

MLV - Music Light Visualizer

Augusto Carvalho de Freitas
augustofreitas@alunos.utfpr.edu.br
(19) 99763-8787

Guilherme Libério de Souza e Silva
guilhermesilva.2018@alunos.utfpr.edu.br
(67) 98145-5425

Henrique Nelson da Silva Vanin
henriquevanin@alunos.utfpr.edu.br
(41) 99285-7522

Agosto 2019

1 Introdução

Este trabalho destina-se a documentar a produção de um interpretador de frequências sonoras em cores. Serão descritos de forma objetiva os marcos a serem alcançados ao longo de seu desenvolvimento, bem como aspectos logísticos como cronograma, orçamento, planejamento e divisão de trabalho.

O projeto consiste na construção de um visualizador de frequência através de fitas de *LED* RGB. Utilizando um microcontrolador *Arduino*, será feito um sistema capaz de associar uma cor específica do espectro de luz à uma frequência da música captada pelo microfone do computador.

2 Desenvolvimento

Por meio de um software desenvolvido em *Python*, utilizando a biblioteca *pyaudio*, o sinal digital da música será captado pelo microfone do computador e extraído sua frequência mais dominante por meio da Transformada de Fourier. Posteriormente um programa feito na *IDE* do *Arduino*[1] irá mapear as diferentes informações recebidas do computador em diferentes tensões.

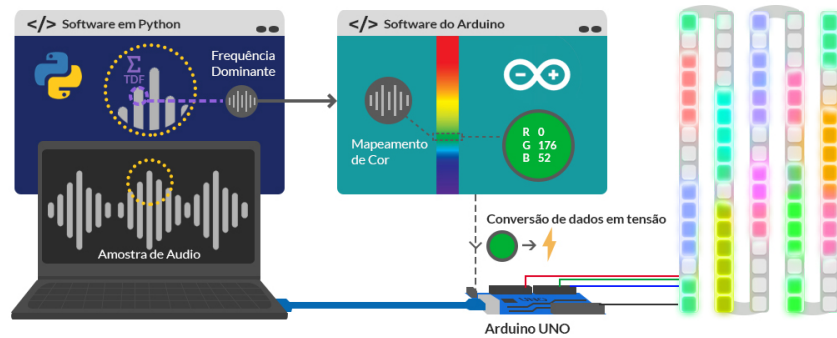


Figure 1: Fluxograma ilustrativo do projeto

O código de análise de dados das músicas terá suporte para reprodutores como *Spotify*. Ele busca otimizar a extração de dados, resgatando apenas as frequências mais relevantes para uma determinada amostragem de dados. Para isso será utilizada uma biblioteca chamada *numpy*, a qual tem um algoritmo da FFT (*Fast Fourier Transform*), para o cálculo da Transformada Discreta de Fourier[2].

A Transformada Discreta de Fourier registra os principais valores de frequência obtido numa faixa da música. Em termos físicos, é uma combinação de ondas capaz de representar de forma eficiente um número maior de informações. Portanto, este processo destina-se a otimizar a passagem de dados a fim de transferir os dados da música de maneira mais fidedigna [3].

O *Arduino* recebe a informação condensada por meio de uma entrada USB realizando o mapeamento das frequências de acordo com o espectro de luz visível. O microcontrolador será responsável por converter os dados de entrada em tensão de saída, necessária para a resposta específica dos *LEDs*, utilizando de um circuito simples que contém apenas um capacitor de 10uF e um resistor de 330 (ohms).

A correlação será feita, primeiramente, de acordo com a magnitude de cada informação, ou seja, um sinal agudo, de alta frequência, será associado a uma matiz violeta, enquanto um sinal grave com o vermelho. Contudo, este mapeamento ainda será analisado a obter um resultado visual adequado. A parametrização das frequências sonoras em luminosas será feita de forma suave e não estática, ou seja, para diferentes sons graves, diferentes tonalidades de um mesmo matiz podem ser representadas.

Foi feita uma tabela relacionando as distribuições de tarefas requeridas pelo trabalho. Uma porcentagem base de participação nas três áreas foi estabelecida para cada membro, sendo dividida o restante de forma a especificar os diferentes campos a serem focados por cada um conforme a Figura 2.

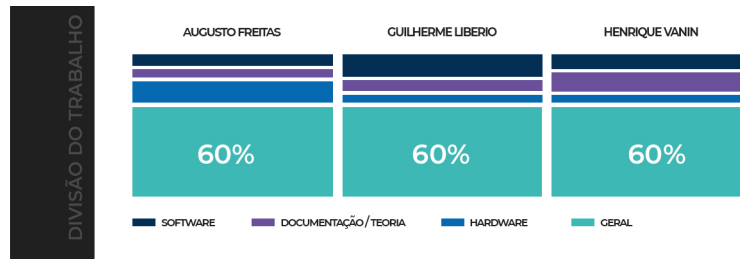


Figure 2: Divisão do Trabalho entre os membros da Equipe

3 Objetivos

Principais marcos:

- Desenvolver um programa em *Python* responsável por interpretar sinais digitais para abstrair o pico da frequência mais relevante da música em um determinado intervalo de tempo;
- Desenvolver um programa na *IDE* do *Arduino* para interpretação de frequência e emissão de cor na fita de *LED*;
- Realizar a integração de hardware e software de forma efetiva.

Ideias a serem trabalhadas ao longo do projeto:

- Possibilidade da integração de um módulo *bluetooth* capaz de transferir os dados da música para o arduino;

4 Cronograma

Semanas	Cronograma
19 ago	Definição de equipe e ideia de projeto
26 ago	Concepção de projeto e pesquisas necessárias
02 set	Diagrama completo e proposta inicial
09 set	Aquisição de Componentes e Introdução ao Software e à teoria Física
16 set	Testes de Hardware e familiarização inicial com Software
23 set	MARCO I - Hardware funcional: PC, Arduino e Fita de LED integrados
30 set	Familiarização com numpy e C like finalizada
14 out	Input e output de dados pelo Python efetivado
21 out	MARCO II - Softwares funcionais: Python finalizado / Arduino em Desenvolvimento
4 nov	Possibilidade do uso de módulos adicionais definido
11 nov	Algoritmo de Mapeamento de Cores finalizado
18 nov	MARCO III - Projeto finalizado: hardware e software completamente integrados e funcionais
25 nov	Testes finais de hardware e software
2 dez	Verificações finais, Documentação e Apresentação

Figure 3: Cronograma do Projeto dividido pelas semanas

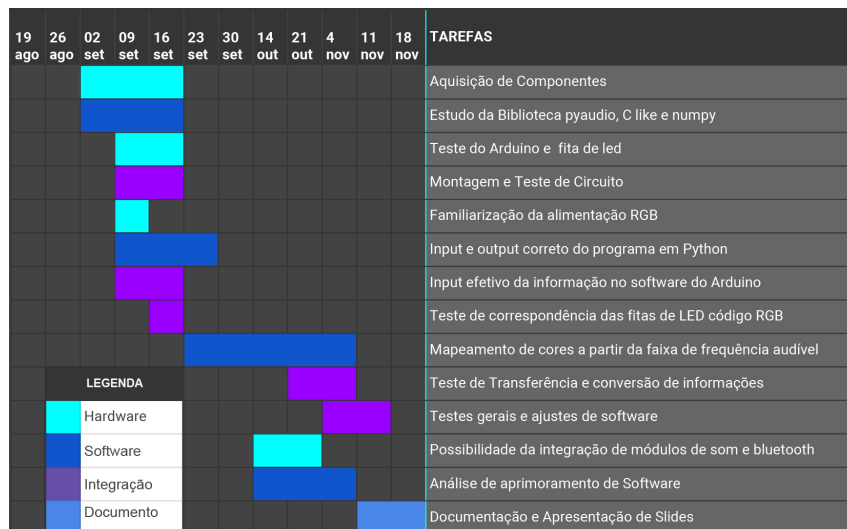


Figure 4: Cronograma do Projeto detalhado por tarefas

5 Componentes e ferramentas

Os componentes e ferramentas necessários para este projeto estão listados abaixo:

- Microcontrolador Arduino UNO
- Cabo USB/SERIAL
- Conectores
- Fita de LED endereçável RGB
- Software na IDE do Arduino
- Software em Python

Orçamento:

- Arduino UNO + Cabo USB : 25 reais
- Fita de LED endereçável (5 metros): 200 reais

Orçamento total : 225 reais

References

- [1] Mads Aasvik. What is fft and how can you implement it on an arduino?
- [2] William T Cochran, James W Cooley, David L Favin, Howard D Helms, Reginald A Kaenel, William W Lang, George C Maling, David E Nelson, Charles M Rader, and Peter D Welch. What is the fast fourier transform? *Proceedings of the IEEE*, 55(10):1664–1674, 1967.
- [3] Sho Nakagome. Fourier transform 101 — part 4: Discrete fourier transform.