

# Implementação de estrutura de reverberação de áudio digital em FPGA

Diego V. C. do Nascimento<sup>1</sup>, Valentin Obac Roda<sup>1</sup>, José A. N. de Oliveira<sup>1</sup>

<sup>1</sup>Departamento de Engenharia Elétrica – Universidade Federal do Rio Grande do Norte  
Natal, Rio Grande do Norte, 59072-970, Brasil

diego.cirilo@ifrn.edu.br, valentin@ct.ufrn.br, nicolau@ufrnet.br

**Abstract.** *This paper presents a novel implementation in FPGA (Field Programmable Gate Array) of a standard architecture for digital audio reverberation, using allpass filters, as proposed by Manfred Schroeder, presenting the use of reconfigurable hardware as a platform for development and implementation of digital audio effects, focusing on the characteristics of modularity and reuse.*

**Resumo.** *Neste artigo é apresentada uma implementação inédita em FPGA (Field Programmable Gate Array) de uma estrutura de reverberação digital de áudio clássica, usando filtros allpass, baseada na proposta de Manfred Schroeder, demonstrando a utilização de hardware reconfigurável como plataforma de desenvolvimento e implementação de efeitos digitais de áudio, com foco nas características de modularidade e de reuso.*

## 1. Introdução

O efeito de reverberação ocorre pela reflexão do som nas superfícies adjacentes à fonte sonora durante sua propagação até o ouvinte e a resposta ao impulso de um ambiente representa suas características de reverberação. Um exemplo pode ser observado na Figura 1. Por ser dependente do ambiente, a reverberação leva ao ouvinte características do espaço onde o som está sendo reproduzido e comumente sua ausência não soa como “natural”. Como nem sempre é possível ter características desejáveis de reverberação em gravações, métodos para reverberação artificial vêm sendo desenvolvidos há décadas [Valimaki et al., 2012].

O menor custo, maior portabilidade e confiabilidade dos equipamentos digitais levou parte do processamento de áudio para tais plataformas [Owsinski, 2006], e um dos primeiros trabalhos nesse sentido foi o de Manfred Schroeder que propôs uma estrutura de reverberação de áudio digital em seu artigo “*Colorless artificial reverberation*” [Schroeder and Logan, 1961] [Zölzer, 1997]. Uma das possibilidades para processamento digital de áudio é o FPGA (*Field Programmable Gate Array*), um dispositivo

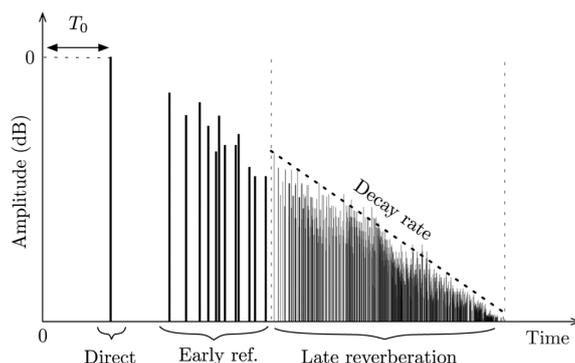


Figura 1: Resposta ao impulso de um ambiente genérico [Valimaki et al., 2012].

reconfigurável que sintetiza arquiteturas digitais descritas por meio de uma linguagem de descrição de *hardware*. Dentre as vantagens da utilização do FPGA, convém destacar a velocidade na implementação e teste de sistemas, modularidade e possibilidade de reuso [Rodriguez-Andina et al., 2007].

Considerando o exposto, neste artigo é apresentada a implementação em FPGA da estrutura de reverberação proposta por Schroeder.

## 2. Estruturas de reverberação de Schroeder

Em seu artigo, Schroeder apresenta a utilização de filtros *allpass* em cascata como alternativa para obter uma reverberação mais natural, se comparada ao método anterior, o uso de filtros *comb*. O grande problema de tais filtros é a sua resposta de frequência em formato de “pente”, que apresenta picos de ganho em algumas frequências resultando em um som dito “metálico”.

Os filtros *allpass* apresentam uma resposta em magnitude plana para todas as frequências, resultando em um áudio sem “coloração”. O filtro em questão pode ser implementado pela seguinte função de transferência, onde  $M$  é o atraso e  $g$  o ganho de realimentação:

$$H(z) = \frac{z^{-M} - g}{1 - gz^{-M}} \quad (1)$$

Para se aproximar da reverberação real o último passo foi a utilização de vários filtros em cascata, de forma a aumentar a densidade e reduzir a periodicidade das reflexões. [Schroeder and Logan, 1961]. Essa arquitetura, uma sequência de *allpass-filters* em cascata, é a arquitetura implementada em FPGA neste artigo.

## 3. Implementação

Para a implementação dos filtros apresentados em FPGA foi utilizado o *kit* de desenvolvimento *Digilent Atlys*, que entre vários dispositivos conta com um CODEC de áudio *National Semiconductor LM4550*, que segue o padrão *Intel AC’97*, trabalhando com 18 bits de resolução em até 48 kHz de amostragem [National Semiconductor, 2002]. O FPGA disponível no *kit* é o *Xilinx Spartan-6 LX45* [Digilent Incorporated, 2012]. O sistema foi descrito em VHDL (*VHSIC (Very High Speed Integrated Circuit) Hardware Description Language*) no ambiente proprietário da *Xilinx*, fabricante do FPGA. A Figura 2 mostra um diagrama de blocos do sistema em questão.

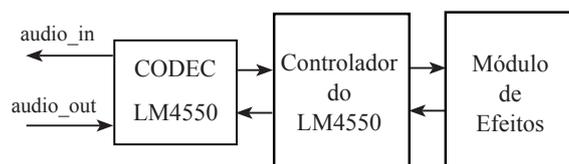


Figura 2: Diagrama de Blocos

O bloco de controle do LM4550 é responsável pelo protocolo de comunicação e configuração do CODEC de áudio, baseado na especificação do padrão AC’97.

O módulo de efeitos é responsável por implementar os filtros *allpass* necessários. O ponto mais importante do bloco é a estrutura para os longos atrasos nos sinais necessários para o efeito. Para os atrasos foi utilizado um *buffer* circular, implementado por

um bloco DPRAM com 64 KB de capacidade, o suficiente para um atraso de cerca de 500 ms a uma taxa de amostragem de 48 kHz e uma resolução de 18 bits. Para atrasos maiores seria necessário a redução da taxa de amostragem.

Seu funcionamento consiste na escrita de dados em posições incrementais, implementada por um contador simples, e a leitura com um *offset* que é dado pelo tamanho do atraso necessário. Como é possível ter atrasos diferentes apenas lendo posições diferentes, é possível replicar as estruturas usando um mesmo *buffer* circular, o que reduz o uso de blocos de memória do FPGA. Os sinais passam ainda por estágios de ganho e somadores, sendo uma implementação direta na linguagem VHDL da Equação 2, onde  $x(n)$  é o sinal de entrada,  $y(n)$  o sinal de saída,  $g$  o ganho e  $d(n)$  o próprio sinal  $y(n)$  atrasado pelo *buffer* circular na amostra anterior.

$$y(n) = g * (x(n) - g * d(n)) + d(n) \quad (2)$$

#### 4. Resultados

Os testes foram realizados aplicando-se um sinal de áudio muito curto, simulando um impulso, na entrada do sistema e lendo-se a saída de áudio. O sinal de entrada foi gerado pelo *software Audacity*, assim como o mesmo *software* foi responsável pela gravação e corte do áudio de saída. O *software MATLAB* foi utilizado para importar os arquivos .WAV gerados pelo *Audacity* e gerar os gráficos aqui apresentados.

A Figura 3 apresenta os sinais de saída do sistema utilizando apenas um *allpass filter* e com um atraso de 200 ms, escolhido para melhor visualização. Utilizando o a possibilidade de obter a leitura de sinais em atrasos distintos devido o uso de um *buffer* circular, foram adicionados mais blocos *allpass* em série, para que a densidade de reflexões aumentasse e a característica periódica dessas reflexões fosse reduzida. O resultado para quatro blocos, apresentado na Figura 4 respectivamente com atrasos de 50, 30, 200 e 10 ms. Por fim, oito blocos *allpass* foram utilizados, e na Figura 5 é possível notar a diferença na densidade de reflexões em relação aos dois exemplos anteriores.

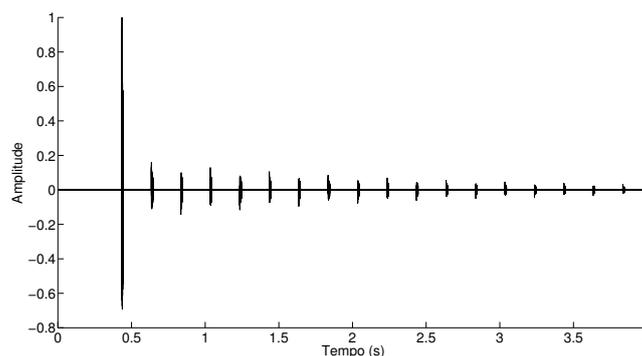
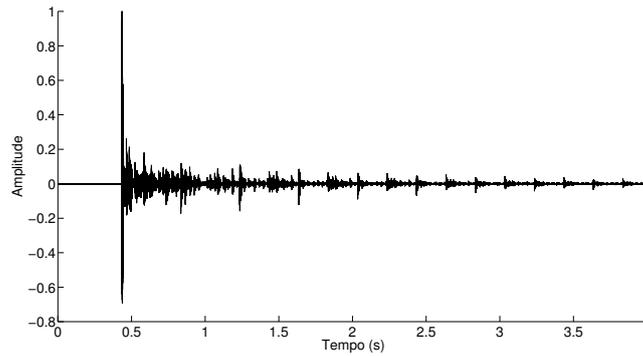


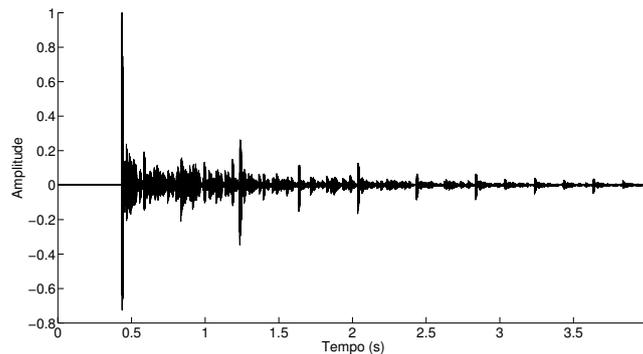
Figura 3: Resposta ao impulso,  $g = 0,9$ , atraso de 200 ms

#### 5. Conclusão

Nesse artigo foi apresentada a implementação de uma estrutura de reverberação em *hardware* e algumas variações, utilizando um dispositivo FPGA. Foi possível observar o funcionamento da implementação por meio dos experimentos e medições realizados, sendo apresentados neste artigo na forma de gráficos. Observou-se a conformidade entre os resultados obtidos e os resultados esperados, de acordo com Schroeder e Logan (1961).



**Figura 4: Resposta ao impulso,  $g = 0,9$ , atrasos de 50, 30, 200 e 10 ms**



**Figura 5: Resposta ao impulso,  $g = 0,9$ , atrasos de 50, 30, 200, 10, 80, 1, 5, 100 ms**

O FPGA se mostra como uma ferramenta de prototipagem rápida para *hardware* de efeitos de áudio digital, além de ser mais barata e menos complexa de utilizar que o fluxo tradicional com componentes discretos. As características de reuso da plataforma podem ser exploradas em trabalhos futuros, onde os blocos de reverberação aqui descritos podem ser combinados com outros efeitos em uma unidade multi-efeitos e mesmo o *buffer* circular, elemento básico de atraso utilizado neste trabalho, pode ser utilizado para outros efeitos baseados em atraso, como *flanger*, *vibrato*, entre outros.

## Referências

- Digilent Incorporated (2012). *Atlys Board Reference Manual*.
- National Semiconductor (2002). *LM4550 Data Sheet*. National Semiconductor.
- Owsinski, B. (2006). *The Mixing Engineer's Handbook*. Course Technology Ptr.
- Rodriguez-Andina, J., Moure, M., and Valdes, M. (2007). Features, design tools, and application domains of FPGAs. *IEEE Transactions on Industrial Electronics*, 54(4):1810–1823.
- Schroeder, M. and Logan, B. (1961). Colorless artificial reverberation. *IRE Transactions on Audio*, AU-9(6):209 – 214.
- Valimaki, V., Parker, J., Savioja, L., Smith, J., and Abel, J. (2012). Fifty years of artificial reverberation. *Audio, Speech, and Language Processing, IEEE Transactions on*, 20(5):1421–1448.
- Zölzer, U. (1997). *Digital Audio Signal Processing*. John Wiley & Sons.