



VI SEMANA DO CONHECIMENTO

**UNIVERSIDADE EM TRANSFORMAÇÃO:
INTEGRALIZANDO SABERES E EXPERIÊNCIAS**

2 A 6 DE SETEMBRO/2019



Marque a opção do tipo de trabalho que está inscrevendo:

(X) Resumo () Relato de Experiência () Relato de Caso

DISPOSITIVO ELETRÔNICO DE PERCUSSÃO BASEADO NO JOGO GENIUS PARA A APRENDIZAGEM DE PULSO E RITMO MUSICAL

AUTOR PRINCIPAL: Angelo Elias Dalzotto
CO-AUTOR: Sandro Cartier Larangeira
ORIENTADOR: Adriano Canabarro Teixeira
UNIVERSIDADE: Universidade de Passo Fundo

INTRODUÇÃO

Este relato de pesquisa refere-se ao primeiro protótipo desenvolvido para a tese de doutoramento de Sandro Cartier Larangeira.

Baseado no jogo Genius, o dispositivo (figura 1) pretende facilitar a aprendizagem de ritmo, pulso e a identificação de diferentes instrumentos musicais de percussão. A sequência de utilização se dá primeiramente pela audição de um ritmo proposto em que o usuário deve reconhecer os instrumentos, sua sequência e o pulso musical. Em seguida, o jogador deve repetir o que ouviu por meio dos botões referentes a cada instrumento, respeitando as condições do ritmo. Também é possível para o usuário gravar um ritmo no dispositivo, com a finalidade que ele mesmo ou um outro jogador seja desafiado pela nova sequência proposta.

DESENVOLVIMENTO:

O desenvolvimento do dispositivo foi feito utilizando o microcontrolador ATmega2560. Ele foi escolhido pela familiaridade com a arquitetura AVR, por conter o barramento *Inter-Integrated Circuit* (I2C) para o uso de um *display* e possuir o número de canais de *Pulse Width Modulation* (PWM) necessários para a síntese sonora, além da *Read-Only Memory* (ROM) de 256KiB, o suficiente para 2,9 segundos de *samples* de áudio em qualidade 16-bit 44.1KHz não comprimidos. A linguagem utilizada foi C++ com o compilador avr-gcc.



VI SEMANA DO CONHECIMENTO

**UNIVERSIDADE EM TRANSFORMAÇÃO:
INTEGRALIZANDO SABERES E EXPERIÊNCIAS**

2 A 6 DE SETEMBRO/2019



A interface de usuário foi feita em um display de caracteres 20x4 conectado ao I2C e quatro botões que mudam sua função em cada menu. A tela inicial tem as opções de gravar um ritmo ou ser desafiado por uma sequência já gravada. É possível tocar os instrumentos livremente no menu principal para conhecer os sons disponíveis. O menu de gravação permite que seja inserido um nome para o ritmo utilizando os botões cima e baixo para trocar o caractere, selecionando-os até finalizar ou apagando até voltar à tela inicial. Ao fim de tocar o ritmo para gravar, o jogador deve finalizar pela interface. O menu de desafio mostra os ritmos disponíveis em uma lista com rolagem para cima e para baixo. Ao selecionar a gravação, ela é reproduzida, podendo repetir a audição quantas vezes forem necessárias. Finalmente, o jogador tenta imitar o ritmo que ouviu. Caso acerte todos os instrumentos dentro da tolerância de tempo, uma mensagem de sucesso é mostrada. Caso erre um instrumento ou o pulso rítmico, o jogador tem a opção de tentar novamente. Também foram adicionados LEDs que acendem para cada instrumento para melhorar a identificação.

Os instrumentos são representados por quatro botões, cada um com um som de percussão: caixa; bumbo; prato; e *hi-hat*. O áudio é sintetizado a partir da soma de dois canais PWM de 8-bit, filtrados e condicionados por um amplificador operacional LMC6482 (DALZOTTO, 2017). Este foi escolhido por ser *rail-to-rail*, o que facilita embarcar com a alimentação por bateria. O dispositivo suporta quatro *samples* simultâneos a partir de um *mixer* implementado em *software*. A saída de áudio pode ocorrer por um falante interno ou fone de ouvidos, cada um com seu controle de volume individual.

Os ritmos são gravados em um cartão *Secure Digital* (SD) através do barramento *Serial Peripheral Interface* (SPI). Foi necessário implementar o sistema de arquivos *File Allocation Table* (FAT) porque as soluções existentes, como FatFs (CHAN, 2018), possui dificuldades para montar cartões SD *High Capacity* (SDHC), e a SdFat (GREIMAN, 2018) tem muitas dependências do Arduino, que torna o código grande e ineficiente para o objetivo do projeto.

Os resultados obtidos mostraram a necessidade de aumentar a duração dos *samples*, pois os áudios como o do prato devem soar por longo tempo e a diminuição da qualidade sonora não é aceitável. Também foi constatado que a interface deve ser refinada para melhorar a interação com jovens nascidos na era digital.

CONSIDERAÇÕES FINAIS:

O ATmega2560 foi suficiente para a finalidade escolhida, mas não permite mais nenhuma melhoria devido à suas limitações de memória e de performance. Uma nova versão pode ser implementada utilizando o



VI SEMANA DO CONHECIMENTO

**UNIVERSIDADE EM TRANSFORMAÇÃO:
INTEGRALIZANDO SABERES E EXPERIÊNCIAS**

2 A 6 DE SETEMBRO/2019



microcontrolador ESP32, *display* de maior resolução que permite desenhos mais complexos, e disposição dos botões similar à um controle de *videogame*.

REFERÊNCIAS

ELM by ChaN. FatFs - Generic FAT Filesystem Module R0.13c. 2018.
Disponível em: <<https://bit.ly/1vvpwQq>>. Acesso em 3 de maio de 2019.

GREIMAN, B. SdFat: Arduino FAT16/FAT32 Library 1.0.7. Disponível em:
<<https://bit.ly/2ZWmfew>>. Acesso em 3 de maio de 2019.

DALZOTTO, A. E. SINTETIZADOR DIGITAL DE ONDAS SIMPLES
UTILIZANDO UM CONVERSOR DIGITALANALÓGICO POR SOFTWARE EM
MICROCONTROLADORES AVR. Disponível em: <<https://bit.ly/2Y0OKpm>>.
Acesso em 3 de maio de 2019.

VI SEMANA DO CONHECIMENTO

**UNIVERSIDADE EM TRANSFORMAÇÃO:
INTEGRALIZANDO SABERES E EXPERIÊNCIAS**

2 A 6 DE SETEMBRO/2019



ANEXOS



Figura 1. O dispositivo finalizado ligado em seu menu principal.