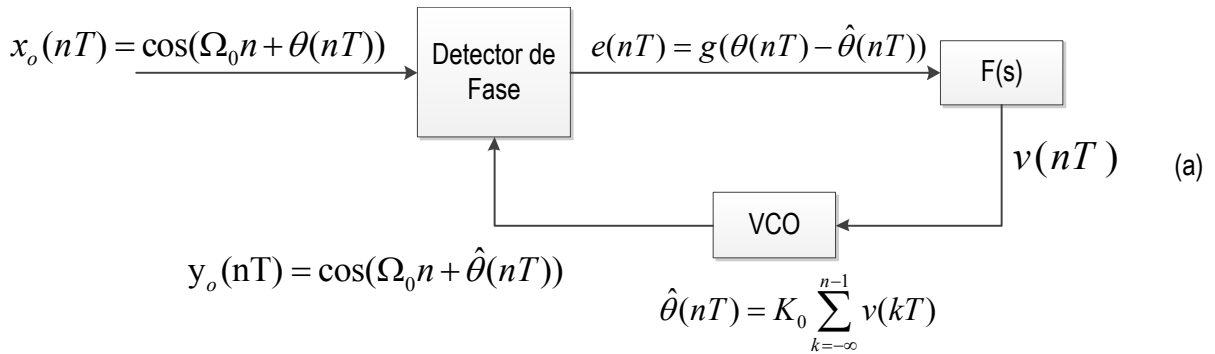


Figura 20 – Estrutura básica de um PLL em tempo discreto (a) e seu correspondente PLL equivalente de fase (b) (RICE, 2009).

Neste trabalho é utilizada uma aproximação híbrida para o filtro *proportional-plus-integrator*. Inicialmente mostramos o PLL em tempo contínuo e seu laço de fase equivalente. Estes laços são mostrados na Figura 21 e seu correspondente em tempo discreto é mostrado na Figura 22.

Na Figura 22 (a) a parte integral do filtro usa um filtro simples com um pólo em $z = 1$. O PLL em tempo discreto usa um sintetizador digital direto (DDS) ao invés de um oscilador controlado por tensão (VCO). Para calcular a fase instantânea, utiliza-se um filtro de polo único para integrar a entrada do DDS. O laço em tempo discreto equivalente é mostrado na Figura 22 (b).

A função de transferência do laço para os casos contínuos e discretos das figuras 21 e 22, respectivamente, são:

$$H_a(s) = \frac{2\zeta \omega_n s + \omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2} \quad (4.3)$$

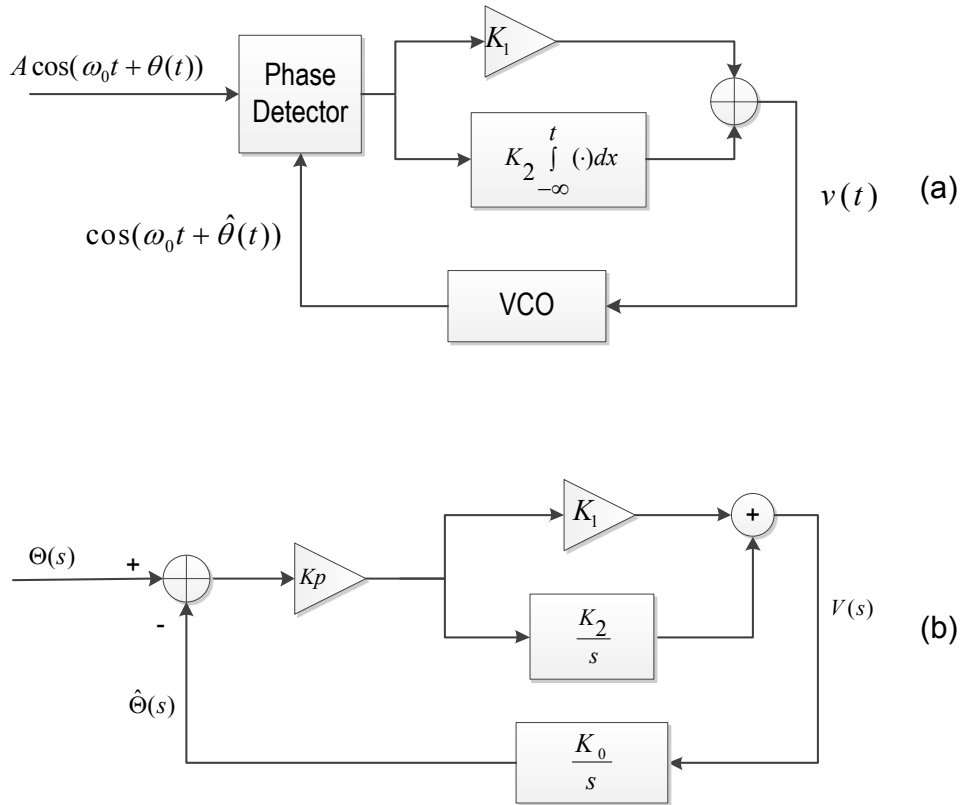


Figura 21 – Laço de captura de fase em tempo contínuo de segunda ordem com filtro *proportional-plus-integrator* (a) e PLL equivalente de fase linearizada (b).

Onde ω_n é a frequência natural e ζ é o fator de amortecimento.

$$H_d(z) = \frac{K_p K_0 (K_1 + K_2) z^{-1} - K_p K_0 K_1 z^{-2}}{1 - 2 \left(1 - \frac{1}{2} K_p K_0 (K_1 + K_2)\right) z^{-1} + (1 - K_p K_0 K_1) z^{-2}} \quad (4.4)$$

O fator de amortecimento ζ e a frequência natural ω_n são obtidos de:

$$\zeta = \frac{k_1}{2} \sqrt{\frac{k_0 k_p}{k_2}}, \quad \omega_n = \sqrt{k_0 k_p k_2} \quad (4.5)$$

A relação destas duas equações é dada pela transformação bilinear ou de Tustin:

$$\frac{1}{s} \rightarrow \frac{T}{2} \frac{1 + z^{-1}}{1 - z^{-1}} \quad (4.6)$$

Resolvendo o sistema de equações, obtemos os seguintes resultados para as constantes da função de transferência em tempo discreto:

$$K_p K_0 K_1 = \frac{4\zeta \theta_n}{1 + 2\zeta \theta_n + \theta_n^2} \quad (4.7)$$

$$K_p K_0 K_2 = \frac{4\theta_n^2}{1 + 2\zeta \theta_n + \theta_n^2} \quad (4.8)$$

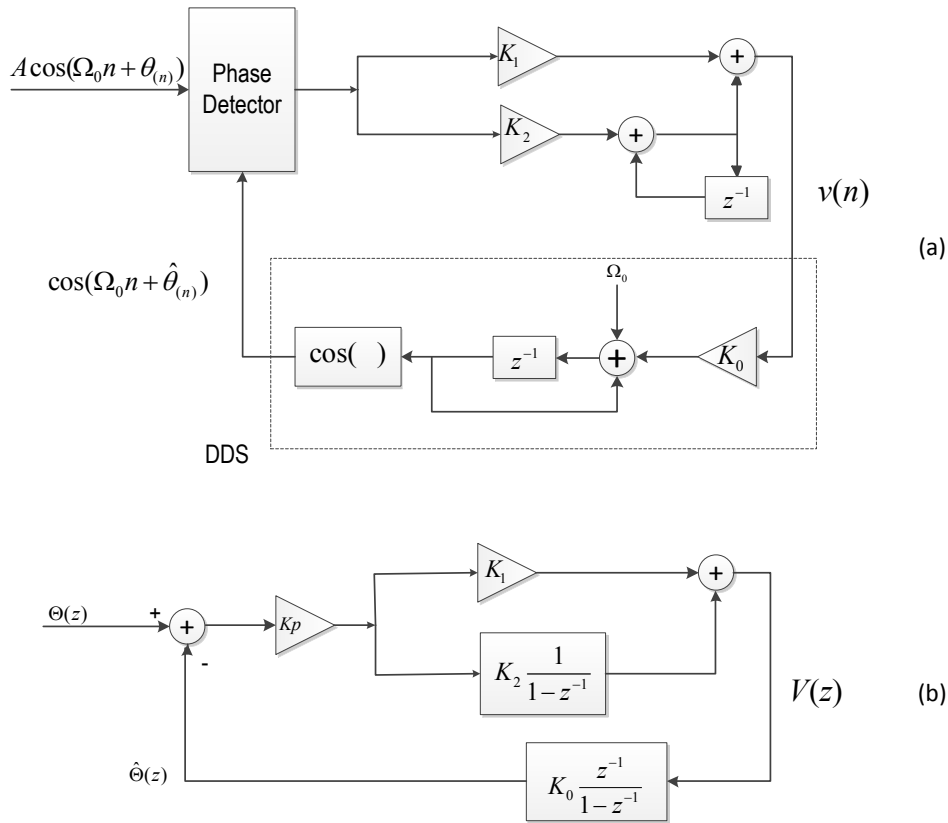


Figura 22 – Laço de fase bloqueada em tempo discreto de segunda ordem com filtro *proportional-plus-integrator* (a) e PLL equivalente de fase linearizada em tempo discreto (b).

Agora, sabendo-se que

$$\theta_n = \frac{B_n T}{\zeta + \frac{1}{4\zeta}} \quad (4.9)$$

e substituindo-se esta relação em 4.7 e 4.8 e assumindo-se que $B_n T \ll 1$, obtém-se os seguintes resultados aproximados:

$$K_p K_0 K_1 \approx \frac{4\zeta}{\zeta + \frac{1}{4\zeta}} (B_n T) \quad (4.10)$$

$$K_p K_0 K_2 \approx \frac{4}{\left(\zeta + \frac{1}{4\zeta}\right)^2} (B_n T)^2 \quad (4.11)$$

Este procedimento de projeto requer que a largura de banda seja especificada à taxa de $1/T$. No campo das comunicações digitais, é uma prática comum especificar a largura de banda do ruído com a taxa $1/T_s$. Então podemos usar $N = T_s/T$ e dizer que:

$$\theta_n = \frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta}\right)} \quad (4.12)$$

Então, fazendo as respectivas substituições em (4.7) e (4.8) obtemos:

$$K_p K_0 K_1 = \frac{4\zeta \left(\frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta} \right)} \right)}{1 + 2\zeta \left(\frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta} \right)} \right) + \left(\frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta} \right)} \right)^2} \quad (4.13)$$

$$K_p K_0 K_2 = \frac{4 \left(\frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta} \right)} \right)^2}{1 + 2\zeta \left(\frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta} \right)} \right) + \left(\frac{B_n T_s}{N \left(\zeta + \frac{1}{4\zeta} \right)} \right)^2} \quad (4.14)$$

4.2 Recuperação da Portadora

No transmissor, o sinal de informação em banda base é modulado em uma portadora de alta frequência f_c , o que acarreta um deslocamento de seu espectro ao redor da frequência da portadora como é mostrado na Figura 23 (a) e (b). No receptor, esse processo é revertido empregando um oscilador local similar ao utilizado na transmissão.

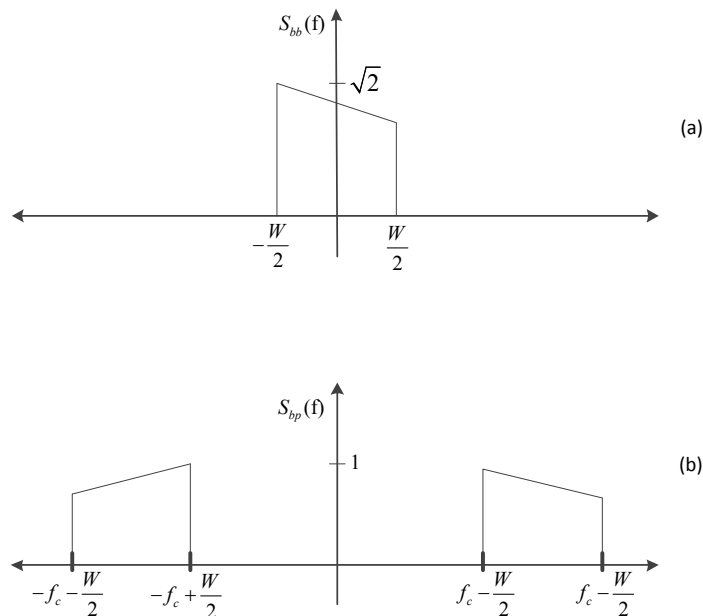


Figura 23 – No domínio da frequência o sinal de banda base (a) é deslocado ao redor frequência da portadora gerando o sinal de banda passante (b)

A Figura 24 mostra um sistema ideal de comunicações passabanda onde as portadoras do transmissor e receptor são coerentes. Em contraposição, a Figura 25 apresenta um sistema não coerente de comunicações que expressa uma melhor aproximação com a realidade. Nestas Figuras, d_I e d_Q representam respectivamente os dados em fase e quadratura que geram

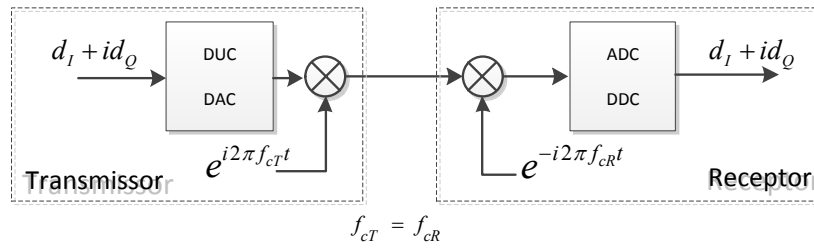


Figura 24 – Sistema de comunicações passabanda ideal.

a constelação QPSK (conforme introduzido na seção 3.1.3). Os parâmetros f_{cT} e θ_T representam respectivamente a frequência e fase da portadora no transmissor, enquanto que f_{cR} e θ_R desenvolvem o mesmo papel no receptor.

A referência utilizada para gerar a frequência da portadora no transmissor e receptor é baseada em cristais de quartzo bastante precisos. Contudo, é comum que exista uma diferença entre a fase e frequência da portadora no transmissor e receptor como se mostra na Figura 25. Os desvios de fase ($\Delta\theta$) e de frequência (Δf_c) do sinal recebido são gerados pela passagem do sinal através do canal de transmissão e pela não linearidade dos circuitos envolvidos no transmissor e receptor. O efeito que esta diferença de frequência produz nos símbolos complexos transmitidos é um giro contínuo na constelação recebida. Por outro lado, uma diferença de fase constante produz uma rotação constante na constelação do alfabeto de símbolos. Por último a representação da constelação QPSK bem dado por: $d'_I + id'_Q$ como é mostrado na Figura 25.

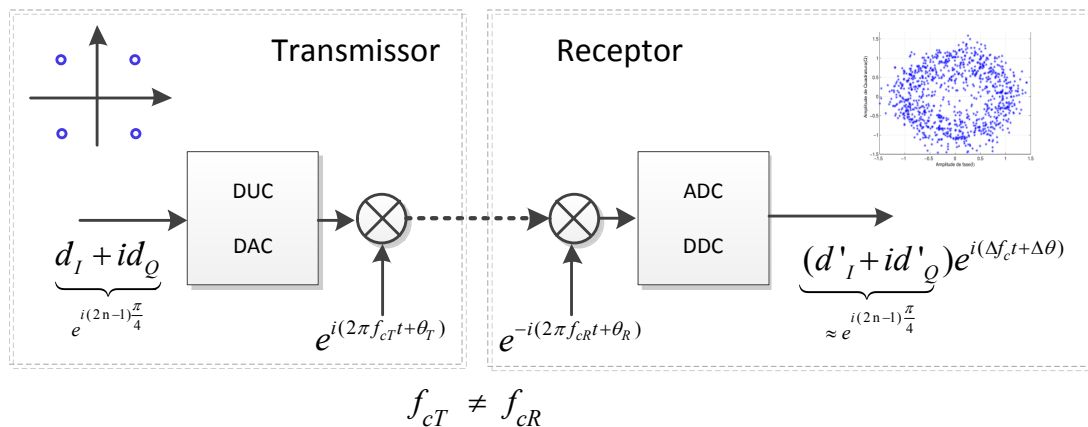


Figura 25 – Sistema de comunicações passabanda mais realista, adaptado de (EVERETT, 2013)

A Figura 26 apresenta o processo de sincronismo e as suas etapas. No receptor, o periférico SDR em uso entrega um sinal QPSK em banda base. Este sinal passa por um controle de ganho automático (AGC), e posteriormente passa por um filtro de recepção. Na saída do filtro de recepção, o sinal é definido pela equação 4.15. O intuito do bloco de sincronismo de frequência é eliminar o giro da constelação QPSK (MATHWORKS, 2015).

$$s_{QPSK} = e^{i(2n-1)\frac{\pi}{4}} e^{i(2\pi\Delta f_c t + \Delta\theta)}; n = 1, 2, 3, 4 \quad (4.15)$$

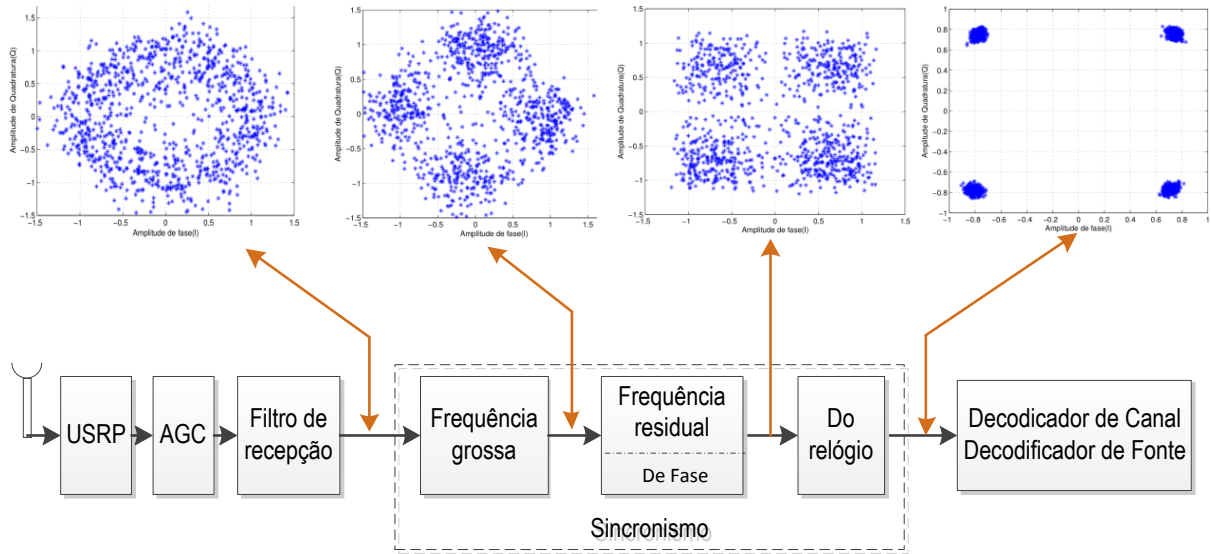


Figura 26 – Etapas do processo sincronismo.

O sincronismo da portadora deve se realizar a nível de frequência e fase. O sincronismo de frequência da portadora consiste em duas partes: a primeira é chamada sincronismo de frequência grossa, enquanto que a segunda denomina-se sincronismo de frequência residual. Ambas partes serão detalhadas nas subseções 4.2.1 e 4.2.2 respectivamente. Cabe destacar que a segunda etapa do sincronismo de frequência da seção 4.2.2 (sincronismo de frequência residual) desenvolve também o processo de sincronismo de fase da portadora.

O desvio da frequência Δf_c é igual à soma do desvio grosseiro em frequência Δf_g e do desvio em frequência residual Δf_r , como mostrado na equação (4.16)

$$\begin{aligned} \Delta f_c &= \Delta f_g + \Delta f_r \\ \Delta f_g &> \Delta f_r \end{aligned} \tag{4.16}$$

4.2.1 Sincronismo Grosseiro em Frequência

O bloco de sincronismo grosseiro em frequência faz uma estimativa para obter o desvio da frequência Δf_g , mediante o seguinte processo: o sinal de entrada s_{QPSK} é elevado à quarta potência com o intuito de obter a equação (4.17), que não é dependente da modulação QPSK. Uma vez retirada a dependência QPSK, pode-se calcular o deslocamento grosseiro da frequência Δf_g . Com tal intuito, aplica-se FFT no sinal independente de modulação para estimar Δf_g .

$$\begin{aligned} s^4_{QPSK} &= \left[e^{i(2n-1)\frac{\pi}{4}} e^{i(\Delta f_c t + \Delta \theta)} \right]^4 \\ e^{2n\pi i} &= 1; \Delta \theta' = \Delta \theta - \pi \\ s^4_{QPSK} &= e^{2n\pi i} e^{4(\Delta \theta' + \Delta f_c)t} \\ s^4_{QPSK} &= e^{(4\Delta \theta' + 4\Delta f_c)t} \end{aligned} \tag{4.17}$$

Após encontrar o valor de Δf_g , subtrai-se Δf_g do sinal de entrada. Logo, a saída do bloco de estimativa do desvio em frequência é $(d'_I + id'_Q)e^{i(\Delta f_r t + \Delta \theta)}$.

4.2.2 Sincronismo de Frequência Residual e da Fase da Portadora

O erro de fase da portadora causa uma rotação na projeção do espaço de sinais. Se essa rotação ultrapassar a região de decisão permitida, haverá erro na detecção dos símbolos. É importante destacar que, havendo erro de fase, uma decisão equivocada pode acontecer mesmo que haja um perfeito sincronismo entre os relógios do transmissor e do receptor e mesmo que não haja ruído aditivo no canal de comunicação.

Para corrigir este erro de fase, deve-se realizar uma estimativa de fase da portadora. Tal estimativa pode ser feita usando o módulo de sincronismo de fase, que é o circuito responsável por sincronizar a fase do transmissor e receptor (RICE, 2009).

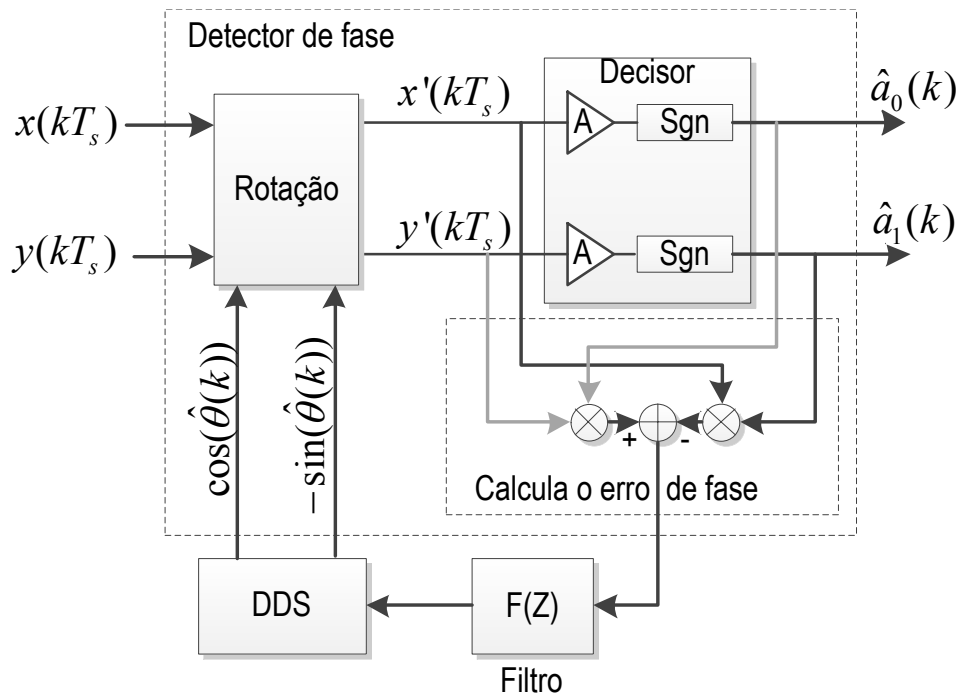


Figura 27 – Funcionamento do sincronismo de fase

O processo de sincronismo de fase para uma modulação QPSK, mostrado na Figura 27, é composto por 3 etapas:

1. **Detector de fase** : É um projeto do detector de fase de máxima verossimilhança *Maximum Likelihood* (ML). A detecção inicia com a rotação da constelação de dados de entrada $(x(kT_s), y(kT_s))$ por um ângulo obtido através da estimativa do valor de deslocamento da

fase da portadora ($-\hat{\theta}_{(k)}$) conforme descrito na equação (4.18).

$$\begin{bmatrix} x'_{(kT_s)} \\ y'_{(kT_s)} \end{bmatrix} = \begin{bmatrix} \cos(\hat{\theta}_{(k)}) & \sin(\hat{\theta}_{(k)}) \\ -\sin(\hat{\theta}_{(k)}) & \cos(\hat{\theta}_{(k)}) \end{bmatrix} \begin{bmatrix} x_{(kT_s)} \\ y_{(kT_s)} \end{bmatrix} \quad (4.18)$$

Para a modulação QPSK, o decisor opera segundo as equações (4.19) e (4.20)

$$\hat{a}_{0(k)} = A \times \text{sgn} \left\{ x'_{(kT_s)} \right\} \quad (4.19)$$

$$\hat{a}_{1(k)} = A \times \text{sgn} \left\{ y'_{(kT_s)} \right\} \quad (4.20)$$

Para detectar o erro de fase, usa-se o laço Costas (WAL; MONTREUIL, 1995), que é uma versão de decisão direita do detector de erro de fase ML. Esse algoritmo é dado pela equação (4.21)

$$e_{(k)} = y'_{(kT_s)} \hat{a}_{0(k)} - x'_{(kT_s)} \hat{a}_{1(k)} \quad (4.21)$$

2. **Filtro** : O filtro de primeira ordem de tipo passa baixa é descrito na sua forma contínua na equação (4.22), e mostrada em sua forma discreta na Figura 22.

$$F(s) = k_1 + \frac{k_2}{s} \quad (4.22)$$

Esse filtro remove todos os sinais de interferência que ocorreram na detecção da fase, além disso, proporciona uma certa memória quando se perde o sincronismo devido à presença de um transitório interferente. Ao se reduzir a largura de banda do filtro, melhora-se a rejeição para os sinais fora da banda, mas ao mesmo tempo, reduz-se a faixa de captura e incrementa-se o tempo de captura.

3. **DDS** : É a versão digital do *Voltage Controlled Oscillator* (VCO), que gera as senóides em quadratura.

4.2.3 Sincronismo do Relógio

O tempo correto de amostragem no receptor é o problema que abordaremos nessa seção. O sincronismo de relógio ou recuperação do relógio encarrega-se da correção do erro de temporização no sinal recebido, determinando assim o instante ótimo de tempo de amostragem. Na Figura 28, mostra-se um diagrama de olho com um deslocamento de $\hat{\tau}$ segundos em relação ao instante ótimo de amostragem. Cabe ao circuito de sincronismo de relógio derivar os instantes corretos de amostragem. O processo de sincronismo de relógio pode ser decomposto em quatro partes:

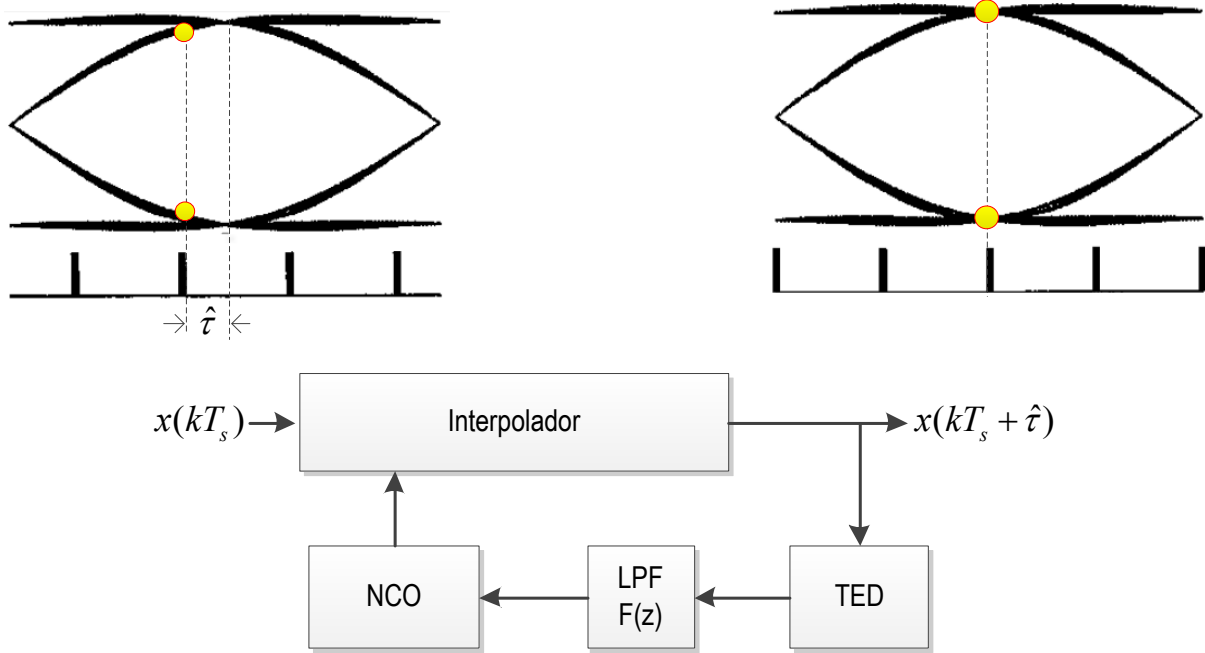


Figura 28 – Funcionamento do sincronismo do relógio (RICE, 2009)

1. **Detector de Erro de Tempo:** O Detector de Erro de Tempo, *Timing Error Detector* (TED), calcula os erros gerados no sinal recebido no decorrer do tempo. O TED gera um sinal de erro por cada símbolo recebido baseado na estimativa de tempo atual. Assim, o sinal de erro de tempo discreto é atualizado conforme chegam novos símbolos. Existem vários tipos de TED, como: de máxima verossimilhança, adiantado-atrasado, de cruzamento de zero (ZCTED), de Gardner, e de *Mueller-Muller*. Neste trabalho usaremos o ZCTED.

O *Zero-Crossing TED* (ZCTED) é baseado no processo de encontrar os cruzamentos de zeros no diagrama de olho. O ZCTED opera a cada duas amostras/símbolo e fornece um erro nulo quando todas as outras amostras estão alinhadas no tempo com um cruzamento de zero (*zero crossing*) na saída do filtro adaptado.

2. **Interpolação:** O diagrama apresentado na Figura 29 ilustra a interpolação. Neste diagrama, as amostras encontram-se espaçadas por um período T .

A expressão fundamental de interpolação é mostrada na equação 4.23. A equação mostra que o sinal interpolado pode ser obtido como uma soma ponderada das amostras disponíveis.

$$x(kT_I) = \sum_i x((m(k) - i)T)h_I((i + u(k))T) \tag{4.23}$$

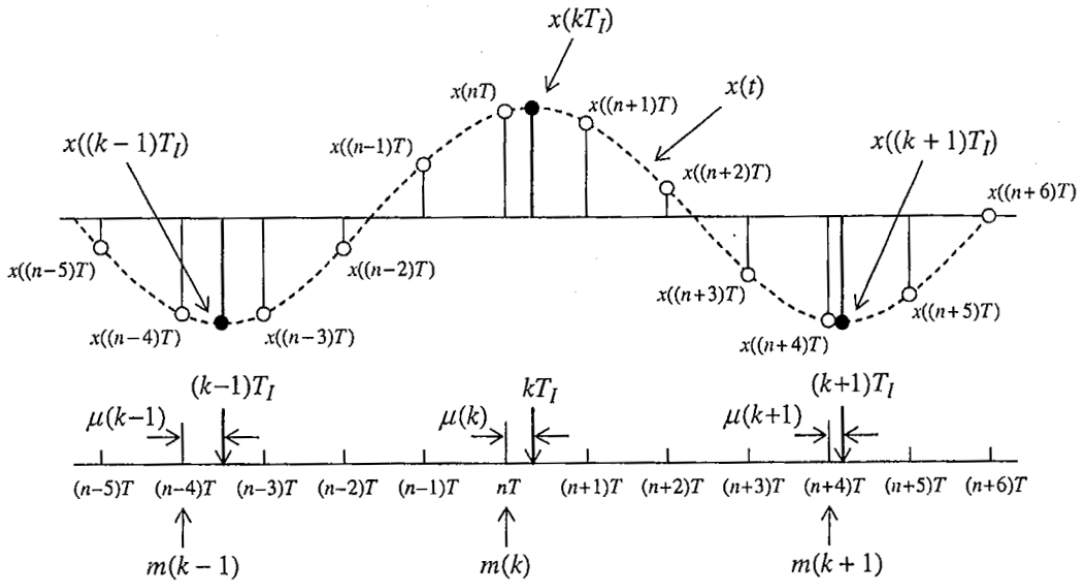


Figura 29 – Interpolação (RICE, 2009)

3. **Filtro:** O filtro passa-baixas possui resposta ao impulso dada por (4.24):

$$h_I(z) = k_1 + k_2 \frac{1}{1 - z_1} \quad (4.24)$$

A principal tarefa desse filtro é minimizar variações bruscas no sinal de erro.

4. **Controle do interpolador:**

O objetivo do bloco de controle do interpolador é fornecer ao interpolador o índice de pontos base $m(k)$ e o intervalo fracionário $u(k)$.

Quando os interpolantes são requeridos a cada N amostras, o controle de interpolação pode ser obtido usando um registrador da primeira ordem projetado para gerar um "underflow" a cada N amostras, onde os "underflows" são alinhados com os índices de pontos de base. Um diagrama de blocos deste método é mostrado na Figura 31. Em geral, o valor de contador satisfaz a recursão:

$$\eta(n + 1) = \eta(n) - W(n) : \text{modulo } 1 \quad (4.25)$$

Incorporando a redução módulo 1:

$$\eta(m(k) + 1) = 1 + \eta(m(k)) - W(m(k)) \quad (4.26)$$

Com isso, obtemos a seguinte relação:

$$\frac{\mu(m(k))}{\eta(m(k))} = \frac{1 - \mu(m(k))}{1 - \eta(m(k) + 1)} \quad (4.27)$$

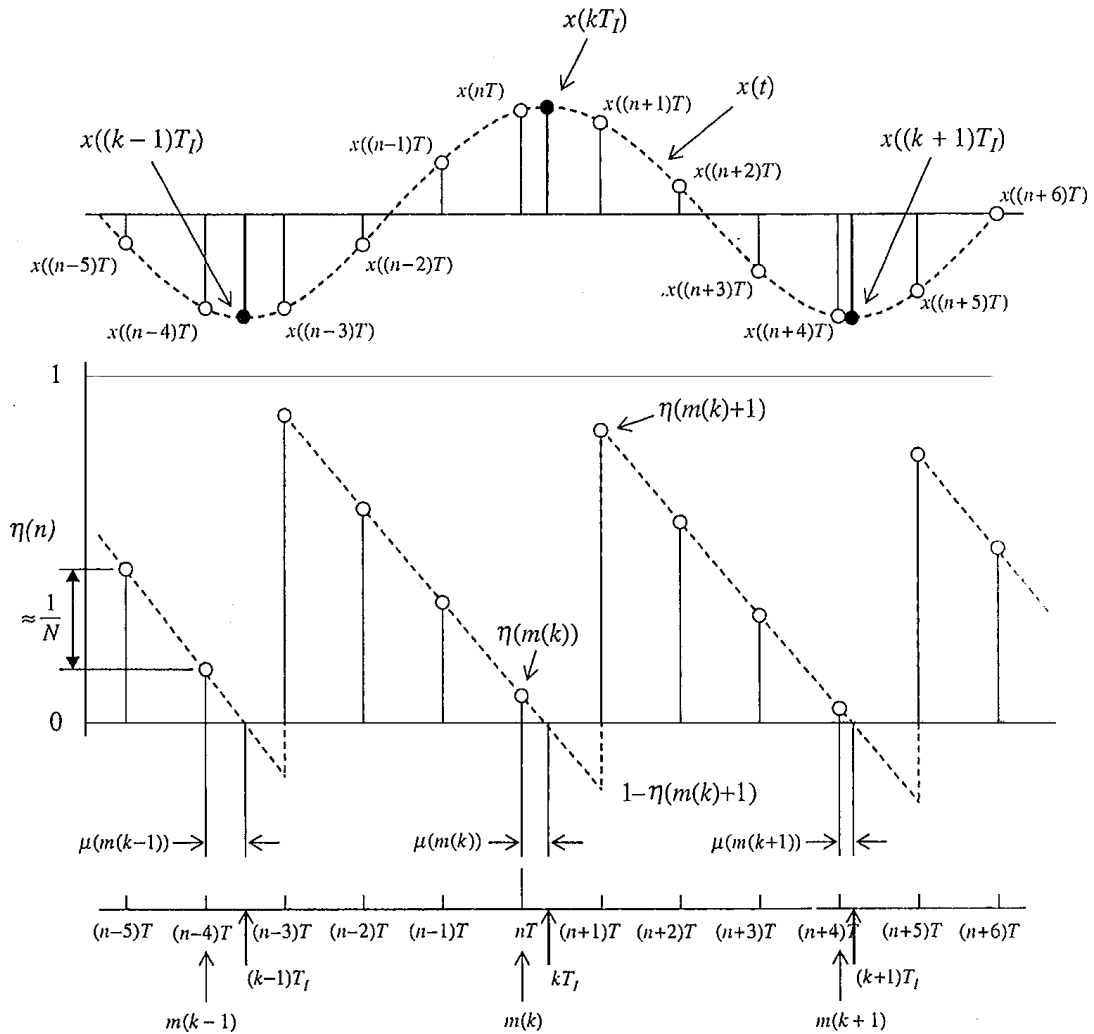


Figura 31 – Relação entre as amostras disponíveis, as amostras geradas da interpoladas e o conteúdo do contador de módulo-1 (RICE, 2009)

nessa posição no fluxo de dados recebidos. Esta técnica de correlação com limiar é mostrada na Figura 32. A palavra SW é gerada localmente no receptor e correlacionada com os dados recebidos. Quando os dados recebidos deslizam sobre a SW, um pico maior que o limiar é gerado o que indica o início do quadro.

A equação (4.30) mostrar a função de vizinhança usada na técnica de correlação.

$$L(\mu) = \sum_{i=0}^{L-1} y_{i+\mu} S_i \quad (4.30)$$

Na equação anterior, $y_{i+\mu}$ é o dado recebido, que é igual a $x_i + n_i$ onde x_i é o bit transmitido e n_i é o ruído aditivo, e S_i é a SW gerada localmente. Existem duas possíveis configurações desta correlação: rígida ou suave (do inglês *soft* ou *hard*) dependendo dos níveis de quantização dos símbolos do receptor. Se os símbolos recebidos são quantizados para mais de dois níveis, então deve-se utilizar uma correlação do tipo suave ou *soft*. Se os símbolos do receptor assumem apenas dois níveis, como os valores 1 ou -1 , então a correlação deve ser do tipo rígida ou *hard*. A correlação dura tem melhor performance que a correlação suave especialmente quando se tem

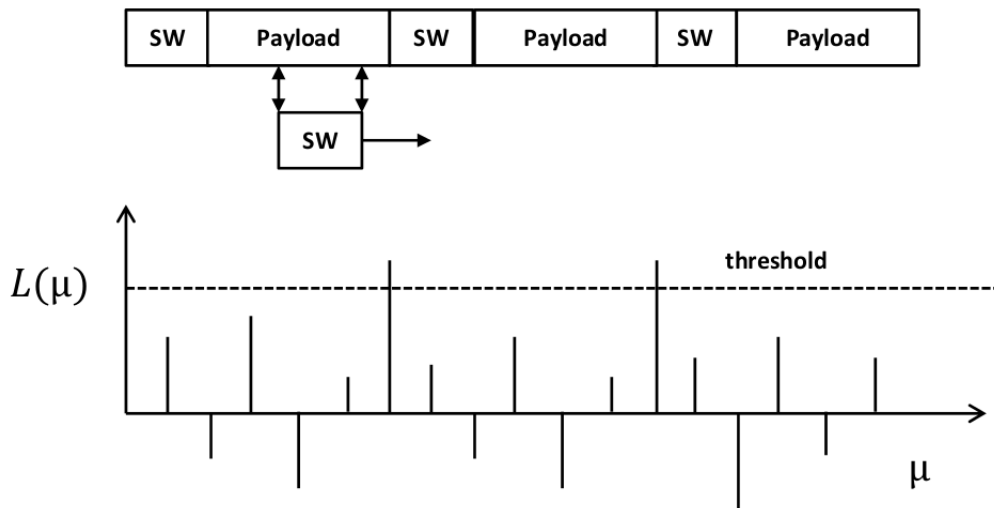


Figura 32 – Correlação da palavra de sincronismo (SW) com a data recebida.

elevados valores de SNR. Esta técnica de correlação com limiar é simples e muito usada para o sincronismo de quadro. Uma desvantagem observável experimentalmente, mas que também é fácil de deduzir analiticamente, é que este método tem alto custo computacional, como pode ser visto pelo número de operações de multiplicação no somatório da equação (4.30). Nota-se que este processo não precisa ser executado continuamente e só é necessário ou quando os dados ainda não estão sincronizados ou quando acontece alguma perda de sincronismo.

5 Implementação do Sistema e Resultados

Neste capítulo são apresentados os equipamentos, o desenvolvimento do sistema propriamente dito, e os resultados.

5.1 Equipamentos Empregados na Implementação

A seguir apresentam-se o *hardware* e o *software* mais destacados empregados nesta implementação.

- **Hardware:**

- Ettus USRP N210.
- Computador Intel Core 2 Duo.
- Placa de rede Ethernet Gigabit.
- WBX-FE Placa filha (50 MHz-2,2 GHz).
- Antena Log Periodica (850 MHz-6,5 GHz).

- **Software:**

- Linux OS: Ubuntu 12.04 LTS.
- Matlab R2013b.

USRP N210

O USRP utilizado neste trabalho é o modelo N210 da empresa *Ettus Research LLC*. O USRP N210 é um dispositivo capaz de receber e transmitir sinais de radio usando um computador com placa de rede Ethernet Gigabit. Várias placas filhas podem ser conectadas nele para usar o dispositivo em diferentes bandas de frequência. O *design* completo do hardware é *open source*, incluindo as placas filhas. O dispositivo pode ser sintonizado em diferentes frequências de portadora e taxas de amostragem. Ele desempenha o processo de deslocamento do sinal da frequência de portadora especificada para banda base. O sinal é então amostrado à taxa de amostragem especificada e as amostras são transmitidas sobre a interface Ethernet Gigabit até um computador para posterior processamento.

As principais especificações deste modelo são apresentadas na Tabela 3.

Nas Figuras 33 e 38 mostra-se a disposição dos componentes de um USRP em uma aplicação SDR.

Parâmetro	Detalhe
Taxa de amostragem ADC	100 MSPS
Resolução ADC	14 Bits
Taxa de amostragem DAC	400 MSPS
Resolução DAC	16 Bits
FPGA	Spartan 3A-DSP 3400
Interface	Gigabit Ethernet
Slots para placas-filhas	1
MIMO	Si

Tabela 3 – Especificações do USRP N210.

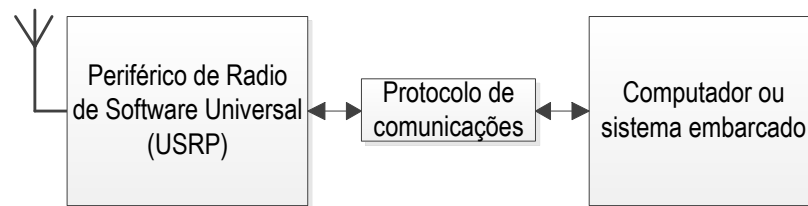


Figura 33 – Periférico USRP-N210 da Ettus (ETTUS, 2014).

O N210 pode ser usado junto com Matlab. Para mais informações e instruções de instalação, veja (ETTUS, 2014).

5.2 Implementação

Conforme mencionado, neste trabalho implementou-se um Sistema de Comunicação usando SDR com intuito de auxiliar no ensino das telecomunicações.

O sistema foi desenvolvido por completo em Matlab. Para o desenvolvimento de diversas funcionalidades, usou-se o ambiente de trabalho Simulink que faz parte do Matlab. Outros blocos foram criados mediante a implementação adequada. O Simulink administra a utilização de funcionalidades por meio de blocos. Cada bloco corresponde a um modelo matemático implementado. A maior parte dos blocos foi extraída dos blocos Simulink disponíveis para realizar algumas das etapas vistas nos capítulos anteriores. Cabe destacar que o Simulink possui um amplo conjunto de blocos com diversas funcionalidades já implementadas disponível para uso.

5.3 Testes

Para avaliar o desempenho do sistema, testaram-se dois cenários. O primeiro foi desenvolvido em um ambiente controlado (simulação dentro de um computador) e o segundo foi realizado usando o periférico USRP-N210.

Para o ambiente controlado, usaram-se os seguintes parâmetros:

- **BER:** Representa a taxa de erro de bit (do inglês *Bit Error Rate*). O BER é uma métrica para avaliar um sistema de comunicações que indica a quantidade de bits errados dividido pelo número total de bits recebidos.
- **E_b:** Energia de bit.
- **N₀:** Densidade de ruído espectral.
- **E_b/N₀:** Relação entre E_b e N₀. Com frequência usa-se esta relação para avaliar o impacto da energia do bit na presença de ruído. A unidade de medida desta relação é o dB.

Nas seções a seguir, serão apresentados os resultados de diferentes testes realizados relacionando a BER e a razão E_b/N₀.

5.3.1 Cenário Controlado no Computador

Este cenário de simulação consiste no transmissor, no receptor e um canal configurável, localizados em um único computador. O sistema de comunicações implementado é mostrado na Figura 34, o diagrama de blocos no simulink é mostrada na Figura 35.

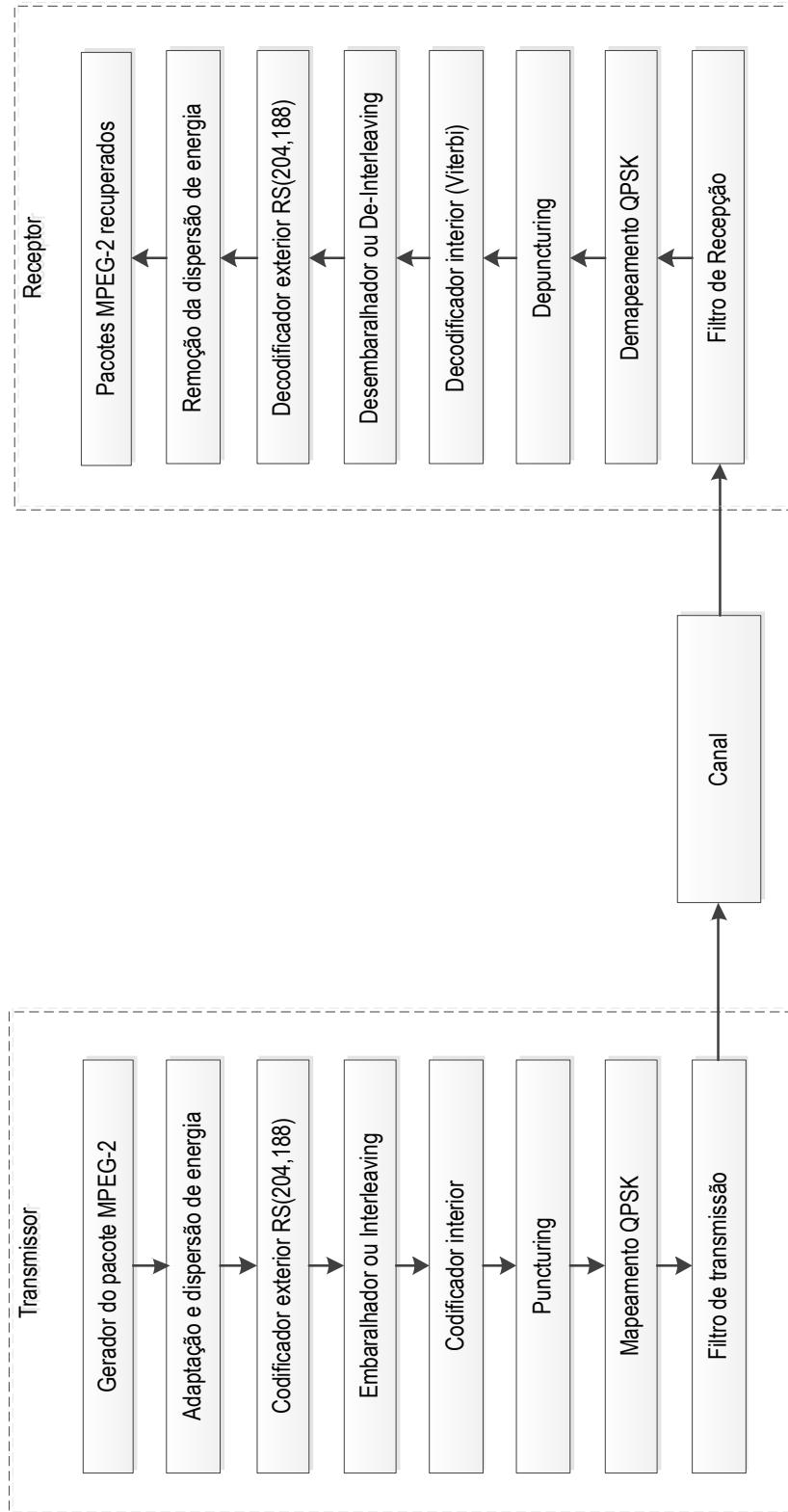


Figura 34 – Diagrama de blocos do transmissor, canal e receptor implementado no cenário controlado no computador

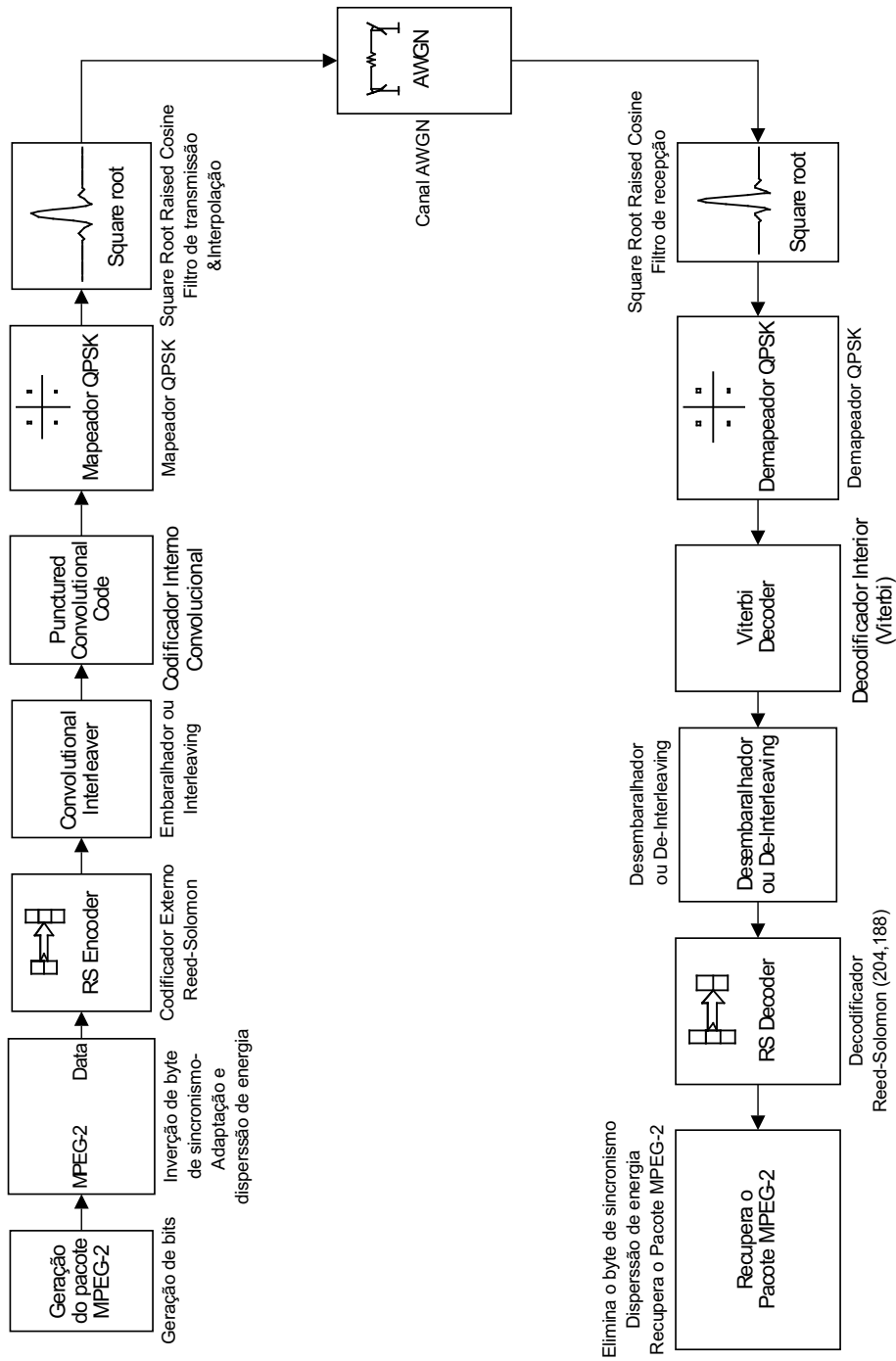


Figura 35 – Diagrama de blocos no Simulink

O canal configurável no simulink contém os parâmetros a seguir:

- E_b/N_0
- Número de bits por símbolo (empregado para o mapeamento da constelação)
- Potência do sinal de entrada, referenciado para um ohm (Watts)
- Período do símbolo (segundos).

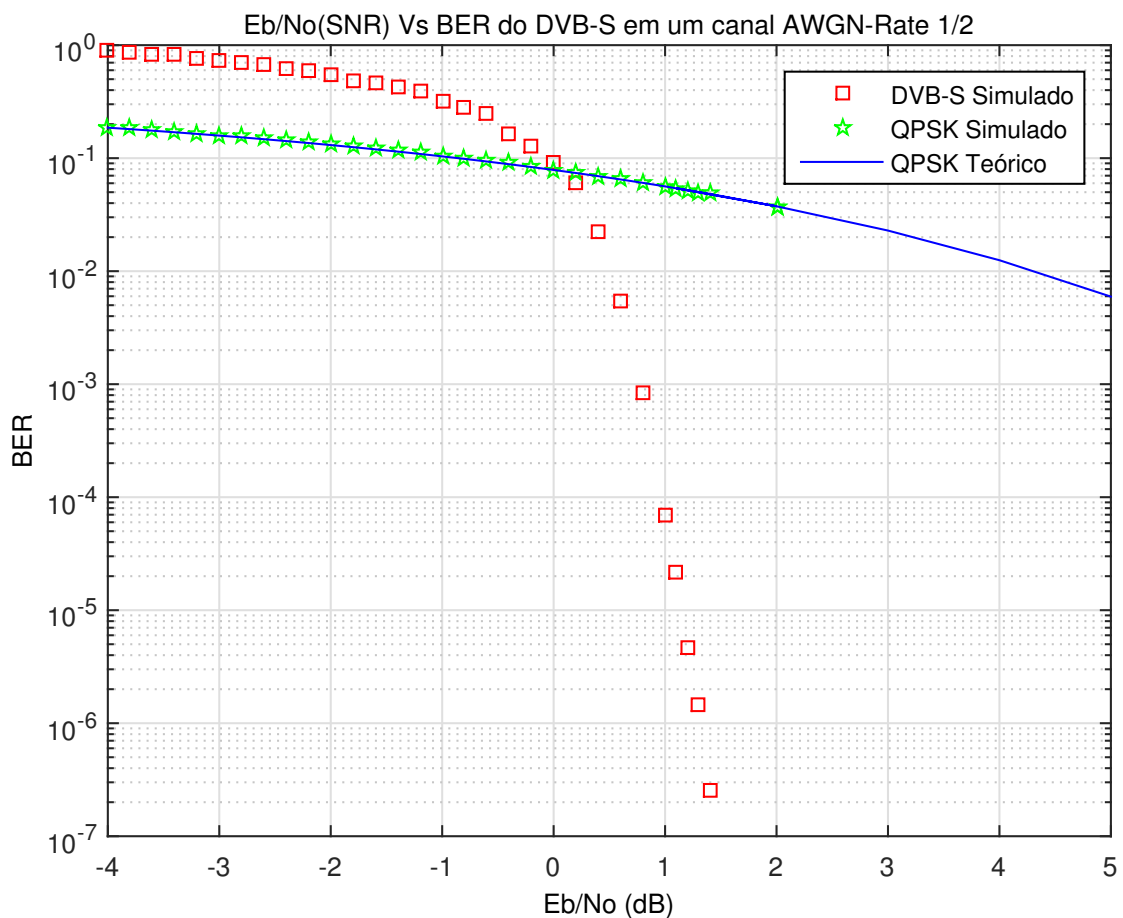


Figura 36 – BER versus E_b/N_0 . Comparação entre QPSK teórico, QPSK simulado, e DVB-S simulado com *puncturing* 1/2.

As Figuras 36 e 37 mostram o comportamento do DBV-S em comparação com o QPSK simulado e teórico. No eixo horizontal de ambas Figuras encontra-se a relação E_b/N_0 , e no eixo vertical, o BER. A diferença entre ambas é a taxa de código (*code rate*). Essa taxa é 1/2 na Figura 36, e 3/4 na Figura 37.

Cabe destacar que o QPSK simulado nos experimentos apresentou um comportamento muito similar ao QPSK teórico.

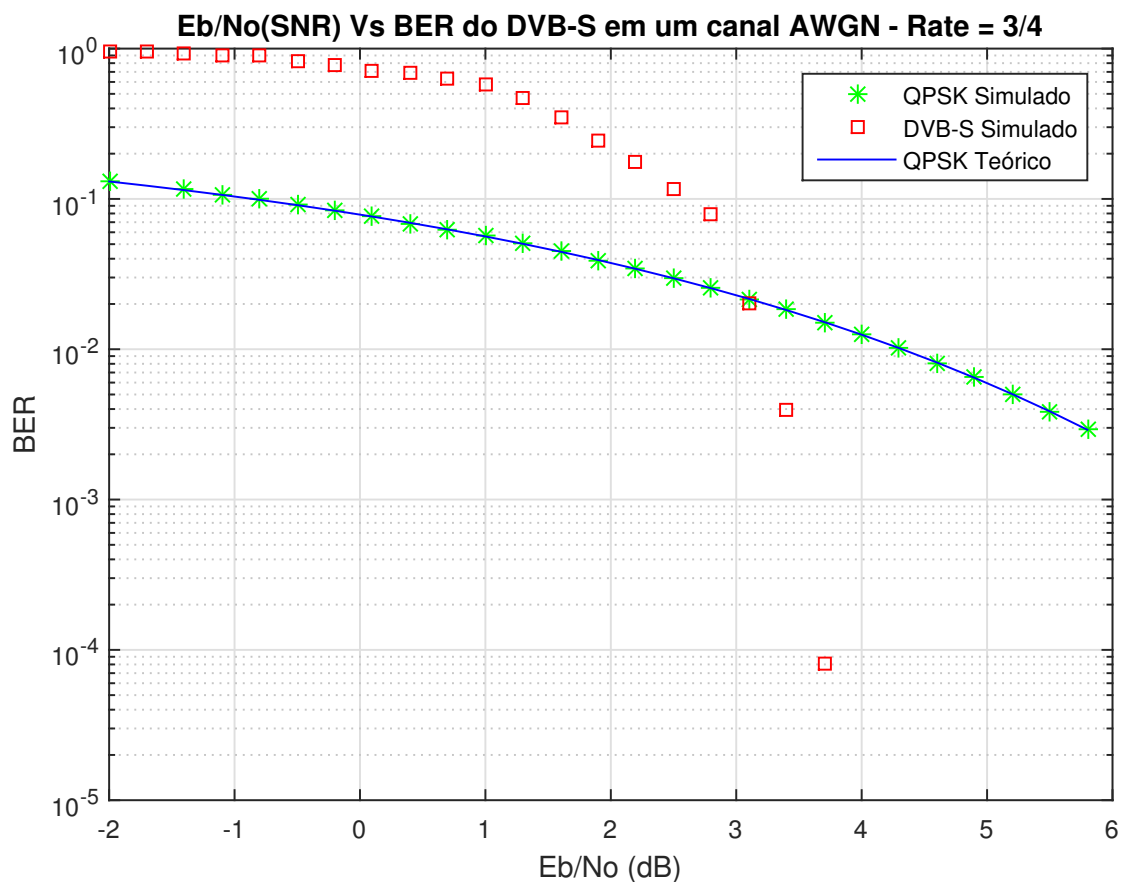


Figura 37 – BER versus Eb/No, para QPSK e o sistema DVB-S, com *puncturing* em uma taxa de código 3/4.

5.3.2 Cenário Real

Os equipamentos usados para a implementação deste cenário são mostrados nas Figuras 38 e 39. Conforme pode ser visualizado em ambas figuras, o sistema de comunicação real que consiste em um transmissor, um receptor, e o canal real (ar). O transmissor consiste em um computador, um equipamento USRP N210 da Ettus Research LLC e uma antena (Figura 40). O receptor possui os mesmos componentes (Figura 41). Cada equipamento USRP N210 está conectado a uma placa filha WBX A. Cabe destacar que a comunicação entre o computador e o equipamento USRP acontece mediante uma conexão gigabit ethernet.

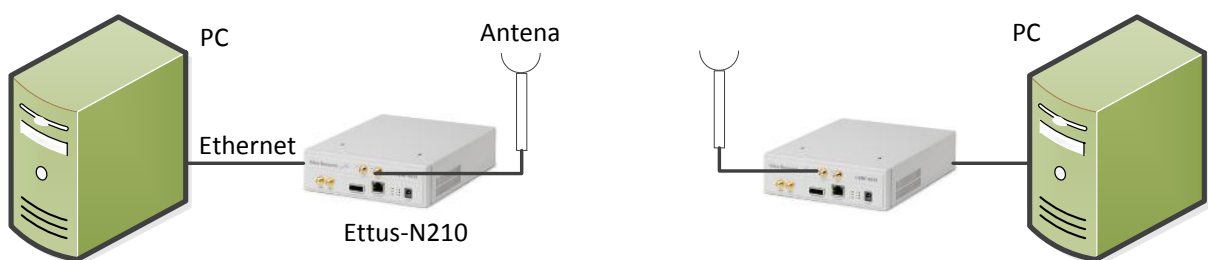


Figura 38 – Equipamento para teste do sistema de comunicação.

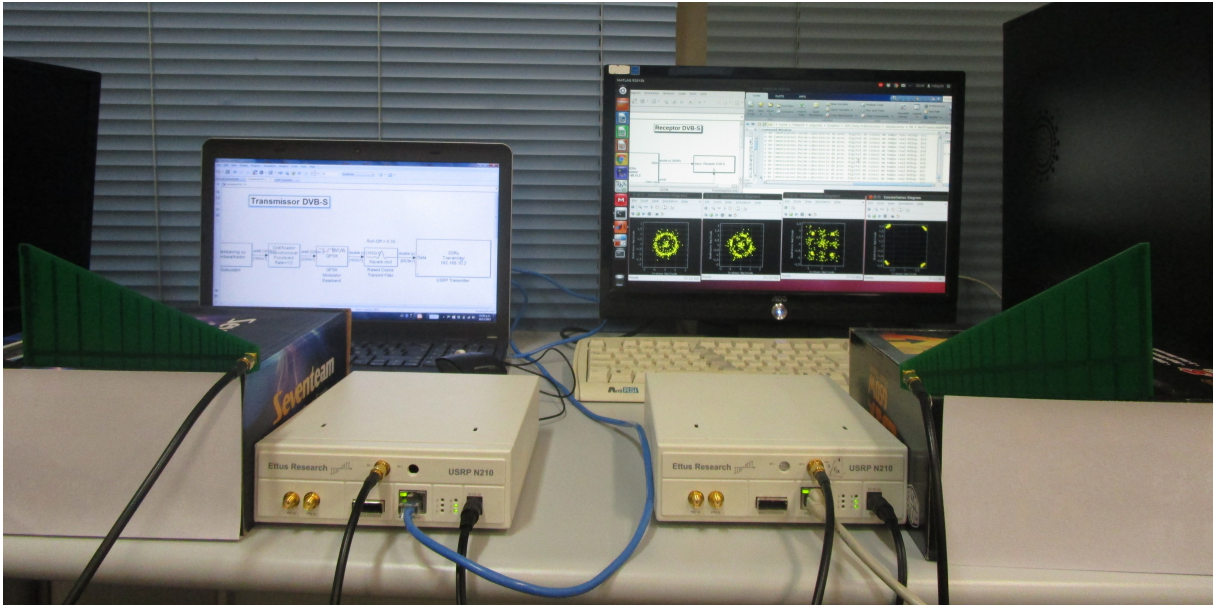


Figura 39 – Diagrama de componentes usados no cenário real.

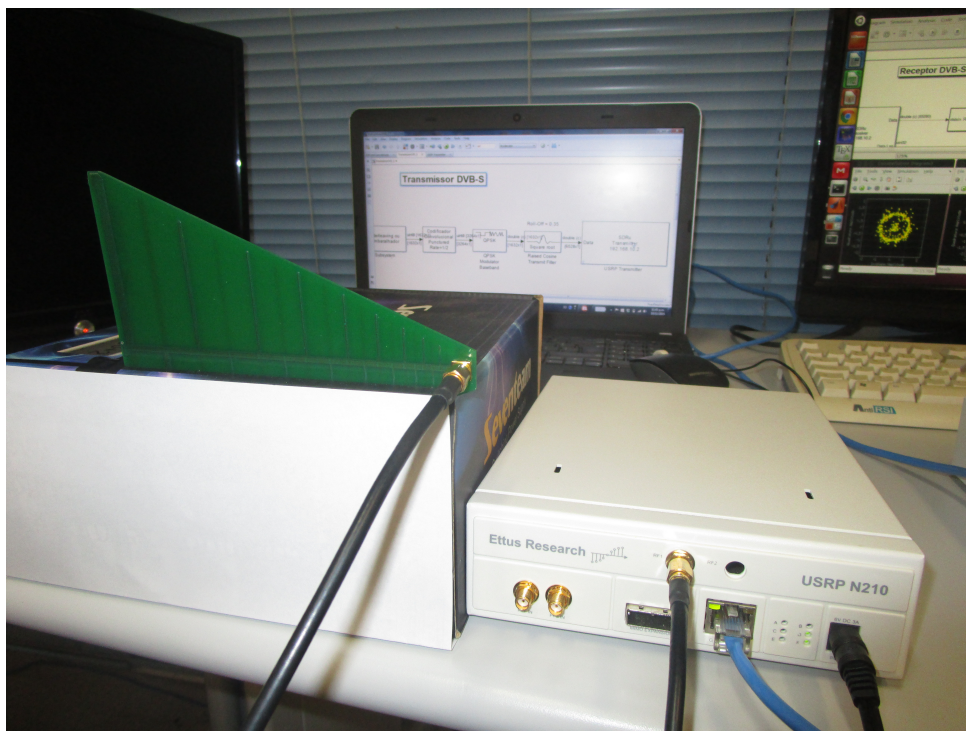


Figura 40 – Diagrama do transmissor.

A Figura 42 mostra os dados na entrada do bloco de sincronismo de frequência (justamente na saída do filtro de recepção). As Figuras 43, 44, 45 mostram o comportamento do sistema após o desenvolvimento das etapas: sincronismo de frequência, sincronismo de fase e sincronismo de relógio respectivamente. No eixo horizontal das Figuras 42, 43, 44, 45 representa-se a amplitude de fase (I) e no eixo vertical está a amplitude de quadratura (Q).

No decorrer do processamento dos sinais recebidos, pode-se observar como os pontos convergem ao centro dos quadrantes. Esse fato possibilita a classificação de cada ponto a

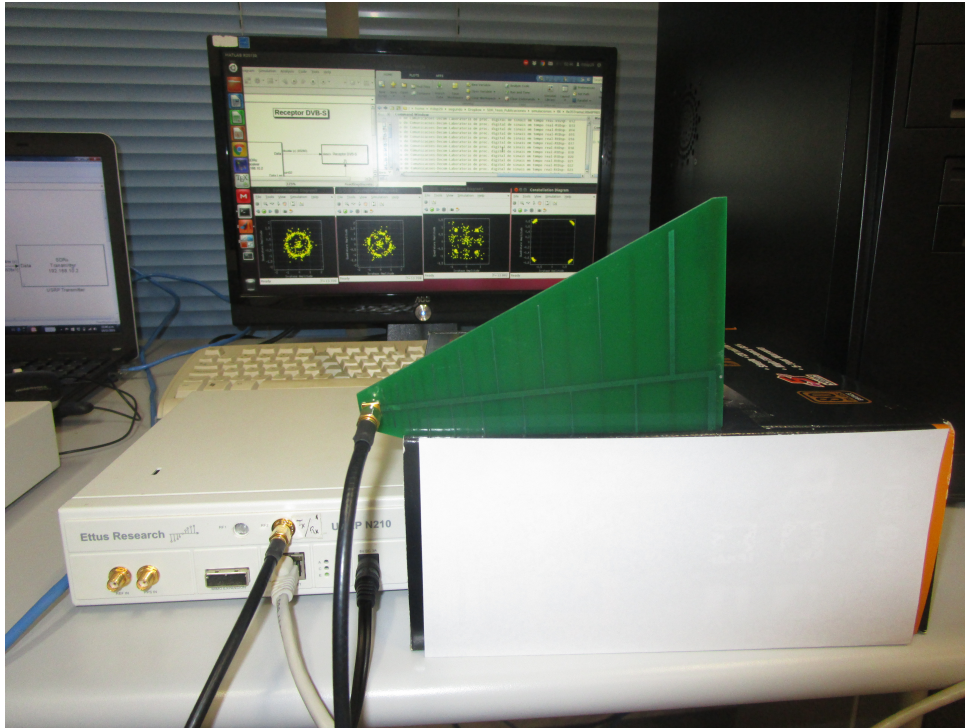


Figura 41 – Diagrama do receptor.

uma das 4 regiões de decisão (uma em cada quadrante). Cada região de decisão representa um símbolo da constelação. Portanto a separação de regiões de decisão facilitará a recuperação da informação no receptor, satisfazendo o propósito do sistema.

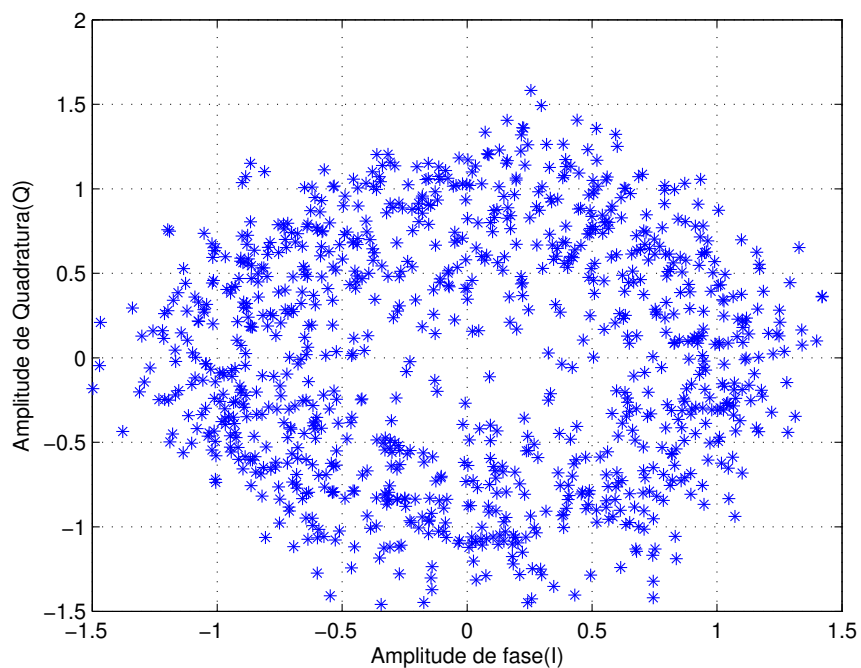


Figura 42 – Constelação obtida na entrada do sincronismo de frequência

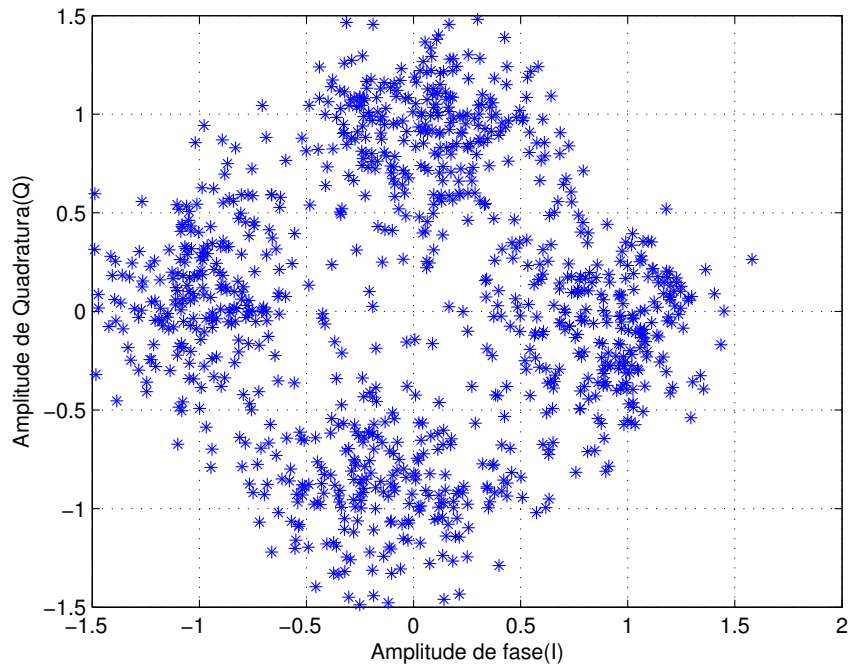


Figura 43 – Constelação obtida na saída do sincronismo de frequência.

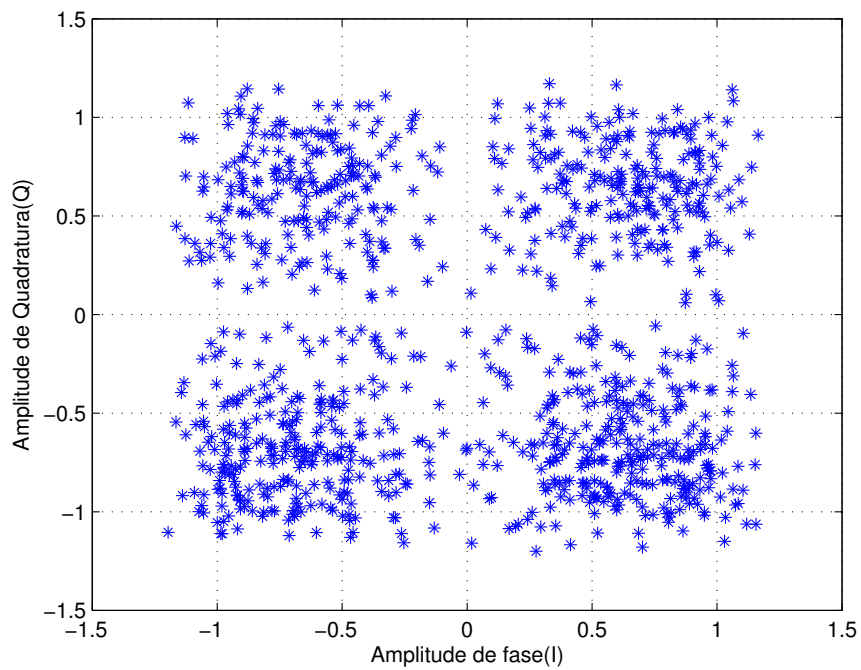


Figura 44 – Constelação obtida na saída do sincronismo de fase.

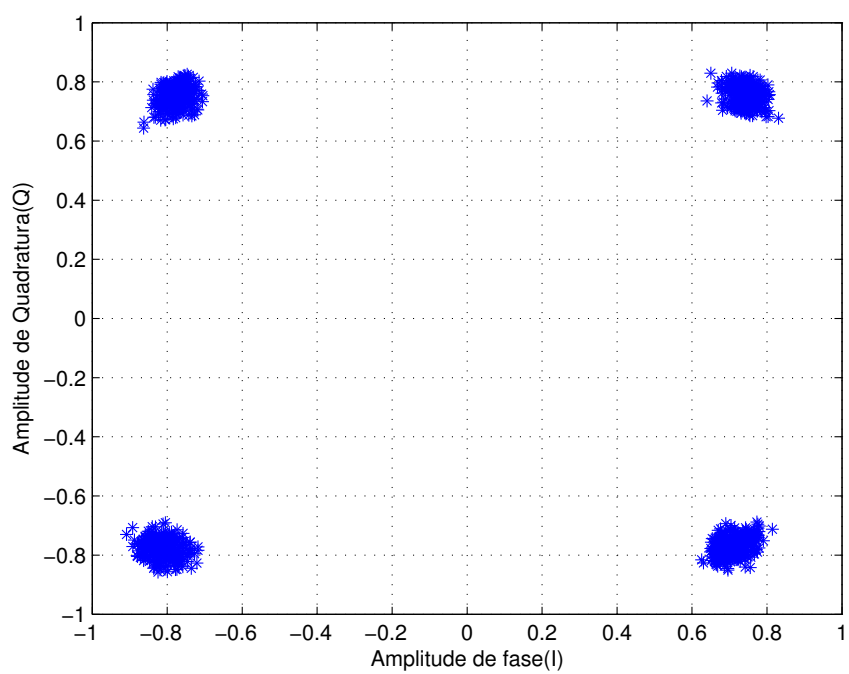


Figura 45 – Constelação obtida na saída do sincronismo de relógio.

6 Considerações finais

Esta dissertação tem como enfoque o uso da tecnologia de rádio definido por software. Essa tecnologia apresenta grande interesse na atualidade, pois permite operar de forma independente das tecnologias de hardware e apresenta diversas vantagens quando comparada às tecnologias convencionais. O uso de SDR permite ainda a operação em diversos padrões e mesmo em diversas faixas de frequências, aumentando desta forma o ciclo de vida dos equipamentos de comunicações. Por meio de uma simples mudança de software ou firmware, pode-se, no mesmo hardware, implementar sistemas de comunicações totalmente distintos. Tal versatilidade e flexibilidade ganha maior importância, pois hoje em dia, tem-se disponíveis sistemas de SDR que podem operar em faixa ampla do espectro.

Primeiramente é descrita um dos principais problemas dos equipamentos de comunicações tradicionais, que é o fato destes equipamentos tornarem-se obsoletos depois de poucos anos de operação. Este fato ressalta a necessidade do uso de tecnologias mais adaptáveis e reconfiguráveis que possam trabalhar com diversos padrões de comunicações sem que haja mudança de equipamento.

Também é abordado o uso da tecnologia SDR com o intuito de contribuir para o ensino de telecomunicações, no sentido de se criar códigos que podem ser aplicados a roteiros práticos de disciplinas de Laboratório de Comunicações Digitais. Para isso, como ponto de partida, aprofundou-se no estudo e implementação de um sistema de comunicação sem fio em portadora única, por ser mais simples em comparação com os sistemas baseados em múltiplas portadoras. Além disto, buscou-se um sistema com uso de técnicas de detecção e correção de erros de uso em um padrão de comunicação real, ou seja, um sistema aderente ao padrão para radiodifusão por satélite, o DVB-S.

Na sequência da dissertação, explica-se os componentes de um sistema de comunicações. Em seguida, estuda-se aspectos teóricos e de implementação de SDR, enfatizando-se as vantagens desse tipo de tecnologia. Também foi feita uma breve descrição das plataformas de hardware e software utilizadas nesse trabalho. Como esse trabalho baseia-se no padrão DVB-S, as suas especificações técnicas foram apresentadas com um maior nível de detalhes que o disponível no documento padrão. O padrão DVB-S introduz várias camadas de proteção aos dados com o intuito de aumentar a robustez da comunicação. O estudo de um padrão prático, o DVB-S, foi de grande importância, pois permitiu abordar aspectos reais que ocorrem em sistemas de comunicações por satélite e que exigem diversos tratamentos. Foram dados detalhes sobre a formatação dos dados para o fluxo de transporte, que permite o tráfego de vídeo, áudio e dados, sendo descrita a formação de pacotes TS. A seguir, adicionou-se um código de detecção e correção de erros de Reed-Solomon. Adicionou-se 16 bytes de redundância para cada pacote

TS de 188 bytes. Este código é chamado de código externo e é de uso comum nos padrões DVB. Em sequência, aplicou-se um embaralhador ou *interleaving* de tipo convolucional que incrementa a eficiência do codificador corretor de erros. O embaralhador mistura as posições dos bytes, com isso uma versão embaralhada dos bytes é enviada, e caso aconteçam mais de 8 erros consecutivos, por desvanecimento do canal ou ruído, o receptor ainda consegue recuperar os dados originais. Em seguida os bits passam por o codificador interno de tipo convolucional. Este codificador oferece uma proteção ao nível de bit e duplica a quantidade de bits. Já o processo de pontuação consiste na remoção de alguns bits codificados. Este processo possibilita o incremento da taxa de código, com a contrapartida que reduz o desempenho do corretor de erros. Finalmente, neste padrão, os dados são transmitidos usando modulação QPSK. Portanto, a escolha deste padrão deu-se pelo fato de ser um padrão de complexidade moderada, mas que abrange muitos tópicos relevantes ao ensino de comunicações digitais, a saber: modulação, demodulação, sincronismo de frequência e símbolo, códigos corretores de erros, técnicas de *interleaving*, entre outras.

Outro aspecto importante abordado foi o sincronismo de frequência, fase e relógio. Esta etapa é de vital importância para a recuperação das informações no receptor. A intenção é reproduzir no receptor uma estimativa da base de tempo do transmissor. Por isso se estudou o PLL, primeiro em sua forma analógica e depois em sua forma digital. Os princípios do PLL são usados para compreender o funcionamento do circuito de sincronismo de fase e relógio. Por fim, tratou-se do sincronismo de quadro cujo objetivo é a detecção do início de cada pacote.

Em seguida, descreve-se os equipamentos e software usados na implementação efetuada, descrevendo-se detalhes das especificações para o periférico SDR usado (USRP-N210). Posteriormente, mostram-se dois cenários de testes para o sistema. O primeiro em um ambiente controlado (simulação usando um computador) e o segundo foi realizado usando o periférico USRP-N210. Para o primeiro cenário foi realizado testes de medidas da BER em função da razão Eb/No para taxa de código de 1/2 e 3/4.

A principal motivação deste trabalho foi o estudo da tecnologia SDR visando sua aplicação no ensino de telecomunicações. Conclui-se que a tecnologia em estudo pode ser usada como uma valiosa ferramenta educacional que permite a inclusão de novos conteúdos teóricos e aplicações práticas quando comparada aos cursos clássicos de telecomunicações. O SDR abre as possibilidades de fazer demonstrações, em sala de aula ou em laboratórios, de aspectos práticos que ilustrem os conceitos teóricos, motivando o aluno e ajudando na compreensão (KATZ; FLYNN, 2009) dos conteúdos.

Também ficou claro o ganho de produtividade que se consegue com o uso de ferramentas de alto nível, como o Simulink no lado do software e a USRP no lado do hardware. O uso dessas ferramentas de alto nível permite que se abstraia certos detalhes de implementação, o que possibilita um maior foco nos aspectos sistêmicos do transmissor e receptor. Do ponto de vista de ensino, essa é, certamente, uma característica desejável. Por outro lado, a medida que

a metodologia de *system level design* (PUTTEN *et al.*, 1998) ganha popularidade em projetos, especialmente de telecomunicações, o uso desse tipo de ferramenta é justificado também do ponto de vista prático. Assim, há um ganho na abordagem de ensino proposta neste trabalho.

6.1 Trabalhos Futuros

Como continuidade deste trabalho, seguem abaixo algumas linhas que podem ser adotadas, realçando-se que muitos outros tópicos poderiam naturalmente ser incluídos:

- Implementação do padrão DVB-S2
- Análise e estimativa e caracterização de um sistema completo de SDR, incluindo a parte analógica
- Implementação de *soft decision* no decodificador de Viterbi, e
- Implementação de outros padrões de comunicações digitais.

Referências

- AMADO, P. A. *Sistema DVB para Transmissão de Televisão Digital*. 2015. Disponível em: <http://www.img.lx.it.pt/~fp/cav/Additional_material/Transmissao_DVB.pdf>. Acesso em: 05/11/2015. Citado na página 31.
- CCSDS, C. C. for S. D. S. *TM SYNCHRONIZATION AND CHANNEL CODING SUMMARY OF CONCEPT AND RATIONALE*. 2012. Disponível em: <<http://public.ccsds.org/publications/archive/130x1g2.pdf>>. Acesso em: 01/11/2012. Citado 2 vezes nas páginas e 33.
- CLARKE, C. K. P. Reed-solomon error correction. *Electronics Letters*, v. 1, n. 31, p. 1009–1010, Jul 2002. ISSN 0013-5194. Citado na página 28.
- ETSI. *Padrão europeu EN 300 421 do DVB-S*. 1993. [Http://www.etsi.org](http://www.etsi.org). Acesso em: 24/07/2014. Citado 4 vezes nas páginas 15, 26, 33 e 37.
- ETTUS. *Ettus Research*. 2014. Disponível em: <<http://home.ettus.com/>>. Acesso em: 17/07/2014. Citado 4 vezes nas páginas , 15, 22 e 56.
- EVERETT, M. W. E. *Carrier Offset Recovery*. 2013. Disponível em: <<http://cmclab.rice.edu/433/slides/7-CarrierRecovery.pdf>>. Acesso em: 05/12/2013. Citado 2 vezes nas páginas e 46.
- FISCHER, W. *Digital Video and Audio Broadcasting Technology: A Practical Engineering Guide*. 3rd. ed. [S.l.]: Springer Publishing Company, Incorporated, 2010. ISBN 3642116116, 9783642116117. Citado 3 vezes nas páginas , 32 e 40.
- FORNEY G., J. The viterbi algorithm. *Proceedings of the IEEE*, v. 63, n. 3, p. 268–278, March 1973. ISSN 0018-9219. Citado na página 39.
- FORUM, W. I. *The Wireless Innovation Forum*. 2014. [Http://www.wirelessinnovation.org](http://www.wirelessinnovation.org). Acesso em: 24/07/2014. Citado 2 vezes nas páginas 14 e 18.
- GNURADIO. *GNURadio*. 2015. [Http://gnuradio.org/](http://gnuradio.org/). Acesso em: 11/09/2015. Citado na página 24.
- INSTRUMENTS, N. *National Instruments*. 2014. [Http://ni.com](http://ni.com). Acesso em: 17/07/2014. Citado na página 15.
- ISO/IEC-DIS-13818-1. *Coding of moving pictures and associated audio*. June 1994. Citado na página 27.
- KATZ, S.; FLYNN, J. Using software defined radio (sdr) to demonstrate concepts in communications and signal processing courses. In: *Frontiers in Education Conference, 2009. FIE '09. 39th IEEE*. [S.l.: s.n.], 2009. p. 1–6. ISSN 0190-5848. Citado 2 vezes nas páginas 15 e 67.
- KAZAZ, T.; KULIN, M.; HADZIALIC, M. Design and implementation of sdr based qpsk modulator on fpga. In: *Information Communication Technology Electronics Microelectronics (MIPRO), 2013 36th International Convention on*. [S.l.: s.n.], 2013. p. 513–518. Citado na página 35.

LACKEY, D. R. Speakeasy: The military software radio. *IEEE Communications Magazine*, v. 33, n. 5, p. 56–61, May 1995. Citado na página 19.

MATHWORKS. *Mathworks*. 2014. [Http://www.mathworks.com/](http://www.mathworks.com/). Acesso em: 04/11/2014. Citado na página 25.

MATHWORKS. *QPSK Receiver with SDR Hardware*. 2015. [Http://www.mathworks.com/help/supportpkg/xilinxfpgabasedradio/examples/qpsk-receiver-with-sdr-hardware-1.html?requestedDomain=www.mathworks.com](http://www.mathworks.com/help/supportpkg/xilinxfpgabasedradio/examples/qpsk-receiver-with-sdr-hardware-1.html?requestedDomain=www.mathworks.com). Acesso em: 14/01/2015. Citado na página 46.

MATHWORKS. *Simulink*. 2015. [Http://www.mathworks.com/products/simulink/](http://www.mathworks.com/products/simulink/). Acesso em: 18/01/2015. Citado na página 25.

MITOLA J., I. Software radios-survey, critical evaluation and future directions. In: *Telesystems Conference, 1992. NTC-92., National*. [S.l.: s.n.], 1992. p. 13/15–13/23. Citado na página 14.

MOON, T. K. *Erro Correction Coding*. [S.l.: s.n.], 2002. Citado 4 vezes nas páginas , 27, 31 e 35.

PUTTEN, P. van der; VOETEN, J.; GEILEN, M.; STEVENS, M. System level design methodology. In: *VLSI '98. System Level Design. Proceedings. IEEE Computer Society Workshop on*. [S.l.: s.n.], 1998. p. 11–16. Citado na página 68.

REED, J. H. *Software Radio. A Modern Approach to Radio engineering*. [S.l.: s.n.], 2002. Citado na página 18.

REIS, A. L. G.; BARROS, A. F.; LENZI, K. G.; MELONI, L. G. P.; BARBIN, S. E. Introduction to the software-defined radio approach. *IEEE Latin America Transactions*, v. 10, n. 1, p. 1156–1161, Jan 2012. ISSN 1548-0992. Citado na página 18.

RICE, M. *Digital Communications: A Discrete-Time Approach*. [S.l.: s.n.], 2009. Citado 8 vezes nas páginas , 41, 42, 48, 50, 51, 52 e 53.

SUN, D.-Y.; XU, J. Zero-if topology. *Electronics Letters*, v. 36, n. 12, p. 1009–1010, Jun 2000. ISSN 0013-5194. Citado 2 vezes nas páginas 21 e 23.

TOMASI, W. *Electronic Communications Systems: Fundamentals Through Advanced, Fifth edition*. [S.l.]: Prentice Hall, 2003. ISBN 0130494925. Citado na página 17.

VITERBI, A. J. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Proceedings of the IEEE*, v. 13, n. 2, p. 260–269, Abril 1967. ISSN 0018-9448. Citado na página 39.

WAL, R. van der; MONTREUIL, L. Qpsk and bpsk demodulator chip-set for satellite applications. *Consumer Electronics, IEEE Transactions on*, v. 41, n. 1, p. 30–41, Feb 1995. ISSN 0098-3063. Citado na página 49.

ZIVKOVIC, M.; SCHMITZ, J.; MATHAR, R. Sdr in teaching: An OFDM communication primer. In: *8th Karlsruhe Workshop on Software Radios*. Karlsruhe, Germany: [s.n.], 2014. p. 41–46. Citado na página 15.

Anexos

ANEXO A – Placa filha WBX.

É um transceptor IQ de conversão direta para 50-2200 MHz, que alimenta os conversores analógico-digital (ADC) e digital-analógico (DAC) da placa mãe USRP. O ADC e o DAC são controlados por uma matriz de portas programáveis por campo (FPGA) que envia e recebe amostras do computador. A demodulação e decodificação são feitas no computador.

O diagrama de blocos simplificado da cadeia de recepção da placa filha WBX é mostrado na Figura 46. O diagrama não mostra as chaves de transmissão/recepção e de seleção de antena, além da proteção contra alta potência do MGA82563 e do misturador.

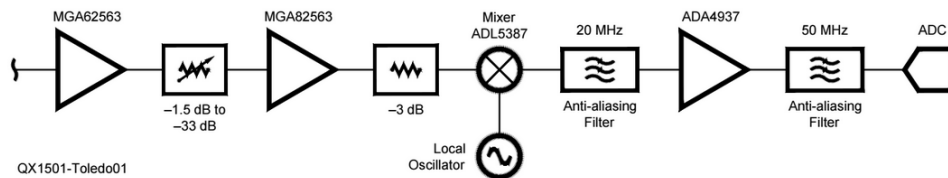


Figura 46 – Diagrama de blocos simplificado da cadeia de recepção da placa filha WBX e o ADC da placa mãe USRP.

A resposta do receptor é plana para toda a banda de frequência. O misturador converte sinais próximos à frequência do oscilador local (LO) em frequências próximas à CC, e o filtro *anti-aliasing* as passa aos ADCs. Sinais muito longe do LO são convertidas em altas frequências que são bloqueadas por o filtro *anti-aliasing*.