



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

RELATÓRIO:

Implementação da FFT por decimação no tempo

1. INTRODUÇÃO

O presente trabalho, que será melhor descrito abaixo, encarrega-se de implementar a FFT (Fast Fourier Transform), ou simplesmente transformada rápida de Fourier, por decimação no tempo. Sua centralização idealizadora, rotaciona em torno da otimização do código de programação da FFT, de modo a ser implementada uma forma mais rápida e consistente de resolução dos problemas.

2. DESCRIÇÃO DA IMPLEMENTAÇÃO

Utilizando-se da linguagem de programação python, em execução no google colab, no desenvolvimento do código, se fez presente a seguinte formatação, indicada abaixo, para a resolução da FFT.

Assim como indicado pela literatura, para realizar-se o cálculo da transformada rápida, separamos o sinal em duas componentes, pares e ímpares, uma vez que conseguimos realizar a soma dos somatórios dessas duas componentes, e entregar ao usuário o cálculo da FFT completa. De maneira a garantir que o cálculo fosse realizado sem danos ao resultado, foi testado o tamanho do N indicado, uma vez que só iria retornar o valor da FFT, caso esse fosse maior que 1, lembrando-se que o N, deve ser $2^l = N$.

Quando testados os pontos que poderiam falhar, recebendo-se a resposta positivamente, deu-se continuidade ao procedimento, desta vez, realizando o cálculo da sequência de números complexos que representam os fatores de rotação, seguindo posteriormente a impressão dos valores obtidos, como também, plotagem desses, em um gráfico que relaciona a amplitude com a frequência do sinal.

Para melhor acompanhamento e análise do trabalho realizado e resultados obtidos, segue o link de acesso ao programa que realiza o cálculo da FFT, anexado junto ao github.

3. APRESENTAÇÃO DE RESULTADOS

- a. Matematicamente, após os cálculos, foram obtidos os seguintes resultados:

```
Transformada de Fourier dos dados de entrada:  
(120+0j)  
(-8.000000000000007+40.218715937006785j)  
(-8.000000000000007+19.31370849898476j)  
(-8.000000000000005+11.97284610132391j)  
(-8.000000000000005+7.999999999999964j)  
(-8+5.345429103354388j)  
(-8.000000000000002+3.313708498984756j)  
(-8+1.5912989390372623j)  
(-8-7.83773951454306e-15j)  
(-7.999999999999997-1.5912989390372658j)  
(-7.999999999999998-3.3137084989847647j)  
(-7.999999999999993-5.345429103354395j)  
(-7.999999999999994-8.000000000000004j)  
(-7.999999999999998-11.972846101323913j)  
(-7.999999999999989-19.31370849898476j)  
(-7.999999999999997-40.21871593700678j)
```

- b. Tomando os resultados obtidos, matematicamente, pode-se realizar a plotagem do gráfico de frequência versus Amplitude

