

**MNPEF**  
Mestrado Nacional  
Profissional em  
Ensino de Física



## O USO DO *ARDUINO* E DO *PROCESSING* NO ENSINO DE FÍSICA

Luis Henrique Monteiro de Castro

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ensino de Física da UNIRIO no Curso de Mestrado Nacional Profissional de Ensino de Física (MNPEF), como parte dos requisitos necessários à obtenção do título de Mestre em Ensino de Física.

Orientador:  
Prof. Dr. Felipe Mondaini

Rio de Janeiro  
Setembro de 2016

O USO DO *ARDUINO* E DO *PROCESSING* NO ENSINO DE FÍSICA

Luis Henrique Monteiro de Castro

Orientador:

Prof. Dr. Felipe Mondaini

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Ensino de Física da UNIRIO no Curso de Mestrado Profissional de Ensino de Física (MNPEF), como parte dos requisitos necessários à obtenção do título de Mestre em Ensino de Física

Aprovada por:

---

Prof. Dr. Felipe Mondaini  
(Orientador)  
MNPEF UNIRIO / CETET Petrópolis

---

Prof<sup>a</sup>. Dr. Rosana Bulos Santiago  
(Membro Interno)  
MNPEF UNIRIO / UERJ

---

Prof. Dr. Antônio Carlos Fontes da Silva  
(Membro Externo)  
MNPEF UFRJ

---

Prof. Dr. André Tenório Leite  
(Membro Interno - Suplente)  
MNPEF UNIRIO / IFRJ

---

Prof. Dr. Vitorvani Soares  
(Membro Externo - Suplente)  
MNPEF UFRJ

Rio de Janeiro  
Setembro de 2016

## MODELO de FICHA CATALOGRÁFICA

CXXXc\* Castro, Luis Henrique Monteiro de  
O uso do *Arduino* e do *Processing* no Ensino de Física / Luis Henrique Monteiro de Castro - Rio de Janeiro: UNIRIO / IBIO, 2016.  
Número de seções, Número de páginas; dimensões da encadernação.  
Orientador: Felipe Mondaini  
Dissertação (mestrado) – UNIRIO / Instituto de Biociências / Programa de Pós-Graduação em Ensino de Física, 2016.  
Referências Bibliográficas: páginas inicial e final.  
1. Palavra-chave 1. 2. Palavra-chave 2. 3. Palavra-chave 3.  
I. Felipe Mondaini.  
II. Universidade Federal do Estado do Rio de Janeiro, Instituto de Biociências, Programa de Pós-Graduação em Ensino de Física.  
III. O uso do arduino no Ensino de Física.

\*Código da obra (fornecido pela biblioteca)

Consultar: <http://www.biblioteca.unirio.br/servicos-1/fichas-catalogograficas>

## **Agradecimentos**

Ao Professor Felipe Mondaini, meu grande mestre e querido amigo. Pela orientação firme e constante ao longo de tantos meses. Pelo incansável esforço que não me permitiu esmorecer, mesmo nos momentos mais difíceis.

À minha esposa Andreia Alves, meu verdadeiro e eterno amor. Por estar sempre ao meu lado e sempre solícita. Pelas noites mal dormidas e pelas que nem dormidas foram. Por sempre desejar e torcer pelo meu sucesso e vibrar com as minhas conquistas. Por acreditar em mim sempre.

À Janaina e ao Luis Henrique, meus amados filhos. Pela alegria, carinho e afeto que me ofertam continuamente. Ser Pai, sem dúvida, é uma das melhores experiências da minha vida.

A todos os professores do Programa que contribuíram para os meus estudos. Pelos ensinamentos. Pela disposição.

À toda a tripulação da Escola Técnica do Arsenal de Marinha professores e funcionários e, em especial, ao seu Coordenador, o Capitão de Mar e Guerra (RM1-T) Wilson Gonzaga Palmeira, pelo seu apoio e pela sua amizade, sem os quais não poderia haver este trabalho.

À equipe da Direção da Escola Técnica Estadual Ferreira Viana, em especial, à Diretora Geral Professora Andrea Reis Albino pela imprescindível cooperação e amizade, fundamentais para o crescimento profissional dos membros da comunidade escolar.

Em especial aos professores e amigos, Cesár Bastos, Lucia Helena e José Roberto que muito me ajudaram e me incentivaram nessa jornada.

Aos meus amigos professores e funcionários da Escola Técnica Estadual Ferreira Viana e da Escola Técnica do Arsenal de Marinha, pelo apoio e pelo incentivo.

A todos os meus amigos, colegas e parentes, sem os quais a vida não teria a mesma graça. Pelo incentivo. Pelas risadas. Por todos os bons momentos.

À CAPES pelo apoio financeiro por meio da bolsa concedida, o que em muito favoreceu o desenvolvimento das pesquisas, a construção dos experimentos e a conclusão deste trabalho.

## **RESUMO**

### **O USO DO ARDUINO E DO *PROCESSING* NO ENSINO DE FÍSICA**

Luis Henrique Monteiro de Castro

Orientador:  
Professor Dr. Felipe Mondaini

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Ensino de Física no Curso de Mestrado Nacional Profissional de Ensino de Física da UNIRIO (MNPEF), como parte dos requisitos necessários à obtenção do título de Mestre em Ensino de Física

Propomos nesta dissertação o emprego do Arduino e do Processing como ferramentas para o desenvolvimento de experimentos relevantes, de baixo custo e de qualidade adequada para o uso em substituição ou complementação aos laboratórios tradicionais de Física. Nesta dissertação, propusemos três objetos educacionais contendo as informações e o procedimento para sua construção, exemplo de dados obtidos e comprovação da qualidade do seu funcionamento, bem como roteiros de atividades para serem utilizados em sala de aula a fim de viabilizar, através de atividades práticas, o aprendizado dos conceitos de Física aos alunos do Ensino Médio.

Palavras-chave: Ensino de Física, Arduino, Processing e Laboratório didático.

Rio de Janeiro  
Setembro de 2016

## **ABSTRACT**

### **USE OF ARDUINO AND PROCESSING IN PHYSICS EDUCATION**

Luis Henrique Monteiro de Castro

Supervisor:  
Professor Dr. Felipe Mondaini

Abstract of master's thesis submitted to Programa de Pós-Graduação em Ensino de Física da UNIRIO no Curso de Mestrado Nacional Profissional de Ensino de Física (MNPEF), in partial fulfillment of the requirements for the degree Mestre em Ensino de Física.

We propose in this paper the use of Arduino and Processing as tools for the development of relevant experiments, low cost and suitable quality for use in replacement or complement to traditional laboratory physics. In this thesis, we proposed three educational objects containing the information and the procedure for its construction, sample data obtained and proof of the quality of its operation and activities of scripts for use in the classroom in order to facilitate, through practical activities, learning of physics concepts to high school students.

Keywords: Physics education, Arduino, Processing and Educational Laboratory

Rio de Janeiro  
Setembro de 2016

## Lista de Figuras

Figura 1 – A placa <i>Arduino UNO</i> .....	10
Figura 2 – Identificação dos pinos de conexão e dos principais componentes da placa do <i>Arduino UNO</i> .....	11
Figura 3 – Demonstração de alguns níveis de PWM .....	14
Figura 4 – Tela do software de programação <i>Arduino</i> .....	15
Figura 5 – Exemplo de conexão de <i>shields</i> em uma placa <i>Arduino UNO</i> .....	18
Figura 6 – Semelhança entre as IDEs do <i>Arduino</i> e do <i>Processing</i> .....	20
Figura 7 – Tela da IDE <i>Arduino</i> com o código .....	23
Figura 8 – Tela do <i>Serial Monitor</i> .....	23
Figura 9 – Tela da IDE <i>Processing</i> com o <i>Serial Monitor</i> .....	26
Figura 10 – Representação do movimento oscilatório .....	30
Figura 11 – Geração da função senoidal .....	31
Figura 12 – Modelo do <i>Bungee Jumping</i> .....	32
Figura 13 – Oscilador massa-mola na configuração horizontal .....	32
Figura 14 – Oscilador massa-mola na configuração vertical .....	33
Figura 15 – Modelo do pêndulo simples .....	33
Figura 16 – Movimento oscilatório de uma régua com uma extremidade livre.....	34
Figura 17 – Curva de resposta de um LDR do tipo CdS disponibilizada na folha de dados do fabricante .....	35
Figura 18 – Curva de resposta do sensor TCRT5000 disponibilizada na folha de dados do fabricante .....	36
Figura 19 – Esquema simplificado da montagem do experimento .....	36
Figura 20 – Diagrama da montagem eletrônica do experimento.....	37
Figura 21 – Representação da referência de distância do sensor .....	38
Figura 22 – Esquema simplificado da montagem do experimento .....	41
Figura 23 – Terminal serial do <i>Arduino</i> recebendo dados, sendo os dez primeiros valores de posição em sinal analógico de 10 bits (0 a 1024) e um valor de tempo em ms. ....	44

Figura 24 – Tela da interface computacional realizando medida de oscilação com indicação de referências para a determinação gráfica do período (vermelho).....	45
Figura 25 – Montagem experimental para realizar medição do período de oscilação .....	47
Figura 26 – Tela do osciloscópio digital com a leitura dos parâmetros da oscilação.....	48
Figura 27 – Atividade 1 – Exemplo de uma das medições realizadas na interface computacional desenvolvida.....	51
Figura 28 – Atividade 2 – Exemplo de arranjo com as moedas para a determinação da constante elástica .....	52
Figura 29 – Atividade 2 – Exemplo de uma das medições realizadas na interface computacional para um dos arranjos de moedas de R\$ 0,50.....	53
Figura 30 – Oscilação na interface computacional para um balanço da régua de 20 cm ....	54
Figura 31 – Amortecimento para uma régua de aço considerando um balanço de 20 cm ..	56
Figura 32 – Esquema de ligações da estação meteorológica com o uso do <i>Arduino</i> e do <i>Processing</i> .....	61
Figura 33 – Detalhe do circuito integrado BMP180.....	62
Figura 34 – Placa comercial do sensor BMP180.....	62
Figura 35 – Vista do sensor AM2302 sem e com o invólucro plástico de proteção .....	63
Figura 36 – Fotografia e esquema do modelo de sensor capacitivo para a medição de umidade relativa .....	64
Figura 37 – Fotografia e esquema do modelo de sensor resistivo para a medição de umidade relativa .....	64
Figura 38 – Fotografia do LDR .....	66
Figura 39 – Curva de resposta do LDR (Resistência versus Lux) .....	66
Figura 40 – Curva de resposta espectral do LDR.....	66
Figura 41 – Interface gráfica desenvolvida com o uso do <i>Processing</i> para a apresentação dos dados meteorológicos adquiridos pelo <i>Arduino</i> .....	81
Figura 42 – Primeiro cíclotron construído por Lawrence.....	88
Figura 43 – Digrama simplificado da trajetória do experimento.....	93
Figura 44 – Detalhe do processo de confecção da bobina.....	94
Figura 45 – Fonte de alimentação e seu conector padrão .....	95



Figura 46 – Diagrama das ligações da fonte de alimentação para seu funcionamento e para alimentação dos ventiladores de resfriamento e do banco de capacitores.....	96
Figura 47 – Esquema de polarização das bobinas e sua interação com a velocidade da esfera .....	96
Figura 48 – Esquema da montagem do experimento .....	97
Figura 49 – Esquema sugerido para a montagem do circuito de controle na placa de contatos .....	98
Figura 50 – Interface computacional do experimento desenvolvida em <i>Processing</i> .....	99
Figura 51 – Atividade 1 – Detalhe do simulador na interface computacional do experimento .....	102
Figura 52 – Atividade 2 – Detalhe da aquisição de dados na interface computacional do experimento .....	104
Figura 53 – Atividade 2 – Detalhe da aquisição de dados na interface computacional do experimento em velocidade máxima .....	105
Figura 54 - Proposta para montagem dos sensores ópticos para a medição da velocidade da esfera na entrada e na saída da bobina.....	106
Figura 55 - Vista parcial do Laboratório de Simulação Eletrônica .....	108
Figura 56.....	109
Figura 57 – Protótipo do acelerador de partículas com o uso do <i>Arduino</i> e do <i>Processing</i> .....	110
Figura 58 – Protótipo do oscilador harmônico amortecido com o uso do <i>Arduino</i> e do <i>Processing</i> .....	110
Figura 59 – Protótipo da estação meteorológica com o uso do <i>Arduino</i> e do <i>Processing</i> .....	111
Figura 60 – Desenvolvimento dos experimentos em sala de aula.....	112
Figura 61 – Fotografia do experimento <i>Oscilador Harmônico Amortecido com o uso do Arduino e do Processing</i> .....	118
Figura 62 – Fotografia da montagem eletrônica do experimento <i>Oscilador Harmônico Amortecido com o uso do Arduino e do Processing</i> .....	118
Figura 63 – Fotografia da montagem do experimento <i>Estação Meteorológica com o uso do Arduino e do Processing</i> .....	119
Figura 64 – Fotografia da montagem eletrônica do experimento <i>Estação Meteorológica com o uso do Arduino e do Processing</i> .....	119
Figura 65 – Fotografia da montagem do experimento <i>Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing</i> .....	120

Figura 66 – Fotografia da montagem do experimento <i>Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing</i> .....	120
Figura 67 – Fotografia do detalhe da montagem da barreira de luz, da bobina e do ventilador de resfriamento no experimento <i>Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing</i> .....	121
Figura 68 – Fotografia da montagem do conjunto bobina e do ventilador de resfriamento no experimento <i>Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing</i> .....	121

## Lista de Gráficos

Gráfico 1 – Curva de resposta do sensor TRTC-5000 para reflexão difusa no intervalo 5-130 mm .....	39
Gráfico 2 – Curva de resposta do sensor TRTC-5000 para reflexão difusa no intervalo 15-80 mm .....	40
Gráfico 3 – Curva de resposta do sensor TRTC-5000 para reflexão difusa no intervalo 15-80 mm .....	42
Gráfico 4 – Gráfico gerado a partir dos dados oriundos da interface computacional .....	46
Gráfico 5 – Atividade 2 – Constante elástica (k) da lâmina nos diversos arranjos de moedas de R\$ 0,50.....	53
Gráfico 6 – Comparação entre os dados de temperatura obtidos pelo sensor BMP180 e pelo Termo higrômetro MT-240 .....	69
Gráfico 7 – Comparação entre os dados de temperatura obtidos pelo sensor AM2302 e pelo Termo higrômetro MT-240 .....	70
Gráfico 8 – Comparação das medidas atinentes a umidade relativa do ar obtidas a partir do sensor AM2302 e do Termo higrômetro MT-240 .....	71
Gráfico 9 – Gráfico da variação resistiva do LDR para diversos níveis de iluminação .	74
Gráfico 10 – Comparação entre os dados de temperatura coletados pela estação meteorológica com o uso o <i>Arduino</i> e do Processing e pela estação meteorológica do Sistema Alerta Rio de São Cristóvão .....	76
Gráfico 11 – Gráfico dos dados referentes à luminosidade, representado em lux, obtidos da estação meteorológica com o uso do <i>Arduino</i> e do Processing.....	77
Gráfico 12 – Gráfico dos dados referentes à luminosidade, representado em níveis digitais de 10 bits, obtidos da estação meteorológica com o uso do <i>Arduino</i> e do Processing.....	77
Gráfico 13 – Gráfico dos dados referentes à luminosidade do dia 22/04/2016, representado em lux, obtidos da estação meteorológica com o uso do <i>Arduino</i> e do Processing .....	78
Gráfico 14 – Gráfico dos dados referentes à luminosidade do dia 22/04/2016, representado em níveis digitais de 10 bits, obtidos da estação meteorológica com o uso do <i>Arduino</i> e do Processing.....	78
Gráfico 15 – Gráfico dos dados referentes à luminosidade, representado em kJ/m <sup>2</sup> , obtidos da estação meteorológica do INMET localizada no Forte de Copacabana .....	79
Gráfico 16 – Gráfico dos dados referentes à luminosidade no dia 22/04/2016, representado em kJ/m <sup>2</sup> , obtidos da estação meteorológica do INMET localizada no Forte de Copacabana .....	79

Gráfico 17 – Comparação entre os dados de umidade relativa do ar coletados pela estação meteorológica com o uso do <i>Arduino</i> e do <i>Processing</i> e pela estação meteorológica do Sistema Alerta Rio de São Cristóvão .....	80
Gráfico 18 – Comparação entre os dados de pressão atmosférica coletados pela estação meteorológica com o uso do <i>Arduino</i> e do <i>Processing</i> e pela estação meteorológica do Sistema Alerta Rio de São Cristóvão .....	80
Gráfico 19 – Atividade 2 – Temperaturas horárias referentes ao dia 25/04/2016.....	86

## Lista de Tabelas

Tabela 1 – Versões do software de desenvolvimento e programação da plataforma <i>Arduino</i> .....	8
Tabela 2 – Alguns dos modelos desenvolvidos pelo Projeto <i>Arduino</i> .....	9
Tabela 3 – Características técnicas do <i>Arduino UNO</i> e similares .....	10
Tabela 4 – Descrição dos pinos de alimentação da placa do <i>Arduino UNO</i> . .....	11
Tabela 5 – Funções especiais de alguns dos pinos de entrada/saída do <i>Arduino UNO</i> .....	12
Tabela 6 – Bibliotecas e funções de programação. ....	16
Tabela 7 – Bibliotecas específicas.....	17
Tabela 8 – Alguns exemplos de <i>shields</i> que podem ser conectadas em placas do projeto <i>Arduino</i> .....	19
Tabela 9 – Alguns sensores para <i>Arduino</i> .....	19
Tabela 10 – Principais versões do software <i>Processing</i> .....	21
Tabela 11 – Algumas funções da biblioteca serial do <i>Arduino</i> .....	22
Tabela 12 – Algumas funções da biblioteca Serial do <i>Processing</i> .....	24
Tabela 13 – Dados da resposta do sensor TRTC-5000 para reflexão difusa no intervalo 5-130 mm.....	39
Tabela 14 – Dados da resposta do sensor TRTC-5000 para reflexão difusa no intervalo 15-80 mm.....	40
Tabela 15 – Determinação do erro relativo no processo de linearização do sensor TRTC-5000 .....	41
Tabela 16 – Dados da resposta do sensor TRTC-5000 para reflexão difusa no intervalo 20 - 60 mm.....	42
Tabela 17 – Determinação do erro relativo na linearização do sensor no experimento .....	43
Tabela 18 – Dados oriundos da interface computacional.....	46
Tabela 19 – Determinação do erro relativo médio do padrão de tempo.....	48
Tabela 20 – Descritores relacionados ao estudo das ondas .....	49
Tabela 21 – Atividade 1 – Exemplo de dados obtidos pela interface <i>Processing</i> a partir da medição de posição de oscilação da régua medida por uma régua auxiliar milimetrada....	50

Tabela 22 – Atividade 1 – Registro dos valores de cálculo do erro relativo.....	50
Tabela 23 – Atividade 2 – Exemplo de dados obtidos, via interface gráfica, para cada arranjo de moedas de R\$ 0,50.....	52
Tabela 24 – Atividade 2 – Exemplo do cálculo da constante elástica para cada arranjo de moedas de R\$ 0,50 .....	52
Tabela 25 – Atividade 3 – Determinação dos parâmetros da oscilação para cada arranjo da régua .....	54
Tabela 26 – Amplitudes de oscilação em diferentes ciclos consecutivos, considerando um balanço da régua de 20 cm .....	55
Tabela 27 - Características técnicas básicas do sensor BMP180 .....	63
Tabela 28 – Características básicas do sensor AOSONG AM2302.....	65
Tabela 29 – Dados da localização geográfica e da altitude das Estações Meteorológicas..	67
Tabela 30 – Dados referentes a Pressão atmosférica coletados nas quatro estações meteorológicas usadas como referência para calibração do sensor BMP180 .....	68
Tabela 31 – Quadro comparativo do erro relativo calculado para os dados referentes à pressão atmosférica obtidos nas quatro estações meteorológicas em análise .....	69
Tabela 32 – Características básicas da medição de temperatura do Termo higrômetro MT-240 .....	69
Tabela 33 – Quadro comparativo do erro calculado para os dados referentes à temperatura obtidos nos sensores BMP180, AM2302 e o Termo higrômetro MT-240.....	70
Tabela 34 – Características básicas da medição de umidade relativa do ar do Termo higrômetro MT-240 .....	71
Tabela 35 – Determinação do erro de leitura da umidade relativa do ar obtida a partir do sensor AM2302 e comparada com o Termo higrômetro MT-240 .....	71
Tabela 36 – Características básicas da medição de luminosidade, em lux, do Luxímetro MLM-1011 .....	72
Tabela 37 – Dados da resposta da digitalização da tensão presente no LDR pelo <i>Arduino</i> para diferentes níveis de iluminamento .....	73
Tabela 38 – Valores de tensão e resistência calculadas a partir dos Dados da resposta da digitalização da tensão presente no LDR pelo <i>Arduino</i> .....	74
Tabela 39 – Determinação do erro relativo no processo de linearização do LDR .....	75
Tabela 40 – Descritores relacionados a Termometria, Calor, Pressão e Energia.....	82
Tabela 41 – Atividade 1 – Registro dos dados meteorológicos coletados no dia 26/04/2016 .....	83

Tabela 42 – Atividade 1 – Registro dos valores médios dos dados meteorológicos coletados no dia 26/04/2016 .....	84
Tabela 43 – Atividade 1 – Registro das tendências de alteração dos dados meteorológicos coletados no dia 26/04/2016 .....	84
Tabela 44 – Atividade 2 – Registro dos dados horários de temperatura referentes ao dia 25/04/2016.....	85
Tabela 45 – Dados referentes a alguns aceleradores de partículas.....	90
Tabela 46 – Descritores relacionados ao estudo do movimento, eletricidade, eletromagnetismo e física moderna .....	100
Tabela 47 – Atividade 1 – Exemplo de dados medidos utilizando o cronometro de um aparelho celular.....	101
Tabela 48 – Atividade 1 – Exemplo do cálculo do erro relativo.....	101
Tabela 49 – Atividade 1 – Exemplo do cálculo de velocidade angular e velocidade linear .....	102
Tabela 50 – Atividade 2 – Exemplo de dados medidos utilizando o cronometro de um aparelho celular.....	103
Tabela 51 – Atividade 2 – Exemplo do cálculo do erro relativo.....	103
Tabela 52 – Atividade 2 – Exemplo do cálculo de velocidade angular e linear.....	104

## Sumário

<b>INTRODUÇÃO .....</b>	<b>1</b>
<b>CAPÍTULO 1 – Objetos educacionais de baixo custo com o <i>Arduino</i> e o <i>Processing</i> ..</b>	<b>4</b>
<b>CAPÍTULO 2 – O <i>Arduino</i> e o <i>Processing</i> no Ensino de Física: Materiais e bibliografia existentes.....</b>	<b>27</b>
<b>CAPÍTULO 3 – Oscilador Harmônico Amortecido com o uso do <i>Arduino</i> e do <i>Processing</i> .....</b>	<b>30</b>
<b>CAPÍTULO 4 – Estação Meteorológica com o uso do <i>Arduino</i> e do <i>Processing</i> .....</b>	<b>58</b>
<b>CAPÍTULO 5 – Analogia funcional do acelerador de partículas com o uso do <i>Arduino</i> e do <i>Processing</i> .....</b>	<b>88</b>
<b>CAPÍTULO 6 – Experiência docente do uso do <i>Arduino</i> e do <i>Processing</i> no Ensino de Física .....</b>	<b>108</b>
<b>CONCLUSÃO.....</b>	<b>113</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>114</b>
<b>APÊNDICE A – Registro fotográfico dos protótipos dos experimentos .....</b>	<b>118</b>
<b>APÊNDICE B – Roteiro de atividades para os experimentos .....</b>	<b>122</b>



O ensino de ciências, e principalmente o Ensino de Física, deve estar amparado em práticas pedagógicas que favoreçam a associação entre o ensino teórico, ministrado em sala de aula, e a prática desenvolvida, preferencialmente, nos laboratórios. As atividades práticas são o momento pedagógico de comprovação e de sedimentação dos modelos apresentados aos alunos nas atividades teóricas.

A proposta apresentada nesta dissertação é o desenvolvimento de projetos colaborativos e participativos, entre alunos do Ensino Médio e professores, para a construção de experimentos de baixo custo com o uso do *Arduino* para a aquisição de sinais/dados oriundos de sensores e do *Processing* para o processamento destes sinais/dados e apresentação de resultados numéricos ou sob a forma de gráficos em uma interface computacional.

O uso do *Arduino* e do *Processing* propicia um aumento de qualidade e de confiabilidade nos resultados alcançados nos experimentos, permitindo a obtenção de um padrão de qualidade análogo ou superior daqueles obtidos pelos equipamentos produzidos comercialmente para os laboratórios de física. Estes equipamentos, para além do alto custo de aquisição e da dificuldade de manutenção, ainda possuem arquitetura no modelo “caixa preta”. Tal modelo não favorece a autonomia dos alunos, uma vez que eles, apenas, realizam o experimento e acessam seus resultados, porém não chegam a conhecer os métodos utilizados para a obtenção/aquisição desses resultados nem tão pouco os princípios físicos envolvidos.

O *Arduino* realiza a digitalização das medições das variáveis físicas através de sensores que transformam essas variáveis em variáveis elétricas. O microcontrolador também propicia o envio dos dados coletados e digitalizados ao computador através da interface serial. Para a aquisição e o tratamento, em tempo real, destes dados, há uma carência de softwares ou plataformas gratuitas. Nesse contexto, o uso dos códigos apresentados nesta dissertação, com a utilização da linguagem *Processing*, se apresentam como uma real possibilidade para a aquisição e para a apresentação de resultados tanto na forma de dados quanto na forma de análises gráficas, em tempo real e de forma gratuita e aberta, favorecendo o desenvolvimento de experimentos didáticos com interface computacional a baixo custo.

Neste trabalho, há o desenvolvimento da linearização e da comparação a medidores comerciais das medições obtidas, procedimento este, fundamental para a correta observância dos fenômenos físicos. Este procedimento realizado com o uso do Arduino e do Processing nesta dissertação não possui similar na literatura.

Para cumprir com esse objetivo o presente trabalho foi dividido em quatro partes. No primeiro capítulo, *Objetos educacionais de baixo custo com o Arduino e o Processing*, serão abordados os conceitos básicos da plataforma *Arduino* e da linguagem de programação *Processing*. Quanto a plataforma microcontrolada *Arduino*, serão apresentadas a sua arquitetura básica, seus recursos de Hardware e Software e suas possibilidades de conexão com sensores para a construção de experimentos físicos, tanto quantitativos quanto qualitativos. Com relação a linguagem *Processing* será abordada a sua integração com o *Arduino* para o desenvolvimento de uma interface computacional que possibilite a apresentação dos dados adquiridos na forma numérica e/ou na forma de gráficos.

No segundo capítulo, *O Arduino e o Processing no Ensino de Física: Materiais e bibliografia existentes*, será apresentada uma sucinta revisão da literatura sobre o uso das novas tecnologias no ensino de Física, com foco na aplicação do *Arduino* e do *Processing* em atividades experimentais na sala de aula.

O terceiro capítulo, *Oscilador Harmônico Amortecido com o uso do Arduino e do Processing*, será apresentada uma aplicação para sensores ópticos. Nesta aplicação será desenvolvida uma metodologia de linearização de um detector óptico para possibilitar a medição de posição de uma régua, sem contato mecânico e em tempo real. O experimento e a sua interface computacional que serão apresentados no desenvolvimento desse capítulo e estão acompanhados com sugestões de abordagem para o Ensino de Física em turmas do Ensino Médio.

Já o quarto, será proposto a construção de uma Estação Meteorológica com o uso do *Arduino* e do *Processing*, com sensores capazes de medir a pressão atmosférica, a umidade relativa do ar, a temperatura do ar e a luminosidade. No desenvolvimento do capítulo serão apresentados os princípios físicos de funcionamento dos sensores, bem como serão realizadas comparações dos dados medidos por estes sensores com os dados disponibilizados de estações meteorológicas oficiais e com valores obtidos por instrumentos de medição. Estas comparações têm por finalidade validar a qualidade dos dados obtidos pela estação construída. Também serão apresentadas propostas de como abordar e trabalhar este tema em turmas do Ensino Médio.

No quinto capítulo, *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*, será apresentado um experimento que tem por finalidade principal a inserção prática dos conteúdos da Física Moderna nas turmas de Ensino Médio. Este experimento possibilita a análise, pelos alunos, do princípio de funcionamento dos aceleradores de partículas, perpassando por conhecimentos de eletricidade, de eletromagnetismo, de movimento circular e de energia cinética. Ao final deste capítulo também são apresentadas propostas de aplicação do experimento nas turmas de Ensino Médio.

No sexto capítulo, apresentamos os resultados da aplicação, em sala de aula e na plataforma Moodle, dos roteiros de atividades, presentes nos apêndices, analisando as respostas dadas pelos alunos em aulas com e sem a aplicação dos experimentos.

Na parte final do trabalho, os apêndices, são apresentados os roteiros de atividades dos experimentos que são sugestões de como esses experimentos podem ser abordados na prática pedagógica do docente em turmas de Ensino Médio. Esse material também é composto do registro fotográfico e os detalhes da montagem dos experimentos, inclusive contendo a lista de materiais necessários e o custo de cada uma das montagens, que favorecem o planejamento da montagem e a sua reprodução. E por fim, os códigos do Arduino e os códigos do Processing são apresentados na íntegra para serem utilizados nos experimentos ou serem customizados pelos alunos ou docentes.

### **Objetos educacionais de baixo custo com o *Arduino* e o *Processing***

Para favorecer o aprendizado dos conteúdos apresentados no ensino de ciências e, principalmente, no Ensino de Física, as aulas deveriam ser amparadas em práticas pedagógicas que favoreçam uma associação entre o ensino teórico ministrado em sala de aula e a prática desenvolvida nos laboratórios didáticos. Uma vez que as atividades experimentais deveriam ser o momento pedagógico de comprovação e de sedimentação dos modelos apresentados aos alunos nas atividades teóricas, fazendo, também, com que eles passem a ver, através da utilização de experimentos, a Física como algo presente em seu cotidiano, promovendo o interesse de investigar e tirar conclusões, deixando, assim, de ser uma disciplina cheia de leis, conceitos e exercícios repetitivos, minimizando as dificuldades de aprendizado e propiciando o Ensino de Física de maneira significativa e consistente. A oportunidade de conhecer, de manipular e de controlar os experimentos, acaba por facilitar o aprendizado dos conceitos e por despertar o interesse do aluno.

Essa visão do laboratório didático é proposta pelos PCN's (Parâmetros Curriculares Nacionais), desenvolvidos pelo Ministério da Educação. Estes parâmetros preconizam a utilização de objetos educacionais que simulem situações problema ou tentativas de solução, e não apenas uma proposta realizada com o intuito de verificar através de atividades experimentais apenas as leis e teorias previamente determinadas, onde muitas vezes o aluno não possui tempo de entender ou de relacionar o significado das atividades:

É indispensável que a experimentação esteja sempre presente ao longo de todo o processo de desenvolvimento das competências em Física, privilegiando-se o fazer, manusear, operar, agir, em diferentes formas e níveis. É dessa forma que se pode garantir a construção do conhecimento pelo próprio aluno, desenvolvendo sua curiosidade e o hábito de sempre indagar, evitando a aquisição do conhecimento científico como uma verdade estabelecida e inquestionável. (PCN+, 2013, p.81)

No entanto, a realidade das escolas públicas do Brasil está bem distante deste ideal. Segundo o último levantamento divulgado pelo Observatório<sup>1</sup> do PNE (Plano Nacional de

---

<sup>1</sup> O Observatório do PNE é uma plataforma online que tem como objetivo monitorar os indicadores referentes a cada uma das 20 metas do Plano Nacional de Educação (PNE) e de suas respectivas estratégias, e oferecer análises sobre as políticas públicas educacionais já existentes e que serão implementadas ao longo dos dez anos de vigência do Plano.

Educação), em 2014 somente 8,4%<sup>2</sup> das unidades escolares de Ensino Básico da Rede Pública de Ensino contavam com um laboratório de Ciências em suas instalações.

Ademais, laboratórios modernos, com equipamentos sofisticados, nem sempre garantem bons resultados, pois estão, muitas das vezes, distantes da vivência dos alunos:

As abordagens mais tradicionais precisariam, portanto, ser revistas, evitando-se “experiências” que se reduzem à execução de uma lista de procedimentos previamente fixados, cujo sentido nem sempre fica claro para o aluno. É tão possível trabalhar com materiais de baixo custo, tais como pedaços de fio, pequenas lâmpadas e pilhas, quanto com kits mais sofisticados, que incluem multímetros ou osciloscópios. A questão a ser preservada, menos do que os materiais disponíveis, é, novamente, que competências estarão sendo promovidas com as atividades desenvolvidas. (PCN+, 2013, p.81).

Experimentos de baixo custo não são, exatamente, uma novidade<sup>3</sup>. Alguns já têm sido utilizados, e não apenas pela razão financeira, que os caracteriza, mas pela facilidade de montagem e exemplificação de fenômenos físicos em ambientes desprovidos de um laboratório didático. No entanto, muitos desses experimentos limitam-se a aspectos qualitativos, sendo incapazes de realizar medições com a precisão adequada.

Neste sentido, o uso de microcontroladores eletrônicos torna-se uma opção interessante na criação de dispositivos de baixo custo, pois possuem capacidade de medições com precisão adequada e com grande versatilidade de aplicações. Entre elas podemos citar o uso dos sensores de pressão, temperatura, campo magnético, altitude, umidade, posição, dentre outros. Porém a sua utilização possui uma forte resistência devido à necessidade de conhecimentos prévios de eletrônica básica e de programação, o que por si só restringe o público-alvo e seu viés de divulgação científica.

A utilização destes materiais, em geral, permite que se realizem experimentos físicos sem a necessidade de ambientes especiais para esse fim. Além disso, os fenômenos não ficam “ocultos” em equipamentos que o estudante não sabe exatamente como funcionam. Para Ribeiro, “aparelhos e montagens improvisadas, executadas com os recursos mais modestos laboratórios, deve ser considerada não como uma solução de emergência, mas ao contrário, como uma nova técnica desejável para desenvolver as capacidades construtivas e inventivas do estudante” (1955, p.54).

---

<sup>2</sup> Laboratório de ciências está presente em 8,19% das Escolas de Ensino Fundamental e em 44,40% das Escolas de Ensino Médio.

<sup>3</sup> Pelo menos desde a década de 90 do século XX, são propostos experimentos de baixo custo em substituição aos precários ou inexistentes laboratórios de Física no Brasil. Vide Instrumentação para o Ensino de Física em Níveis de 1º e 2º Graus In: Revista Brasileira de Estudos Pedagógicos, 1993, p.713-738.

Neste cenário, a plataforma *Arduino* pode ser uma opção extremamente barata e de fácil programação, não exigindo de seu usuário um conhecimento rebuscado em eletrônica e programação. Além disso, esta plataforma é de acesso livre, sendo seus códigos amplamente compartilhados pelos seus diversos usuários na internet.

A maior vantagem do Arduino sobre outras plataformas de desenvolvimento de microcontroladores é a facilidade de sua utilização; pessoas que não são da área técnica podem, rapidamente, aprender o básico e criar seus próprios projetos em um intervalo de tempo relativamente curto. Artistas, mais especificamente, parecem considerá-lo a forma perfeita de criar obras de arte interativas rapidamente, e sem conhecimento especializado em eletrônica. Há uma grande comunidade de pessoas utilizando Arduinos, compartilhando seus códigos e diagramas de circuito para que outros os copiem e modifiquem. A maioria dessa comunidade também está muito disposta a auxiliar outros desenvolvedores (McRoberts, 2011, p.20).

Aliás, o uso de microcontroladores na eletrônica não é recente, se iniciou na primeira metade da década de 1970, com o desenvolvimento, pela *Texas Instruments*, do TMS-1000, que era um microcontrolador de 4 bits com memórias incorporadas. Esse microcontrolador foi usado nas linhas de produção das calculadoras *Texas* entre 1972 a 1974 e foi disponibilizado para uso das indústrias eletrônicas a partir de 1974.

Assim como a *Texas Instruments*, a *Intel*, em 1976, lançou o 8048 que foi um dos primeiros microcontroladores desta empresa. O 8051 o seguiu em 1980, e se tornou uma das famílias mais populares de microcontroladores. Variantes da arquitetura do 8051 ainda são produzidas atualmente, tornando o 8051 um dos projetos eletrônicos mais duradouros da história.

O uso de microcontroladores nas escolas técnicas e nos cursos de engenharia se iniciou na década de 1980, com circuitos complexos e a necessidade de conhecimento técnico avançado, limitando, assim, a difusão deste conhecimento em outras áreas da ciência.

Em 2005, foi desenvolvido o projeto *Arduino*, uma plataforma de prototipagem eletrônica, de código aberto, que utiliza um microcontrolador para implementar de maneira simples e com baixo custo os mais diversos circuitos e projetos nas áreas de eletrônica e automação. Em termos práticos:

Um Arduino é um pequeno computador que você pode programar para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele. O Arduino é o que chamamos de plataforma de computação física ou embarcada, ou seja, um sistema que pode interagir com seu ambiente por meio de hardware e software (McRoberts, 2011, p.22).

O Projeto do *Arduino* nasceu no *Interaction Design Institute* na cidade de Ivrea, comuna italiana da região do Piemonte, província de Turim, em 2005. O professor do

Instituto, Massimo Banzi buscava um meio de possibilitar o uso de tecnologia nos projetos dos estudantes de design, com baixo custo e facilidade de aprendizado e utilização.

Com a colaboração de David Cuartielles, pesquisador visitante da Universidade de Malmö, na Suécia e Gianluca Martino, um engenheiro italiano, foi produzida, no Instituto, uma tiragem inicial de 200 placas, que foram rapidamente vendidas.

A nova placa foi chamada *Arduino* em referência a um bar local *Bar di Re Arduino*, frequentado por membros do corpo docente e alunos do Instituto. O nome do bar homenageia um rei europeu do início do século XI. A popularidade desta placa cresceu rapidamente quando o grande público percebeu sua fácil utilização e seu baixo custo.

As primeiras placas da plataforma *Arduino* iniciaram com comunicação serial e componentes discretos e eram vendidas desmontadas em kits ou apenas a placa de circuito impresso. Essas primeiras versões ainda não apresentavam o nome *Arduino*, porém são conhecidas como *Arduino Serial* e *Arduino Serial v2*.

A placa *Arduino USB* foi lançada após a popularização da conexão USB<sup>4</sup> nos computadores. Sua primeira versão apresentava um erro na ligação elétrica do conector USB, limitando sua utilização. Para corrigir este problema foi logo lançada a placa *Arduino USB v2* que corrigia este problema.

Utilizando o padrão SMD<sup>5</sup> de montagem dos componentes adicionais ao microcontrolador, o que viabiliza a produção dessas placas em larga escala, foram lançadas as placas *Arduino extreme* e *Arduino extreme v.2*. Essa versão já trazia impresso em sua placa o link [www.arduino.cc](http://www.arduino.cc).

Com um novo controle da porta USB foram lançadas as placas *Arduino NG* e *Arduino NG v2*, essa última já utilizando o microcontrolador ATMEGA168 em vez do ATMEGA8 que foi utilizado nas placas anteriores.

Em 2007, foi desenvolvido um melhor controle de programação pela interface USB, gerando uma nova placa, o *Arduino diecimila*. Em 2008 foi alcançado o número de 50.000 placas vendidas. A placa *Arduino duemilanove* implementou a funcionalidade de seleção automática de fonte externa/USB. A partir de março de 2009, esta versão passou a ser equipada com o microcontrolador ATMEGA328, duplicando sua capacidade de memória.

---

<sup>4</sup> USB é a sigla para *Universal Serial Bus* – Barramento Serial Universal. Tecnologia que possibilita conexões entre computadores e periféricos de forma simples e praticamente universal. Nessa conexão os dados são transmitidos de forma serial.

<sup>5</sup> Surface Mounting Devices ou dispositivos para montagem em superfície. Refere-se a técnica de montagem de componentes na superfície de placas de circuito eletrônico. Esta técnica de produção minimiza o tempo necessário para montagem e aumenta consideravelmente a produtividade.

Em 2010, após um novo desenvolvimento, foi lançada a placa *Arduino UNO*, a mais popular das placas *Arduino*. Atualmente em sua 3ª revisão, esta placa é a padrão para projetos com o uso desta plataforma. Em 2011 foi alcançado o patamar de 500.000 placas vendidas. Com o objetivo de acompanhar o desenvolvimento do hardware, foram lançadas algumas versões do software de desenvolvimento e programação da plataforma *Arduino*.




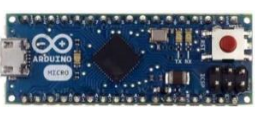
Listamos a seguir, suas principais versões e suas datas de lançamento:

1.6.8	09/03/2016	1.0.4	11/03/2013
1.6.7	17/12/2015	1.0.3	10/12/2012
1.6.6	03/11/2015	1.0	30/11/2011
1.6.5	15/06/2015	0023	09/11/2011
1.6.4	06/05/2015	0022	24/12/2010
1.5.8 BETA	01/10/2014	0017	25/07/2009
1.5.5 BETA	28/11/2013	0012	18/09/2008
1.5.1 BETA	05/11/2012	0010	11/10/2007
1.5 BETA	22/10/2012	0007	25/12/2006
1.0.6	16/09/2014	0001	25/08/2005

Tabela 1 – Versões do software de desenvolvimento e programação da plataforma *Arduino*  
 Fonte: *Arduino Software Release Notes*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

*Arduino* é uma plataforma de *hardware* e *software open source*<sup>6</sup>, programável, que permite interagir com o mundo a partir da aquisição de variáveis físicas através de sensores e interagir controlando luzes, motores e outros atuadores. O projeto *Arduino* é disponibilizado em diversas versões para atender aplicações bem diferenciadas.

Relacionamos, a seguir, algumas destas versões que são disponibilizadas comercialmente.

<i>Arduino UNO</i>		<i>Arduino Mini</i>	
<i>Arduino Leonardo</i>		<i>Arduino Micro</i>	

<sup>6</sup> Código aberto, ou *open source*, é um modelo de desenvolvimento que disponibiliza todas as informações a todos os usuários. Apesar dos projetos desenvolvidos sob esta ideologia possuírem boa qualidade, a gratuidade é o principal atrativo.



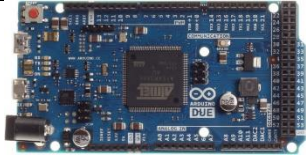





<i>Arduino Due</i>		<i>Arduino Nano</i>	
<i>Arduino Mega</i>		<i>Arduino LilyPad</i>	
<i>Arduino YUN</i>		<i>Arduino Gemma</i>	

Tabela 2 – Alguns dos modelos desenvolvidos pelo Projeto *Arduino*  
 Fonte: *Arduino Products*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

Pelo fato do projeto ser *open source* e de ser disponibilizado sob uma licença *Creative Commons*<sup>7</sup>, é possível copiar, modificar, ou desenvolver um projeto baseado nos diagramas e layout das placas, disponibilizadas pelo projeto *Arduino*. A única restrição é o uso do nome *Arduino*, que é exclusivo da equipe que desenvolveu esta plataforma. Por isso encontra-se no mercado, os originais *Arduino*, produzidos na Itália, e vários compatíveis, porém com outros nomes e países de origem como: *Freeduino*, *Severino*, *Sanguino*, *Roboduino*, *Seeduino*, *Garagino*, dentre outros.

Dentre todas as plataformas, as compatíveis com o *Arduino UNO* apresentam uma ótima relação custo x benefício, sendo assim se tornam um excelente ponto de partida para o usuário iniciante e para o uso em aplicações de baixa complexidade atrelada a um baixo custo. Baseado no microcontrolador ATmega328, produzido pela ATMEL<sup>8</sup>, o *Arduino UNO* possui 14 pinos de entrada/saída digital, dos quais 6 pinos podem ser usados como saídas PWM (*Pulse Width Modulation* - Modulação por Largura de Pulso), 6 entradas analógicas, uma conexão USB, uma entrada de alimentação, uma conexão ICSP (*In-Circuit Serial Programming* - Conector Serial de Programação), um botão de reset, LEDs (*Light Emissor*

<sup>7</sup> *Creative Commons* é uma entidade sem fins lucrativos criada para permitir maior flexibilidade na utilização de obras protegidas por direitos autorais. Suas licenças possibilitam que um autor/criador permita o uso mais amplo de seus materiais por terceiros, sem que estes o façam infringindo as leis de proteção à propriedade intelectual.

<sup>8</sup> A *Atmel Corporation* desenvolve projetos e fabrica microcontroladores, soluções em tecnologia sensível ao toque capacitivas, lógica avançada, mixagem de sinais, memórias não voláteis e componentes para radiofrequência (RF) para os mercados industrial, de consumo, segurança, comunicações, computação e automotivo.

*Diode* – Diodo Emissor de Luz) indicativos do estado do pino 13 e da transmissão/recepção de dados através da interface serial (Pinos 0 e 1). A placa contém todos os componentes necessários para o adequado funcionamento do microcontrolador e sua interface com o computador.

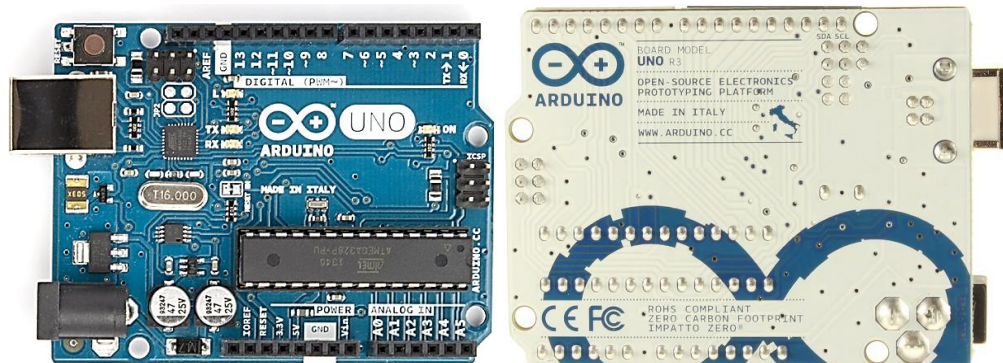


Figura 1 – A placa *Arduino UNO*

Microcontrolador	ATmega328
Tensão Operacional em corrente contínua	5V
Tensão de entrada em corrente contínua (recomendada)	7-12V
Tensão de entrada em corrente contínua (limites)	6-20V
Pinos E/S digitais	14 (dos quais 6 podem ser saídas PWM)
Pinos de entrada analógica	6
Corrente contínua máxima por pino E/S	40 mA (total de 200 mA)
Corrente contínua máxima para o pino de 3,3V	50 mA
Flash Memory <sup>9</sup>	32 KB dos quais 0,5KB são utilizados pelo bootloader <sup>10</sup>
SRAM <sup>11</sup>	2 KB
EEPROM <sup>12</sup>	1 KB
Frequência de Clock <sup>13</sup>	16 MHz

Tabela 3 – Características técnicas do *Arduino UNO* e similares  
 Fonte: *Overview Arduino UNO*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

Para o uso do *Arduino UNO* em aplicações práticas, torna-se necessário uma descrição mais detalhada dos conectores presentes em sua placa.

<sup>9</sup> Memória do tipo não volátil, ou seja, não necessita de energia para manter as informações armazenadas. Empregada para armazenar o programa a ser executado pelo *Arduino*.

<sup>10</sup> Código que é gravado previamente no microcontrolador e realiza a função de gerenciador de inicialização.

<sup>11</sup> *Static Random Access Memory* - memória de acesso aleatório – tipo de memória que mantém os dados armazenados desde que seja mantida sua alimentação. Empregada como memória temporária durante a execução do programa.

<sup>12</sup> *Electrically Erasable Programmable Read-Only Memory* - Memória não-volátil que é utilizada para criar variáveis que não terão seus valores apagados quando o *Arduino* for desligado.

<sup>13</sup> É a velocidade de processamento. Quanto maior a frequência de *clock* de um processador maior será o número de instruções executadas em um segundo.

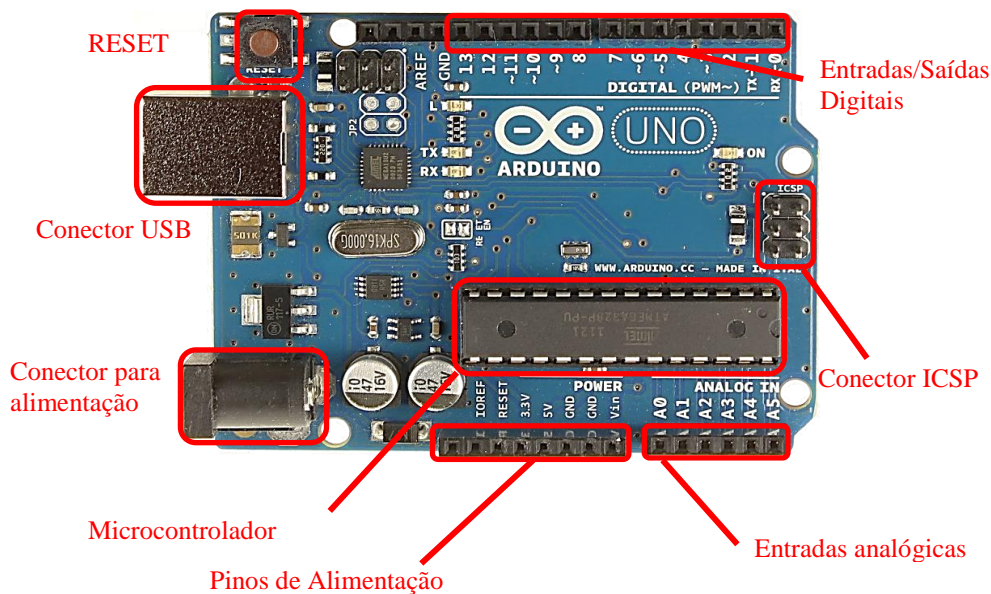


Figura 2 – Identificação dos pinos de conexão e dos principais componentes da placa do *Arduino UNO*

A placa pode ser alimentada pela conexão USB, com uma fonte de alimentação externa ou por uma bateria acoplada ao conector para alimentação. A alimentação é selecionada automaticamente. Uma outra possibilidade, porém, menos recomendada, é a alimentação por meio dos pinos Vin e GND. A tensão da fonte externa pode variar de 6 a 20 volts. No entanto, recomenda-se uma tensão entre 7V e 12V.

Apresentamos a seguir a descrição dos pinos de alimentação da placa *UNO*:

VIN	Entrada de alimentação para a placa quando se está utilizando uma fonte de alimentação externa, em oposição à conexão USB ou outra fonte de alimentação regulada ligada através do conector de alimentação. Quando a placa é alimentada através do conector Jack, a tensão fornecida pela fonte estará nesse pino.
GND	Pinos terra, tensão 0 V de referência.
5V	Fornecer tensão de 5 V para alimentação de sensores, placas auxiliares e circuitos externos. A corrente máxima suportada pelo microcontrolador é de 200 mA.
3V3	Fornecer tensão de 3,3 V para alimentação de sensores, placas auxiliares e circuitos externos. A corrente máxima suportada é de 50mA.
RESET	Pino conectado ao RESET do microcontrolador. Pode ser utilizado para um reset externo do microcontrolador. Quando este pino comuta para o valor LOW (0V) o microcontrolador é reiniciado.
IOREF	Fornecer uma tensão de referência para que placas dedicadas ao Arduino ( <i>shields</i> ) possam selecionar o tipo de interface apropriada, dessa forma <i>shields</i> que funcionam com as placas <i>Arduino</i> que são alimentadas com 3,3V podem se adaptar para ser utilizados em 5V e vice-versa.

Tabela 4 – Descrição dos pinos de alimentação da placa do *Arduino UNO*.  
Fonte: *Documentation Arduino UNO*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

Cada um dos 14 pinos digitais presentes na placa do *Arduino UNO* pode ser utilizado como uma entrada ou uma saída digital e podem apresentar duas tensões distintas 0 (LOW – Nível baixo) e 5v (HIGH – Nível alto), com uma corrente máxima de 40mA. Alguns desses pinos tem funções secundárias específicas, conforme apresentadas a seguir:

Serial	Pinos 0 (RX) e 1 (TX). Usados para receber (RX) e transmitir (TX) dados seriais TTL.
Interrupções Externas	Pinos 2 e 3. Estes pinos podem ser configurados para disparar uma interrupção de acordo com alguma variação de parâmetro percebida pelo circuito.
SPI (Serial Peripheral Interface ou Interface de comunicação serial para periféricos)	Pinos 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). A SPI é um protocolo de dados seriais síncronos utilizado em microcontroladores para comunicação entre o microcontrolador e um ou mais periféricos. Também pode ser utilizado entre dois microcontroladores. A comunicação SPI sempre tem um <i>master</i> . Isto é, sempre um será o <i>master</i> e o restante será <i>slave</i> . Por exemplo, o <i>Arduino</i> é o <i>master</i> e os outros periféricos são <i>slave</i> . Esta comunicação contém 4 conexões: MISO ( <i>Master IN Slave OUT</i> ) - Dados do <i>Slave</i> para <i>Master</i> ; MOSI ( <i>Master OUT Slave IN</i> ) - Dados do <i>Master</i> para <i>Slave</i> ; SCK ( <i>Serial Clock</i> ) - <i>Clock</i> de sincronização para transmissão de dados entre o <i>Master</i> e <i>Slave</i> ; SS ( <i>Slave Select</i> ) - Seleciona qual <i>Slave</i> receberá os dados.
LED	Pino 13. Há um LED integrado ao pino digital 13. Quando este pino está no valor <i>HIGH</i> este LED está aceso, quando o pino está em <i>LOW</i> o LED está apagado.
I2C (Inter Integrated Circuit)	Pinos 4 (SDA) e 5 (SCL). Fornecem suporte a comunicação I2C. I2C é um protocolo de comunicação (ou bus), ou seja, com os mesmos fios conectamos todos os dispositivos da rede I2C. Essa característica, de barramento, é um dos grandes atrativos do I2C, pois reduzimos em muito a necessidade de pinos de conexão no <i>Arduino</i> , pois usaremos sempre os mesmos fios para a conexão, não importa se estamos utilizando 1 ou 127 dispositivos. SDA significa <i>Serial Data</i> e SCL significa <i>Serial Clock</i> . O SDA é o pino que efetivamente transfere os dados, e o SCL serve para temporização entre os dispositivos, de modo que a comunicação pela SDA possa ter confiabilidade. Tanto o envio quanto a recepção de dados é realizada utilizando a linha SDA, ou seja, é uma linha bi-direcional de comunicação, ora estamos enviando dados por este pino, ora estamos recebendo dados.

Tabela 5 – Funções especiais de alguns dos pinos de entrada/saída do *Arduino UNO*.  
 Fonte: *Documentation Arduino UNO*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

O microcontrolador apenas processa dados digitais, sendo assim se torna necessário converter o sinal analógico em digital. O conversor Analógico digital, ADC ou conversor A/D, presente no projeto do *UNO*, quantifica o valor analógico conforme a quantidade de bits da sua resolução. O tempo para essa conversão é de aproximadamente 100 microsegundos, dessa forma a máxima frequência de leitura que pode ser obtida é de 10.000 vezes por segundo ou 10 kHz.

O *Arduino UNO* possui 6 entradas analógicas, identificadas de A0 a A5, cada uma tem 10 bits de resolução (1024 valores diferentes, de 0000000000 a 1111111111). Por padrão elas medem de 0 a 5,0 V, embora seja possível alterar o limite superior utilizando o pino AREF. Considerando o padrão de 0 a 5,0 volts podemos concluir que a resolução nas entradas analógicas é de, aproximadamente, 4,89 mV.

$$Resolução = \frac{V_{referência}}{2^{numero\ de\ bits}}$$

$$Resolução = \frac{5}{2^{10}} = \frac{5}{1024} = 0,0048828125\ V \approx 4,89mV$$

Considerando a polarização do pino AREF (Referência de tensão para entrada analógica) com 1,0 V, notamos um aumento significativo na resolução da medida, porém o intervalo de medição ficará restrito ao intervalo de 0 a 1,0 volt.

$$Resolução = \frac{1}{2^{10}} = \frac{1}{1024} = 0,0009765625\ V \approx 0,98mV$$

O *Arduino UNO* não possui saídas analógicas, somente digitais. O PWM (*Pulse Width Modulation* ou Modulação por largura de pulso), é uma técnica utilizada por sistemas digitais para emular uma saída analógica através da variação do valor médio de uma forma de onda periódica. A técnica consiste em manter a frequência de uma onda quadrada fixa e em variar o tempo que o sinal fica em nível lógico alto. Esse tempo é chamado de ciclo de trabalho (*duty cycle*), ou seja, o ciclo ativo da forma de onda.

Este PWM tem uma resolução de 8 bits, ou seja, é possível variar o valor de saída em 256 valores começando do 0 (ciclo de trabalho 0%) e chegando no seu valor máximo de 255 (ciclo de trabalho 100%), com incremento mínimo de 0,39%. Os pinos 3,5,6,10 e 11 são disponibilizados para o uso de PWM no *Arduino UNO*.

Incremento mínimo de ciclo de trabalho

$$\frac{100\%}{2^8} = \frac{100\%}{256} = 0,390625\% = 0,39\%$$

Determinação da tensão de saída:

$$V_{saída} = \left( \frac{\text{ciclo de trabalho}}{100} \right) * V_{cc}$$

Se o ciclo de trabalho = 40%, temos:

$$V_{saída} = \left( \frac{40}{100} \right) * 5 = 2 V$$

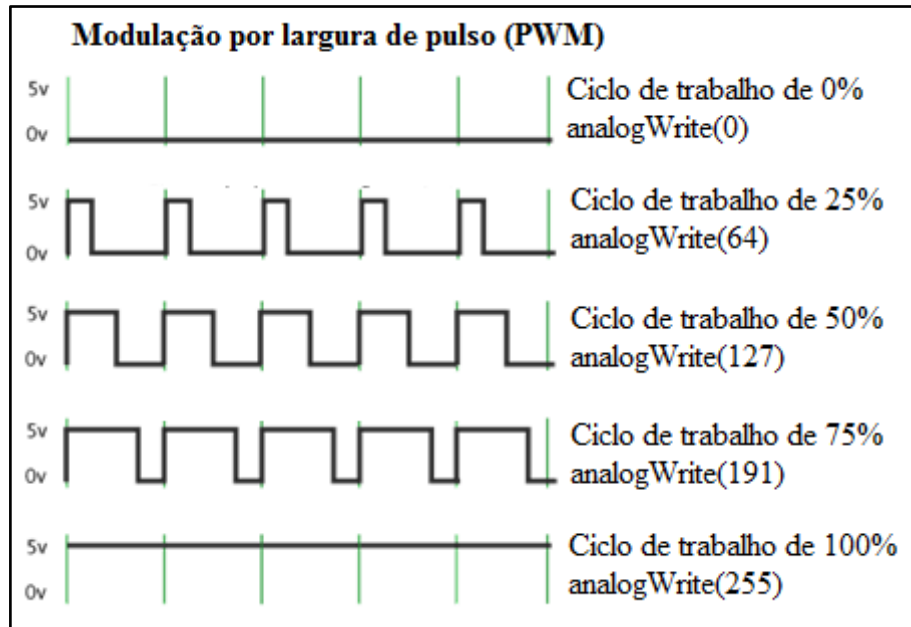


Figura 3 – Demonstração de alguns níveis de PWM  
(Adaptada do original, disponibilizado em: <http://arduino.cc/en/Tutorial/PWM>)

O *Arduino UNO* possui uma série de facilidades para se comunicar com um computador, com outro *Arduino*, ou com outros microcontroladores. O ATmega328 fornece comunicação serial UART TTL (5V), que está disponível nos pinos digitais 0 (RX) e 1 (TX). Um ATmega8U2<sup>14</sup> na placa direciona esta comunicação para a interface USB que aparece como uma porta virtual para o sistema operacional do computador. O software do *Arduino* inclui um monitor serial que permite o envio e o recebimento de dados textuais pela placa. LEDs conectados ao RX e TX piscarão enquanto os dados estiverem sendo transmitidos pelo chip USB-para-serial e pela conexão USB (mas não para comunicação serial nos pinos 0 e 1).

Para a programação do *Arduino*, é utilizada uma linguagem de programação própria, baseada no *Wiring*<sup>15</sup>, linguagem de programação oriunda do C++. O ambiente de desenvolvimento (IDE - *Integrated Development Environment* ou Ambiente Integrado de

<sup>14</sup> Microcontrolador de 8 bits de arquitetura AVR com interface USB implementada, 8K de memória flash.

<sup>15</sup> Iniciado em 2003 por Hernando Barragan, o projeto wiring é um framework de código aberto desenvolvido exclusivamente para programação de microcontroladores. Atualmente é desenvolvido na Escola de Arquitetura e Design de Los Andes em Bogotá, Colômbia.

Desenvolvimento), é baseado no ambiente *Processing*<sup>16</sup> e pode ser utilizado nos mais diversos sistemas operacionais, como *Windows* ou *Linux*, bastando apenas realizar o download da versão que corresponda ao seu sistema operacional no endereço:

<https://www.arduino.cc/en/Main/Software>

O ambiente de desenvolvimento é uma plataforma de programação composta de um editor de texto para escrever o programa, de uma área de mensagens, de uma barra de ferramentas para funções mais comuns, e de uma série de menus. Este software se comunica com a placa, através de um cabo USB, e transfere o programa para o mesmo. Os programas escritos para *Arduino* neste ambiente de desenvolvimento são chamados de *sketches*.

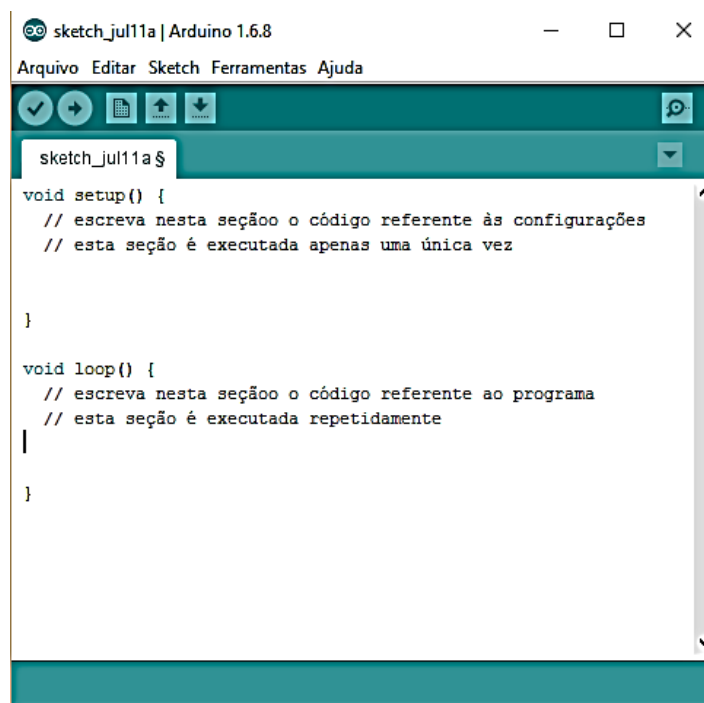


Figura 4 – Tela do software de programação Arduino

Antes de um *sketch* ser transferido para a placa, o código é verificado quanto a sua sintaxe, é compilado pelo compilador *avr-gcc* (compilador apenas para os microcontroladores da ATMEL) e transformado em um arquivo intermediário, conhecido como arquivo objeto. Ao arquivo objeto, são adicionadas as bibliotecas<sup>17</sup> (*libraries*) necessárias e realiza-se finalmente a conversão em código de máquina. Este código contém instruções compreensíveis pelo microcontrolador e praticamente incompreensíveis para nós.

<sup>16</sup> Software de desenvolvimento de desenhos e interfaces.

<sup>17</sup> Bibliotecas ou *libraries* são programas que implementam funções específicas que facilitam a programação. Ex: Biblioteca *math.h* implementa as funções matemáticas complexas, como seno, cosseno, log, etc.

Para aproximar a linguagem de programação à linguagem humana foram criadas as funções. Estas são referências essenciais para o desenvolvimento de um programa ou código, principalmente para os usuários iniciantes. As funções, já implementadas e as disponíveis em bibliotecas, direcionam e exemplificam as funcionalidades básicas do microcontrolador. Temos como exemplo de funções básicas e de referências as seguintes:

Digital I/O (entradas e saídas digitais)	pinMode(pino, modo); Define o pino referenciado como entrada ( <i>INPUT</i> ) ou saída ( <i>OUTPUT</i> ) digital
	digitalWrite(pino, valor); Escreve no pino referenciado o valor ( <i>LOW</i> ou <i>HIGH</i> )
	digitalRead(pino); Lê o valor do pino referenciado ( <i>LOW</i> ou <i>HIGH</i> )
Analog I/O (entradas e saídas analógicas)	analogRead(pino); Lê o valor analógico do pino referenciado (0 a 1023)
	analogWrite(pino, valor); Escreve no pino referenciado o valor (0 a 255) - PWM
Timer (Temporizadores)	delay(tempo em ms); Atraso de milissegundos
	delayMicroseconds(tempo em µs); Atraso de microsegundos
Math (Matemática)	min(x, y); Determina o menor entre dois números
	max(x, y); Determina o maior entre dois números
	abs(x); Determina o valor absoluto de um número
	map(valor, nível mínimo de entrada, nível máximo de entrada, nível mínimo de saída, nível máximo de saída); Re-mapeia um número de uma faixa de valores para outra
	pow (base, expoente); Calcula o valor de um número elevado a uma potência
	sq(x); Calcula o quadrado de um número
	sqrt(x); Calcula a raiz quadrada de um número
Serial (Comunicação Serial)	Serial.begin(velocidade); Define a velocidade da comunicação serial
	Serial.read(); Lê dados que estejam entrando pela porta serial
	Serial.print(dado); Envia dados pela porta serial
	Serial.println(dado); Envia dados pela porta serial e pula linha

Tabela 6 – Bibliotecas e funções de programação.

Fonte: *Language Reference*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016



Com os exemplos apresentados, podemos concluir que as funções facilitam a programação e tornam o programa mais compreensível. Diversas funções já estão implementadas no núcleo da IDE de programação. As funções para aplicações específicas são implementadas através de bibliotecas auxiliares.

Uma biblioteca é um trecho de software que fornece funcionalidade específica a um programa, como por exemplo a capacidade de escrever em um display de LCD ou de controlar a posição de um servo motor. O uso de uma biblioteca simplifica o desenvolvimento de aplicações, pois o código da biblioteca já está pronto, e só precisa ser incorporado ao programa em desenvolvimento para que suas funções possam ser acessadas e utilizadas pelo desenvolvedor. Assim, podemos estender o uso do *Arduino* incorporando bibliotecas específicas durante o desenvolvimento de um sketch. A seguir listamos algumas dessas bibliotecas e suas funcionalidades.

EEPROM	Para reprogramar a memória de armazenamento permanente.
Ethernet	Para se conectar a uma rede Ethernet usando o <i>Arduino Ethernet Shield</i> .
LiquidCrystal	Para controlar telas de cristal líquido (LCDs).
Servo	Para controlar Servo motores.
SPI	Para se comunicar com dispositivos que utilizam a <i>Serial Peripheral Interface (SPI)</i> .
SoftwareSerial	Para a comunicação serial em qualquer um dos pinos digitais.
Stepper	Para controlar motores de passo.
Wire	( <i>Two Wire Interface TWI/I2C</i> ) - para enviar e receber dados através de uma rede de dispositivos ou sensores.

Tabela 7 – Bibliotecas específicas.

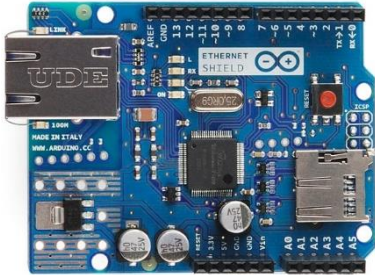
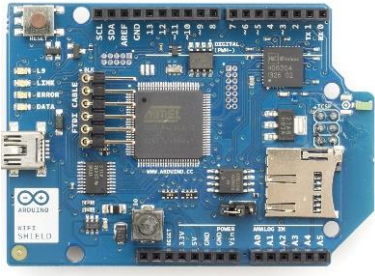

Fonte: *Libraries*, em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

De maneira análoga às bibliotecas, que são customizações aplicadas ao software, as *Shields* são placas com implementação de um hardware específico. A principal característica de uma *shield* é a de ser projetada para encaixar de maneira precisa nos conectores da placa *Arduino*. Estas placas podem conter displays de LCD, sensores, módulos de comunicação ou relês, por exemplo. A capacidade de expansão possibilita uma infinidade de aplicações de maneira simples e rápida. As desvantagens são a arquitetura de hardware normalmente fechada e o custo, que muitas vezes inibe a sua utilização nos projetos.



Figura 5 – Exemplo de conexão de *shields* em uma placa *Arduino UNO*  
 Fonte: *Hardware & Related Initiatives* em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

Alguns exemplos de Shields:

<p><i>Arduino Ethernet Shield</i></p> <p>Esta <i>Shield</i> fornece acesso à rede (IP) nos protocolos TCP ou UDP e possui um slot para cartão microSD que pode ser usado para armazenar arquivos que vão servir na rede, sendo facilmente utilizada usando a biblioteca <i>Ethernet Library</i> e <i>SD Library</i>.</p>	
<p><i>Arduino WIFI Shield</i></p> <p>Esta <i>shield</i> conecta o seu <i>Arduino</i> às redes wireless mais comuns do mercado (padrão 802.11 b/g), utilizando criptografia WEP, WPA ou WPA2. Também é possível utilizar o slot para cartão microSD para armazenar dados de sensores ou mesmo da internet.</p>	
<p><i>Arduino GSM Shield</i></p> <p>Esta é uma <i>shield</i> GPRS (<i>General Packet Radio Service</i>) para <i>Arduino</i> baseado no módulo wireless SIM900 Quad-Band GSM/GPRS. O módulo é capaz de enviar serviços GSM/GPRS em Quad-Band (850/900/1800/1900MHz) como SMS, Voz (ligações), Dados via Internet e Fax.</p>	


<p><i>Adafruit B&amp;W LCD Shield with keypad</i></p> <p>É uma <i>shield</i> que implementa um display LCD 16 caracteres x 2 linhas e um conjunto de 5 chaves do tipo <i>push botton</i>.</p>	
---	--

Tabela 8 – Alguns exemplos de *shields* que podem ser conectadas em placas do projeto *Arduino*  
 Fonte: *Hardware & Related Initiatives* em [www.arduino.cc](http://www.arduino.cc), consultado em 29/07/2016

Além das *shields*, podemos implementar aplicações com o uso direto de sensores analógicos ou digitais. Os sensores são dispositivos eletroeletrônicos que transformam em variações elétricas as variações de alguma grandeza física. Há diversos sensores que podem ser utilizados para a medição ou detecção de diversas grandezas, como por exemplo:

 <p>Sensor ultrassônico Medição de distâncias</p>	 <p>Sensor infravermelho Medição de distâncias, detecção de proximidade e de passagem em pequenas distâncias</p>
 <p>Sensor infravermelho Medição de distâncias, detecção de proximidade e de passagem em distâncias até 1,5 m</p>	 <p>Sensor de efeito Hall Medição de corrente elétrica 0 a 5A</p>
 <p>Sensor BMP085 Medição de pressão atmosférica e temperatura</p>	 <p>Sensor DHT11/DHT22 Medição de umidade e temperatura do ar</p>

Tabela 9 – Alguns sensores para *Arduino*

Para a construção de objetos educacionais que permitam a obtenção de experimentos quantitativos a partir do *Arduino* é necessária a utilização de uma interface para a apresentação de forma numérica e/ou gráfica dos dados adquiridos ao aluno.

Comercialmente, há diversos softwares proprietários para esta finalidade, porém com elevado custo de licenciamento e muitas vezes se tornando inviável para as escolas públicas. Para manter a ideia do baixo custo, realizamos uma pesquisa dos softwares e linguagens de programação livres que atendessem a este objetivo.

Dentre as linguagens disponíveis, o *Processing* foi a escolhida por apresentar grande similaridade com a Linguagem utilizada para a programação do *Arduino*. Além de Ambientes de Desenvolvimento praticamente idênticos, a semelhança entre as duas linguagens perpassa a sua origem compartilhada, a linguagem *wiring*, o uso de funções e bibliotecas e a mesma estrutura de comandos, se distanciando apenas nas funções e bibliotecas específicas. Essa similaridade em muito facilita o uso do *Arduino* em conjunto com o *Processing*.

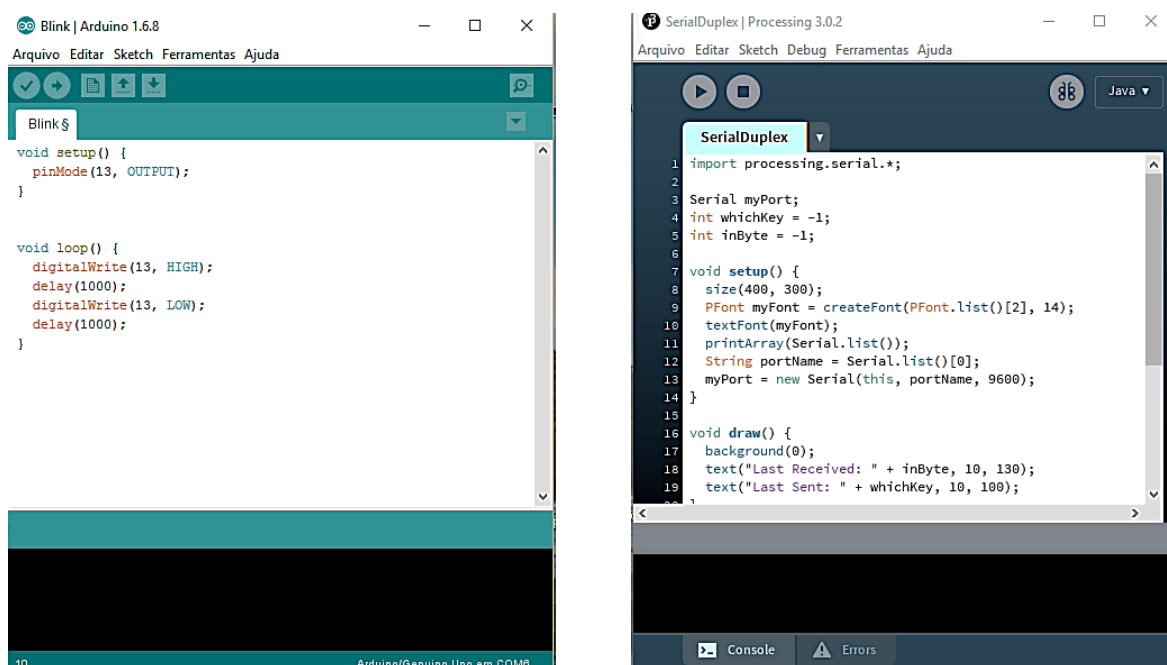


Figura 6 – Semelhança entre as IDEs do *Arduino* e do *Processing*

Inicialmente desenvolvido para servir como um caderno de desenho e como um software para ensinar fundamentos de programação dentro de um contexto visual, o *Processing* evoluiu para uma ferramenta de desenvolvimento profissional, tornando-se uma alternativa às ferramentas de software proprietário com licenças restritivas e dispendiosas, sendo acessível para as escolas públicas e para os alunos individuais.

Criado em 2001 por Ben Fry e Casey Reas, enquanto ambos eram estudantes de graduação no MIT Media Lab e participantes do grupo de pesquisa em computação de John Maeda, o *Processing* é na verdade um aplicativo da linguagem Java, mas com algumas

facilidades destinadas a produção de objetos gráficos, gráficos 2D e 3D, animações, interações etc. Sua versão 1.0 beta foi lançada em 20 de abril de 2005, sendo seguida por um desenvolvimento incremental durante alguns anos. Apresentamos a seguir as principais versões do *Processing* com suas respectivas datas de lançamento:

3.0.2	13/02/2016	2.1	27/10/2014
3.0.1	23/10/2015	2.0	3/06/2013
3.0	30/09/2015	1.0.9	20/10/2009
2.2	12/05/2014	1.0	24/11/2008

Tabela 10 – Principais versões do software *Processing*  
 Fonte: *The list of revisions* em [www.processing.org](http://www.processing.org), consultado em 29/07/2016

Versões do *Processing* que utilizam para seu processamento as linguagens *Python* (*Processing.py*), *Ruby*, *JavaScript* (*p5.js*) e *Scala* também estão em desenvolvimento. Atualmente o *Processing* é desenvolvido em Boston pelo Fathom Information Design, em Los Angeles no Arts Software Studio da UCLA e em Nova York pelo ITP da NYU. O download gratuito da última versão do *Processing*, exemplos de códigos e referências da linguagem são disponibilizados gratuitamente em:

<https://processing.org>

Em nosso contexto, o *Processing* é o responsável pela interface, quase sempre, em tempo real, e é capaz de armazenar esses mesmos dados em arquivo no formato texto, para análise posterior, em softwares livres do tipo planilha eletrônica como o *Libreoffice Calc* ou de análise como o *SCIDavis*.

O canal utilizado para a comunicação entre o *Arduino* e o *Processing* é a conexão serial, em nossa proposta, estabelecida através de uma interface USB entre o computador e a placa *Arduino*.

No *Arduino* a comunicação serial é estabelecida com a biblioteca serial que já está inclusa em sua IDE. A interface necessita de uma configuração prévia que estabelece a velocidade de transmissão em bits por segundo (baud).

*Serial.begin(9600);*

Para a comunicação com o *Processing*, podemos utilizar as velocidades de 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600, 74880, 115200, 230400 e

250000 bauds. A escolha de maiores velocidades de transmissão garante, por consequência, uma interação mais rápida com o *Processing* propiciando uma melhor apresentação dos resultados. Nos projetos em que a aquisição de dados é realizada em função do tempo, é necessária a utilização da maior velocidade disponível.

*Serial.begin(250000);*

Para a transmissão e a recepção de dados por parte do *Arduino* são utilizadas as funções:

<i>Serial.read( );</i>	Lê dados seriais de entrada.
<i>Serial.print( );</i>	Imprime dados para a porta serial como texto ASCII <sup>18</sup> legível. Este comando pode assumir muitas formas. Os números são impressos usando um caractere ASCII para cada dígito. Números decimais são também impressos como dígitos ASCII, com duas casas decimais. Bytes são enviados como um único caractere. Caracteres e strings <sup>19</sup> são enviadas sem nenhuma alteração.
<i>Serial.println( );</i>	Imprime dados para a porta serial como texto legível ASCII seguido por um caractere de retorno de carro (ASCII 13 ou '\r') e um caractere de nova linha (ASCII 10 ou '\n'). Este comando assume as mesmas formas como <i>Serial.print( );</i>
<i>Serial.available( );</i>	Obtém o número de bytes disponíveis no buffer <sup>20</sup>
<i>Serial.parseInt( );</i>	Retorna o primeiro número inteiro válido, do tipo longo, presente no buffer serial. Caracteres que não são números inteiros são ignorados.
<i>Serial.write( );</i>	Grava dados binários para a porta serial. Estes dados são enviados como um byte ou uma série de bytes;

Tabela 11 – Algumas funções da biblioteca serial do *Arduino*  
 Fonte: *Serial library* em [www.processing.org](http://www.processing.org), consultado em 29/07/2016

As funções de comunicação serial podem ser testadas em sua própria IDE, através da funcionalidade *Serial Monitor*, que disponibiliza uma interface gráfica para a visualização dos dados transmitidos/recebidos via interface serial.

<sup>18</sup> (American Standard Code for Information Interchange - Código Padrão Norte-americano para Intercâmbio de Informações) Código numérico binário usado para representar os caracteres, compreendido por equipamentos e softwares. Foi desenvolvido pelo American National Standards Institute a partir da proposta de *Robert William Bemer* e sua equipe.

<sup>19</sup> Sequência de caracteres, geralmente utilizada para representar palavras ou frases.

<sup>20</sup> Memória física utilizada para armazenar temporariamente dados, no caso do buffer serial, ele armazena os dados enquanto aguardam para serem transmitidos serialmente, bit a bit, pela interface serial.

Código:

<pre>int valor = 0;  void setup() {   Serial.begin(9600);   while (!Serial){} }  void loop() {   if (Serial.available() &gt; 0) {</pre>	<pre>    valor = Serial.parseInt();   }   Serial.print("valor: ");   Serial.println(valor);   valor = valor+1;   delay(2000); }</pre>
---	---

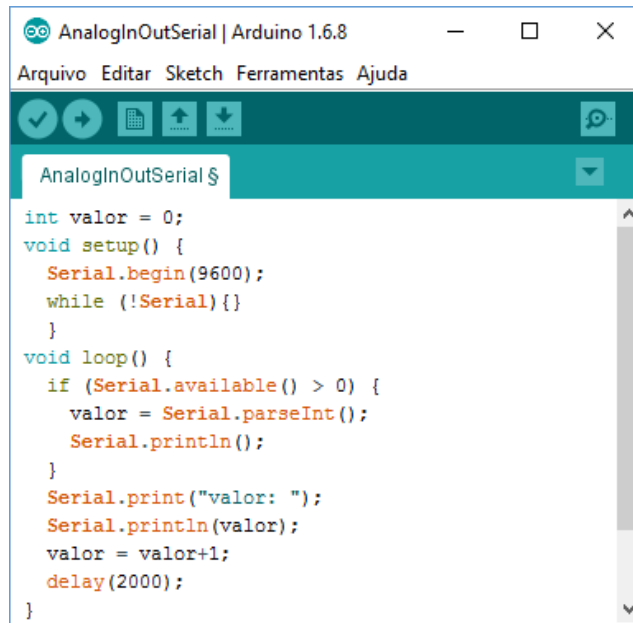


Figura 7 – Tela da IDE *Arduino* com o código

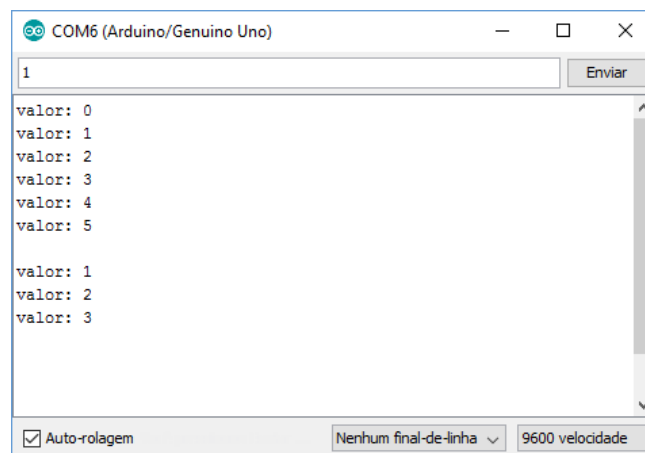


Figura 8 – Tela do *Serial Monitor*

No *Processing*, a comunicação serial é estabelecida com o uso da biblioteca serial que deve ser importada no início do código com a instrução *import* e com a criação de uma instância do tipo *Serial*.

```
import processing.serial.*
Serial portaSerial;
```

Assim como no *Arduino*, a interface necessita de uma configuração prévia. No *Processing*, é necessário definir uma nova classe serial vinculada a instância previamente criada e definir a velocidade de transmissão em bits por segundo e a porta na qual o software irá realizar a leitura dos dados.

```
portaSerial = new Serial(this, Serial.list( ) [0], 250000);
```

Com o intuito de visualizar as portas seriais ativas no computador, podemos incluir a função: `printArray(Serial.list( ))`; que apresentará estes dados no terminal da IDE do *Processing*. A porta a ser utilizada será a mesma porta utilizada para a gravação do *Arduino*. Apresentamos algumas das funções presentes na biblioteca serial necessárias para a comunicação com o *Arduino*:

<code>portaSerial.read( );</code>	Retorna um número entre 0 e 255 para o próximo byte que está esperando no buffer. Retorna -1 se não houver nenhum byte, embora isso deve ser evitado com o uso da função <code>available( )</code> para verificar se os dados estão disponíveis na porta especificada.
<code>portaSerial.readStringUntil(lf);</code>	Combinação das funções <code>readBytesUntil( )</code> e <code>readString( )</code> . Que retorna a sequência de caracteres (string) contida na porta especificada antes do byte especificado. Retorna NULL se não houver dados que atendam a condição.
<code>portaSerial.print( );</code>	Escreve bytes, caracteres, inteiros, arrays <sup>21</sup> e strings na porta serial especificada.
<code>portaSerial.println( );</code>	Escreve bytes, caracteres, inteiros, arrays e strings na porta serial especificada, seguido por um caractere de retorno de carro (ASCII 13 ou '\ r') e um caractere de nova linha (ASCII 10 ou '\ n').
<code>portaSerial.available( );</code>	Retorna o número de bytes disponíveis na porta serial especificada, neste caso, <code>portaSerial</code>
<code>Serial.list( )</code>	Obtém uma lista de todas as portas seriais disponíveis.

Tabela 12 – Algumas funções da biblioteca Serial do *Processing*  
 Fonte: *Serial library* em [www.processing.org](http://www.processing.org), consultado em 29/07/2016

<sup>21</sup> Estrutura de dados que armazena uma coleção de elementos de tal forma que cada um dos elementos possa ser identificado por um índice. Também são chamados de vetores.



As funções de comunicação serial do *Processing* também podem ser testadas em sua própria IDE, porém é necessário primeiro programar o *Arduino* e depois o *Processing*. Os códigos são apresentados a seguir:

Código *Arduino*:

```
int valor = 0;
void setup() {
  Serial.begin(9600);
  while (!Serial){ }
}
void loop() {
  if (Serial.available() > 0) {
    valor = Serial.parseInt();
  }
  Serial.println(valor);
  valor = valor+1;
  delay(2000);
}
```

Código *Processing*:

```
import processing.serial.*;
String serialString;
Serial portaSerial;
int lf = 10;
void setup()
{
  printArray(Serial.list());
  portaSerial = new Serial(this, Serial.list()[0], 9600);
  portaSerial.clear();
  size(10, 10);
}

void draw() {
  while (portaSerial.available() > 0) {
    serialString = portaSerial.readStringUntil(lf);
    if (serialString != null) {
      print("valor = ");
      println(serialString);
    }
  }
}
```

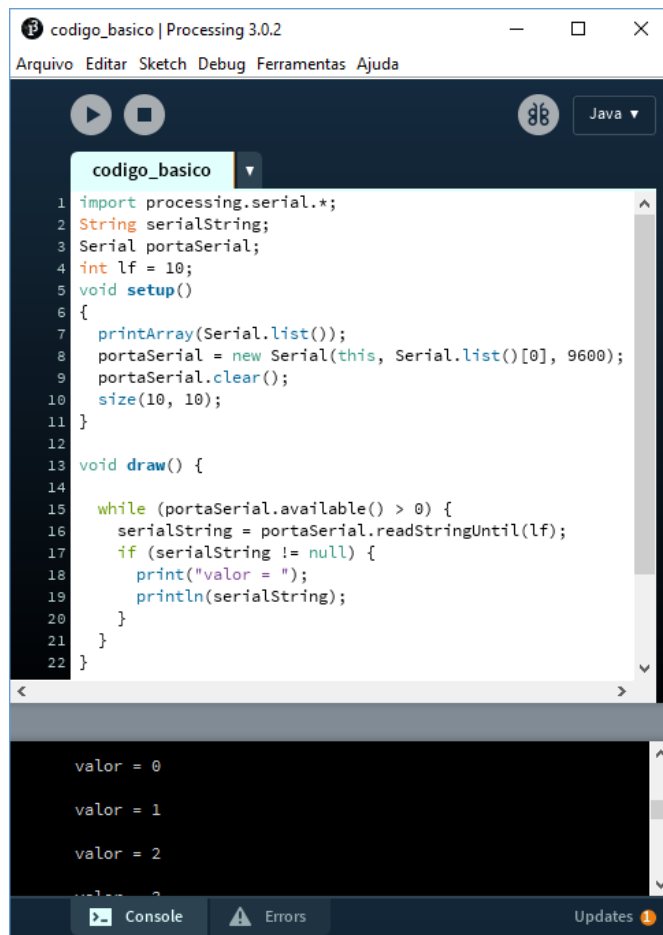


Figura 9 – Tela da IDE *Processing* com o Serial Monitor

Com a comunicação serial estabelecida entre o *Arduino* e o *Processing*, é possível construir circuitos para medir, através de sensores, as variáveis da natureza, com o *Arduino* realizar a aquisição e o envio desses valores para o *Processing* que, com a programação adequada poderá apresentar essas informações de forma gráfica e em tempo real na tela do computador e armazenar os dados em arquivos para posterior análise. A partir dessa estrutura, é possível criar os objetos educacionais de baixo custo, propostos neste capítulo a fim de contribuir para o desenvolvimento de uma metodologia alternativa para o Ensino de Física em nossas escolas.

Nos próximos três capítulos, serão abordados, individualmente, três sugestões didáticas referentes aos experimentos físicos propostos por essa dissertação: Oscilador harmônico amortecido, a Estação meteorológica e a Analogia funcional do acelerador de partículas.

### **O Arduino e o Processing no Ensino de Física: Materiais e bibliografia existentes**

A revisão da literatura está centrada na tentativa de explorar melhor o computador em atividades didáticas experimentais que vão desde o emprego das próprias interfaces do computador para a aquisição de dados até o uso do *Arduino* e do *Processing*, para a coleta, aquisição, processamento e apresentação dos dados, uma vez que uma grande quantidade de artigos voltados para o assunto tem sido publicada.

A começar com artigo intitulado *Por que e como introduzir a aquisição automática de dados no laboratório de Física?*, publicado na revista *Física na Escola* em 2001, os autores, Haag, Araujo e Veit, afirmam que a revolução digital tem impacto semelhante ao causado pela revolução industrial, ocorrida na segunda metade do século XVIII, sobre os hábitos e os costumes humanos. Para eles, o computador ocupa um papel fundamental em praticamente todas as áreas do conhecimento humano.

Como os autores apontam, nas ciências em geral e na Física em particular, o computador é muito utilizado para a criação de modelos científicos altamente complexos, entretanto, na escola, ele é usado prioritariamente na produção de textos, apresentações, páginas na *web* e como fonte de consulta. Essa situação poderia ser mudada, por exemplo, com o sensoriamento, medição e aquisição automática de dados no laboratório de Física, apresentado no artigo.

A professora V. M. Kenski, em seu livro *Tecnologias e ensino presencial e à distância*, publicado em 2003, também analisa as alterações na atuação docente decorrentes do uso mais intenso das novas tecnologias digitais de comunicação e informação. Contudo, segundo a autora, muitos professores implementam os recursos digitais em suas aulas, mas não mudam o seu jeito de ensinar. Usam os ambientes virtuais apenas para introduzir textos ou livros digitalizados e continuam a manter uma aula baseada na exposição oral e centrada em si.

Em parte, tal situação é causada pela forma com que o assunto é abordado nos cursos e treinamentos voltados para professores que “baseiam-se na premissa que basta apenas ensinar os professores a ‘botar a mão na massa’, a ‘mexer com o computador’, basta-lhes aprender as linguagens e técnicas para a utilização dessas máquinas, para fazer a transformação no ensino” (KENSKI, 2003, p.132).

Para uma mudança efetiva desse contexto, é necessário facilitar o acesso a textos, a vídeos, a experimentos e a simulações, na internet, proporcionando aos professores e alunos o cruzamento de informações e a comparações em um tempo mínimo e a baixo custo. Uma vez que esse tipo de objeto educacional não precisaria ser lido, reproduzido ou implementados necessariamente, da forma com a qual é, primariamente, apresentado, como acontece com a maioria dos materiais didáticos tradicionais, que são modelos fechados. Isto permite ao professor reduzi-los, ampliá-los e transformá-los de acordo com as características e necessidades de suas turmas. Deste modo, o acesso e a utilização das tecnologias no ambiente educacional pode, de fato, alterar as práticas pedagógicas e induzindo profundas modificações na organização do currículo escolar.

Porém, o trabalho com esse tipo de material requer uma mudança na visão e na postura que o professor tem diante da educação. Para isso, além do desejo do professor em melhorar suas competências profissionais e metodologias, é absolutamente necessário que as instituições de ensino se reestruturem e incentivem oferecendo condições permanentes de aperfeiçoamento e atualização para todos os seus funcionários.

M. A. Cavalcante, A. Bonizzia e L. C. P. Gomes, no artigo intitulado *O ensino e aprendizagem de física no Século XXI: sistemas de aquisição de dados nas escolas brasileiras, uma possibilidade real*, publicado na *Revista Brasileira de Ensino de Física* em 2009, apresentam propostas de experimentos que utilizam a entrada de microfone do computador como meio para aquisição de dados.

As leituras dos dados podem ser feitas através de versões shareware e/ou freeware de softwares de análise disponíveis na internet. De modo geral, os arranjos experimentais descritos nestes trabalhos são de fácil construção e com equipamentos de baixo custo. Porém estes experimentos apresentam as limitações eletro-eletrônicas da entrada de microfone do computador, limitando seu uso.

De acordo com um grupo de pesquisadores da Universidade Tecnológica do Paraná, em artigo intitulado *Tecnologias livres e ensino de Física: uma experiência na UTFPR*, publicado nas *Atas do XVIII Simpósio Nacional de Ensino de Física (SNEF, 2009)*, existe um espaço e, também a necessidade, para a pesquisa de experimentos voltados para o ensino de Física “que incluem o software livre no desenvolvimento dos aplicativos, o hardware de código aberto no circuito associado, e o acesso aberto, na documentação”. (BEZERRA JR. et al., 2009, p. 02).

Segundo E. A. Veit, em *O laboratório didático de Física e os Parâmetros Curriculares Nacionais para o ensino médio*, essa perspectiva oferece autonomia

pedagógica, visto que professores e estudantes também podem participar do processo de construção das ferramentas, auxiliando no processo de desmistificação da aquisição automática de dados como uma caixa preta, visão compartilhada e citada por tantos autores (2003).

Com objetivo de difundir o uso do Arduino para fins educacionais e particularmente no estudo de ondas sonoras estacionárias em tubos, contribuindo com a melhora na abordagem deste conteúdo no ensino e aprendizagem de Física, Cavalcante, no artigo intitulado *Novas Tecnologias no Estudo de Ondas Sonoras*, publicado em 2013, defende o uso dessa plataforma microcontrolada aliada ao emprego do processing construção de um Tubo de Kundt, comprovando como o uso dessas tecnologias favorecem o ensino de física.

Grupo de Pesquisa em Ensino de Física da Pontifícia Universidade Católica de São Paulo formado por M. A. Cavalcante, C. R. C. Tavoraro e E. Molisani, apresentou no artigo *Física com Arduino para iniciantes*, diferentes modos de operar o *Arduino* como uma interface alternativa na aquisição e automação de dados em atividades experimentais de física via porta USB do computador. Para cada etapa, os autores apresentaram propostas didáticas de utilização, todos os códigos fontes necessários para a interação com o *Arduino* e os links para acesso a tutoriais que possibilitam a reprodução deste e outros experimentos.

A bibliografia analisada indica que ainda existe um vasto campo de pesquisa e aplicação do *Arduino* e do *Processing* para o Ensino de Física nas escolas de Ensino Médio, indicando um caminho de objetos educacionais de baixo custo e tecnologia livre, o que incentivou a proposta da pesquisa e o desenvolvimento do material instrucional nesta dissertação de mestrado.

**Oscilador Harmônico Amortecido com o uso do *Arduino* e do *Processing***

Os fenômenos oscilatórios são de extremo interesse em diversos campos do conhecimento, tais como a Física, a Matemática, a Educação Física e a Biologia. Devido a essa característica transdisciplinar, esse é um assunto bastante abordado no Ensino Médio e no Ensino Superior.

Para que o aluno perceba a importância e, ao mesmo tempo, tenha maior interesse no assunto, cabe destacar, que no seu cotidiano, há várias situações nas quais um corpo realiza um movimento de ida e volta em torno de uma posição central. Como afirmam os físicos David Halliday e Robert Resnick:

Nosso mundo está repleto de oscilações, nas quais os objetos se movem repetidamente de um lado para outro. Muitas são simplesmente curiosas ou desagradáveis, mas outras podem ser economicamente importantes ou perigosas. Eis alguns exemplos: quando um taco rebate uma bola de beisebol, o taco pode sofrer uma oscilação suficiente para machucar a mão do bateador ou mesmo se partir em dois. Quando o vento fustiga uma linha de transmissão de energia elétrica, a linha às vezes oscila (“galopa”, no jargão dos engenheiros elétricos) com tanta intensidade que pode se romper, interrompendo o fornecimento de energia elétrica a toda uma região. Nos aviões, a turbulência do ar que passa pelas asas faz com que oscilem, causando fadiga no metal que pode fazer com que as asas se quebrem. Quando um trem faz uma curva, as rodas oscilam horizontalmente quando são forçadas a mudar de direção, produzindo um som peculiar. (2012, p.88)

É importante salientar que, em condições ideais, esse movimento se perpetua por todo o tempo, gerando uma onda de oscilação como a apresentada a seguir:

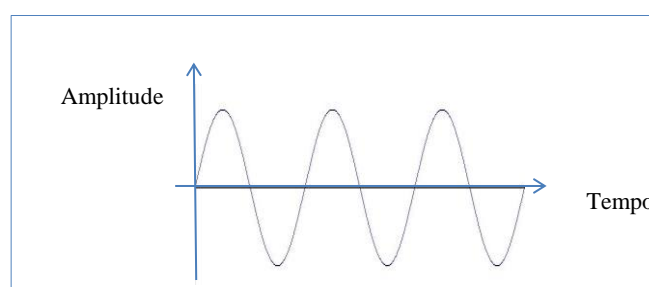


Figura 10 – Representação do movimento oscilatório

O movimento harmônico simples é um movimento periódico no tempo, ou seja, um movimento que se repete em intervalos iguais com frequência, período e amplitude constantes. A frequência é uma grandeza física que indica o número de ocorrências de um evento (ciclos, voltas, oscilações, etc.) em um determinado intervalo de tempo. Alternativamente, podemos medir o tempo decorrido para uma única oscilação.

Esse tempo, em particular, recebe o nome de período (T). Desse modo, a frequência (f) é o inverso do período. A amplitude de um movimento oscilatório é uma medida escalar da magnitude da oscilação de uma onda.

Na natureza, devido a interações com o ambiente, como a resistência do ar, o movimento harmônico passa a apresentar uma força não conservativa que atua em sentido contrário, ocasionando a redução da amplitude de oscilação em função do tempo. Esse movimento é denominado movimento harmônico amortecido.

As oscilações existentes no mundo real são normalmente amortecidas, ou seja, desaparecem gradualmente, transformando a energia mecânica em energia térmica, em virtude do atrito. Embora não seja possível eliminar completamente o atrito em sistemas oscilantes, podemos alimentar o sistema com uma fonte de energia externa para compensar a energia que é perdida pelo atrito. (HALLIDAY, RESNICK, 1993, p.21)

Como dito anteriormente, o tópico associado às oscilações, em particular, é um assunto que pode ser abordado em várias disciplinas:

Na Matemática, o estudo da função senoidal é empregado para descrever o sinal gerado por um corpo que oscila em movimento harmônico simples. Os cálculos matemáticos possibilitam a previsão do comportamento que esse corpo terá em sua trajetória.

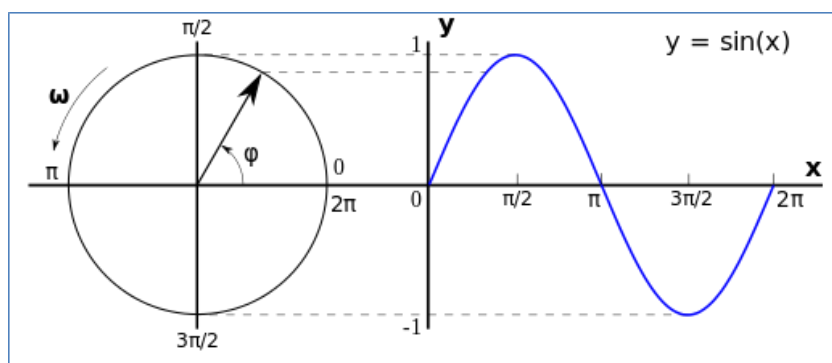


Figura 11 – Geração da função senoidal

Na Biologia, como afirma o professor Hamilton Varela, em entrevista ao Instituto de Estudos Avançados da Universidade de São Paulo (IEA-USP), “Os regimes oscilatórios são uma característica essencial dos seres vivos que sobreviveriam às instabilidades ambientais graças à flexibilidade proporcionada pelas oscilações presentes em seus processos vitais”. Como exemplo disto é possível citar desde o movimento de algumas espécies de algas como a *Oscillatoria*, que realizam movimentos oscilatórios, facilmente observáveis ao microscópio e com velocidade de 2 a 11  $\mu\text{m/s}$ , até o comportamento reprodutivo de várias espécies animais, como no caso dos vaga-lumes, mencionado pelo matemático Ian Stewart em seu livro, *Mania de Matemática 2*:

Os vaga-lumes utilizam uma substância química especial, capaz de emitir luz, para criar um clarão. Possuem um bom estoque da substância, mas sua liberação ocorre em pequenos surtos, segundo um ciclo repetitivo de "prontidão". Com efeito, é como se o inseto começasse a contar regularmente a partir do zero logo após piscar, só piscando outra vez ao chegar a cem. O estado de prontidão – o número ao qual a contagem chegou, por assim dizer – é a "fase" do ciclo. (2006, p.139)

Na Educação Física, podemos considerar alguns esportes radicais, como o *Bungee Jumping* e o *Canyon Swinging*, que aplicam, na prática, os conceitos de oscilador e de movimento oscilatório. O *Bungee Jumping* consiste em saltar de uma grande altura amarrado aos tornozelos ou cintura a uma corda elástica, enquanto o *Canyon Swinging* consiste em um "balanço" pendurado num cabo estendido entre as duas paredes de um cânion.

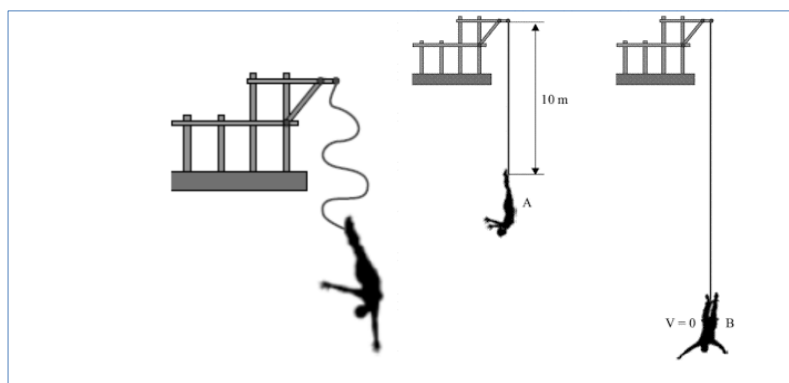


Figura 12 – Modelo do *Bungee Jumping*  
 Fonte: Prova da UNESP 1ª fase, 2014

Nos livros didáticos de Física<sup>22</sup>, o movimento oscilatório é demonstrado, geralmente, a partir de três modelos: o oscilador massa-mola, o pêndulo simples e a lâmina vibrante. Mas a aplicação prática desses experimentos ou não é simples ou é apenas qualitativa.

O experimento do oscilador massa-mola, abordado nos livros didáticos, normalmente se apresenta na forma horizontal e seu funcionamento baseia-se no movimento de um corpo sob a ação de uma mola, de massa desprezível, sob uma superfície sem atrito. Esta configuração necessita do uso de um trilho de ar para que o atrito com a superfície seja minimizado e não haja o comprometimento do experimento, dependendo, assim, de um laboratório específico e minimamente equipado para a sua realização.

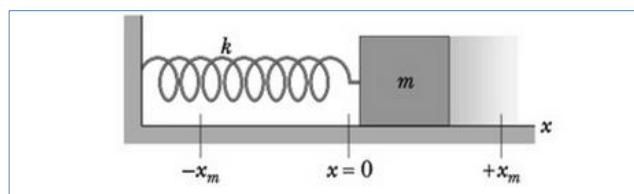


Figura 13 – Oscilador massa-mola na configuração horizontal  
 Fonte: HALLIDAY, 2012 p.92

<sup>22</sup> Vide: MAXIMO, A.; ALVARENGA, B. **Física**. Rio de Janeiro: Scipione, 2002, p. 531-535 e YAMAMOTO, K.; FUKU, L. F. **Física para o Ensino Médio**. São Paulo: Saraiva, 2013, p. 250-256.



Esse mesmo oscilador pode ser montado verticalmente, o que minimiza o aparato essencial para sua realização, porém depende do acionamento manual de um cronômetro e a atenção do aluno para marcar a passagem da massa por uma escala para a determinação dos pontos máximos da oscilação.

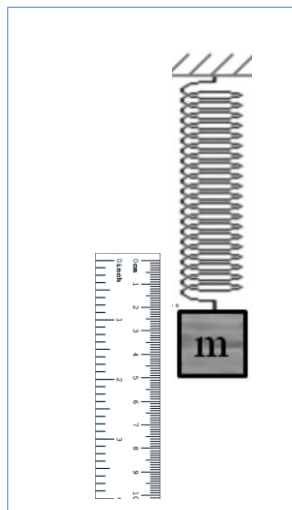


Figura 14 – Oscilador massa-mola na configuração vertical

O do pêndulo simples é um sistema constituído de uma partícula pendurada por um fio inelástico com massa desprezível. Mesmo tendo uma configuração que independe de laboratório, para que ele, de fato, funcione como um experimento quantitativo, necessita, assim como o experimento massa-mola vertical, que o aluno possua destreza para acionar um cronômetro e acuidade visual para medir os pontos máximos da oscilação na passagem do fio por uma escala fixa.

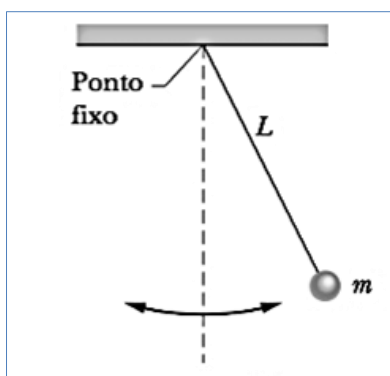


Figura 15 – Modelo do pêndulo simples  
Fonte: HALLIDAY, 2012 p.98

Já o da lâmina vibrante, por contar, apenas, com uma lâmina, normalmente, uma régua, presa em uma de suas extremidades, que, quando a outra extremidade é afastada de sua posição de equilíbrio e solta, ela executa um movimento oscilatório em torno da posição de repouso, indo do ponto A ao ponto B, sucessivamente, não oportuniza nenhum tipo de medição.

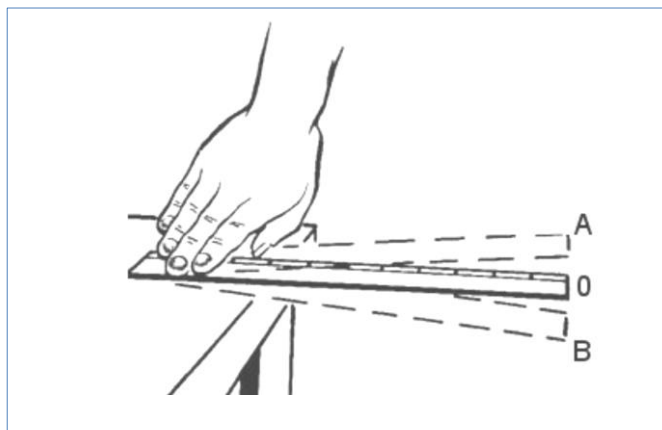


Figura 16 – Movimento oscilatório de uma régua com uma extremidade livre  
(Adaptada do original, disponibilizado em: <http://audioescola.blogspot.com.br>, acessado em 01/06/2016)

Este último, além de ser muito fácil e de baixo custo, acaba por ser um dos experimentos mais próximos do cotidiano dos alunos, que o reconhecem como uma brincadeira realizada, de maneira intuitiva, com as régua, em geral de plástico, nos cantos das carteiras estudantis nas salas de aula.

Esse experimento divertido, lúdico e acessível pode se tornar quantitativo com o uso do *Arduino* e do *Processing*, que possibilitam medições realizadas em uma escala temporal da posição instantânea da régua, apresentadas em um gráfico posição x tempo na tela do computador e armazenadas em arquivo para posterior análise.

Para tornar este experimento quantitativo, é primordial utilizar um sensor que monitore a posição da régua, sem interferir em seu movimento. Para este desenvolvimento utilizamos como referência o artigo *A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC*, publicado na *Revista Brasileira de Ensino de Física* de nº 33. Neste artigo, os autores relatam a montagem realizada:

Colamos numa das extremidades de uma régua de plástico um pequeno espelho plano. A outra extremidade da régua é mantida fixa por uma garra e sustentada na horizontal com auxílio de um pequeno pedestal. (...) Com uma pequena lanterna colocada na vertical, iluminamos o espelho de baixo para cima com um feixe de luz divergente. A luz refletida no espelho ilumina um resistor sensível a luz (LDR - *Light Dependent Resistor*) fixada ao lado da lanterna (...). Com o movimento da extremidade da régua alteramos a distância entre a fonte de luz e o LDR alterando a intensidade luminosa e por consequência a resistência do LDR. Temos assim um sensor de posição muito simples. (2011, p. 1702-2)

Contudo a replicação e o uso em sala de aula desse experimento mostraram que a variação da iluminação do ambiente altera o valor da resistência do LDR, gerando interferência aditiva nos valores lidos pelo *Arduino*. Deste modo, em salas pouco iluminadas ou escuras os dados obtidos no experimento são diferentes daqueles adquiridos em salas com a iluminação adequada. Nestas salas, o intervalo de medição fica diminuído, devido a tensão residual gerada pela iluminação do ambiente que se soma ao sinal, diminuindo a amplitude de medição. Além disso, na proposta no artigo referido anteriormente as medições são apresentadas em unidades arbitrárias (U.A.).

Para dirimir esta interferência e, assim, tornar o experimento adequado a qualquer ambiente, foi substituído o conjunto LDR/Lanterna pelo sensor TCRT5000 que possui uma construção compacta, unindo, em um mesmo dispositivo, a fonte de luz (LED infravermelho) e o detector (fototransistor<sup>23</sup> infravermelho). Esta alteração já havia sido proposta no artigo *Damped Harmonic oscillator with Arduino*, publicado no *Journal of Applied Mathematics and Physics*, v. 3 n. 6.

O TCRT5000, produzido pela *Vishay Telefunken*, é um sensor óptico por reflexão direta ou difusa com saída a transistor, que opera na faixa do infravermelho com comprimento de onda de 950 nm. Possui filtro contra a luz do dia e pode ser alimentado pela saída +5V do *Arduino*. Além disto, a velocidade de resposta<sup>24</sup> do fototransistor, que opera em microsegundos, é maior do que a do LDR, que responde em milissegundos.

A curva de resposta do LDR é linear para medições de luminosidade, porém, em medições de distância por reflexão direta, a linearidade da detecção pode ser comprometida. No caso do fototransistor a curva de resposta, disponibilizada pelo fabricante, para reflexão direta, obedece, a partir de uma dada distância, à uma função matemática.

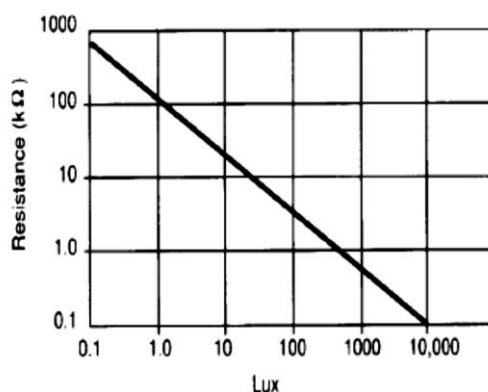


Figura 17 – Curva de resposta de um LDR do tipo CdS disponibilizada na folha de dados do fabricante  
Fonte: Folha de dados do LDR da SUNROM Technologies

<sup>23</sup> Transistor cuja junção coletor-base fica exposta a luz. A energia luminosa desloca elétrons para a banda de condução, reduzindo sua barreira de potencial.

<sup>24</sup> Trata-se da velocidade com que a medida fornecida pelo sensor alcança o valor real da variável física.

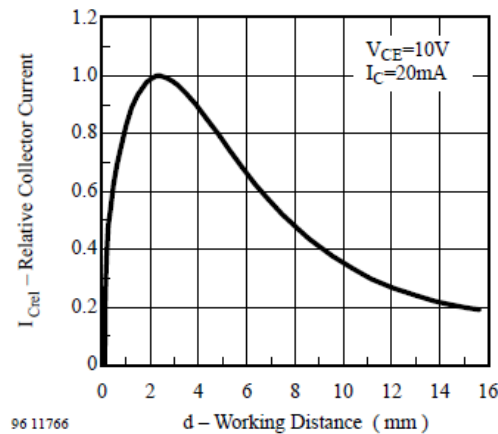


Figura 18 – Curva de resposta do sensor TCRT5000 disponibilizada na folha de dados do fabricante  
 Fonte: Folha de dados do LDR da SUNROM Technologies

Em algumas aplicações com os sensores ópticos os dados são apresentados em uma escala arbitrária. No artigo, os autores, com o uso do LDR, relatam uma

assimetria no amortecimento das oscilações da régua que deve ser creditada a não estrita linearidade da resposta do sensor de posição. Este fato é menos crítico na medida em que os deslocamentos são menores (...). Naturalmente, é perfeitamente possível a construção de uma curva de calibração para o sensor de posição que possa corrigir todas as alterações introduzidas pela não-linearidade da resposta, trabalho este que, por si só, agrega muitas questões novas a serem exploradas em sala de aula. (2011, p. 1702-3)

Para que o experimento, de fato, se torne matematicamente verificável e fisicamente significativo é importante que ele apresente dados de posição e tempo em unidades do Sistema Internacional. Para obter uma escala milimétrica de posição, é preciso realizar o procedimento de linearização do sensor. Cabe ressaltar que a resposta de um sensor óptico de medição por reflexão difusa depende da geometria, do acabamento superficial, da cor do objeto a ser detectado e do ângulo de incidência/reflexão.

Para simplificar a linearização, primeiramente, foi executado um arranjo experimental específico, excluindo a variável do ângulo de incidência/reflexão. Conforme apresentado a seguir:

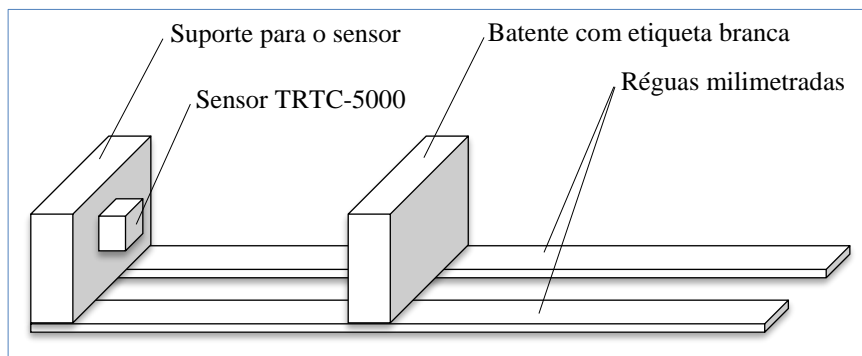


Figura 19 – Esquema simplificado da montagem do experimento

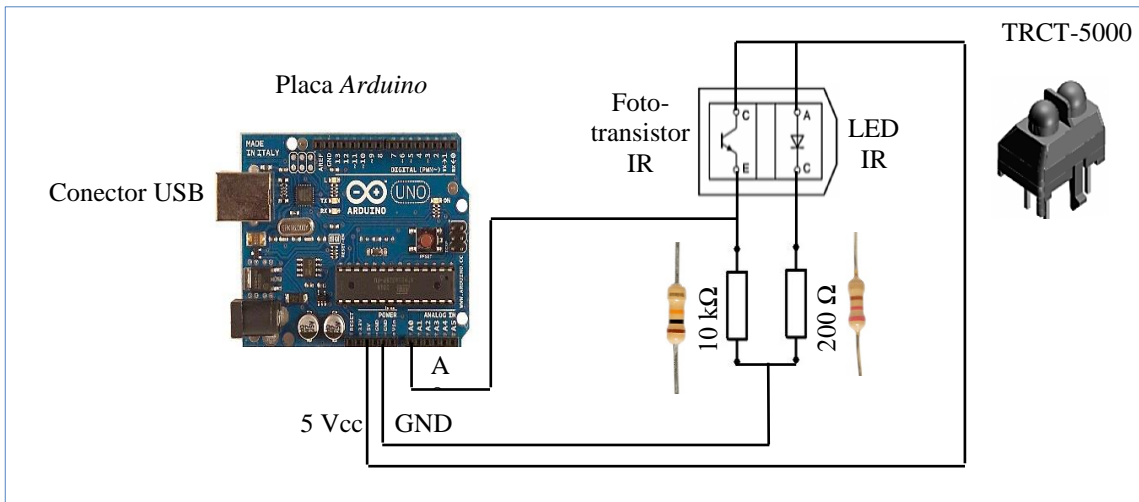


Figura 20 – Diagrama da montagem eletrônica do experimento

Cabe salientar que a linearização de sensores pode ser realizada utilizando-se a tecnologia analógica ou a tecnologia digital. No método analógico, é imprescindível a utilização de um componente eletrônico que apresente uma resposta inversa ao sensor a ser corrigido. Por exemplo, um circuito que possua resposta exponencial pode corrigir um sensor de resposta logarítmica.

Porém, por ser improvável a escolha de um circuito que possua saída perfeitamente simétrica ao sensor e que alcance uma resolução adequada de correção, seria inevitável a utilização de outros componentes para realizar os ajustes necessários, tornando o circuito eletrônico complexo e dispendioso, contrariando o objetivo deste trabalho.

Os métodos digitais são numerosos e podem ser implementados por software, realizando-se a linearização sem o uso de nenhum componente adicional ao circuito do experimento. O método utilizado neste trabalho será uma variação do método de linearização por tabela. Neste método, é fundamental o registro em uma tabela dos dados referentes a coleta do valor métrico de posição e o respectivo valor analógico medido pelo sensor, sendo que o número de dados constantes nesta tabela determinará, de maneira direta, a resolução da linearização do sensor.

Após o registro destes dados e com a utilização de um software de planilha eletrônica (*Libre Office Calc, SciDavis, etc.*) determina-se a função que representa a variação destes dados. Com o conhecimento desta função é possível implementá-la via software e assim linearizar o sensor.

```

int pinoAnalogico = 0;
int valorAnalogico,somatorio,media;
void setup(){Serial.begin(250000);}
void loop(){
somatorio = 0;
for (int i=0; i <= 10; i++){
valorAnalogico = analogRead(pinoAnalogico);
somatorio = somatorio + valorAnalogico;
delay(10);
}
media = somatorio/10;
Serial.println("valor Medido = ");
Serial.println("media ");
delay(500);
}

```

Este programa realiza 10 medições de posição, com o uso do sensor TRCT-5000, com um intervalo de 10 milissegundos entre cada uma das medições e apresenta o valor da média aritmética dessas medidas, via interface serial e aguarda 500 milissegundos para refazer de forma cíclica o mesmo processo. Para o processo de medição foi considerada a referência de distância do próprio sensor (sensor-objeto).

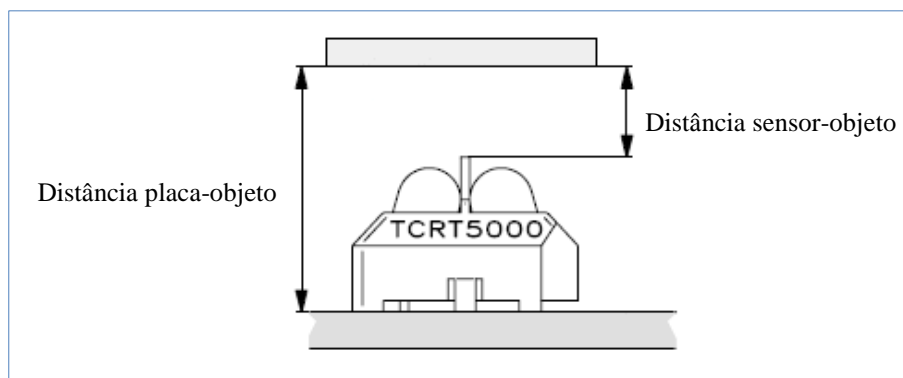


Figura 21 – Representação da referência de distância do sensor  
Fonte: Folha de dados do TCRT5000 da Vishay Semiconductors

Para a observação do comportamento de resposta do sensor, foi montado um arranjo linear, apresentado na figura 19, e foram realizadas medições no intervalo de 5 a 130 mm, totalizando 26 pontos de referência. A partir dos dados obtidos, pode-se concluir que o intervalo de distâncias útil para a medição de distâncias no experimento é de 15 a 80 mm. Os valores de distância menores de 15 mm apresentaram comportamento não linear, como esperado, pois os sensores de distância, quer ópticos ou ultrassônicos, apresentam uma distância mínima de operação. Para valores de distância maiores que 80 mm, a resolução

apresentada é inferior a necessária para uma equivalência milimétrica, pelo menos 5 níveis para cada intervalo de 5 mm.

Distância (mm)	Nível após digitalização	Distância (mm)	Nível após digitalização
5	993	70	78
10	986	75	70
15	837	80	64
20	553	85	58
25	399	90	54
30	299	95	50
35	231	100	46
40	185	105	43
45	152	110	41
50	129	115	39
55	111	120	37
60	98	125	36
65	86	130	34

Tabela 13 – Dados da resposta do sensor TRTC-5000 para reflexão difusa no intervalo 5-130 mm

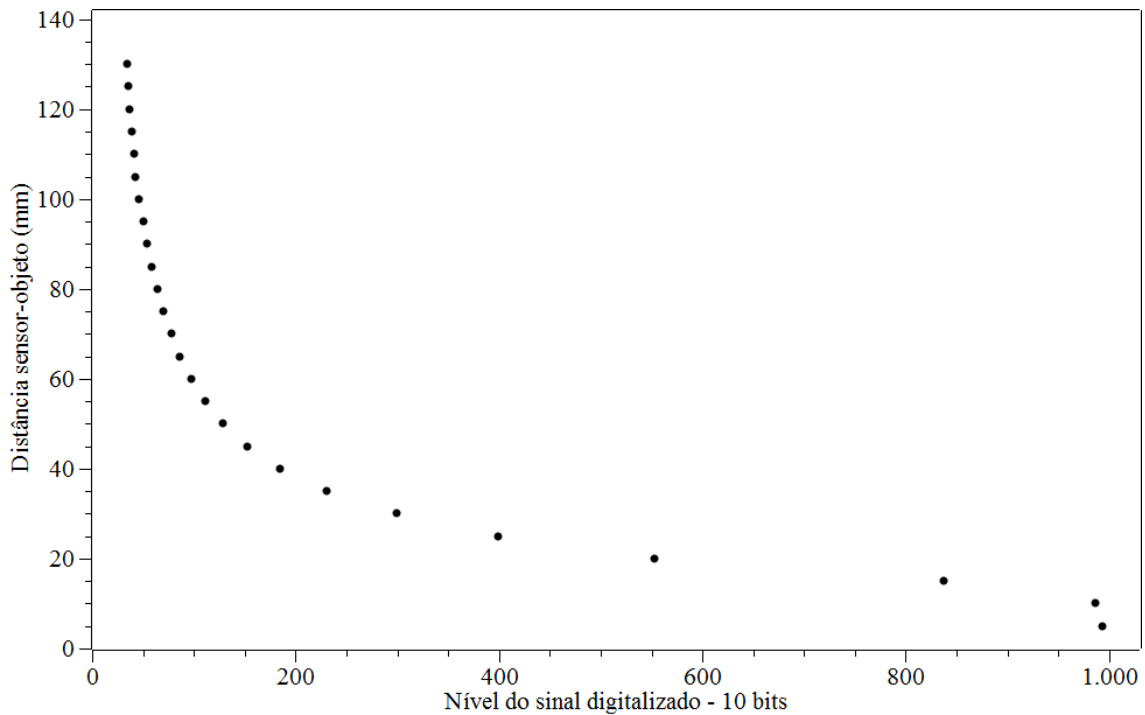


Gráfico 1 – Curva de resposta do sensor TRTC-5000 para reflexão difusa no intervalo 5-130 mm

No intervalo escolhido, entre 15 e 80 mm, foram realizados 3 ciclos de medição com um degrau entre distâncias de 5 mm e calculada a média aritmética para a construção do gráfico da função de resposta do sensor.

Distância (mm)	Nível após digitalização Medição 1	Nível após digitalização Medição 2	Nível após digitalização Medição 3	Nível após digitalização Média
15	837	822	836	832
20	553	554	560	556
25	399	397	402	399
30	299	293	297	296
35	231	228	231	230
40	185	184	184	184
45	152	152	152	152
50	129	130	129	129
55	111	111	112	111
60	98	98	98	98
65	86	87	87	87
70	78	78	78	78
75	70	71	71	71
80	64	64	64	64

Tabela 14 – Dados da resposta do sensor TRTC-5000 para reflexão difusa no intervalo 15-80 mm

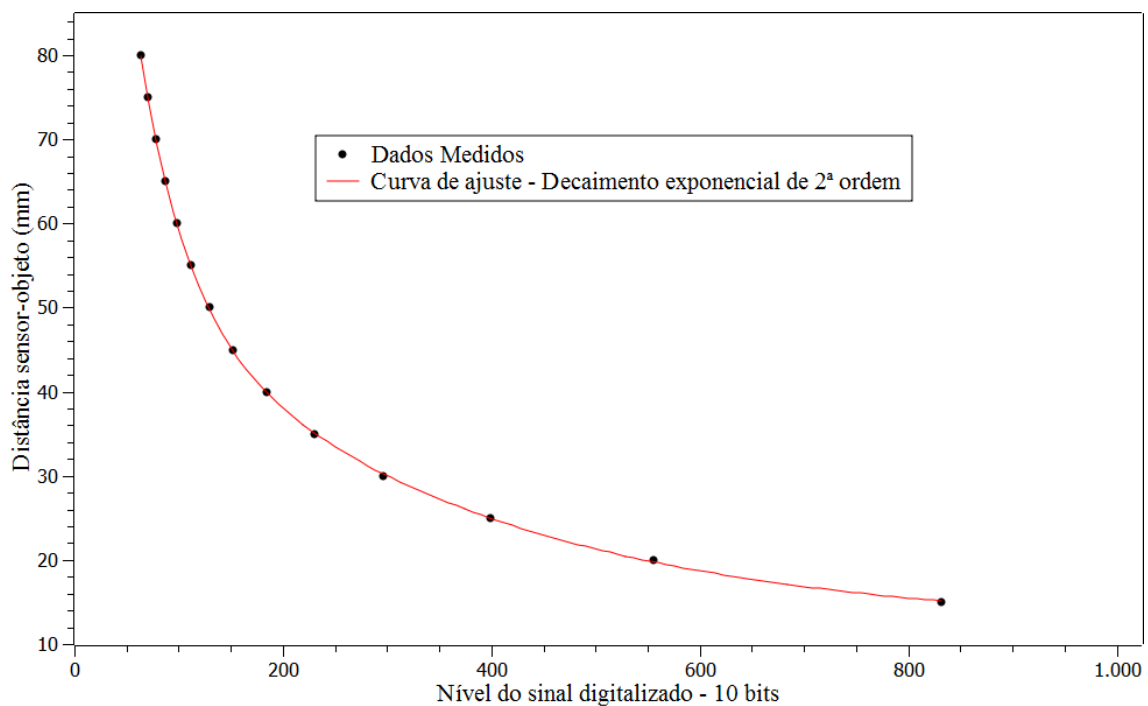


Gráfico 2 – Curva de resposta do sensor TRTC-5000 para reflexão difusa no intervalo 15-80 mm

A Função de linearização obtida do sensor TRTC-5000 é uma exponencial de decaimento de 2ª ordem:  $y = 11,847 + 47,2524 \times e^{-\frac{x}{311,232}} + 127,501 \times e^{-\frac{x}{43,903}}$ , sendo y a distância entre o sensor e a régua em mm, e x o valor de resposta digitalizada em 10 bits do sensor.



Com a definição da função de resposta do sensor, é preciso determinar o erro relativo para que seja possível a avaliação da qualidade do processo de linearização do sensor. Para alcançar este objetivo, foi aplicada a equação matemática a cada um dos valores obtidos na digitalização da resposta direta do sensor e comparado o resultado a equivalente medição de distância, obtendo-se o erro relativo a cada uma das distâncias e o erro relativo médio de 0,36%.

Distância sensor-régua (mm)	Nível após digitalização (Média)	Distância obtida após linearização	Erro Relativo
15	832	15,1	0,72%
20	556	19,8	1,18%
25	399	25,0	0,11%
30	296	30,3	0,84%
35	230	35,1	0,26%
40	184	39,9	0,16%
45	152	44,8	0,35%
50	129	49,8	0,36%
55	111	55,1	0,18%
60	98	60,0	0,03%
65	87	65,2	0,23%
70	78	70,2	0,28%
75	71	74,8	0,32%
80	64	80,0	0,01%
Erro relativo médio			0,36%

Tabela 15 – Determinação do erro relativo no processo de linearização do sensor TRTC-5000

Para de fato construir o experimento da lâmina oscilatória, no qual o ângulo de incidência/reflexão é incontornável, é necessário a execução da sua montagem mecânica:

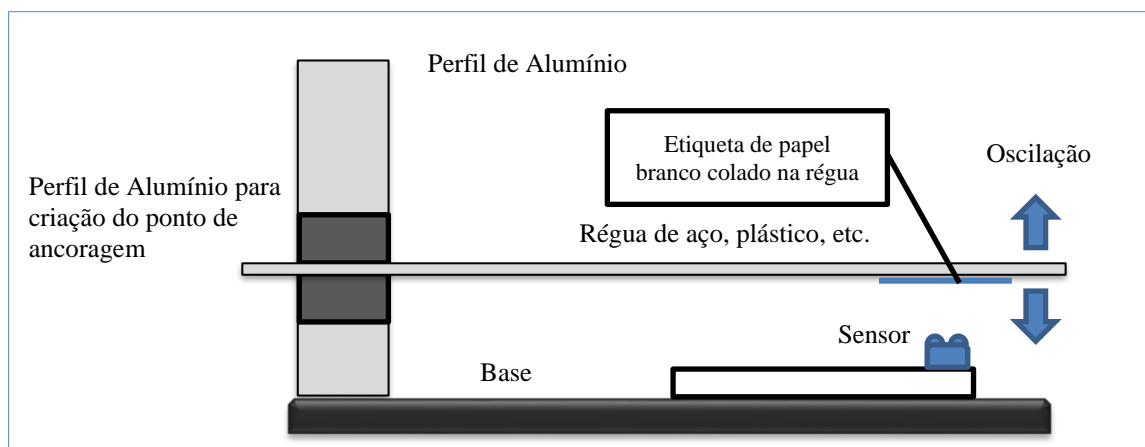


Figura 22 – Esquema simplificado da montagem do experimento

Durante o experimento, a régua oscila entre os dois pontos de máxima amplitude e por estar ancorada em uma das extremidades esta oscilação ocorre em conjunto com a inclinação da extremidade, em balanço, da régua gerando um movimento composto. Desta maneira, é imprescindível a linearização do sensor no próprio experimento.

Nas medições necessárias para este procedimento, considerou-se a base do protoboard como referência (distância placa-objeto, ver figura 21), pois a posição do sensor sob a régua impossibilita a utilização da referência física do mesmo. A régua está ancorada na posição referente a leitura de 20 cm e o sensor está posicionado a 3 cm da ponta livre da régua. Para realizar uma leitura de distância com maior precisão entre a régua e a placa (protoboard), foi utilizado um paquímetro marca Starret modelo 125MEB com resolução de 0,05 mm.

Distância (mm)	Nível após digitalização Medição 1	Nível após digitalização Medição 2	Nível após digitalização Medição 3	Nível após digitalização Média
20	814	804	802	807
25	531	537	533	534
30	372	372	370	371
35	271	272	275	273
40	206	206	203	205
45	159	159	159	159
50	128	128	128	128
55	104	104	104	104
60	84	84	84	84

Tabela 16 – Dados da resposta do sensor TRTC-5000 para reflexão difusa no intervalo 20 - 60 mm

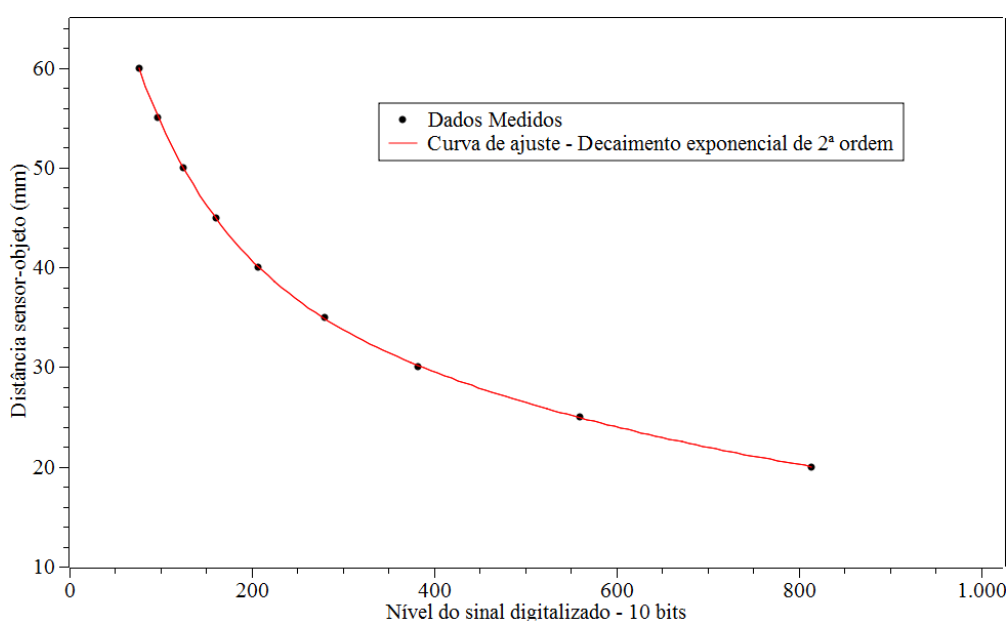


Gráfico 3 – Curva de resposta do sensor TRTC-5000 para reflexão difusa no intervalo 15-80 mm

A Função de linearização do sensor TRTC-5000 no experimento, como esperado, também é uma exponencial de decaimento de 2ª ordem, como apresentado:

$$y = 14,585 + 36,733 \times e^{-\frac{x}{421,707}} + 50,075 \times e^{-\frac{x}{71,122}}$$

Sendo y a distância entre a placa (*protoboard*) e a régua em mm, e x o valor de resposta digitalizada com resolução de 10 bits do sensor.

A função que caracteriza a resposta do sensor é similar àquela obtida na medição estática, o que indica que a inclinação da régua não induziu uma grande alteração na resposta do sensor. A determinação do erro relativo é crucial para que seja possível uma avaliação da qualidade do processo de linearização do sensor.

Distância sensor-régua (mm)	Nível após digitalização (Média)	Distância obtida após linearização	Erro Relativo
20	807	20,0	0,03%
25	534	25,0	0,13%
30	371	30,1	0,32%
35	273	34,9	0,32%
40	205	40,0	0,05%
45	159	45,1	0,30%
50	128	50,0	0,04%
55	104	54,9	0,20%
60	84	60,1	0,09%
Erro relativo médio			0,16%

Tabela 17 – Determinação do erro relativo na linearização do sensor no experimento

Caso a montagem mecânica seja alterada, a metodologia para linearização do conjunto sensor/régua deve ser realizada novamente.

Uma vez concluído o procedimento da linearização do sensor TRTC-5000 e sua verificação de linearidade, para que ele apresente os dados de posição em unidade compatível com o Sistema Internacional (SI), será indispensável a implementação de um programa de aquisição dos dados. O código a ser transferido para o *Arduino* deve contemplar a digitalização dos dados analógicos, oriundos do sensor e sua transmissão por interface serial. A base de tempo da aquisição e transmissão dos dados pelo *Arduino* deve ser constante e possuir uma base de tempo compatível ao SI.

A determinação desta base de tempo depende do número de amostras necessárias para uma boa representação da oscilação da régua. Para isso, foi primordial realizar uma

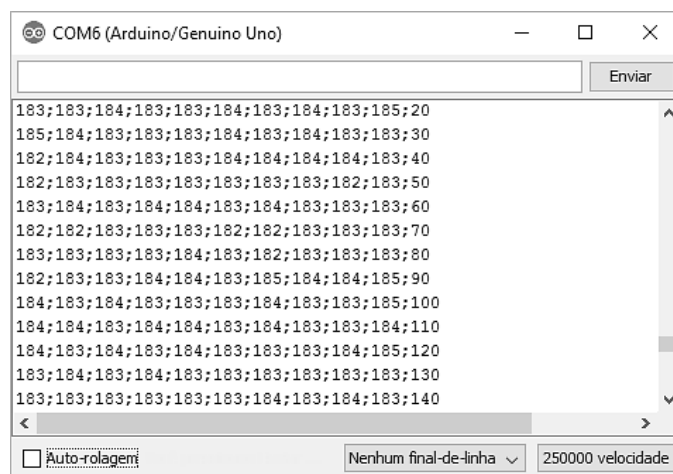
medição prévia, com o uso de um osciloscópio. Para a régua de aço com um balanço de 200 mm foi medida uma frequência média de oscilação de aproximadamente 14 Hz e período de 70 ms. Com base nestes dados e no tempo exigido para que o *Arduino* digitalize o sinal analógico, que é da ordem de 100  $\mu$ s, adotaremos o padrão de tempo de 1ms, ou seja, a geração de 1 sinal referente a medição de posição da régua a cada 1 ms.

```
int valor = 0, contador=0;
long tempoAtual = 0, tempoAnterior = 0, tempo = 0;

void setup(){Serial.begin(250000);}

void loop(){
  valor = analogRead(0);
  tempoAtual = micros() - tempoAnterior;
  while ( tempoAtual < 993){tempoAtual = micros() - tempoAnterior;}
  tempoAtual = micros()-tempoAnterior;
  tempoAnterior = micros();
  tempo = tempo + tempoAtual / 1000;
  Serial.print(valor); Serial.print(";");
  contador = contador + 1;
  if (contador == 10){
    Serial.println(tempo);
    contador = 0;
  }
}
```

O código proposto gera uma medição de posição da régua a cada 1 ms, com um erro absoluto, medido, de 0,004 ms ou relativo de 0,4%. Com a finalidade de obter uma maior qualidade e velocidade do canal de comunicação serial, a cada transmissão são enviados 10 medições de posição e uma medição do tempo decorrido destas medições, conforme é mostrado no terminal serial do *Arduino* a seguir:



```
COM6 (Arduino/Genuino Uno)
183;183;184;183;183;184;183;184;183;185;20
185;184;183;183;183;184;183;184;183;183;30
182;184;183;183;183;184;184;184;183;40
182;183;183;183;183;183;183;182;183;50
183;184;183;184;184;183;184;183;183;60
182;182;183;183;183;182;182;183;183;70
183;183;183;183;184;183;182;183;183;80
182;183;183;184;184;183;185;184;184;90
184;183;184;183;183;183;184;183;183;100
184;184;183;184;184;183;184;183;183;110
184;183;184;183;184;183;183;183;184;120
183;184;183;184;183;183;183;183;183;130
183;183;183;183;183;184;183;184;183;140
```

Figura 23 – Terminal serial do *Arduino* recebendo dados, sendo os dez primeiros valores de posição em sinal analógico de 10 bits (0 a 1024) e um valor de tempo em ms.

A interface do experimento desenvolvida em *processing*, recebe os dados via conexão serial e os apresenta em forma de gráfico em uma tela de 20 divisões horizontais por 8 divisões verticais. A escala vertical é milimétrica enquanto que a escala horizontal possui 6 possíveis resoluções por divisão: 10 ms, 20 ms, 50 ms, 100 ms, 500 ms e 2 s. Os detalhes da interface computacional podem ser observadas na figura 24.

Estas diferentes escalas horizontais possibilitam que o aluno realize uma adequada medição do período e da amplitude da onda, avaliando o seu amortecimento. Outras funcionalidades foram implementadas como o disparo único, que permite reter a visualização do sinal a cada tela completa; os indicadores de nível do sinal e posição em mm localizados no rodapé da tela, o ajuste de offset para compensar oscilações no ajuste de zero e a gravação, em arquivo, de todas as medições realizadas na sessão para uma análise posterior.

Da mesma forma que foi desenvolvido na linearização do sensor, é necessário realizarmos a avaliação da qualidade do padrão de tempo implementado pelo *Arduino* e processado pelo *Processing*. Para alcançar este objetivo, foi escolhido o comprimento em balanço da régua de 20 cm e foram realizadas dez medições via interface e via análise de dados.

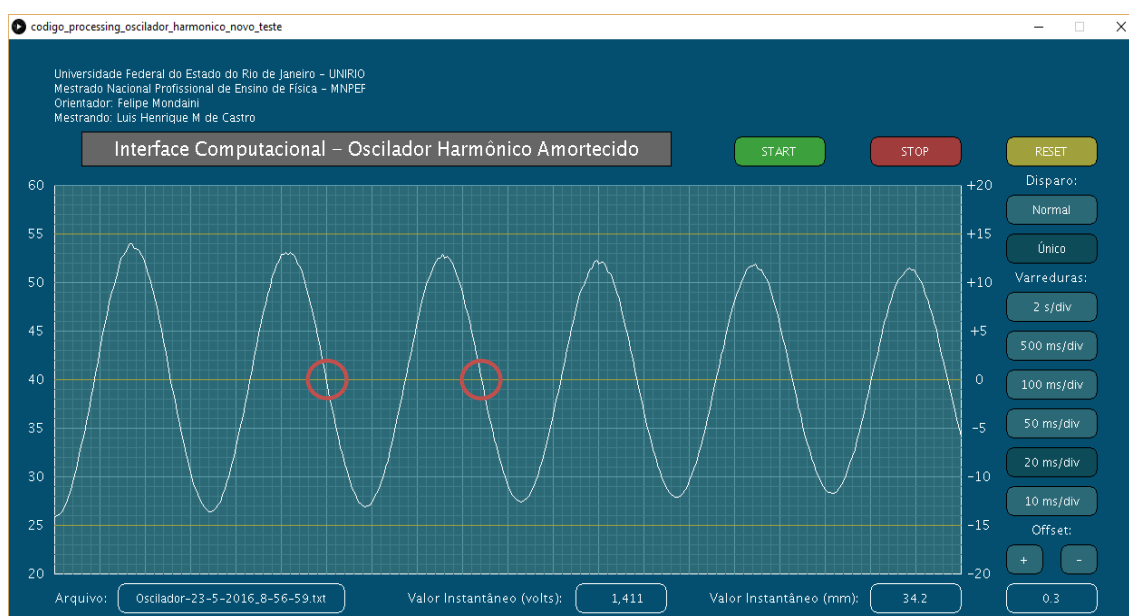


Figura 24 – Tela da interface computacional realizando medida de oscilação com indicação de referências para a determinação gráfica do período (vermelho)

O sinal medido pela interface computacional possui um período de 3 divisões e 2 subdivisões. Considerando a varredura selecionada, podemos concluir que cada divisão

possui o valor de 20 ms e cada subdivisão possui o valor de 1/5 da divisão, ou seja, 4 ms. Desta maneira temos:  $3 \times 20 + 2 \times 4 = 60 + 8 = 68 \text{ ms}$ .

O período pode ser obtido com base nos dados armazenados no arquivo gerado pela interface computacional de duas maneiras: a análise direta dos dados e a análise do gráfico gerado a partir dos dados. O primeiro método é mostrado a partir dos dados apresentados a seguir:

4010	35,2	4023	42,6	4036	47,5	4049	43,0	4062	35,4	4075	34,4
4011	35,5	4024	43,4	4037	47,5	4050	42,0	4063	35,0	4076	34,6
4012	36,0	4025	44,0	4038	47,4	4051	41,7	4064	34,6	4077	35,0
4013	36,5	4026	44,4	4039	47,4	4052	41,1	4065	34,4	4078	35,3
4014	37,1	4027	45,0	4040	47,0	4053	40,4	4066	34,1	4079	35,8
4015	37,6	4028	45,3	4041	46,7	4054	39,7	4067	33,9	4080	36,3
4016	38,2	4029	45,9	4042	46,5	4055	39,2	4068	33,8	4081	36,7
4017	38,7	4030	46,4	4043	46,1	4056	38,6	4069	33,7	4082	37,3
<b>4018</b>	<b>39,5</b>	4031	46,6	4044	45,7	4057	38,0	4070	33,7	4083	37,8
4019	40,0	4032	47,0	4045	45,2	4058	37,3	4071	33,7	4084	38,4
4020	40,8	4033	47,1	4046	44,7	4059	36,8	4072	33,8	4085	39,0
4021	41,5	4034	47,3	4047	44,2	4060	36,2	4073	34,0	<b>4086</b>	<b>39,6</b>
4022	42,0	4035	47,4	4048	43,4	4061	35,8	4074	34,2	4087	40,3

Tabela 18 – Dados oriundos da interface computacional

O período será determinado na segunda vez que o sinal passar o mais próximo da distância considerada como referência. Consultando a tabela, podemos definir as amostras 4018 e 4086, como o tempo é dado em milissegundo temos:  $4086 - 4018 = 68 \text{ ms}$ . Observa-se que o valor não é exato mesmo tendo 86 amostras em apenas um ciclo de onda.

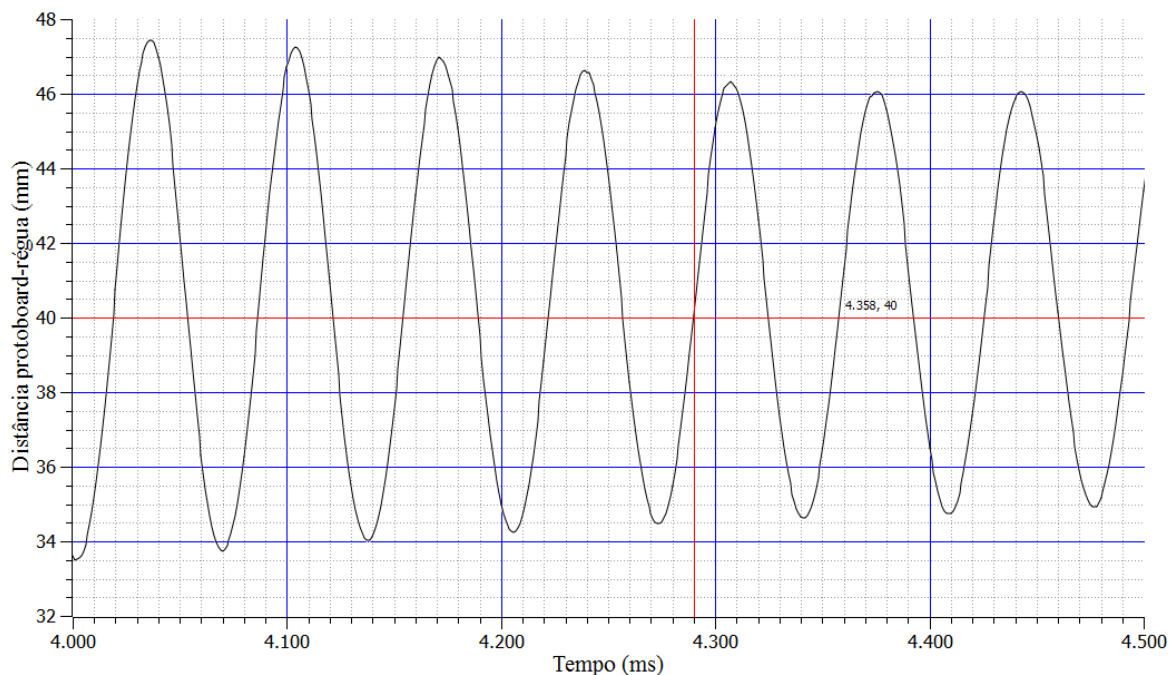


Gráfico 4 – Gráfico gerado a partir dos dados oriundos da interface computacional

Usando a interface do *SCIDavis* ou de qualquer software similar, é possível medir valores de pontos no gráfico, assim como exemplificado no gráfico 4, no qual a linha vermelha indica a referência, no caso 4290 ms e onde se posiciona o cursor sobre a curva temos a medição deste ponto, no caso, 4358 ms. Realizando a subtração destas leituras obtém-se o período.  $4358 - 4290 = 68\text{ms}$ .

Foram utilizadas três metodologias para determinar o período da oscilação da lâmina no experimento, porém é imperioso medir, de maneira externa, a oscilação da lâmina para que possamos ter um padrão de comparação e verificarmos a qualidade da medição proporcionada pelo experimento. Para isto, o sensor foi separado do *Arduino*, alimentado por uma fonte externa de 5 VDC e conectado através de uma ponta de prova a um osciloscópio digital marca INSTEK modelo GDS806S para também realizar medições do período de oscilação da lâmina. O osciloscópio foi ajustado, manualmente, com os parâmetros: varredura vertical: 200 mV/divisão; acoplamento: AC; varredura horizontal: 25 ms/divisão e trigger: DC com disparo na rampa de subida.

Pelo fato da oscilação ser de baixa frequência e possuir o amortecimento em sua amplitude, foram realizadas dez medições para minimizar o erro de aquisição do osciloscópio.

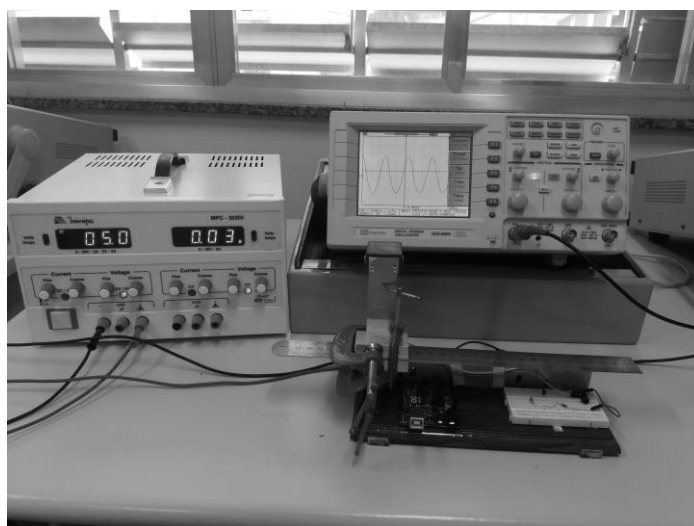


Figura 25 – Montagem experimental para realizar medição do período de oscilação

Na tela do osciloscópio, no acionamento do menu medidas, são apresentadas medições automáticas de frequência, período, tensão de pico a pico, tensão eficaz e tensão máxima. No rodapé da tela, são apresentadas as escalas de volts/divisão dos canais 1 e 2,

com seus respectivos tipos de acoplamento e a escala de tempo/divisão. Estas indicações dos valores das escalas permitem a medição gráfica na tela.

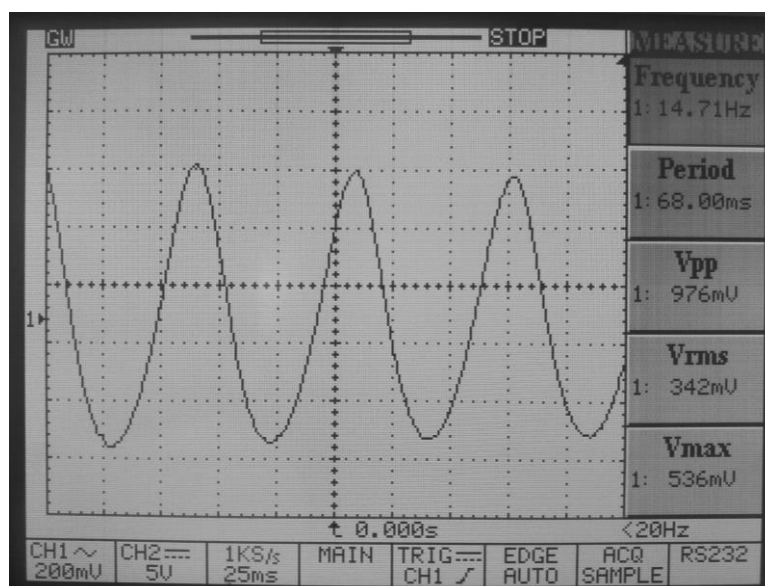


Figura 26 – Tela do osciloscópio digital com a leitura dos parâmetros da oscilação

Referência	Período medido na interface (ms)	Período medido no osciloscópio (ms)	Erro Relativo
1	68	67,75	0,37%
2	68	68,33	0,49%
3	68	68,00	0,00%
4	68	68,32	0,47%
5	68	68,40	0,59%
6	68	67,70	0,44%
7	68	67,96	0,06%
8	68	68,33	0,49%
9	68	68,01	0,01%
10	68	67,88	0,18%
Erro relativo médio			0,31%

Tabela 19 – Determinação do erro relativo médio do padrão de tempo

Na análise dos erros, foi considerado o período de 68 ms como o tempo correto de uma oscilação da lâmina. Desta maneira, calculou-se o erro relativo de cada medida do osciloscópio e o erro relativo médio em dez medições. O erro calculado foi inferior a 1%.

As comprovações dos erros relativos à linearização do sensor e da base de tempo implementadas pelo *Arduino* e pelo *Processing* são imprescindíveis para que, de fato, o



experimento seja um objeto educacional funcional quantitativo e não uma simples demonstração da oscilação da régua.

O uso desse experimento, objetivamente, favorece o desenvolvimento das habilidades e das competências preconizadas pelos descritores relacionados ao estudo das ondas, presentes na Base Nacional Comum e no Currículo Mínimo do Estado do Rio de Janeiro, conforme apresentado na tabela a seguir:

Estudo de Ondas
Base Nacional Comum
– Rever ou avaliar movimentos em quedas, sistemas planetários ou objetos sob a ação de molas, tendo como pressuposto a conservação da energia mecânica.
– Conceber e realizar arranjos experimentais em que se possa avaliar ou verificar a antecipação de movimentos, como a oscilação de pêndulos, ou de deformações, como na colisão de objetos elásticos.
– Determinar experimentalmente relações entre variáveis como o valor da aceleração da gravidade, a altura e o tempo numa queda livre, ou no percurso de uma esfera em canaleta inclinada, ou entre comprimento de pêndulos e sua frequência de oscilação.
– Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.
Currículo Mínimo do Estado do Rio de Janeiro
– Compreender as propriedades das ondas e como elas explicam fenômenos presentes em nosso cotidiano.
– Compreender a importância dos fenômenos ondulatórios na vida moderna sobre vários aspectos, entre eles sua importância para a exploração espacial e na comunicação.

Tabela 20 – Descritores relacionados ao estudo das ondas

Atividade1: Demonstrar o uso do experimento.

Propor aos alunos que manipulem livremente a régua, para cima e para baixo e visualizem o efeito do registro temporal de posição da régua na interface computacional. Em um segundo momento, com o auxílio de uma régua realizar medições de 30, 35, 40, 45, 50 e registrar, em uma tabela, os valores obtidos. A partir dos valores medidos e dos valores registrados na interface, calcular o valor médio para cada medição e o erro experimental parcial e total.

Habilidades e Competências trabalhadas:

- Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.

Distância (mm)	Medição 1 (volts)	Medição 1 (mm)	Medição 2 (volts)	Medição 2 (mm)	Medição 3 (volts)	Medição 3 (mm)
30	1,807	30,3	1,797	30,4	1,836	30,1
35	1,318	35,0	1,318	35,0	1,323	34,9
40	1,011	39,7	1,006	39,8	1,006	39,8
45	0,791	44,6	0,781	44,9	0,786	44,8
50	0,620	50,1	0,625	49,9	0,620	50,1

Tabela 21 – Atividade 1 – Exemplo de dados obtidos pela interface *Processing* a partir da medição de posição de oscilação da régua medida por uma régua auxiliar milimetrada

Distância (mm)	Valor Médio (volts)	Valor Médio (mm)	Erro Relativo
30	1,813	30,27	0,89%
35	1,320	34,97	0,10%
40	1,008	39,77	0,58%
45	0,786	44,77	0,52%
50	0,622	50,03	0,07%
Erro relativo médio			0,43%

Tabela 22 – Atividade 1 – Registro dos valores de cálculo do erro relativo

$$Erro\% = \frac{|valor\ medido - valor\ real|}{valor\ real} \times 100$$

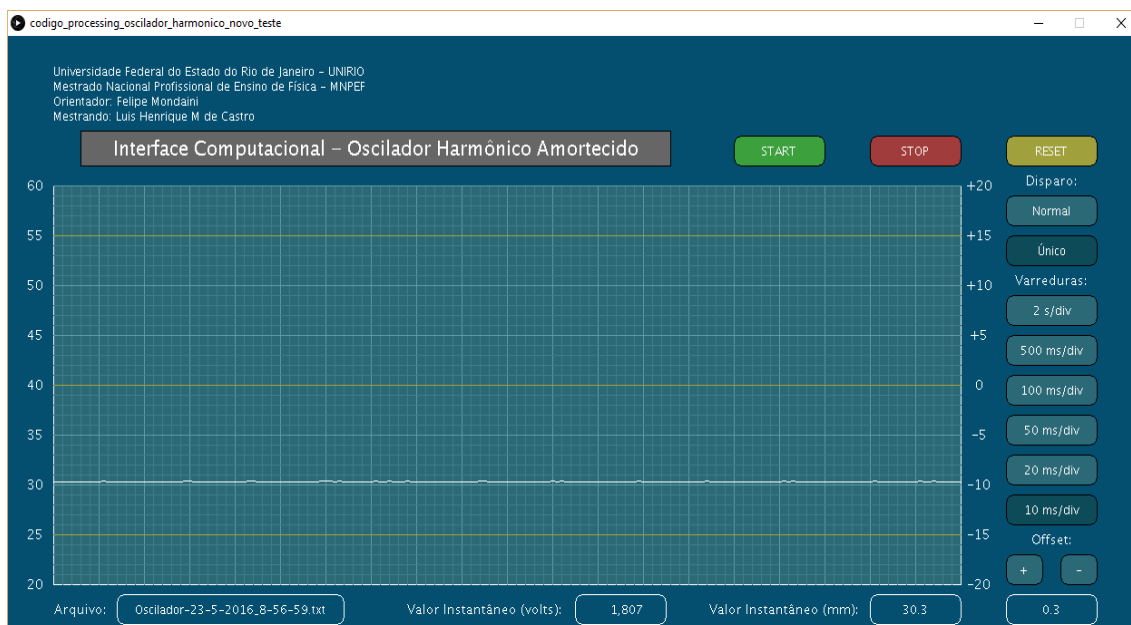


Figura 27 – Atividade 1 – Exemplo de uma das medições realizadas na interface computacional desenvolvida

## Atividade 2: Análise do fenômeno da deformação elástica da lâmina.

A partir do emprego de pequenos valores de massas para a determinação da constante elástica da lâmina. Na sala de aula podemos utilizar moedas de valores correntes como elementos de massa, colocando-as em uma posição definida da régua e realizar as medições de deslocamento com o uso da interface computacional, registrando os resultados em uma tabela. A partir dos valores de massa e deslocamento registrados, construir o gráfico força x deslocamento e determinar a constante elástica. Para obtermos outros valores da constante de elástica podemos alterar a posição da massa na régua ou alterar a própria régua.

### Habilidades e Competências trabalhadas:

- Rever ou avaliar movimentos em quedas, sistemas planetários ou objetos sob a ação de molas, tendo como pressuposto a conservação da energia mecânica.
- Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.

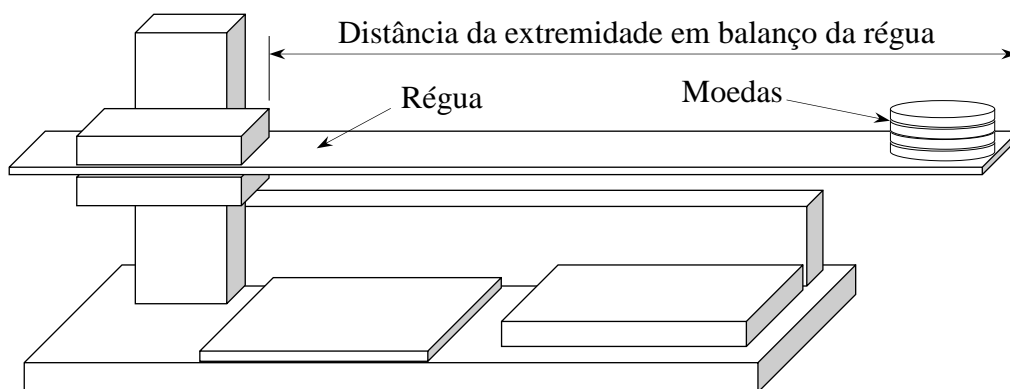


Figura 28 – Atividade 2 – Exemplo de arranjo com as moedas para a determinação da constante elástica

Número de Moedas	Massa Total (g)	Medições de posição			Media (mm)	Deslocamento (mm)
		1 (mm)	2 (mm)	3 (mm)		
0	0	39,9	40,0	40,0	40,00	0
1	7,81	38,9	39,0	39,0	38,97	1,03
2	15,62	38,1	38,1	38,2	38,13	1,87
3	23,43	37,3	37,4	37,4	37,37	2,63
4	31,24	36,5	36,5	36,5	36,50	3,50
5	39,05	35,6	35,6	35,6	35,60	4,40
6	46,86	34,7	34,7	34,7	34,70	5,30
7	54,67	33,9	33,9	33,9	33,90	6,10
8	62,48	33,0	33,0	33,0	33,00	7,00
9	70,29	32,2	32,2	32,1	32,17	7,83
10	78,10	31,3	31,3	31,3	31,30	8,70

Tabela 23 – Atividade 2 – Exemplo de dados obtidos, via interface gráfica, para cada arranjo de moedas de R\$ 0,50

Número de Moedas	Massa Total (kg)	Força Peso (N)	Deslocamento (m)	k (N/m)
0	0	0	0	0
1	$7,81 \cdot 10^{-3}$	$7,66 \cdot 10^{-2}$	$1,03 \cdot 10^{-3}$	74,385
2	$1,562 \cdot 10^{-2}$	$1,532 \cdot 10^{-1}$	$1,87 \cdot 10^{-3}$	81,942
3	$2,343 \cdot 10^{-2}$	$2,298 \cdot 10^{-1}$	$2,63 \cdot 10^{-3}$	87,395
4	$3,124 \cdot 10^{-2}$	$3,064 \cdot 10^{-1}$	$3,50 \cdot 10^{-3}$	87,561
5	$3,905 \cdot 10^{-2}$	$3,830 \cdot 10^{-1}$	$4,40 \cdot 10^{-3}$	87,064
6	$4,686 \cdot 10^{-2}$	$4,600 \cdot 10^{-1}$	$5,30 \cdot 10^{-3}$	86,735
7	$5,467 \cdot 10^{-2}$	$5,363 \cdot 10^{-1}$	$6,10 \cdot 10^{-3}$	87,920
8	$6,248 \cdot 10^{-2}$	$6,129 \cdot 10^{-1}$	$7,00 \cdot 10^{-3}$	87,561
9	$7,029 \cdot 10^{-2}$	$6,895 \cdot 10^{-1}$	$7,83 \cdot 10^{-3}$	88,064
10	$7,810 \cdot 10^{-2}$	$7,661 \cdot 10^{-1}$	$8,70 \cdot 10^{-3}$	88,064

Tabela 24 – Atividade 2 – Exemplo do cálculo da constante elástica para cada arranjo de moedas de R\$ 0,50

$$F = -kx$$

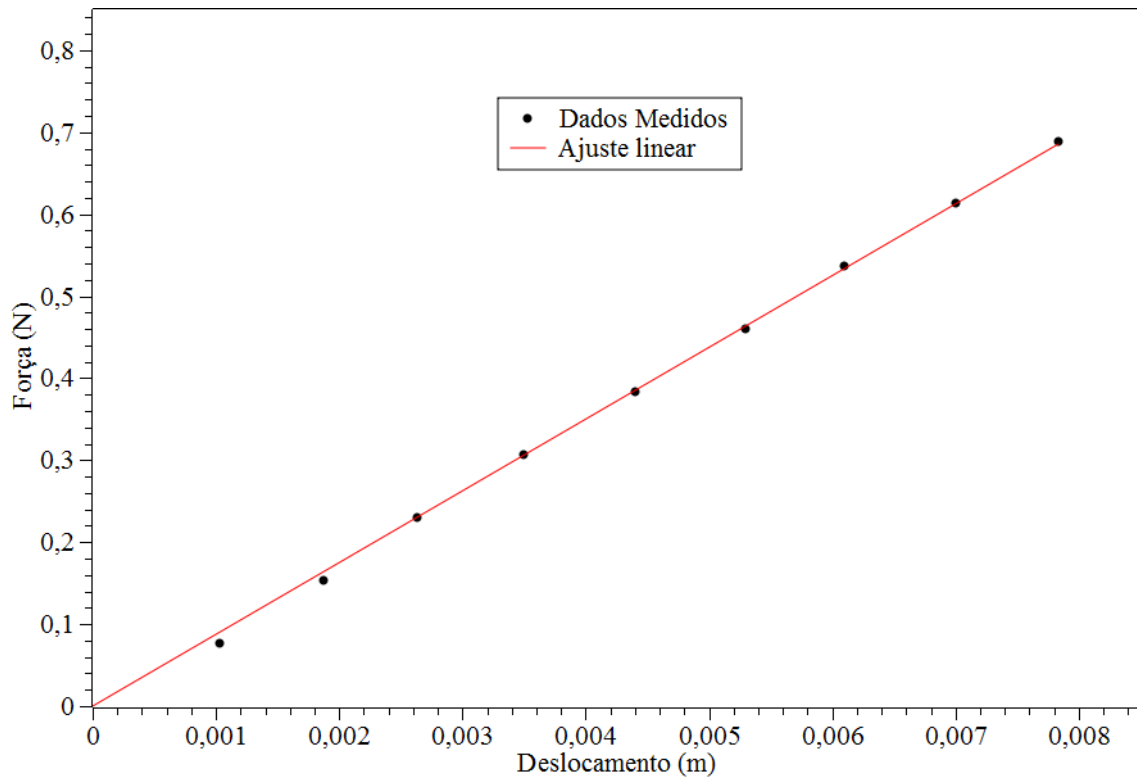


Gráfico 5 – Atividade 2 – Constante elástica (k) da lâmina nos diversos arranjos de moedas de R\$ 0,50

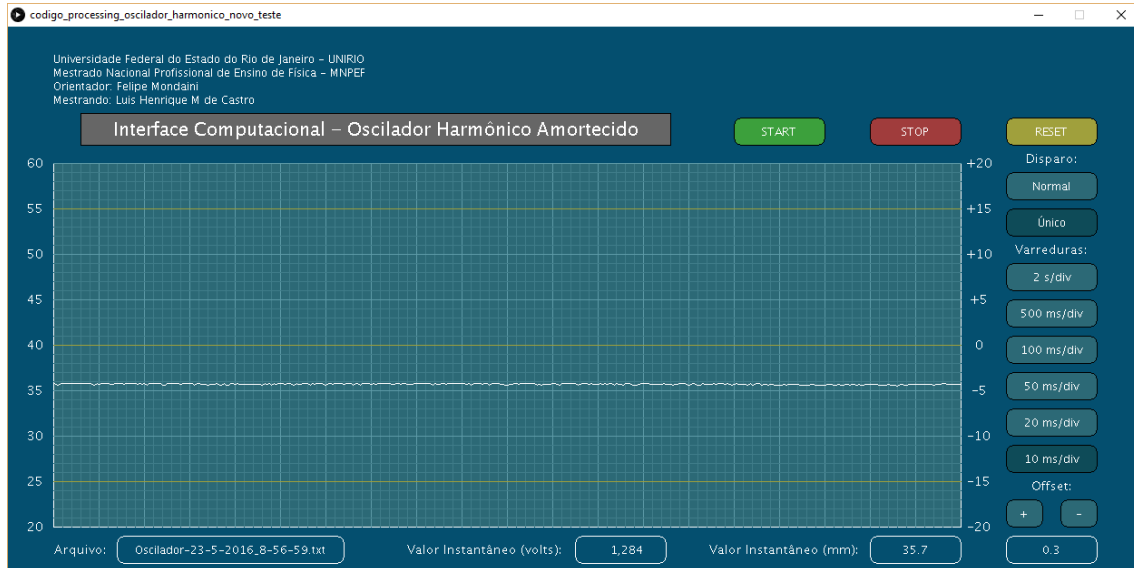


Figura 29 – Atividade 2 – Exemplo de uma das medições realizadas na interface computacional para um dos arranjos de moedas de R\$ 0,50

### Atividade 3: Análise do movimento oscilatório

A partir do acionamento da régua, observar o registro do sinal na interface computacional, determinando período, frequência e amplitude de um ciclo. Altere o

comprimento do balanço da régua determinando a relação da variação do comprimento livre da régua com a variação do período de oscilação. Também podemos alterar o material da lâmina, utilizando outras régua, de mesmo comprimento, e definindo a frequência natural de oscilação ou inserindo massas solidárias na extremidade livre da régua.

Habilidades e Competências trabalhadas:

- Conceber e realizar arranjos experimentais em que se possa avaliar ou verificar a antecipação de movimentos, como a oscilação de pêndulos, ou de deformações, como na colisão de objetos elásticos.
- Determinar experimentalmente relações entre variáveis como o valor da aceleração da gravidade, a altura e o tempo numa queda livre, ou no percurso de uma esfera em canaleta inclinada, ou entre comprimento de pêndulos e sua frequência de oscilação.

Balanço da régua	Amplitude de pico positivo (mm)	Período (s)	Frequência (Hz)
18 cm	16,0	0,056	17,86
19 cm	15,0	0,063	15,87
20 cm	16,5	0,068	14,71
21 cm	14,0	0,075	13,33
22 cm	12,0	0,083	12,05
23 cm	15,0	0,090	11,11
24 cm	14,0	0,097	10,31

Tabela 25 – Atividade 3 – Determinação dos parâmetros da oscilação para cada arranjo da régua

A relação entre o comprimento em balanço da régua com a variação do período de oscilação é inversa, ou seja, quanto maior o comprimento, em balanço, da régua menor será a frequência de oscilação.



Figura 30 – Oscilação na interface computacional para um balanço da régua de 20 cm

#### Atividade 4: Análise da taxa de decaimento

A partir do acionamento da régua podemos medir a amplitude do sinal em cinco ciclos subsequentes e com base em tabelas e gráficos definirmos o aspecto e a taxa de decaimento. Deve ser também apresentado o decaimento total da oscilação, analisar a qual função matemática pertence e analisar as possíveis causas deste decaimento. Podemos também alterar materiais ou inserir massas solidárias à régua.

Habilidades e Competências trabalhadas:

- Rever ou avaliar movimentos em quedas, sistemas planetários ou objetos sob a ação de molas, tendo como pressuposto a conservação da energia mecânica.
- Conceber e realizar arranjos experimentais em que se possa avaliar ou verificar a antecipação de movimentos, como a oscilação de pêndulos, ou de deformações, como na colisão de objetos elásticos.
- Determinar experimentalmente relações entre variáveis como o valor da aceleração da gravidade, a altura e o tempo numa queda livre, ou no percurso de uma esfera em canaleta inclinada, ou entre comprimento de pêndulos e sua frequência de oscilação.
- Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.

Ciclo	Amplitude relativa positiva (mm)	Varição em relação ao ciclo anterior (mm)
1	14,0	-----
2	13,2	0,8
3	12,8	0,4
4	12,2	0,6
5	11,8	0,4
6	11,4	0,4
7	11,0	0,4
8	10,4	0,6
9	10,0	0,4
10	9,6	0,4
11	9,2	0,4
12	9	0,2
Variação total		5,0
Variação média		0,6

Tabela 26 – Amplitudes de oscilação em diferentes ciclos consecutivos, considerando um balanço da régua de 20 cm

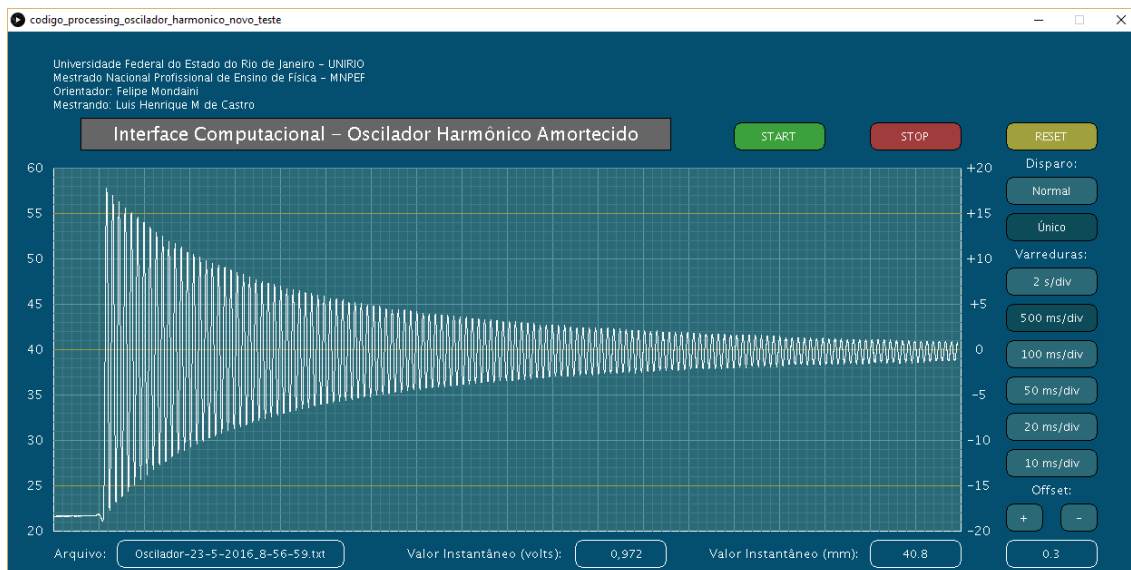


Figura 31 – Amortecimento para uma régua de aço considerando um balanço de 20 cm

A envoltória observada no decaimento da amplitude do sinal é o amortecimento presente na oscilação da régua. A amplitude de um oscilador amortecido em um tempo  $t$  qualquer é dado por uma função exponencial:

$$x = Ae^{-\left(\frac{b}{2m}\right)t} \cos(\omega t + \delta)$$

Sendo:

- |                                  |                                    |
|----------------------------------|------------------------------------|
| A: amplitude máxima de oscilação | $\omega$ : frequência do oscilador |
| b: constante de amortecimento    | $\delta$ : fase                    |
| m: massa do oscilador            |                                    |
| t: tempo do movimento            |                                    |

Também podemos formular a equação em função do tempo para a amplitude:

$$A(t) = Ae^{-t/2\tau}, \text{ sendo } \tau = \frac{m}{b}$$

A força de resistência do ar, no caso da régua, é a principal força que contraria o movimento neste experimento, gerando por consequência o amortecimento.

Neste objeto educacional foram realizados experimentos envolvendo conceitos como a constante elástica de uma lâmina e movimento harmônico amortecido, porém este elemento sensor pode ser aplicado a qualquer experimento físico que envolva a medição de pequenos deslocamentos. Alterando apenas o modelo do sensor infravermelho e mantendo a mesma arquitetura do experimento realizado podemos alcançar medições de distância



superiores a 1 metro, o que possibilita o desenvolvimento de novos experimentos como registrador de deslocamento de móveis ou como um analisador de choques mecânicos.

Como visto, é possível criar um objeto didático a baixo custo. No entanto, um laboratório não é feito de apenas um experimento, é preciso bem mais para, de fato, cumprir com todo o planejamento anual. Sendo assim, no próximo capítulo, será apresentado, descrito e analisado mais um produto educacional que, mesmo demandando poucos recursos, também é muito atraente e funcional.

### **Estação Meteorológica com o uso do *Arduino* e do *Processing***

A previsão do tempo está presente nos jornais, nos telejornais e na internet. É ela que permite, muitas vezes, as pessoas se prepararem para enfrentar as mudanças climáticas, levando, por exemplo, ao sair de casa, agasalhos e guarda-chuva, ou escolherem a melhor data para uma viagem ou para um evento.

Deste modo, é possível dizer que a meteorologia é um assunto que facilmente demonstra aos alunos as aplicações da Física no dia a dia, conforme salientam as orientações Educacionais Complementares aos Parâmetros Curriculares Nacionais: “a Física deve apresentar-se, portanto, como um conjunto de competências específicas que permitam perceber e lidar com os fenômenos naturais e tecnológicos, presentes tanto no cotidiano mais imediato quanto na compreensão do universo distante, a partir de princípios, leis e modelos por ela construídos” (PCN+, 2002, p. 2).

O termo meteorologia foi cunhado por Aristóteles, no século IV A.C, derivado de *meteoro* (partículas em suspensão) e *logos* (conhecimento). O termo foi empregado para nomear a ciência que estuda os fenômenos ocorridos na atmosfera terrestre ou mesmo de outros planetas.

No Brasil, o estudo da meteorologia se inicia em 1781, com a campanha de medições meteorológicas nas cidades do Rio de Janeiro e de São Paulo e a profissão de meteorologista foi regulamentada em 14 de outubro de 1980.

As estações meteorológicas monitoram, muitas em tempo real, nosso tempo e o nosso clima, possibilitando, através das medidas realizadas, o monitoramento do clima e a previsão do tempo. Para realizar essas atividades em sala de aula, um equipamento de baixo custo para a medição de algumas das variáveis atmosféricas pode ser desenvolvido com o uso do *Arduino* e do *Processing*, viabilizando o monitoramento da pressão atmosférica, da temperatura, da umidade relativa do ar e da luminosidade.

O estudo e a análise dos dados provenientes da medição desses parâmetros podem promover a integração entre algumas disciplinas como a química, física e biologia, da área de ciências da natureza; a geografia da área de ciências humanas e as disciplinas das áreas de linguagem.

Os estudos dos fenômenos atmosféricos estão intimamente relacionados à Física e à Química. Podemos citar, como exemplo destas relações, os processos termodinâmicos, a

composição e a estrutura da atmosfera, a propagação da radiação eletromagnética e das ondas acústicas através da atmosfera, os processos físicos envolvidos na formação de nuvens e precipitação, a eletricidade atmosférica, as reações físico-químicas dos gases e partículas, a microfísica de gotas e cristais e a própria química da atmosfera. O conhecimento da atmosfera e das mudanças climáticas são apontados como um exemplo de aplicação da disciplina de química pela Base Nacional Curricular Comum:

As mudanças climáticas e o efeito estufa, o uso de feromônios como alternativa aos agrotóxicos no combate às pragas agrícolas, a necessidade de informações sobre a presença de transgênicos em rótulos de alimentos e os custos ambientais das minerações são apenas alguns exemplos de assuntos em que o conhecimento químico é vital para que o/a estudante possa posicionar-se e tomar decisões com consciência (2016, p. 146).

O processo biológico da fotossíntese e as reações químicas possibilitaram a evolução da atmosfera primordial formada por metano ( $\text{CH}_4$ ), amônia ( $\text{NH}_3$ ), dióxido de carbono ( $\text{CO}_2$ ) e vapor de água ( $\text{H}_2\text{O}$ ) para a atmosfera que conhecemos hoje, onde predominam os gases nitrogênio (N) e oxigênio (O). Além dessa evolução, os processos biológicos interagem de forma constante com a atmosfera, como podemos exemplificar pelas florestas que mantêm o equilíbrio ambiental (umidade do ar, temperatura, taxa de polinização, etc.).

Na geografia, a conexão com as variáveis atmosféricas (temperatura, umidade relativa do ar, luminosidade, etc.) se materializa nos estudos do clima, da vegetação, do relevo, da hidrografia e da climatologia presentes nos livros didáticos do Ensino Médio.

Nos livros didáticos de Física, as variáveis pressão e temperatura são normalmente apresentadas de forma generalizada e não específicas a atmosfera, enquanto os conceitos de pressão atmosférica, temperatura atmosférica, umidade relativa do ar e luminosidade estão presentes em fontes bibliográficas de geografia ou de meteorologia.

A pressão é definida como a razão entre a força aplicada em uma superfície e a área de aplicação dessa força. A pressão atmosférica é o peso exercido por uma coluna de ar, com seção reta de área unitária, que se encontra acima do observador, em um dado instante e local. Fisicamente, representa o peso que a atmosfera exerce por unidade de área, conforme define Varejão-Silva em seu livro *Meteorologia e Climatologia* (2006). A unidade padrão de pressão no Sistema Internacional (SI) é o Pascal (Pa) ( $\text{Newton/m}^2$ ). Meteorologistas tem usado tradicionalmente a unidade bar, mas a unidade Pascal vem sendo cada vez mais adotada. Por tradição histórica, ainda é utilizada a unidade milímetros de mercúrio (mmHg). A referência para os valores de pressão atmosférica é o seu valor médio ao nível do mar: 1013,25 mbar, 101,325 KPa, 1013,25 hPa ou 760 mmHg.

A temperatura é a medida da agitação das partículas que compõe um certo material. Se considerarmos as moléculas de um gás, quanto maior a sua temperatura mais energia cinética terão essas moléculas. Quando queremos medir a temperatura de um corpo, precisamos usar uma escala termométrica, isto é, uma maneira de correlacionar um conjunto de números às temperaturas. As três escalas termométricas mais comuns são Celsius (C), Fahrenheit (F) e Kelvin (K).

A noção comum subjetiva de temperatura não é, de forma alguma, confiável. Uma experiência familiar consiste em um dia muito frio tocar-se uma grade de metal que esteja ao ar livre e, em seguida, tocar-se um objeto de madeira próximo. Tem-se a sensação de que a grade está mais fria, embora ambos os objetos estejam à mesma temperatura. O que se está testando quando se toca um objeto frio não é somente a sua temperatura, mas também a sua capacidade de transferir energia (na forma de calor) da mão (provavelmente quente). Nestes casos, a mão fornece uma medida subjetiva e incorreta da temperatura. (HALLIDAY, 2003)

A umidade relativa do ar é definida como a razão entre o percentual (em número de moléculas) de água no ar pelo percentual que corresponde à saturação naquela temperatura do ambiente. Por exemplo, a 30°C pode existir no máximo 4% de moléculas de água no estado de vapor no ar; caso efetivamente exista apenas 3%, então a umidade relativa do ar será 3 dividido por 4, portanto, 0,75 ou 75%. A autora Ercília Torres Steink nos esclarece a relação entre a umidade e a formação de chuvas em seu livro *Climatologia Fácil*:

Quando os instrumentos indicam umidade relativa de 100%, isso quer dizer que o ar está totalmente saturado com vapor d'água e, àquela temperatura, ele não pode mais absorver vapor d'água. (...) quando a umidade relativa do ar está em torno de 100%, ocorre a condensação, ou seja, a formação das nuvens, e, se as condições permitirem, a ocorrência de chuva. Isso não significa, porém, que a umidade relativa do ar deva ser de 100% para que chova — basta que seja 100% onde as nuvens estão se formando. (2012, p. 60)

A medição da luminosidade permite determinar a iluminação solar, ou seja, o período de tempo que estamos sujeitos a irradiação solar. Com esta medição, também podemos observar, de maneira aproximada, o horário do nascer do sol e do pôr do sol (ocaso) em dias sem nebulosidade. Também podemos ter um indicador comparativo da nebulosidade do dia, pois dias menos nebulosos serão por consequência mais claros.

Sob o ponto de vista estritamente geométrico, o nascimento e o ocaso do Sol ocorrem quando o centro do disco solar aparentemente coincide com o plano do horizonte local. Na prática, porém, o nascimento e o ocaso do Sol são definidos como os instantes em que o bordo do disco solar parece tangenciar o plano do horizonte local, supostamente desobstruído. Nessas ocasiões, a verdadeira posição do centro do disco solar é 50' abaixo daquele plano. Isso advém do fato do raio daquele disco subtender um arco de 16' e da refração atmosférica aumentar em cerca de 34' o ângulo de elevação do Sol, quando próximo à linha do horizonte (LIST, 1971).

Para o desenvolvimento da estação meteorológica com o uso do *Arduino* e *Processing* são necessários, além do *Arduino* que realiza a aquisição dos dados e do *Processing* que implementa a interface gráfica, sensores que realizam a medição das variáveis meteorológicas. O esquema de ligações dos sensores bem como a ligação do LED indicador de funcionamento da estação ao *Arduino* são apresentadas a seguir:

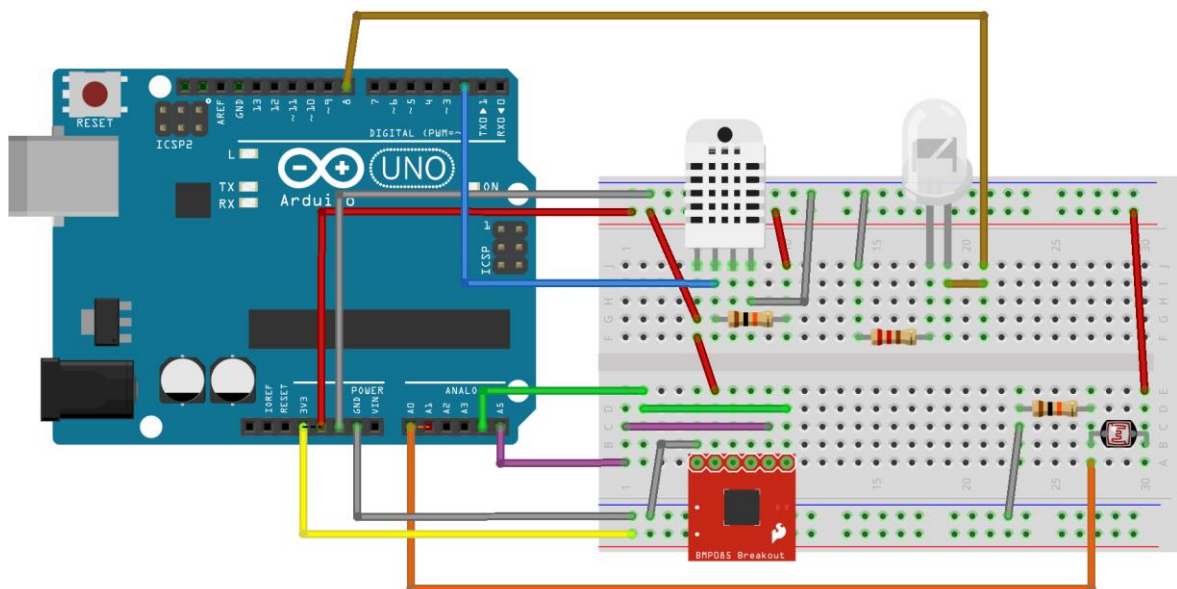


Figura 32 – Esquema de ligações da estação meteorológica com o uso do *Arduino* e do *Processing*

O módulo BMP180 tem seu funcionamento baseado no fenômeno físico da piezo resistividade e é fabricado com o uso da tecnologia EMC (*Epoxy Molding Compound*) que lhe confere robustez, alta precisão e linearidade assim como estabilidade a longo prazo. Este módulo encapsulado em um único chip é formado por elementos piezo resistivos para medições da pressão atmosférica e da temperatura do ar e por um microprocessador que realiza a digitalização do sinal oriundo dos sensores, a calibração, via software, dos dados referentes as grandezas medidas e a conexão serial bidirecional para configuração do chip e leituras dos dados referentes à pressão atmosférica e à altitude absolutas, à pressão atmosférica e à altitude corrigidas em relação ao nível do mar e à temperatura ambiente.

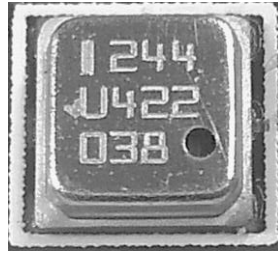


Figura 33 – Detalhe do circuito integrado BMP180

O termo piezo resistividade é oriunda do grego, deriva da palavra *piezin*, que significa pressionar. O efeito da piezo resistividade consiste na alteração da resistência de um material quando sobre ele é aplicada uma determinada tensão mecânica. A descoberta deste fenômeno deve-se ao matemático, físico e engenheiro britânico Lord Kelvin, que em 1856 verificou que a resistência de fios de cobre e ferro aumentava quando sobre eles era aplicada uma determinada tensão mecânica.

Os materiais piezo resistivos são caracterizados pelo que se designa *gauge factor* (GF) ou fator de sensibilidade, definido como a variação fracionária da resistência por unidade de deformação e traduz as mudanças na resistência do material em função das deformações provocadas pela tensão mecânica aplicada.

O sensor BMP180 utiliza piezo resistores baseados em materiais semicondutores, como o silício, pois esses possuem elevados GF's, permitindo-lhe medir pequenas alterações em sua resistência elétrica provocadas por sutis deformações oriundas da ação da pressão atmosférica. A temperatura também influencia bastante os GF's desses sensores, sendo da ordem de  $-0,15\%/^{\circ}\text{C}$ , permitindo ao sensor medir a variação da temperatura.

Este sensor é comercializado para aplicações direcionadas a microcontroladores, como o *Arduino*, em placas contendo o próprio sensor, um regulador de tensão para 3.3 V DC e componentes adicionais para otimizar a alimentação do módulo e a conexão serial.

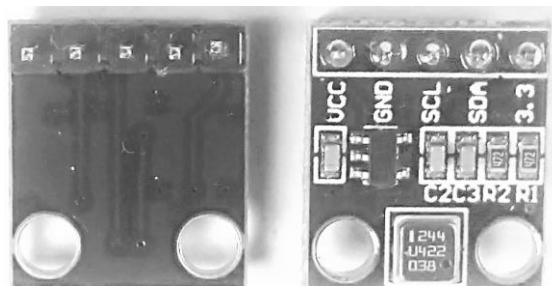


Figura 34 – Placa comercial do sensor BMP180

Tensão de alimentação	1.8 – 3,6 V DC	
Grandeza medida	Pressão atmosférica	Temperatura
Elemento do sensor	<i>piezo-resistive</i>	
Amplitude de medição	300 a 1100hPa (+9000 m a -500 m ref. nível do mar)	- 40 a +85 °C
Acurácia	± 0.12 hPa ±1.0 m	± 1.0 °C
Resolução	0.01 hPa	0.1 °C

Tabela 27 - Características técnicas básicas do sensor BMP180  
 Fonte: Folha de dados do BMP180 da Bosch Sensortec

O módulo digital AOSONG AM2302 é uma versão compatível ao DHT22 para a medição de umidade e temperatura. Este módulo é composto por um sensor capacitivo para a medição da umidade do ar, um termistor para medir a temperatura do ar, e um microcontrolador de 8 bits para digitalizar as medições, realizar a compensação e a calibração via software e possibilitar uma conexão serial unidirecional, por um único fio.

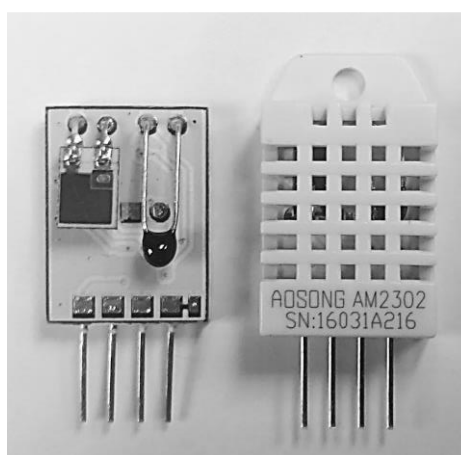


Figura 35 – Vista do sensor AM2302 sem e com o invólucro plástico de proteção

Neste sensor, a medição da umidade relativa do ar é do tipo capacitiva. Este sensor é formado por um capacitor do tipo plano constituído por duas placas metálicas separadas por um dielétrico higroscópico<sup>25</sup> composto de um polímero ativo. O valor da capacitância aumenta quando as moléculas de água são absorvidas pelo dielétrico aumentando sua constante dielétrica. As placas do capacitor consistem de uma placa base de filme de ouro

<sup>25</sup> Capacidade que o material possui de absorver água.

ou platina depositada sobre o substrato cerâmico e uma placa de ouro ou platina permeável ao vapor de água. O valor da capacitância do sensor depende da umidade do ar, tendo um valor de 240pF quando submetido a uma umidade relativa de 60 % a temperatura de 25°C. A variação de 0 a 1000% na umidade relativa provoca uma variação de capacitância de 40 pF, tendo uma variação unitária igual a 0,4 pF /% HR.

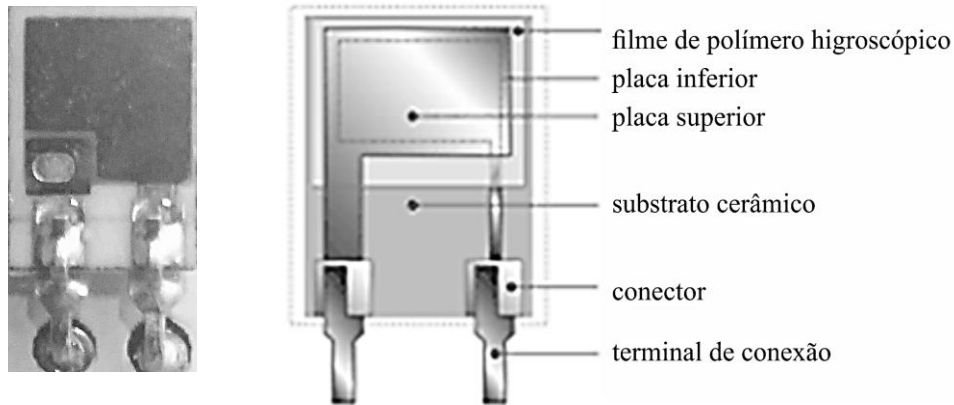


Figura 36 – Fotografia e esquema do modelo de sensor capacitivo para a medição de umidade relativa

Outra forma de realizar a medição da umidade relativa do ar é com o uso de um sensor do tipo resistivo, como utilizado no DHT11. Estes sensores são formados por uma base cerâmica, um condutor na forma de pente de filme espesso de metais preciosos como, por exemplo, o ouro, formando o eletrodo. Sob este eletrodo, é depositado um filme polimérico higroscópico que atua como uma película de detecção de umidade, devido à existência dos ions livres há a alteração da impedância do eletrodo, a partir da variação da umidade relativa do ar. Com a finalidade de melhorar o tempo de resposta, sensores deste tipo são equipados com um micro aquecedor resistivo para uma recuperação mais rápida da condensação. Este sensor permite medições no intervalo de 20 a 90 % de RH com acuidade de  $\pm 5\%$  de RH.

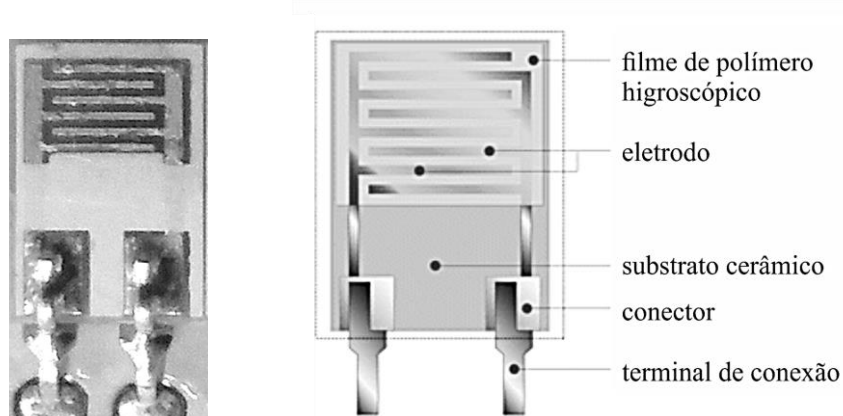


Figura 37 – Fotografia e esquema do modelo de sensor resistivo para a medição de umidade relativa



O termistor utilizado para a medição da temperatura do ar é fabricado a partir de materiais semicondutores sensíveis à temperatura, ou seja, materiais que tem a sua resistência elétrica alterada a partir de variações da temperatura, apresentando um valor definido de resistência elétrica para cada temperatura absoluta. Existem dois tipos básicos de termistores: o termistor PTC (Positive Temperature Coefficient – Coeficiente de Temperatura Positivo), que aumenta a sua resistência elétrica com o aumento da temperatura e o termistor NTC (Negative Temperature Coefficient – Coeficiente de Temperatura Negativo), que diminui a sua resistência elétrica com o aumento da temperatura. As curvas de resposta tanto do PTC quanto do NTC não são lineares. No AOSONG AM2302 é usado o NTC e sua linearização é realizada, via software, pelo microcontrolador de 8 bits presente no módulo.

Tensão de alimentação	3.3 – 6 V DC	
Grandeza medida	Umidade relativa do ar	Temperatura
Elemento do sensor	Capacitivo	Termistor do tipo NTC
Amplitude de medição	0 a 99.9% RH	-40 a +80°C
Acurácia	± 2% RH	± 0.5°C
Resolução	0.1% RH	0.1°C

Tabela 28 – Características básicas do sensor AOSONG AM2302  
 Fonte: Folha de dados do AM2302 da Aosong (Guangzhou) Electronics Co.,Ltd.

Para a medição de luminosidade, foi utilizado um LDR (Light Dependent Resistor ou Resistor que Depende da Luz) conectado a uma das entradas analógicas do *Arduino*. O LDR ou foto-resistência é um componente cujo valor nominal da resistência elétrica é função da intensidade da radiação eletromagnética incidente. A foto-resistência é geralmente construída com base em materiais semicondutores, como silício, germânio, arsênio, telúrio e compostos de cádmio e de chumbo. Estes materiais possuem a densidade de portadores livres na banda de condução como uma função, entre outras, da intensidade e do comprimento de onda dos fótons incidentes. Atualmente, existem no mercado foto-resistências que cobrem as gamas de radiação eletromagnética infravermelha, visível e ultravioleta.

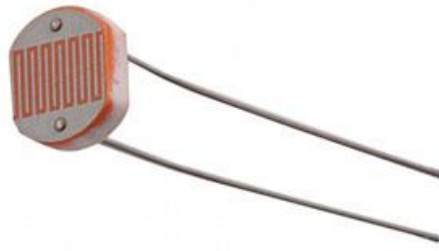


Figura 38 – Fotografia do LDR

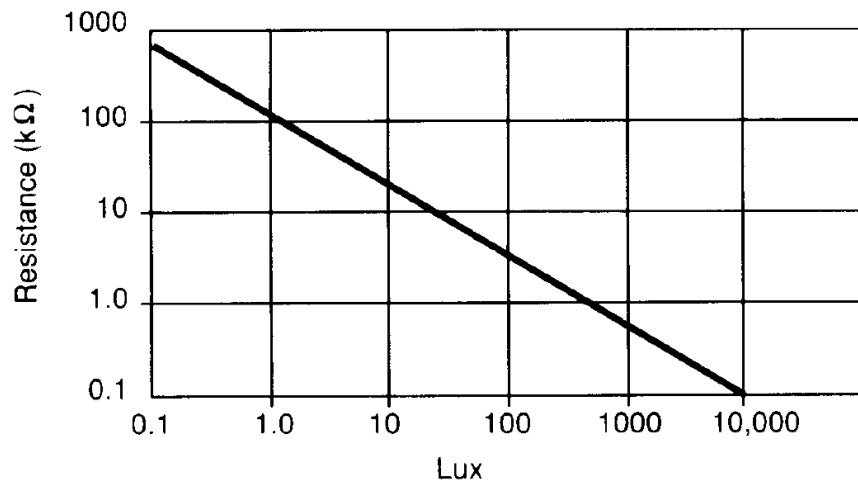


Figura 39 – Curva de resposta do LDR (Resistência versus Lux)  
 Fonte: Folha de dados do LDR da SUNROM Technologies

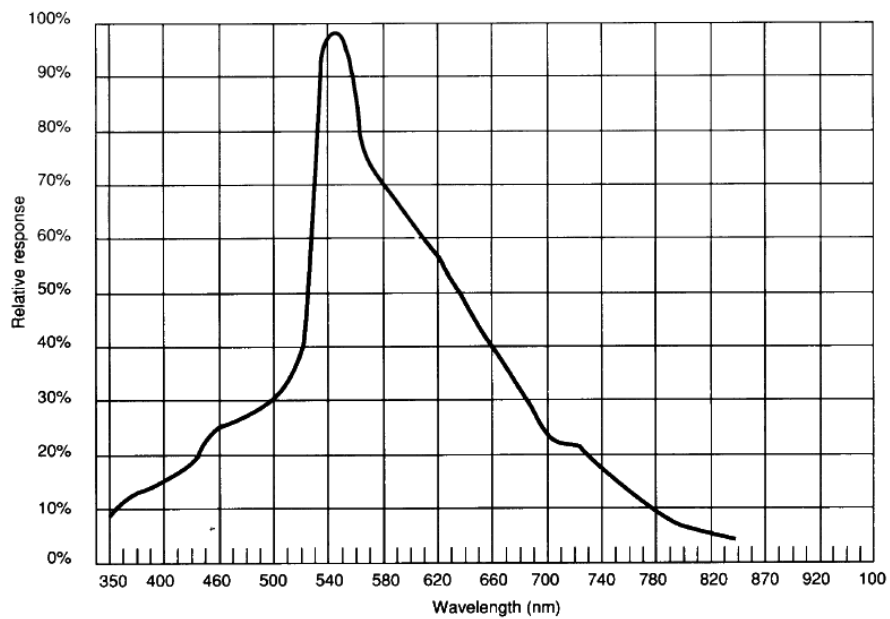


Figura 40 – Curva de resposta espectral do LDR  
 Fonte: Folha de dados do LDR da SUNROM Technologies

Para que o experimento, de fato, se torne matematicamente verificável e fisicamente significativo é importante que ele apresente os dados das variáveis medidas em unidades do Sistema Internacional e calibrados a um padrão, mesmo que este padrão não apresente a rastreabilidade metrológica desejada.

Para a pressão atmosférica, por não termos acessíveis medidores manuais e por esta grandeza depender, basicamente, da altitude da medição foi realizado um procedimento de comparação com base nas estações meteorológicas do Sistema Alerta Rio da Prefeitura do Rio de Janeiro, localizada no Campo de São Cristóvão; do Instituto Nacional de Meteorologia - INMET, localizada no Forte de Copacabana e do Instituto Nacional de Pesquisas Espaciais – INPE, localizada no Aeroporto Santos Dummont.

	Estação com o uso do <i>Arduino</i> e <i>Processing</i>	Estação do Alerta Rio	Estação do INMET	Estação do INPE
Localização	Escola Técnica Estadual Ferreira Viana	Campo de São Cristóvão	Forte de Copacabana	Aeroporto Santos Dummont
Latitude	- 22,913932	- 22,896667	- 22,988286	- 22,91038
Longitude	- 43,226412	- 43,221667	- 43,190436	- 43,16665
Altitude	14 m (estimado) <sup>26</sup>	35 m (estimado) <sup>27</sup>	26 m (informado) <sup>28</sup>	3 m (informado) <sup>29</sup>
Distância	---	1,9809 km	9,0538 km	6,1354 km
Direção	---	14,21°	155,98°	86,31°

Tabela 29 – Dados da localização geográfica e da altitude das Estações Meteorológicas

O procedimento baseia-se na comparação da média obtida a partir da coleta de dados nas estações com o valor obtido matematicamente, tendo como referência o valor da pressão atmosférica ao nível do mar de 1016 hPa, obtido na carta de pressão ao nível do mar emitida pela Marinha do Brasil em 19/04/2016 às 12:00 horas. Para cada estação meteorológica, foram realizadas cinco medições próximas ao horário referenciado na carta e calculada a média da pressão atmosférica em cada uma das estações.

<sup>26</sup> Altitude estimada com base no valor de 13 m de altitude do solo obtido pelo google Earth acessado em 20/04/2016.

<sup>27</sup> Altitude estimada com base no valor de 35 m de altitude do solo obtido pelo google Earth acessado em 20/04/2016.

<sup>28</sup> Altitude de 26 m informada pelo INMET no endereço <http://www.inmet.gov.br> no link estações automáticas, acessado em 20/04/2016.

<sup>29</sup> Altitude informada no INPE no endereço <http://bancodedados.cptec.inpe.br/downloadBDM/> acessado em 20/04/2016.

Horário de medição	Estação com o uso do <i>Arduino</i> e <i>Processing</i>	Estação do Alerta Rio	Estação do INMET	Estação do INPE
19/04/2016 às 10:00	1015,82	1012,8	1012,3	1015,93
19/04/2016 às 11:00	1015,27	1012,6	1012,7	1015,93
19/04/2016 às 12:00	1014,70	1011,9	1013,2	1016,94
19/04/2016 às 13:00	1013,92	1011,1	1013,3	1016,94
19/04/2016 às 14:00	1013,20	1010,0	1013,0	1015,93
Valor médio	1014,58	1011,7	1012,9	1016,33

Tabela 30 – Dados referentes a Pressão atmosférica coletados nas quatro estações meteorológicas usadas como referência para calibração do sensor BMP180

O cálculo da variação da pressão atmosférica como função da altitude pode ser realizado utilizando a equação barométrica. Com o intuito de simplificar os cálculos para o uso em sala de aula, nas turmas de Ensino Médio, devemos considerar algumas generalizações: a atmosfera é um gás ideal em equilíbrio, com temperatura constante e a aceleração da gravidade é constante em toda a atmosfera. Assim, podemos escrever a equação barométrica<sup>30</sup>:

$$P(h) = P_{\text{nível do mar}} \cdot e^{-\frac{\rho_{\text{ar}}}{P_{\text{nível do mar}}}gh}$$

Considerando  $\rho_{\text{ar}} = 1,18 \text{Kg/m}^3$ ,  $P_{\text{nível do mar}} = 1,016 \times 10^5 \text{N/m}^2$  (como indicado na carta sinótica de 19/04/2016 às 12:00) e  $g = 9,81 \text{ m/s}^2$ , temos:

$$\left(\frac{\rho_{\text{ar}}}{P_{\text{nível do mar}}}\right)g = \left(\frac{1,18 \frac{\text{Kg}}{\text{m}^3}}{1,016 \times 10^5 \frac{\text{N}}{\text{m}^2}}\right)\left(9,81 \frac{\text{m}}{\text{s}^2}\right) = 1,139 \times 10^{-4} \text{m}^{-1}$$

Portanto, podemos calcular a pressão atmosférica para cada uma das estações:

$$P(3\text{m}) = 1,016 \times 10^5 \frac{\text{N}}{\text{m}^2} \cdot e^{-1,139 \times 10^{-4} \text{m}^{-1} \cdot 3\text{m}} = 1015,653 \text{ hPa}$$

$$P(16\text{m}) = 1,016 \times 10^5 \frac{\text{N}}{\text{m}^2} \cdot e^{-1,139 \times 10^{-4} \text{m}^{-1} \cdot 16\text{m}} = 1014,150 \text{ hPa}$$

$$P(26\text{m}) = 1,016 \times 10^5 \frac{\text{N}}{\text{m}^2} \cdot e^{-1,139 \times 10^{-4} \text{m}^{-1} \cdot 26\text{m}} = 1012,996 \text{ hPa}$$

$$P(35\text{m}) = 1,016 \times 10^5 \frac{\text{N}}{\text{m}^2} \cdot e^{-1,139 \times 10^{-4} \text{m}^{-1} \cdot 35\text{m}} = 1011,958 \text{ hPa}$$

<sup>30</sup> BAUER, W.; WESTFALL, G.D.; DIAS, H. **Física para Universitários – Relatividade, Oscilações, Ondas e Calor**. São Paulo: Mc Graw Hill, 2013, p. 17.

	Estação com o uso do <i>Arduino e Processing</i>	Estação do Alerta Rio	Estação do INMET	Estação do INPE
Pressão média	1014,58 hPa	1011,7 hPa	1012,9 hPa	1016,33 hPa
Pressão calculada	1014,150 hPa	1011,958 hPa	1012,996 hPa	1015,653 hPa
Erro relativo	0,0424%	0,0255%	0,0095%	0,0667%

Tabela 31 – Quadro comparativo do erro relativo calculado para os dados referentes à pressão atmosférica obtidos nas quatro estações meteorológicas em análise

Como podemos observar, na apresentação do erro relativo a medição de pressão atmosférica a partir do sensor BMP180 está bem próxima aos padrões de erro das estações meteorológicas comerciais.

Para a verificação das medições de temperatura pelo sensor BMP180 e pelo sensor AOSONG AM2302, foi utilizado o termo higrômetro digital modelo MT-240 da Minipa, que apresenta as características técnicas para temperatura:

Termo higrômetro MT-240	
Grandeza medida	Temperatura
Amplitude de medição	-50 a +70°C
Precisão	$\pm 2^{\circ}\text{C}$
Resolução	0,1°C

Tabela 32 – Características básicas da medição de temperatura do Termo higrômetro MT-240  
Fonte: Manual de instruções do Termo higrômetro digital MT-240 da Minipa Indústria e Comércio Ltda

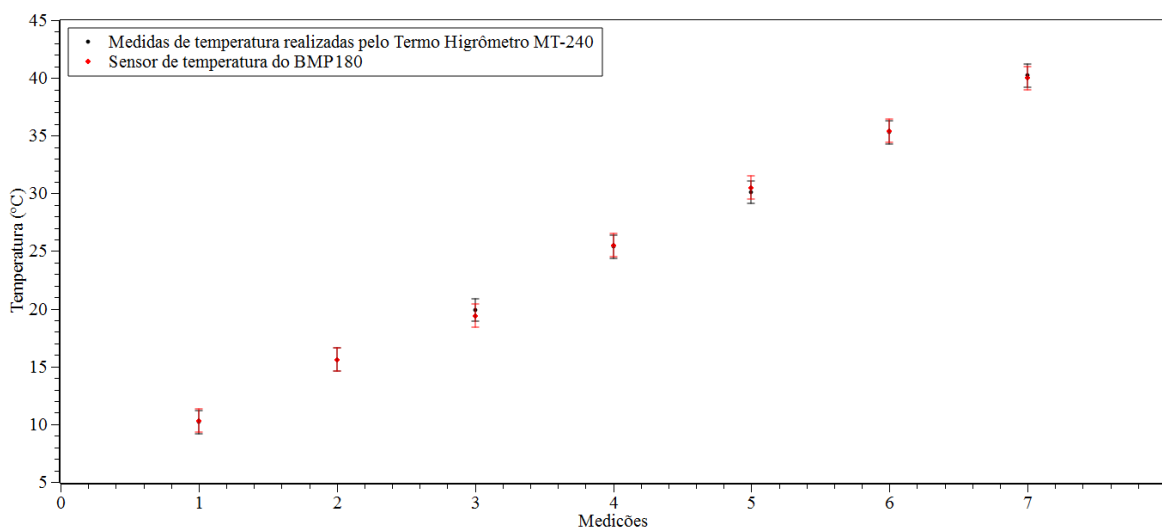


Gráfico 6 – Comparação entre os dados de temperatura obtidos pelo sensor BMP180 e pelo Termo higrômetro MT-240

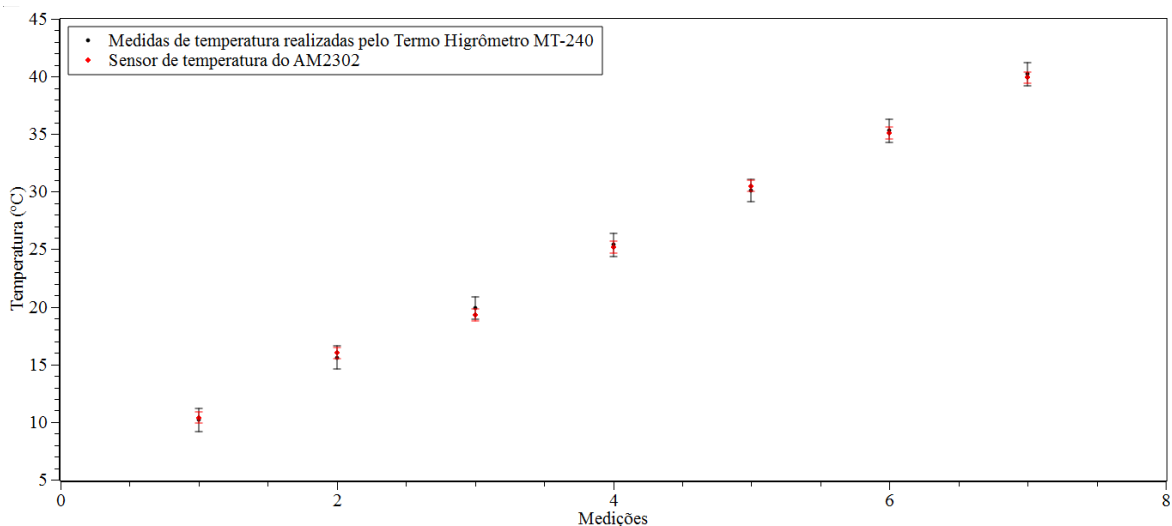


Gráfico 7 – Comparação entre os dados de temperatura obtidos pelo sensor AM2302 e pelo Termo higrômetro MT-240

O valor médio do erro de leitura obtido a partir das medições de temperatura do ar, realizadas, considerando como referência o valor medido pelo higrômetro digital modelo MT-240, é inferior a precisão do instrumento, sendo assim o valor obtido pelos sensores BMP180 e AOSONG AM2302 podem ser considerados como fisicamente significativos. Durante as medições foi constatada que a velocidade de resposta às variações de temperatura do BMP180 é consideravelmente maior que a do AM2302.

Valor de referência MT-240	Valor medido BMP180	Erro BMP180	Valor medido AM2302	Erro AM2302
10,2	10,3	0,1	10,4	0,2
15,6	15,6	0,0	16	0,4
19,9	19,4	0,5	19,3	0,6
25,4	25,5	0,1	25,2	0,2
30,1	30,5	0,4	30,5	0,4
35,3	35,4	0,1	35,1	0,2
40,2	40	0,2	39,9	0,3
Erro médio		0,2	---	0,33

Tabela 33 – Quadro comparativo do erro calculado para os dados referentes à temperatura obtidos nos sensores BMP180, AM2302 e o Termo higrômetro MT-240

Para a verificação das medições de umidade relativa do ar pelo sensor AOSONG AM2302, também foi utilizado o termo higrômetro digital modelo MT-240 da Minipa, que apresenta as características técnicas para umidade relativa do ar:

Termo higrômetro MT-240	
Grandeza medida	Umidade Relativa do ar
Amplitude de medição	20 a 90% RH
Precisão	$\pm 5\%$ RH
Resolução	1 % RH

Tabela 34 – Características básicas da medição de umidade relativa do ar do Termo higrômetro MT-240  
Fonte: Manual de instruções do Termo higrômetro digital MT-240 da Minipa Indústria e Comércio Ltda

Cabe ressaltar que os dados do fabricante do Termo higrômetro MT-240 referenciam a precisão no valor de 5% de umidade relativa do ar e não em 5% do valor obtido na medição.

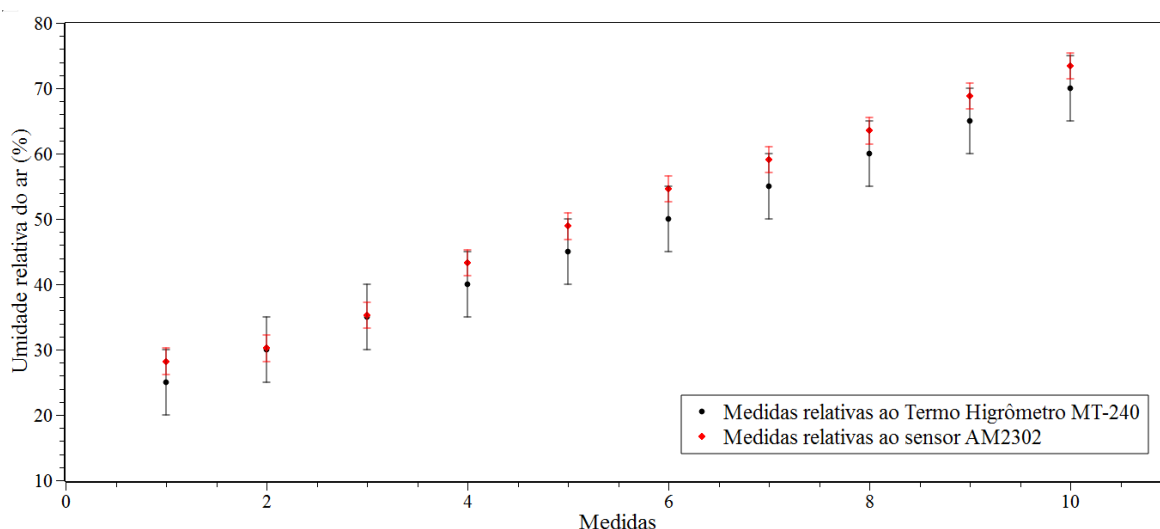


Gráfico 8 – Comparação das medidas atinentes a umidade relativa do ar obtidas a partir do sensor AM2302 e do Termo higrômetro MT-240

Valor de referência MT-240 (% RH)	Valor medido AM2302 (%RH)	Erro (%RH)
$25 \pm 5$	28,2	3,9
$30 \pm 5$	30,2	0,2
$35 \pm 5$	35,3	0,3
$40 \pm 5$	43,3	3,3
$45 \pm 5$	48,9	3,9
$50 \pm 5$	54,6	4,6
$55 \pm 5$	59,1	4,1
$60 \pm 5$	63,5	3,5
$65 \pm 5$	68,8	3,8
$70 \pm 5$	73,4	3,4
Erro de leitura médio		3,03

Tabela 35 – Determinação do erro de leitura da umidade relativa do ar obtida a partir do sensor AM2302 e comparada com o Termo higrômetro MT-240

O valor médio do erro de leitura obtido a partir das medições de umidade relativa do ar, realizadas, considerando como referência o valor medido pelo higrômetro digital modelo MT-240, é inferior a precisão do instrumento, sendo assim o valor obtido pelo sensor AOSONG AM2302 é fisicamente significativo.

Da mesma forma, para que a aquisição de dados do sensor de luminosidade, o LDR, também apresente um significado físico, é necessária a correlação da variação de resistência do LDR, percebida pelo *Arduino* em forma de tensão em sua entrada analógica, em uma grandeza de luminosidade, no caso o lux. Para alcançar este objetivo, realizaremos esta comparação com o uso do luxímetro digital modelo MLM-1011 fabricado pela Minipa. Este instrumento apresenta as seguintes características técnicas:

Luxímetro MLM-1011	
Grandeza medida	Umidade Relativa do ar
Amplitude de medição	1 lux a 100000 lux
Faixas de medição	2000, 20000 (leitura x10) e 100000 lux (leitura x100)
Precisão	$\pm 4\%$ Calibrado com o padrão de lâmpada incandescente 2856K.
Repetibilidade	$\pm 2\%$
Resolução	1 lux, 10 lux e 100 lux.

Tabela 36 – Características básicas da medição de luminosidade, em lux, do Luxímetro MLM-1011  
Fonte: Manual de instruções do Luxímetro Digital da Minipa Indústria e Comércio Ltda

No intervalo de operação do LDR, entre 0 e 5 volts ou entre os níveis digitais entre 0 e 1023 ( $2^0$  e  $2^{10}$ ) após a digitalização realizada pelo *Arduino*, foram realizadas 3 medições com o LDR para cada valor medido com o luxímetro do iluminamento, gerado por um aparato experimental baseado em uma lâmpada incandescente (12V 24w) e uma fonte de tensão variável (0 a 12 VDC). A partir das três medições, foi calculada a média aritmética para a construção do gráfico da função de resposta do sensor.



Lux	Nível após digitalização Medição 1	Nível após digitalização Medição 2	Nível após digitalização Medição 3	Nível após digitalização Média
25	312	309	305	308,7
50	445	446	443	444,7
75	555	552	554	553,7
100	611	611	610	610,7
250	787	788	787	787,3
500	875	873	873	873,7
750	909	908	908	908,3
1000	927	929	928	928,0
2500	969	971	969	969,7
5000	987	988	987	987,3
7500	995	994	994	994,3
10000	1001	1001	1001	1001,0
25000	1010	1009	1010	1009,7

Tabela 37 – Dados da resposta da digitalização da tensão presente no LDR pelo *Arduino* para diferentes níveis de iluminação

Após a determinação da média do nível de digitalização foi realizada a conversão deste nível em tensão elétrica.

$$V_{1bit} = \frac{V_{Total}}{2^{\text{número de bits}} - 1} = \frac{5,0}{2^{10} - 1} = \frac{5,0}{1023} = 4,888 \text{ mV}$$

$$V = \text{valor médio após digitalização} \times V_{1bit}$$

Para a conversão da tensão sobre o LDR em resistência, foi utilizada a equação para o divisor de tensão resistivo. Na montagem deste circuito na estação meteorológica, foi utilizado um resistor de 10kΩ em série com o LDR.

$$V = \frac{R}{R + R_{LDR}} V_{Total}$$

$$R_{LDR} = \frac{R \times V_{Total}}{V}$$

Nível após digitalização Média	Tensão calculada (volts)	Resistência Calculada (ohms)
308,7	1,5086	23142,5
444,7	2,1733	13006,0
553,7	2,7061	8476,8
610,7	2,9847	6752,2
787,3	3,8482	2993,2
873,7	4,2701	1709,3
908,3	4,4396	1262,4
928,0	4,5357	1023,7
969,7	4,7393	550,0
987,3	4,8257	361,2
994,3	4,8599	288,3
1001,0	4,8925	219,8
1009,7	4,9348	132,1

Tabela 38 – Valores de tensão e resistência calculadas a partir dos Dados da resposta da digitalização da tensão presente no LDR pelo *Arduino*

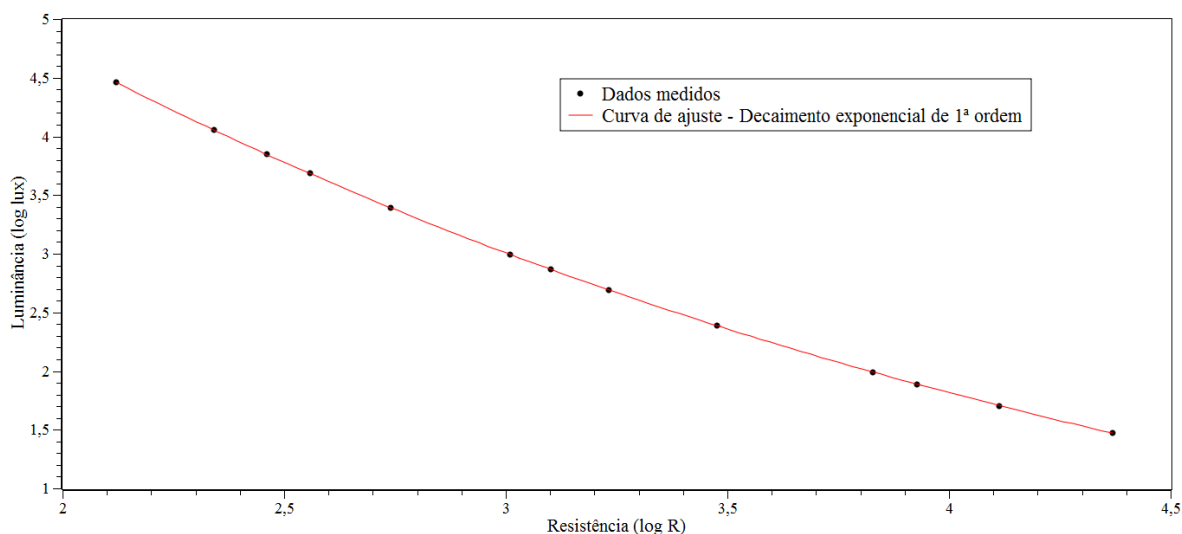


Gráfico 9 – Gráfico da variação resistiva do LDR para diversos níveis de iluminação

A Função de linearização obtida para o LDR é uma exponencial de decaimento de 1ª ordem:  $y = 1,026 + 11,535 \times e^{-\frac{x}{2,855}}$ , sendo y o log do valor de resistência em ohms calculado a partir do nível de tensão digitalizado em 10 bits, e x o valor do log da luminosidade medida pelo luxímetro em lux.

Com a definição da função de resposta do sensor, é preciso determinar o erro relativo para que seja possível a avaliação da qualidade do processo de linearização do sensor. Para

alcançar este objetivo, foi aplicada a equação matemática a cada um dos valores obtidos na digitalização da resposta direta do sensor e comparado com o resultado da medição de luminosidade, obtendo-se o erro relativo a cada uma das medições e o erro relativo médio de 5,54% para a faixa de 25 a 25000 lux e de 2,11% para a faixa de 50 a 5000 lux.

Nível de iluminamento em lux	Nível do LDR após digitalização (Média)	Nível de iluminamento obtida após linearização	Erro Relativo
25	308,7	29,5	18,13%
50	444,7	50,6	1,20%
75	553,7	77,2	2,99%
100	610,7	97,8	2,18%
250	787,3	244,2	2,31%
500	873,7	491,5	1,70%
750	908,3	736,0	1,86%
1000	928,0	984,0	1,60%
2500	969,7	2462,2	1,51%
5000	987,3	4819,9	3,60%
7500	994,3	7037,4	6,17%
10000	1001,0	11292,7	12,93%
25000	1009,7	28972,9	15,89%
Erro relativo médio na faixa de 50 a 5000 lux			2,11%
Erro relativo médio na faixa de 25 a 25000 lux			5,54%

Tabela 39 – Determinação do erro relativo no processo de linearização do LDR

Após a conclusão da rotina de testes e de comparação dos sensores, inicia-se uma rotina dinâmica de funcionamento da estação meteorológica com todos os seus sensores ligados e montados na caixa plástica. Foram realizadas medições por alguns dias para a comprovação de seu funcionamento. Os dados coletados pela estação desenvolvida são, para fins de análise, comparados qualitativamente aos dados coletados no mesmo período por uma estação meteorológica automática convencional mais próxima. Para a pressão atmosférica, temperatura e umidade relativa do ar a comparação foi realizada com a estação do Sistema Alerta Rio, localizada no Campo de São Cristóvão. Para luminosidade a comparação foi realizada com a estação do INMET localizada no Forte de Copacabana.

Na análise dos dados de temperatura, observa-se a presença de um ciclo diário, apresentando maiores temperatura durante o dia e menores temperaturas durante a noite.

Existe uma correlação direta entre os picos de temperatura com os picos de luminosidade, que representa a insolação ou a energia absorvida das radiações solares.

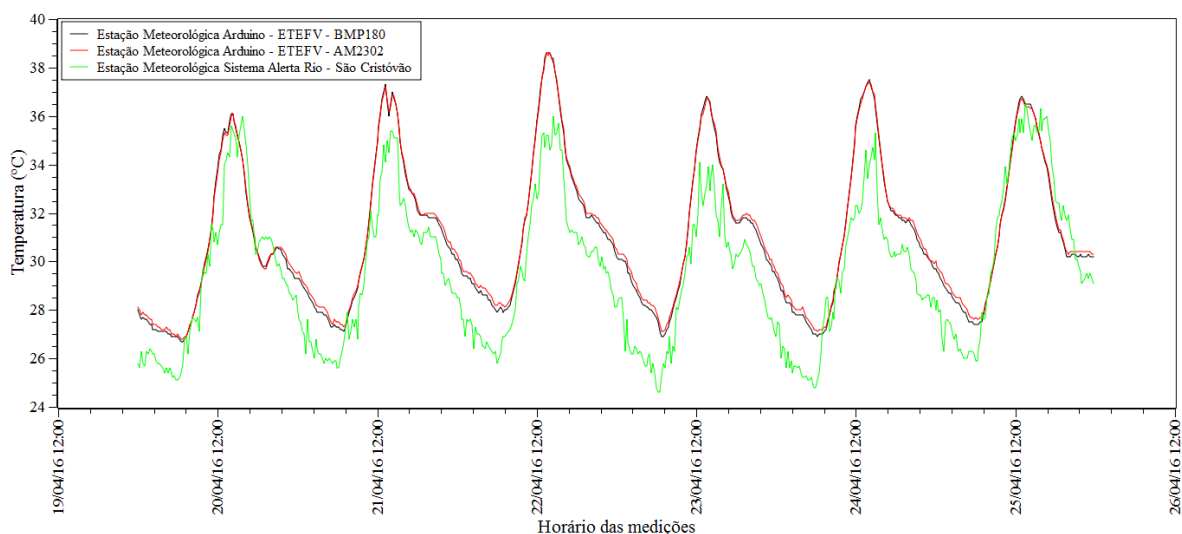


Gráfico 10 – Comparação entre os dados de temperatura coletados pela estação meteorológica com o uso o *Arduino* e do *Processing* e pela estação meteorológica do Sistema Alerta Rio de São Cristóvão

Ainda analisando os dados individuais de temperatura, nota-se a proximidade das leituras dos sensores BMP180 e AM2302 e uma pequena disparidade com o sensor da Estação Meteorológica do Sistema Alerta Rio de São Cristóvão. Esta disparidade pode ser explicada pela diferença de localização dos sensores em cada estação. A estação de São Cristóvão se localiza no alto de um prédio, sujeita a repentinas variações de direção e velocidade dos ventos, enquanto a estação da ETEFV se encontra em um local abrigado e menos sujeita a variações repentinas de temperatura.

O gráfico de luminosidade demonstra a alteração periódica da luminância entre o nascer do sol e o crepúsculo ou ocaso. Observa-se uma taxa de variação acentuada da luminosidade a partir do amanhecer, um pico próximo ao meio-dia e um decréscimo, também acentuado de luminosidade a partir deste ponto, até a escuridão total da noite. É possível notar interferências no sensor após o ocaso nos dias 20/04 e 25/04 geradas a partir de interferências luminosas externas. As variações repentinas, durante o dia, são consequência da presença ou passagem de nuvens na área de visada do sensor. São apresentados dois gráficos da luminosidade, sendo que o primeiro expressa a luminosidade em lux e o segundo expressa a luminosidade em níveis digitais de 10 bits oriundos da aquisição da tensão sobre o LDR pelo *Arduino*.

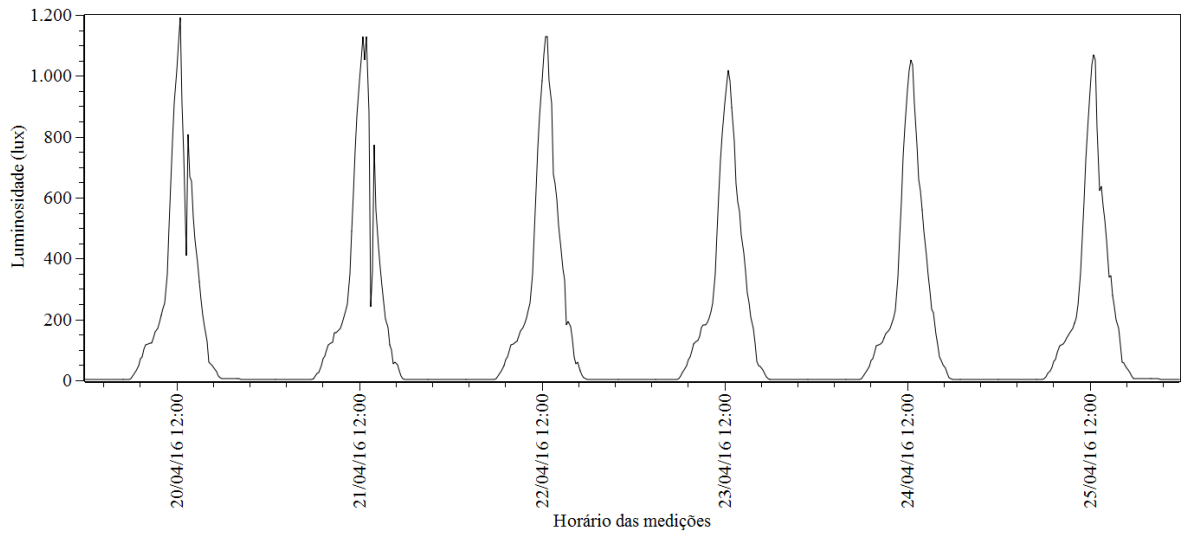


Gráfico 11 – Gráfico dos dados referentes à luminosidade, representado em lux, obtidos da estação meteorológica com o uso do *Arduino* e do *Processing*

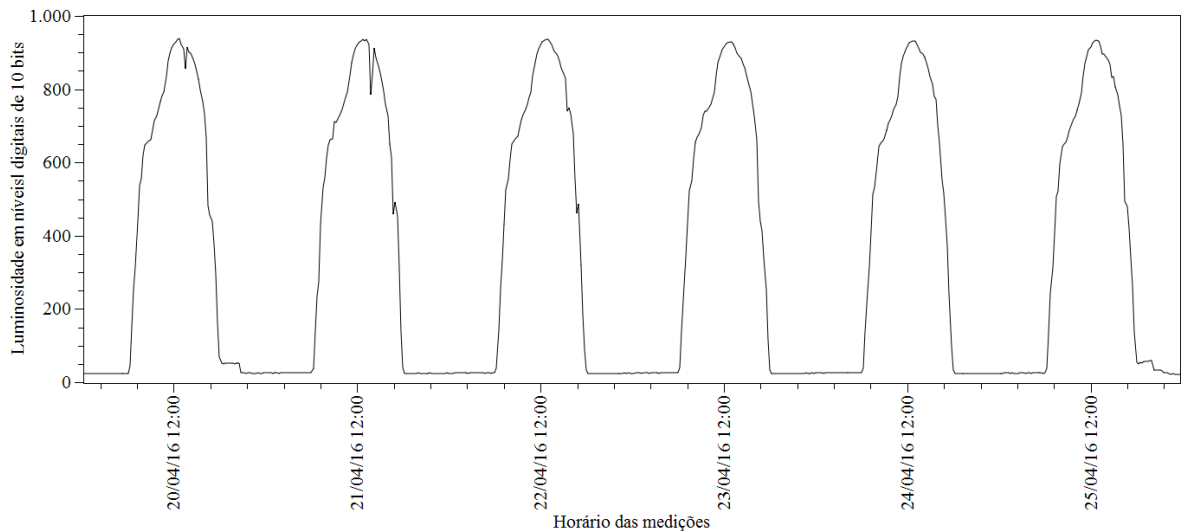


Gráfico 12 – Gráfico dos dados referentes à luminosidade, representado em níveis digitais de 10 bits, obtidos da estação meteorológica com o uso do *Arduino* e do *Processing*

Quando o período observacional se restringe a 24 horas o gráfico da luminosidade se apresenta de forma mais ampla e menos homogênea devido as variações de luminosidade, que deixam de ser pontuais e passam a se manifestar como deformações na tendência natural das taxas de incremento ou decremento da luminosidade no transcorrer do dia. Estas deformações são provocadas, principalmente, pela nebulosidade do dia ou pela passagem de nuvens. Neste gráfico também foram inseridas as posições efetivas do nascer do sol e do ocaso, que no dia 22/04/2016 ocorreram, respectivamente, às 6:09 e às 17:35.

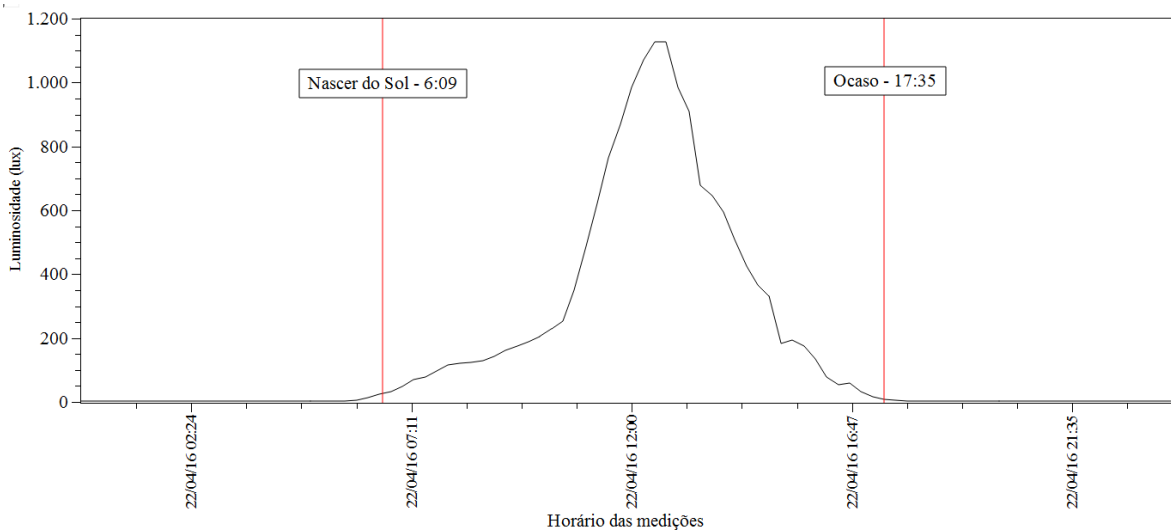


Gráfico 13 – Gráfico dos dados referentes à luminosidade do dia 22/04/2016, representado em lux, obtidos da estação meteorológica com o uso do *Arduino* e do *Processing*

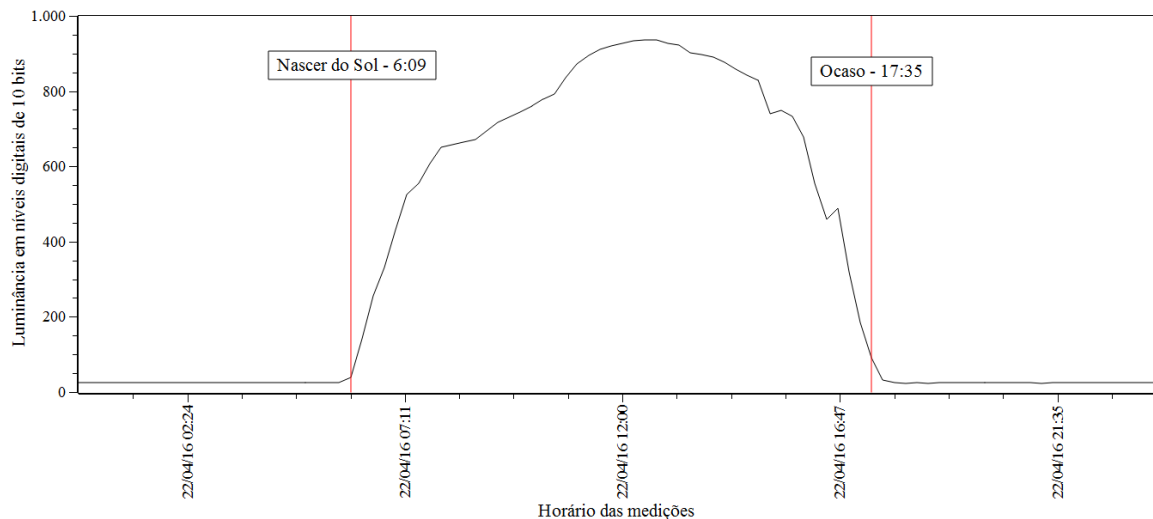


Gráfico 14 – Gráfico dos dados referentes à luminosidade do dia 22/04/2016, representado em níveis digitais de 10 bits, obtidos da estação meteorológica com o uso do *Arduino* e do *Processing*

Os dados de radiação solar coletados pela estação do INMET para o mesmo período 20 a 25/04/2016 são apresentados no gráfico a seguir. Cabe ressaltar que os dados disponibilizados<sup>31</sup> pelo INMET são no padrão horário. Podemos observar que o sinal é similar ao obtido pelo sensor de luminosidade, diferenciando apenas no horário de pico da radiação, que melhor pode ser visualizado quando o período observacional se restringe a 24 horas.

<sup>31</sup>Disponível em: [http://www.inmet.gov.br/sonabra/maps/pg\\_automaticas.php](http://www.inmet.gov.br/sonabra/maps/pg_automaticas.php). Acessado em 01/04/2016.

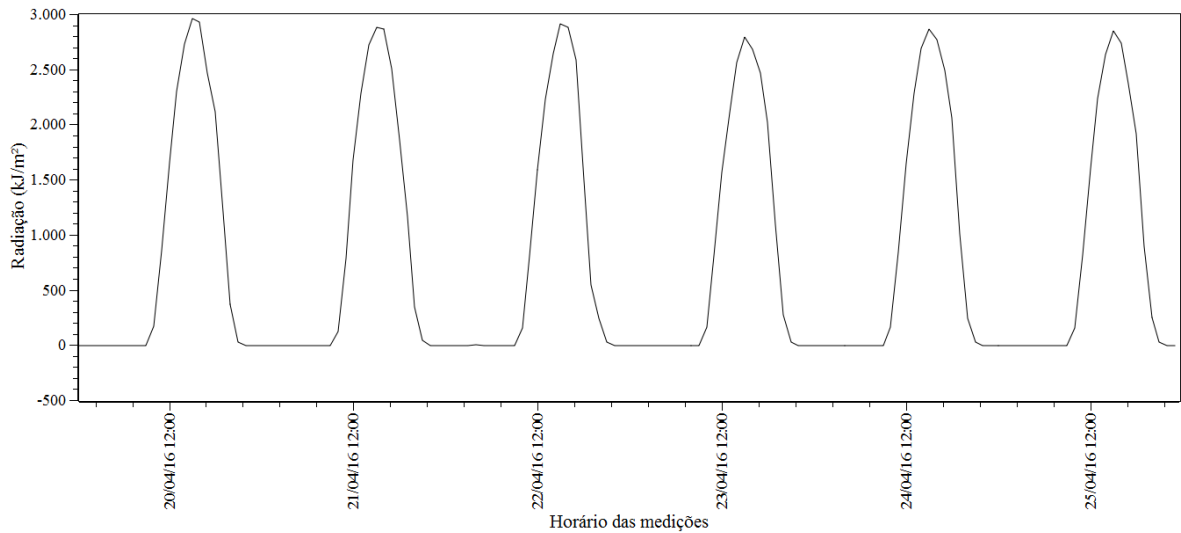


Gráfico 15 – Gráfico dos dados referentes à luminosidade, representado em kJ/m<sup>2</sup>, obtidos da estação meteorológica do INMET localizada no Forte de Copacabana

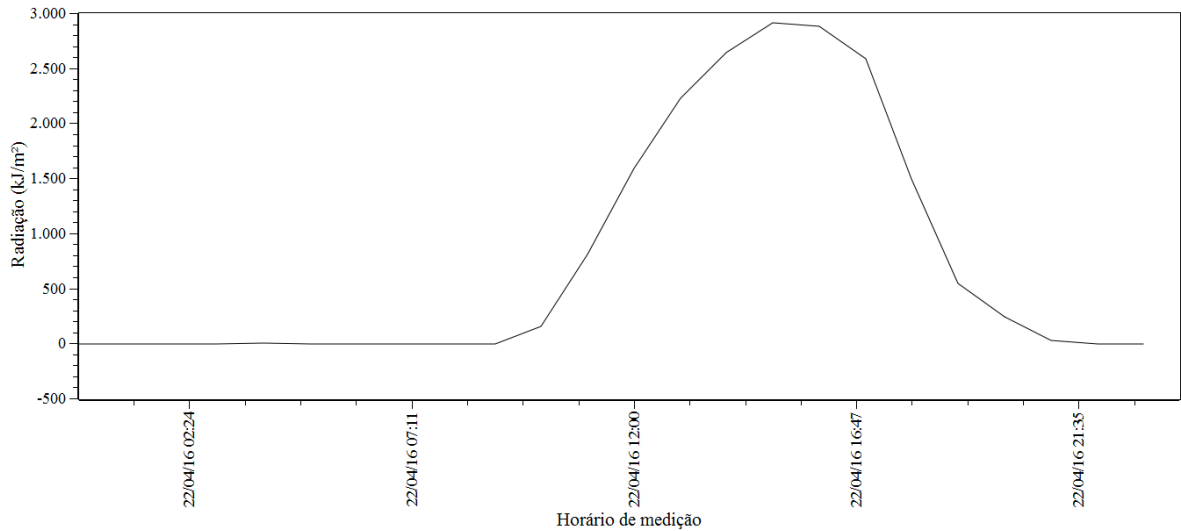


Gráfico 16 – Gráfico dos dados referentes à luminosidade no dia 22/04/2016, representado em kJ/m<sup>2</sup>, obtidos da estação meteorológica do INMET localizada no Forte de Copacabana

Analisando o gráfico de umidade relativa do ar, é possível perceber a formação de ciclos diários com medições de umidade maiores à noite e menores durante o dia. Em comparação aos dados da Estação Meteorológica do Sistema Alerta Rio de São Cristóvão, os dados da estação com *Arduino* estão coerentes e dentro da variação máxima de amplitude. A disparidade encontrada nas medições pode ser explicada pela diferença de localização dos sensores de cada estação. A estação da ETEFV se encontra em um local abrigado e menos sujeito a variações repentinas de temperatura, formando um gráfico mais comportado e com menor amplitude.

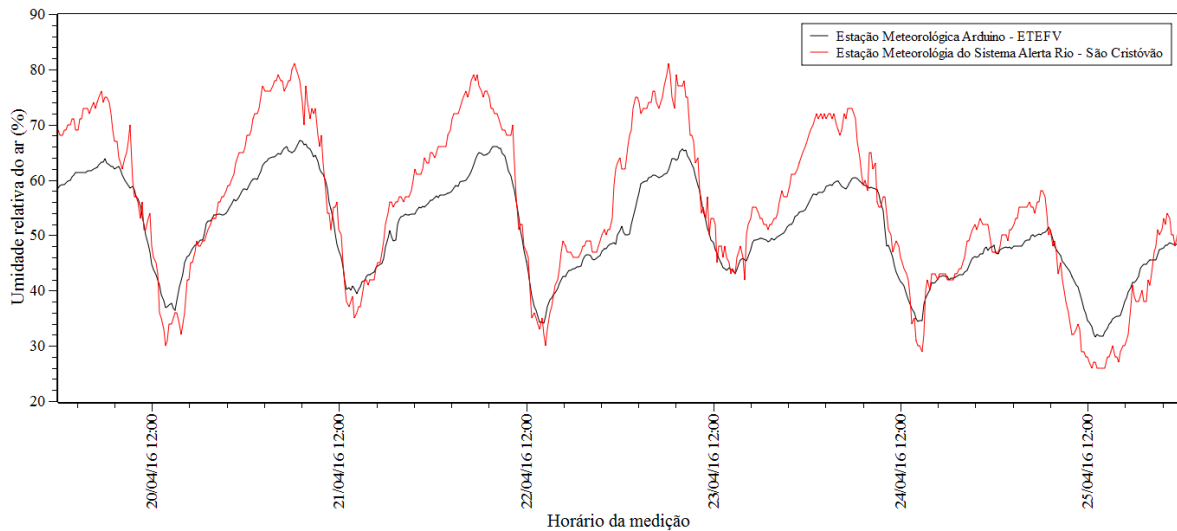


Gráfico 17 – Comparação entre os dados de umidade relativa do ar coletados pela estação meteorológica com o uso do *Arduino* e do *Processing* e pela estação meteorológica do Sistema Alerta Rio de São Cristóvão

Em comparação aos dados da Estação Meteorológica do Sistema Alerta Rio de São Cristóvão, os dados referentes à pressão atmosférica da estação com *Arduino* são semelhantes, porém com uma pequena discrepância em nível, oriunda da diferença de altitude entre as estações. Analisando o gráfico da pressão atmosférica, diferentemente das outras grandezas, não se observa a formação de um único ciclo diário. A cada ciclo de 24 horas, é possível notar a formação de dois picos e dois vales, ou seja, há dois ciclos de variação da pressão por dia. Este fenômeno é conhecido como maré barométrica ou maré atmosférica.

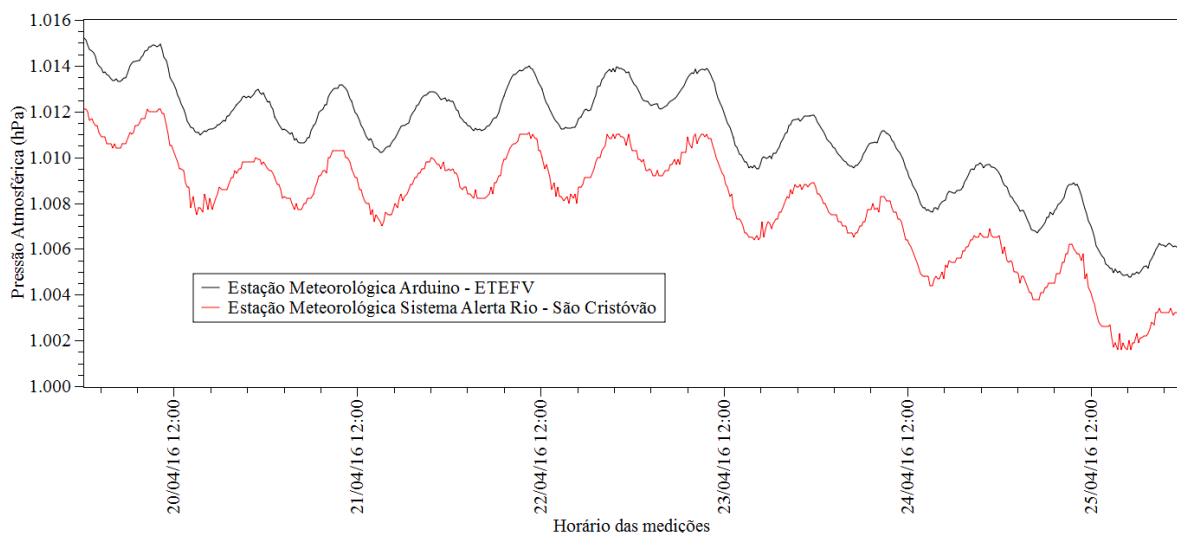


Gráfico 18 – Comparação entre os dados de pressão atmosférica coletados pela estação meteorológica com o uso do *Arduino* e do *Processing* e pela estação meteorológica do Sistema Alerta Rio de São Cristóvão

Os estudos desenvolvidos, sobre este tema, concluem que as oscilações barométricas não são de origem gravitacional, mas fundamentalmente de origem térmica, devido ao



aquecimento da atmosfera produzido pela radiação solar. O efeito gravitacional está presente, mas é muito menor que o efeito térmico, como apontam Luiz Raimundo Moreira de Carvalho e Hélio Salim de Amorim em seu artigo, *Observando as marés atmosféricas: Uma aplicação da placa Arduino com sensores de pressão barométrica e temperatura* (2014).

O fenômeno da oscilação barométrica sem relação direta com a altitude é um assunto pouco abordado e, por consequência, pouco discutido nos cursos de Física, tanto no Ensino Médio quanto nas Graduações. Isto ocorre devido à complexidade dos seus modelos explicativos. A alta sensibilidade do sensor BMP180 aliado a automação de medições propiciada pelo *Arduino* e pelo *Processing* possibilita a fácil verificação deste fenômeno, contribuindo para a difusão deste conhecimento entre nossos alunos.

Já a interface do experimento, desenvolvida em *Processing*, recebe os dados via conexão serial (USB) do *Arduino* e os exibe, em tempo real, de forma textual, apresentando valores instantâneos, máximos, médios e mínimos de cada uma das grandezas medidas, bem como expando, em forma de gráficos, as variações temporais das grandezas. Além dessas funcionalidades, são gerados arquivos no formato .txt contendo todos os dados obtidos pela estação meteorológica em um dia, para possível análise posterior. Para esta análise sugerimos softwares como o *SCIDavis* ou planilhas como o *Libreoffice Calc*.

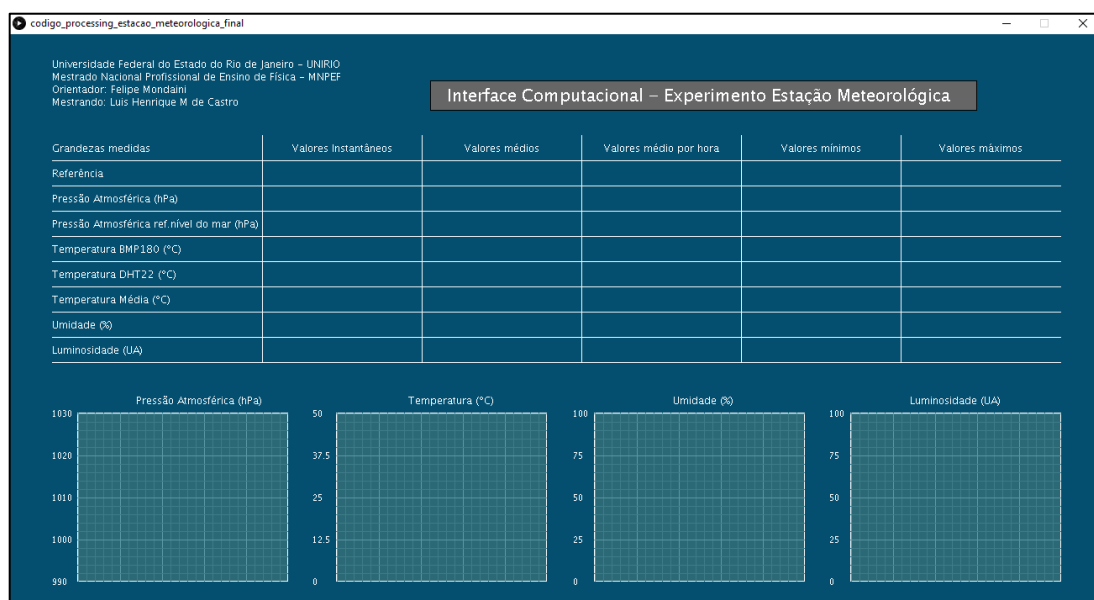


Figura 41 – Interface gráfica desenvolvida com o uso do *Processing* para a apresentação dos dados meteorológicos adquiridos pelo *Arduino*

O uso desse experimento, objetivamente, favorece o desenvolvimento das habilidades e das competências preconizadas pelos descritores, presentes na Base Nacional

Comum e no Currículo Mínimo do Estado do Rio de Janeiro em diversos ramos da física como: termodinâmica, eletricidade, conservação e transformação de energia, energia nuclear, etc. Esses descritores estão relacionados na tabela a seguir:

Termometria, Calor, Pressão e Energia
Base Nacional Comum
Identificar processos de trocas térmicas e mudanças de temperatura, como condução, convecção e irradiação, em ciclos naturais e equipamentos como fornos, refrigeradores e caldeiras, representando esses processos em diagramas e modelos cinéticos.
Analisar fontes de energia de grande escala, como petróleo, hidrelétricas e termelétricas, considerando seu sentido econômico e social, os impactos ambientais e climáticos, a viabilidade e custo de alternativas como biocombustíveis, usinas eólicas e coletores solares.
Compreender fenômenos naturais ou sistemas tecnológicos, identificando e relacionando as grandezas envolvidas
Compreender a diferença entre temperatura e calor a partir do modelo atomista da matéria.
Currículo Mínimo do Estado do Rio de Janeiro
Reconhecer que trabalho e calor são diferentes formas de transferência de energia. (Termodinâmica)
Reconhecer, utilizar, interpretar e propor modelos explicativos para fenômenos naturais ou sistemas tecnológicos.
Compreender fenômenos naturais ou sistemas tecnológicos, identificando e relacionando as grandezas envolvidas.
Compreender as diferentes manifestações da energia mecânica na natureza. (Conservação e transformação de energia)
Analisar, argumentar e posicionar-se criticamente em relação a temas de ciência, tecnologia e sociedade.
Compreender que o Sol é a fonte primária da maioria das formas de energia de que dispomos. (Energia nuclear)
Identificar fenômenos e grandezas elétricas, estabelecer relações, identificar regularidades, invariantes e transformações. (Eletricidade)
Consultar, analisar e interpretar textos e símbolos referentes a representações técnicas.

Tabela 40 – Descritores relacionados a Termometria, Calor, Pressão e Energia

### Atividade1: Coletando as variáveis meteorológicas

Apresentar aos alunos a interface Processing do experimento, realizando a coleta dos dados de temperatura, pressão atmosférica, umidade relativa do ar e luminosidade. Solicitar aos alunos que realizem o registro destas medições, percebendo as interferências destas grandezas em seus sistemas sensoriais.

Repita este procedimento de coleta dos dados em outros dois momentos, enfatizando a comparação da percepção corporal destas grandezas no período das medições. Após o registro das três medições, calcular o valor médio e identificar a tendência de variação de cada uma das variáveis no período observacional.

### Habilidades e Competências trabalhadas:

- Reconhecer e saber utilizar corretamente símbolos, códigos e nomenclaturas de grandezas da Física, nas previsões climáticas, identificar temperaturas, pressão, umidade relativa do ar e luminosidade.
- Compreender fenômenos naturais ou sistemas tecnológicos, identificando e relacionando as grandezas envolvidas.

### Objetivos:

- Realizar registros adequados em tabelas e planilhas das medições realizadas
- Identificar taxas e tendências de variação

### Procedimento:

Preencha a tabela abaixo com os dados coletados na estação meteorológica no início da aula, no intervalo da aula e ao término da aula.

Data	Hora	Pressão Atmosférica	Temperatura	Umidade	Luminosidade
26/04/2016	07:00	100766	29,5	61,4	406
26/04/2016	08:40	100874	31,9	55,2	668
26/04/2016	09:30	100880	33,1	53,8	718

Tabela 41 – Atividade 1 – Registro dos dados meteorológicos coletados no dia 26/04/2016

Com base nos dados coletados e registrados, calcular os valores médios de cada um dos parâmetros:

Pressão Atmosférica	Temperatura	Umidade	Luminosidade
100840	31,5	56,8	597,3

Tabela 42 – Atividade 1 – Registro dos valores médios dos dados meteorológicos coletados no dia 26/04/2016

Considerando os dados coletados no início, no intervalo e no final da aula, podemos perceber a tendência para cada um dos parâmetros. Completar a tabela abaixo com esta previsão de tendência que pode ser: em alta, estável, em baixa:

Pressão Atmosférica	Temperatura	Umidade	Luminosidade
em alta	em alta	em baixa	em alta

Tabela 43 – Atividade 1 – Registro das tendências de alteração dos dados meteorológicos coletados no dia 26/04/2016

O professor pode questionar os alunos e propiciar um debate quanto a percepção corporal das grandezas temperatura, luminosidade, pressão atmosférica e umidade relativa do ar e avaliar através desta percepção se as tendências de variação para estas grandezas registradas na tabela estão corretas.

Outra proposta é a de confrontar os dados obtidos na estação meteorológica com os dados disponíveis em sites de previsão meteorológica.

[www.climatempo.com.br](http://www.climatempo.com.br)

[www.tempoagora.com.br](http://www.tempoagora.com.br)

<http://www.accuweather.com>

<http://tempo.cptec.inpe.br/>

#### Atividade 2: Analisando dados meteorológicos

Nesta atividade a proposta é o desenvolvimento de um trabalho colaborativo entre os alunos com o registro, em forma de tabelas e gráficos, de uma das variáveis medidas durante um dia inteiro (dados gravados no formato txt pela interface *Processing*) por cada aluno ou por um grupo de alunos.

Após os registros e a montagem dos gráficos propõe-se uma discussão entre os alunos ou grupos, da interferência de uma grandeza sobre a outra e quais seriam as implicações em diferentes regiões, diferentes latitudes e diferentes altitudes.

Habilidades e Competências trabalhadas:

- Reconhecer e saber utilizar corretamente símbolos, códigos e nomenclaturas de grandezas da Física, nas previsões climáticas, identificar temperaturas, pressão, umidade relativa do ar e luminosidade.
- Compreender fenômenos naturais ou sistemas tecnológicos, identificando e relacionando as grandezas envolvidas.
- Identificar relações entre grandezas e unidades de medida.
- Interpretar gráfico que represente relações entre grandezas.

Objetivos:

- Realizar registros adequados em tabelas e planilhas das medições realizadas
- Determinar pontos de mínimo e máximo
- Calcular médias e amplitudes de variação

Procedimento:

Com base nos dados obtidos em um arquivo diário, de medição horária, do tipo texto, da estação meteorológica preencha uma tabela para cada um dos parâmetros medidos: Pressão Atmosférica, Temperatura, Umidade ou Luminosidade

Valores obtidos por hora:

00:00	31,7	06:00	29,4	12:00	37,8	18:00	33,7
01:00	31,2	07:00	29,5	13:00	38,8	19:00	33,0
02:00	30,7	08:00	30,8	14:00	38,5	20:00	32,2
03:00	30,3	09:00	32,3	15:00	37,9	21:00	32,3
04:00	30,0	10:00	33,9	16:00	36,7	22:00	32,2
05:00	29,5	11:00	35,6	17:00	35,5	23:00	32,3

Tabela 44 – Atividade 2 – Registro dos dados horários de temperatura referentes ao dia 25/04/2016

Após o preenchimento das tabelas, orientar os alunos para registrar sob a forma de gráficos os valores referentes a cada um dos parâmetros medidos e determinar os valores

mínimo e máximo, com seus respectivos horários de ocorrência e calcular o valor médio e a amplitude de variação.

Gráfico:

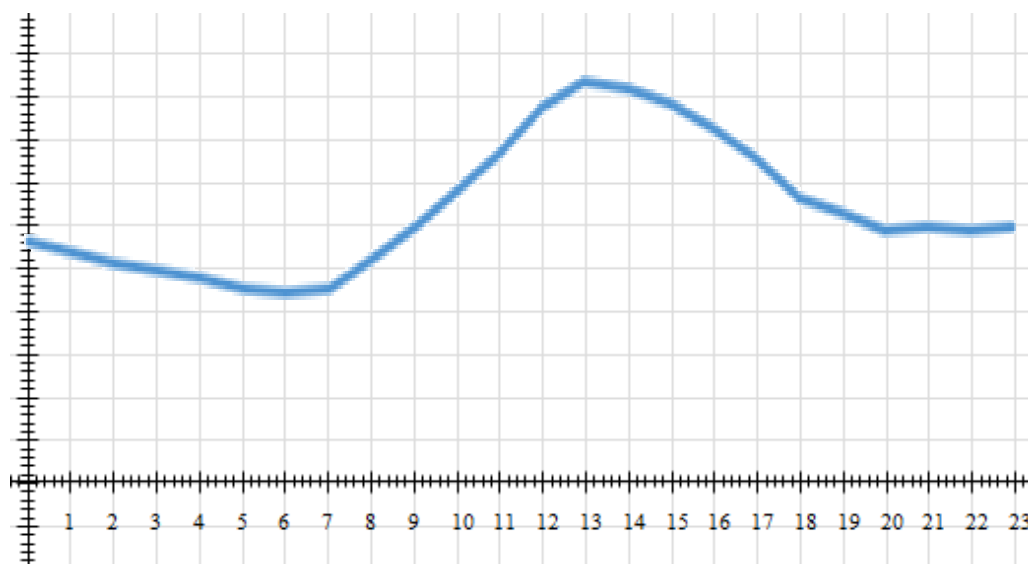


Gráfico 19 – Atividade 2 – Temperaturas horárias referentes ao dia 25/04/2016

Valor máximo: 38,8 °C

Horário: 13:00 h

Valor mínimo: 29,4 °C

Horário: 6:00 h

Variação (amplitude): 9,4 °C

Valor médio: 33,16 °C

Com base nos gráficos, podemos relacionar a variação de temperatura e a variação de umidade com relação a variação de luminosidade. Observamos que durante o transcorrer deste dia (25/04/2016) observamos o aumento da luminosidade a partir do nascer do sol (entre 6:00 e 7:00 horas), tendo um pico máximo (por volta de 12:00 horas), e o decréscimo até o ocaso (entre 17:00 e 18:00 horas). Neste mesmo dia observamos a temperatura tendo um acréscimo até as 13:00 horas e após este horário um decréscimo e a umidade tendo seu ponto mínimo às 13:00 horas. Desta forma podemos relacionar diretamente a temperatura com a luminosidade e inversamente a umidade com a luminosidade ou com a temperatura.

Considerando o gráfico da pressão atmosférica, podemos realizar uma análise com base nos dados da estação meteorológica e com os dados referentes a tábua de marés. Desta forma podemos comprovar a não relação entre as oscilações atmosféricas e as variações de marés.

As tábuas de marés são disponibilizadas pela Marinha do Brasil em:

<http://www.mar.mil.br/dhn/chm/box-previsao-mare/tabuas/>

A discussão sobre as oscilações da pressão atmosférica não será concluída no ambiente escolar de Ensino Médio, porém esta oscilação está intimamente atrelada a um modelo de reação da atmosfera a enorme quantidade de energia que recebe do sol e como esta energia absorvida altera a sua dinâmica. O professor poderá incentivar os alunos na pesquisa científica sobre este tema sugerindo alguns artigos disponibilizados na internet, como por exemplo: <http://www.scielo.br/pdf/rbef/v36n3/13.pdf>

As propostas apresentadas neste capítulo não esgotam o tema ou sua aplicação em sala de aula. Diversos outros experimentos podem decorrer do uso desta estação meteorológica, ou de seus sensores de forma individualizada em sala de aula, como por exemplo, experimentos de termometria, calor específico, propriedades dos materiais, ponto de orvalho, dentre outros que devem ser pensadas a partir deste e de outros trabalhos.

Como foi apresentado neste capítulo, é possível criar um objeto didático a baixo custo e com sensibilidade e precisão adequadas para análises dos fenômenos físicos mais sutis, como a pequena variação da pressão atmosférica durante o dia. No próximo capítulo será abordado um objeto educacional que a partir de sua funcionalidade poderá incentivar os alunos no ramo da Física Moderna.

**Analogia funcional do acelerador de partículas com o uso do *Arduino* e do *Processing***

As primeiras experiências com aceleração de partículas são creditadas a Joseph John Thomson, físico britânico que, em 1897, inventou o tubo de raios catódicos. Com esta invenção, Thomson descobriu o elétron e, em 1906, recebeu o Prêmio Nobel de Física, por essa descoberta.

Os aceleradores de partículas foram inventados na década de 1920 como uma ferramenta para a investigação em Física. Em um pequeno acelerador desenvolvido em 1927, na Universidade de Cambridge na Inglaterra, pelos físicos ingleses John Douglas Cockcroft e Ernest Thomas Sinton Walton foram realizados experimentos que possibilitaram a primeira reação nuclear induzida artificialmente. Devido a esse experimento os dois físicos ingleses ganharam, em 1951, o Prêmio Nobel de Física.

Pela invenção do acelerador de partículas cíclotron, em 1929, o americano Ernest Orlando Lawrence ganhou o prêmio Nobel de Física de 1939. O acelerador cíclotron é um dispositivo que acelera partículas carregadas utilizando a diferença de potencial elétrica. Com o aumento da velocidade da partícula, o feixe vai tendo seu raio aumentado, numa trajetória em espiral, até que ele é deslocado em direção ao alvo a ser bombardeado.

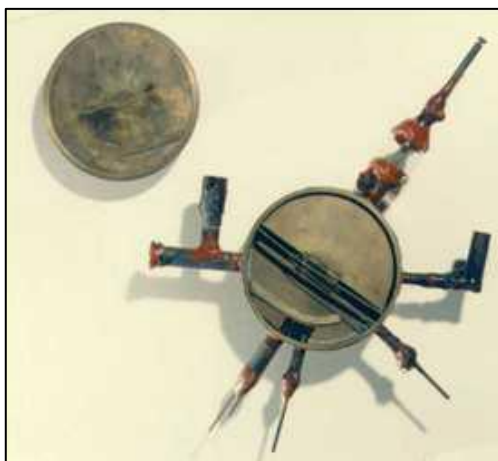


Figura 42 – Primeiro cíclotron construído por Lawrence  
Fonte: <http://www.rbfarma.org.br/files/rbf-2011-92-3-2.pdf>

A década de 1950 marcou o surgimento da Física de partículas como um novo ramo da Física e a necessidade da definição de um novo modelo para explicar a nova gama de partículas descobertas e suas interações, conhecido como Modelo Padrão. Este modelo é



uma teoria que descreve as forças fundamentais forte, fraca e eletromagnética, bem como as partículas fundamentais que constituem toda a matéria, de maneira consistente com a mecânica quântica e a relatividade especial.

Muitas das descobertas recentes são oriundas de experimentos realizados em grandes aceleradores de partículas. Atualmente, os aceleradores de partículas aumentaram consideravelmente suas dimensões e a energia transferida em uma busca das condições presentes no início do universo.

Entretanto, embora os cientistas continuem a promover colisões entre partículas com energias cada vez mais altas, usando aceleradores cada vez maiores, são forçados a reconhecer que nenhum acelerador terrestre será capaz de gerar partículas com energia suficiente para testar as teorias mais gerais. Só existiu uma fonte de partículas com essas energias: o próprio universo, no primeiro milissegundo de existência. (HALLIDAY, RESNICK e WALKER, 2012, p. 351)

Alguns dos maiores aceleradores do mundo são: O LHC, um dos experimentos do CERN (*European Council for Nuclear Research* - Organização Europeia para Pesquisa Nuclear), localizado na periferia da cidade de Genebra, na Suíça e é formado por um enorme tubo circular com circunferência de 26,7 km e diâmetro de 7 m, instalado a cerca de 100 m abaixo do solo e o acelerador de partículas Tevatron do Fermilab (Fermi National Accelerator Laboratory) localizado na cidade de Batavia, próximo a Chicago, Illinois, que é formado por um tubo circular com circunferência de 6.3 km.

No Brasil, há alguns aceleradores destinados a pesquisas e a produção de rádio fármacos. O acelerador do Laboratório Nacional de Luz Síncrotron (LNLS) do Centro Nacional de Pesquisa em Energia e Materiais (CNPEM), localizado na cidade de Campinas no estado de São Paulo, é utilizado para pesquisas nas áreas de Física, Química, Engenharia, Meio Ambiente e Ciências da Vida, desde 1997. Para a produção de rádio fármacos, podemos destacar o Cyclone 30 adquirido em 1998 pelo Instituto de Pesquisas Energéticas e Nucleares (IPEN/CNEN) que pode acelerar prótons entre 15 e 30 MeV, o CV-28 adquirido em 1974 e o ciclotron RDS 111 instalado em 2003, estes últimos do Instituto de Engenharia Nuclear (IEN/CNEN).

Existem diversos tipos de aceleradores, porém eles podem ser classificados de acordo com a trajetória descrita pelas partículas em aceleradores lineares e aceleradores circulares.

Aceleradores	Partículas	Energia
Cockcroft-Walton (1930) <sup>32</sup>	prótons	400 kV
Cockcroft-Walton (Fermilab-EUA) <sup>33</sup>	íons	750 keV
Van de Graff (1931) <sup>34</sup>	íons	2 MeV
Van de Graff (USP-Brasil-1954)	prótons	3,8MeV
Van de Graff (atual)	íons	20 MeV
Linac (1931) <sup>3</sup>	íons	1,25 MeV
Linac (CBPF – Brasil – 1963)	elétrons	2 MeV
Linac (SLAC – EUA) <sup>2</sup>	elétrons	20 GeV
Ciclotron (1931) <sup>3</sup>	prótons	1,1MeV
Ciclotron (IPEN - Brasil) <sup>2</sup>	prótons	30 MeV
Sincrotron (1949) <sup>2</sup>	elétrons	300 MeV
Sincrotron (LNLS – Brasil)	elétrons	1,37 GeV
Sincrotron (Tevatron-Fermilab-EUA) <sup>2</sup>	prótons	1 TeV
Sincrotron (LEP – CERN) <sup>2</sup>	elétrons	104,5 GeV
Sincrotron (LHC – CERN – 2010) <sup>2</sup>	prótons	7 TeV

Tabela 45 – Dados referentes a alguns aceleradores de partículas

Nos aceleradores lineares, as partículas se deslocam no vácuo ao longo de um tubo longo e reto. Os elétrons são acelerados por transferência de energia pelos *clístrons*, que são geradores de micro-ondas de alta energia, localizados em intervalos regulares ao longo do acelerador. Para manter o confinamento e a focalização do feixe de partículas são utilizados ímãs supercondutores ou eletroímãs convencionais. Quando o feixe de partículas atinge um alvo no final do túnel, vários detectores registram os eventos: as partículas subatômicas e a radiação liberada. Estes aceleradores são enormes, e são mantidos no subsolo. Um exemplo de acelerador linear é o Linac no Laboratório de Acelerador Linear de Stanford (SLAC) na Califórnia, que tem cerca de 3 km de comprimento.

Nos aceleradores circulares, como Sincrotron do CERN, as partículas são geradas e aceleradas inicialmente por um acelerador linear (Linac), depois seguem até o *booster*, ou síncrono injetor, que é um acelerador circular por micro-ondas e rádio frequência, aumentando a energia das partículas. Após esta etapa, as partículas viajam em um grande anel circular comprimidas em um fino feixe de partículas, gerando emissões de luz, que variam do infravermelho aos raios X, a cada passagem pelos magnetos de deflexão que são

<sup>32</sup> LEE, Shyh-Yuan. **Accelerator Physics**. New Jersey, USA: World Scientific, 2004.

<sup>33</sup> LUIZ, L. C.; MONTEIRO, K. T. S.; BATISTA, R. T. Os aceleradores de partículas e sua utilização na produção de radiofármacos. **Revista Brasileira de Fármacos**, 2011.

<sup>34</sup> STEERE, Andrew Robert. **A timeline of major particle accelerators**, Thesis submitted to Michigan State University, 2005.

utilizados para a manutenção da órbita circular. A cada passagem pelos magnetos, são emitidos fótons que são dirigidos as estações de pesquisa. Neste anel, ainda estão presentes aceleradores de rádio frequência para manter a energia das partículas no anel.

Em ambos os aceleradores, a aceleração das partículas é realizada por interação com campos elétricos oriundos das micro-ondas ou da rádio frequência e a determinação da trajetória e concentração do feixe é alcançada pela interação com campos magnéticos, oriundos da ação de ímãs permanentes ou pela ação do eletromagnetismo. Estes ímãs ou estes magnetos precisam ser mais intensos à medida que a velocidade da partícula aumenta, pois, com o aumento da velocidade, essas partículas aumentam também a sua energia cinética, tornando-se mais difícil interferir em sua trajetória.

Estando o feixe de partículas devidamente orientado e com energia cinética adequada, ele colide com um alvo, onde detectores vão colher informações de acordo com o interesse dos pesquisadores. Essas colisões podem ocorrer com o uso de um Alvo fixo - o feixe de partículas, após obter uma grande quantidade de energia no acelerador, colide com um alvo imóvel; ou com Alvo móvel - são utilizados dois feixes de partículas em sentidos contrários que produzirão uma colisão entre partículas de cada um dos feixes.

A partir destas colisões, os detectores registram as partículas e a radiação resultante da própria colisão. Eles são uma das mais importantes partes do acelerador e podem ser de vários tipos, desde câmaras de bolhas e névoa até detectores eletrônicos em estado sólido. Um laboratório de colisões de partículas pode ter vários tipos de detectores localizados em várias partes do acelerador.

A descoberta de novas partículas ocorre nas colisões frontais, assim como afirma HALLIDAY, RESNICK e WALKER em seu livro Fundamentos da Física – Óptica e Física Moderna.

As novas partículas são sempre produzidas em colisões frontais entre prótons ou elétrons de alta energia produzidos em aceleradores situados em laboratórios Como o Brookhaven National Laboratory (perto de Nova York), o Fermilab (perto de Chicago), o CERN (perto de Genebra), o SLAC (perto de San Francisco) e o DESY (perto de Hamburgo). Foram com o auxílio de detectores cuja sofisticação aumentou até se tornarem tão grandes e complexos como os próprios de partículas de algumas décadas atrás.

Hoje em dia são conhecidas centenas de partículas. Para batizá-las, os físicos esgotaram as letras do alfabeto grego e a maioria é conhecida apenas pelo número de Ordem em um catálogo de que é publicado regularmente, para tentar fazer sentido dessa profusão de partículas, os cientistas procuram de acordo com critérios simples. O resultado é conhecido como Modelo Padrão de partículas (2012, p. 352)

As colisões são estudadas buscando o conhecimento da dinâmica do universo desde antes da formação dos próprios átomos até os dias de hoje. As colisões também propiciam estudos em busca de respostas do que acontece no mundo das velocidades relativísticas, teoria criada por Albert Einstein que prevê o comportamento da matéria quando sua velocidade se aproxima a velocidade da luz. Os aceleradores ainda propiciam pesquisas para a melhoria de nossa qualidade de vida ou pesquisas que objetivam responder a nossas perguntas sobre nossa própria existência.

Com o objetivo de aproximar os aceleradores do contexto de sala de aula, Albert Pujol Fillat e Celestí Capell Arqués propuseram o experimento *Acelerador de partículas “Maqueta educativa”* que foi premiado na terceira edição do Concurso de Divulgação do Centro Nacional de Física de Partículas, Astropartículas y Nuclear - CPAN em 2012. Neste experimento os autores propuseram, ao custo de 106,00 €, estimado em 2012, uma maquete educativa e funcional baseada em eletromagnetismo e controlada por um microcontrolador PIC.

Por ser esta dissertação uma proposta para experimento de baixo custo, a *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing* tem por objetivo principal aproximar os alunos das escolas públicas de Ensino Médio da Física Moderna, que é o tópico curricular que abrange os aceleradores de partículas e suas descobertas. Este experimento propõe uma aproximação didática do funcionamento de um acelerador de partículas circular, porém com a interação de um campo magnético para a aceleração de uma esfera de aço, fazendo o papel de uma grande partícula, dentro de um tubo plástico transparente e flexível fechado na forma de um círculo. No funcionamento desta analogia são abrangidos conceitos básicos de energia, eletricidade, eletromagnetismo e movimento circular.

Cabe ressaltar, assim como o fizeram em seu experimento Albert Pujol Fillat e Celestí Capell Arqués, que para efeitos práticos, didáticos e pela facilidade que implica foi escolhida a aceleração da esfera por campo magnético, diferentemente do que ocorre nos aceleradores de partículas, que a aceleração é produzida por campos eletrostáticos. E que os campos magnéticos presentes nestes aceleradores têm a função de produzir a deflexão do feixe de elétrons a fim de obter a trajetória circular.

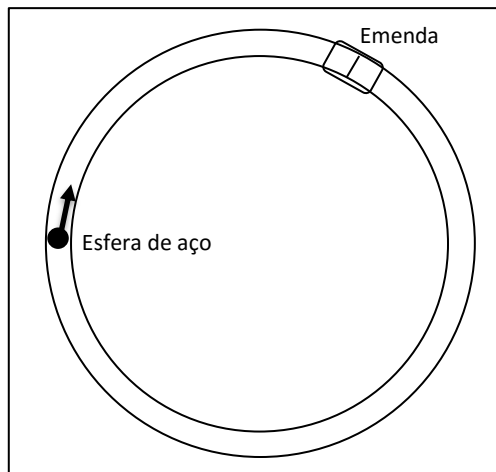


Figura 43 – Digrama simplificado da trajetória do experimento

O objetivo desta analogia é o mesmo do acelerador de partículas circular, ou seja, fornecer energia cinética ou de movimento a uma partícula, que, nessa analogia, é representada pela esfera de aço. A energia obtida pelo movimento da partícula é a energia cinética:

$$E_c = \frac{1}{2}mv^2$$

Uma vez que a esfera possui massa constante, a energia cinética depende exclusivamente do quadrado da velocidade. Desta maneira, para gerar um incremento na energia da partícula é necessário um aumento de sua velocidade. Para isto, é fundamental aplicar à partícula uma força, gerando por consequência uma aceleração. O condicionamento da aceleração à aplicação de uma força é enunciado pela segunda lei de Newton: A força resultante que age sobre um corpo é igual ao produto da massa do corpo pela aceleração.

$$\vec{F} = m\vec{a}$$

Diferentemente do acelerador de partículas, onde as partículas se movimentam no vácuo e sem contato mecânico com as paredes do duto, deslocando-se praticamente sem atrito, nesta analogia a esfera possui contato mecânico com as paredes internas da mangueira plástica e com o próprio ar do seu interior, ocasionando perdas de energia. Deste modo é essencial a aplicação de uma força para apenas manter o movimento, compensando todas as perdas do sistema (inerciais, por atrito, etc.).

A força necessária para manter ou incrementar a velocidade da esfera é obtida a partir de sua interação com a força magnética de atração, gerada a partir da eletrização das bobinas. Desta forma, a esfera deve ser constituída de material ferromagnético<sup>35</sup> para que seja atraída magneticamente pela bobina. A intensidade da força magnética de atração gerada depende da geometria da bobina (diâmetro, comprimento e número de espiras) e da corrente que circula por ela.

Para a construção das bobinas, foi crucial a utilização de um núcleo provisório com o objetivo de propiciar uma geometria adequada. Para este núcleo, pode ser utilizado um segmento de aproximadamente 15 cm de um tubo ½ polegada de PVC roscável, normalmente utilizado para instalações hidráulicas de água fria.

Para o enrolamento de cada uma das bobinas, foram utilizados, aproximadamente, 16 metros de fio esmaltado AWG 18. Este fio possui as seguintes características: 1,02 mm de diâmetro da alma de cobre, 1,07 mm de diâmetro quando esmaltado, 0,817 mm<sup>2</sup> de seção reta transversal, uma resistência média de 20,8 ohms/km e uma capacidade de condução máxima, contínua, de 2,45 A. A bobina é formada por 218 espiras distribuídas em quatro camadas. Sendo 56 espiras na primeira camada, 55 espiras na segunda camada, 54 espiras na terceira camada e 53 espiras na quarta e última camada.

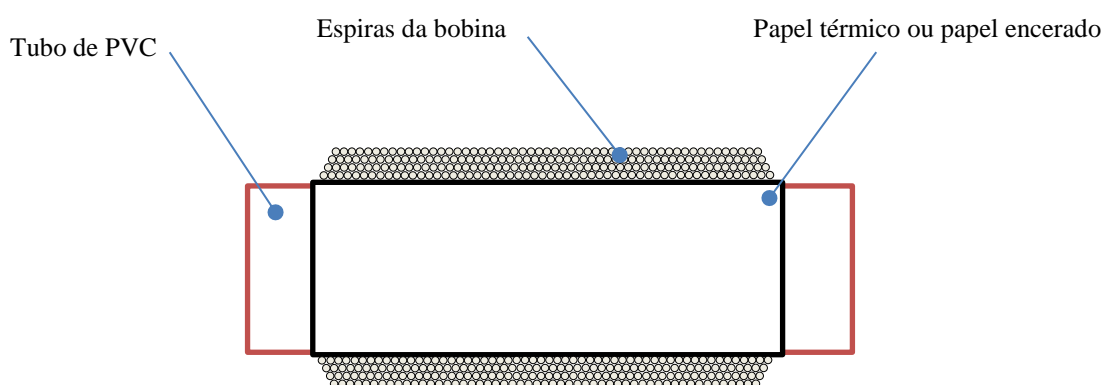


Figura 44 – Detalhe do processo de confecção da bobina

Para que as bobinas gerem energia magnética suficiente para interagir com a esfera é essencial uma potência elétrica que pode ultrapassar os 150 watts. Esta exigência elétrica

---

<sup>35</sup> Nos materiais ferromagnéticos possuem ímãs elementares em sua estrutura física que sofrem grande influência do campo magnético indutor. De modo que, eles ficam majoritariamente orientados no mesmo sentido do campo magnético aplicado e são fortemente atraídos por um ímã.

torna-se um óbice quanto ao custo do projeto, pois a montagem de uma fonte de alimentação convencional para atender esta potência é de alto custo.

Com o objetivo de mantermos o baixo custo do experimento, optamos pelo uso de uma fonte ATX utilizadas em computadores pessoais com pelo menos 400 watts de potência, acoplada a um banco de capacitores formado por, pelo menos, 4 capacitores de 5000  $\mu\text{F}$  x 35 V obtendo uma capacitância total mínima de 20.000  $\mu\text{F}$  ou 20 mF. Este banco de capacitores tem como função acumular a energia necessária para possibilitar o breve instante de funcionamento das bobinas impedindo que esse pulso de consumo se propague até a fonte acionando seu circuito de proteção e desligando-a.



Figura 45 – Fonte de alimentação e seu conector padrão

A primeira ligação elétrica imprescindível para o funcionamento da fonte é a do terminal PS\_ON com o terminal COM. Esta ligação pode ser realizada com um fio, mantendo o terminal da fonte ou apenas ligando o fio verde a qualquer um dos fios pretos da fonte. Esta ligação habilita o funcionamento da fonte.

Apenas serão utilizadas nesse projeto as derivações 12V (amarelo), utilizado para a alimentação das bobinas; -12V (azul), empregado para a alimentação dos ventiladores de resfriamento das bobinas e COM ou 0V (preto) operando como comum ou neutro da fonte. As demais saídas podem ser retiradas ou apenas isoladas. O diagrama das ligações referentes a fonte de alimentação é apresentado a seguir:

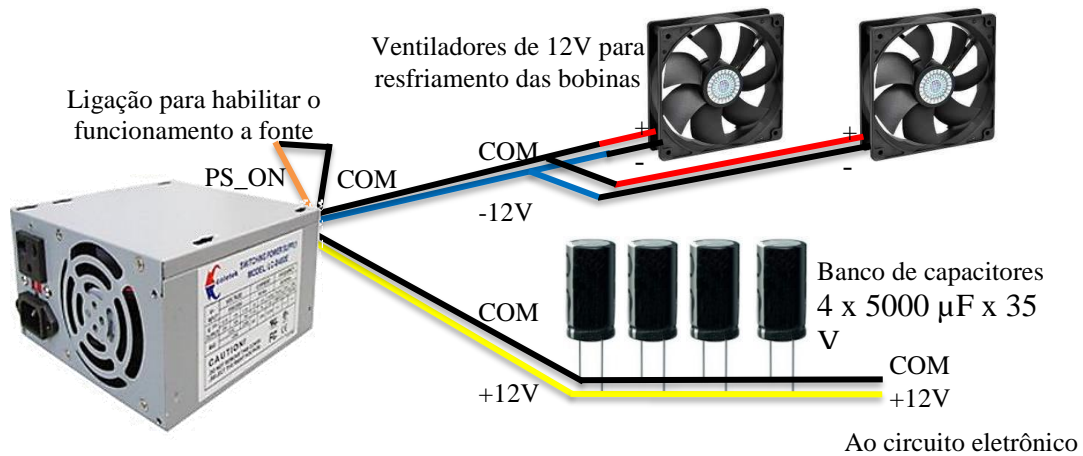


Figura 46 – Diagrama das ligações da fonte de alimentação para seu funcionamento e para alimentação dos ventiladores de resfriamento e do banco de capacitores

As bobinas são energizadas pela fonte de alimentação e enquanto permanecem energizadas funcionam de maneira análoga a um ímã. A energização da bobina gera uma força de atração que interage com a esfera ferromagnética de aço atraindo-a para o seu centro, independentemente do lado da bobina que a esfera se encontre.

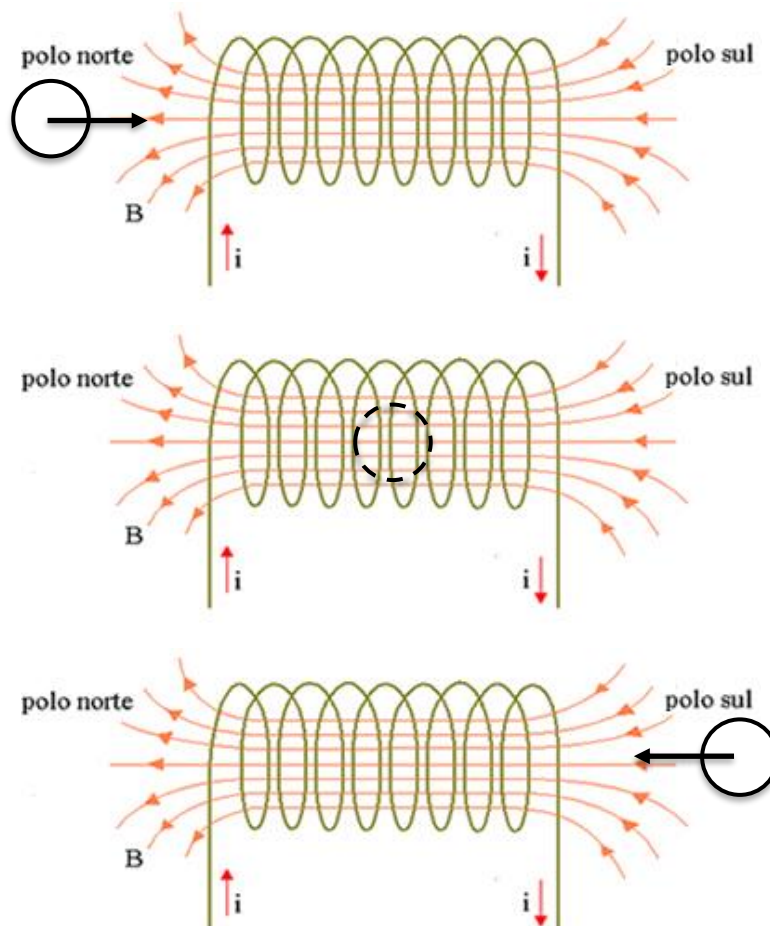


Figura 47 – Esquema de polarização das bobinas e sua interação com a velocidade da esfera  
Adaptado de <http://brasilecola.uol.com.br/fisica/campo-magnetico-no-solenoides.htm>



Para que o efeito da aceleração sobre a esfera ocorra, é crucial que exista um sistema de detecção para determinar o momento exato que a esfera penetre na zona de atração magnética da bobina, para que a mesma seja energizada. E, a partir deste momento, a bobina precisa permanecer energizada apenas o período em que a esfera se desloca até o centro da bobina. Nesta condição a força de atração gera uma aceleração na esfera, transferindo energia cinética para a mesma. Caso o tempo de energização da bobina supere este período, haverá uma frenagem magnética progressiva da esfera, diminuindo sua velocidade ou até interrompendo o seu movimento. Desta maneira, podemos concluir que a velocidade de deslocamento da esfera na mangueira é inversamente proporcional ao tempo de energização da bobina.

A detecção de proximidade da esfera é realizada por uma barreira óptica formada por uma fonte luminosa e por um fotodetector ambos instalados a uma distância de 1 cm da borda de entrada da esfera de cada uma das bobinas. Quando a esfera passa pela barreira ela interrompe o fluxo luminoso e altera a condição de polarização do detector. Para a fonte luminosa foi utilizado um LED de alto brilho na cor branca, para permitir que a interrupção do feixe luminoso seja facilmente percebida. Para a detecção foi empregado um fototransistor infravermelho, componente com um baixo tempo de resposta e imune às variações de luminosidade do ambiente.

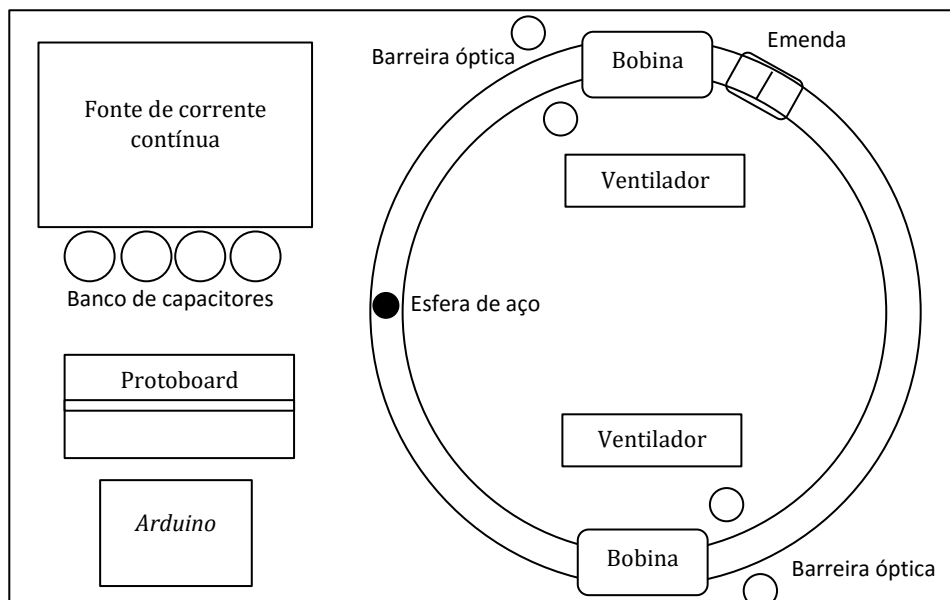


Figura 48 – Esquema da montagem do experimento

O sinal oriundo do fototransistor é captado por uma das entradas analógicas do *Arduino* que, a partir do ajuste de um resistor variável, acoplado a outra das entradas analógicas, determina o período de disparo da bobina. A variação deste resistor altera o tempo de disparo das bobinas, controlando a velocidade de deslocamento da esfera de aço na trajetória circular.

Como as saídas do *Arduino* possuem uma severa limitação de corrente, cerca de 40 mA, para o acionamento das bobinas se faz indispensável o uso de uma interface de potência. Esta interface simples e de fácil montagem é constituída de um optoacoplador<sup>36</sup>, e um transistor do tipo MOSFET<sup>37</sup> de potência. Com o circuito desta interface o *Arduino* é capaz de controlar, de maneira indireta, correntes que podem atingir 60 A de pico. A montagem do circuito eletrônico necessário a este experimento foi realizada na matriz de contatos e está apresentada no diagrama a seguir

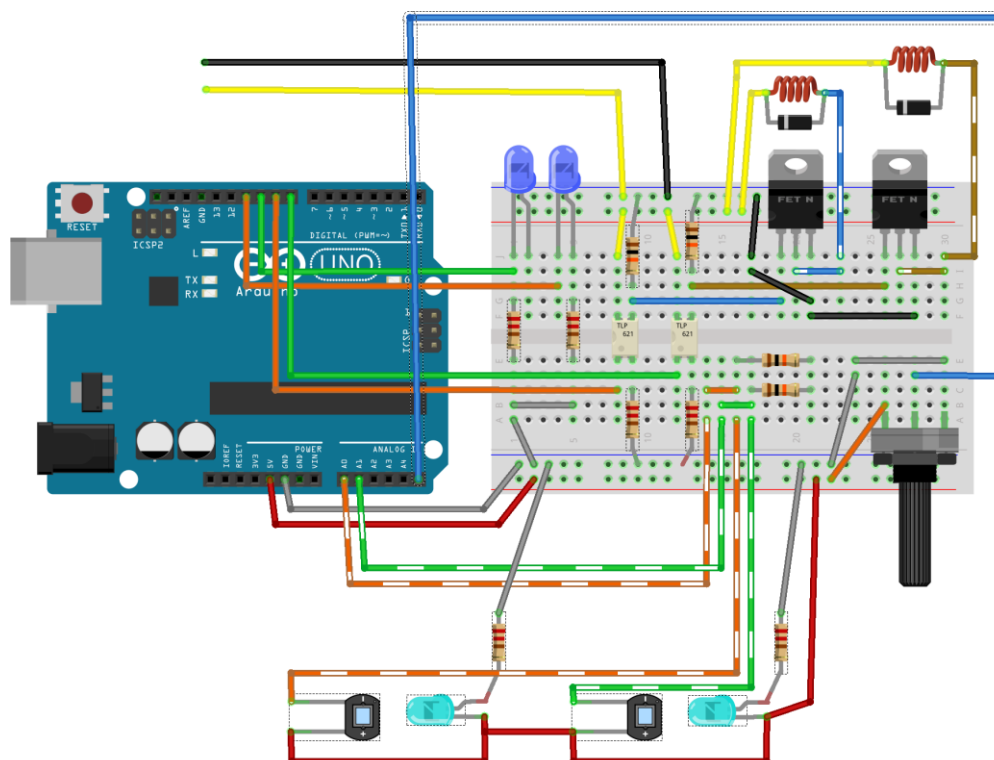


Figura 49 – Esquema sugerido para a montagem do circuito de controle na placa de contatos

<sup>36</sup> Fotoacoplador, também chamado de acoplador óptico, optoacoplador ou opto isolador, é um componente formado basicamente por um LED e um foto transistor dentro de um CI com a função de transferir uma informação elétrica entre dois circuitos através de luz, ou seja, sem contato elétrico entre eles

<sup>37</sup> O transistor MOSFET (acrônimo de Metal Oxide Semiconductor Field Effect Transistor, ou transistor de efeito de campo de semiconductor de óxido metálico), é, de longe, o tipo mais comum de transistores de efeito de campo em circuitos tanto digitais quanto analógicos. Seu funcionamento baseia-se no controle de condução de corrente a partir da variação de tensão no pino referente à porta ou gate.

A interface do experimento desenvolvida em *Processing*, possui dois ambientes: o primeiro é uma simulação virtual do experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*, com ajuste de incremento e decremento de velocidade, bem como realiza o cálculo e a apresentação na tela das informações de raio médio da trajetória, perímetro da trajetória, massa da esfera, tempo de disparo, período e frequência, a velocidade angular, a velocidade linear, o número de voltas realizadas e a respectiva distância percorrida. O segundo ambiente intitulado de *Dados reais do Experimento*, recebe os dados oriundos do *Arduino*, via conexão serial, calculando e apresentando na tela as informações de raio médio da trajetória, perímetro da trajetória, massa da esfera, tempo de disparo, período e frequência, a velocidade angular, a velocidade linear, o número de voltas realizadas e a respectiva distância percorrida.

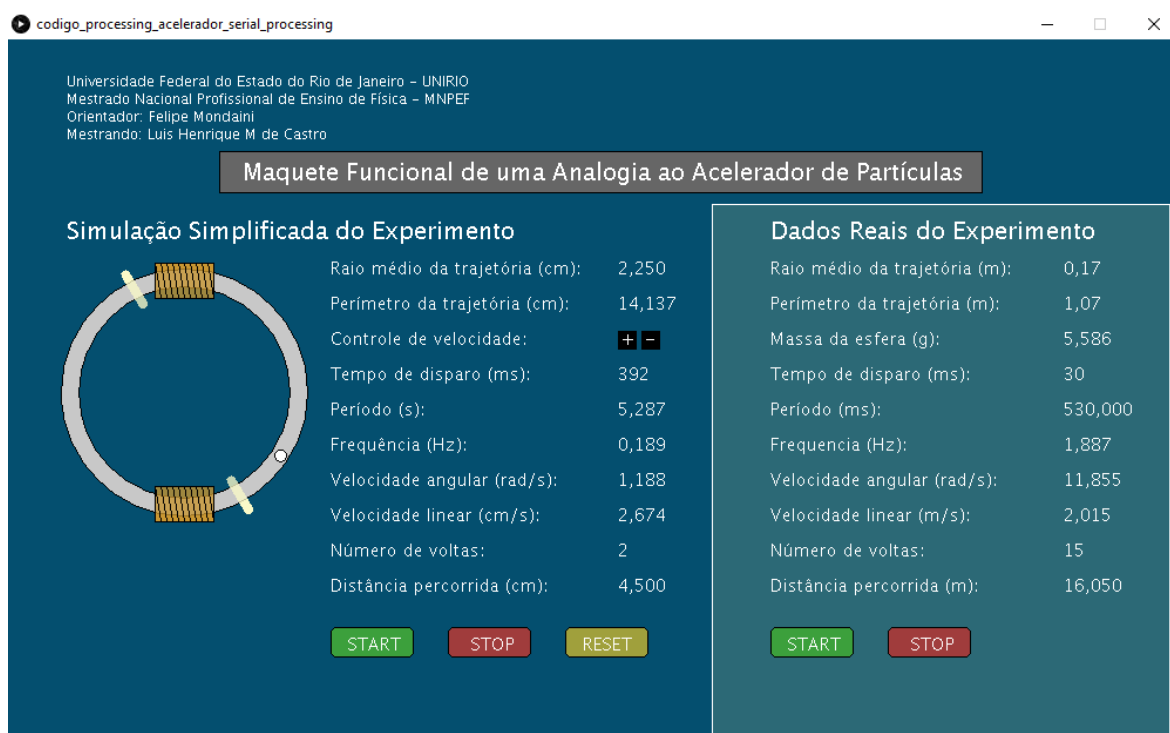


Figura 50 – Interface computacional do experimento desenvolvida em *Processing*

O uso desse experimento, objetivamente, incentiva e favorece o desenvolvimento das habilidades e das competências preconizadas pelos descritores relacionados ao estudo da física moderna, do movimento, da eletricidade e do eletromagnetismo, presentes na Base Nacional Comum e no Currículo Mínimo do Estado do Rio de Janeiro, conforme apresentado na tabela a seguir:

Estudo do movimento, eletricidade, eletromagnetismo e física moderna
<b>Base Nacional Comum</b>
– Observar e descrever posições, deslocamentos e velocidades de objetos em linhas, como carros em estradas; em planos, como bolas numa quadra; ou no espaço, como aviões em torno do globo.
– Atribuir a permanência de um objeto numa mesma posição, ou a constância de sua velocidade, ao cancelamento das forças agindo nele, e atribuir qualquer alteração na sua velocidade à força resultante atuando nele.
– Compreender e explicar o funcionamento de circuitos elétricos simples de equipamentos do cotidiano, como chuveiros, aquecedores e lâmpadas, a partir de princípios gerais e modelos simples, associando suas tensões, resistências e potências.
– Estabelecer as relações entre campos elétricos e magnéticos, em suas variações no tempo e no espaço, identificando os princípios de funcionamento de equipamentos e sistemas, tais como eletroímãs, motores, geradores, transformadores, microfones, alto-falantes.
– Construir equipamentos simples de sentido prático ou lúdico, a partir de componentes de fácil acesso, explicitando os fenômenos elétricos e magnéticos envolvidos em cada parte do seu arranjo.
– Descrever e explicar dispositivos semicondutores e lasers, em termos dos princípios físicos que presidem sua operação, e revelar suas propriedades funcionais para uso em equipamentos e sistemas.
– Comparar diferentes modelos explicativos para a constituição da matéria, desenvolvidos ao longo da história, como os modelos de Dalton, Rutherford e Bohr, e esboçar o modelo atual das forças fundamentais em termos de quarks, léptons e bósons mediadores.
<b>Currículo Mínimo do Estado do Rio de Janeiro</b>
– Compreender fenômenos magnéticos para explicar, por exemplo, o magnetismo terrestre, o campo magnético de um ímã e a inseparabilidade dos polos magnéticos.
– Utilizar a conservação do momento linear e a identificação de forças para fazer análises, previsões e avaliações de situações cotidianas que envolvem os movimentos.
– Reconhecer a conservação do momento linear e, por meio dela, as condições impostas aos movimentos

Tabela 46 – Descritores relacionados ao estudo do movimento, eletricidade, eletromagnetismo e física moderna

Atividade1: Demonstrar o uso do simulador.

Apresentar aos alunos a interface de simulação do experimento, realizando a demonstração dos controles de velocidade. Com o auxílio de um cronômetro realizar a medição do tempo de cada volta completa e comparar ao valor registrado na interface.

A partir dos valores medidos e calculados realizar o registro nas tabelas e determinar o valor médio e o erro relativo das medições.

Habilidades e Competências trabalhadas:

- Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.
- Conceituar e determinar as principais grandezas envolvidas no movimento circular.

Ajuste de velocidade	Período registrado no simulador (s)	Medição 1 (s)	Medição 2 (s)	Medição 3 (s)	Valor Médio (s)
Velocidade 1	5,243	5,27	5,27	5,10	5,19
Velocidade 2	3,493	3,53	3,46	3,48	3,47
Velocidade 3	2,334	2,27	2,31	2,36	2,34

Tabela 47 – Atividade 1 – Exemplo de dados medidos utilizando o cronometro de um aparelho celular

Ajuste de velocidade	Período registrado no simulador (s)	Período Médio (s)	Erro relativo (%)
Velocidade 1	5,243	5,19	1,11
Velocidade 2	3,493	3,47	0,66
Velocidade 3	2,334	2,34	0,04
Erro relativo médio			0,6%

Tabela 48 – Atividade 1 – Exemplo do cálculo do erro relativo

$$Erro\% = \frac{|valor\ medido - valor\ real|}{valor\ real} \times 100$$

Com base nos valores registrados na tabela e considerando o raio médio da trajetória igual a 2,250 cm, calcular a velocidade angular e a velocidade linear em cada um dos períodos médios registrados na tabela.

Ajuste de velocidade	Período Médio (s)	Velocidade angular (rad/s)	Velocidade linear (m/s)
Velocidade 1	5,19	1,21	0,027
Velocidade 2	3,47	1,81	0,040
Velocidade 3	2,34	2,68	0,060

Tabela 49 – Atividade 1 – Exemplo do cálculo de velocidade angular e velocidade linear

$$\omega = \frac{2\pi}{T} \text{ ou } \omega = 2\pi f$$

$$v = \frac{2\pi r}{T}$$

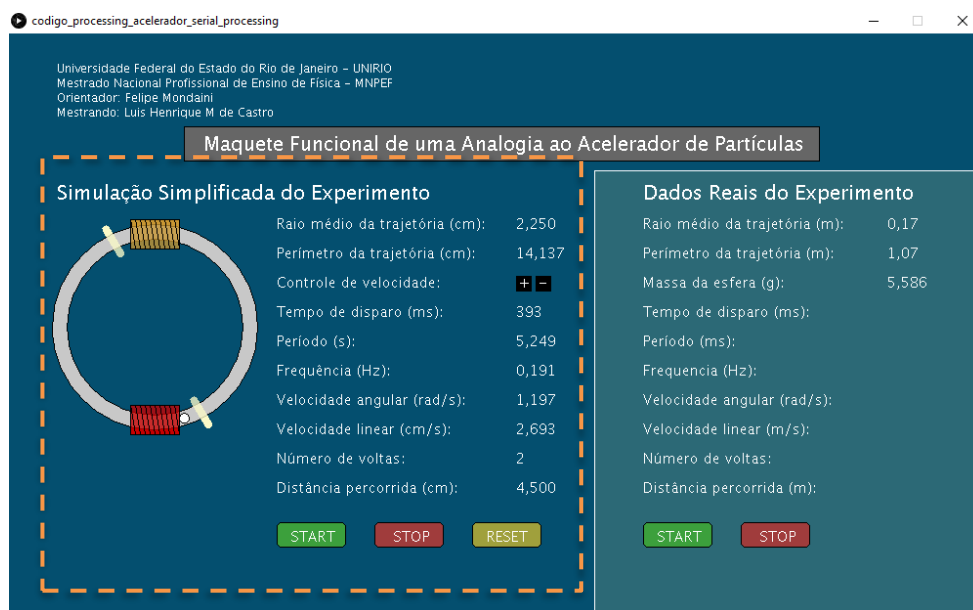


Figura 51 – Atividade 1 – Detalhe do simulador na interface computacional do experimento

Atividade 2: Demonstrar o uso do experimento.

Demonstrar aos alunos o funcionamento do experimento, alterando a velocidade de deslocamento da esfera a partir de variações da posição do potenciômetro.

Ajustar no potenciômetro a velocidade mínima de rotação e com o auxílio de um cronômetro realizar a medição do tempo de cada volta completa e comparar ao valor registrado na interface. Repetir este procedimento com pequenas variações de velocidade

A partir dos valores medidos e calculados realizar o registro nas tabelas e determinar o valor médio e o erro relativo das medições.

Habilidades e Competências trabalhadas:

- Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.
- Conceituar e determinar as principais grandezas envolvidas no movimento circular.

Ajuste de velocidade	Período registrado no experimento (s)	Medição 1 (s)	Medição 2 (s)	Medição 3 (s)	Valor Médio (s)
Velocidade 1	1,049	1,04	0,99	1,05	1,03
Velocidade 2	0,980	0,89	1,04	0,97	0,97
Velocidade 3	0,850	0,83	0,89	0,79	0,84

Tabela 50 – Atividade 2 – Exemplo de dados medidos utilizando o cronometro de um aparelho celular

$$Erro\% = \frac{|valor\ medido - valor\ real|}{valor\ real} \times 100$$

Ajuste de velocidade	Período registrado no experimento (s)	Período Médio (s)	Erro relativo (%)
Velocidade 1	1,049	1,03	2,13
Velocidade 2	0,980	0,97	1,36
Velocidade 3	0,850	0,84	1,57
Erro relativo médio			1,69

Tabela 51 – Atividade 2 – Exemplo do cálculo do erro relativo

Com base nos valores registrados na tabela e considerando o raio médio da trajetória igual a 0,17 m e a massa da esfera igual a 5,586 g, calcular a velocidade angular e a velocidade linear em cada um dos períodos médios registrados na tabela.

Ajuste de velocidade	Período Médio (s)	Velocidade angular (rad/s)	Velocidade linear (m/s)	Energia Cinética (mJ)
Velocidade 1	1,03	6,09	1,04	3,02
Velocidade 2	0,97	6,47	1,10	3,55
Velocidade 3	0,84	7,48	1,27	4,50

Tabela 52 – Atividade 2 – Exemplo do cálculo de velocidade angular e linear

$$\omega = \frac{2\pi}{T} \text{ ou } \omega = 2\pi f$$

$$v = \frac{2\pi r}{T}$$

$$Ec = \frac{1}{2}mv^2$$

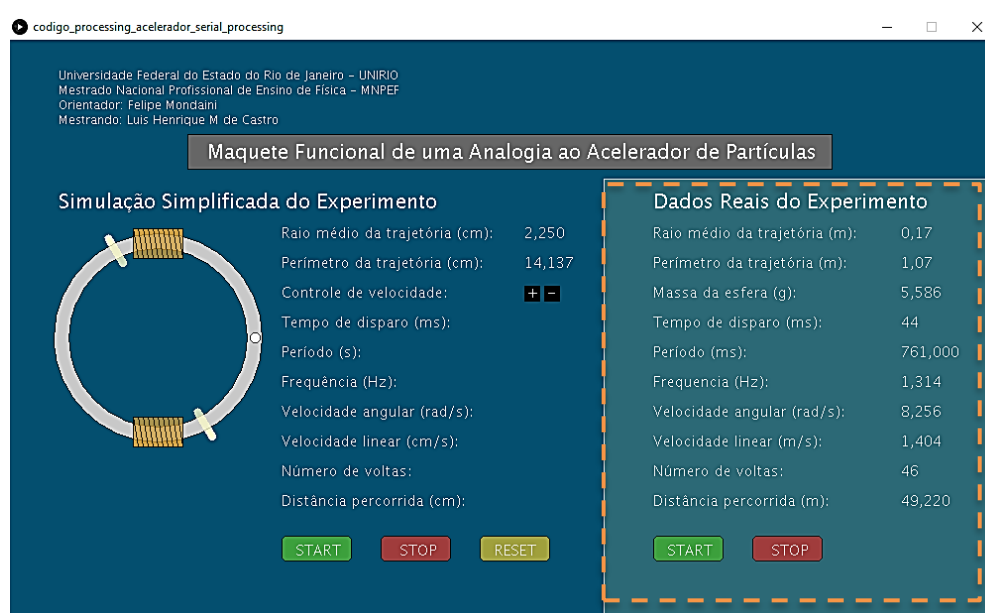


Figura 52 – Atividade 2 – Detalhe da aquisição de dados na interface computacional do experimento

Cabe ressaltar que o experimento consegue implementar uma velocidade de aproximadamente 3 m/s a esfera, gerando uma energia cinética de aproximadamente 25,13 mJ.



Dados Reais do Experimento	
Raio médio da trajetória (m):	0,17
Perímetro da trajetória (m):	1,07
Massa da esfera (g):	5,586
Tempo de disparo (ms):	13
Período (ms):	344,000
Frequência (Hz):	2,907
Velocidade angular (rad/s):	18,265
Velocidade linear (m/s):	3,105
Número de voltas:	1113
Distância percorrida (m):	1190,910

START STOP

Figura 53 – Atividade 2 – Detalhe da aquisição de dados na interface computacional do experimento em velocidade máxima

### Atividade 3: Demonstrar qualitativamente a aceleração da esfera

O Transistor MOSFET Funciona como uma chave eletrônica. Sendo assim podemos com pequenas alterações no circuito e com o auxílio de um multímetro digital comprovarmos a aceleração provocada pela polarização da bobina na esfera.

Procedimento:

- Desligar o cabo USB, interrompendo a alimentação do *Arduino* UNO.
- Com o auxílio de um ímã ou apenas inclinando o experimento posicione a esfera em uma das barreiras ópticas.
- Identifique o MOSFET chaveador da bobina referente a esta barreira óptica.
- Retire da matriz de contato este MOSFET.
- Selecionar a escala de medição de corrente com pelo menos 20 A de fundo de escala
- Ligar o conector oriundo da bobina ao borne +20 A do multímetro com o auxílio de uma ponteira jacaré
- Encostar brevemente a ponteira comum do osciloscópio no conector oriundo da fonte de alimentação que permaneceu no *proto-board*.
- A esfera será acelerada dentro da mangueira plástica.

Após alguns testes de polarização da bobina é possível alcançar uma volta completa com este procedimento, provando o efeito do chamado canhão magnético.

Para medir a corrente elétrica que percorre a bobina, pode ser realizado o procedimento de encostar a ponteira comum do osciloscópio ao conector oriundo da fonte

de alimentação que permaneceu na matriz de contatos por um tempo não superior ao necessário para a leitura do valor da corrente pelo multímetro. O valor obtido para a corrente deverá ficar próximo a 15 A, definindo uma potência de pulso próxima a 180 w.

#### Atividade 4: Análise do experimento e analogia ao acelerador de partículas

Nesta atividade é proposta uma dinâmica para a construção coletiva de uma explicação fisicamente correta do funcionamento do experimento. Cabe salientar a importância da geração do movimento e de sua manutenção, considerando todas as perdas presentes no experimento.

Após a conclusão desta primeira etapa é proposto uma nova dinâmica para realizar a comparação entre este experimento e o acelerador de partículas, salientando as diferenças na construção, na operação e na aplicação dos resultados.

Conclui-se, esta atividade, apresentando aos alunos o modelo padrão e as partículas já encontradas com o uso dos aceleradores de partículas.

Como abordado neste capítulo, o experimento Analogia funcional do acelerador de partículas com o uso do *Arduino* e do *Processing* pode ser de grande valia no incentivo e na inserção do conhecimento “Física Moderna” em sala de aula de turmas de ensino médio, pois o experimento permite a visualização, por parte do aluno, da interação eletromagnética das bobinas com a esfera, gerando uma analogia ao acelerador de partículas.

Como possibilidade futura de desenvolvimento deste experimento, podemos sugerir a inserção de mais três barreiras ópticas (conjunto LED e fototransistor) em uma das bobinas, possibilitando a medição da velocidade da esfera na entrada e na saída desta bobina, como esquematizado a seguir:

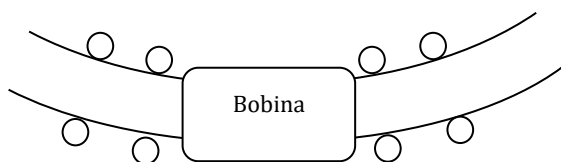


Figura 54 - Proposta para montagem dos sensores ópticos para a medição da velocidade da esfera na entrada e na saída da bobina

Com esta alteração, podemos determinar a perda de energia associada ao atrito, rotação, etc., calculando-se a diferença entre a energia cinética da esfera, por meio do uso do Teorema Trabalho-Energia Cinética, e a energia consumida pela bobina.

Outra possibilidade de desenvolvimento futuro para este experimento é a montagem de um *Coilgun* usando os dispositivos/componentes presentes no próprio experimento e realizando pequenas alterações de software no *Arduino* e no *Processing*. O *Coilgun* ou Canhão de Gauss é um dispositivo que opera a partir do efeito de movimento mecânico (“gun”) gerado a partir da interação de uma bobina (“coil”) sobre um projétil ferromagnético.

Concluimos que a plataforma *Arduino* também pode ser utilizada em aplicações para controle de sistemas, suplantando a função de adquirir os dados oriundos dos sensores e transmitir para a interface *Processing*, conforme fora abordado nos capítulos anteriores.

---

**Experiência docente do uso do *Arduino* e do *Processing* no Ensino de Física**

A aplicação dos objetos educacionais foi realizada com alunos do Curso Técnico em Eletrônica da Escola Técnica Estadual Ferreira Viana, em uma turma de doze alunos. No início de fevereiro de 2016, os alunos começaram a manusear e a apreender como usar o *Arduino*. Em março desse mesmo ano, teve início a aplicação dos objetos. No total, foram quinze encontros de dois tempos cada, uma parte dos horários da disciplina de Laboratório.

Nestes encontros, os alunos trabalharam em pequenos grupos desenvolvendo a construção dos três protótipos dos experimentos propostos nessa dissertação. As aulas foram desenvolvidas no Laboratório de Simulação Eletrônica, que possui uma infraestrutura muito próxima a de um Laboratório de Informática, conforme mostra a figura a seguir:



Figura 55 - Vista parcial do Laboratório de Simulação Eletrônica

No primeiro encontro, os estudantes foram informados do objetivo do projeto e foram apresentados à plataforma *Arduino* e à plataforma *Moodle*<sup>38</sup>. Antes de iniciar a montagem dos protótipos, os alunos executaram o procedimento de cadastro e realizaram um questionário diagnóstico, via plataforma Moodle, contendo perguntas sobre a sua familiaridade com alguns recursos tecnológicos e computacionais. Este questionário foi

---

<sup>38</sup> MOODLE é o acrônimo de "Modular Object-Oriented Dynamic Learning Environment", um software livre, de apoio à aprendizagem. O Moodle utilizado nas atividades desenvolvidas nesta dissertação pode ser acessado em: [www.aulanaweb.com](http://www.aulanaweb.com).

aplicado com a intenção de traçar um perfil, ainda que sem grande aprofundamento, da turma e iniciar a ambientação dos alunos à plataforma *Moodle*.

No segundo encontro, foi realizado uma roda de discussão sobre os conceitos Físicos básicos relacionados ao oscilador amortecido, à estação meteorológica e ao acelerador de partículas, com a finalidade de verificar o conhecimento prévio dos estudantes a respeito dos assuntos e, também, para fazer um comparativo com a avaliação final, que seria realizado no último dia de aplicação dessa proposta pedagógica. Os três roteiros de atividades foram inseridos na plataforma Moodle e também foram criados fóruns e grupos de discussão para que os alunos pudessem trabalhar colaborativamente dentro e fora do horário escolar, com a participação nestes recursos da própria plataforma.

Também foi solicitado aos alunos que interagissem com as mídias educacionais, simulações e vídeos, dispostos na plataforma *Moodle* e que trouxessem dúvidas, avaliassem o referido material e sugerissem questões para serem discutidas em sala de aula no terceiro encontro.

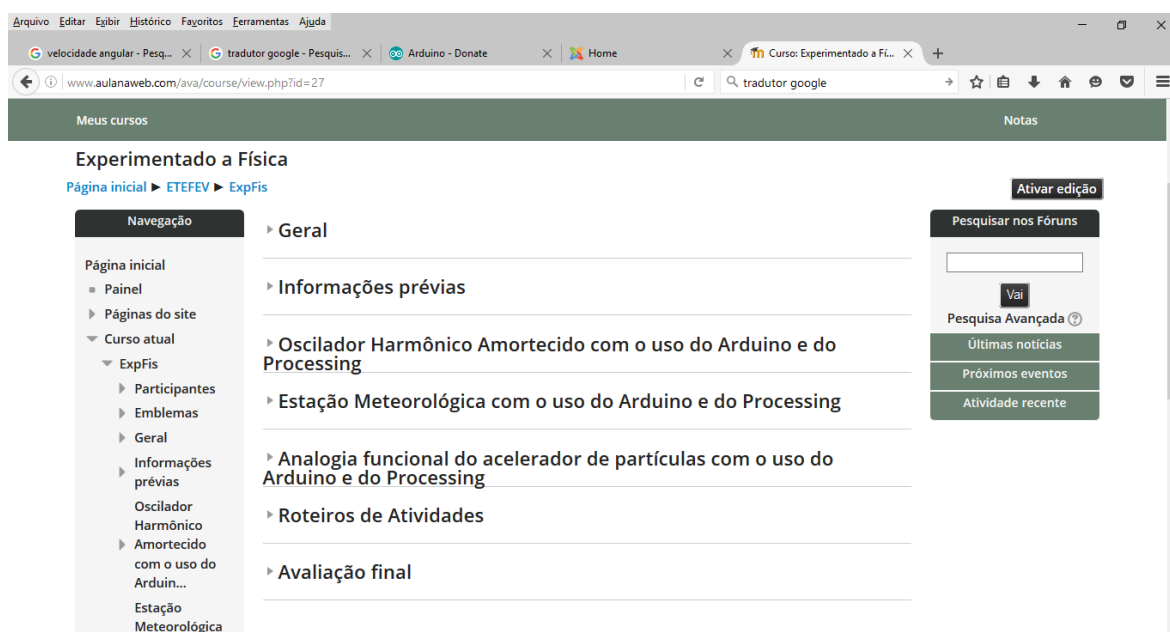


Figura 56

No quarto encontro, a turma foi dividida em três grupos, e, cada grupo, recebeu o os componentes referentes a cada um dos experimentos para o desenvolvimento da montagem dos protótipos. Esta aula também contou com uma breve explicação sobre os componentes e materiais utilizados.

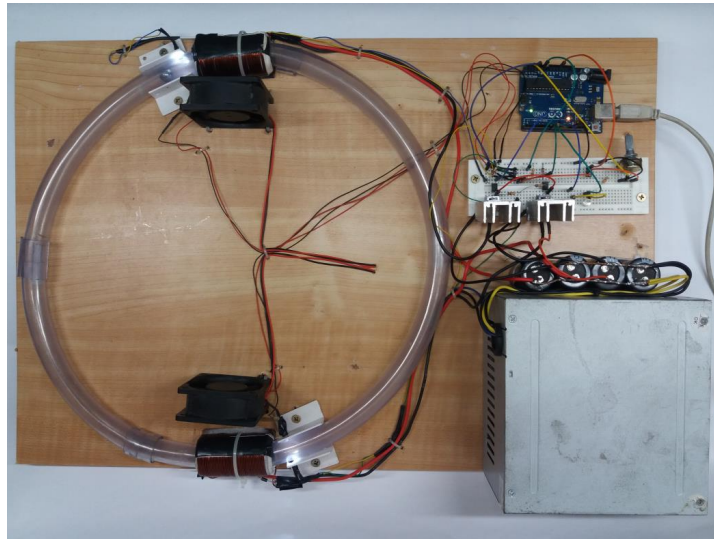


Figura 57 – Protótipo do acelerador de partículas com o uso do *Arduino* e do *Processing*

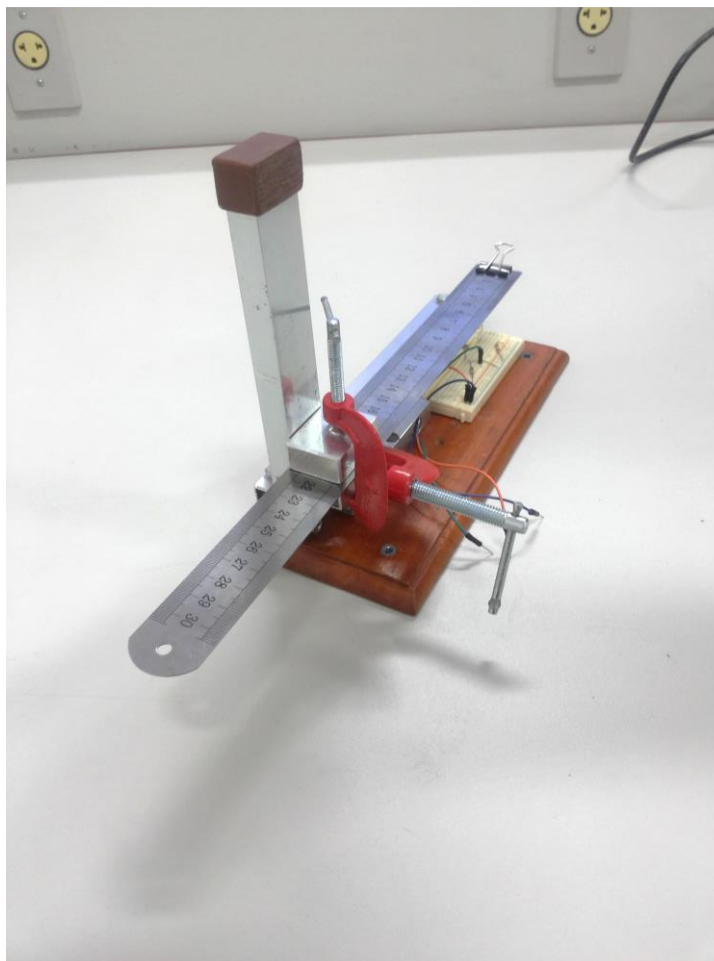


Figura 58 – Protótipo do oscilador harmônico amortecido com o uso do *Arduino* e do *Processing*

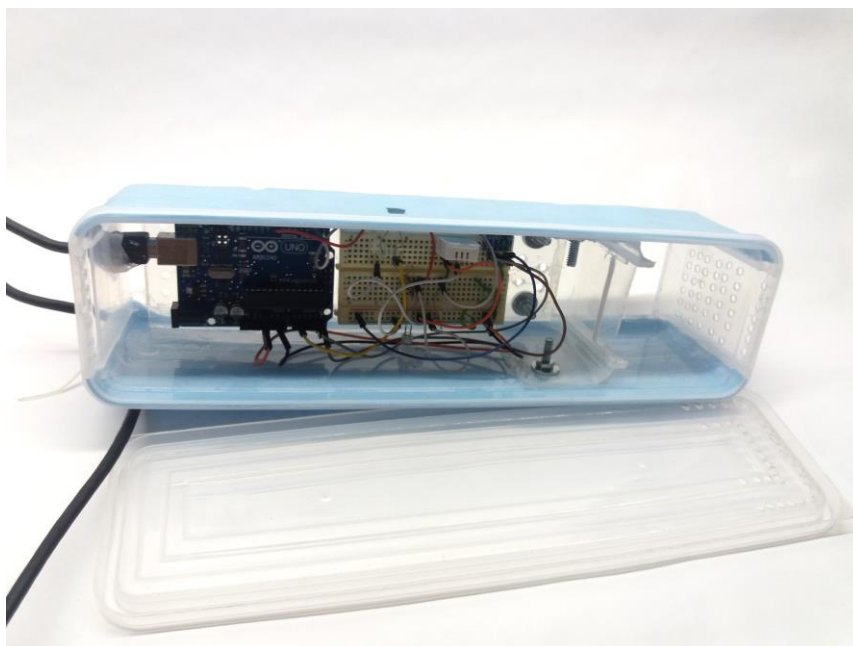


Figura 59 – Protótipo da estação meteorológica com o uso do *Arduino* e do *Processing*

Mais dois encontros foram destinados para ao desenvolvimento de atividades na plataforma *Arduino*. Foram montados duas pequenas aplicações presentes no menu Exemplos - basics da própria IDE: a *Blink*, empregando um LED para implementar um piscapisca e a *AnalogReadSerial*, estabelecendo a comunicação serial com a leitura da resistência de um LDR. Estas montagens permitiram a ambientação dos alunos ao *Arduino*, ao software de programação e às montagens eletroeletrônicas na matriz de contatos. Cabe ressaltar que o interesse dos alunos pelo *Arduino* superou às expectativas. Alguns alunos até adquiriram seu próprio *Arduino UNO* e desenvolveram diversas montagens em paralelo as atividades propostas relacionadas a outras disciplinas do curso durante o projeto.

Os quatro encontros subsequentes foram dedicados ao desenvolvimento das montagens mecânica e eletrônica dos experimentos propostos, à programação do *Arduino* e à implementação da interface *Processing*. Os três objetivos foram alcançados com certa facilidade devido ao interesse e a habilidade dos alunos.

No décimo primeiro encontro, o grupo 1 instalou o protótipo da estação meteorológica na parte externa da janela do laboratório e iniciou a aquisição dos dados através da interface *Processing*. O grupo 2 realizou a pesagem das moedas para determinar a constante elástica da régua e iniciou os procedimentos de calibração do sensor óptico do protótipo do oscilador harmônico amortecido; e o grupo 3 utilizou esta aula para realizar a medição da corrente de pulso na bobina no protótipo do acelerador de partículas.



Figura 60 – Desenvolvimento dos experimentos em sala de aula

Com os protótipos montados e em funcionamento, as próximas aulas foram dedicadas às apresentações do material e dos dados obtidos por cada um dos grupos, à realização das tarefas propostas nos roteiros de atividade e a uma breve avaliação. É possível ressaltar que os alunos puderam comprovar, na prática, os conceitos abordados, anteriormente, em aulas puramente expositivas.

Uma comparação entre o primeiro momento de avaliação realizado no segundo encontro na atividade roda de discussão e a avaliação final pode comprovar a eficácia do uso dos experimentos com *Arduino* e *Processing* como ferramentas educacionais. Os temas foram debatidos de forma mais aprofundada e satisfatória após o desenvolvimento e apresentação dos protótipos em sala de aula pelos próprios alunos, que mostraram grande aptidão e satisfação com a apropriação dos dispositivos tecnológicos.



Apresentamos, neste trabalho, três objetos educacionais com o uso do *Arduino* e do *Processing*, que possibilitam, a baixo custo, realizar experimentos com adequada medição de variáveis físicas e apresentação destas medições em uma interface computacional. Esta interface computacional, comprovadamente, permite a apresentação dos dados de forma textual ou de forma gráfica, bem como o registro destes dados em arquivos do tipo texto para futuras análises.

Tais objetos estão acompanhados de atividades didáticas complementares que possibilitam a análise dos dados obtidos como principal ferramenta pedagógica para a introdução de alguns conceitos de Física aos alunos do ensino médio. Estes roteiros foram desenvolvidos com a colaboração de alguns professores da Escola Técnica Estadual Ferreira Viana e foram aplicados com sucesso em uma das turmas do Curso Técnico em Eletrônica da mesma Instituição de Ensino.

A proposta de montagem do experimento de forma colaborativa com os alunos, oportunizou a experiência de lidar com situações, fossem elas cotidianas ou acadêmicas, exigindo deles soluções ou elaboração de estratégias e de planos de ação para resolvê-las. Com estas práticas, pudemos desenvolver o interesse e a curiosidade do grupo, fundamentais para a investigação científica e o desenvolvimento tecnológico, mas também favorecemos o desenvolvimento das habilidades e das competências necessárias para agir de forma objetiva na busca de uma solução para um desafio original dentro ou fora do contexto escolar.

Também pretendemos que estes objetos educacionais possam ser aprimorados, adaptados para outras realidades ou serem o ponto de partida para a criação de novos experimentos relacionados aos tópicos curriculares do Ensino de Física nas turmas de Ensino Médio.

Esperamos, então, que a iniciativa proposta por esta dissertação seja, de fato e a curto prazo, uma opção economicamente viável para a ausência ou deficiência de Laboratórios de Física nas escolas públicas ou privadas de Ensino Médio possibilitando o que é preconizado pelos PCN's:

É indispensável que a experimentação esteja sempre presente ao longo de todo o processo de desenvolvimento das competências em Física, privilegiando-se o fazer, manusear, operar, agir, em diferentes formas e níveis. É dessa forma que se pode garantir a construção do conhecimento pelo próprio aluno, desenvolvendo sua curiosidade e o hábito de sempre indagar, evitando a aquisição do conhecimento científico como uma verdade estabelecida e inquestionável. (PCN+, 2013, p.81)

ANDRADE, B. L.; ZYLLBERSZTAJN, A.; FERRARI, N. Analogias e metáforas no ensino de ciências à luz da epistemologia de Gaston Bachelard. **Ensaio Pesquisa em Educação em Ciências**, Belo Horizonte, v. 2, n. 2, p. 231- 245, 2000.

AOSONG, Electronics Co.,Ltd. **Temperature and humidity module – AM2302**. Disponível em: < <http://akizukidenshi.com/download/ds/aosong/AM2302.pdf>>. Acesso em 10 abr. 2016.

ARAUJO, I. S.; VEIT, E. A.; MOREIRA, M. A. Atividades de modelagem computacional no auxílio da interpretação de gráficos da cinemática. **Revista Brasileira de Ensino de Física**, v. 26, n. 2, p. 179-184, 2004.

ARAUJO, I. S.; VEIT, E. A.; MOREIRA, M. A. Physics students' performance using computational modelling activities to improve kinematics graphs interpretation. **Computers & Education**, Amsterdam, v. 50, n. 4, p. 1128-1140, may 2008.

ARAUJO, I. V.; VEIT, E. A.; MOREIRA, M. A. **Uma revisão da literatura sobre estudos relativos a tecnologias computacionais no ensino de Física**. Disponível em: <<http://www.if.ufrgs.br/cref/ntef/producao/IIIEIBIEC.PDF>>. Acesso em: 24 nov. 2015.

ARDUINO. Disponível em: < <https://www.arduino.cc/>>. Acesso em: 3 de jul. 2016.

BAUER, W.; WESTFALL, G. D.; DIAS, H. **Física para Universitários – Relatividade, Oscilações, Ondas e Calor**. São Paulo: Mc Graw Hill, 2013.

BOSCH. **Digital Barometric Pressure Sensor – BMP180**. Disponível em: <[https://www.bosch-sensortec.com/bst/products/all\\_products/bmp180](https://www.bosch-sensortec.com/bst/products/all_products/bmp180)>. Acesso em 20 abr. 2016.

BRASIL. Ministério da Educação, Secretaria de Educação Média e Tecnológica. **Base Nacional Comum**: Ministério da Educação, 2016.

BRASIL. Ministério da Educação, Secretaria de Educação Média e Tecnológica. **Parâmetros Curriculares Nacionais**: ensino médio. Brasília: Ministério da Educação, 2000.

BRASIL. Ministério da Educação, Secretaria de Educação Média e Tecnológica. **Orientações Educacionais Complementares aos Parâmetros Curriculares Nacionais (PCN+)**: ensino médio. Brasília: Ministério da Educação, 2013.

CARVALHO, L. R. M.; AMORIM, H. S. Observando as marés atmosféricas: uma aplicação da placa Arduino com sensores de pressão barométrica e temperatura. **Revista Brasileira de Ensino de Física**, São Paulo, v. 36, n. 3, set. 2014.

CASTRO, L. H. M. de; LAGO, B. L.; MONDAINI, F. Damped Harmonic Oscillator with Arduino. **Journal of Applied Mathematics and Physics**, v. 3 n. 6, jun. 2015.

CAVALCANTE, M. A. Novas tecnologias no estudo de ondas sonoras. **Caderno Brasileiro de Ensino de Física**, Santa Catarina, v. 30, n. 3, p. 579-613, dez. 2013.

CAVALCANTE, M. A.; BONIZZIA, A.; GOMES, L. C. P. O ensino e aprendizagem de física no século XXI: sistemas de aquisição de dados nas escolas brasileiras, uma possibilidade real. **Revista Brasileira de Ensino de Física**, São Paulo, v. 31, n. 4, dez. 2009.

CAVALCANTE, M. A.; RODRIGUES, T. T. T.; BUENO, D. A. Controle remoto: princípio de funcionamento (parte 1 de 2). **Caderno Brasileiro de Ensino de Física**, Santa Catarina, v. 30, n. 3, p. 554-565, dez. 2013.

CAVALCANTE, M.A.; RODRIGUES, T. T. T.; BUENO, D. A. Controle remoto: observando códigos com o Arduino (parte 2 de 2). **Caderno Brasileiro de Ensino de Física**, Santa Catarina, v. 31, n. 3, p. 614-641, dez. 2014.

CAVALCANTE, M. A.; TAVOLARO, C. R. C. Projete você mesmo experimentos assistidos por computador: construindo sensores e analisando dados. **Revista Brasileira de Ensino de Física**, São Paulo, v. 22, n. 3, p. 421-425, set. 2000.

CAVALCANTE, M. A.; TAVOLARO, C. R. C.; MOLISANI, E. Física com Arduino para iniciantes. **Revista Brasileira de Ensino de Física**, São Paulo, v. 33, n. 4, dez. 2011.

DORNELES, P. F.T. **Integração entre atividades computacionais e experimentais como recurso instrucional no ensino de eletromagnetismo em Física Geral**. 2010. 367 f. Tese (Doutorado em Ciências) - Instituto de Física, UFRGS, Porto Alegre, 2010.

EVANS, B. **Beginning Arduino Programming**. New York, USA: Apress, 2010.

EVANS, M.; NOBLE, J.; HOCHENBAUM, J. **Arduino em ação**. São Paulo: Novatec, 2013.

FILLAT, A.P.; ARQUÉS, C.C. **Acelerador de partículas - Maqueta educativa**. III Concurso de Divulgación CPAN, Espanha, 2012.

FRITZING. Disponível em: <[www.fritzing.org](http://www.fritzing.org)>. Acesso em: 25 nov. 2015.

GREENBERG, I; XU, D.; KUMAR, D. **Processing: Creative Coding and Generative Art in Processing 2**. New York, USA: Apress, 2013.

GRIFFITHS, D. J. **Eletrodinâmica**, São Paulo: Pearson, 2014.

HAAG, R.; ARAUJO, I. S.; VEIT, E. A. Por que e como introduzir a aquisição automática de dados no laboratório didático de Física. **A Física na Escola**, São Paulo, v. 6, n. 1, p. 89-94, maio 2005. Disponível em: <<http://sbfisica.org.br/fne/Vol6/Num1/aquisicao.pdf>>. Acesso em: 24 nov. 2015.

HALLIDAY, D.; RESNICK, R.; MERRILL, J. **Fundamentos de física: eletromagnetismo**. Rio de Janeiro: LTC, 1993.

HALLIDAY, D.; RESNICK, R. **Fundamentos de física: gravitação, ondas e termodinâmica**. 9. ed. Rio de Janeiro: LTC, 2003.

HALLIDAY, D.; RESNICK, R.; WALKER, J. **Fundamentos de física: gravitação, ondas e termodinâmica**. 9. ed. Rio de Janeiro: LTC, 2012.

LEE, Shyh-Yuan. **Accelerator Physics**. New Jersey, USA: World Scientific, 2004.

LIST, R. J. **Smithsonian Meteorological Tables**. Washington, USA: Smithsonian Institution, 1971.

LUIZ, L. C.; MONTEIRO, K. T. S.; BATISTA, R. T. Os aceleradores de partículas e sua utilização na produção de radiofármacos. **Revista Brasileira de Fármacos**, 2011.

MAXIMO, A.; ALVARENGA, B. **Física**. Rio de Janeiro: Scipione, 2002.

MCROBERTS, M. **Arduino básico**. Tradução Rafael Zanolli. São Paulo: Novatec, 2011.

MCROBERTS, M. **Beginning Arduino**. New York, USA: Apress, 2010.

MINIPA. **Manual de Instruções - Termo Higrômetro Digital MT-240**. Disponível em: <<http://www.minipa.com.br/Content/Manuais/MT-240-1101-BR-EN-ES.pdf>>. Acesso em 16 mar. 2016.

MINIPA. **Manual de Instruções - Luxímetro - MLM-1011**. Disponível em: <<http://www.minipa.com.br/Content/Manuais/MLM-1011-1102-BR-EN-ES.pdf>>. Acesso em 16 mar. 2016.

MONK, S. **30 Arduino Projects for the Evil Genius**. New York, USA: Mc Graw Hill, 2010.

NOBLE, J. **Programming Interactivity: A Designer's Guide to Processing, Arduino, and openFrameworks**. California, USA: O'Reilly & Associates Inc, 2012.

OBSERVATÓRIO DO PNE. **Indicadores de acompanhamento das metas e estratégias - Porcentagem de escolas da Educação Básica com acesso à água tratada, esgoto sanitário, energia elétrica, banda larga, biblioteca ou sala de leitura, quadra e laboratório de ciências**. Brasília, 2014.

OLIVEIRA, C.L.V; ZANETTI, H.A.P. **Arduino Descomplicado - Como Elaborar Projetos de Eletrônica**. São Paulo, Érica, 2015.

OXER, J; BLEMINGS, H. **Practical Arduino**. New York, USA: Apress, 2009.

PEROTONI, M.B.; MERGL, M. Desenvolvimento e análise de um Protótipo Coilgun. **Revista Brasileira de Ensino de Física**, São Paulo, v. 38, n. 2, abr. 2016.

PURDUM, J. **Beginning C for Arduino**. New York, USA: Apress, 2012.

PROCESSING. Disponível em: <<https://processing.org/>>. Acesso em: 3 de jul. 2016.

REAS, C.; FRY, B. **Processing: A Programming Handbook for Visual Designers and Artists**. Cambridge, USA: MIT Press, 2014.

RIBEIRO, J. C. O ensino experimental da Física no curso secundário. **II Curso de aperfeiçoamento para professores de Física do ensino secundário**. Atas do encontro. Pág.: 49-56. IBCEC. MEC-ITA. São Paulo, 1955.

ROCHA, F. S.; GUADAGNINI, P. H. Projeto de um sensor de pressão manométrica para ensino de física em tempo real. **Caderno Brasileiro de Ensino de Física**, Santa Catarina, v. 31, n. 1, p. 124-148, abr. 2014.

ROCHA, F. S.; MARRANGHELLO, G. F.; LUCHESE, M. M. Acelerômetro eletrônico e a placa Arduino para ensino de Física em tempo real. **Caderno Brasileiro de Ensino de Física**, Santa Catarina, v. 31, n. 1, p. 98-123, abr. 2014.

SHIFFMAN, D. **Learning Processing, Second Edition: A Beginner's Guide to Programming Images, Animation, and Interaction**. California, USA: Morgan Kaufmann Publishers, Inc, 2015.

SOUZA, A. R.; PAIXÃO, A. C.; UZÊDA, D. D.; DIAS, M. A.; DUARTE, S.; AMORIM, H. S. A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC. **Revista Brasileira de Ensino de Física**, São Paulo, v. 33, n. 1, 4503, mar. 2011.

STEERE, Andrew Robert. **A timeline of major particle accelerators**, Thesis submitted to Michigan State University, 2005.

STEINKE, E.T. **Climatologia Fácil**. São Paulo: Oficina de Textos, 2012.

STEVAN JR, S.L.; SILVA, R. A. **Automação e Instrumentação Industrial Com Arduino - Teoria e Projetos**. São Paulo, Érica, 2012.

STEWART, I. **Mania de Matematica 2 - Novos Enigmas e Desafios Matemáticos**. Rio de Janeiro: Zahar, 2009.

TIMMIS, H. **Practical Arduino Engineering**. New York, USA: Apress, 2012.

VAREJÃO-SILVA, M.A. **Meteorologia e Climatologia**. Recife, 2006.

VARELA, H. A análise da oscilação de ciclos como ferramenta para entender o presente e prever o futuro. **Instituto de Estudos Avançados da Universidade de São Paulo**. Disponível em: < <http://www.iea.usp.br/noticias/oscilacoes-de-ciclos>>. Acesso em 20 mai. 2016.

VEIT, E. A. Modelagem computacional no ensino de Física. In: SIMPÓSIO NACIONAL DE ENSINO DE FÍSICA, 16., 2005, Rio de Janeiro.

VISHAY. **Reflective Optical Sensor with Transistor Output: TCRT5000, TCRT5000L**. Disponível em: <<http://www.vishay.com/docs/83760/tcrt5000.pdf>>. Acesso em 20 abr. 2016.

WILCHER, D. **Learn Eletronics with Arduino**. New York, USA: Apress, 2012.

YAMAMOTO, K.; FUKU, L. F. **Física para o Ensino Médio**. São Paulo: Saraiva, 2013.

## APÊNDICE A – Registro fotográfico dos protótipos dos experimentos

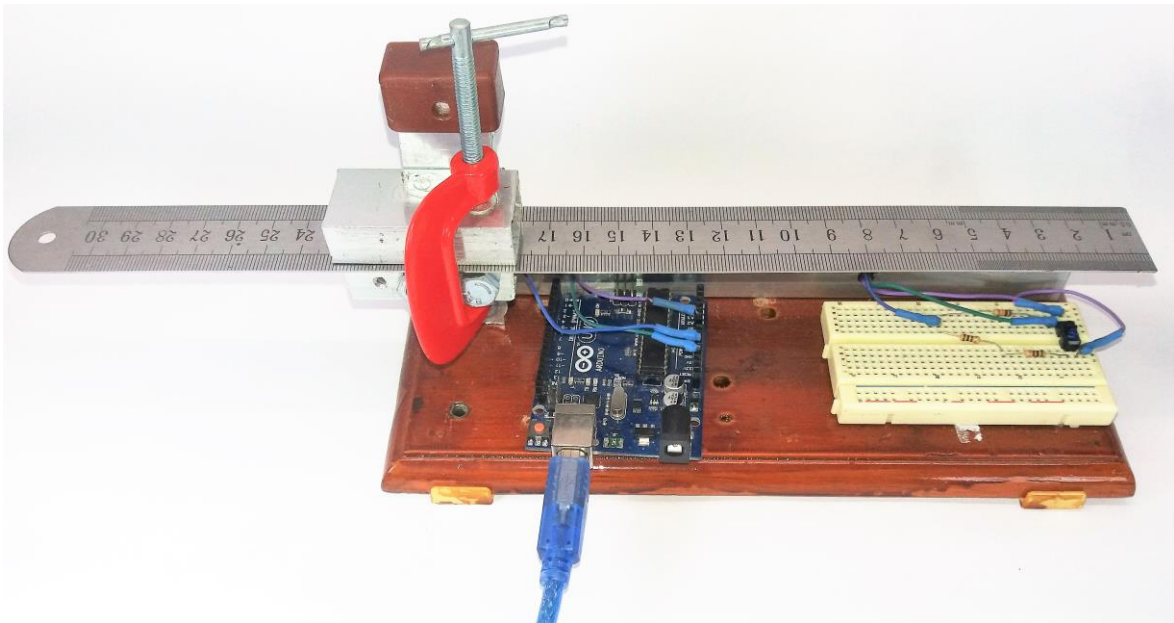


Figura 61 – Fotografia do experimento *Oscilador Harmônico Amortecido* com o uso do *Arduino* e do *Processing*

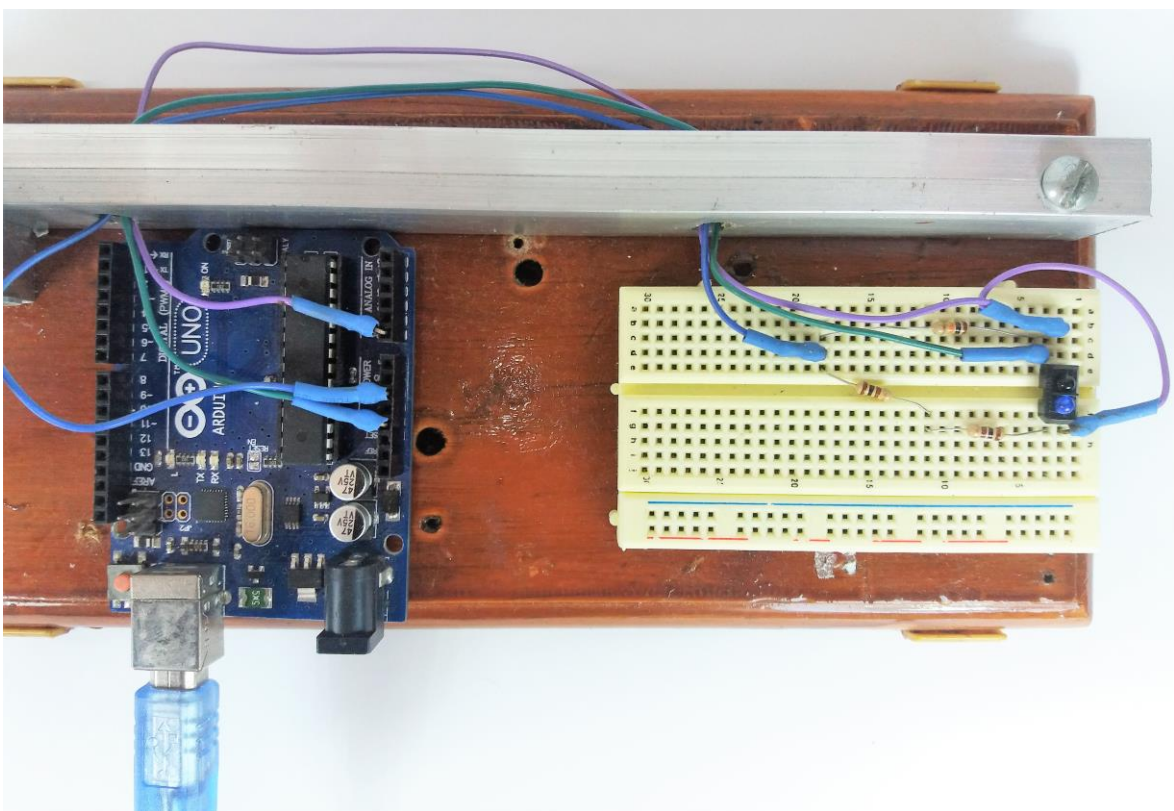


Figura 62 – Fotografia da montagem eletrônica do experimento *Oscilador Harmônico Amortecido* com o uso do *Arduino* e do *Processing*

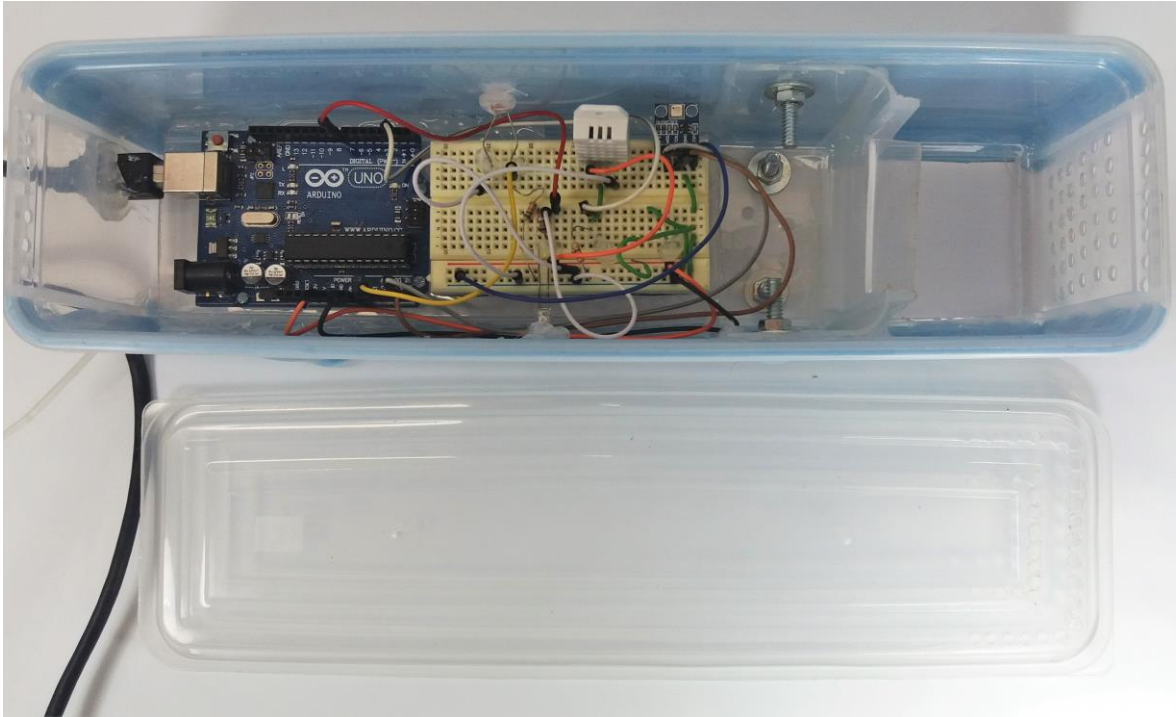


Figura 63 – Fotografia da montagem do experimento Estação Meteorológica com o uso do *Arduino* e do *Processing*

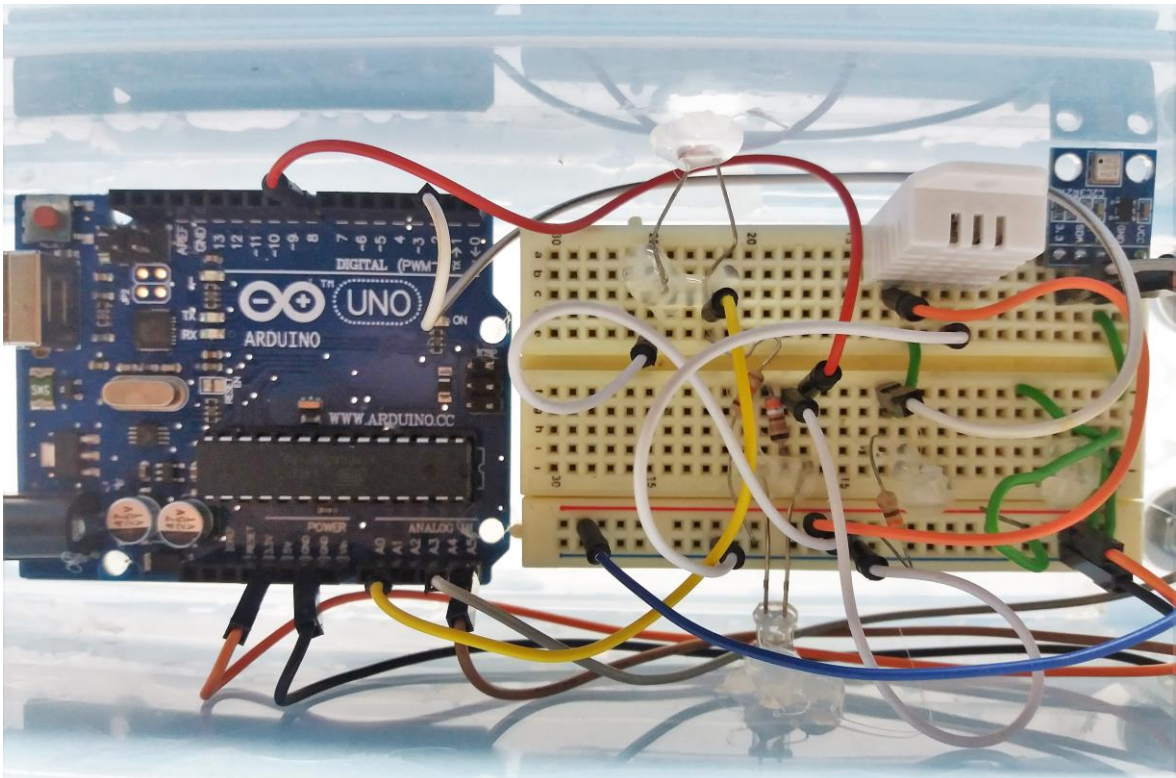


Figura 64 – Fotografia da montagem eletrônica do experimento Estação Meteorológica com o uso do *Arduino* e do *Processing*

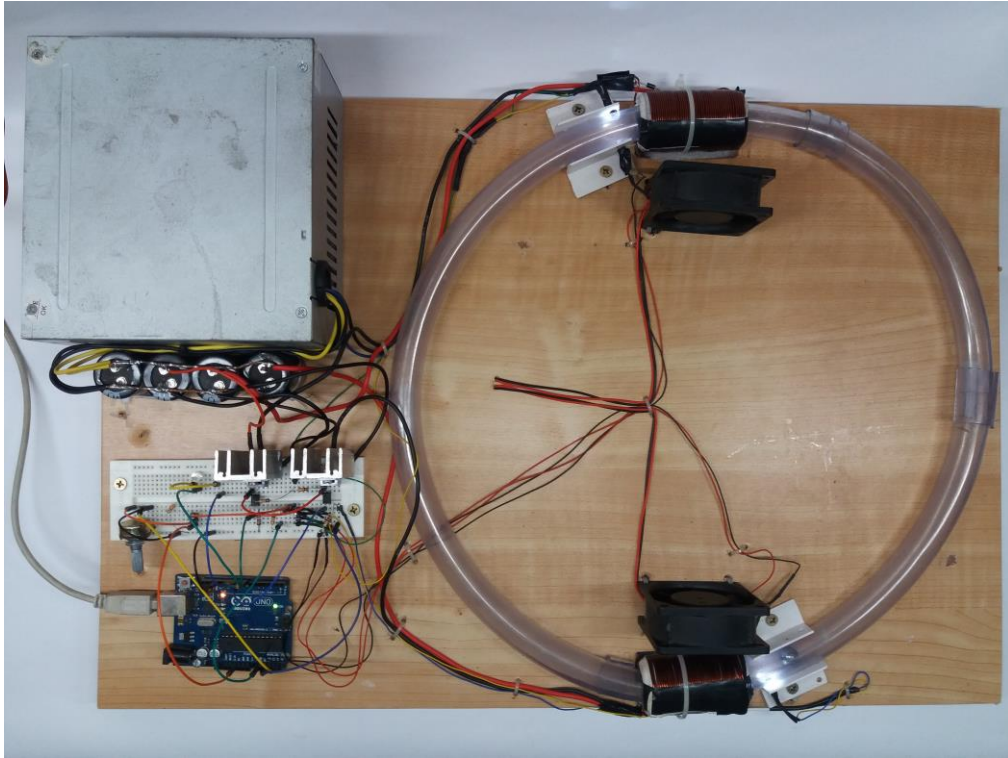


Figura 65 – Fotografia da montagem do experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*

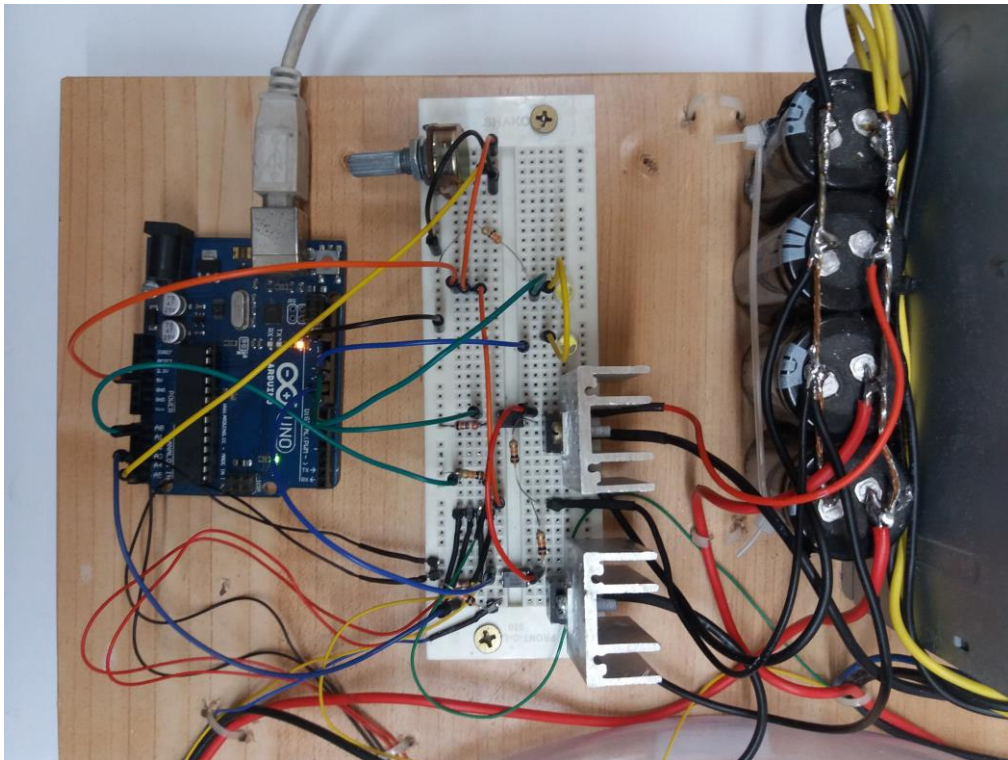


Figura 66 – Fotografia da montagem do experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*



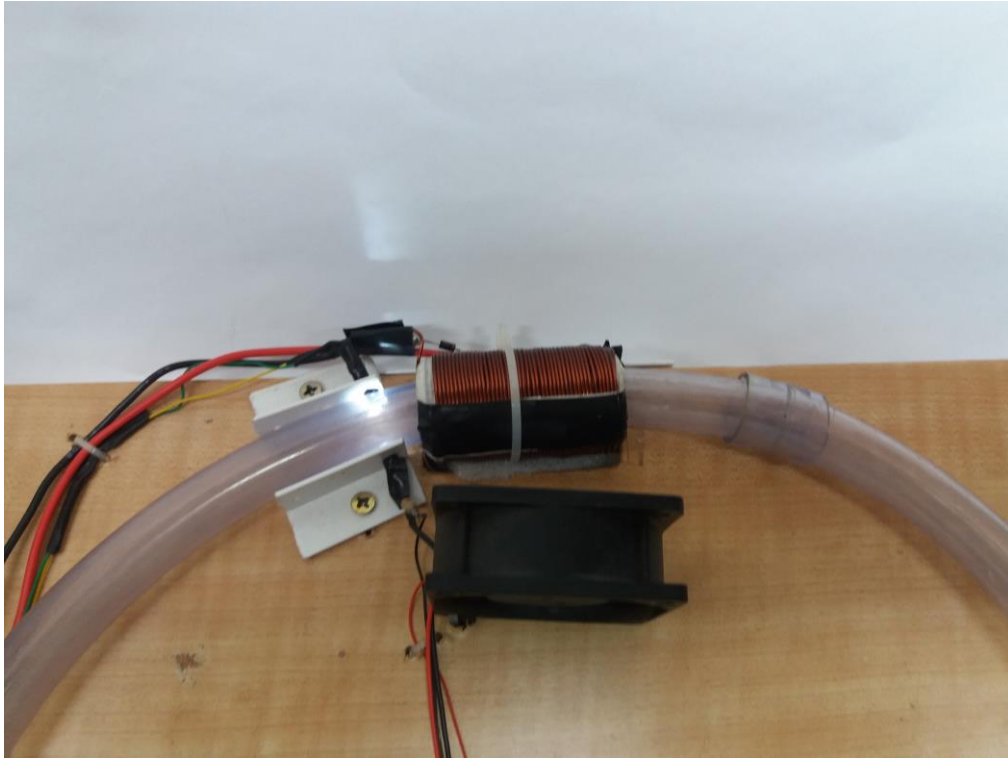


Figura 67 – Fotografia do detalhe da montagem da barreira de luz, da bobina e do ventilador de resfriamento no experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*

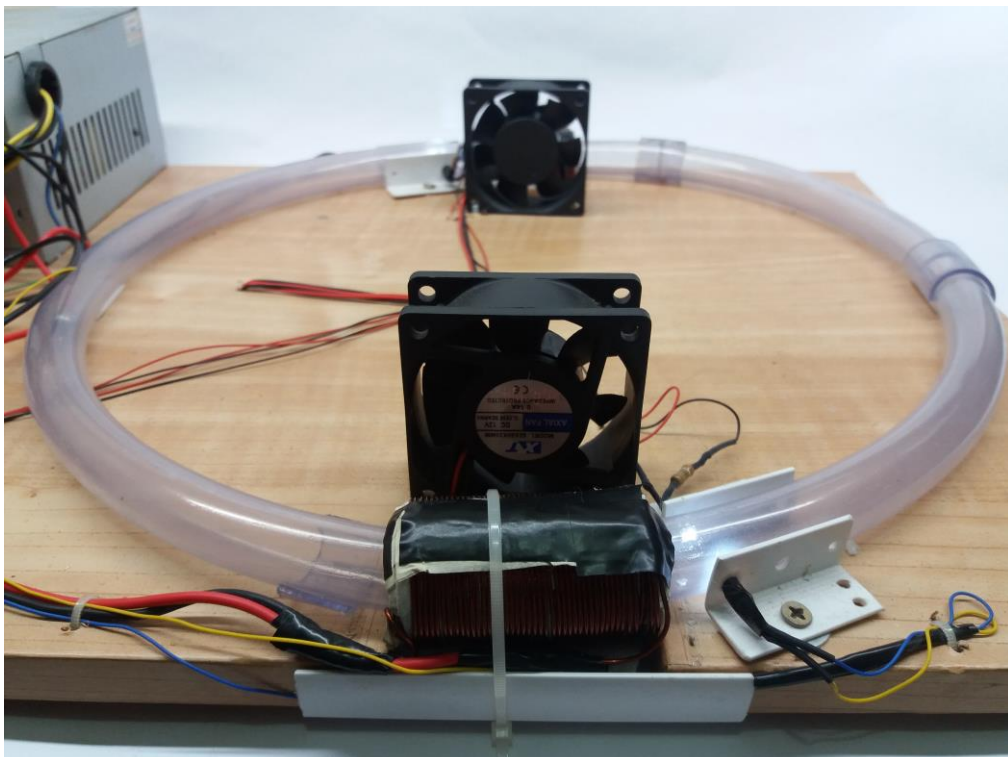


Figura 68 – Fotografia da montagem do conjunto bobina e do ventilador de resfriamento no experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*

## APÊNDICE B – Roteiro de atividades para os experimentos

---

O ensino de ciências, e principalmente o Ensino de física, deve estar amparado em práticas pedagógicas que favoreçam a associação entre o ensino teórico, ministrado em sala de aula, e a prática desenvolvida, preferencialmente, nos laboratórios. As atividades práticas são o momento pedagógico de comprovação e de sedimentação dos modelos apresentados aos alunos nas atividades teóricas.

Para cumprir este objetivo, são apresentados, neste material, os modelos propostos para os roteiros dos Produtos Educacionais, compostos de: listagem do material necessário, os diagramas de montagem e ligação eletroeletrônica, os códigos do *Arduino* e os códigos do *Processing* e sugestões de atividades.

Para a aplicação em sala de aula dos experimentos sugeridos, o professor de Física do Ensino Médio deverá reservar um tempo do planejamento de suas aulas para a montagem e o teste de cada um dos experimentos.

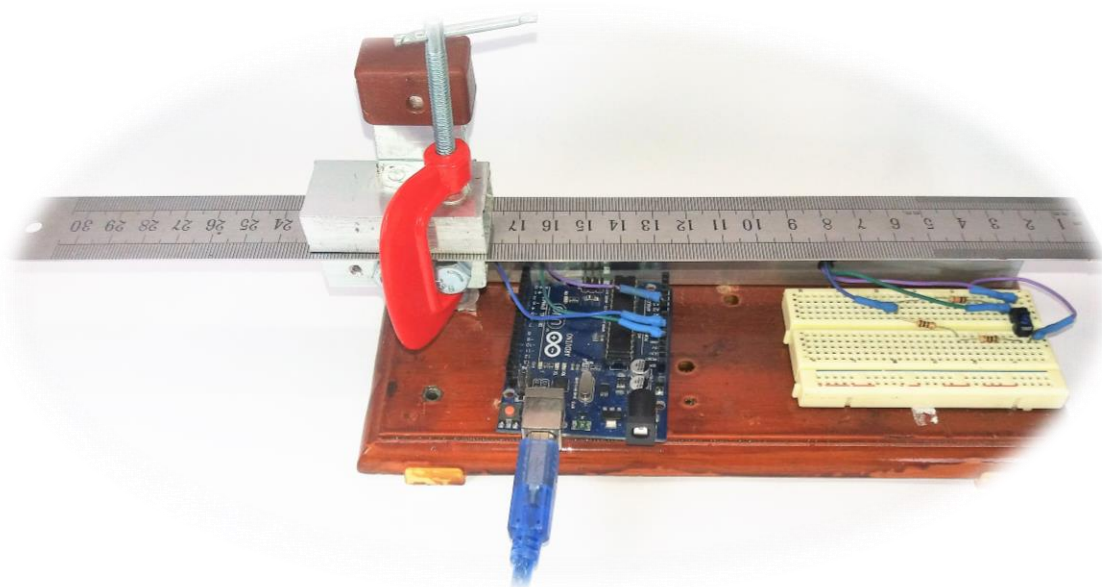
Para balizar o professor, neste projeto foram utilizadas 4 horas para a montagem do experimento *Oscilador Harmônico Amortecido com o uso do Arduino e do Processing*, 6 horas para o experimento *Estação Meteorológica com o uso do Arduino e do Processing* e cerca de 10 horas para o experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*.

Os modelos propostos para os roteiros das atividades desenvolvidos para os experimentos abordados foram elaborados e apresentados a professores de Física da Escola Técnica Estadual Ferreira Viana e da Escola Técnica do Arsenal de Marinha. Esses roteiros foram testados com sucesso em uma turma de alunos do Curso Técnico em Eletrônica da Escola Técnica Estadual Ferreira Viana.

Todos as atividades são estruturadas com a apresentação das habilidades e competências trabalhadas, com os objetivos a serem alcançados no decorrer do experimento, com uma introdução teórica ao conteúdo, com o procedimento a ser adotado pelos alunos e com questões a serem respondidas e/ou discutidas pelos alunos.

# PRODUTO EDUCACIONAL

## Oscilador Harmônico Amortecido com o uso do Arduino e do Processing



Construção do experimento e roteiros de atividades

Luis Henrique Monteiro de Castro

Setembro de 2016

**MNPEF**  
Mestrado Nacional  
Profissional em  
Ensino de Física



## Sumário

Algumas Palavras .....	1
Materiais necessários .....	1
Montando o experimento .....	2
Código do Arduino .....	3
Código do Processing .....	3
Atividade 1: Demonstrar o uso do experimento .....	5
Atividade 2: Análise do fenômeno da deformação elástica da lâmina .....	7
Atividade 3: Análise do movimento oscilatório .....	10
Atividade 4: Análise do amortecimento .....	12

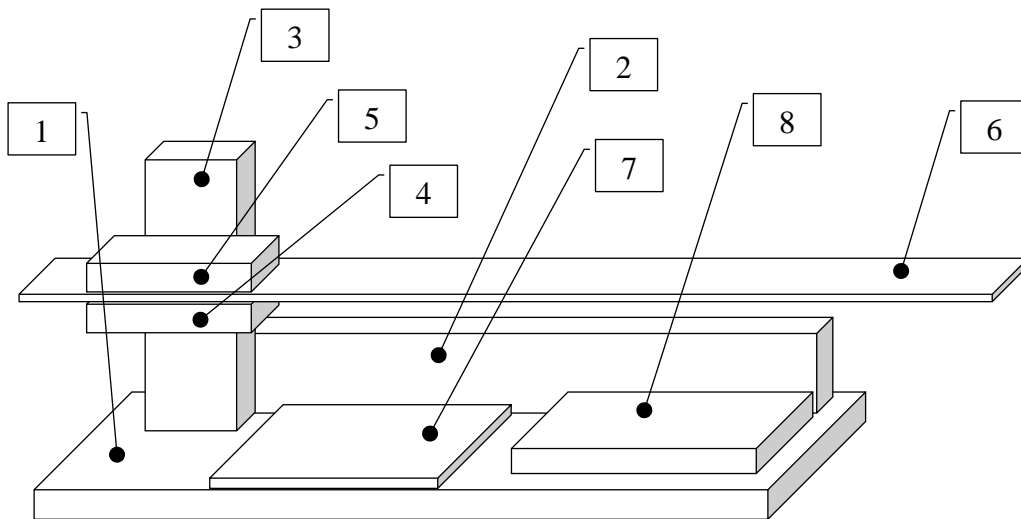
## Algumas Palavras

Este produto educacional é uma sugestão de uso do Arduino e do Processing como experimento para o Ensino de Física. Neste material professores e alunos encontrarão informações quanto aos materiais necessários para a construção mecânica e eletrônica do experimento, bem como os códigos de programação do Arduino e do Processing e ainda sugestões de atividades envolvendo os conceitos de tensão e deformação, constante elástica de uma mola e oscilações harmônicas amortecidas. O material foi desenvolvido considerando que o leitor tenha algum conhecimento, mesmo que rudimentar, de programação e de eletrônica. Todo o material e os códigos também estão disponibilizados no endereço [www.aulanaweb.com](http://www.aulanaweb.com).

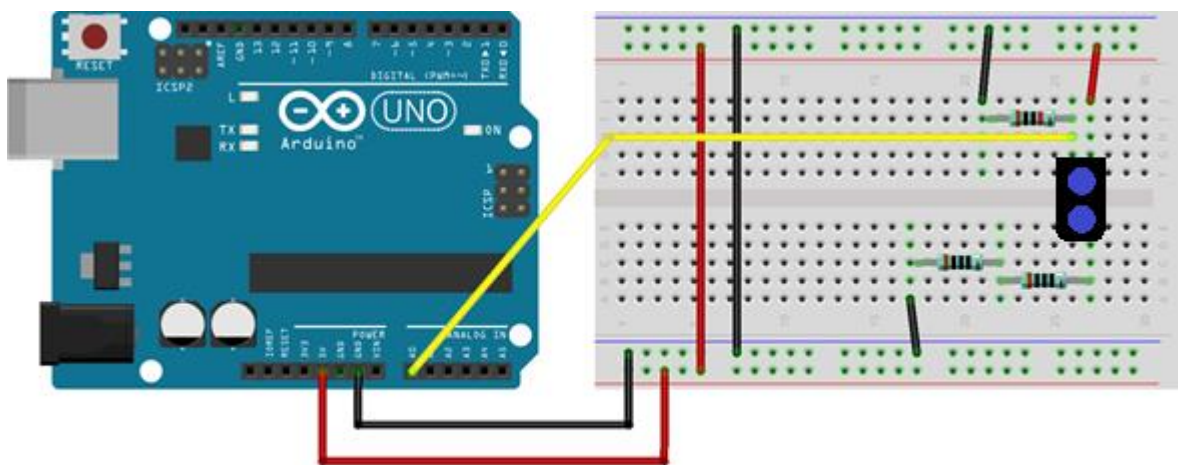
## Materiais necessários

Item	Descrição	Quantidade
1	Base de madeira medindo 23 cm x 10 cm x 1,2 cm	01
2	Perfil de alumínio (25,4 mm x 12,7 mm x 1,3 mm)	0,50 m
3	Parafusos 5mm x 40 mm com porca e arruela	04
4	Parafusos 6 mm x 50 mm com porca e arruela	02
5	Régua de aço com 30 cm de comprimento	01
6	Placa Arduino UNO	01
7	Protoboard 830 pontos – Matriz de contatos elétricos para montagem de circuitos	01
8	Grampo nº1 para fixação	01
9	TRCT-5000 – Sensor óptico reflexivo infravermelho	01
10	Resistor 10 k $\Omega$ x 1/8 w	01
11	Resistor 100 $\Omega$ x 1/8 w	02
12	Jumpers para as conexões elétricas	08

## Montando o experimento



1. Base de madeira medindo 23 cm x 10 cm x 1,2 cm (comp x larg x alt);
2. Perfil de alumínio (25,4 mm x 12,7 mm x 1,3 mm) com 21,5 cm de comprimento, fixado na base de madeira com 2 parafusos 5mm x 40 mm;
3. Perfil de alumínio (25,4 mm x 12,7 mm x 1,3 mm) com 9 cm de comprimento, fixado no perfil 2 com 2 parafusos 5mm x 40 mm;
4. Perfil de alumínio (25,4 mm x 12,7 mm x 1,3 mm) com 5 cm de comprimento, fixado no perfil 3 com 2 parafusos 6 mm x 50 mm;
5. Perfil de alumínio (25,4 mm x 12,7 mm x 1,3 mm) com 5 cm de comprimento
6. Régua de aço com 30 cm de comprimento;
7. Placa Arduino UNO; e
8. *Protoboard* com pelo menos 170 furos.



## Código do Arduino

```

int valor = 0, contador=0;
long tempoAtual = 0, tempoAnterior = 0, tempo = 0;
void setup(){Serial.begin(250000);}
void loop(){
tempoAtual = micros()-tempoAnterior;
valor = analogRead(0);
while ( tempoAtual < 993){tempoAtual = micros()-tempoAnterior;}
tempoAtual = micros()-tempoAnterior;
tempoAnterior = micros();
tempo = tempo+tempoAtual/1000;
Serial.print(valor); Serial.print(",");
contador = contador + 1;
if (contador == 10){
//Serial.print(tempoAtual); Serial.print(",");
Serial.println(tempo);
contador=0;
}
}

```

## Código do Processing

<pre> import processing.serial.*; // Variáveis PrintWriter arquivoGeral; String serialString = null; Serial portaSerial; int lf = 10; int contador = 0; float controle = 40; String nomeArquivo = ""; String[] dados; float valor; float valorCorrigido; float valorAnterior = 350; int divisor = 1; float incremento = 1; float incrementoAnterior = 1; String dados2; float tensao; float[] dados1 = new float[10]; float dados3, dados4, dados5; long tempo; boolean disparoUnico = false; float offset = 0; float teste; String teste2, teste3; //Configurações Gerais e ciclo único void setup() { nomeArquivo += "Oscilador-"; nomeArquivo += String.valueOf(day()); nomeArquivo += "-"; nomeArquivo += String.valueOf(month()); nomeArquivo += "-"; nomeArquivo += String.valueOf(year()); nomeArquivo += "-"; nomeArquivo += String.valueOf(hour()); nomeArquivo += "-"; nomeArquivo += String.valueOf(minute()); nomeArquivo += "-"; nomeArquivo += String.valueOf(second()); //Criação do Arquivo arquivoGeral = createWriter(nomeArquivo+".txt"); //configuração da comunicação serial printArray(Serial.list()); portaSerial = new Serial(this, Serial.list()[0], 250000); portaSerial.clear(); </pre>	<pre> text(nomeArquivo+".txt",245,580); tela(); stroke(255); } //ciclo infinito void draw() { if (portaSerial.available() &gt; 0) { serialString = portaSerial.readStringUntil(lf); if (serialString != null) { if (controle &lt; 50){controle += incremento;} if (controle &gt;= 50){ dados = split(serialString,','); for (int i = 0; i &lt;= 9; i = i+1) { dados2 = dados[i]; println (dados2); dados3 = int(dados2); tensao = int(dados2)*0.0048828125; dados4 = 50.075*exp(- dados3/71.122)+36.733*exp(- dados3/421.707)+14.485; dados4 = dados4 + (offset/10); dados4 = round(dados4*10); dados4 = dados4/10; if(dados4 &gt; 60){dados4 = 60;} if(dados4 &lt; 20){dados4 = 20;} float m = map(dados4, 20, 60, 550, 150); if(incremento != 0){ line(controle,valorAnterior,controle+inc remento,m); valorAnterior = m; fill(4, 79, 111); rect(625,560,100,30,10); rect(950,560,100,30,10); rect(1100,560,100,30,10); fill(255); text(tensao,680,580); teste2= str(dados4); teste = offset/10; teste3= str(teste); text(teste2,1000,580); //text(dados4,1000,580); text(teste3,1150,580); arquivoGeral.print(tempo); arquivoGeral.print(","); arquivoGeral.println(dados4); arquivoGeral.flush(); </pre>	<pre> stroke(94, 155, 168); } else{ line(50, i, 1050, i); } stroke(255); fill(255); } //função captura do acionamento do botão do mouse void mousePressed() { if (mouseX&gt;800 &amp;&amp; mouseX&lt;900 &amp;&amp; mouseY&gt;100 &amp;&amp; mouseY&lt;130) { incremento = incrementoAnterior; } if (mouseX&gt;950 &amp;&amp; mouseX&lt;1050 &amp;&amp; mouseY&gt;100 &amp;&amp; mouseY&lt;130) { incrementoAnterior = incremento; incremento = 0; } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;100 &amp;&amp; mouseY&lt;130) { tela(); controle = 50; } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;160 &amp;&amp; mouseY&lt;190) { tela(); disparoUnico = false; if (incremento == 0){ incremento = incrementoAnterior; } controle = 50; botao(1100,160,"Normal",11); botao(1100,200,"Único",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;200 &amp;&amp; mouseY&lt;230) { botao(1100,160,"Normal",1); botao(1100,200,"Único",11); disparoUnico = true; incrementoAnterior = incremento; } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;260 &amp;&amp; mouseY&lt;290) { incremento = 0.025; tela(); controle = 50; botao(1100,260,"2 s/div",11); </pre>
--	--	---

<pre> serialString = portaSerial.readStringUntil(1f); serialString = null; //configuração da tela size(1250, 600); frameRate(200); noSmooth(); background(4, 79, 111); stroke(84, 145, 158); fill(255); textSize(12); text("Universidade Federal do Estado do Rio de Janeiro - UNIRIO", 50, 40); text("Mestrado Nacional Profissional de Ensino de Física - MNPEF", 50, 55); text("Orientador: Felipe Mondaini", 50, 70); text("Mestrando: Luis Henrique M de Castro", 50, 85); textSize(20); fill(102); stroke(0); rect(80, 95, 650, 35); fill(255); textAlign(CENTER); text("Interface Computacional - Oscilador Harmônico Amortecido", 405, 120); textSize(14); text("Disparo:",1150,150); text("Varreduras:",1150,250); text("Valor Instantâneo (volts):",525,580); text("Valor Instantâneo (mm):",855,580); text("Arquivo:",80,580); text("Offset:",1150,510); text("60",30,155); text("55",30,205); text("50",30,255); text("45",30,305); text("40",30,355); text("35",30,405); text("30",30,455); text("25",30,505); text("20",30,555); text("+20",1070,155); text("+15",1070,205); text("+10",1070,255); text("+5",1070,305); text("0",1070,355); text("-5",1070,405); text("-10",1070,455); text("-15",1070,505); text("-20",1070,555); fill(44, 105, 118); stroke(255); botao(1100,160,"Normal",11); botao(1100,200,"Único",1); botao(1100,260,"2 s/div",1); botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); botao(800,100,"START",2); botao(950,100,"STOP",3); botao(1100,100,"RESET",4); botao2(1100,520,"+",1); botao2(1160,520,"-",1); fill(4, 79, 111); rect(120,560,250,30,10); rect(625,560,100,30,10); rect(950,560,100,30,10); rect(1100,560,100,30,10); fill(255); </pre>	<pre> tempo = tempo+1; controle += incremento; } } } if (controle &gt; 1045){ if(disparoUnico == true) { incremento = 0; } else{ controle = 50; tela(); } } } } //função botão geral void botao(int x,int y,String texto, int cor){ float r=255,g=255,b=255; if (cor==1){r=44; g=105; b=118;} if (cor==11){r=14; g=75; b=88;} if (cor==2){r=60; g=160; b=60;} if (cor==3){r=160; g=60; b=60;} if (cor==4){r=160; g=160; b=60;} stroke(0); fill(r,g,b); rect(x,y,100,30,10); stroke(255); fill(255); textSize(12); textAlign(CENTER); text(texto,x+50,y+20); } //função botão pequeno void botao2(int x,int y,String texto, int cor){ float r=255,g=255,b=255; if (cor==1){r=44; g=105; b=118;} if (cor==11){r=14; g=75; b=88;} if (cor==2){r=60; g=160; b=60;} if (cor==3){r=160; g=60; b=60;} if (cor==4){r=160; g=160; b=60;} stroke(0); fill(r,g,b); rect(x,y,40,30,10); stroke(255); fill(255); textSize(12); textAlign(CENTER); text(texto,x+20,y+20); } //função reticulado da tela void tela(){ fill(44, 105, 118); stroke(255); rect(50, 150, 1000, 400); stroke(64, 125, 138); for(int i = 60; i&lt;1050; i = i+10){ line(i, 150, i, 550); } stroke(94, 155, 168); for(int i = 100; i&lt;1050; i = i+50){ line(i, 150, i, 550); } stroke(64, 125, 138); for(int i = 150; i&lt;550; i = i+10){ line(50, i, 1050, i); } stroke(94, 155, 168); for(int i = 150; i&lt;550; i = i+50){ if(i== 200    i == 350    i==500){ stroke(160,160,60); line(50, i, 1050, i); } } } </pre>	<pre> botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;300 &amp;&amp; mouseY&lt;330) { incremento = 0.1; tela(); controle = 50; botao(1100,260,"2 s/div",1); botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;340 &amp;&amp; mouseY&lt;370) { incremento = 0.5; tela(); controle = 50; botao(1100,260,"2 s/div",1); botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;380 &amp;&amp; mouseY&lt;410) { incremento = 1; tela(); controle = 50; botao(1100,260,"2 s/div",1); botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;420 &amp;&amp; mouseY&lt;450) { incremento = 2.5; tela(); controle = 50; botao(1100,260,"2 s/div",1); botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;460 &amp;&amp; mouseY&lt;490) { incremento = 5; tela(); controle = 50; botao(1100,260,"2 s/div",1); botao(1100,300,"500 ms/div",1); botao(1100,340,"100 ms/div",1); botao(1100,380,"50 ms/div",1); botao(1100,420,"20 ms/div",1); botao(1100,460,"10 ms/div",1); } if (mouseX&gt;1100 &amp;&amp; mouseX&lt;1140 &amp;&amp; mouseY&gt;520 &amp;&amp; mouseY&lt;550) { offset = offset+1; } if (mouseX&gt;1160 &amp;&amp; mouseX&lt;1200 &amp;&amp; mouseY&gt;520 &amp;&amp; mouseY&lt;550) { offset = offset-1; } } } </pre>
--	--	--

## Atividade 1: Demonstrar o uso do experimento

### Habilidades e Competências trabalhadas:

- ✓ Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.

### Objetivos:

- ✓ Conhecer sensores
- ✓ Conhecer a interface *Arduino* e *Processing*
- ✓ Realizar medições de posição, com auxílio de régua ou paquímetro
- ✓ Desenvolver análise de valor médio e erro relativo

### Introdução:

Sensores são dispositivos eletrônicos que convertem uma variação de grandeza física em uma variação de grandeza elétrica: tensão, corrente ou resistência. Os sensores são usados em portas automáticas, em termômetros digitais, em sistemas antifurto, na rotação automática da tela do celular ou tablet e em outras inúmeras aplicações.

No experimento *Oscilador harmônico amortecido com o uso do Arduino e Processing* é utilizado um sensor óptico por reflexão. Este é formado por um LED (Diodo Emissor de Luz) e um fototransistor infravermelho. O LED emite uma luz, no comprimento do infravermelho, que é refletida pela etiqueta colada na régua (reflexão difusa). A partir da intensidade luminosa captada pelo fototransistor, é gerada uma variação de tensão que é percebida e digitalizada pelo *Arduino*. O sinal digitalizado é transmitido ao computador pela conexão USB, sendo processado pelo *Processing* e apresentado em sua interface.

### Procedimento:

Comprovar o funcionamento do sensor manipulando livremente a régua, para cima e para baixo e visualizando o efeito no registro temporal de posição da régua na interface computacional.

Com o auxílio de uma régua ou de um paquímetro, realizar medições de posição da régua e registrar, em uma tabela, os valores obtidos pela interface computacional. A partir dos valores medidos e dos valores registrados na interface, calcular o valor médio para cada medição e o erro relativo parcial e total.



Distância (mm)	Medição 1 (volts)	Medição 1 (mm)	Medição 2 (volts)	Medição 2 (mm)	Medição 3 (volts)	Medição 3 (mm)
30						
35						
40						
45						
50						

Distância (mm)	Valor Médio (volts)	Valor Médio (mm)	Erro Relativo
30			
35			
40			
45			
50			
Erro relativo médio			

Com base no erro relativo, escrever o valor médio de cada uma das leituras com a respectiva incerteza de medição

## Atividade 2: Análise do fenômeno da deformação elástica da lâmina.

### Habilidades e Competências trabalhadas:

- ✓ Rever ou avaliar movimentos em quedas, sistemas planetários ou objetos sob a ação de molas, tendo como pressuposto a conservação da energia mecânica.
- ✓ Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.

### Objetivos:

- ✓ Conhecer a força elástica
- ✓ Determinar a constante elástica em função do deslocamento da extremidade da lâmina
- ✓ Conhecer o gráfico da força em função do deslocamento e interpretar o significado da área do gráfico sob a função

### Introdução:

Uma mola é um dispositivo elástico e flexível que armazena energia em sua forma potencial. Em geral, são confeccionadas em arame de aço temperado e apresentam forma helicoidal, mas, dependendo da finalidade, podem ser confeccionados em borracha, plástico ou outros materiais e podem apresentar uma grande gama de formas, como lâminas, cilindros, etc.

Ao estudar as deformações de molas a partir da aplicação de forças, Robert Hooke (1635-1703) verificou que a deformação da mola aumenta proporcionalmente à força. A partir desta constatação estabeleceu-se a chamada Lei de Hooke:

$$F = -K \cdot x$$

Onde: F: intensidade da força aplicada (N); k: constante elástica (N/m) e x: deformação da mola (m).

O sinal negativo na equação indica que a força exercida pela mola tem sempre o sentido oposto do deslocamento da sua extremidade livre. A constante elástica da mola depende de suas características físicas e geométricas.

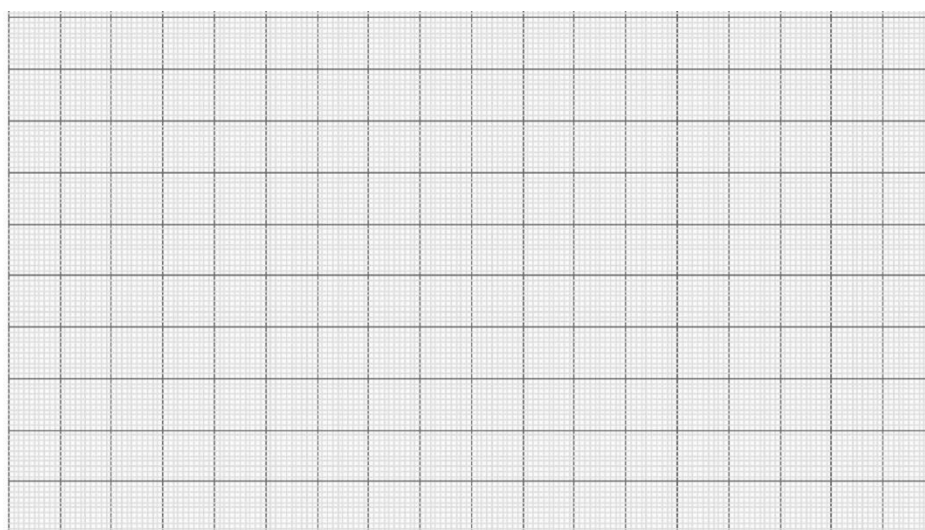
**Procedimento:**

Para determinar a constante elástica da lâmina, é necessário inserir massas na lâmina e medir o deslocamento ocorrido. Para realizar tal procedimento, é possível utilizar moedas de valores correntes como elementos de massa, colocando-as em uma posição definida da régua e realizar as medições de deslocamento com o uso da interface computacional, registrando os resultados em uma tabela. A partir dos valores de massa e deslocamento registrados, construa o gráfico força x deslocamento e determine a constante elástica. Para que você obtenha outros valores da constante elástica, altere a posição da massa na régua ou altere a própria régua.

	Diâmetro: 22 mm Massa: 4,10 g		Diâmetro: 23 mm Massa: 7,81 g
	Diâmetro: 20 mm Massa: 4,80 g		Diâmetro: 27 mm Massa: 7,00
	Diâmetro: 25 mm Massa: 7,55 g	Tolerância da massa: $\pm 2\%$	

Número de Moedas	Massa Total (g)	Medições de posição			Media (mm)	Deslocamento (mm)
		1 (mm)	2 (mm)	3 (mm)		
0						-----
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

Número de Moedas	Massa Total (kg)	Força Peso (N)	Deslocamento (m)	k (N/m)
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				



Determinação da constante elástica pelo gráfico:

1 – Método do coeficiente angular:

2 – Método da área sob a reta:

### Atividade 3: Análise do movimento oscilatório

#### Habilidades e Competências trabalhadas:

- ✓ Conceber e realizar arranjos experimentais em que se possa avaliar ou verificar a antecipação de movimentos, como a oscilação de pêndulos, ou de deformações, como na colisão de objetos elásticos.
- ✓ Determinar experimentalmente relações entre variáveis como o valor da aceleração da gravidade, a altura e o tempo numa queda livre, ou no percurso de uma esfera em canaleta inclinada, ou entre comprimento de pêndulos e sua frequência de oscilação.
- ✓ Compreender as propriedades das ondas e como elas explicam fenômenos presentes em nosso cotidiano.
- ✓ Compreender a importância dos fenômenos ondulatórios na vida moderna sobre vários aspectos, entre eles sua importância para a exploração espacial e na comunicação.

#### Objetivos:

- ✓ Conhecer as características de uma onda
- ✓ Determinar a amplitude, frequência e período de uma onda
- ✓ Conhecer o gráfico de representação de uma onda

#### Introdução:

O MHS (Movimento Harmônico Simples) é um movimento que ocorre de modo periódico ou cíclico. O MHS também pode ser descrito como o movimento de oscilação mais elementar, e pode ser observado em qualquer sistema em equilíbrio estável que subitamente tem essa situação modificada, passando a executar um movimento periódico, cíclico ou oscilatório, sendo o último o termo mais usado para designar esse tipo de situação.

Quando a lâmina é retirada, por ação de uma força, de sua posição de equilíbrio e solta, inicia um movimento de vai-e-vem regido pela força restauradora exercida pelo efeito mola da lâmina. A força aplicada determina a amplitude da oscilação, enquanto o período de oscilação depende de alguns fatores como a geometria da lâmina, do material de construção, do comprimento em balanço, dentre outros.

**Procedimento:**

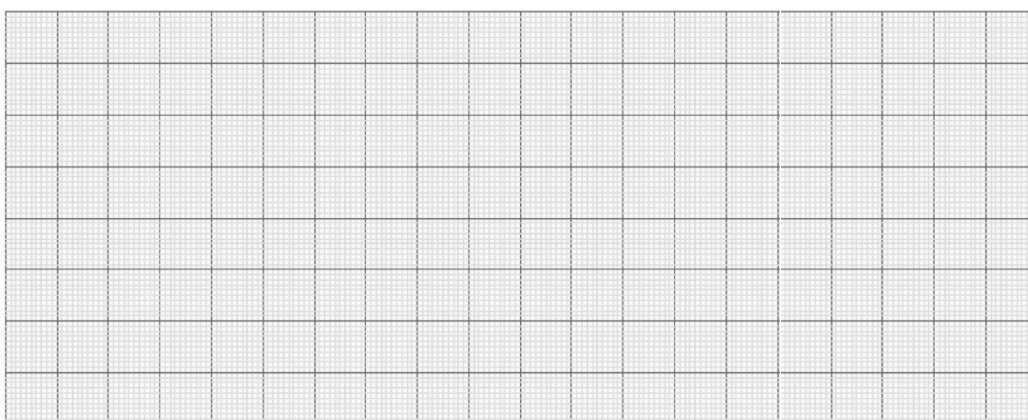
A partir do acionamento da régua, observe o registro do sinal na interface computacional, determinando período, frequência e amplitude de um ciclo, verificando qual a varredura mais adequada para esta medição.

Altere o comprimento da haste em balanço, determinando a relação da variação do comprimento em balanço da régua com a variação do período de oscilação. É possível alterar o material da lâmina, utilizando outras réguas, de mesmo comprimento, e definindo a sua frequência natural de oscilação. Também coloque uma massa solidária a régua e analise a variação da frequência de oscilação.

Balanço da régua	Amplitude (mm)	Período (s)	Frequência (Hz)
18 cm			
19 cm			
20 cm			
21 cm			
22 cm			
23 cm			
24 cm			

Correlacione o comprimento em balanço da régua com a variação do período de oscilação.

Represente no gráfico a onda observada com o comprimento em balanço da régua igual a 20 cm.



## Atividade 4: Análise do amortecimento

### Habilidades e Competências trabalhadas:

- ✓ Rever ou avaliar movimentos em quedas, sistemas planetários ou objetos sob a ação de molas, tendo como pressuposto a conservação da energia mecânica.
- ✓ Conceber e realizar arranjos experimentais em que se possa avaliar ou verificar a antecipação de movimentos, como a oscilação de pêndulos, ou de deformações, como na colisão de objetos elásticos.
- ✓ Determinar experimentalmente relações entre variáveis como o valor da aceleração da gravidade, a altura e o tempo numa queda livre, ou no percurso de uma esfera em canaleta inclinada, ou entre comprimento de pêndulos e sua frequência de oscilação.
- ✓ Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.

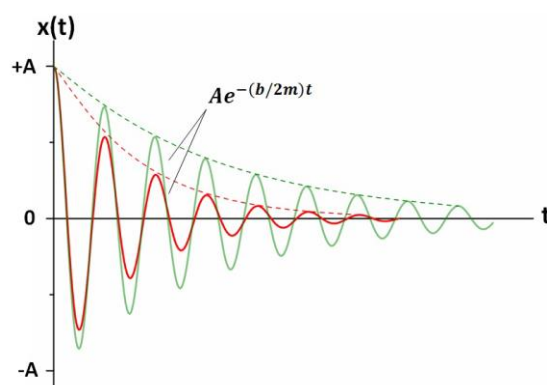
### Objetivos:

- ✓ Reconhecer a taxa de amortecimento de um oscilador harmônico amortecido
- ✓ Identificar as causas do amortecimento
- ✓ Observar o efeito do atrito em movimentos oscilatórios
- ✓ Identificar o tipo de função matemática que expressa o amortecimento

### Introdução:

Na natureza todos os movimentos harmônicos macroscópicos são amortecidos, pois perdem gradativamente sua energia mecânica em virtude do atrito, principalmente com o ar. Observamos esta perda de energia com o decaimento gradual da amplitude do movimento, tendendo ao longo do tempo alcançar a posição de repouso.

A função matemática que representa este decaimento da amplitude é a função exponencial.



**Procedimento:**

A partir do acionamento da régua, meça a amplitude do sinal em cinco ciclos subsequentes. Registre os valores obtidos em uma tabela e analisar, nestes 5 ciclos a taxa de decaimento.

Usando o maior tempo de varredura da interface, avalie o decaimento total da oscilação. Com base na envoltória do sinal correlacionar, a qual função matemática pertence o decaimento da amplitude, enumere as possíveis causas deste decaimento.

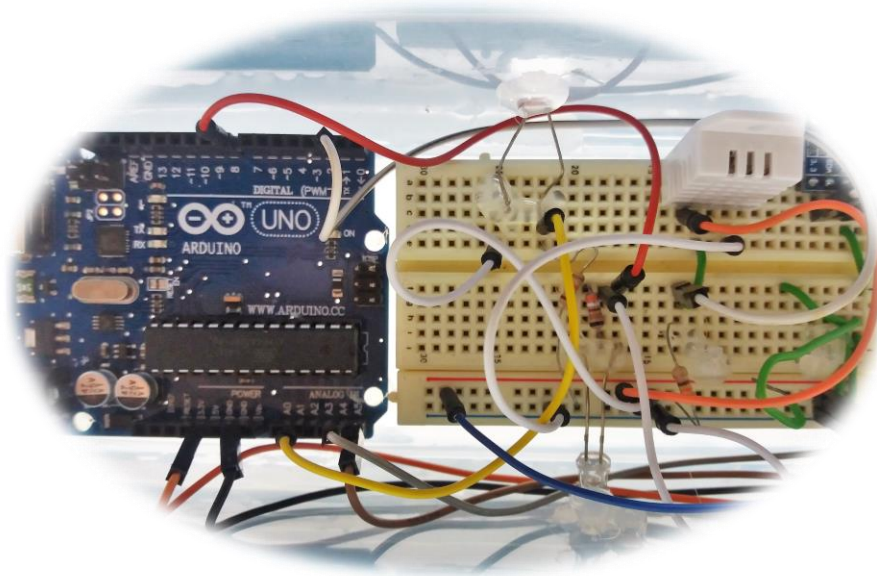
Também é possível alterar o material ou o comprimento em balanço da régua para visualizar outras taxas de decaimento. A inserção de massas solidarias à régua pode ser realizada para alterar a taxa de decaimento da amplitude

Ciclo	Amplitude relativa positiva (mm)	Varição em relação ao ciclo anterior (mm)
1		-----
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
Variação total		



# PRODUTO EDUCACIONAL

## Estação Meteorológica com o uso do Arduino e do Processing



Construção do experimento e roteiros de atividades

Luis Henrique Monteiro de Castro

Setembro de 2016

**MNPEF**  
Mestrado Nacional  
Profissional em  
Ensino de Física



## Sumário

<b>Algumas Palavras</b> .....	1
<b>Materiais necessários</b> .....	1
<b>Montando o experimento</b> .....	2
<b>Código do Arduino</b> .....	3
<b>Código do Processing</b> .....	4
<b>Atividade 1: Coletando as variáveis meteorológicas</b> .....	7
<b>Atividade 2: Analisando dados meteorológicos.</b> .....	10

## Algumas Palavras

Este produto educacional é uma sugestão de uso do Arduino e do Processing como experimento para o Ensino de Física. Neste material professores e alunos encontrarão informações quanto aos materiais necessários para a construção mecânica e eletrônica do experimento, bem como os códigos de programação do Arduino e do Processing e ainda sugestões de atividades envolvendo os conceitos de temperatura, pressão atmosférica, umidade do ar e luminosidade. O material foi desenvolvido considerando que o leitor tenha algum conhecimento, mesmo que rudimentar, de programação e de eletrônica. Todo o material e os códigos também estão disponibilizados no endereço [www.aulanaweb.com](http://www.aulanaweb.com).

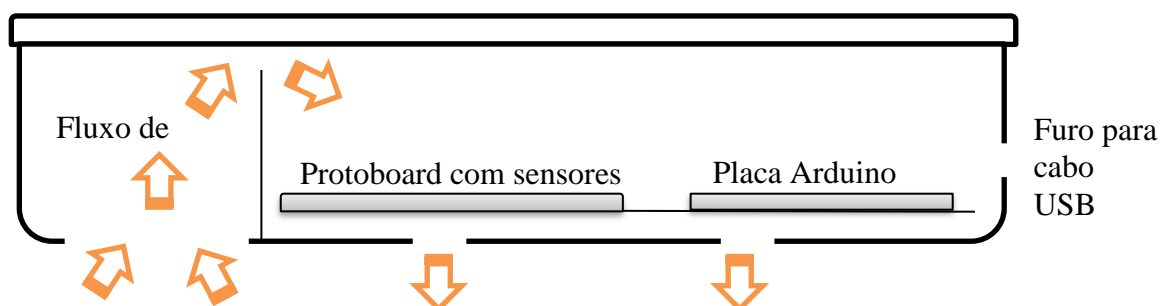
## Materiais necessários

Item	Descrição	Quantidade
1	Caixa plástica para acomodar a montagem	01
2	Placa Arduino UNO	01
3	<i>Protoboard</i> 830 pontos – Matriz de contatos elétricos para montagem de circuitos	01
4	Resistores 10 k $\Omega$ x 1/8 w	02
5	Resistores 220 $\Omega$ x 1/8 w	01
6	LEDs de alto brilho branco	01
7	LDR 5mm	01
8	Sensor AOSONG AM2302	01
9	BMP180	01
10	Jumpers para ligações	25

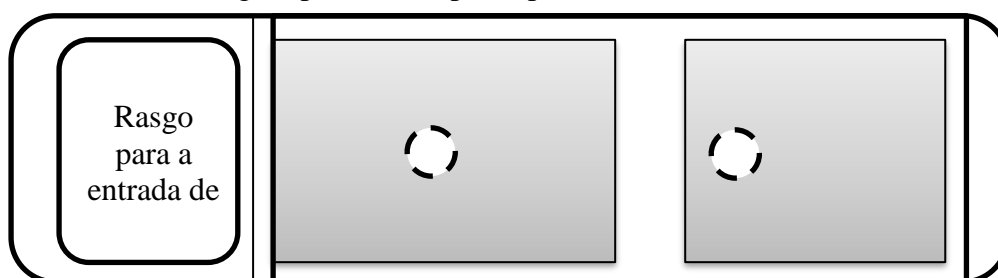
## Montando o experimento



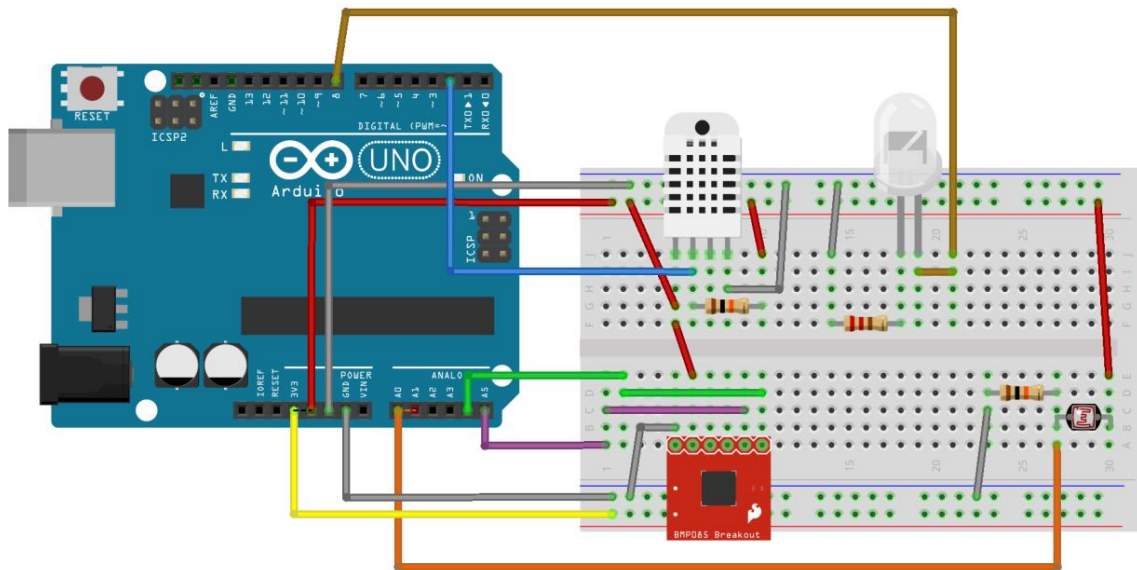
Para o funcionamento adequado dos sensores, é necessário realizar algumas modificações no pote plástico a fim de permitir a circulação de ar não permitindo a entrada de água. Apresentamos, como sugestão, o desenho esquemático das alterações realizadas no pote plástico para o devido acondicionamento da estação meteorológica.



Embalagem plástica adaptada para o Arduino e sensores



Para que haja uma boa circulação de ar dentro da embalagem plástica foram realizados diversos furos na base da embalagem e nas áreas que não comprometessem o isolamento com relação a água da chuva. Nas fotos a seguir mostramos a montagem mecânica



## Código do Arduino

```

#include <Wire.h>
#include <DHT.h>
#include <Adafruit_085.h>
#define DHTPIN 2
#define DHTTYPE DHT22
#define oversampling 3
DHT dht(DHTPIN, DHTTYPE);
Adafruit_BMP085 bmp;
int sensorLDR;
int contador = 1;
float humidity,h, t;
String data;
String controleString;
void setup() {
  bmp.begin();
  dht.begin();
  pinMode(8, OUTPUT);
  Serial.begin(9600);
  while (!Serial) ; // wait for serial
  delay(200);
}
void loop(){
  if (Serial.available() > 0) {
    controleString = Serial.read();
    if (controleString == "49"){
      h = dht.readHumidity();
      t = dht.readTemperature();
      sensorLDR = analogRead(1);
      data = "";
      data += contador;
      data += ",";
      data += String(bmp.readPressure());
      data += ",";
      data += String(bmp.readAltitude());
      data += ",";
      data += String(bmp.readSealevelPressure());
      data += ",";
      data += String(bmp.readAltitude(101500));
      data += ",";
      data += String(bmp.readTemperature());
      data += ",";
      data += String(t, 2);
      data += ",";
      data += String(h, 2);
      data += ",";
      data += String(sensorLDR);
      data += ",";
      data += String("99999");
      Serial.println(data);
      contador +=1;
    }
    if (controleString == "50"){
      digitalWrite(8, HIGH);
    }
    if (controleString == "51"){
      digitalWrite(8, LOW);
    }
    if (controleString == "52"){
      contador = 1;
    }
  }
}

```

## Código do Processing

<pre> import processing.serial.*; // Variáveis PrintWriter arquivoGeral; PrintWriter arquivoMediasHorarias; PrintWriter arquivoGeralCorrigido1min; PrintWriter arquivoGeralCorrigido10min; PrintWriter arquivoGeralCorrigido30min; String serialString = null; Serial portaSerial; int lf = 10; int contador = 0; int referencia = 1; String nomeArquivo = ""; String[] dados; float[] valorPressao = new float[326]; float[] valorTemperatura = new float[26]; float[] valorUmidade = new float[26]; float[] valorLuminosidade = new float[26]; String dadosCorrigidos,dadosMedia; String dia,mes,ano,hora,minuto,segundo; int controle=0; int controleMedia = 0, controleMediaGeral = 0; float pressao = 0, minipressao = 0, maxpressao = 0, mediapressao = 0, refpressao = 0; float pressaoC = 0, minipressaoC = 0, maxpressaoC = 0, mediapressaoC = 0, refpressaoC = 0; float temp1 = 0, mintemp1 = 0, maxtemp1 = 0, mediatemp1 = 0, reftemp1 = 0; float temp2 = 0, mintemp2 = 0, maxtemp2 = 0, mediatemp2 = 0, reftemp2 = 0; float tempMedia = 0, mintempMedia = 0, maxtempMedia = 0, mediatempMedia = 0, reftempMedia=0; float umidade = 0, minumidade = 0, maxumidade = 0, mediaumidade = 0, refumidade = 0; float luminosidade = 0, minluminosidade = 0, maxluminosidade = 0, medialuminosidade = 0, refluminosidade = 0; int referenciaMedia = 0; boolean controleGeracaoMedia = true; boolean antirepete = false; boolean antirepete2 = false; boolean antirepete3 = false; float rascunho; float rascunhoAnterior; void setup() { nomeArquivo(); arquivoGeral = createWriter(nomeArquivo+".txt"); arquivoGeralCorrigido1min = createWriter(nomeArquivo+"- Corrigido1min.txt"); arquivoGeralCorrigido10min = createWriter(nomeArquivo+"- Corrigido10min.txt"); arquivoGeralCorrigido30min = createWriter(nomeArquivo+"- Corrigido30min.txt"); arquivoMediasHorarias = createWriter(nomeArquivo+"- MediasHorarias.txt"); printArray(Serial.list()); </pre>	<pre> dadosCorrigidos(); arquivoGeralCorrigido10min.println(dado sCorrigidos); arquivoGeralCorrigido10min.flush(); } if (controle &gt; 0 &amp;&amp; second() == 0 &amp;&amp; minute() == 0    minute() == 30){ dadosCorrigidos(); arquivoGeralCorrigido30min.println(dado sCorrigidos); arquivoGeralCorrigido30min.flush(); } if (controle &gt; 0 &amp;&amp; second() == 0){ dadosCorrigidos(); arquivoGeralCorrigido1min.println(dados Corrigidos); arquivoGeralCorrigido1min.flush(); tela(80, 450, 250, 200); tela(388, 450, 250, 200); tela(695, 450, 250, 200); tela(1000, 450, 250, 200); for (int i=0;i&lt;25;i=i+1){ valorPressao[i] = valorPressao[i+1]; valorTemperatura[i] = valorTemperatura[i+1]; valorUmidade[i] = valorUmidade[i+1]; valorLuminosidade[i] = valorLuminosidade[i+1]; } valorPressao[25] = float(dados[1]); valorTemperatura[25] = float(dados[5]); valorUmidade[25] = float(dados[7]); valorLuminosidade[25] = float(dados[8]); for (int i=0;i&lt;25;i=i+1){ if (valorPressao[i]&lt;99000){valorPressao[i] = 99000;} if (valorPressao[i+1]&lt;99000){valorPressao[ i+1] = 99000;} rascunhoAnterior = map(valorPressao[i], 99000, 103000, 650,450); rascunho = map(valorPressao[i+1], 99000, 103000, 650,450); fill(255,0,0); stroke(255,0,0); line(80+(i*10),rascunhoAnterior,80+((i+1 )*10),rascunho); fill(255); stroke(255); } for (int i=0;i&lt;25;i=i+1){ rascunhoAnterior = map(valorTemperatura[i], 0, 50, 650,450); rascunho = map(valorTemperatura[i+1], 0, 50, 650,450); fill(255,0,0); stroke(255,0,0); line(388+(i*10),rascunhoAnterior,388+((i +1)*10),rascunho); fill(255); stroke(255); } for (int i=0;i&lt;25;i=i+1){ rascunhoAnterior = map(valorUmidade[i], 0, 100, 650,450); rascunho = map(valorUmidade[i+1], 0, 100, 650,450); fill(255,0,0); </pre>	<pre> if(maxumidade &lt;= refumidade){apaga(1061,331);maxumida de = refumidade;text(maxumidade+" às "+hora+"."+minuto+"."+segundo, 1155, 350);} if(maxluminosidade &lt;= refluminosidade){apaga(1061,361);maxlu minosidade = refluminosidade;text(maxluminosidade+" às "+hora+"."+minuto+"."+segundo, 1155, 380);} tabela2(); text(dia+"/"+mes+"/"+String.valueOf(yea r()), 965, 170); text(dia+"/"+mes+"/"+String.valueOf(yea r()), 1155, 170); if(minute() == 0 &amp;&amp; second() == 0 &amp;&amp; controleGeracaoMedia == true){ controleGeracaoMedia = false; referenciaMedia += 1; dadosMedia = referenciaMedia+""; if(day(&lt;10){dia="0"+String.valueOf(day ());}else{dia=String.valueOf(day());} if(month(&lt;10){mes="0"+String.valueOf(mo nth());}else{mes=String.valueOf(mont h());} dadosMedia += dia+"/"+mes+"/"+String.valueOf(year()+ ""; if(hour(&lt;10){hora="0"+String.valueOf(h our());}else{hora=String.valueOf(hour()); } if(minute(&lt;10){ minuto="0"+String.valu eOf(minute());}else{minuto=String.valu eOf(minute());} if(second(&lt;10){segundo="0"+String.val ueOf(second());}else{segundo=String.val ueOf(second());} dadosMedia += hora+"."+minuto+"."+segundo+""; dadosMedia += pressao/controlMedia+""; dadosMedia += pressaoC/controlMedia+""; dadosMedia += temp1/controlMedia+""; dadosMedia += temp2/controlMedia+""; dadosMedia += umidade/controlMedia+""; dadosMedia += luminosidade/controlMedia; arquivoMediasHorarias.println(dadosMed ia); arquivoMediasHorarias.flush(); fill(4, 79, 111); stroke(4, 79, 111); rect(270,400,80,180); rect(130,400,120,20); textSize(15); fill(255); //textAlign(RIGHT); //text(dia+"/"+mes+"/"+String.valueOf(ye ar()+hora+"."+minuto+"."+segundo, 345, 420); //text(pressaoC/controlMedia, 345, 480); //text(temp2/controlMedia, 345, 510); //text(umidade/controlMedia, 345, 540); //text(luminosidade/controlMedia, 345, 570); controleMedia = 0; pressao = 0; </pre>
---	---	--

<pre> portaSerial = new Serial(this, Serial.list()[1], 9600); portaSerial.clear(); serialString = portaSerial.readStringUntil(If); serialString = null; size(1300, 680); frameRate(120); noSmooth(); background(4, 79, 111); stroke(84, 145, 158); fill(255); textSize(12); text("Universidade Federal do Estado do Rio de Janeiro - UNIRIO", 50, 40); text("Mestrado Nacional Profissional de Ensino de Física - MNPEF", 50, 55); text("Orientador: Felipe Mondaini", 50, 70); text("Mestrando: Luis Henrique M de Castro", 50, 85); textSize(20); fill(102); stroke(0); rect(500, 55, 650, 35); fill(255); text("Interface Computacional - Experimento Estação Meteorológica", 520, 80); fill(255); textSize(12); textAlign(LEFT); text("Grandezas medidas", 50, 140); text("Referência", 50, 170); text("Pressão Atmosférica (hPa)", 50, 200); text("Pressão Atmosférica ref.nível do mar (hPa)", 50, 230); text("Temperatura BMP180 (°C)", 50, 260); text("Temperatura DHT22 (°C)", 50, 290); text("Temperatura Média (°C)", 50, 320); text("Umidade (%)", 50, 350); text("Luminosidade (UA)", 50, 380); fill(4, 79, 111); stroke(255); fill(255); tabela(); textAlign(CENTER); text("Valores Instantâneos", 395, 140); text("Valores médios", 585, 140); text("Valores médio por hora", 775, 140); text("Valores mínimos", 965, 140); text("Valores máximos", 1155, 140); text("Pressão Atmosférica (hPa)", 225, 440); text("Temperatura (°C)", 525, 440); text("Umidade (%)", 825, 440); text("Luminosidade (UA)", 1125, 440); tela(80, 450, 250, 200); tela(388, 450, 250, 200); tela(695, 450, 250, 200); tela(1000, 450, 250, 200); textSize(10); textAlign(LEFT); text("1030", 50,455); text("1020", 50,505); text("1010", 50,555); text("1000", 50,605); text("990", 50,655); text("50", 360,455); text("37.5", 360,505); text("25", 360,555); text("12.5", 360,605); </pre>	<pre> stroke(255,0,0); line(695+(i*10),rascunhoAnterior,695+((i +1)*10),rascunho); fill(255); stroke(255); } for (int i=0;i&lt;25;i=i+1){ rascunhoAnterior = map(valorLuminosidade[i], 0, 1024, 650,450); rascunho = map(valorLuminosidade[i+1], 0, 1024, 650,450); fill(255,0,0); stroke(255,0,0); line(1000+(i*10),rascunhoAnterior,1000+ ((i+1)*10),rascunho); fill(255); stroke(255); } tabela(); textSize(12); fill(255); textAlign(CENTER); text(dia+"/"+mes+"/"+String.valueOf(yea r)+"-"+hora+":."+minuto+":."+segundo, 395, 170); text(dados[1], 395, 200); text(dados[3], 395, 230); text(dados[5], 395, 260); text(dados[6], 395, 290); text((((float(dados[5]))+(float(dados[6])) )/2), 395, 320); text(dados[7], 395, 350); text(dados[8], 395, 380); textAlign(LEFT); controleMediaGeral += 1; controleMedia += 1; pressao += float(dados[1]); pressaoC += float(dados[3]); temp1 += float(dados[5]); temp2 += float(dados[6]); tempMedia = (temp1+temp2)/2; umidade += float(dados[7]); luminosidade += float(dados[8]); refpressao = float(dados[1]); refpressaoC = float(dados[3]); reftemp1 = float(dados[5]); reftemp2 = float(dados[6]); reftempMedia = (reftemp1+reftemp2)/2; refumidade = float(dados[7]); refluminosidade = float(dados[8]); if(minpressao == 0){minpressao = refpressao; maxpressao = refpressao;} if(minpressaoC == 0){minpressaoC = refpressaoC;maxpressaoC = refpressaoC;} if(mintemp1 == 0){mintemp1 = reftemp1;maxtemp1 = reftemp1;} if(mintemp2 == 0){mintemp2 = reftemp2;maxtemp2 = reftemp2;} if(mintempMedia == 0){mintempMedia = reftempMedia;maxtempMedia = reftempMedia;} if(minumidade == 0){minumidade = refumidade;maxumidade = refumidade;} if(minluminosidade == 0){minluminosidade = refluminosidade;maxluminosidade = refluminosidade;} textAlign(CENTER); if(minpressao &gt;= refpressao){apaga(871,181);minpressao = refpressao;text(minpressao+" às "+hora+":."+minuto+":."+segundo, 965, 200);} </pre>	<pre> pressaoC = 0; temp1 = 0; temp2 = 0; umidade = 0; luminosidade = 0; } } } } } void nomeArquivo(){ nomeArquivo = "DadosETEFV-"; nomeArquivo += String.valueOf(day()); nomeArquivo += "-"; nomeArquivo += String.valueOf(month()); nomeArquivo += "-"; nomeArquivo += String.valueOf(year()); nomeArquivo += "-"; nomeArquivo += String.valueOf(hour()); nomeArquivo += "-"; nomeArquivo += String.valueOf(minute()); nomeArquivo += "-"; nomeArquivo += String.valueOf(second()); } void dadosCorrigidos(){ dados = split(serialString,','); dadosCorrigidos = controle+";"; if(day()&lt;10){dia="0"+String.valueOf(day ());}else{dia=String.valueOf(day());} if(month()&lt;10){mes="0"+String.valueOf( month());}else{mes=String.valueOf(mont h());} dadosCorrigidos += dia+"/"+mes+"/"+String.valueOf(year()+ ";"; if(hour()&lt;10){hora="0"+String.valueOf(h our());}else{hora=String.valueOf(hour()); } if(minute()&lt;10){minuto="0"+String.valu eOf(minute());}else{minuto=String.valu eOf(minute());} if(second()&lt;10){segundo="0"+String.val ueOf(second());}else{segundo=String.val ueOf(second());} dadosCorrigidos += hora+":."+minuto+":."+segundo+";"; dadosCorrigidos += dados[0]+";"; dadosCorrigidos += dados[1]+";"; dadosCorrigidos += dados[2]+";"; dadosCorrigidos += dados[3]+";"; dadosCorrigidos += dados[4]+";"; dadosCorrigidos += dados[5]+";"; dadosCorrigidos += dados[6]+";"; dadosCorrigidos += dados[7]+";"; dadosCorrigidos += dados[8]; } void tela(int x, int y, int w, int h){ fill(44, 105, 118); stroke(255); rect(x, y, w, h); stroke(64, 125, 138); for(int i = x+1; i&lt;x+w; i = i+10){ line(i, y, i, y+h); } stroke(94, 155, 168); for(int i = x+1; i&lt;x+w; i = i+50){ line(i, y, i, y+h); } stroke(64, 125, 138); for(int i = y+1; i&lt;y+h; i = i+10){ line(x, i, x+w, i); } </pre>
---	---	---

<pre> text("0", 360,655); text("100", 670,455); text("75", 670,505); text("50", 670,555); text("25", 670,605); text("0", 670,655); text("100", 975,455); text("75", 975,505); text("50", 975,555); text("25", 975,605); text("0", 975,655); } void draw() { if (hour() == 0 &amp;&amp; minute() == 0 &amp;&amp; second() == 0 &amp;&amp; antirepete2 == false){ antirepete2 = true; arquivoGeral.close(); arquivoGeralCorrigido1min.close(); arquivoGeralCorrigido10min.close(); arquivoGeralCorrigido30min.close(); arquivoMediasHorarias.close(); nomeArquivo(); arquivoGeral = createWriter(nomeArquivo+".txt"); arquivoGeralCorrigido1min = createWriter(nomeArquivo+"- Corrigido1min.txt"); arquivoGeralCorrigido10min = createWriter(nomeArquivo+"- Corrigido10min.txt"); arquivoGeralCorrigido30min = createWriter(nomeArquivo+"- Corrigido30min.txt"); arquivoMediasHorarias = createWriter(nomeArquivo+"- MediasHorarias.txt"); controle = 1; portaSerial.write("4"); } if(second()%2 == 0 &amp;&amp; second() !=0){ portaSerial.write("2"); } if(second()%2 == 1 &amp;&amp; second() !=59){ portaSerial.write("3"); antirepete = false; } if(second() == 59){ antirepete = false; antirepete2 = false; controleGeracaoMedia = true; } if(second() == 0 &amp;&amp; antirepete == false) { portaSerial.write("1"); antirepete = true; delay(100); while (portaSerial.available() &gt; 0) { serialString = portaSerial.readStringUntil(lf); if (serialString != null) { println(serialString); arquivoGeral.print(serialString); arquivoGeral.flush(); controle += 1; if (controle &gt; 0 &amp;&amp; second() == 0 &amp;&amp; minute() == 0    minute() == 10   minute() == 20    minute() == 30   minute() == 40    minute() == 50){ </pre>	<pre> if(minpressaoC &gt;= refpressaoC){ apaga(871,211);minpressao C = refpressaoC;text(minpressaoC+" às "+hora+":"+minuto+":"+segundo, 965, 230);} if(mintemp1 &gt;= reftemp1){ apaga(871,241);mintemp1 = reftemp1;text(mintemp1+" às "+hora+":"+minuto+":"+segundo, 965, 260);} if(mintemp2 &gt;= reftemp2){ apaga(871,271);mintemp2 = reftemp2;text(mintemp2+" às "+hora+":"+minuto+":"+segundo, 965, 290);} if(mintempMedia &gt;= reftempMedia){ apaga(871,301);mintemp Media = reftempMedia;text(mintempMedia+" às "+hora+":"+minuto+":"+segundo, 965, 320);} if(minumidade &gt;= refumidade){ apaga(871,331);minumidade = refumidade;text(minumidade+" às "+hora+":"+minuto+":"+segundo, 965, 350);} if(minluminosidade &gt;= refluminosidade){ apaga(871,361);minlu minosidade = refluminosidade;text(minluminosidade+" às "+hora+":"+minuto+":"+segundo, 965, 380);} if(maxpressao &lt;= refpressao){ apaga(1061,181);maxpressao = refpressao;text(maxpressao+" às "+hora+":"+minuto+":"+segundo, 1155, 200);} if(maxpressaoC &lt;= refpressaoC){ apaga(1061,211);textAlign( CENTER);text(maxpressaoC+" às "+hora+":"+minuto+":"+segundo, 1155, 230);} if(maxtemp1 &lt;= reftemp1){ apaga(1061,241);maxtemp1 = reftemp1;text(maxtemp1+" às "+hora+":"+minuto+":"+segundo, 1155, 260);} if(maxtemp2 &lt;= reftemp2){ apaga(1061,271);maxtemp2 = reftemp2;text(maxtemp2+" às "+hora+":"+minuto+":"+segundo, 1155, 290);} if(maxtempMedia &lt;= reftempMedia){ apaga(1061,301);maxtem pMedia = reftempMedia;text(maxtempMedia+" às "+hora+":"+minuto+":"+segundo, 1155, 320);} </pre>	<pre> } stroke(94, 155, 168); for(int i = y+1; i&lt;y+h; i = i+50){ line(x, i, x+w, i); } stroke(255); fill(255); } void tabela(){ fill(4, 79, 111); stroke(4, 79, 111); rect(300,150,380,240); fill(255); stroke(255); line(50,150,1250,150); line(50,180,1250,180); line(50,210,1250,210); line(50,240,1250,240); line(50,270,1250,270); line(50,300,1250,300); line(50,330,1250,330); line(50,360,1250,360); line(50,390,1250,390); line(300,120,300,390); line(490,120,490,390); line(680,120,680,390); line(870,120,870,390); line(1060,120,1060,390); } void tabela2(){ fill(255); stroke(255); line(50,150,1250,150); line(50,180,1250,180); line(50,210,1250,210); line(50,240,1250,240); line(50,270,1250,270); line(50,300,1250,300); line(50,330,1250,330); line(50,360,1250,360); line(50,390,1250,390); line(300,120,300,390); line(490,120,490,390); line(680,120,680,390); line(870,120,870,390); line(1060,120,1060,390); } void apaga(int x, int y){ fill(4, 79, 111); stroke(4, 79, 111); rect(x,y,178,28); fill(255); stroke(255); } </pre>
---	--	--

## Atividade 1: Coletando as variáveis meteorológicas

### Habilidades e Competências trabalhadas:

- ✓ Reconhecer e saber utilizar corretamente símbolos, códigos e nomenclaturas de grandezas da Física, nas previsões climáticas, identificar temperaturas, pressão, umidade relativa do ar e luminosidade.
- ✓ Compreender fenômenos naturais ou sistemas tecnológicos, identificando e relacionando as grandezas envolvidas.

### Objetivos:

- ✓ Realizar registros adequados em tabelas e planilhas das medições realizadas
- ✓ Identificar taxas e tendências de variação

### Introdução:

Os aspectos mais tradicionais e conhecidos da meteorologia são a previsão do tempo e a climatologia, o estudo do clima. Mas antes de esgotarmos estes aspectos precisamos definir os termos tempo e clima. O tempo é o comportamento momentâneo da atmosfera (breves períodos, como um dia, uma semana, etc.). O clima, por sua vez, é o comportamento da atmosfera em um período mais longo de tempo ou até de forma permanente. Para conhecer o clima de uma determinada região é preciso que os estudos ocorram por um longo período, algumas dezenas de anos, para determiná-lo. As condições do tempo são descritas através de alguns elementos básicos. Dentre elas podemos listar:

- a) Temperatura do ar;
- b) Umidade do ar;
- c) Pressão do ar;
- d) Irradiância solar;
- e) Intensidade e direção dos ventos; e
- f) Pluviosidade.

A proposta da estação meteorológica com o uso do *Arduino* e do *Project* visa, inicialmente, o monitoramento de 4 parâmetros meteorológicos: temperatura do ar, pressão atmosférica, umidade relativa, e luminosidade.

- ✓ A temperatura é a medida da agitação das partículas que compõem um certo material. Se considerarmos as moléculas de um gás, quanto maior a sua temperatura mais energia cinética terão essas moléculas. Quando queremos medir a temperatura de um corpo, precisamos usar uma escala termométrica, isto é, uma maneira de correlacionar um



conjunto de números às temperaturas. As três escalas termométricas mais comuns são Celsius (C), Fahrenheit (F) e Kelvin (K).

- ✓ A pressão é a razão entre a força aplicada em uma superfície e a área de aplicação dessa força. Pressão atmosférica é a pressão que a atmosfera exerce sobre a superfície da Terra. A unidade padrão de pressão no Sistema Internacional (SI) é o Pascal (Pa) (Newton/m<sup>2</sup>). Meteorologistas tem usado tradicionalmente a unidade milibar (1 mb = 100 Pa), mas a unidade Pa é cada vez mais adotada. Usa-se ainda a unidade milímetros de mercúrio (mmHg). A pressão média do ar ao nível do mar é 101,325 KPa ou 760 mmHg.
- ✓ A umidade relativa do ar é definida como a razão entre o percentual (em número de moléculas) de água no ar pelo percentual que corresponde à saturação naquela temperatura do ambiente. Por exemplo, a 30 graus C pode existir no máximo 4% de moléculas de água no estado de vapor no ar; caso efetivamente exista apenas 3%, então a umidade relativa do ar será 3 dividido por 4, portanto, 0,75 ou 75%.
- ✓ A medição da luminosidade permite determinar a iluminação solar, ou seja, o período de tempo que estamos sujeitos a irradiação solar. Com esta medição também definimos o horário do nascer do sol e do pôr do sol, bem como monitorar, de maneira indireta e comparativa a nebulosidade do dia.

**Procedimento:**

Preencha a tabela abaixo com os dados coletados na estação meteorológica no início da aula

Data	Hora	Pressão Atmosférica	Temperatura	Umidade	Luminosidade

Repita o procedimento descrito acima no horário do intervalo da aula:

Data	Hora	Pressão Atmosférica	Temperatura	Umidade	Luminosidade

Ao término da aula, repita mais uma vez o procedimento de coleta de dados:

Data	Hora	Pressão Atmosférica	Temperatura	Umidade	Luminosidade

Com base nos dados coletados nas tabelas anteriores, calcule os valores médios de cada um dos parâmetros:

Pressão Atmosférica	Temperatura	Umidade	Luminosidade

Considerando os dados coletados no início, no intervalo e no final da aula, podemos prever a tendência para cada um dos parâmetros. Complete a tabela abaixo com esta previsão de tendência que pode ser: em alta, estável, em baixa:

Pressão Atmosférica	Temperatura	Umidade	Luminosidade

Considerando a sua percepção quanto as grandezas temperatura e luminosidade, avaliar se a tendência para estas grandezas estão corretas? Por que?

## Atividade 2: Analisando dados meteorológicos.

### Habilidades e Competências trabalhadas:

- ✓ Reconhecer e saber utilizar corretamente símbolos, códigos e nomenclaturas de grandezas da Física, nas previsões climáticas, identificar temperaturas, pressão, umidade relativa do ar e luminosidade.
- ✓ Compreender fenômenos naturais ou sistemas tecnológicos, identificando e relacionando as grandezas envolvidas.
- ✓ Identificar relações entre grandezas e unidades de medida.
- ✓ Interpretar gráfico que represente relações entre grandezas.

### Objetivos:

- ✓ Realizar registros adequados em tabelas e planilhas das medições realizadas
- ✓ Determinar pontos de mínimo e máximo
- ✓ Calcular médias e amplitudes de variação

### Procedimento:

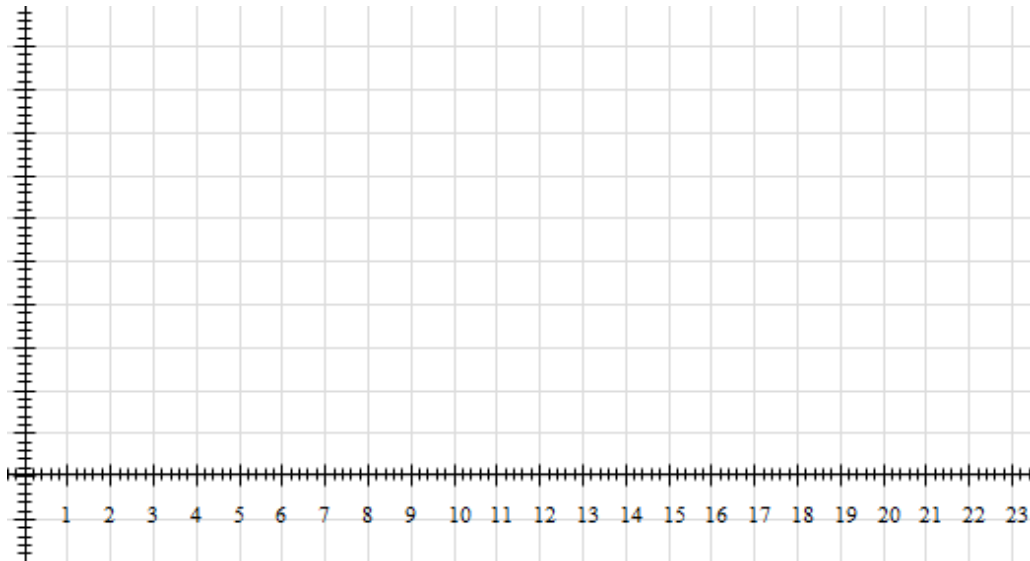
Com base nos dados obtidos em um arquivo diário, de medição horária, do tipo texto, da estação meteorológica preencha uma tabela para cada um dos parâmetros medidos: Pressão Atmosférica, Temperatura, Umidade ou Luminosidade

Valores obtidos por hora:

00:00		06:00		12:00		18:00	
01:00		07:00		13:00		19:00	
02:00		08:00		14:00		20:00	
03:00		09:00		15:00		21:00	
04:00		10:00		16:00		22:00	
05:00		11:00		17:00		23:00	

Após o preenchimento das tabelas, desenhar, nos gráficos, as curvas horárias referentes a cada um dos parâmetros medidos e determinar os valores referentes aos valores mínimo e máximo, com seus respectivos horários de ocorrência e calcular o valor médio e a amplitude de variação.

Gráfico:



Valor máximo: \_\_\_\_\_ Horário: \_\_\_\_\_

Valor mínimo: \_\_\_\_\_ Horário: \_\_\_\_\_

Variação (amplitude): \_\_\_\_\_

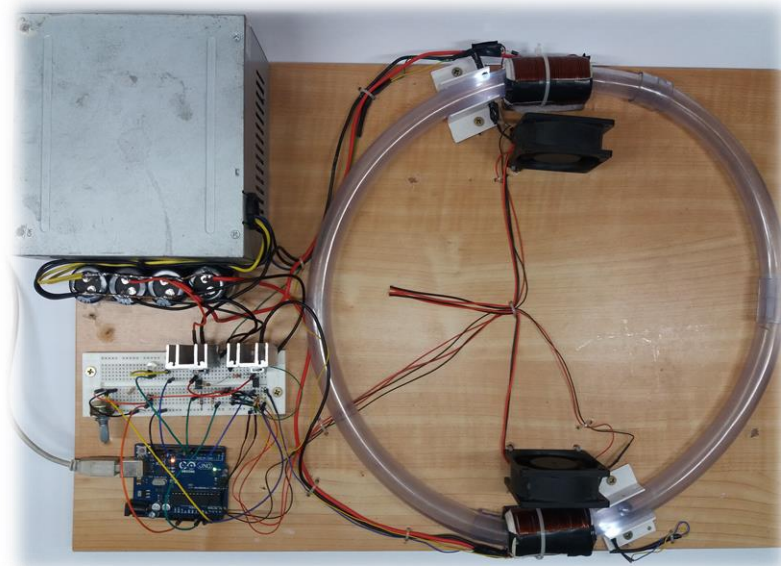
Valor médio: \_\_\_\_\_

Com base nos gráficos, relacionar a variação de temperatura e a variação de umidade com relação a variação de luminosidade

Considerando o gráfico da pressão atmosférica, explicar os ciclos de variação e verificar se a oscilação atmosférica é análoga a oscilação das marés. Caso não se comprove esta relação, como explicamos estas oscilações?

# PRODUTO EDUCACIONAL

## Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing



Construção do experimento e roteiros de atividades

Luis Henrique Monteiro de Castro

Setembro de 2016

**MNPEF**  
Mestrado Nacional  
Profissional em  
Ensino de Física



## Sumário

Algumas Palavras .....	1
Materiais necessários .....	1
Montando o experimento .....	2
Código do Arduino .....	3
Código do Processing .....	3
Atividade 1: Demonstrar o uso do experimento .....	6
Atividade 2: Análise do fenômeno da deformação elástica da lâmina .....	9
Atividade 3: Análise do movimento oscilatório .....	12
Atividade 4: Análise do amortecimento .....	15

## Algumas Palavras

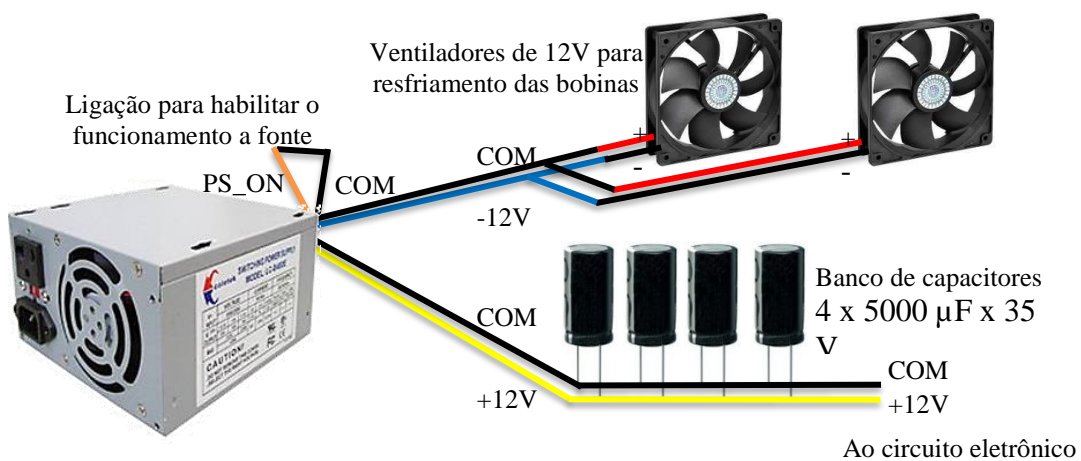
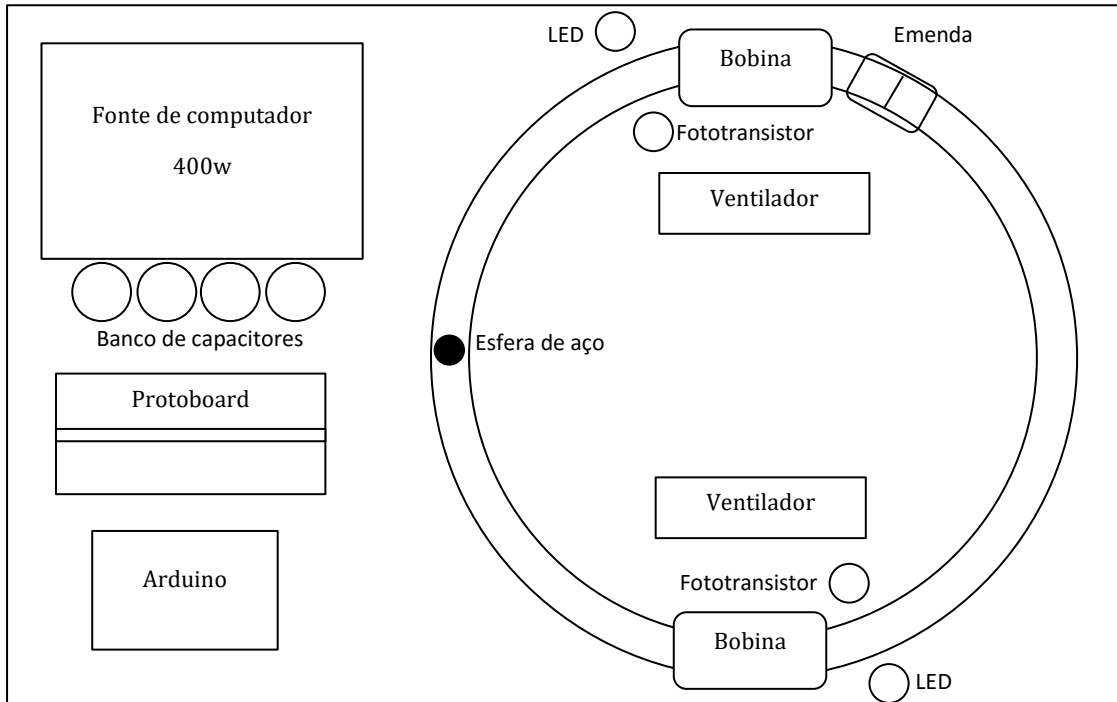
Este produto educacional é uma sugestão de uso do Arduino e do Processing como experimento para o Ensino de Física. Neste material professores e alunos encontrarão informações quanto aos materiais necessários para a construção mecânica e eletrônica do experimento, bem como os códigos de programação do Arduino e do Processing e ainda sugestões de atividades envolvendo os conceitos de movimento circular, energia cinética e física moderna. O material foi desenvolvido considerando que o leitor tenha algum conhecimento, mesmo que rudimentar, de programação e de eletrônica. Todo o material e os códigos também estão disponibilizados no endereço [www.aulanaweb.com](http://www.aulanaweb.com).

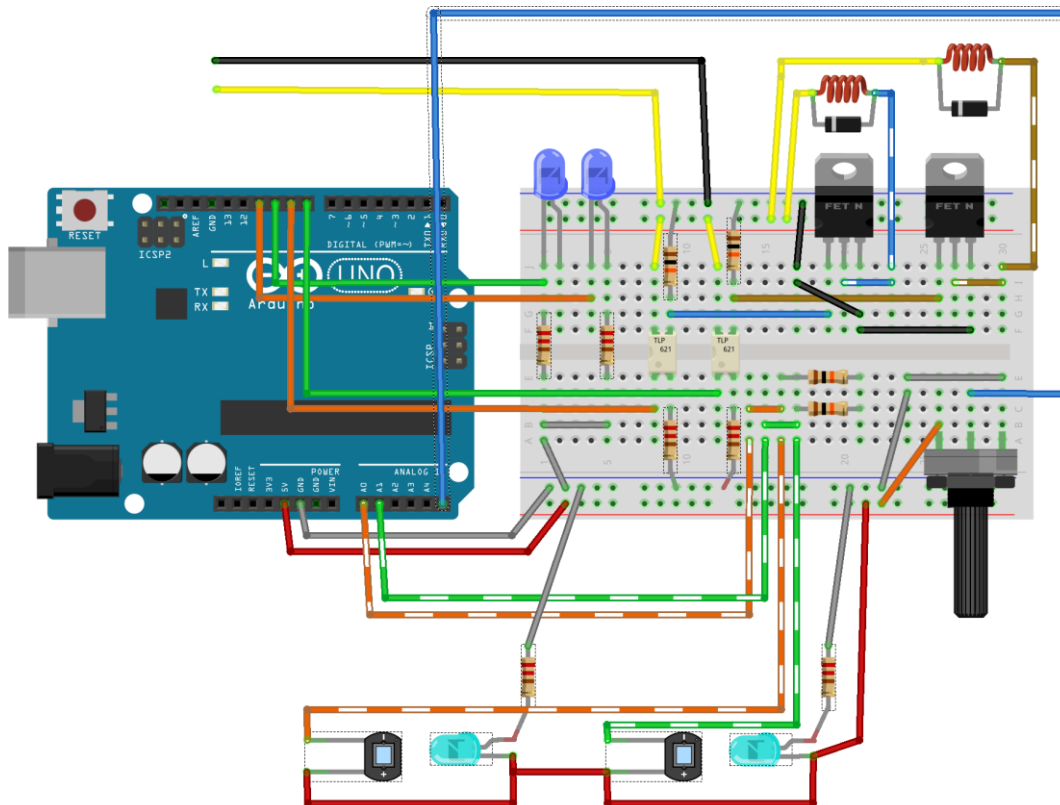
## Materiais necessários

Item	Descrição	Quantidade
1	Fonte ATX de 400 watts ou superior	01
2	Ventiladores 12 V de velocidade fixa, tipo ventoinha de fonte	04
3	Capacitores eletrolíticos 5000 $\mu$ F x 35 V ou superior	04
4	Fio esmaltado AWG 18	16 m
5	Placa de compensado medindo 550 x 360 x 15 mm	01
6	Mangueira plástica flexível e transparente com diâmetro interno de 5/8 ”	1,5 m
7	Mangueira plástica flexível e transparente com diâmetro interno de 3/4 ”	0,5 m
8	Esfera de aço com 10, 11 ou 12 mm de diâmetro	01
9	MOSFET tipo N 50N06 com dissipadores de calor	02
10	Optoacopladores PC817 ou 4N35	02
11	Diodos 1N4002 ou 1N4007	02
12	Resistores 10 k $\Omega$ x 1/8 w	04
13	Resistores 220 $\Omega$ x 1/8 w	06
14	LEDs de alto brilho branco	04
15	Fototransistores infravermelhos	02
16	Potenciômetro de 10 k $\Omega$	01

17	Placa Arduino UNO	01
18	Protoboard 830 pontos – Matriz de contatos elétricos para montagem de circuitos	01
19	Diversos (cabinhos, presilhas plásticas, etc.)	---

Montando o experimento





## Código do Arduino

```

unsigned long tempo = 0;
unsigned long tempoVolta = 0;
int tempoVoltaInt;
int sensor1 = A0;
int sensor2 = A3;
int pot = A5;
int bobina1 = 8;
int bobina2 = 9;
int menorTempo = 11;
int menorValor = 900;
int valDelay;
int valSensor1;
int valSensor2;
int valPot;
int serial = 0;
int numeroVoltas = 0;
boolean controle1 = LOW;
boolean controle2 = LOW;
void setup() {
  pinMode(bobina1, OUTPUT);
  pinMode(bobina2, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(2, INPUT);
  Serial.begin(250000);
}
void loop() {
  valSensor1 = analogRead(sensor1);
  valSensor2 = analogRead(sensor2);
  valPot = analogRead(pot)/16;
  valDelay = (menorTempo+valPot);
  if (valSensor1<menorValor && controle1 == LOW){
    controle1 = HIGH;
    digitalWrite(bobina1, HIGH);
    digitalWrite(10, HIGH);
    delay(valDelay);
    digitalWrite(bobina1, LOW);
    digitalWrite(10, LOW);
    tempoVolta = millis()-tempo;
    tempoVoltaInt=int(tempoVolta);
    tempo = millis();
  }
  if (valSensor2 < menorValor && controle2 == LOW){
    controle2 = HIGH;
    digitalWrite(bobina2, HIGH);
    digitalWrite(11, HIGH);
    delay(valDelay);
    digitalWrite(bobina2, LOW);
    digitalWrite(11, LOW);
    numeroVoltas = numeroVoltas + 1;
  }
  if (valSensor1 > menorValor && controle1 ==
HIGH){ controle1 = LOW;}
  if (valSensor2 > menorValor && controle2 ==
HIGH){ controle2 = LOW;}
  serial = serial+1;
  if (serial > 1000){
    Serial.print(valDelay);
    Serial.print(",");
    Serial.print(tempoVoltaInt);
    Serial.print(",");
    Serial.println(numeroVoltas);
    serial = 0;
  }
}

```



## Código do Processing

<pre>import processing.serial.*; // Variáveis String myString = null; Serial myPort; int[] nums; int lf = 10; int value = 0; int tempo = 0; int periodo = 0; int tempoDisparo = 0; int periodoDisparo = 0; int simNumeroVoltas = 0; int controleDisparo=0; int expTempoDisparo = 0; float expPeriodo = 0; int expNumeroVoltas = 0; float simXesfera = 0.0; float simYesfera = 0.0; float simAngulo = 0.0; float incremento = 0; float simPeriodo = 0; float simFrequencia; float simVelocidadeAngular; float simVelocidadeLinear; float simDistanciaPercorrida; float expFrequencia = 0; float expVelocidadeAngular = 0; float expVelocidadeLinear = 0; float expDistanciaPercorrida = 0; color off = color(4, 79, 111); color on = color(84, 145, 158); boolean expControleSerial = false; int controle = 0; void setup() {   printArray(Serial.list());   myPort = new Serial(this, Serial.list()[0],   250000);   myPort.clear();   myString = myPort.readStringUntil(lf);   myString = null;   size(1000, 600);   frameRate(120);   noSmooth();   background(4, 79, 111);   stroke(on);   fill(255);   textSize(12);   text("Universidade Federal do Estado do   Rio de Janeiro - UNIRIO", 50, 40);   text("Mestrado Nacional Profissional de   Ensino de Física - MNPEF", 50, 55);   text("Orientador: Felipe Mondaini", 50,   70);   text("Mestrando: Luis Henrique M de   Castro", 50, 85);   textSize(20);   fill(102);   stroke(0);   rect(180, 95, 650, 35);   fill(255);   text("Maquete Funcional de uma   Analogia ao Acelerador de Partículas",   200, 120);   fill(255);   textSize(20);   text("Simulação Simplificada do   Experimento ", 50, 170);   textSize(15);   text("Raio médio da trajetória (cm): ",   275, 200);   text("2,250", 520, 200);</pre>	<pre>stroke(255);fill(255); text(" START ", 660, 520); stroke(0);fill(160,60,60); rect(750, 500, 70, 25,5); stroke(255);fill(255); text(" STOP ", 760, 520); } void draw() {   stroke(0);fill(0);   //Simulação   simAngulo = simAngulo + incremento;   if (simAngulo &gt;= (2*PI)){simAngulo =   0.0;   periodo = millis()- tempo;   tempo = millis();   simPeriodo = periodo/1000.0;   simFrequencia = 1/simPeriodo;   simVelocidadeAngular =   simFrequencia*(2*PI);   simVelocidadeLinear =   simVelocidadeAngular*2.25;   simNumeroVoltas = simNumeroVoltas   +1;   simDistanciaPercorrida =   simNumeroVoltas * 2.250;   fill(4, 79, 111);   stroke(4, 79, 111);   rect(500, 260, 98, 225);   fill(255);   text(periodoDisparo, 520, 290);   text(simPeriodo, 515, 320);   text(simFrequencia, 515, 350);   text(simVelocidadeAngular, 515, 380);   text(simVelocidadeLinear, 515, 410);   text(simNumeroVoltas, 520, 440);   text(simDistanciaPercorrida, 515, 470); } simXesfera = 150.0 + 98 * cos(simAngulo); simYesfera = 300.0 + 98 * sin(simAngulo); fill(200); ellipse(150, 300, 210, 210); fill(off); ellipse(150, 300, 180, 180); fill(255); ellipse(simXesfera, simYesfera, 10, 10); if (simXesfera &gt; 155 &amp;&amp; simYesfera &gt;385){   fill(200, 0, 0, 200); rect(125, 380, 50,   30); } else{   fill(200, 150, 50, 200); rect(125, 380,   50, 30); } if (simXesfera &lt; 145 &amp;&amp; simYesfera &lt; 215){   fill(200, 0, 0, 200); rect(125, 190, 50,   30); } if(controleDisparo == 0){   tempoDisparo = millis();   controleDisparo = 1; } } else{   fill(200, 150, 50, 200); rect(125, 190,   50, 30); } if(controleDisparo == 1){   periodoDisparo = millis()-   tempoDisparo;   controleDisparo=0;</pre>	<pre>} else{   stroke(0);fill(160,160,60);   rect(475, 500, 70, 25,5);   stroke(255);fill(255);   text(" RESET ", 485, 520); } // Experimento if (expControleSerial == true) {   while (myPort.available() &gt; 0) {     myString = myPort.readStringUntil(lf);     if (myString != null) {       nums = int(split(myString,','));       controle +=1;     }     if (controle&gt;5){       print(" disparo "); print(nums[0]);       print(" tempo "); print(nums[1]);       print(" voltas "); println(nums[2]);       expTempoDisparo = int(nums[0]);       expPeriodo = float(nums[1]);       expNumeroVoltas = int(nums[2]);       expFrequencia = 1000/expPeriodo;       expVelocidadeAngular =       expFrequencia*(2*PI);       expVelocidadeLinear =       expVelocidadeAngular*0.17;       expDistanciaPercorrida =       expNumeroVoltas * 1.07;     }     fill(44, 105, 118);     stroke(44, 105, 118);     rect(885, 265, 100, 225);     fill(255);     text(expTempoDisparo, 900, 290);     text(expPeriodo, 895, 320);     text(expFrequencia, 895, 350);     text(expVelocidadeAngular, 895, 380);     text(expVelocidadeLinear, 895, 410);     text(expNumeroVoltas, 900, 440);     text(expDistanciaPercorrida, 895, 470);   } } if (mouseX&gt;650 &amp;&amp; mouseX&lt;720 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) {   stroke(0);fill(100,200,100);   rect(650, 500, 70, 25,5);   stroke(0);fill(0);   text(" START ", 660, 520); } else{   stroke(0);fill(60,160,60);   rect(650, 500, 70, 25,5);   stroke(255);fill(255);   text(" STOP ", 760, 520); } if (mouseX&gt;750 &amp;&amp; mouseX&lt;820 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) {   stroke(0);fill(200,100,100);   rect(750, 500, 70, 25,5);   stroke(0);fill(0);   text(" STOP ", 760, 520); } else{   stroke(0);fill(160,60,60);   rect(750, 500, 70, 25,5);   stroke(255);fill(255);   text(" STOP ", 760, 520); } if (mouseX&gt;850 &amp;&amp; mouseX&lt;920 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) {   stroke(0);fill(200,200,100);   rect(850, 500, 70, 25,5);   stroke(0);fill(0);   text(" RESET ", 860, 520);</pre>
--	--	---

<pre> text("Perímetro da trajetória (cm): ", 275, 230); text("14,137", 520, 230); text("Controle de velocidade: ", 275, 260); fill(0); rect(520, 248, 15, 15); rect(540, 248, 15, 15); fill(255); text("+", 523, 260); text("-", 543, 260); text("Tempo de disparo (ms): ", 275, 290); text("Período (s): ", 275, 320); text("Frequência (Hz): ", 275, 350); text("Velocidade angular (rad/s): ", 275, 380); text("Velocidade linear (cm/s): ", 275, 410); text("Número de voltas: ", 275, 440); text("Distância percorrida (cm): ", 275, 470); stroke(0);fill(60,160,60); rect(275, 500, 70, 25,5); stroke(0);fill(160,60,60); rect(375, 500, 70, 25,5); stroke(0);fill(160,160,60); rect(475, 500, 70, 25,5); stroke(255);fill(255); text(" START ", 285, 520); text(" STOP ", 385, 520); text(" RESET ", 485, 520); fill(44, 105, 118); stroke(255); rect(600, 140, 390, 530); fill(255); textSize(20); text("Dados Reais do Experimento ", 650, 170); textSize(15); text("Raio médio da trajetória (m): ", 650, 200); text("0,17", 900, 200); text("Perímetro da trajetória (m): ", 650, 230); text("1,07", 900, 230); text("Massa da esfera (g): ", 650, 260); text("5,586", 900, 260); text("Tempo de disparo (ms): ", 650, 290); text("Período (ms): ", 650, 320); text("Frequencia (Hz): ", 650, 350); text("Velocidade angular (rad/s): ", 650, 380); text("Velocidade linear (m/s): ", 650, 410); text("Número de voltas: ", 650, 440); text("Distância percorrida (m): ", 650, 470); stroke(0);fill(60,160,60); rect(650, 500, 70, 25,5); stroke(0);fill(160,160,60); rect(850, 500, 70, 25,5); stroke(255);fill(255); text(" START ", 660, 520); text(" STOP ", 760, 520); text(" RESET ", 860, 520); stroke(0);fill(60,160,60); rect(650, 500, 70, 25,5); </pre>	<pre> } } for (int i = 0; i &lt; 50; i=i+5) { stroke(0); line(125+i, 190, 130+i, 220); line(125+i, 380, 130+i, 410); }  strokeWeight(8); stroke(250, 250, 200, 200); line(205, 400, 190, 375); line(115, 225, 100, 200); strokeWeight(1); stroke(0); // Experimento if (mouseX&gt;520 &amp;&amp; mouseX&lt;535 &amp;&amp; mouseY&gt;248 &amp;&amp; mouseY&lt;263) { fill(255); rect(520, 248, 15, 15); fill(0); text("+", 523, 260); } else { fill(0); rect(520, 248, 15, 15); fill(255); text("+", 523, 260); } if (mouseX&gt;540 &amp;&amp; mouseX&lt;555 &amp;&amp; mouseY&gt;248 &amp;&amp; mouseY&lt;263) { fill(255); rect(540, 248, 15, 15); fill(0); text("-", 543, 260); } else { fill(0); rect(540, 248, 15, 15); fill(255); text("-", 543, 260); } if (mouseX&gt;275 &amp;&amp; mouseX&lt;345 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { stroke(0);fill(100,200,100); rect(275, 500, 70, 25,5); stroke(0);fill(0); text(" START ", 285, 520); } else { stroke(0);fill(60,160,60); rect(275, 500, 70, 25,5); stroke(255);fill(255); text(" START ", 285, 520); } if (mouseX&gt;375 &amp;&amp; mouseX&lt;445 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { stroke(0);fill(200,100,100); rect(375, 500, 70, 25,5); stroke(0);fill(0); text(" STOP ", 385, 520); } else { stroke(0);fill(160,60,60); rect(375, 500, 70, 25,5); stroke(255);fill(255); text(" STOP ", 385, 520); } if (mouseX&gt;475 &amp;&amp; mouseX&lt;545 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { stroke(0);fill(200,200,100); rect(475, 500, 70, 25,5); stroke(0);fill(0); text(" RESET ", 485, 520); } </pre>	<pre> } else { stroke(0);fill(160,160,60); rect(850, 500, 70, 25,5); stroke(255);fill(255); text(" RESET ", 860, 520); } } void mousePressed() { if (mouseX&gt;520 &amp;&amp; mouseX&lt;535 &amp;&amp; mouseY&gt;248 &amp;&amp; mouseY&lt;263) { incremento = incremento + 0.0025; if (incremento &gt;= 0.06) {incremento = 0.06;} } if (mouseX&gt;540 &amp;&amp; mouseX&lt;555 &amp;&amp; mouseY&gt;248 &amp;&amp; mouseY&lt;263) { incremento = incremento - 0.0025; if (incremento &lt;= 0) {incremento = 0;} } if (mouseX&gt;275 &amp;&amp; mouseX&lt;345 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { incremento = 0.01; } if (mouseX&gt;375 &amp;&amp; mouseX&lt;445 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { incremento = 0; } if (mouseX&gt;475 &amp;&amp; mouseX&lt;545 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { simAngulo = 0; periodo = 0; tempo = 0; periodoDisparo = 0; simPeriodo = 0; simFrequencia = 0; simVelocidadeAngular = 0; simVelocidadeLinear = 0; simNumeroVoltas = 0; simDistanciaPercorrida = 0; fill(4, 79, 111); stroke(4, 79, 111); rect(500, 260, 98, 225); fill(255); text(periodoDisparo, 520, 290); text(simPeriodo, 515, 320); text(simFrequencia, 515, 350); text(simVelocidadeAngular, 515, 380); text(simVelocidadeLinear, 515, 410); text(simNumeroVoltas, 520, 440); text(simDistanciaPercorrida, 515, 470); } if (mouseX&gt;650 &amp;&amp; mouseX&lt;720 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { printArray(Serial.list()); myPort = new Serial(this, Serial.list()[0], 250000); myPort.clear(); myString = myPort.readStringUntil(If); myString = null; expControleSerial = true; } if (mouseX&gt;750 &amp;&amp; mouseX&lt;820 &amp;&amp; mouseY&gt;500 &amp;&amp; mouseY&lt;525) { expControleSerial = false; myPort.clear(); myPort.stop(); myString = null; myPort = null; } } } </pre>
---	---	---

## Atividade 1: Demonstrar o uso do simulador

### Habilidades e Competências trabalhadas:

- ✓ Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.
- ✓ Conceituar e determinar as principais grandezas envolvidas no Movimento Circular Uniforme.

### Objetivos:

- ✓ Realizar registros adequados em tabelas e planilhas das medições realizadas;
- ✓ Identificar taxas e tendências de variação;
- ✓ Identificar as variáveis presentes no Movimento Circular Uniforme; e
- ✓ Correlacionar variáveis temporais, angulares e lineares.

### Introdução:

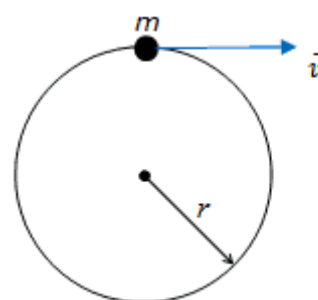
O Movimento Circular Uniforme (MCU) é caracterizado por possuir como trajetória a circunferência e por ter o módulo de sua velocidade constante no decorrer do tempo. No nosso cotidiano, é comum observarmos o movimento realizado por ventiladores, rodas de carros ou de bicicletas, engrenagens, etc.

O tempo que uma partícula leva para percorrer uma única vez a sua trajetória circular é o período ( $T$ ) do movimento. O número de voltas dadas pela partícula na unidade de tempo adotada é a frequência ( $f$ ) do movimento. No Sistema Internacional (SI) a unidade de período é o segundo (s) e de frequência é o Hertz (Hz).

$$T = \frac{1}{f}$$

O módulo da velocidade linear da partícula, considerando o referencial em que ela descreve um MCU, é definido como a distância percorrida sobre a trajetória circular dividida pelo tempo levado para percorrê-la. Assim, tomando como o tempo o período, podemos escrever que o módulo da velocidade linear é:

$$v = \frac{2\pi r}{T}$$



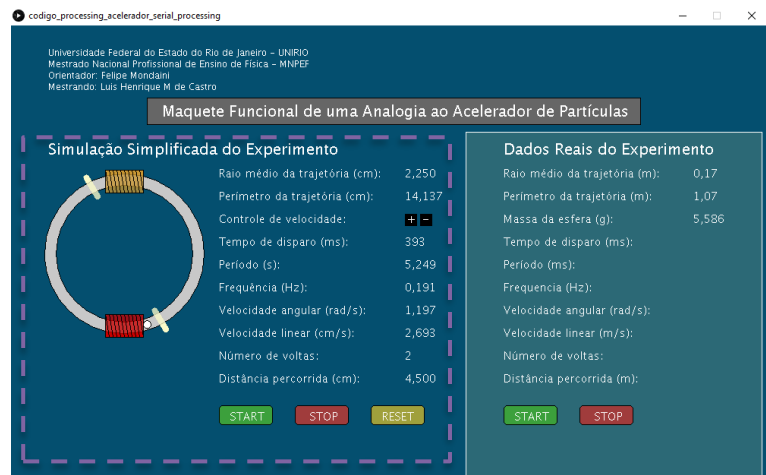
Cabe salientar que o vetor velocidade linear é sempre tangente à trajetória da partícula.

A velocidade angular considera a distância que a partícula percorre sobre a trajetória, considerando como referência o ângulo formado entre o deslocamento da partícula e o centro da trajetória. Este ângulo é representado em radianos.

$$\omega = \frac{2\pi}{T} \text{ ou } \omega = 2\pi f$$

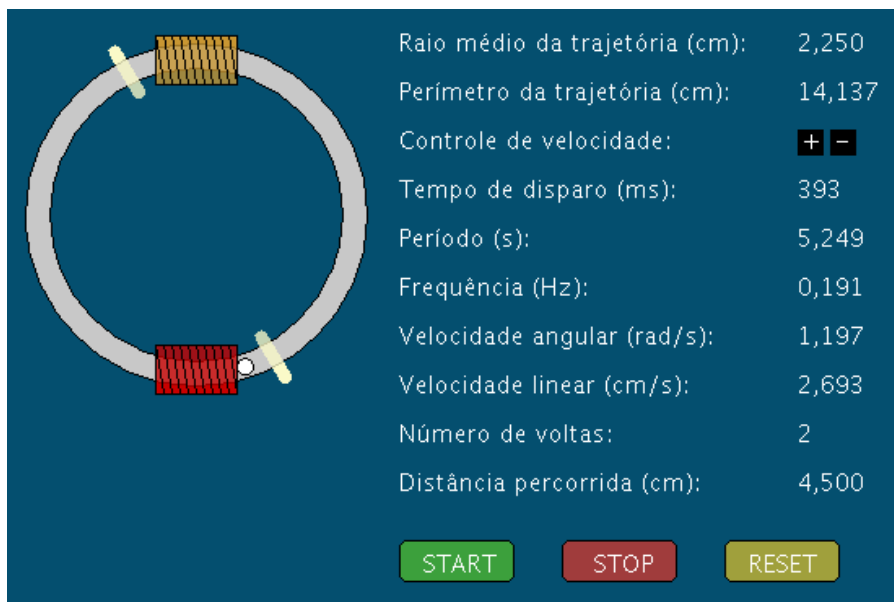
### Procedimento:

Com o auxílio de um cronometro, realizar a medição do tempo de cada volta completa da esfera em sua trajetória circular e comparar ao valor registrado na interface *processing* do simulador. A partir dos valores medidos e calculados, realizar o registro



nas tabelas e determinar o valor médio e o erro relativo das medições.

Repetir este procedimento três vezes com velocidades diferentes.



Ajuste de velocidade	Período registrado no simulador (s)	Medição 1 (s)	Medição 2 (s)	Medição 3 (s)	Valor Médio (s)
Velocidade 1					
Velocidade 2					
Velocidade 3					

Ajuste de velocidade	Período registrado no simulador (s)	Período Médio (s)	Erro relativo (%)
Velocidade 1			
Velocidade 2			
Velocidade 3			
Erro relativo médio			

Com base nos valores registrados na tabela e considerando o raio médio da trajetória igual a 2,25 cm, calcular a velocidade angular e a velocidade linear em cada um dos períodos médios registrados na tabela anterior.

Ajuste de velocidade	Período Médio (s)	Velocidade angular (rad/s)	Velocidade linear (m/s)
Velocidade 1			
Velocidade 2			
Velocidade 3			

## Atividade 2: Demonstrar o uso do experimento.

### Habilidades e Competências trabalhadas:

- ✓ Representar e/ou obter informações de tabelas, esquemas e gráficos de valores de grandezas que caracterizam movimentos ou causas de suas variações; converter tabelas em gráficos e vice-versa; estimar e analisar variações com base nos dados.
- ✓ Conceituar e determinar as principais grandezas envolvidas no Movimento Circular Uniforme.

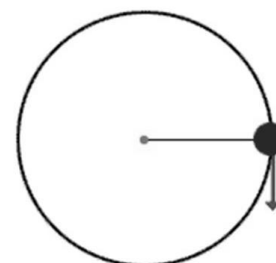
### Objetivos:

- ✓ Realizar registros adequados em tabelas e planilhas das medições realizadas;
- ✓ Identificar taxas e tendências de variação;
- ✓ Identificar as variáveis presentes no Movimento Circular Uniforme; e
- ✓ Correlacionar variáveis temporais, angulares e lineares.

### Introdução:

O Movimento Circular Uniforme (MCU) é caracterizado por possuir como trajetória a circunferência e por ter o módulo de sua velocidade constante no decorrer do tempo. No nosso cotidiano, é comum observarmos o movimento realizado por ventiladores, rodas de carros ou de bicicletas, engrenagens, etc.

O tempo que uma partícula leva para percorrer uma única vez a sua trajetória circular é o período (T) do movimento. O número de voltas dadas pela partícula na unidade de tempo adotada é a frequência (f) do movimento. No Sistema Internacional (SI) a unidade de período é o segundo (s) e de frequência é o Hertz (Hz).



$$T = \frac{1}{f}$$

O módulo da velocidade linear da partícula, considerando o referencial em que ela descreve um MCU, é definido como a distância percorrida sobre a trajetória circular dividida pelo tempo levado para percorrê-la. Assim, tomando como o tempo o período, podemos escrever que o módulo da velocidade linear é:

$$v = \frac{2\pi r}{T}$$

Cabe salientar que o vetor velocidade linear é sempre tangente à trajetória da partícula.

A velocidade angular considera a distância que a partícula percorre sobre a trajetória, considerando como referência o ângulo formado entre o deslocamento da partícula e o centro da trajetória. Este ângulo é representado em radianos.

$$\omega = \frac{2\pi}{T} \text{ ou } \omega = 2\pi f$$

A energia cinética é a energia ligada ao movimento dos corpos. Resulta da transferência de energia do sistema que põe o corpo em movimento.

$$E_c = \frac{1}{2}mv^2$$

**Procedimento:**

Com o auxílio de um cronometro, realizar a medição do tempo de cada volta completa da esfera em sua trajetória e comparar ao valor registrado na interface *processing* do experimento. A partir dos valores medidos e calculados, realizar o registro nas tabelas e determinar o valor médio e o erro relativo das medições.

Ajustar, no potenciômetro, a velocidade mínima de rotação e com o auxílio de um cronometro realizar a medição do tempo de cada volta completa e comparar ao valor registrado na interface. Repetir este procedimento com pequenas variações de velocidade

Ajuste de velocidade	Período registrado no experimento (s)	Medição 1 (s)	Medição 2 (s)	Medição 3 (s)	Valor Médio (s)
Velocidade 1					
Velocidade 2					
Velocidade 3					

Ajuste de velocidade	Período registrado no experimento (s)	Período Médio (s)	Erro relativo (%)
Velocidade 1			
Velocidade 2			
Velocidade 3			
Erro relativo médio			

Com base nos valores registrados na tabela e considerando o raio médio da trajetória igual a 0,17 m e a massa da esfera igual a 5,586 g, calcular a velocidade angular e a velocidade linear em cada um dos períodos médios registrados na tabela.

Ajuste de velocidade	Período Médio (s)	Velocidade angular (rad/s)	Velocidade linear (m/s)	Energia Cinética (J)
Velocidade 1				
Velocidade 2				
Velocidade 3				



### Atividade 3: Demonstrar qualitativamente a aceleração da esfera

#### Habilidades e Competências trabalhadas:

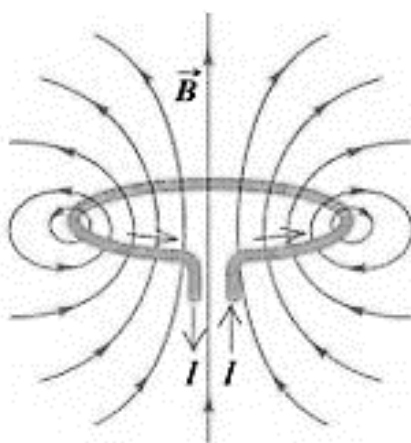
- ✓ Reconhecer a relação entre fenômenos magnéticos e elétricos para explicar o funcionamento de motores elétricos e seus componentes, interações envolvendo bobinas e transformações de energia;
- ✓ Compreender o funcionamento de equipamentos e circuitos baseados em eletrodinâmica e eletromagnetismo.
- ✓ Identificar a atração de um campo magnético gerado por uma bobina e perceber sua interação com objetos ferromagnéticos

#### Objetivos:

- ✓ Medir a corrente que circula em um circuito elétrico
- ✓ Correlacionar a intensidade de corrente elétrica que percorre uma bobina com o campo gerado por ela; e
- ✓ Identificar a atração eletromagnética.

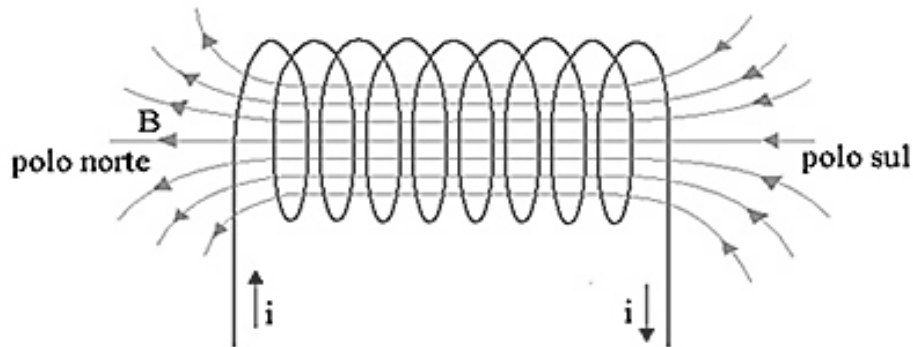
#### Introdução:

Uma carga elétrica em movimento ou uma corrente elétrica percorrendo um condutor produz um campo magnético em sua vizinhança. Este fenômeno é conhecido como eletromagnetismo e possui diversas aplicações em nosso cotidiano, como nos motores elétricos, nas campainhas elétricas, nos eletroímãs, etc.



Com o objetivo de aumentar o campo magnético gerado pelos condutores percorridos por uma corrente elétrica, foram desenvolvidas as bobinas ou solenoides. Estas podem ser definidas como um enrolamento de fio condutor que envolve a superfície de um cilindro.

Quando o condutor da bobina é percorrido por uma corrente elétrica, ele também terá um campo magnético associado e, praticamente, uniforme em seu interior. A bobina tem suas extremidades associadas aos polos norte e sul, um comportamento muito parecido com um ímã natural em forma de barra.



#### Procedimento:

- Desligar o cabo USB, interrompendo a alimentação do *Arduino UNO*;
- Com o auxílio de um ímã ou apenas inclinando o experimento posicione a esfera em uma das barreiras ópticas;
- Identifique o MOSFET chaveador da bobina referente a esta barreira óptica;
- Retire da matriz de contato este MOSFET;
- Selecionar a escala de medição de corrente com pelo menos 20 A de fundo de escala
- Ligar o conector oriundo da bobina ao borne +20 A do multímetro com o auxílio de uma ponteira jacaré;
- Encostar brevemente a ponteira comum do osciloscópio no conector oriundo da fonte de alimentação que permaneceu no protoboard; e
- A esfera será acelerada dentro da mangueira plástica.



## Atividade 4: Análise e analogia ao acelerador de partículas

### Habilidades e Competências trabalhadas:

- ✓ Compreender os procedimentos atuais de pesquisa em laboratórios destinados ao estudo das partículas elementares;
- ✓ Compreender o funcionamento de um acelerador de partículas circular

### Objetivos:

- ✓ Discussão e construção de um modelo explicativo do funcionamento do experimento;
- ✓ Conhecer o modelo padrão e as partículas elementares.

### Introdução:

Um modelo científico busca criar uma representação concreta do resultado de pesquisas científicas. Com a aplicação do modelo, podem surgir novas evidências que culminem na modificação ou substituição do modelo previamente proposto por outro que melhor se adapte aos novos dados obtidos.

Um exemplo desta evolução nos modelos é o modelo atômico, que evoluiu desde a ideia do átomo como partícula indivisível, passou pela descoberta de novas partículas (elétrons, prótons e nêutrons) que seriam as novas partículas indivisíveis, chegando ao modelo padrão em que temos os quarks constituindo os prótons e nêutrons (o elétron continua como partícula elementar) e tantas outras partículas e interações que constituem o objeto de estudo da chamada Física de partículas.

Atualmente, o experimento central no desenvolvimento da Física de Partículas é o acelerador de partículas, como por exemplo, o LHC (*Large Hadron Collider* ou Grande Colisor de Hádrons) do CERN (*European Council for Nuclear Research* - Organização Europeia para Pesquisa Nuclear). Seu funcionamento se baseia na aceleração das partículas gerada a partir da presença de campos elétricos que fornecem novos impulsos às partículas a cada volta. Ímãs de alta potência magnética são utilizados para manter o feixe de partículas em sua trajetória circular. Todo o percurso das partículas é realizado em vácuo quase absoluto e baixas temperaturas. A sua estrutura permite acelerar dois feixes de partículas a velocidades próximas da velocidade da luz em sentidos contrários e então colidi-los. Esta colisão é detectada por diversos sensores que registram as partículas geradas.

Analisando o funcionamento do experimento *Analogia funcional do acelerador de partículas com o uso do Arduino e do Processing*, como você o descreveria fisicamente?

Realize uma comparação entre o funcionamento do experimento e o acelerador de partículas, por exemplo o LHC do CERN, utilizado para fins de pesquisa científica

Desenvolva uma pesquisa sobre o modelo padrão e as principais partículas já encontradas com o uso dos aceleradores de partículas.