

# Projeto de um controlador MIDI baseado em MSP430

Afonso Tresbach 12/0049031  
Universidade de Brasília - UnB - FGA  
Brasília, Brasil

**Resumo**—Este documento apresenta proposta de projeto de um dispositivo controlador MIDI com comunicação USB, baseado no microcontrolador MSP430, como trabalho final para a disciplina de Microprocessadores e Microcontroladores, da Faculdade do Gama - UnB.

## I. INTRODUÇÃO

### MIDI

O padrão *Musical Instrument Digital Interface* (MIDI) define protocolos de comunicação e interface com objetivo de conectar diferentes equipamentos como instrumentos musicais digitais e computadores, para composição e performance musical, ou artística em geral [1]. Aplicações menos conhecidas de MIDI são, por exemplo, no controle de iluminação e efeitos especiais em apresentações teatrais e parques de diversão.



Figura 1. Controlador MIDI Roland A-300PRO. Fonte: Roland

No contexto musical, a especificação MIDI não trata da representação de áudio digital em si, e sim de uma representação simbólica de notas e sons, através de características como velocidade, duração, altura [2]. Os chamados "comandos" ou "mensagens" MIDI podem ter tamanho de 1 a 3 bytes e são transmitidos de forma serial [3]. Um equipamento capaz de gerar áudio a partir destas informações no formato MIDI é um sintetizador, em contraposição ao controlador, que é o equipamento que gera o sinal MIDI. Ambos papéis podem ser cumpridos pelo mesmo aparelho, ou até mesmo puramente

através de *software* sequenciador. A Figura 2 apresenta a tela de um exemplo deste tipo de *software*.

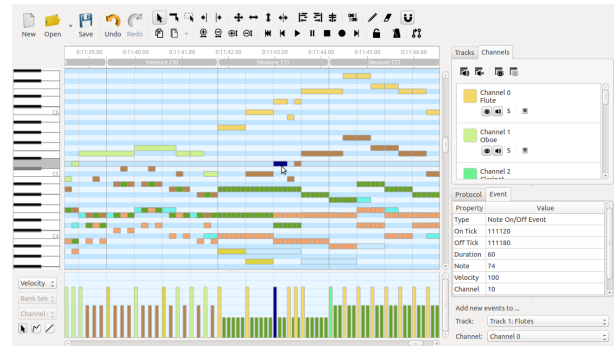


Figura 2. Software sequenciador MidiEditor. Fonte: Sourceforge

A versatilidade do MIDI permite que existam, em conjuntura, tanto equipamentos sofisticados fabricados por marcas consagradas no mercado, como ilustrado na Figura 1, quanto soluções de baixo custo, em geral utilizando microcontroladores, como o projeto ilustrado na Figura 3.



Figura 3. Controlador MIDI caseiro baseado em Arduino. Fonte: Instructables

### USB e USB-MIDI

O padrão Universal Serial Bus (USB) define conectores, cabos e protocolos de comunicação serial entre computadores

e periféricos [4]. Há especificações diversas para diferentes classes de dispositivos. Essas classes são identificadas por seus respectivos códigos e determinam o funcionamento específico dos dispositivos que as compõem, de acordo com seu contexto de aplicação. Controladores MIDI, por exemplo, integram a classe de código "01h", dedicada a equipamentos de áudio [5]. Como o padrão MIDI especifica não somente o protocolo, mas também o *hardware* subjacente para transferência dos dados, a comunicação de informação MIDI utilizando outro *hardware* não está em conformidade com a especificação. Esse tipo de comunicação é referido como USB-MIDI [6]. Esta configuração entre dispositivos em um sistema pode apresentar vantagens como expansão do limite de 16 canais MIDI, e desvantagens como a necessidade de conexão com um computador (ou um *host* USB), enquanto dispositivos MIDI podem ser conectados livremente sem impor uma arquitetura hierárquica.

## MSP430

A linha MSP430 da fabricante Texas Instruments é composta por microcontroladores para diversos fins que compõem, juntamente a periféricos, acessórios e ferramentas de *software*, uma plataforma completa de desenvolvimento. Esta plataforma única atende todas as diversas áreas de aplicação de microcontroladores e é largamente adotada pela indústria e pesquisa. Esta é uma forte razão para que seja adotado no ensino de arquitetura de microprocessadores e microcontroladores, apresentando futuros profissionais a uma possível ferramenta de trabalho. A existência de kits de desenvolvimento de grande qualidade e baixo custo especialmente destinados a fins educativos é outra forte razão.

O MSP430 suporta por padrão somente as classes USB HID, CDC, MSC e PHDC [7]. As variantes F5xx e F6xx deste microcontrolador vêm dotadas de periféricos específicos para o desenvolvimento de dispositivos USB. Estão disponíveis também, para esta plataforma, bibliotecas de código e ferramentas para auxiliar na criação de novos descritores, além de uma extensa documentação. A Texas Instruments ainda compartilha sob uma licença com seus clientes usuários de MSP430 seu *Vendor ID* (VID), necessário para criação de novos dispositivos USB [8], e que do contrário representaria um custo de milhares de dólares.

## II. OBJETIVOS

### Objetivos Gerais

Neste projeto, objetiva-se desenvolver um controlador MIDI baseado no microcontrolador MSP430, capaz de acionar efeitos de áudio através de botões e potenciômetros com funcionalidade programável. A programação dos controles deve ser possível via USB utilizando um computador com *software* sequenciador.

### Objetivos Específicos

Para possibilitar a codificação de comandos MIDI com praticidade e elegância, é desejável que se utilize uma *Application Programming Interface* (API). Para tanto, objetiva-se desenvolver uma biblioteca em C/Assembly para MSP430 com tal finalidade.

## III. REQUISITOS

O *software* desenvolvido, além de ser bem documentado, ter código organizado, e fornecer uma API consistente (seguindo boas práticas para nomeação de tipos, variáveis e funções; fornecendo códigos de retorno e saídas coerentes em casos de erro; etc), deve ser capaz de:

- Atuar instantaneamente em resposta aos controles de entrada, lidando adequadamente com ruído
- Interpretar e tratar adequadamente os níveis das entradas
- Gerar comandos MIDI de acordo com os controles acionados
- Transportar por USB os comandos MIDI de forma "*class-compliant*", ou seja, conforme especificações da classe USB respectiva

O *hardware* implementado deve permitir ao usuário:

- Controlar entradas digitais e analógicas por meio de 9 chaves e 2 potenciômetros
- Visualizar o estado atual de cada uma das entradas por meio de LEDs

A estrutura onde o circuito será montado deve:

- Garantir boa fixação de todas partes que serão manuseadas com frequência, como conectores; ou com vigorosidade, como os controles
- Possibilitar transporte e mobilidade
- Ser o mais leve e compacto possível, mas dispor controles com espaçamento suficiente

## REFERÊNCIAS

- [1] MIDI Association. Welcome to midi association. Acesso em 04-Abril-2018. [Online]. Available: <https://www.midi.org/>
- [2] Wikipedia. Midi. Acesso em 04-Abril-2018. [Online]. Available: <https://en.wikipedia.org/wiki/MIDI>
- [3] P. D. Lehrman. What is midi? Acesso em 04-Abril-2018. [Online]. Available: [https://www.midi.org/downloads?task=callelement&format=raw&item\\_id=150&element=f85c494b-2b32-4109-b8c1-083cca2b7db6&method=download](https://www.midi.org/downloads?task=callelement&format=raw&item_id=150&element=f85c494b-2b32-4109-b8c1-083cca2b7db6&method=download)
- [4] Wikipedia. Usb. Acesso em 04-Abril-2018. [Online]. Available: <https://en.wikipedia.org/wiki/USB>
- [5] U. I. Forum. Usb class codes. Acesso em 04-Abril-2018. [Online]. Available: [http://www.usb.org/developers/defined\\_class](http://www.usb.org/developers/defined_class)
- [6] U. I. Forum. Universal serial bus device class definition for midi devices. Acesso em 04-Abril-2018. [Online]. Available: [http://www.usb.org/developers/docs/devclass\\_docs/midi10.pdf](http://www.usb.org/developers/docs/devclass_docs/midi10.pdf)

- [7] Texas Instruments. Msp430 design workshop – usb devices. Acesso em 04-Abril-2018. [Online]. Available: [http://software-dl.ti.com/trainingTTO/trainingTTO\\_public\\_sw/MSP430\\_LaunchPad\\_Workshop/v4/Chapters/MSP430m10\\_USB.pdf](http://software-dl.ti.com/trainingTTO/trainingTTO_public_sw/MSP430_LaunchPad_Workshop/v4/Chapters/MSP430m10_USB.pdf)
- [8] —. Starting a usb design using msp43 mcus. Acesso em 04-Abril-2018. [Online]. Available: <http://www.ti.com/lit/an/slaa457a/slaa457a.pdf>