

PROCESSAMENTO DIGITAL DE SINAIS APLICADO A DETECÇÃO AUTOMÁTICA DA MODULAÇÃO EM RÁDIO DEFINIDO POR SOFTWARE

Alysson Vasconcelos Gomes de Menezes¹, Edmar Candeia Gurjão²

RESUMO

Neste trabalho foi implementado um rádio por software capaz de identificar de forma automática o tipo de modulação do sinal recebido pela sua antena. Utilizando uma USRP (*Universal Software Radio Peripheral*), são captados sinais de estações AM e FM da cidade de Campina Grande, que após serem digitalizados são enviados a um computador para serem processados. O processamento do sinal foi feito usando o GNU Radio, no qual foi implementado o cálculo de um parâmetro que permitiu quantificar a variação na amplitude do sinal recebido, e assim classificar o tipo de modulação do mesmo em comparação com um valor limite especificado na técnica utilizada. Verificou-se que mesmo para sinais de estações comerciais, com todas as imperfeições e ruído, o sistema conseguiu decidir corretamente as modulações utilizadas.

Palavras-chave: modulação, detecção, rádio por software.

DIGITAL SIGNALS PROCESSING APPLIED TO AUTOMATIC MODULATION DETECTION USING SOFTWARE DEFINED RADIO

ABSTRACT

In this work it was implemented a software radio to automatically identify the modulation type of a signal received by its antenna. Using an USRP (*Universal Software Radio Peripheral*), we received signals from AM and FM broadcasting stations in Campina Grande, digitalized and sent these signals to a computer to process them. This processing was implemented in GNU Radio from which we obtain a parameter, to quantify the variation in the received signals amplitude and classify their modulation type after doing a comparison with the threshold specified by the used technique. It was verified that even for signals received from commercial stations, with noise and imperfections, the system was capable of correctly identifying their modulation types.

Keywords: modulation, detection, software radio.

INTRODUÇÃO

Um rádio definido por *software* consiste em um sistema no qual os componentes tipicamente implementados em *hardware* (filtros, moduladores, demoduladores, entre outros) são substituídos por implementações em *software*, em geral utilizando computadores ou processadores digitais de sinais (MITOLA, 2000). Algumas das principais vantagens inerentes à substituição do hardware são: o barateamento do sistema e o aumento modularidade, isto é, alterando apenas o *software* podemos construir diversos sistemas com o mesmo hardware.

Os passos necessários para obter um rádio por software na recepção consistem em um etapa para o deslocamento do espectro do sinal modulado para a frequência intermediária seguida da digitalização desse sinal para que em seguida seja processado pelo software. O deslocamento do espectro pode ser feito usando uma etapa super heterodina (LATHI, 1998), e a digitalização usando conversores analógicos digitais de uso comum. Neste trabalho foi utilizada a arquitetura denominada URSP (*Universal Software Radio*

¹ Aluno do Curso de Engenharia Elétrica, Unidade Acadêmica de Engenharia Elétrica, UFPG, Campina Grande, PB, E-mail: vasconcelos.alysson@gmail.com

² Engenheira Elétrica, Prof. Doutor, Unidade Acadêmica de Engenharia Elétrica, UFPG, Campina Grande, PB, E-mail: ecandeia@dee.ufcg.edu.br

Peripheral) que engloba essas duas etapas. O processamento é feito em um computador de uso comum usando uma plataforma de software denominada GNU Radio.

Como a implementação das funcionalidades necessárias para implementar o rádio é feita em software, pode-se incluir técnicas mais elaboradas de processamento de sinais e de inteligência artificial. Essas técnicas podem possibilitar, por exemplo, a detecção do tipo de modulação usada pelo transmissor, tornando o receptor automaticamente adaptável ao transmissor.

Entretanto, para que essas técnicas funcionem corretamente, os sistemas implementados devem receber parâmetros que normalmente estão relacionados ao sinal de entrada. Esses parâmetros são obtidos pelo processamento digital do sinal recebido.

O objetivo deste trabalho é construir um rádio por software capaz de identificar automaticamente o tipo de modulação do sinal captado pelo receptor. Para facilitar a implementação foram escolhidas as modulações em amplitude (AM) e em frequência (FM), e para tornar os resultados mais reais foram utilizados os sinais de estações comerciais.

MATERIAL E MÉTODOS

Este trabalho foi desenvolvido no Laboratório de Processamento de Sinais (LAPS) da Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande – PB.

Material

Foram utilizados uma USRP (Figura 1), um computador de uso geral e a plataforma de desenvolvimento GNU Radio.

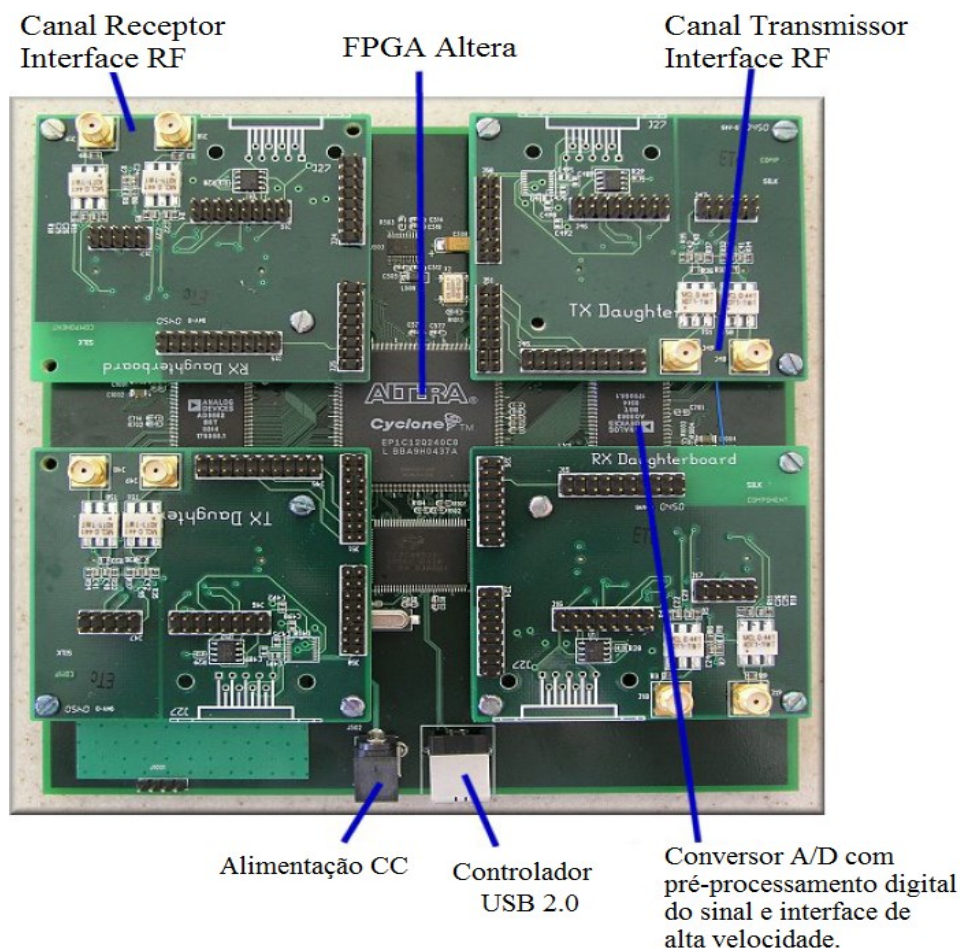


Figura 1. A Universal Software Radio Peripheral (USRP).

A USRP é uma pequena placa-mãe contendo quatro conversores analógico-digital (A/D) de 12 bits e 64M amostras/segundo, quatro conversores digital-analógico (D/A) de 14 bits e 128M amostras/segundo, um *Field Programmable Gate Array* (FPGA) e um controlador USB 2.0. Cada placa-mãe suporta até quatro placas filhas, duas receptoras e duas transmissoras. Neste trabalho foi usada a placa filha conhecida como receptor básico (*Basic RX*) que pode receber sinais de 0 (zero) a 150 MHz, essa placa tem função de captar um sinal na frequência indicada pelo usuário, converter para a frequência intermediária e em seguida repassar para o conversor A/D na placa mãe.

O computador utilizado tem as seguintes configurações: processador Intel Core 2 Duo E7300 2.66 GHz, placa-mãe Asus P5KPL-AM, memória DDRII 4Gb, o que favoreceu o andamento do trabalho. Além disso, esse computador possui USB 2.0 sem a qual não é possível obter conexão com a USRP.

A plataforma de desenvolvimento GNU Radio consiste em um conjunto de *drivers* que além de possibilitar a comunicação entre a placa USRP e o computador através de uma porta USB 2.0, permite a implementação de blocos para processamento de sinais e um conjunto de exemplos prontos para serem executados. Também foi necessário que utilizássemos as linguagens de programação C++ e Python.

Metodologia

O sinal modulado é recebido pela antena conectada a URSP e digitalizado, nesse ponto faz-se necessário indicar a frequência que se deseja sintonizar e a decimação a ser utilizada. Uma decimação 16, por exemplo, indica que após a primeira amostra ser coletada apenas as 16^a, 32^a, 48^a, e assim sucessivamente, são coletadas. Todas as demais amostras são desconsideradas. Esta técnica serve para diminuir a taxa de amostragem do sinal recebido e permitir que amostras adequadas sejam selecionadas. Imaginemos, por exemplo, que o nosso sistema está captando um sinal AM com uma taxa de amostragem suficientemente alta de modo que as amostras coletadas são tão próximas que não se observa variação significativa em suas amplitudes. Nestas circunstâncias, o sistema classificaria erroneamente o tipo de modulação. Para contornar esta situação é necessário utilizar a técnica de decimação. Após ser digitalizado na decimação adequada, o sinal é enviado via porta USB para um computador que faz o processamento do mesmo.

A faixa de frequência utilizada foi de 88kHz a 108MHz, que engloba as faixas de AM e FM comercial. Para identificar possíveis estações de rádio utiliza-se um programa capaz de varrer a faixa desejada do espectro de frequências. Assim, é possível detectar alguns picos característicos da existência de portadoras em algumas frequências do espectro. Então, utiliza-se o sistema detector do tipo de modulação na frequência alvo para verificar se a estação em questão se trata de uma AM ou FM.

No computador é implementado o software responsável pelo processamento. Neste trabalho utilizamos o GNU Radio que necessita para a construção de um sistema de processamento de um grafo de fluxo no qual os blocos de processamento ocupam os vértices e as conexões entre os mesmos indicam o caminho a ser percorrido pelo sinal de entrada. Os atributos de cada bloco incluem o número de portas de entrada e saída, bem como o tipo de dado que flui através de cada porta. Conceitualmente, os blocos são capazes de processar uma seqüência infinita de dados que fluem de sua(s) porta(s) de entrada para a(s) de saída.

Apesar de existir um conjunto de blocos de processamento já implementados, foi necessário implementar novos blocos. Para tanto, observou-se que são necessárias duas camadas de software, uma de mais alto nível escrita em linguagem Python e outra de mais baixo nível escrita em C++. Essa divisão é feita devido a questões de velocidade de processamento, que em C++ é maior.

Para testar o sistema decidiu-se utilizar os sinais de estações comerciais AM e FM operando na cidade de Campina Grande, e captados pela antena da USRP. Essa escolha se justifica pois o sinal recebido dessa forma já vem com as imperfeições impostas pelo meio de transmissão e com o ruído.

A técnica utilizada

Proposta em (AZZOUZ, 1996), a técnica consiste em calcular γ_{\max} , que é o valor máximo da densidade espectral de potência da amplitude instantânea normalizada centrada do sinal interceptado, e comparar com o valor limite, tomado como 6. Se γ_{\max} for maior que o limite o sinal possui informação contida na amplitude e deve ser classificado como AM. Caso contrário, o sinal deve ser classificado como FM. A normalização na amplitude instantânea das amostras é necessária para compensar o ganho do canal.

$$\gamma_{\max} = \max \left(\frac{|FFT(A_{cn}(i))|^2}{N_s} \right) \quad (1)$$

Em que $FFT(x(i))$ é a transformada de Fourier de $x(i)$ e $A_{cn}(i)$ é o valor normalizado centrado da amplitude instantânea nos instantes de tempo $t = i / f_s$ ($i = 1, 2, 3, \dots, N_s$) definido por:

$$A_{cn}(i) = \frac{A(i)}{m_a} - 1 \quad (2)$$

$$m_a = \frac{1}{N_s} \sum_{i=1}^{N_s} A(i) \quad (3)$$

sendo N_s o número de amostras por quadro; f_s a taxa de amostragem; m_a a média da amplitude instantânea ao longo do quadro.

Procedimento para obtenção do parâmetro γ_{\max}

Após compreendermos as técnicas de interligação dos blocos disponibilizados pelo GNU Radio e a criação de novos blocos, implementamos um bloco capaz de calcular a amplitude média das amostras do sinal captado pela USRP e outro para fazer a comparação entre o valor do parâmetro calculado (γ_{\max}) e o limite. Com os blocos disponibilizados (complex_to_real, multiply_const_ff, divide_ff, add_const_ff, fft_vfc, complex_to_mag_squared, max_ff) e os que implementamos (media_ff e decide_ff), pudemos criar o grafo de fluxo da Figura 2, o que nos permitiu classificar o tipo de modulação do sinal recebido em AM ou FM.

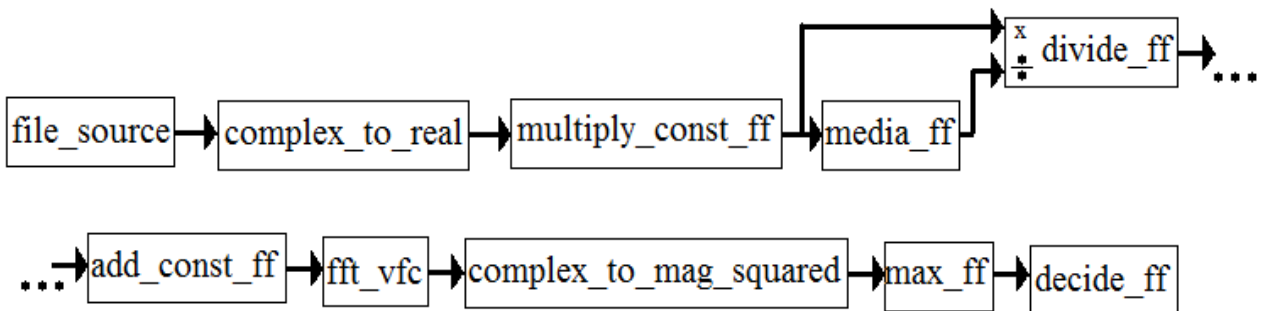


Figura 2. Grafo para classificação do tipo de modulação do sinal recebido.

A Figura 2 retrata o caminho percorrido pelas amostras coletadas através da USRP desde o instante em que são recebidas pelo sistema de classificação da modulação até o momento em que o decisório (decide_ff) nos fornece a informação sobre o tipo de modulação do sinal captado.

Inicialmente as amostras são armazenadas em um arquivo, que deve conter no máximo 8 mil delas, pois uma quantidade excessiva de amostras compromete o bom funcionamento do sistema. Posteriormente, utilizamos o bloco file_source para ler as amostras armazenadas, as quais são números complexos e devem ser convertidas em valores reais antes de prosseguirmos com o processamento. Estas amostras são números complexos porque a USRP fornece as componentes em fase e em quadratura do sinal recebido. A conversão para números reais é feita utilizando-se o bloco complex_to_real e após este processo é preciso que multipliquemos os valores reais por 2. Para isso, utilizamos o bloco multiply_const_ff com 2 em seu argumento.

Em seguida, os valores reais obtidos na saída do bloco multiply_const_ff são normalizados, isto é, calculamos a média aritmética entre eles utilizando o bloco media_ff e os dividimos pelo valor obtido na saída do bloco media_ff. A divisão é feita utilizando o bloco divide_ff.

De acordo com a Equação 2 é preciso subtrair o valor 1 das amplitudes instantâneas normalizadas, as quais se encontram na saída do bloco divide_ff. Assim, interligamos a saída do bloco divide_ff à entrada do

add_const_ff, o qual possuía -1 como argumento. A saída do bloco add_const_ff corresponde ao parâmetro $A_{cn}(i)$, que é detalhado na equação 2.

Utilizamos o bloco fft_vfc para calcular a transformada rápida de Fourier (FFT) do vetor obtido como saída de add_const_ff. Em seguida, utilizamos o bloco complex_to_mag_squared para obtermos um vetor composto pelas magnitudes elevadas ao quadrado dos valores complexos resultantes do cálculo da FFT. Interligamos a saída do bloco complex_to_mag_squared à entrada do max_ff de modo a selecionarmos o maior número real entre os componentes do vetor resultante na saída do bloco complex_to_mag_squared.

Por fim, o decisório faz a divisão entre o valor recebido em sua entrada e o número de amostras utilizadas no processamento, de acordo com a Equação 1, e compara o resultado desta operação com o limite especificado na técnica. Caso este resultado seja maior do que 6, o sinal recebido é classificado como modulado em amplitude, caso contrário, como modulado em frequência.

A saída do bloco decide_ff é uma mensagem de texto contendo "MODULACAO FM" ou "MODULACAO AM", a depender do resultado de processamentos prévios.

RESULTADOS E DISCUSSÃO

O sistema obtido produz como saída uma indicação textual da decisão sobre o tipo de modulação. Para testarmos a eficiência do método, efetuamos o seguinte experimento: coletamos 16 amostras com decimação 16 sucessivas vezes das seguintes estações de rádio em Campina Grande: 93,1MHz (FM), 98,1MHz (FM), 1350kHz (AM), 1050kHz (AM) e constatamos que os erros eram inferiores a 10% para as estações 93,1MHz e 1350kHz. No ambiente em que os testes foram realizados, as transmissões que eram mais nitidamente escutadas correspondiam as das estações nas quais o sistema detector de modulação obteve eficiência acima de 90%. Para estações em que a relação sinal ruído era notadamente menor, a coerência entre o tipo de modulação do transmissor e a decisão do sistema receptor diminuiu.

Para aumentarmos a eficiência na classificação do tipo de modulação, utilizamos a técnica de decisão por maioria, isto é, a cada dez quadros, classificamos a modulação de acordo com a classificação que ocorreu mais vezes entre os dez quadros.

CONCLUSÕES

Diante dos resultados obtidos nesta pesquisa experimental, onde implementamos um sistema capaz de identificar entre os tipos AM e FM a modulação de um sinal recebido através de um USRP, concluímos que: a coerência na classificação do tipo de modulação por parte do sistema implementado depende, dentre outros fatores, da relação sinal ruído da estação em questão.

O número de amostras e a decimação utilizadas influem na decisão tomada pelo sistema. Um valor que obtivemos experimentalmente e que melhor se adequou aos resultados esperados foi 16 tanto para a decimação quanto para o número de amostras.

O kit de desenvolvimento utilizado contém vários exemplos que contribuem significativamente para o entendimento do GNU Radio, porém, é notável a escassez de documentação que oriente novos usuários no processo de construção de blocos para processamento de sinais.

Para frequências em que não há estações de rádio, o sistema implementado pode classificar o tipo de modulação do sinal captado tanto como AM quanto como FM.

AGRADECIMENTOS

Ao CNPq pela bolsa de Iniciação Científica. Este trabalho foi apoiado em parte pelo projeto PPP No 2158/08, Edital 00108 FAPESQ/MCT/CNPq.

REFERÊNCIAS BIBLIOGRÁFICAS

MITOLA III, J. **Software Radio Architecture: Object-Oriented Approaches to Wireless Systems Engineering**. 1.ed. New York: John Wiley & Sons, Inc, 2000. p.1-110.

AZZOUZ, E.E.; NANDI, A. K. **Automatic Modulation Recognition of Communication Signals**. 1.ed. Netherlands: Kluwer Academic Publishers, 1996. p.1 -76.

GNU Radio. **Rádio definido por Software**. Disponível em: <<http://gnuradio.org/trac>> Último acesso em: 28 de jul. 2009.

LATHI, B. P. **Modern Digital and Analog Communication Systems**. 3.ed. New York: Oxford University Press, 1998. p. 1-245.