

DA ANÁLISE EM SÉRIE DE FOURIER AO FPGA

Adriano dos Santos Cardoso¹, Alexandre César Rodrigues da Silva² e Aparecido Augusto de Carvalho³

Resumo — Sinais analógicos ou digitais são descritos por funções. Dessa forma, a expansão em séries são métodos válidos para a descrição desses tipos de sinais. A expansão em série de Fourier é, provavelmente, a mais utilizada dentre todas as formas de expansões em séries ortogonais. Atualmente, os recursos disponibilizados nos osciloscópios digitais têm permitido a análise espectral de sinais analógicos, gerando, assim, os coeficientes da Série de Fourier. Os coeficientes da Série de Fourier descrevem a forma de onda sob análise. Neste trabalho, desenvolveu-se uma ferramenta de síntese denominada F2HDL (Fourier para HDL), onde os coeficientes da Série de Fourier, obtidos de um sinal analógico, a partir de uma análise espectral, pode ser implementado em FPGA. Estudos realizados com diferentes tipos de sinais comprovaram que a ferramenta desenvolvida possibilita a geração de formas de onda digitais em uma faixa muito grande de frequência, podendo atingir dezenas de MHz.

Palavras Chave — Análise Espectral, Ferramentas de Síntese, FPGA, HDL, VHDL.

INTRODUÇÃO

O PROCESSO DE SISTEMAS DIGITAIS

Em [2], o processo de projeto foi definido de forma simples como a especificação e a descrição final. A diferença fundamental entre a descrição inicial e a descrição final está no fato desta última conter todas as informações necessárias à fabricação do sistema, ao contrário da primeira. O processo de projeto pode ser dividido em duas etapas distintas: a de síntese, onde se agrega algum tipo de informação à descrição, e, assim, desenvolve-se uma descrição mais próxima da descrição final, e a validação, também chamada de análise, onde é verificado os acertos das tomadas de decisões durante as etapas de síntese.

Durante o processo de projeto, dois conceitos devem ser atendidos: Correção e Otimização. Um projeto é correto se e somente se atende todos os requisitos da descrição inicial e é factível. E, ainda, a descrição de um projeto é ótima se é

correta e ainda possui menor custo e melhor desempenho dentre todas as soluções.

Em resumo, pode-se dividir os critérios em:

- Espaço – O sistema deve ter o menor tamanho possível;
- Tempo – O sistema deve apresentar o menor tempo de resposta possível;
- Consumo de energia – O sistema deve apresentar o menor consumo possível.

Para se escolher uma dentre as soluções ótimas, deve-se adequar critérios de acordo com o tipo de aplicação do sistema.

A partir de uma descrição abstrata o projetista deve ser capaz de uma descrição detalhada de forma tal que o projeto seja, então, ótimo e mantenha a funcionalidade e desempenho a um custo mínimo.

UM MODELO PARA O PROCESSO DE PROJETO:

O DIAGRAMA Y

Em termos de hierarquia, um projeto digital pode ser dividido em quatro níveis ascendentes de abstração:

- nível elétrico – É o mais baixo nível, nele estão contidas informações da teoria de circuitos e da física dos semicondutores, ou ainda, de diagramas elétricos com resistores capacitores, transistores etc;
- nível lógico – Contém informações onde se agrega conceitos da teoria de circuitos digitais tais como equações booleanas, teoria de grafos, tabelas de transição de diagramas de máquinas de estados, etc;
- nível microarquitetura – Carrega informações sobre macroblocos, transferência entre registradores, planta baixa com a localização dos módulos, ULAs, grafos de fluxo de dados, etc;
- nível sistêmico – São descritos algoritmos, fluxogramas, memórias, processadores, etc.

Pode-se decompor as informações de um sistema digital em três diferentes categorias:

- físico - contém informação geométrica sobre os componentes, módulos e sobre a disposição espacial destes no sistema a ser fabricado;
- estrutural - contém informação sobre como interconectar blocos de base de comportamento;

¹ Adriano dos Santos Cardoso, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Av Brasil Centro, 56, Ilha Solteira, SP, Departamento de Engenharia Elétrica, cardoso@dee.feis.unesp.br

² Alexandre César Rodrigues da Silva, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Av Brasil Centro, 56, Ilha Solteira, SP, Departamento de Engenharia Elétrica, acrsilva@dee.feis.unesp.br

³ Aparecido Augusto de Carvalho, Universidade Estadual Paulista “Júlio de Mesquita Filho”, Av Brasil Centro, 56, Ilha Solteira, SP, Departamento de Engenharia Elétrica, aac@dee.feis.unesp.br

- comportamental - carrega informação sobre o comportamento do sistema, sem se envolver em como tal comportamento pode ser obtido, seja do ponto de vista físico, seja do ponto de vista estrutural.

A representação de um modelo que contém os níveis de abstração e os domínios de descrição de um sistema digital foi proposta pela primeira vez por Gajski e Kuhn [1], o chamado Diagrama Y, apresentado na Figura 1.

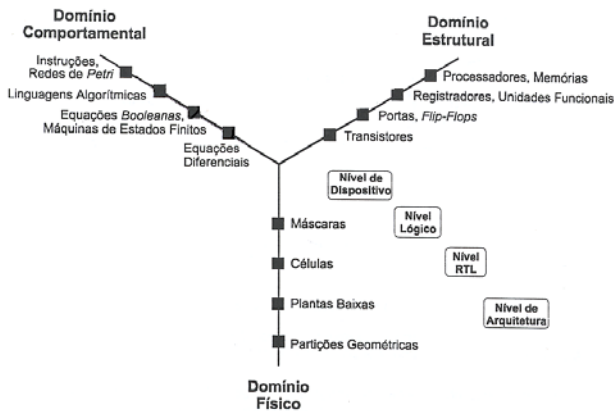


FIGURA 1.

ELEMENTOS PRIMITIVOS DE CADA NÍVEL DE ABSTRAÇÃO NO DIAGRAMA DE GAJSKI E KUHN (DIAGRAMA Y).

Neste diagrama, os níveis de abstração (Nível de Dispositivo, Nível Lógico, Nível RTL, Nível de Arquitetura) são identificados pelas suas distâncias do centro do diagrama, enquanto os segmentos de reta radiais correspondem a domínios de descrição. Por exemplo, um diagrama de esquemáticos de portas lógicas TTL é uma descrição estrutural lógica, estando portanto localizado na intersecção do eixo rotulado Domínio Estrutural com o nível lógico de abstração.

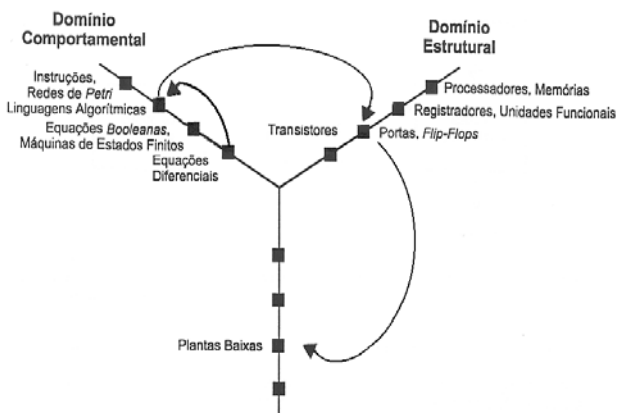


FIGURA 2.

REPRESENTAÇÃO DAS TRANSIÇÕES OCORRIDAS COM A F2HDL.

A Figura 2 representa o diagrama de Gajski da ferramenta desenvolvida. A F2HDL leva do nível de abstração de Equações Algébricas para o nível de Equações Booleanas. O ambiente de projeto MAX + Plus II leva para o nível de portas lógicas (Compilador) e finalmente, para o nível de Planta Baixa (Configurador de FPGA).

FERRAMENTAS DE PROJETO BASEADAS EM SÍNTESE AUTOMÁTICA

Métodos de síntese automatizada ótima para equações Booleanas datam das décadas de 50 e 60 e não tinham a capacidade de resolver sistemas muito complexos. A necessidade de manipular quantidades crescentes de informações durante o projeto de sistemas digitais exigia a consideração constante de novas ferramentas. Estas deveriam transcender as atividades de facilitar a captura e de exercitar descrições, passando a ser capazes de gerar novas descrições de forma automatizada e corretas por construção, baseados nos mesmos requisitos manipulados pelos projetistas. O estilo de projeto estabelecido por estas novas ferramentas baseia-se no uso de ferramentas computacionais e guia o processo de projeto. A ferramenta central possui embutido um conjunto de modelos de síntese e é capaz de gerar uma descrição correta por construção, bem como uma avaliação do desempenho desta descrição. O laço de realimentação é fechado pelo projetista, que julga os resultados da síntese e aceita descrição, ou a rejeita e escolhe um novo modelo de síntese para ser usado. Essa escolha pode ser feita com o auxílio de outras ferramentas distintas e apropriadas no processo de validação.

As ferramentas de simulação podem ainda ser empregadas visando fornecer mais detalhes sobre a implementação. Frequentemente, o sintetizador gera uma descrição a partir dos requisitos, e refina esta descrição através de técnicas de otimização. Somente a descrição resultante deste processo é fornecida ao projetista.

CRIAÇÃO DE DECODIFICADORES DIGITAIS UTILIZANDO FERRAMENTAS DE SÍNTESE AUTOMÁTICA

SISTEMAS DIGITAIS

Um sistema digital é um sistema discreto, isto é, transforma valores discretos de entrada em valores discretos de saída. Ele trabalha aplicando operações ou transformações nos valores de entrada. Os resultados dessas operações são passados a outras operações e finalmente para os valores de saída.

A descrição através de uma interpretação funcional do sistema é chamado de Descrição Comportamental. Em linguagem de descrição de hardware, a descrição comportamental é diretamente integrada na linguagem. Assim, pode-se trabalhar em uma descrição com um nível mais alto de abstração.

As ferramentas de síntese, trabalham em torno de modelos funcionais que representam algum conjunto de funções.

Um dos modelos que se pode trabalhar nessas ferramentas são os decodificadores. Estes representam funções booleanas. Assim, cada um dos pinos de saída é uma função de todas as entradas.

Um decodificador de 7 segmentos representa bem esta ideia, ele possui 4 entradas (A, B, C, D) e 7 saídas (a, b, c, d, e, f, g) e a saída "a", por exemplo, pode ser transcrita como:

$$a = F(A, B, C, D) = \Sigma(0,2,3,5,6,7,8,10,12,14,15) \quad (1)$$

Extrapolando para as outras saídas, determina-se as funções b, c, d, e, f, g e aplica-se os processos de minimização (fornecidas pelo ambiente de projeto Max + Plus II da Altera).

Em termos de linguagem de descrição de hardware (HDL), utilizou-se como base o uso de tabelas, assim, descreve-se o comportamento de uma função em todos os valores (passos) possíveis conforme o exemplo da Figura 3.

IMPLEMENTAÇÃO DE FORMAS DE ONDA ANALÓGICAS

O intervalo de valores possíveis de um contador, depende da quantidade de flip-flops que se está trabalhando, ou de outra forma, do número de bits. Assim, um contador de $n = 8$ bits, por exemplo, apresenta em função da contagem, valores em sua saída entre 0 e 255 ou seja $2^n - 1$, onde n é o número de bits. Desta forma, a cada pulso de relógio a contagem aumenta em uma unidade percorrendo todos os valores de forma crescente. Acoplando-se a saída

do contador a um decodificador com o mesmo número de bits, a cada valor de contagem, tem-se no barramento de saída do decodificador valores em função da contagem atual do contador. A frequência de saída do sistema, depende do número de bits e do relógio conforme descrito em (2). A Figura 4, representa o sistema que gera uma onda analógica partindo de recursos digitais.

$$f_{\text{signal}} = \frac{f_{\text{clock}}}{2^n} \quad (2)$$

Onde:

n = número de bits do contador;

f_{clock} = frequência do relógio (entrada);

f_{signal} = frequência do sinal (saída);

```

SUBDESIGN 7segment
(
    i[3..0]      : INPUT;
    a,b,c,d,e,f,g : OUTPUT;
)
BEGIN
    TABLE
        i[3..0] => a,b,c,d,e,f,g;
        H"0"   => 1,1,1,1,1,0;
        H"1"   => 0,1,1,0,0,0;
        H"2"   => 1,1,0,1,1,0;
        H"3"   => 1,1,1,1,0,0;
        H"4"   => 0,1,1,0,0,1;
        H"5"   => 1,0,1,1,0,1;
        H"6"   => 1,0,1,1,1,1;
        H"7"   => 1,1,1,0,0,0;
        H"8"   => 1,1,1,1,1,1;
        H"9"   => 1,1,1,1,0,1;
        H"A"   => 1,1,1,0,1,1;
        H"B"   => 0,0,1,1,1,1;
        H"C"   => 1,0,0,1,1,0;
        H"D"   => 0,1,1,1,0,1;
        H"E"   => 1,0,0,1,1,1;
        H"F"   => 1,0,0,0,1,1;
    END TABLE;
END;
    
```

FIGURA 3. DESCRIÇÃO EM AHDL DO DECODIFICADOR DE 7 SEGMENTOS

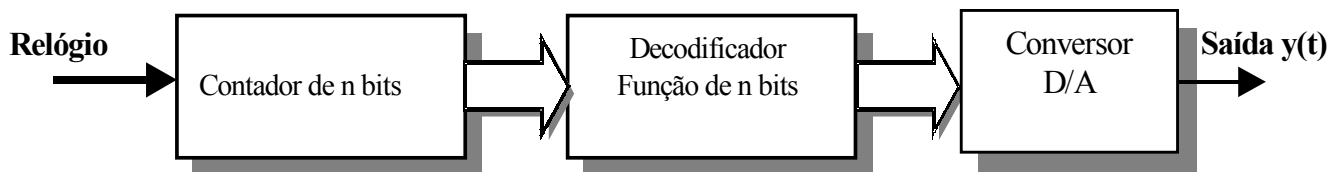


FIGURA 4.
GERAÇÃO DE SINAL ANALÓGICO UTILIZANDO DECODIFICADOR

ENGENHARIA REVERSA

A utilização da tecnologia FPGA para a geração e implementação de formas de onda foi descrita em [3]-[4].

Algumas formas de onda podem ser facilmente descritas por meio de equações e, de modo particular, com os coeficientes de um polinômio. Outras, porém, podem apresentar uma forma um tanto complexa para esse tipo de modelagem, ou ainda, pode-se ter apenas a forma visual (ou capturada por Osciloscópio Digital) para a possível reprodução. Deste problema, surgiu a necessidade de se gerar formas de onda, partindo da análise espectral que é facilmente gerada pelos softwares que fazem análises de formas de ondas capturadas por osciloscópios digitais. Uma vez capturada a forma de onda que se quer reproduzir, pode-se efetuar a análise espectral e os resultados serão a entrada para uma Ferramenta de Síntese. Através da superposição dos coeficientes e os respectivos pesos multiplicados por funções senos e cosenos, pode-se reproduzir a forma desejada. Conforme a necessidade, pode-se, ainda, acrescentar ou diminuir a quantidade de bits da forma de onda ou, também, alterar a quantidade de dados da análise espectral fornecida à ferramenta F2HDL.

MATERIAIS E MÉTODOS

Para a geração de formas de onda usando somente recursos digitais, desenvolveu-se uma ferramenta de síntese automática onde a partir da descrição da forma de onda pode-se gerar a descrição em termos de elementos digitais.

A ferramenta de síntese F2HDL gera o arquivo texto de HDL.

O ambiente de programação MAX + Plus II da Altera, simula e compila e gera vários arquivos de reportagem e de interface com outros ambientes de programação.

O aplicativo TBL2M, foi desenvolvido para interpretar dados do arquivo .TBL (arquivo de simulação) e convertê-los em um arquivo gráfico do MatLab.

A Figura 5, apresenta o fluxo do processo de geração de arquivos utilizado para implementação e validação das formas de onda. E, de forma geral, podemos descrever os passos para a geração de formas de onda como:

1. Descrição da forma de onda (análise espectral);

2. Utilização de uma das ferramentas de síntese para gerar o arquivo HDL (AHDL ou VHDL);
3. Compilação, Simulação ;
4. Criação do Arquivo de Simulação TBL;
5. Execução do Programa TBL2M;
6. Visualização da forma de onda proposta no MatLab;
7. Validação ou correção do Circuito;

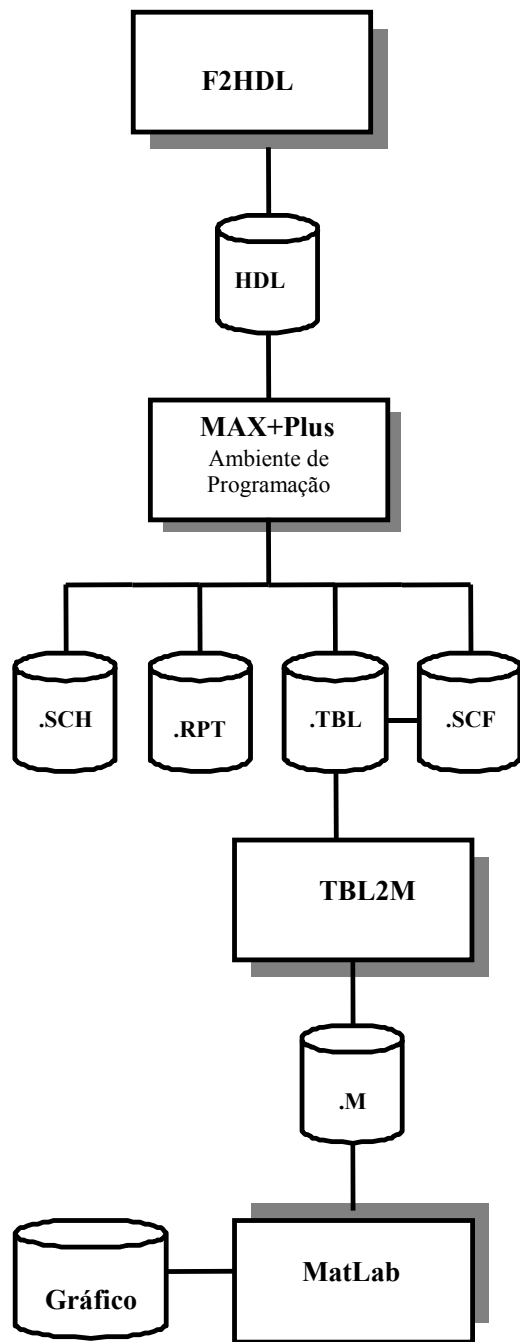


FIGURA 5.
ESQUEMA DE PROCESSO DE GERAÇÃO DE FORMAS DE ONDA EM FPGA

RESULTADOS

A Figura 6, apresenta o resultado de um processo de geração de forma de onda com os dados de entrada apresentado na Tabela I. Na figura 6, a rampa representa a contagem do contador. Quando esta atinge o valor máximo, automaticamente é zerado dando início a um novo ciclo. A

forma de onda apresentada é a composição espectral, considerando-se os ângulos de fase, e as respectivas amplitudes de cada harmônica.

TABELA I
Valores de Entrada

N. da Harmônica	Amplitude (p.u)	Defasagem (rad)
1	1	0
2	1	0.7

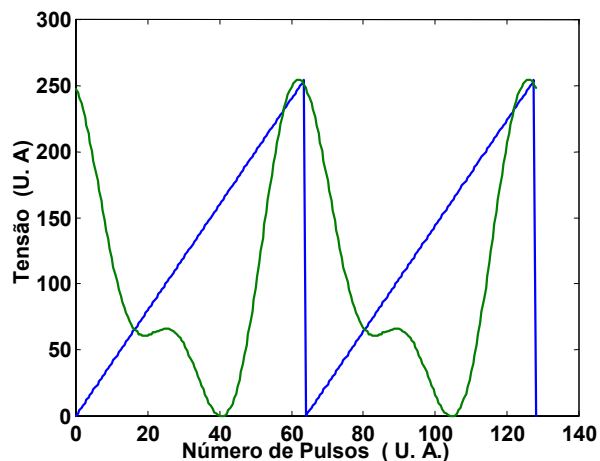


FIGURA 6.
FORMA DE ONDA IMPLEMENTADA EM FPGA

CONCLUSÕES

As ferramentas de síntese automática, desenvolvidas nesse trabalho, apresentaram-se como poderosas ferramentas para a geração de circuitos digitais.

A ferramenta F2HDL apresenta uma nova maneira de se obter uma forma de onda. Através da engenharia reversa pode-se capturar um sinal periódico qualquer, fazer a análise espectral e a partir daí gerar o circuito digital em FPGA correspondente.

A estrutura em hierarquia de projeto permite que uma vez necessária a modificação da forma de onda previamente descrita, é fácil a substituição desta por uma outra mais adequada dentro do projeto.

As frequências de trabalho realizadas nas medições e simulações permitem a implementação física das formas de onda, uma vez que o objetivo é a aplicação em sistemas de Estimuladores Neuromusculares onde a frequência de trabalho é quase sempre menor que 10 KHz.

AGRADECIMENTOS

Este trabalho foi parcialmente financiado pela agência Capes, Brasil e pelo Departamento de Engenharia Elétrica – Universidade Estadual Paulista.

REFERÊNCIAS

- [1] Gajski, D.D & Kuhn, R. H. “*New VLSI Tools*”. IEEE Computer, New York, 16 (12): 11-14, Dezembro 1993.
- [2] <http://www.inf.pucri.br/~moraes/org/metodos.pdf>, acessado em 29/08/2002.
- [3] A. C.R. Silva, A. S. Cardoso, “*A new Environment to Generation of analog Waveform Using AHDL*”. In: SBMICRO 2000 - XV International Conference on Microelectronics and Packaging. ,Manaus -MA –Brasil, Setembro de 2000.
- [4] A. C.R. Silva, A. S. Cardoso, “*An Environment to aid the Synthesis of Threephase Analogue Waveform Using AHDL*”. In: SBCCI 2001 - 14th Symposium on Integrated Circuits and Systems Design, Brasília - DF, Brasil, Setembro de 2001.
- [5] Altera, Max + Plus II AHDL, Versão 6, Novembro de 1995.
- [6] Altera, Max + Plus II VHDL, Versão 6, Novembro de 1995.
- [7] Robinson, J., “*Eletroterapia e teste fisiológico*” – Editora Artmed – Segunda Edição, 2001.