

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ENGENHARIA DA COMPUTAÇÃO

FELIPE LUIZ BILL
FELIPE MICHELS FONTOURA
LEANDRO PIEKARSKI DO NASCIMENTO
LUCIO EIJI ASSAOKA HOSSAKA

INTERFACE DE INSTRUMENTO MUSICAL ELETRÔNICO

TRABALHO ACADÊMICO

CURITIBA

2009

FELIPE LUIZ BILL
FELIPE MICHELS FONTOURA
LEANDRO PIEKARSKI DO NASCIMENTO
LUCIO EIJI ASSAOKA HOSSAKA

INTERFACE DE INSTRUMENTO MUSICAL ELETRÔNICO

Trabalho Acadêmico apresentado à Unidade Curricular de Oficina de Integração II do Curso de Engenharia da Computação da Universidade Tecnológica Federal do Paraná como requisito parcial para aprovação.

Orientador: Prof. Dr. Miguel Antonio Sovierzoski.

CURITIBA

2009

Se você tem uma maçã e eu tenho uma maçã, e nós trocamos as maçãs, nós, ambos, continuamos a ter apenas uma maçã. Contudo se você tem uma idéia e eu tenho uma idéia, e nós trocamos as idéias, cada um de nós agora tem duas idéias. (SHAW, George Bernard).

RESUMO

BILL, Felipe; Nascimento, Leandro do; FONTOURA, Felipe; HOSSAKA, Lucio. Interface de Instrumento Musical Eletrônico. 32 f. Trabalho Acadêmico – Curso de Engenharia da Computação, Universidade Tecnológica Federal do Paraná. Curitiba, 2009.

Este trabalho apresenta, com detalhamento prático e teórico, o projeto de uma interface humano-máquina que simula um instrumento musical de cordas. Faz-se uma breve explanação acerca do histórico dos instrumentos eletrônicos e do funcionamento dos instrumentos musicais de corda. Expõe-se em detalhes os aspectos técnicos do hardware utilizado, e em seguida lista os independentemente os componentes de hardware e de software envolvidos na execução da interface. Finaliza-se com uma avaliação da relevância do projeto através da discussão de suas aplicações. Este trabalho é complementado pela análise do projeto, do ponto de vista da metodologia da pesquisa, apresentando considerações sobre a arquitetura do sistema, implementação, testes, validação, gerenciamento de riscos, cronograma, orçamento e pessoal. Traz como resultado uma das muitas possibilidades inexploradas na perspectiva do uso da tecnologia no meio musical e artístico.

Palavras-chave: Instrumento Musical, Sintetizador

ABSTRACT

BILL, Felipe; Nascimento, Leandro do; FONTOURA, Felipe; HOSSAKA, Lucio. Electronic Musical Instrument Interface. 32 f. Trabalho Acadêmico – Curso de Engenharia da Computação, Universidade Tecnológica Federal do Paraná. Curitiba, 2009.

This paper presents, with theoretical and practical details, the design of a human-machine interface that simulates a string musical instrument.

It makes a brief explanation about the history of electronic instruments and the operation of string musical instruments. Exposes in detail the technical aspects of the hardware used, and then lists the independent components of hardware and software involved in the implementation of the interface.

It finishes with an assessment of the relevance of the project by discussing its applications. This work is complemented by the analysis of the project, in terms of research method, with respect to system architecture, implementation, testing, validation, risk management, schedule, budget and staff. As a result, it brings a new possibility when it comes to artistic and musical expression through technology.

Keywords: Musical Instrument, Synthesizer

LISTA DE FIGURAS

FIGURA 1	– GRÁFICO EM EXCEL DA FREQUÊNCIA VS. POSIÇÃO	10
FIGURA 2	– FOTO DO ARDUINO DIECIMILA	16
FIGURA 3	– POTENCIÔMETRO	17
FIGURA 4	– PONTE DE WHEASTONE	18
FIGURA 5	– AMPLIFICADOR OPERACIONAL	19
FIGURA 6	– ESQUEMA DO CIRCUITO DE LEITURA	20
FIGURA 7	– FOTO DO CIRCUITO DE LEITURA	21
FIGURA 8	– FOTO DA FONTE	23
FIGURA 9	– COMUNICAÇÃO DO SOFTWARE	25

LISTA DE TABELAS

TABELA 1	–	FREQUÊNCIAS DAS NOTAS MUSICAIS DA ESCALA DIATÔNICA	11
TABELA 2	–	ORÇAMENTO DO PROJETO	27

SUMÁRIO

1	INTRODUÇÃO	7
1.1	MOTIVAÇÃO	7
1.2	OBJETIVOS	8
1.2.1	Objetivos Gerais	8
1.2.2	Objetivos Específicos	8
1.3	PRINCÍPIO DE FUNCIONAMENTO DOS INSTRUMENTOS DE CORDA	8
2	PROPOSTA	12
3	HARDWARE	15
3.1	CIRCUITO DE LEITURA DA RESISTÊNCIA DAS CORDAS	15
3.1.1	Kit de Hardware Arduino	15
3.1.2	Fio de Níquel-Cromo	16
3.1.3	Potenciômetro	17
3.1.4	Ponte de Wheatstone	18
3.1.5	Amplificador Operacional	19
3.1.6	Diagrama e Componentes do Circuito de Leitura	20
3.2	CIRCUITO DE DETECÇÃO DE OBSTRUÇÃO (TOQUE DA CORDA DO INSTRUMENTO)	21
3.2.1	LED Infravermelho e Fotodiodo	22
3.3	FONTE DE ALIMENTAÇÃO	23
4	SOFTWARE	25
4.1	BIBLIOTECA DE FUNÇÕES	25
4.2	COMUNICAÇÃO COM O HARDWARE	26
5	CONSIDERAÇÕES FINAIS	27
5.1	ORÇAMENTO	27
5.2	COMENTÁRIO DO DESENVOLVIMENTO DO PROJETO	27
5.3	POTENCIALIDADES DO PROJETO	28
6	REFERÊNCIAS BIBLIOGRÁFICAS	29
	APÊNDICE A – DOCUMENTAÇÃO DA BIBLIOTECA DE SOFTWARE	30
	APÊNDICE B – MENSAGENS DO SISTEMA DE COMUNICAÇÃO	32

1 INTRODUÇÃO

Na década de 70, um grupo de fabricantes de instrumentos musicais desenvolveu um padrão de mensagens de controle entre instrumentos musicais, o padrão MIDI (Musical Instrument Digital Interface). Essas mensagens não contêm o áudio propriamente dito, mas instruções protocoladas, que definem notas, timbre, ritmo e efeitos usados por sintetizadores para produzir som.

Seguindo o caminho inverso, recentemente, surgiram interfaces eletrônicas que simulam instrumentos musicais. Popularizadas pela série de jogos “Guitar Hero”, guitarras, baterias e outros instrumentos são simulados. Essas interfaces recebem entradas de dados através de botões e as convertem em instruções que são interpretadas por um software. Por sua vez, o software gera o áudio, de acordo com as entradas.

Em um esforço semelhante, um instrumento musical eletrônico foi criado: o traútônio. Vale ressaltar que sua invenção foi realizada muito antes dos eventos acima descritos. Concebido por Friedrich Trautwein, em 1929, logo atraiu músicos como, mais notadamente, Oskar Sala, que construiu suas próprias versões do instrumento. Em suma, o aparelho é composto por um fio resistivo, que varia sua diferença de potencial (ddp) conforme é tocado. A ddp serve como entrada para o sintetizador, que a converte em emissão sonora.

O projeto desenvolvido consiste em uma interface eletrônica, cujo funcionamento é semelhante ao do traútônio. Todavia, o projeto é composto por mais de uma corda, e não apresenta sintetizador.

1.1 MOTIVAÇÃO

Com o desenvolvimento da tecnologia após a Terceira Revolução Industrial - nomenclatura considerada discutível por alguns historiadores, porém facilmente compreensível sobre a que se refere -, existe atualmente a forte tendência de se integrar computação, robótica e eletrônica em diversos setores da sociedade, além da habitual aplicação industrial dessas tecnologias.

Um dos setores em que o amparo tecnológico mais se destaca é o da música. Não somente no que se refere à gravação e armazenamento, mas, em especial, no que se refere à composição. Atualmente, existem ferramentas que permitem até mesmo confecção de partituras com sintetização sonora.

Esta proposta é um esforço no sentido de prover um instrumento musical eletrônico que permita também, entre outras funções, a representação musical através de recursos de vídeo. Ela foi motivada pelo interesse comum dos integrantes da equipe de desenvolvimento pela música e pelo design de interação.

1.2 OBJETIVOS

1.2.1 Objetivos Gerais

Criar um protótipo de um instrumento musical eletrônico de cordas e desenvolver um software que receba e interprete os sinais enviados por um circuito elétrico, sendo esse responsável pela interface de interação com o usuário.

1.2.2 Objetivos Específicos

Aprimorar conhecimentos em:

- Eletrônica analógica, eletrônica digital, conversão analógico-digital;
- Interação hardware-software, sensores;
- Trabalho entre equipes multidisciplinares;
- Noções de metodologia da pesquisa;
- Noções de gerenciamento de projetos, testes e noções de gerenciamento de prazos.

Aplicar conhecimentos e esforços condizentes para desenvolver as bases de um projeto posterior de instrumento musical.

1.3 PRINCÍPIO DE FUNCIONAMENTO DOS INSTRUMENTOS DE CORDA

Os instrumentos de cordas apresentam cordas tensionadas, sob ação de algum tipo de força, o que faz com que oscilem, produzindo sons. É comum classificar esses instrumentos de acordo

com a forma como a corda é percutida. Os instrumentos de cordas friccionadas são tocados por fricção sobre as cordas, normalmente usando um arco, como é o caso do violino ou do violoncelo. Os instrumentos de cordas beliscadas são tocados por beliscões ou empurrões tangentes às cordas, como é o caso do violão e do cravo. Os instrumentos de cordas percutidas são tocados por impacto sobre as cordas, como é o caso do piano.

Alguns instrumentos de cordas friccionadas ou beliscadas apresentam as cordas dispostas numa parte do instrumento denominada “braço”. Em tais instrumentos, as cordas são pressionadas pelo instrumentista em determinados pontos do braço, de forma que a corda só oscile a partir do ponto em que foi pressionada, o que define a altura (tonalidade) do som. A forma em que o impacto, o belisco ou a fricção são produzidos define a intensidade (volume) do som. São exemplos instrumentos como o violão, o violino e o violoncelo. Do ponto de vista técnico, pode-se dizer que tais instrumentos recebem dois “parâmetros” ou “sinais de entrada” para cada uma das cordas: o “toque”, ou seja, a oscilação ou não da corda, e a posição do dedo no braço. Em contrapartida, o instrumento envia uma “resposta” ou “sinal de saída”: o som na frequência, no timbre e na intensidade desejados.

medida em que a posição em que a corda pressionada pelo dedo do instrumentista for mais distante da extremidade mais externa do braço do instrumento, a parte oscilante da corda diminui, e a frequência sonora gerada aumenta. Essa variação não é linear, e tampouco o é a percepção dessa frequência pelo ouvido humano. Reduzindo-se o comprimento oscilante da corda pela metade, aumenta-se uma oitava, ou seja, toca-se a mesma nota mais aguda.

É possível estabelecer uma relação entre a posição em que o dedo é pressionado numa corda (a partir da extremidade mais externa do braço) e a frequência gerada ao percutir tal corda. Na fórmula apresentada pela equação 1, o tamanho do trecho tensionado das cordas é representado por L , e a frequência do harmônico fundamental da corda, ou seja, aquela que é produzida ao tocá-la sem pressionar (matematicamente equivalente a pressioná-lo na extremidade o ponto 0) é representada por f_0 .

$$f(x) = f_0 * L - x/L \quad (1)$$

Supondo um instrumento com uma única corda, com comprimento de um metro e afinada em dó (com frequência 523,25 Hz), tem-se o gráfico de frequência por posição exibido na figura 1.

Os pontos em destaque no gráfico apresentado na figura 1 são as notas da escala diatônica maior (a mesma escala representada pelas teclas brancas do piano). O gráfico apresenta ape-

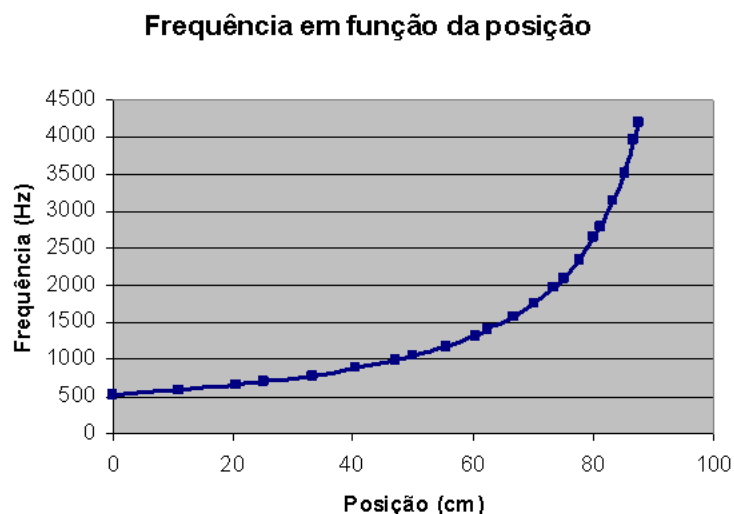


Figura 1: Gráfico de Frequência por Posição de um instrumento de corda fictício.

Fonte: Autoria Própria

nas três oitavas, ou seja, três trechos de dó a dó. Na tabela 1 estão apresentados os valores correspondentes de nota, posição e frequência:

O instrumento fictício apresentado acima tem uma função frequência/posição contínua. Em contrapartida a essa forma de resposta da frequência, alguns outros instrumentos possuem uma função frequência/posição discreta, comumente através do uso de trastes, pequenos pedaços de metal que demarcam a “zona” do braço em que se pressiona para produzir cada nota. Um exemplo de instrumento com função de frequência contínua é o violino, e um exemplo de instrumento com função de frequência discreta é o violão. O projeto descrito neste trabalho é um sistema de entrada de dados cuja forma de utilização, do ponto de vista do usuário, é semelhante à de um instrumento de frequência discreta, especialmente do violão e da guitarra elétrica.

Tabela 1: Frequências das Notas Musicais da Escala Diatônica

Nota	Posição (cm)	Frequência (Hz)
dó	0,00	523,25
ré	10,91	587,33
mi	20,63	659,26
fá	25,08	698,46
sol	33,26	784,00
lá	40,54	880,00
si	47,03	987,77
dó	50,00	1046,50
ré	55,46	1174,66
mi	60,31	1318,51
fá	62,54	1396,91
sol	66,63	1567,98
lá	70,27	1760,00
si	73,51	1975,53
dó	75,00	2093,01
ré	77,73	2349,32
mi	80,16	2637,02
fá	81,27	2793,83
sol	83,31	3135,96
lá	85,13	3520,00
si	86,76	3951,07
dó	87,50	4186,01

Fonte: Autoria própria.

2 PROPOSTA

A especificação de um produto a ser desenvolvido pode ser dividida em três partes: os requisitos funcionais, os requisitos não-funcionais, e os requisitos de domínio. Os requisitos funcionais são aqueles que definem o objetivo e a operação do artefato. Os requisitos não-funcionais são as propriedades emergentes do sistema, como: desempenho, confiabilidade, facilidade de manutenção ou usabilidade.

Para esta proposta os requisitos funcionais são:

- Uma interface (física) de um instrumento musical, composta por uma ou mais cordas. As cordas são fios resistivos, que variam a tensão do circuito elétrico conforme a posição em que são pressionadas. Essa tensão é passada a um interpretador/sintetizador de som;
- Um software de interpretação de sinais e conversão em dados que possam ser posteriormente processados.

Os requisitos não funcionais estão relacionados ao modo como o sistema é implementado. Embora especificá-los nesta etapa do projeto implique na limitação das possibilidades de solução, eles estão listados a seguir:

- Conversão da medida da tensão em um valor que varie de 0 a 1023. Isso permite que o artefato tenha uma extensão de notas significativa.
- Sistema intuitivo, de funcionamento semelhante ao de um violão ou guitarra. O sistema depende de um computador para funcionar.
- As cordas não devem dissipar uma potência significativa, a ponto de aquecer. Bem como, não devem conduzir uma corrente que represente perigo ao usuário.
- O sistema deve estar preparado para receber mais cordas.
- O software deve gerar saídas que possam ser facilmente convertidas em som, gráficos, etc., para ser utilizado no ensino de música, possivelmente.

O modelo de desenvolvimento adotado foi o modelo evolucionário. Os requisitos eram conhecidos, mas não estavam bem definidos no começo do projeto. Por essa razão, diversos protótipos de circuitos-elétricos e funções de software foram desenvolvidos, ensaiados e testados para que o sistema fosse melhor entendido, para que os requisitos se tornassem mais claros. Alguns protótipos foram descartados e outros foram incrementados para formar o sistema atual.

O projeto envolveu diversos riscos, entre os quais, destacam-se aqueles relacionados à resposta do fio. Isso ocorre porque a corrente circulando no fio resistivo deve ser pequena de modo que a potência dissipada seja insignificante para não provocar o seu aquecimento, bem como para não causar riscos ao usuário, uma vez que o contato com o fio será direto. Se a corrente for muito baixa, o Arduino (plataforma de interfaceamento hardware-computador, explicada mais adiante) pode deixar de funcionar, ou funcionar inadequadamente.

O cronograma final não se apresentou de maneira muito diferente das expectativas de prazos. Isso aconteceu porque havia poucas dependências entre as tarefas, uma vez que quase todo o software pode ser implementado sem a necessidade de conhecer o circuito elétrico. O circuito elétrico, por sua vez, também só se comunicava com a interface do software e pôde ser desenvolvido sem depender da finalização do programa. Dos requisitos de hardware, a equipe não dispunha apenas do fio resistivo de nicromo (liga de níquel e cromo), que foi adquirido e entregue dentro do prazo estipulado.

Os testes de componentes revelaram pequenos defeitos na implementação do circuito elétrico e do software. Os testes revelaram a inconsistência das mensagens do protocolo desenvolvido para a comunicação com o circuito elétrico, falhas do mecanismo de funcionamento da comunicação serial e má estruturação da biblioteca de software de comunicação. Quanto ao circuito elétrico, os testes de componente apresentaram saídas ineficientes e imprecisas. A ponte de Wheatstone (explicada mais adiante) não fornecia valores confiáveis. O circuito foi remodelado, de forma a medir o ganho de tensão no próprio amplificador operacional.

Os testes de integração apontaram problemas na aquisição de dados. O Arduino envia dados continuamente, o que dificultava a diferenciação da natureza dos dados recebidos. O sistema foi alinhado ao protocolo de mensagens criado, de modo que o software somente recebe entradas quando as solicita. Os valores recebidos pelo computador passaram a ser mais confiáveis, contudo a leitura tornou-se mais demorada.

A qualidade do produto e dos processos não foi afetada pelo cronograma, pois, como citado, os marcos do processo foram entregues dentro do prazo estimado. A leitura do sistema tornou-se mais demorada, mas isso não significou o atraso de leitura ou “leituras sujas” (erradas). O intervalo de entradas foi limitado, como previsto, pela interface do instrumento. Nas extremi-

dades, os valores medidos não chegam a exatamente 0 ou 1023, pois os contatos são imprecisos. Isso, contudo, é irrelevante, pois a extensão de notas não foi alterada de maneira significativa.

3 HARDWARE

3.1 CIRCUITO DE LEITURA DA RESISTÊNCIA DAS CORDAS

3.1.1 Kit de Hardware Arduino

O Arduino é um kit de hardware que possui um microcontrolador com portas digitais e analógicas para entrada e saída de informações. É disponibilizado numa grande variedade de versões, de portes variados, existindo, por exemplo, versões Mini e Mega, esta última sendo a maior delas; cada uma tem sua melhor utilidade em determinadas situações.

A principal vantagem do Arduino é que ele pode ser considerado um hardware de alto-nível, traçando-se um paralelo com a categorização típica das linguagens de programação. Do ponto de vista físico, ele tem as portas de interfaceamento bem definidas para alimentação e comunicação digital ou analógica. Já no software, sua linguagem de programação é baseada em Wiring, que por sua vez tem a interface baseada na plataforma de desenvolvimento de programas em Processing. Na prática o que se tem é uma linguagem muito similar ao ANSI C, por seguir lógica estruturada e ter palavras reservadas semelhantes, além de fornecer todas as funções necessárias para entrada e saída de informações, como leitura analógica/digital e escrita digital/analógica (essa ltima na forma de PWM).

A figura 2 apresenta a foto da versão Arduino Diecimila, que foi utilizada no projeto, adquirido em 2008 na versão Freeduino BR, produzida e comercializada em Curitiba. Essa versão utiliza um microcontrolador Atmel ATmega168, possui 16 KB de memória flash e uma EEPROM de 512 bytes. A corrente máxima fornecida pelas portas digitais é de 40 mA.

Os principais aspectos do Arduino que serão utilizados são a entrada/saída de dados e seu conversor D/A (digital para analógico) de 10 bits (01023), com a entrada de sinal analógico variando de um valor próximo de 0 V a um valor próximo de 5 V.

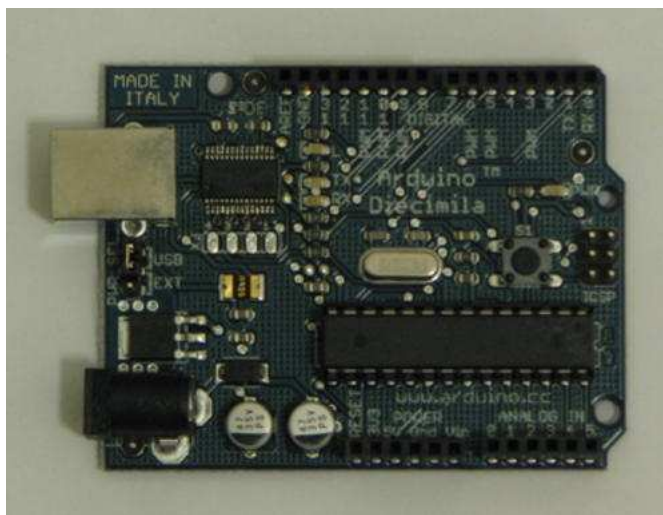


Figura 2: Arduino Diecimila.

Fonte: http://blog.makezine.com/archive/2007/10/arduino_-_diecimila_-_and_-_bt.html

3.1.2 Fio de Níquel-Cromo

O fio de níquel-cromo foi o tipo de fio utilizado como corda no projeto do instrumento. As características que o material apresenta justificam essa escolha, baseando-se na sua utilização como resistência elétrica. Optou-se pelo fio de espessura padrão AWG 32, isto é, diâmetro de aproximadamente 0,202 mm.

O níquel-cromo ou simplesmente nicromo, é uma liga metálica composta por níquel majoritariamente e cromo, e eventualmente ferro. Sua relativa alta resistividade e resistência à oxidação em altas temperaturas fazem dessa liga uma opção muito utilizada na produção de resistências elétricas que tenham por fim produzir calor por efeito Joule, destinadas à utilização em aquecedores, secadores de cabelo, chocadeiras, seladoras de termoplástico, certos tipos de vela de ignição, entre outros. As propriedades esperadas do fio AWG 32 de nicromo seguem abaixo:

- Diâmetro: 0,202mm;
- Seção transversal: 0,0314mm²;
- Resistência por metro: 34,6Ω/m;
- Densidade linear: 0,2625g/m;

Pretende-se utilizar cordas com comprimento aproximado de um metro, de forma que a resistência máxima a ser aferida pelo sistema seja da ordem de algumas dezenas de ohms. O

circuito de aferição da resistência foi projetado de forma que a tensão aplicada sobre o fio seja da ordem de décimos de volt, garantindo que a intensidade da corrente elétrica no fio de valor adequado ao uso no projeto, ou seja, suficientemente pequeno para que a temperatura do fio seja segura para o usuário.

3.1.3 Potenciômetro

O potenciômetro é um dispositivo eletrônico que funciona como um divisor de tensão variável. Possui três terminais, sendo dois deles fixos e um móvel. Utilizando o terminal central com uma das extremidades é possível utilizar esse dispositivo como um resistor variável. Nos modelos comuns, como apresentado na figura 3, o ajuste é feito pelo movimento do eixo móvel exposto, cuja variação na maioria das vezes não ultrapassa 270°. Já nos trimpots o ajuste é feito através do movimento de um parafuso, somente acessível com o uso de uma chave de fenda.



Figura 3: Potenciômetro de 3/4 de volta.

Fonte: <http://greduino.co.cc/wp-content/uploads/2009/09/Potentiometer.jpg>

No projeto fez-se uso de trimpots multivoltas, cujo eixo realiza algumas voltas completas entre a posição de resistência mínima e a de máxima, devido à precisão e confiabilidade requeridas. Eles são utilizados como os resistores de valores conhecidos na ponte de Wheatstone, ajustados para exatamente o valor máximo de resistência do fio de níquel-cromo utilizado. Além disso, são utilizados para ajustar com precisão o ganho dos amplificadores operacionais, para que nem subutilizem e nem ultrapassem a tensão máxima de entrada do Arduino, que é de 5V.

3.1.4 Ponte de Wheatstone

De início, para a construção do instrumento, decidiu-se utilizar uma ponte de Wheatstone, conforme mostrada na figura 4.

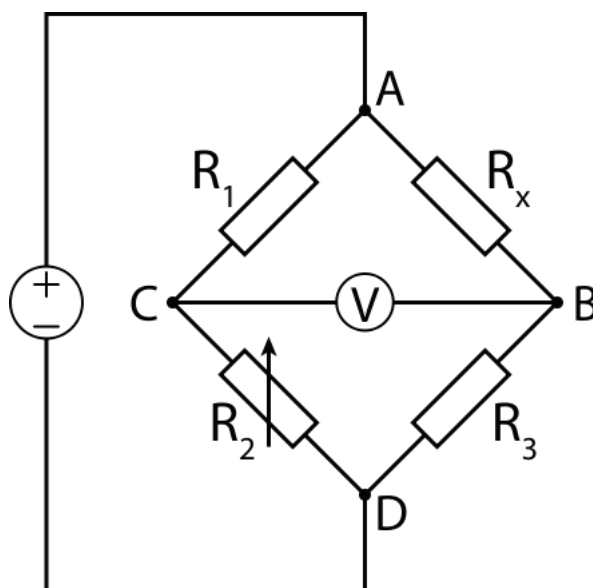


Figura 4: Ponte de Wheatstone.

Fonte: http://pt.wikipedia.org/wiki/Ponte_de_Wheatstone

Esse circuito permite determinar o valor de uma resistência desconhecida a partir de um arranjo com outras três resistências de valor conhecido. A relação entre as resistências que caracteriza essa ponte é dada pela equação 2.

$$R_1 R_3 = R_x R_2 \quad (2)$$

Contudo, essa relação só vale quando V_{cb} é igual a 0, o que não ocorre na maior parte do tempo, já que a resistência R_x é variável. Logo, o que se mede com o circuito é justamente V_{cb} , visto que é difícil projetar um resistor variável no lugar de R_2 que se ajustasse a todo momento para fazer valer a relação matemática da ponte. Com três resistores com valores iguais em R_1 , R_2 e R_3 com valor semelhante ao máximo do resistor desconhecido, V_{cb} só variará entre 0 e V_{ad} . Contudo, variava de forma não-linear, e o cálculo de R_x então tornava-se impreciso, principalmente devido à instabilidade da ponte, que se mostrou inconveniente na hora de se confeccionar e manipular o circuito.

Por esses motivos, descartou-se esse método de medição da resistência do fio. Em lugar da ponte de Wheatstone, utilizou-se uma solução mais simples e elegante, que é o de calcular a

resistência utilizando-se as propriedades do ganho de um amplificador no circuito.

3.1.5 Amplificador Operacional

O amplificador operacional (amp-op) é um circuito integrado de complexidade relativamente alta que possui principalmente uma elevada quantidade de transistores interligados, o que por si só já significa uma dificuldade para compreender plenamente seu funcionamento atualmente e por isso a preocupação nesse momento é utilizá-lo corretamente de acordo com a necessidade. De acordo com Millman (1981), o amplificador operacional tem aplicações interessantes como amplificador de tensão, amplificador de corrente, diferenciador, integrador, entre outras. O interesse é utilizá-lo principalmente como amplificador de tensão, na configuração amplificador não-inversor, apresentado pela figura 5.

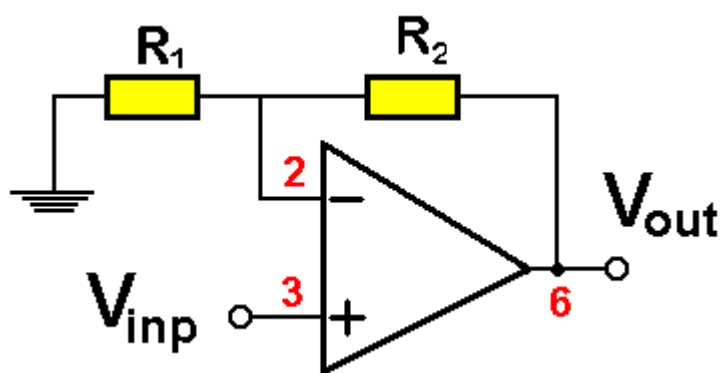


Figura 5: Amplificador Operacional, na configuração amplificador não-inversor.

Fonte: [http://www.ifi.unicamp.br/kleinke/f540/e_amp1.htm#Nao % 20inversor](http://www.ifi.unicamp.br/kleinke/f540/e_amp1.htm#Nao%20inversor)

Nessa configuração, o ganho na saída é dado pela equação 3.

$$G = V_{out}/V_{inp} = 1 + R_2/R_1 \quad (3)$$

Como exemplo, para se ter um ganho de 15 vezes é necessário utilizar qualquer configuração de resistores em que R_2 tenha resistência 14 vezes maior do que R_1 .

Para que o amplificador funcione como desejado é necessário que a alimentação seja no mínimo duas vezes superior à tensão do sinal de saída. O amp-op utilizado no projeto é o LM324N, que tem quatro amplificadores operacionais no seu encapsulamento.

3.1.6 Diagrama e Componentes do Circuito de Leitura

O circuito apresentado na figura 6 é o utilizado para a leitura do valor de resistência dos fios de nicromo. Partindo-se de uma fonte de 5 V, faz-se uso de um diodo comum para que a tensão sobre o fio de nicromo seja constante e aproximadamente igual a 0,7 V, devido à queda de tensão na junção do diodo quando polarizado diretamente. Isso deve ser feito porque a liga de nicromo tende a elevar a sua temperatura quando sujeita à passagem de altas correntes, situação em que dissipa elevada potência por efeito Joule. Além disso, não se deseja a passagem de uma corrente elevada em um circuito que será diretamente utilizado pelo usuário, pela sua própria segurança. Uma fotografia do circuito é exibida na figura 7.

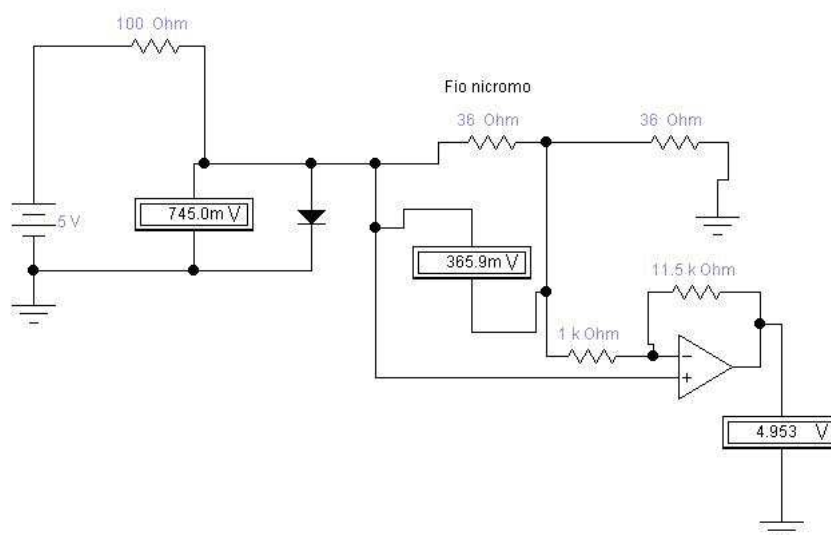


Figura 6: Diagrama do Circuito de leitura de resistência do fio de nicromo.

Fonte: Autoria Própria

A resistência de 100 ohms no terminal da fonte é utilizada para garantir a tensão mínima de condução no diodo, permitindo a passagem de um valor baixo de corrente, suficiente para a amplificação do amp-op ao mesmo tempo que não apresenta riscos de superaquecimento do fio. Já a resistência de 36 ohms é apenas um divisor de tensão de valor aproximadamente igual ao da resistência máxima do fio, assim escolhida para fixar um intervalo conveniente de valores da tensão sobre o fio de nicromo.

A tensão nos pontos em que está ligado o amp-op é a própria tensão sobre o fio de nicromo e varia, portanto, aproximadamente de 0 a 0,35 V. O amp-op é utilizado para que seja possível a obtenção de maior sensibilidade do circuito em relação à variação de resistência do fio, ao impor um ganho sobre a tensão de saída. Ao alargar a faixa de valores dessa tensão, é possível então

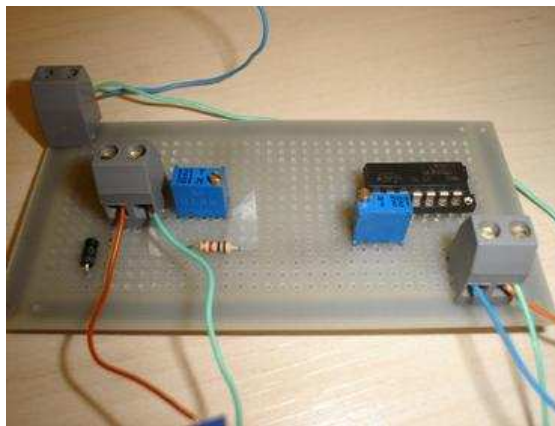


Figura 7: Foto do Circuito de leitura de resistência do fio de nicromo.

Fonte: Autoria Própria

se obter valores satisfatórios na conversão de analógica para digital. O circuito amplificador utilizado tem um ganho aproximado entre 13 a 14 vezes. Alimentando esse circuito com as saídas da fonte de 5 V e o amp-op com as de 12 V, os valores nominais aferidos pelo kit com microcontrolador pouco ultrapassam os 4,25 V. Um ganho maior não é desejável; por exemplo, utilizando-se um ganho de 15 vezes, a tensão de pico chega a aproximados 4,85 V, o que ultrapassa o limite superior de funcionamento do conversor A/D, de tal forma que a medição de um certo intervalo de valores da resistência acabam sendo cortada.

3.2 CIRCUITO DE DETECÇÃO DE OBSTRUÇÃO (TOQUE DA CORDA DO INSTRUMENTO)

Em adição à aferição da tonalidade em que se supõe que a corda do instrumento vibrasse, é necessário que o sistema possa reconhecer o instante em que o usuário toca uma nota, “beliscando” determinada corda. Para isso, adotou-se um sensor de presença como representando cada corda, de tal forma que, quando o usuário deseja tocar uma nota, ele deve simplesmente obstruir o respectivo sensor com os dedos.

Idealizou-se esse detector de obstrução com o uso de LEDs infravermelhos e fotodiodos, numa abordagem baseada no projeto de Oficinas de Integração 1, “Sistema de aquisição e tratamento de dados para experiências de cinemática”. A partir do momento em que haja uma obstrução da recepção de luz infravermelha do fotodiodo, deixará de haver condução e consequentemente é possível ler um nível lógico baixo. Caso contrário, ou seja, na situação normal em que há recepção e emissão de luz infravermelha, a leitura é de um sinal com nível lógico alto. Essas respostas às duas situações, explicadas na seção 3.2.1, fazem o sensor ser capaz de

detectar a presença de um objeto qualquer posicionado entre o LED infravermelho e o fotodiodo.

3.2.1 LED Infravermelho e Fotodiodo

Os diodos são compostos por uma junção PN, isto é, uma composição de dois tipos de semicondutores, P e N, que são classificados de acordo com a natureza da dopagem existente em sua composição. A parte de tipo N é feita pela dopagem de elementos químicos pentavalentes, como o antimônio e o fósforo, cujos elétrons livres em excesso conferem a eles uma grande tendência de doar elétrons. A de tipo P por sua vez, é feita pela dopagem de impurezas trivalentes, como o índio e o alumínio, e apresentam grande tendência a aceitar elétrons, pelo predomínio de lacunas. Os portadores de carga majoritários no semicondutor tipo P e no tipo N são, respectivamente, elétrons e lacunas.

A junção desses dois tipos de semicondutor gera a chamada zona de depleção, que surge pela tendência dos portadores de carga de cada um dos semicondutores de migrar para o outro, recombinando-se, um elétron com uma lacuna. Com isso, uma concentração de íons ocorre na região próxima à junção, pois cada semicondutor passa a ter excesso de cargas após a sua migração.

É devido a essa zona de depleção que o diodo apresenta seu comportamento típico. Se ligado na polaridade direta, e a tensão for suficiente para vencer o potencial da zona de depleção, desalojando os íons e criando portadores de carga livres novamente, ele conduz corrente. Se ligado na polaridade reversa, não passa corrente pelo diodo, pois a polarização do campo elétrico aumenta ainda mais a largura da zona de depleção, ao promover a migração de mais portadores de carga entre os dois tipos de semicondutor.

Quando um elétron retorna da banda de condução para a banda de valência, ele eventualmente encontra uma lacuna, e elimina-se a existência de um par lacuna-elétron, fenômeno chamado de recombinação. Na recombinação em semicondutores, a energia liberada é transferida para a estrutura cristalina na forma de calor. Entretanto, alguns semicondutores, como o feito a partir arseneto de gálio, que apresentam altos índices de recombinação, a liberação de energia ocorre na forma de radiação, parcialmente luz visível e na maior parte luz infravermelha. A maior liberação de luz acontece próxima à junção PN, que é onde ocorre a maior parte das recombinações. O LED infravermelho emite radiação eletromagnética no espectro infravermelho, que não é visível ao olho humano.

O fotodiodo, por sua vez, é basicamente um diodo com invólucro transparente sobre a

junção PN. Esse diodo age como fotodetector, por responder a uma entrada em forma de luz. Pode ser utilizado em duas configurações, na polarização direta ou inversa. Na primeira, a incidência de radiação em certas faixas de frequência sobre o fotodiodo, específicas para cada material do qual ele é feito, faz com que se produza sobre ele uma corrente e uma tensão específicas, dependendo do modo de operação. Já na polarização inversa, o fotodiodo apresenta elevada resistência à passagem de corrente, que diminui com a incidência de luz. Monitorando-se essa mudança, o fotodiodo em polarização reversa pode ser usado como detector da presença de luz.

3.3 FONTE DE ALIMENTAÇÃO

Circuitos que realizam amplificação de sinal, como circuitos que contém amplificadores operacionais, são bastante sensíveis a mudanças na alimentação. Uma bateria comum de 9 V, por exemplo, tornaria o circuito impreciso, pois ao perder carga apresenta flutuações na tensão fornecida. No caso do circuito de leitura analógica, ocorreria uma alteração dos valores extremos de leitura máxima, causando efeitos indesejados na leitura, como a recepção de uma faixa de dados cada vez menor.

Como o circuito de leitura da resistência do fio de nicromo requer precisão e não admite perda de sensibilidade ao longo do tempo, optou-se por construir uma fonte de alimentação de corrente contínua, a ser utilizada sempre que o circuito venha a necessitar de alimentação, incluindo-se nessa situação o kit de microcontrolador Arduino. A fonte é apresentada na figura 8.

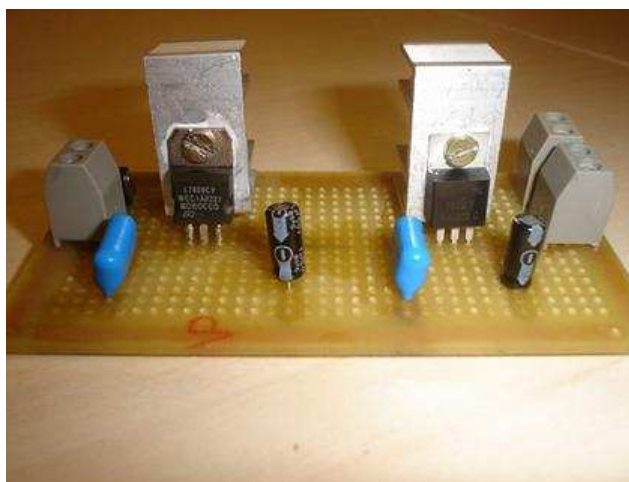


Figura 8: Fonte de alimentação.

Fonte: Autoria Própria

Na prática, a fonte não chega a ser um conversor AC/DC completo, mas sim um adaptador DC/DC que, a partir de uma fonte de 12 V /1 A, gera tensões de saída de 12 V, 9 V e 5 V, para alimentar, respectivamente, o amplificador operacional, o Arduino e o circuito de leitura respectivamente. Os principais componentes do circuito do circuito são os reguladores de tensão LM7809 e LM7805, além de capacitores de $0,33\mu F$ e $1\mu F$.

4 SOFTWARE

4.1 BIBLIOTECA DE FUNÇÕES

O hardware, através da plataforma Arduino, comunica-se com o computador usando a porta serial. A porta serial utilizada é uma porta USB (normalmente com conector USB-A). O driver USB utilizado, FTDI, simula o funcionamento de uma porta COM, criando uma porta COM virtual; ou seja, apesar de usar-se uma porta USB para a comunicação serial, vê-se, do ponto de vista do software, uma porta COM (RS 232).

Para essa comunicação, foi desenvolvida uma biblioteca de funções em linguagem C++, com o objetivo de permitir, de forma encapsulada, o envio de informações para o hardware e o recebimento de informações dele. A biblioteca funciona como uma interface entre a aplicação prática (alto nível) com a comunicação serial e o protocolo de envio (baixo nível), conforme esquematizado na figura 9. A documentação da biblioteca está apresentada no apêndice A. A comunicação é baseada no recebimento de pedidos e envio de respostas, ou seja, o hardware só envia informações para o computador quando, de alguma forma, requisita-se que ele as envie. Tal mecanismo apresenta a vantagem de o computador não ter de ficar constantemente recebendo e processando dados da porta serial, e também permite uma maior facilidade de distinção da natureza das informações recebidas.

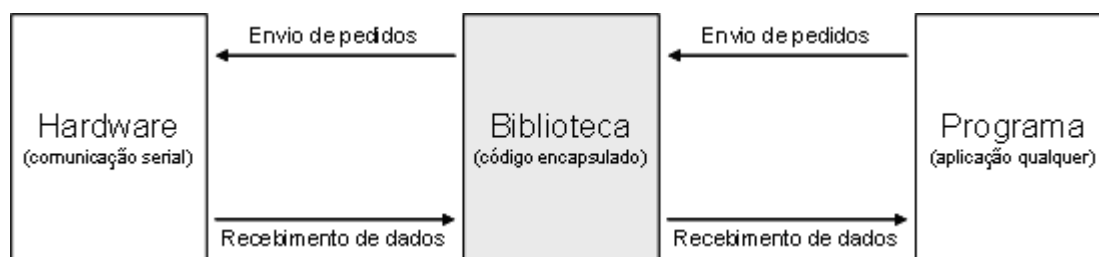


Figura 9: Diagrama ilustrativo do funcionamento da biblioteca.

Fonte: Autoria Própria

A maneira pela qual as informações são enviadas e recebidas através da porta serial foi elaborada de forma que fosse facilmente extensível, e que futuras extensões mantivessem a

compatibilidade reversa. As informações são sempre acompanhadas de algo que identifica sua natureza, o que, apesar de redundante na maioria dos casos, ajuda a evitar erros.

O código da biblioteca é exclusivo para sistema operacional Windows, visto que utiliza funções nativas desse sistema operacional para comunicar-se com a porta serial. Porém, pequenas adaptações no código permitiriam portar o código para sistemas baseados em UNIX.

4.2 COMUNICAÇÃO COM O HARDWARE

A comunicação serial através de uma porta COM pode ser entendida como um fluxo de bits (do inglês, *bitstream*). Tanto a biblioteca quanto o hardware entendem os bytes como caracteres de acordo com a codificação ASCII. A menor unidade de informação que a biblioteca troca com o hardware é chamada de “mensagem”. As mensagens são sequências de caracteres invariavelmente terminadas pelos caracteres de *carriage return* e *line feed*, representados pelos códigos 13 e 10.

Uma mensagem é formada por uma instrução e, se necessário, um ou mais parâmetros. O que divide uma instrução de seus parâmetros, e os parâmetros entre si, são caracteres de espaço, representados pelo código 32. No conteúdo dos parâmetros também podem existir caracteres de espaço.

Cada parâmetro tem um nome e um valor, separados entre si por um sinal de igual, representado pelo código 61. Entre o nome de um parâmetro, o sinal de igual e o seu valor podem existir um ou mais espaços. O valor dos parâmetros é representado por uma cadeia de caracteres circundados por aspas duplas. Para inserir as aspas duplas no valor de algum parâmetro, usa-se o caractere barra invertida (“\”) seguida das aspas duplas. De forma similar, para indicar a barra invertida, usa-se um par de caracteres de barra invertida.

Por conveniência, convencionou-se que as instruções e parâmetros são identificados por palavras ou mnemônicos em língua inglesa, visto que a tabela ASCII padrão não apresenta caracteres específicos do alfabeto da língua portuguesa, como os acentos, a cedilha ou o til. Uma tabela com as instruções trocadas pela biblioteca e pelo hardware, acompanhados de seus respectivos parâmetros, é apresentada no apêndice A.

5 CONSIDERAÇÕES FINAIS

5.1 ORÇAMENTO

Tabela 2: Tabela de custos dos componentes utilizados no projeto

Componente/produto	Valor unitário	Unidades	Valor
Folha de foam não revestido(A0)	R\$12,90	1	R\$12,90
Fio resistivo de nicromo (com frete)	R\$11,00 (m) + R\$8,00	5	R\$63,00
Arduino diecemila	R\$72,00	1	R\$72,00
Placa de circuito impresso universal	R\$3,00	3	R\$9,00
Resistores 1k	R\$0,30	1	R\$0,30
Resistores 100 ohms	R\$ 0,40	1	R\$0,40
Capacitores 330k	R\$0,70	2	R\$1,40
Capacitores 1F	R\$ 0,60	2	R\$ 1,20
CI LM324N	R\$ 0,80	1	R\$ 0,80
Conectores	R\$0,40	8	R\$3,20
Trimpot de multivoltas (100k)	R\$ 2,00	2	R\$4,00
Reguladores	R\$ 1,80	2	R\$3,60
Outros (solda, pasta, cola de isopor...)*			R\$3,00
Outros(socket, plug, diodos...)*			R\$3,00
Total			R\$177,80

Fonte: Autoria própria.

5.2 COMENTÁRIO DO DESENVOLVIMENTO DO PROJETO

O instrumento como um coletor de dados está quase totalmente implementado. O circuito de leitura do fio está finalizado, assim como o software de comunicação com o computador, que recebe os dados lidos através do Arduino. O circuito detector da obstrução, responsável por sinalizar a vibração das cordas, ainda não está concluído.

Além disso, embora não fosse objetivo primário do projeto, um programa que interpretasse as informações recebidas pelo instrumento e a traduzisse em sons não foi desenvolvido. Contudo, devido à flexibilidade do protocolo utilizado no software de recebimento dos dados, isso seria facilmente implementado, não fossem as dificuldades inerentes ao aplicativo que se

tentaria desenvolver.

5.3 POTENCIALIDADES DO PROJETO

Enfatiza-se aqui o fato de que o sistema, como um todo, foi projetado de modo a permitir mudanças substanciais nas características do instrumento físico e da interface de interpretação dos dados emitidos. Ou seja, as propriedades e também a finalidade do sistema são flexíveis.

O projeto foi executado de forma a ser facilmente adaptável posteriormente. Será possível utilizá-lo não apenas para a geração de sons; jogos ou aplicativos gráficos, instrumento de gravação de arquivos de áudio ou ferramenta de aprendizado são algumas das finalidades nas quais o sistema poderá ser empregado.

6 REFERÊNCIAS BIBLIOGRÁFICAS

KLEINKE, Mauricio Urban. "Visão Introdução a eletrônica e dispositivos semicondutores. Campinas, São Paulo. 2002. Disponível em: <http://www.ifi.unicamp.br/kleinke/f540/f540.htm>; Acesso em 14.10.2009.

MILLMAN, Jacob; HALKIAS, Christos C. Eletrônica: Dispositivos e Circuitos. Vol. 2. McGraw Hill. 1981.

MEDIDAS de resistência com a ponte de Wheatstone. Disponível em: <http://www.fisica.ufsc.br/lab2/pdf>; Acesso em 14.10.2009.

MACHADO, Bruno R. P.; SIMAS, Joo H. C.; GIOPOPO, Lucas L. Sistema de Aquisição de Dados para Experiências de Cinemática, Trabalho de Oficinas I, Engenharia de Computação, UTFPR. Curitiba, 2008.

SOMMERVILLE, Ian. Engenharia de Software, 8 edição; tradução: Selma Shin Shimizu Melnikoff, Reginaldo Arakkaki, Edilson de Andrade Barbosa. São Paulo, Pearson Addison Wesley, 2007.

APÊNDICE A – DOCUMENTAÇÃO DA BIBLIOTECA DE SOFTWARE.

A biblioteca de comunicação com o instrumento através da porta serial é composta primariamente da classe *ConexaoInstrumento*, que define métodos de alto nível diversos para obtenção de dados a partir do instrumento. Também compõe a biblioteca a estrutura corda, capaz de armazenar o estado de uma corda do instrumento (tem dois atributos: *posicao*, que indica a posição em que está o dedo, e *soando*, que indica se o sensor da corda está ou não obstruído). Num aplicativo com programação paralela (*multithreading*), não é seguro que o objeto seja acessado simultaneamente por mais de uma linha de execução (*thread*).

Classe <u>ConexaoInstrumento</u>	
Sumário de métodos públicos	
	<p><u>ConexaoInstrumento</u> (char* porta, unsigned int esperaMaxima_lchar, unsigned int esperaMaxima_total)</p> <p>Cria uma nova instância da classe.</p> <p>Parâmetros:</p> <p><u>porta</u>: nome da porta serial a usar.</p> <p><u>esperaMaxima_lchar</u>: espera máxima pelo primeiro caractere ao receber uma mensagem pela porta serial.</p> <p><u>esperaMaxima_total</u>: espera máxima pelo último caractere ao receber uma mensagem pela porta serial.</p>
void	<p><u>definirEsperas</u> (unsigned int esperaMaxima_lchar, unsigned int esperaMaxima_total)</p> <p>Define os tempos de espera dessa conexão com o instrumento.</p> <p>Parâmetros:</p> <p><u>esperaMaxima_lchar</u>: espera máxima pelo primeiro caractere ao receber uma mensagem pela porta serial.</p> <p><u>esperaMaxima_total</u>: espera máxima pelo último caractere ao receber uma mensagem pela porta serial.</p>
void	<p><u>descartarDados</u> ()</p> <p>Descarta dados em espera por leitura na porta serial.</p>
bool	<p><u>lerCorda</u> (int numero, corda* destino)</p> <p>Lê o estado de uma corda.</p> <p>Parâmetros:</p> <p><u>numero</u>: número da corda a consultar.</p> <p><u>destino</u>: ponteiro para estrutura de corda onde os valores com os dados sobre a corda consultada devem ser salvos.</p> <p>Retorno: se a operação foi bem sucedida.</p>
bool	<p><u>lerTotalCordas</u> (unsigned int* destino)</p> <p>Lê o total de cordas do instrumento.</p> <p>Parâmetros:</p> <p><u>destino</u>: ponteiro para variável onde a contagem de cordas deve ser salva.</p> <p>Retorno: se a operação foi bem sucedida.</p>
bool	<p><u>lerPrecisaoCordas</u> (unsigned int* destino)</p> <p>Lê o número máximo de trastes que o instrumento suporta.</p> <p>Parâmetros:</p> <p><u>destino</u>: ponteiro para variável onde a contagem de trastes deve ser salva.</p> <p>Retorno: se a operação foi bem sucedida.</p>

APÊNDICE B – MENSAGENS DO SISTEMA DE COMUNICAÇÃO.

Mensagens de pedido		
Instrução	Parâmetros	Exemplos
<code>get.information</code> Obtém uma informação.	<code>key</code> Identifica nome da informação procurada.	<code>get.information key="string count"</code> Obtém o total de cordas. <code>get.information key="position range"</code> Obtém quantas posições podem ser pressionadas.
<code>get.status</code> Obtém o estado de algum dispositivo.	<code>of</code> Nome do dispositivo. <code>index</code> Número do dispositivo, se houver mais de um.	<code>get.status of="string" index="1"</code> Obtém o estado da primeira corda.
Mensagens de resposta		
Instrução	Parâmetros	Exemplos
<code>set.information</code> Define uma informação	<code>key</code> Identifica o nome informação fornecida <code>value</code> Identifica o conteúdo informação fornecida	<code>set.information key="string count" value="2"</code> Informa que há duas cordas. <code>set.information key="position range" value="10"</code> Informa que cada corda tem precisão de 10 posições.
<code>set.status</code> Define o estado de algum dispositivo.	<code>of</code> Nome do dispositivo. <code>index</code> Número do dispositivo, se houver mais de um. <code>pressed</code> Se for uma corda: posição de pressionamento. <code>sounding</code> Se for uma corda, se está soando	<code>set.status of="string" index="1" pressed="5" sounding="F"</code> Informa que a primeira corda está soando e pressionada na 5ª casa. <code>set.status of="string" index="2" pressed="0" sounding="0"</code> Informa que a segunda corda está soando e não está pressionada.
Mensagens de erro		
Instrução	Parâmetros	Exemplos
<code>raise.error</code> Informa de um erro.	<code>kind</code> tipo do erro	<code>raise.error kind="wrong string index"</code> Informa que o número de corda pedido não é válido. <code>raise.error kind="wrong information name"</code> Informa que o nome de uma informação pedida não é válido. <code>raise.error kind="wrong entity name"</code> Informa que o dispositivo cujas informações foram pedidas não é válido.