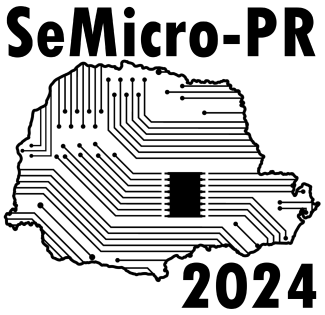

Implementação em FPGA de um polinômio de memória com atrasos baseados em médias móveis

Vitória Mariana da Rocha Gasino, Maria Eduarda Rizo, Sibilla Batista da Luz França, Eduardo Gonçalves de Lima
Grupo de Concepção de Circuitos e Sistemas Integrados
Universidade Federal do Paraná, Curitiba, Brasil
vitoriagasino@ufpr.br

Resumo* — *Amplificadores de potência desempenham um papel crucial em uma ampla variedade de aplicações, desde sistemas de áudio até comunicações sem fio e eletrônica de potência. Eles são responsáveis por amplificar sinais de baixa potência para níveis adequados de saída, garantindo a transmissão eficiente de sinais em diferentes sistemas. O objetivo deste estudo é aprimorar a compreensão do modelo polinomial da série de Volterra como uma forma de modelar o comportamento de um amplificador de potência, utilizando médias móveis, e analisar, a partir dos resultados obtidos de um estudo anterior, os recursos necessários para a implementação em hardware do modelo, com e sem um filtro de médias móveis. Utilizando a linguagem de descrição de hardware VHDL, foram descritas as operações de soma para o cálculo da média móvel de entrada, com diferentes tamanhos de janela. Foi possível observar como cada filtro influenciava a precisão do modelo e o consumo de recursos lógicos da FPGA. Utilizando a FPGA Virtex5 LX50T, os resultados mostraram que o Erro Quadrático Médio Normalizado (NMSE), quando aplicado às médias móveis nos dados de entrada, apresentou uma variação de até aproximadamente 0,8 dB. Como o hardware do filtro descrito utiliza apenas circuitos somadores, seu design é relativamente simples. Com isso, ao adicionar o filtro de médias móveis a um hardware que realiza as equações do modelo MP, foi possível melhorar a relação entre a resposta esperada e a resposta obtida, utilizando poucos recursos lógicos adicionais.

I. Introdução

A modelagem de circuitos elétricos é importante para o desenvolvimento e análise de sistemas elétricos e eletrônicos. A representação precisa e eficiente do comportamento de um circuito permite simulações e análises mais simples. Um exemplo interessante da aplicação da modelagem em circuitos elétricos é a criação de um modelo matemático para descrever o comportamento de um amplificador de potência, utilizando o modelo de polinômio de memória, que é um caso particular da série de Volterra. Em uma série de Volterra, a saída é influenciada pelas entradas aplicadas nos instantes de tempo atual e passados. Em sistemas com memória desvanecente, como os amplificadores de potência, a influência de uma entrada na saída diminui com o tempo, resultando em uma quantidade limitada de amostras passadas, representada por M, conhecida como duração ou profundidade de memória [2]. O modelo polinômio de memória destaca-se pela sua capacidade de representar com precisão o comportamento dinâmico do amplificador, levando em consideração seus efeitos de memória.

Neste trabalho, é descrito o filtro de médias móveis para a modelagem de um sistema com memória utilizando o modelo polinomial. A modelagem matemática foi realizada utilizando Python e, posteriormente, em VHDL, foi descrito o hardware para o filtro e avaliada a sua viabilidade.

Na segunda seção, é apresentada a modelagem matemática utilizada neste trabalho. Na terceira seção, descreve-se a elaboração do código em Python para a extração dos coeficientes do modelo. A quarta seção trata do hardware desenvolvido e descrição do estudo utilizado como base para o trabalho. Na quinta seção, é discutido sobre resultados do estudo. Na sexta e última parte, são expostas as considerações finais e conclusões do estudo.

II. Modelagem comportamental utilizando polinômio de memória

A. Série de Volterra e Polinômio de memória

A série de Volterra é uma técnica de modelagem utilizada para caracterizar o comportamento não linear de sistemas dinâmicos. Ela permite expressar a saída do sistema como uma função das entradas, levando em consideração interações de diferentes ordens entre elas. Isso possibilita uma descrição mais completa dos efeitos não lineares, incluindo aspectos como saturação, distorção e memória, que são comuns em sistemas como amplificadores de potência. A série de Volterra é especialmente útil para

análise e projeto de sistemas que exibem não linearidades complexas.

Na série de Volterra a saída é expressa como uma função polinomial das entradas. Por exemplo, existem termos de diferentes ordens: os de primeira ordem, onde a saída é proporcional à entrada; os de segunda ordem, onde a saída é proporcional ao produto de duas entradas; os de terceira ordem, onde a saída é proporcional ao produto de três entradas, e assim por diante até os termos de ordem infinita. Na prática, a ordem do polinômio é limitada a P [2]. À medida que os valores de M e P aumentam na série de Volterra, a precisão da modelagem tende a aumentar, enquanto o erro diminui. Isso ocorre porque um maior número de termos na série de Volterra permite uma representação mais detalhada e precisa dos efeitos não lineares do sistema. No entanto, esse aumento na precisão vem com um custo computacional significativo. Conforme M e P crescem, o número de coeficientes na série de Volterra aumenta exponencialmente [3]. Portanto, há uma compensação entre a precisão da modelagem e o custo computacional associado, e é importante encontrar um equilíbrio adequado entre eles, levando em consideração as limitações de hardware e tempo de processamento disponíveis.

O polinômio de memória é um modelo matemático que baseia-se na ideia de que o estado atual de um processo pode ser expresso como uma função não linear do estado anterior. O MP pode ser representado como um modelo linear nos coeficientes, identificável por meio de regressão linear. Assim, com este modelo é possível prever o comportamento futuro de uma série estimando os coeficientes do modelo com técnicas de regressão linear. Essa característica permite ao MP capturar a não linearidade de processos dinâmicos. O modelo MP é descrito pela equação:

$$\tilde{y}(n) = \sum_{p=1}^{P} \sum_{m=0}^{M} b_{\widetilde{2p-1,m}} \left|x(\widetilde{n-m})\right|^{2p-2} x(\widetilde{n-m}) \quad \textbf{(1)}$$

onde x é a entrada, y a saída, b o coeficiente, M o fator de memória e P o de não linearidade.

B. *Filtro de médias móveis*

O cálculo de médias móveis é uma técnica utilizada para suavizar as flutuações e ruídos ao calcular a média de um conjunto de dados em uma janela móvel ao longo do tempo. Ao aplicar essa técnica, valores muito diferentes ou ruídos são atenuados, permitindo uma visão mais clara da tendência dos dados. Isso melhora o erro ao fornecer uma estimativa mais estável e precisa dos valores esperados, reduzindo a variabilidade indesejada e destacando padrões mais consistentes na série temporal [1].

Neste estudo, foi empregado o cálculo das médias móveis em um determinado vetor de entrada de dados, esperando que isso diminua o erro e faça com que a saída do modelo desenvolvido seja mais parecida com a saída esperada, obtida a partir dos dados de entrada utilizados no estudo.

III. Modelo MP em Python

Para a construção do código em Python, foi utilizada a biblioteca '*numpy*' para a parte matemática. Essa biblioteca contém funções para o cálculo com arrays e matrizes, facilitando operações de álgebra linear e outras manipulações numéricas necessárias para a implementação do modelo.

Foi realizada uma regressão linear onde a saída Y é descrita por uma multiplicação de matrizes $XX \cdot H$, onde o número de colunas é o mesmo que o número de coeficientes, e uma matriz coluna contendo os coeficientes, de acordo com:

$$Y = XX \cdot H \quad \textbf{(2)}$$

Para o código, foi elaborado um loop principal que preenchia a matriz XX com os resultados das operações $|(x - m)|^{|(2p-2)|} \cdot x(n - m)$, percorrendo o somatório com os valores de M e P definidos. Com o resultado da matriz de regressão XX, foram ajustados os valores dos coeficientes (matriz H) a partir do comando 'np.linalg'.

Com os resultados de H e XX, foi calculado o erro entre a resposta obtida e a esperada a partir da equação:

$$NMSE = 10\, log_{10}\left[\frac{\sum_{n=1}^{N} |e(n)|^2}{\sum_{n=1}^{N} \left|y_{ref}(n)\right|^2}\right] \quad \textbf{(3)}$$

Para o filtro de média móvel, o loop principal descrito foi modificado para receber os valores de entrada já processados pelo filtro, de acordo com o valor de D arbitrado. O valor D representa a quantidade de valores somados, e cada valor é dividido por $D + 1$ para calcular a média móvel. Para avaliarmos os resultados, realizamos os cálculos com a memória fixada em 1, variando P para 4, 5, 6 e 7, e variando D para 1, 3, 5 e 7.

IV. Descrição do Hardware

Para a descrição do hardware de cálculo das médias móveis, com o objetivo de reduzir o consumo de hardware, a operação de divisão foi transferida para o cálculo dos coeficientes. As operações realizadas pelo filtro podem ser descritas essencialmente como uma série de somas, organizadas de modo a otimizar o processamento. Utilizando um componente somador capaz de realizar a soma de dois valores por vez, a soma total dos valores de entrada é realizada em etapas, somando os valores dois a dois, tanto de forma paralela quanto sequencial. Por exemplo, para $D = 5$ temos a soma de seis valores. As somas são distribuídas em ciclos, permitindo a execução de múltiplas operações de soma em paralelo. No primeiro ciclo por exemplo, são realizadas duas somas simultâneas: $x_0 + x_1$ e $x_2 + x_3$. No segundo ciclo, os resultados das somas anteriores são combinados com os valores x_4 e x_5: $(x_0 + x_1) + x_4$ e $(x_2 + x_3) + x_5$. Finalmente, no terceiro ciclo, as duas somas resultantes são combinadas: $[(x_0 + x_1) + x_4] + [(x_2 + x_3) + x_5]$. Assim, o circuito descrito realiza as somas distribuindo as operações em etapas realizadas paralelamente, no mesmo ciclo de

clock, e sequencialmente. O componente 'somador' utilizado foi o descrito no estudo realizado anteriormente em [5]. A figura 1 mostra um dos circuitos gerados para o cálculo das médias.

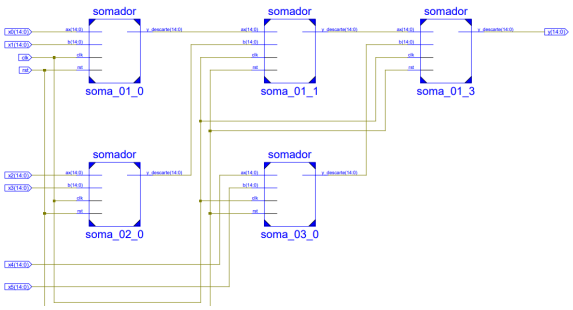

Figura 1- *Circuito do filtro de médias móveis gerado em VHDL para D=5*

Os componentes estão descritos da seguinte forma: '$operação_numero_da_operacao_ciclo_de_clock$', sendo por exemplo $soma_01_0$, o primeiro somador do ciclo 0 de clock , o $soma_01_1$ o primeiro somador do ciclo 1 de clock, e assim por diante.

Como o objetivo era avaliar o impacto do filtro de médias móveis na relação entre erro e consumo de recursos na FPGA, foram utilizados os resultados obtidos por um estudo anterior [4]. O estudo em [4] trata de uma comparação dos recursos de hardware necessários para implementar os modelos da série de Volterra: Polinômio de Memória (MP), Polinômio de Memória de Envoltória (EMP) e Polinômio de Memória e de Memória de Envoltória Combinados (CMEMP). O Polinômio de Memória de Envoltória (EMP) diferencia-se do modelo MP na utilização de apenas a amplitude das amostras passadas para estimar a saída atual [4]. Já o modelo CMEMP, é uma combinação entre o modelo MP e o modelo EMP [4]. O trabalho [4] utilizou a substituição de funções não lineares por uma interpolação linear entre pontos pré-calculados dispostos em uma tabela de busca (LUT). Além disso, foi realizada a conversão de vírgula flutuante para vírgula fixa dos valores utilizados para determinar os coeficientes, onde primeiramente os valores foram normalizados, e em seguida foi definido o número ideal de bits para evitar overflow, incluindo mais um bit de sinal [4] .

Neste trabalho, o objetivo era a análise do impacto do cálculo das médias móveis sobre o modelo polinomial, logo foram utilizados os resultados descritos por [4] referentes ao hardware implementado para o MP. Foi utilizado o mesmo número de bits definidos pelo estudo [4], igual a 15 bits, e também a mesma FPGA. O valor de D foi variado entre 1,3, 5 e 7. Para $D = 1$, foi utilizado um somador simples, operando em um único ciclo de clock. Para $D = 3$, foram necessários dois ciclos, com uma soma realizada em cada ciclo. Para $D = 5$, foram utilizados três ciclos: três somas no primeiro ciclo, uma no segundo e uma no terceiro. Para $D = 7$, também foram necessários três ciclos: três somas no primeiro ciclo, duas no segundo e uma no terceiro.

V. RESULTADOS

Para avaliar a viabilidade da implementação das médias móveis, o código em Python foi executado com variações em P e nos valores de D para a janela das médias móveis. Os resultados obtidos, expressos em NMSE em dB, estão apresentados na Tabela 1.

TABELA 1- VALORES DE NMSE OBTIDOS

Modelo MP com valor de M=1					
P	Sem MM	MM com $D = 1$	MM com $D = 2$	MM com $D = 5$	MM com $D = 7$
4	-21,96259	-22,03694	-22,01239	-22,80819	-20,21370
5	-21,99235	-22,06882	-22,06882	-22,81572	-21,99235
6	-22,03990	-22,13864	-22,13144	-22,86770	-20,25272
7	-22,05014	-22,13895	-22,13432	-22,87543	-20,25392

A partir dos resultados expressos na tabela 1, podemos observar que com $D = 5$, para todos os valores de P simulados, o erro diminui. Quando D ultrapassa 5, o erro passa a aumentar em relação ao calculado com uma janela maior. Já para $D = 7$, o erro foi maior que o calculado com o modelo sem o filtro de médias móveis.

No *hardware* descrito, utilizando 15 bits para a representação de cada entrada e saída, e utilizando a FPGA Virtex5 LX50T, chegamos aos resultados expressos na tabela 2.

TABELA 2- Recursos necessários para a implementação do filtro em VHDL

D	Ciclos de Clock	Flip-Flops	LUTS	nº de somadores
1	1	15	15	2
3	2	45	45	3
5	3	75	75	5
7	3	105	105	7

Portanto, mantendo o fator de memória em 1 e utilizando o fator de não linearidade (P) igual a 5, foi possível obter uma diferença de até 0,8 dB, utilizando $D = 5$. Para avaliar o modelo com as médias móveis, foram utilizados os resultados obtidos em [4], em que foram comparados os recursos necessários para o *hardware* da implementação de modelos da série de Volterra, incluindo o MP. Os resultados expressos na Tabela 2 do trabalho [4] mostram que o modelo MP implementado em FPGA sem o filtro de médias móveis, precisa de 11 ciclos de clock, 122 flip-flops,195 LUTs e 16 unidades DSP48. Sendo assim, para a

implementação do filtro de médias móveis neste estudo, com $D = 5$ aplicado ao modelo descrito em [4], teríamos um circuito final que gera resultados a partir da 13ª batida de clock, utilizando 167 flip-flops e 270 LUTs. Esses números correspondem à soma dos resultados de recurso necessários expressos na Tabela 2 do trabalho [4] com os valores apresentados na Tabela 2 deste artigo.

VI. Conclusão

Este trabalho apresentou um modelo para estimar as saídas de circuitos de memória, como amplificadores de potência, utilizando um polinômio de memória com filtro de médias móveis e posterior implementação em *hardware* utilizando VHDL. Foi observado que, ao incrementar o valor de P, o NMSE apresentou uma melhora significativa. No entanto, esse aprimoramento vem acompanhado de um aumento na complexidade do *hardware*, exigindo um maior número de multiplicações. Por outro lado, ao incorporar o filtro de médias móveis, conseguimos melhorar os resultados sem a necessidade de adicionar mais multiplicações, utilizando um circuito simples de somadores. Essa técnica permite a obtenção de melhorias consideráveis no desempenho do circuito, com um impacto menor nos recursos de *hardware*, pois envolve a adição de apenas alguns somadores ao circuito descrito. Assim, o uso de filtros de médias móveis se apresenta como uma solução eficiente para otimizar o equilíbrio entre a precisão dos resultados e a complexidade do *hardware*.

Referências

[1] DE LIMA, Eduardo Gonçalves. *Behavioral Modeling and Digital Base-Band Predistortion of RF Power Amplifiers*. PhD Thesis, 2009.

[2] BONFIM, Elton John. *Modelagem Comportamental de Amplificadores de Potência de Radiofrequência Usando Termos Unidimensionais e Bidimensionais de Séries de Volterra*. Master's thesis. Universidade Federal do Paraná (UFPR), 2016.

[3] MATHEWS, V. J.; SICURANZA, L. G. *Polynomial Signal Processing*. New York: Wiley, 2000.

[4] SCHUARTZ, L.; WOSNIACK, I.; SCHOULTEN, F.. ; SILVEIRA, E.; DA LUZ FRANÇA, S. B.; DE LIMA, E. G. *Configuração em aritmética de vírgula fixa para síntese em FPGA dos modelos MP, EMP e CMEMP.* Seminário de Microeletrônica do Paraná (SeMicro-PR), Curitiba, 2018

[5] DA ROCHA GASINO, V. M.; DA LUZ FRANÇA, S. B.; DE LIMA, E. G. .*Descrição em VHDL de um Modelo Matemático para Circuito Linear Sem Memória*. Seminário de Microeletrônica do Paraná (SeMicro-PR), Curitiba, 2023