



Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Departamento de Engenharia Elétrica
Processamento Digitais de Sinais
Prof. Dr. Edmar Candeia Gurjão
Aluno: Antonio Felipe de Melo Neto | Matrícula: 121111174

ATIVIDADE:
Filtro IIR

Campina Grande, PB

1. ATIVIDADE

No arquivo [sinal_1.wav](#) foi gravado um sinal composto de soma de cossenos amostrados a uma frequência de 16 kHz. Deseja-se conhecer quantas componentes (cossenos) formam esse sinal;

Também foi solicitado um sistema que separe as componentes do sinal gravado.

Sua tarefa é atender as solicitações acima entregando um conjunto de filtros que separe as componentes do sinal em gráficos separados (você só dispõe de filtros IIR);

2. Introdução

O processamento de sinais desempenha um papel crucial em diversas áreas, proporcionando ferramentas poderosas para análise, modificação e interpretação de dados. Neste contexto, a utilização de filtros digitais se destaca como uma técnica essencial para extrair e manipular informações específicas presentes em sinais complexos. Este relatório aborda a aplicação de filtros passa-banda no processamento de um sinal composto, que é a soma de cossenos amostrados a uma frequência de 16 kHz. O objetivo é separar e analisar as diferentes componentes de frequência presentes no sinal original.

O experimento utiliza um arquivo de áudio como entrada, representando um desafio prático comum em processamento de sinais. O sinal composto é composto por cossenos de frequências específicas, e a aplicação de filtros passa-banda visa isolar cada componente para uma análise mais aprofundada. Serão explorados conceitos como o projeto de filtros digitais, a implementação prática em Python e a visualização das componentes do sinal antes e depois da filtragem. Este estudo é relevante para compreender como os filtros podem ser empregados na extração seletiva de informações em sinais complexos, sendo útil em diversas aplicações, como processamento de áudio, comunicações e análise de dados.

3. Procedimentos Experimental

O experimento teve como objetivo a aplicação de filtros passa-banda para analisar e separar as diferentes componentes de frequência presentes em um sinal composto, que consiste na soma de cossenos amostrados a uma frequência de 16 kHz. O procedimento experimental foi realizado em etapas distintas, envolvendo desde a carga do sinal de áudio até a visualização das componentes filtradas. A seguir, descreve-se detalhadamente cada fase do experimento:

Carregamento do Sinal de Áudio:

- Utilização da biblioteca `scipy` em Python para carregar um arquivo de áudio no formato WAV.
- O caminho do arquivo foi especificado corretamente para garantir a leitura adequada do sinal.

Projeto do Filtro Passa-Banda:

- Implementação de uma função para projetar um filtro passa-banda utilizando a função `butter` da biblioteca `scipy`.
- Especificação dos parâmetros do filtro, incluindo a frequência de corte inferior e superior.

Aplicação do Filtro ao Sinal Original:

- Utilização da função `lfilter` para aplicar o filtro passa-banda ao sinal original.
- Obtenção do sinal filtrado para a componente de frequência desejada (2 kHz no exemplo).

Visualização dos Resultados:

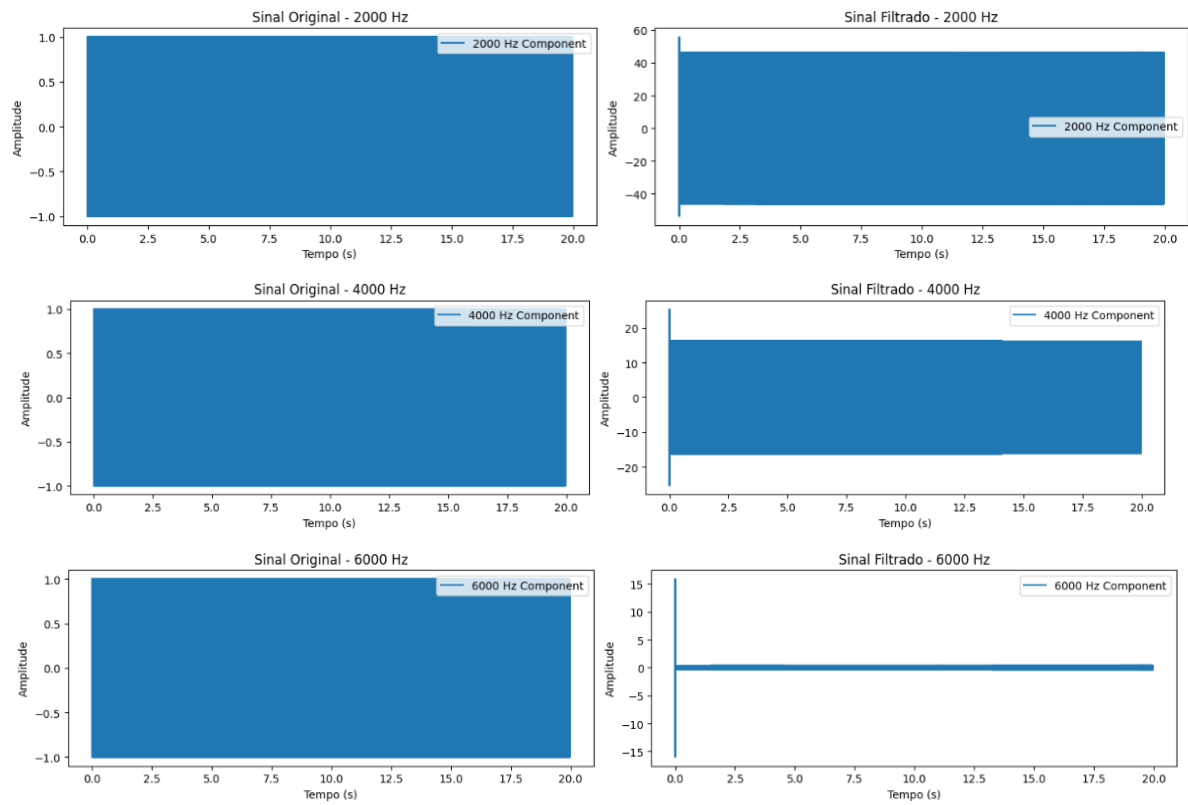
- Utilização da biblioteca `matplotlib` para criar gráficos e visualizar os resultados.
- Geração de subplots para comparar o sinal original e o sinal filtrado para cada componente de frequência.

Análise das Componentes do Sinal:

- Utilização de instruções de impressão para exibir as primeiras amostras das componentes do sinal original e do sinal filtrado.

O procedimento experimental, conforme descrito, proporciona uma abordagem sistemática para a aplicação de filtros passa-banda em um contexto prático, utilizando um sinal de áudio composto. O código em Python, a partir da biblioteca `scipy`, possibilita a implementação eficiente e a visualização clara dos resultados, permitindo a análise das diferentes componentes de frequência do sinal original e do sinal filtrado.

Por fim, obtemos os seguintes resultados:



- Para melhor análise da atividade, o código do programa em python, está disposto no link, que indica para o gitHub <>