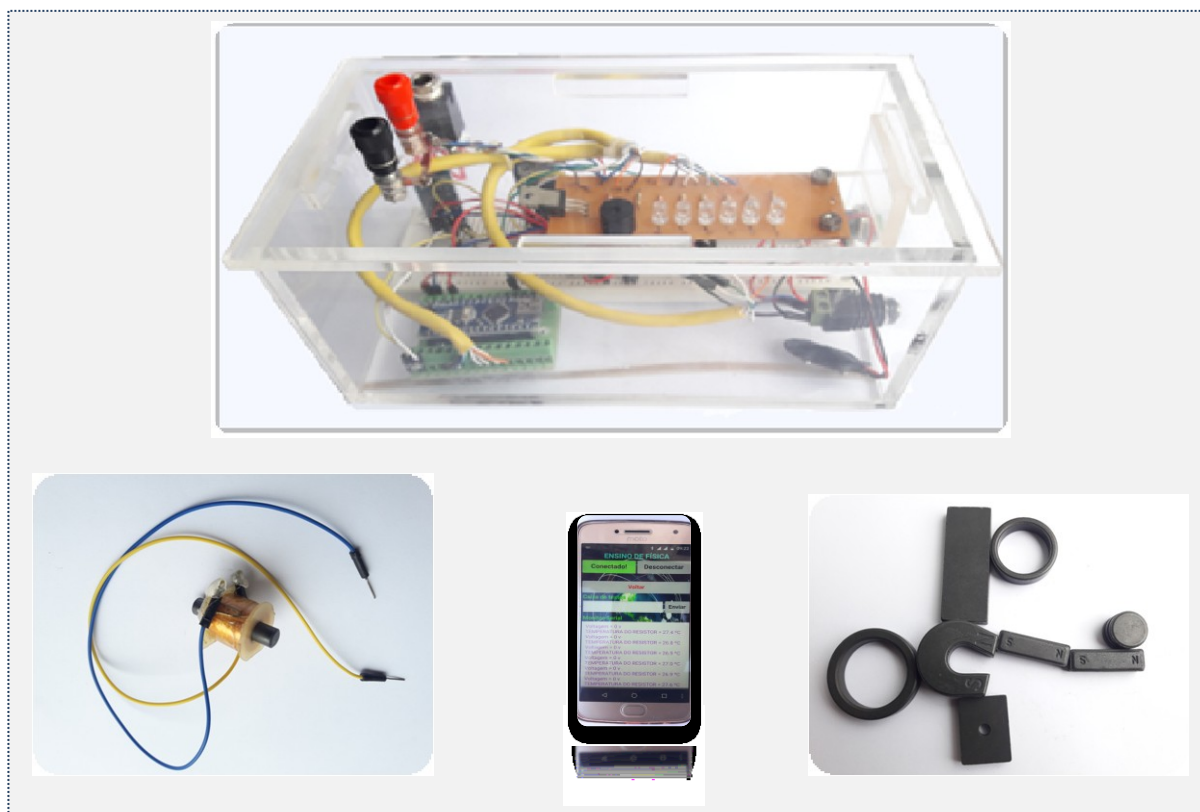


Heberval M. Nunes  
Edson F. V. de Carvalho

## ELETRODUINO: UM KIT EXPERIMENTAL PARA O ENSINO DO ELETROMAGNETISMO





## **ELETRODUINO: UM KIT EXPERIMENTAL PARA O ENSINO DO ELETROMAGNETISMO**

Autor:

**Heberval Moreira Nunes**

Orientador:

**Prof. Dr. Edson Firmino Viana de Carvalho**

UNIVERSIDADE FEDERAL DO MARANHÃO - UFMA

MESTRADO NACIONAL PROFISSIONAL EM ENSINO DE FÍSICA - MNPEF

Foto de capa:

**Heberval Moreira Nunes**

São Luís – MA  
2018

© Heberval Moreira Nunes e Edson Firmino Viana de Carvalho – 2018.

O material apresentado neste documento pode ser reproduzido livremente desde que citada a fonte. As imagens apresentadas são de propriedades dos respectivos autores e utilizadas para fins didáticos. Este documento é veiculado gratuitamente, sem nenhum tipo de retorno comercial a nenhum dos autores, e visa apenas a divulgação do conhecimento científico.

## APRESENTAÇÃO

Prezado Professor,

O presente material contém um produto instrucional didático, fruto do trabalho no curso de Mestrado Nacional Profissional em Ensino de Física (MNPEF), sob a orientação do Prof. Dr. Edson Firmino Viana de Carvalho, no Polo da Universidade Federal do Maranhão. O presente documento foi desenvolvido para auxiliar o professor do Ensino Médio em suas aulas teóricas através das práticas experimentais. Este material instrucional didático tem como objetivo contribuir para o desenvolvimento do processo de ensino e aprendizagem dos conteúdos de Eletromagnetismo, em que o professor poderá utilizar como base pedagógica a teoria da aprendizagem significativa. Todos os detalhes da aplicação deste produto educacional podem ser encontrados na dissertação, de Heberval Moreira Nunes, apresentando o seguinte título: Desenvolvimento e aplicação de um kit experimental com Arduino para o ensino do Eletromagnetismo.

Neste material será apresentado um produto instrucional didático denominado kit Eletroduino. O Eletroduino é um kit experimental envolvente que utiliza tecnologias educacionais, de baixo custo, disponíveis para o ambiente de sala de aula. O kit é composto por: uma programação para um *hardware*; um aplicativo para celular, com sistema operacional Android, denominado de *Ensino de Física* e experimentos de verificação de efeitos observados na Física. O kit experimental Eletroduino poderá ser utilizado para implementar, juntamente com o aplicativo, o estudo qualitativo e quantitativo do eletromagnetismo. Será apresentado, também, um manual para orientá-lo na criação do aplicativo *Ensino de Física* desenvolvido para smartphone ou tablet que poderá ser utilizado nas atividades práticas experimentais no laboratório ou em sala de aula, e traz ainda todas as orientações para ajudá-lo a montar um kit experimental Eletroduino, desenvolvido com o microcontrolador eletrônico Arduino<sup>1</sup> para aquisição de dados experimentais obtidos de forma automática. Este material trás ainda, códigos Arduino para os experimentos; calibração do sensor de campo magnético utilizados nos experimentos. Apresentará também: guias experimentais para os alunos, envolvendo os conceitos de campo magnético, lei de Ohm, efeito Joule em resistores elétricos e outros e, ainda, questionários dos experimentos referentes às atividades práticas utilizando o kit Eletroduino.

---

<sup>1</sup> O Arduino é um pequeno computador usado na aquisição automática de dados. Em que pode ser programado para controlar entradas e saídas eletrônicas entre o dispositivo e os componentes externos conectados a ele.

## Sumário

1. Introdução.....	1
2. Manual de Construção e Uso do Aplicativo.....	2
2.1 Ambiente de Desenvolvimento do AI2.....	2
2.1.1 O Ambiente de Designer .....	3
2.1.2 O Ambiente de Blocos.....	4
2.2 Desenvolvimento do Aplicativo Ensino de Física .....	5
2.2.1 Desenvolvimento do Designer do Aplicativo Ensino de Física .....	6
2.2.2 Desenvolvimento da Lógica de Blocos do Aplicativo Ensino de Física .....	23
2.3 O Aplicativo Ensino de Física .....	37
3. Construção do Kit Experimental Eletroduino .....	40
3.1 A Plataforma Arduino.....	40
3.1.1 Instalação e Utilização do IDE Arduino .....	46
3.2 Materiais e Montagem .....	49
3.2.1 Materiais para o Kit - Campo magnético .....	50
3.2.2 Circuito do Kit - Campo Magnético .....	56
3.2.3 Materiais para o Kit - Efeito Joule.....	60
3.2.4 Circuito do Kit - Efeito Joule .....	60
4. Fundamentos Teóricos de Física .....	64
4.1 A lei de Ohm e o Efeito Joule .....	64
4.2 O Eletromagnetismo .....	65
5. Sequência didática para aplicação do kit Eletroduino.....	67
5.1 Atividade prática 01 – Lei de OHM e Efeito Joule .....	68
5.1.1 Materiais para a atividade prática.....	68
5.1.2 Procedimentos Experimentais .....	68
5.2 Atividade prática 02 – Campo Magnético .....	70
5.2.1 Materiais para a atividade prática.....	70
5.2.2 Procedimentos Experimentais .....	71
Referências Bibliográficas.....	73
Apêndice A .....	74
Códigos dos Programas para o Arduino .....	74

Apêndice B .....	88
Calibração do Sensor 49E.....	88
Apêndice D .....	103
Questionários Referentes às Atividades Experimentais .....	103
Anexo A.....	108
<i>DATASHEET</i> SS49E .....	108

## **1. Introdução**

Qual a situação do ensino de Física no Brasil? A situação é preocupante. No Ensino Médio o professor se depara em sala de aula com alunos que necessitam de práticas pedagógicas novas, diferentes das tradicionais, que geram motivação e despertem nos alunos interesse em querer aprender Física.

O que fazer para motivar os estudantes e tornar as aulas de Física mais atrativas e interessantes? Atualmente, observa-se cada vez mais o uso de tecnologias nas salas de aulas de escolas do Ensino Médio, como recursos didáticos, para auxiliar no desenvolvimento dos processos de ensino e aprendizagem. Isso se deve, principalmente, a disseminação de computadores e programas interativos. Portanto, de maneira geral, as combinações de experimentos didáticos associados à tecnologia têm como objetivos, deixar as aulas de Física mais atrativas e interessantes e melhorar os processos de ensino e aprendizagem de conteúdos, que para este material instrucional, utilizamo-nos de tal combinação para que, o professor, utilize um experimento didático controlado, via celular, associado a uma placa eletrônica, Arduino, que o auxilie nas observações de fenômenos físicos, do Eletromagnetismo, trabalhados nas turmas de terceiro ano do Ensino Médio.

## 2. Manual de Construção e Uso do Aplicativo

O aplicativo utilizado na pesquisa do Mestrado Nacional Profissional em Ensino de Física foi desenvolvido na plataforma MIT App Inventor 2 ou App Inventor 2 (AI2). A plataforma AI2 é um ambiente de programação visual online de código aberto.

O AI2 é uma plataforma educacional atualmente mantida e gerenciada pelo Instituto de Tecnologia de Massachusetts (MIT - *Massachusetts Institute of Technology*).

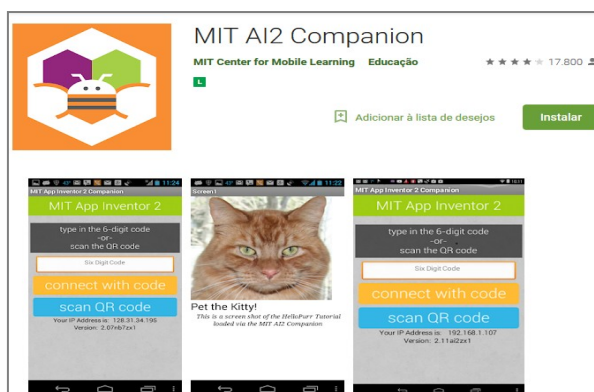
Para desenvolver um aplicativo na plataforma AI2 o usuário precisa acessar a página de internet: <http://ai2.appinventor.mit.edu/>, usando qualquer um dos navegadores: *Google Chrome* 29 ou superior; *Apple Safari* 5 ou superior; *Mozilla Firefox* 23 ou superior. O navegador *Microsoft Internet Explorer* não é suportado. Depois que o usuário fizer *login* através de uma conta *Gmail*, poderá criar um aplicativo para *Smartphone* ou tablet com sistema operacional Android.

### 2.1 Ambiente de Desenvolvimento do AI2

O desenvolvimento de um aplicativo no AI2 é feita em dois ambientes diferentes: o de Designer e o de Blocos.

Para acompanhar o progresso do desenvolvimento do aplicativo em tempo real, o usuário precisará de um emulador virtual instalado no computador ou do aplicativo *MIT AI2 Companion*, figura 1, instalado em um dispositivo com Android conectado na mesma rede. O emulador virtual para Android está disponível no *site* do MIT App Inventor.

Figura 1: Imagens do aplicativo *MIT AI2 Companion*, disponível no *Google Play*.



Fonte: *Google Play*, *MIT AI2 Companion*. Disponível em:

<[https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3&hl=pt\\_BR](https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3&hl=pt_BR)>



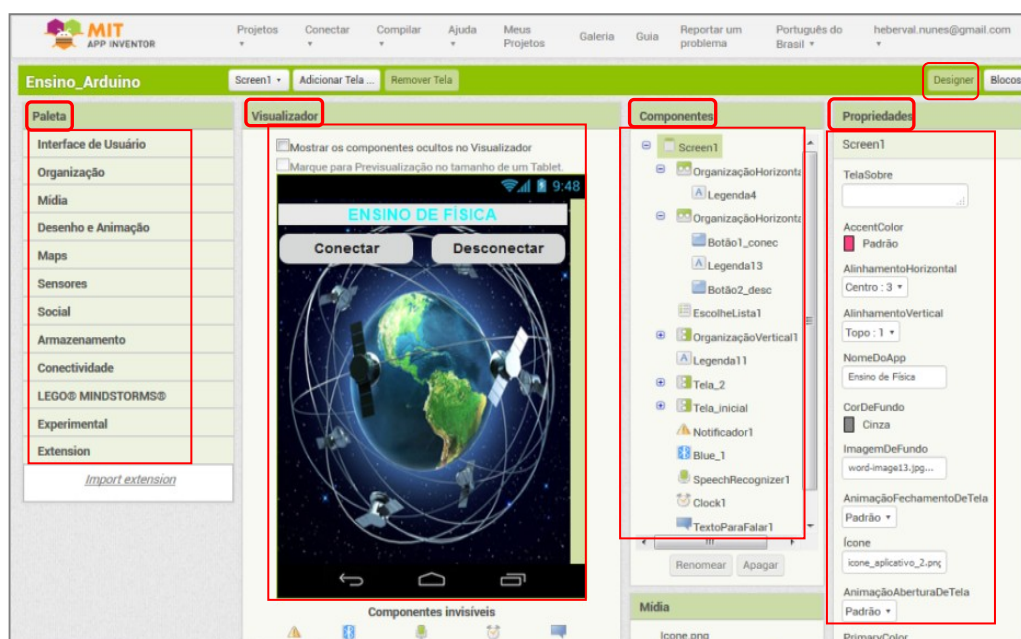
O aplicativo *MIT AI2 Companion* está disponível na loja virtual, Play Store, presente em dispositivos com sistema operacional Android ou no *Google Play* na internet.

Para acompanhar o desenvolvimento do designer do aplicativo em tempo real, sugerimos o uso do emulador *MIT AI2 Companion* por causa da sua praticidade.

### 2.1.1 O Ambiente de Designer

A área de desenvolvimento da aparência (designer) do aplicativo na plataforma AI2, mostrada em destaque na figura 2, é dividida em quatro partes:

Figura 2: As seções do ambiente de Designer do AI2.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552).

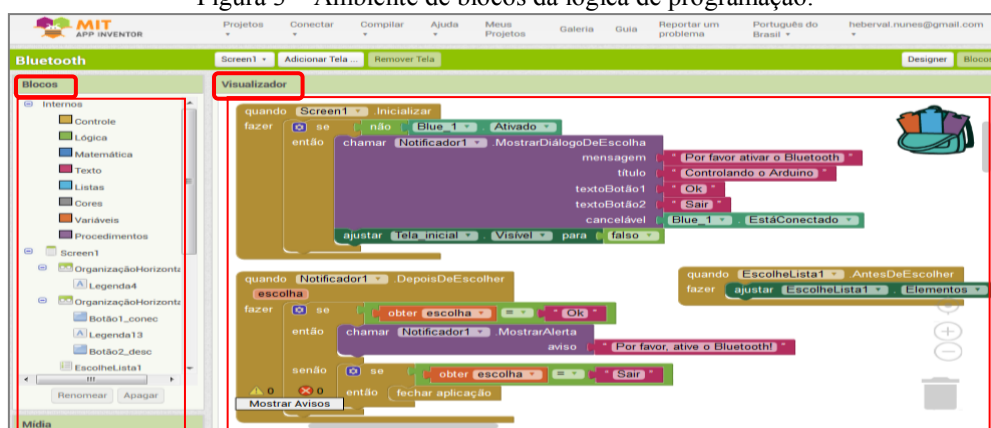
- Paleta – Composta pelas seções: *Interface de Usuário*, *Organização*, *Mídia*, *Desenho e Animação*, *Sensores*, *Social*, *Armazenamento*, *Conectividade* e outros. Todas as seções contêm os componentes que serão utilizados na construção da interface do aplicativo. Para utilizar um componente, o usuário precisa clicar sobre o mesmo e arrastá-lo para dentro da área de visualização de construção do aplicativo (*Screen1*) localizada na parte Visualizador;
- Visualizador – Local de construção do designer do aplicativo. Nessa seção o usuário organiza os componentes arrastados da Paleta para dentro da área *Screen1* e visualiza a forma mais interessante para a aparência da tela do aplicativo;

- Componentes – Local contendo a lista dos componentes que foram adicionados no aplicativo. Nessa área, o usuário pode: alterar as propriedades; deletar ou renomear um componente;
- Propriedades – Local de edição das propriedades de um componente selecionado na área do visualizador ou na seção de componentes. Nessa seção é possível: alterar o tamanho de uma fonte; a cor de um texto; o formato de um botão; imagem de fundo da tela do aplicativo; o título e a imagem do ícone do aplicativo entre outros.

### 2.1.2 O Ambiente de Blocos

O ambiente de blocos do AI2 corresponde o local em que toda a lógica de programação de um aplicativo é desenvolvida. O ambiente de blocos do AI2, mostrado em destaque no exemplo da figura 3, está dividido em duas partes: o ambiente de (Blocos) e o ambiente (Visualizador).

Figura 3 – Ambiente de blocos da lógica de programação.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#6243643214790656](http://ai2.appinventor.mit.edu/?locale=pt_BR#6243643214790656)

A seguir as divisões do ambiente de blocos de programação da plataforma AI2:

- Blocos – área que contém os blocos de comandos internos que serão utilizados na lógica de programação de um aplicativo: *Controle*; *Lógica*; *Matemática*; *Texto*; *Listas*; *Cores*; *Variáveis* e *Procedimentos*. O ambiente de blocos contém, também, os blocos específicos dos componentes adicionados na área de aparência do aplicativo;
- Visualizador – Área de desenvolvimento da lógica de programação do aplicativo. O encaixe dos blocos na área (Visualizador) é parecido com um jogo de quebra-cabeça. Para usar blocos de um determinado elemento presente na tela de designer, o usuário

precisa clicar sobre um componente, na seção (Blocos), e escolher um dos blocos que aparecerá sobre a área (Visualizador). Dessa forma basta arrastar o bloco escolhido para dentro da área de visualização de construção da lógica de programação. Na figura 4, um exemplo dos blocos, em destaque, do botão “Conectar” presente na parte de designer.

Figura 4 – Lógica de programação em blocos.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

## 2.2 Desenvolvimento do Aplicativo Ensino de Física

O aplicativo chamado Ensino de Física foi desenvolvido, como parte do Produto Educacional, para trabalhar em conjunto com o microcontrolador eletrônico Arduino na interpretação e aquisição automática de dados experimentais de Física, em especial estudar alguns conceitos presentes no Eletromagnetismo.

O aplicativo pode ser usado pelo professor em atividades experimentais demonstrativas de fenômenos físicos em sala de aula ou em atividades experimentais investigativas com os alunos trabalhando em equipe.

Com o aplicativo Ensino de Física instalado em um celular com sistema operacional Android é possível, através da comunicação via Bluetooth, controlar e obter dados experimentais coletados pelo Arduino.

Para controlar e monitorar os experimentos de Física, desenvolvemos o aplicativo com: botões (liga/desliga); controles deslizantes proporcionais (*Slides*); controle por comando de voz; controle por escrita de texto de comandos e tela de monitoramento dos dados experimentais (*Monitor Serial do Arduino*).

A seguir será apresentado um guia detalhado de criação do aplicativo Ensino de Física, começando pela parte de designer das telas e depois pela programação lógica de blocos.

## 2.2.1 Desenvolvimento do Designer do Aplicativo Ensino de Física

O designer do aplicativo será desenvolvido em duas etapas: A primeira etapa constará do título do aplicativo e dos botões de conexão via Bluetooth; A segunda etapa será o desenvolvimento da tela de controle contendo os botões (Liga/Desliga) e botões lineares proporcionais (*Slides*), também da tela de monitoramento dos dados experimentais (Monitor Serial Arduino).

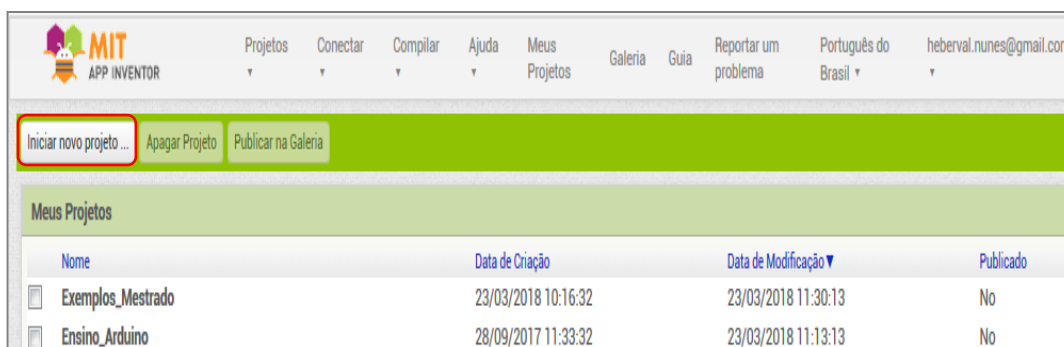
### 2.2.1.1 Primeira Etapa de Criação do Designer do Aplicativo

A primeira etapa de criação do aplicativo, sugerimos começar pela parte do título do aplicativo e depois pela parte de conectividade do celular com o módulo Bluetooth ligado ao Arduino.

A seguir descrevemos o passo a passo do desenvolvimento do *App Ensino de Física*:

1º Passo – Depois de fazer o *login* na plataforma *online* do AI2, o usuário poderá escolher um idioma para trabalhar na construção de um aplicativo. Na sequência o usuário precisará clicar no botão “*Iniciar um novo projeto*”, mostrado em destaque na figura 5;

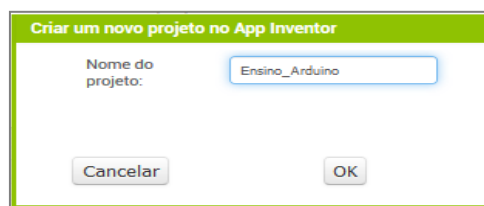
Figura 5 – Iniciar um projeto na plataforma *online* AI2.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

2º Passo – Definir um nome para o projeto do aplicativo. Para nomear o projeto as palavras devem ser escritas sem acentuação e sem espaço, como por exemplo: *Ensino\_Arduino*, mostrado na figura 6;

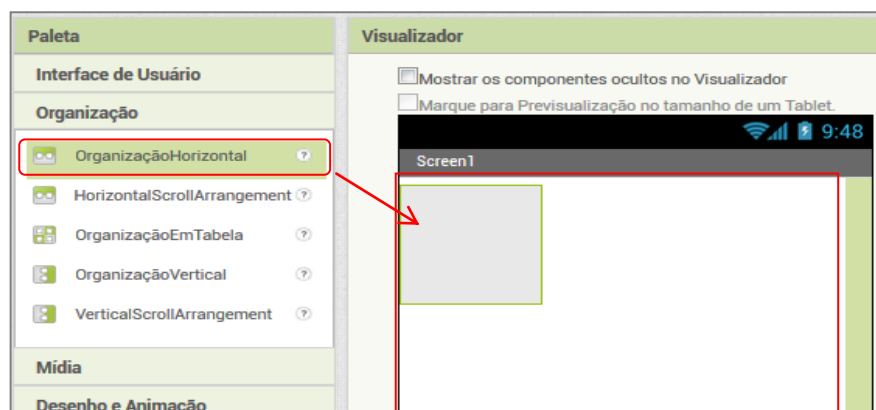
Figura 6 – Nome do projeto no App Inventor.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

3º Passo – Acesse o menu “Organização”, na seção *Paleta*, e em seguida clique e arraste o componente de organização, “OrganizaçãoHorizontal”, para dentro da área Screen1 na seção Visualizador, conforme o exemplo mostrado em destaque na figura 7.

Figura 7 – Componente de organização na área de construção do aplicativo.



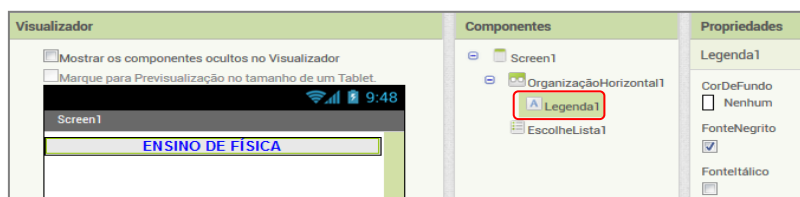
Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#4965029195677696](http://ai2.appinventor.mit.edu/?locale=pt_BR#4965029195677696)

4º Passo – Na sequência clique no componente “OrganizaçãoHorizontal”, localizado na seção Componentes, para fazer as edições das propriedades para: imagem; altura (Preencher principal); largura (Preencher principal) e outras;

5º Passo – Em seguida acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste o componente de texto, “Legenda”, para dentro do retângulo que aparecerá na Screen1 localizada na seção Visualizador;

6º Passo – Clique sobre o componente “Legenda” que aparecerá na seção (Componentes), como mostrado em destaque no exemplo da figura 8. Em seguida, na seção (Propriedades), escreva o título que desejar para o início da tela do aplicativo. O título sugerido para o aplicativo será: Ensino de Física;

Figura 8 – Título do início da tela do aplicativo.



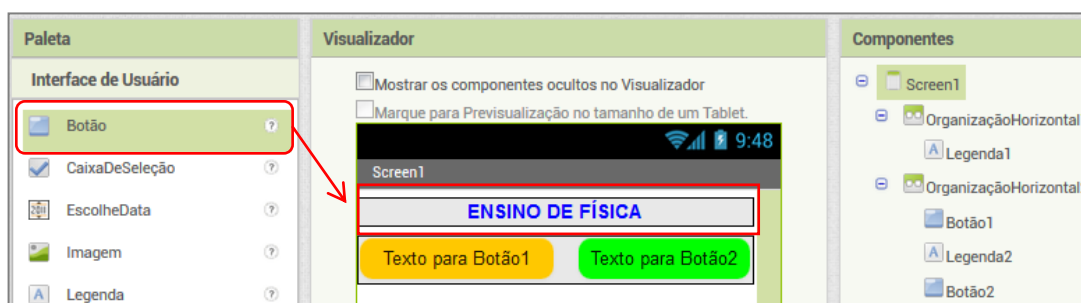
Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#4965029195677696](http://ai2.appinventor.mit.edu/?locale=pt_BR#4965029195677696)

7º Passo – Na seção Componentes, clique em Screen1 para acessar as propriedades: nomear o App; definir cor de fundo da tela; colocar uma imagem de fundo na tela; colocar uma imagem para o ícone que aparecerá no celular ou tablet depois que o aplicativo for instalado. A imagem de fundo utilizada na tela do aplicativo Ensino de Física está disponível em: <<https://br.pinterest.com/pin/596938125580364180/>>;

8º Passo – Para criar os botões de conectividade via Bluetooth, acesse o menu “Organização”, na seção *Paleta*, e em seguida clique e arraste um segundo componente “OrganizaçãoHorizontal” para dentro da área Screen1 na seção Visualizador.

9º Passo – Em seguida acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste dois componentes “Botão” para dentro do segundo retângulo na Screen, como mostrado em destaque na figura 9. Observação: Coloque um componente “Legenda”, sem texto, entre os botões na área Visualizador para separá-los;

Figura 9 – Destaque dos botões de conectividade via Bluetooth.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#4965029195677696](http://ai2.appinventor.mit.edu/?locale=pt_BR#4965029195677696)

10º Passo – Na seção (Componentes) renomear um botão para conectar e o outro para desconectar. Esse procedimento facilita na hora da programação lógica de blocos. Na figura 10, temos em destaque os botões de conectividade renomeados para facilitar a programação de blocos;

Figura 10 – Designer de conectividade do aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

11º Passo – Em seguida acessar o menu “Interface do Usuário”, na *Paleta*, clique e arrastar o componente “EscolheLista” para dentro da Screen1, logo abaixo do componente “OrganizaçãoHorizontal” que contém os botões de conexão via Bluetooth. O componente “EscolheLista” será utilizado na lógica de blocos da programação de conectividade via Bluetooth. Atenção: apague o texto do elemento “EscolheLista”, na seção (Propriedades), para que não fique aparecendo na tela do aplicativo.

### 2.2.1.2 Segunda Etapa de Criação do Designer do Aplicativo

A segunda etapa de criação do designer do aplicativo é a definição de multitelas dentro da Screen1. A finalidade de multitelas, no AI2, é utilizar a mesma conexão para várias telas desenvolvidas em um aplicativo. Isso evita que o usuário tenha que desconectar toda vez que for mudar de tela em um aplicativo. A seguir o passo a passo para criação das telas do aplicativo Ensino de Física:

1º Passo – Coloque um componente “Legenda”, sem texto, para separar os botões de conexão da próxima tela;

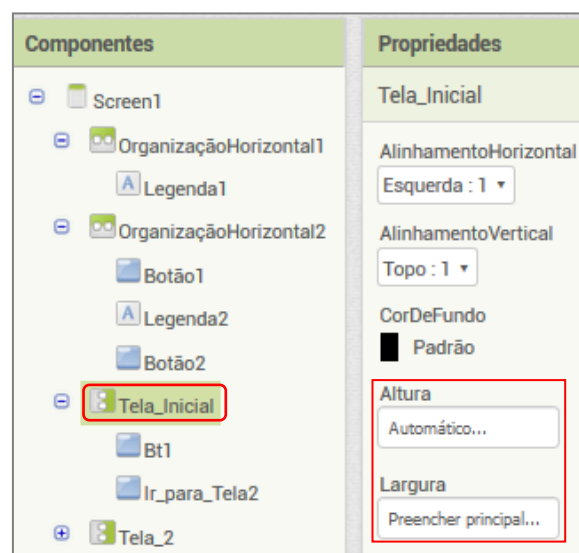
2º Passo – Acesse o menu “Organização”, na seção *Paleta*, e em seguida clique e arrastar o componente “OrganizaçãoVertical” para dentro da área Screen1 na seção Visualizador, depois da “Legenda” adicionada no passo anterior;

3º Passo – Em seguida renomear o componente “OrganizaçãoVertical” adicionado no passo anterior para “Tela\_inicial” na seção Componentes;

4º Passo – Na sequência faça a edição, na seção (Propriedades) do componente “Tela\_inicial”, para: Altura (Automático) e Largura (Preencher principal). Conforme mostrado em destaque na figura 11;

5º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Botão” para dentro do retângulo renomeado “Tela\_inicial”. Esse botão será utilizado para ligar e desligar uma porta de saída do Arduino;

Figura 11 – Tela\_inicial com seus componentes.



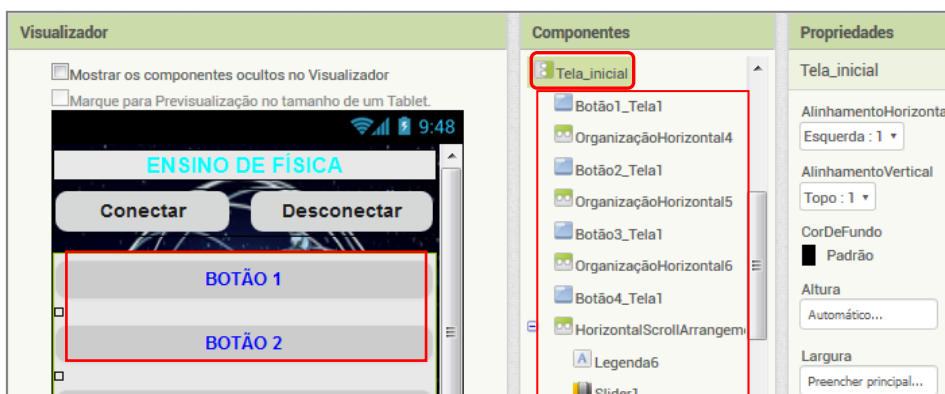
Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#4965029195677696](http://ai2.appinventor.mit.edu/?locale=pt_BR#4965029195677696)

6º Passo – Na seção (Componentes) renomear botão adicionado no passo anterior para “Botão1\_Tela1” ou o nome de sua preferência. Esse procedimento facilita na hora da programação lógica de blocos.

7º Passo – Na sequência, na seção (Propriedades), faça a edição das propriedades do “Botão1\_Tela1” para: Altura (Preencher principal); Largura (Preencher principal); Forma (arredondado); Texto (Botão 1); AlinhamentoDoTexto (centro:1); Cor e tamanho de texto de sua preferência. Na figura 12, temos um exemplo em destaque dos botões “Liga/Desliga” adicionados na primeira tela do aplicativo.



Figura 12 – destaque dos botões na primeira tela do aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#4965029195677696](http://ai2.appinventor.mit.edu/?locale=pt_BR#4965029195677696)

8º Passo – Repetir os passos 5º, 6º e 7º, anteriores, para adicionar outros botões na primeira tela do aplicativo. Para separar os botões adicione um componente “OrganizaçãoHorizontal” ou um componente “Legenda”, sem texto, entre eles. Conforme destacado na figura 13. Como sugestão, adicione mais botões para ligar e desligar portas de saídas do Arduino para futuros projetos de experimentos de Física;

Figura 13 – Botões separados por um componente de organização.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#4965029195677696](http://ai2.appinventor.mit.edu/?locale=pt_BR#4965029195677696)

9º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo renomeado “Tela\_inicial”, depois dos botões “Liga/desliga”;

10º Passo – Na seção (Propriedades), editar o componente “Legenda” adicionado no passo anterior para: Altura (Automático); Largura (Preencher principal); Texto (Slider 1 ou o

nome de sua preferência); AlinhamentoDoTexto (centro:1); CorDeTexto (a cor de sua preferência);

11º Passo – Acesse o menu “Organização”, na seção *Paleta*, clique e arraste um componente “OrganizaçãoHorizontal” para dentro do retângulo renomeado “Tela\_inicial”, logo abaixo do texto (Slider 1) adicionado no passo anterior. Dentro desse “OrganizaçãoHorizontal”, será adicionado: um organizador vertical, um botão deslizante e outro organizado vertical contendo um componente “Legenda”.

12º Passo – Em seguida, editar as propriedades do “OrganizaçãoHorizontal” adicionado no passo anterior, como mostrado na figura 14;

Figura 14 – Propriedades do organizador horizontal.

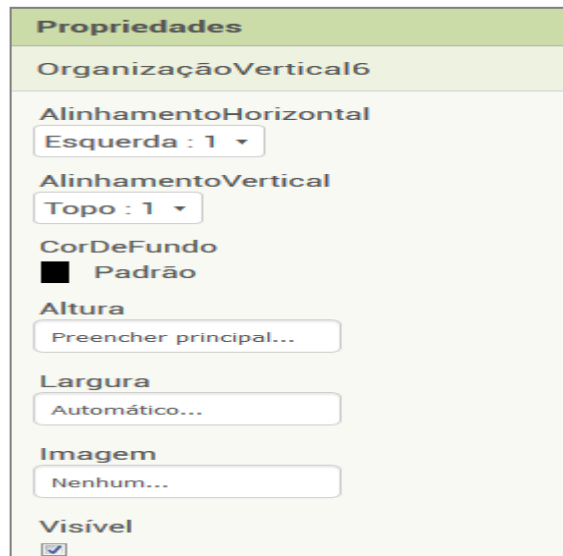


Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

13º Passo – Acesse o menu “Organização”, na seção *Paleta*, clique e arraste um componente “OrganizaçãoVertical” para dentro do retângulo do “OrganizaçãoHorizontal”, adicionado no 11º passo anterior. O “OrganizaçãoVertical” servirá para separar o botão deslizante da borda da tela do celular;

14º Passo – Em seguida faça a edição das propriedades do “OrganizaçãoVertical” adicionado no passo anterior, conforme mostrado na figura 15;

Figura 15 – Propriedades do organizador vertical.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

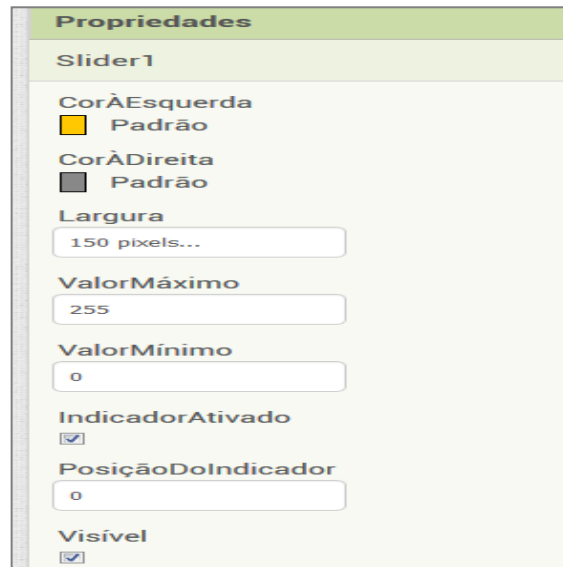
15º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Deslizador” (*slider*) para dentro do retângulo “OrganizaçãoHorizontal”, adicionado no 11º passo anterior. Esse botão será utilizado para fazer o controle proporcional da saída do Arduino. Isto é, o botão deslizante (*slider*) será usado para fazer com que a saída do Arduino tenha uma variação de voltagem de 0v até 5v;

16º Passo – Na seção (Componentes) renomear o componente “Deslizador1”, adicionado no passo anterior, para “Slider1” ou o nome de sua preferência. Esse procedimento facilita na hora da programação lógica de blocos.

17º Passo – Na sequência, em (Propriedades), faça a edição das propriedades do “Slider1”, como mostrado na figura 16. A variação dos valores enviados para o Arduino, pelo aplicativo, será de 0 até 255. Essa é a variação de *bits* (simplificação para dígito binário, do inglês - *Binary digit*) que o Arduino interpreta no intervalo de voltagem de 0v até 5v;

Os controles deslizantes serão utilizados para: controlar a intensidade do campo magnético em uma bobina solenoide, no experimento “Campo magnético” e controlar as correntes elétricas no experimento “Efeito Joule”;

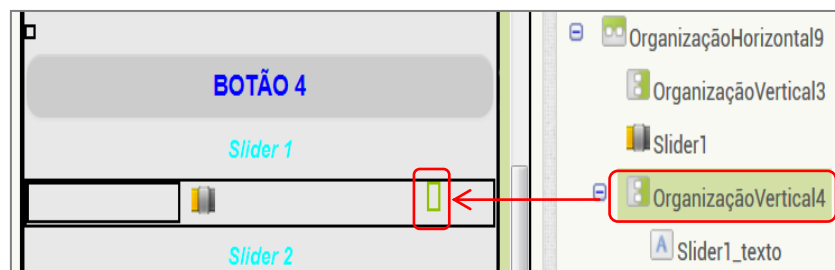
Figura 16 – Propriedades do botão deslizante “Slider1”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

18º Passo – Acesse o menu “Organização”, na seção *Paleta*, clique e arraste um componente “OrganizaçãoVertical” para dentro do retângulo do “OrganizaçãoHorizontal”, adicionado no 11º passo anterior, depois do botão “Slider1”. Em seguida faça a edição das propriedades do “OrganizaçãoVertical” para: AlinhamentoHorizontal (Esquerda:1); AlinhamentoVertical (topo:1); Altura (automático); Lagura (Automático). Esse “OrganizaçãoVertical” conterà o texto que mostrará o progresso do botão “Slider1”, conforme destacado na figura 17;

Figura 17 – organizador vertical conterà o texto que mostrará o progresso do botão “Slider1”.



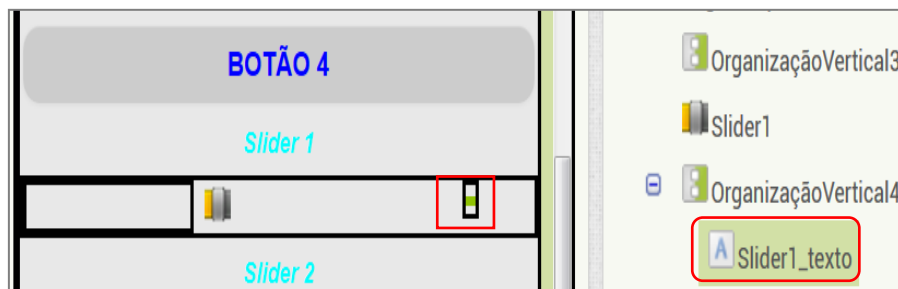
Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

19º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo do “OrganizaçãoVertical”, adicionado no 18º passo anterior, depois do botão “Slider1”.

20º Passo – Na seção (Componentes) renomear o componente “Legenda”, adicionado no passo anterior, para “Slider1\_texto” ou o nome de sua preferência;

21º Passo – Em seguida faça a edição das propriedades do “Slider1\_texto” para: Altura (Automático); Largura (Automático); Texto (Atenção: apague o texto); AlinhamentoDoTexto (centro:1). Na figura 18, em destaque, o componente “Slider1\_texto” dentro do “OrganizadorVertical”. Observação: verificar através de um celular se o controle “Slider” está deslizando com facilidade. Caso não, faça a edição das propriedades dos organizadores verticais entre o botão deslizando;

Figura 18 – O componente “Slider1\_texto” dentro do “OrganizadorVertical”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

22º Passo – Repetir os passos do 11º até o 21º, anteriores, para adicionar outros botões deslizantes na primeira tela do aplicativo. Como sugestão, adicione mais botões para controlar de forma linear as portas de saídas do Arduino para futuros projetos de experimentos de Física;

23º Passo – Acesse o menu “Organização”, na seção *Paleta*, clique e arraste um componente “HorizontalScrollArrangement” para dentro do retângulo renomeado “Tela\_inicial”, logo abaixo do último “OrganizaçãoHorizontal” contendo os botões deslizantes;

24º Passo – Em seguida faça a edição das propriedades do “HorizontalScrollArrangement” para: Altura (Automático); Largura (Preencher principal); Alinhamentohorizontal (centro:3); AlinhamentoVertical (Topo:1);

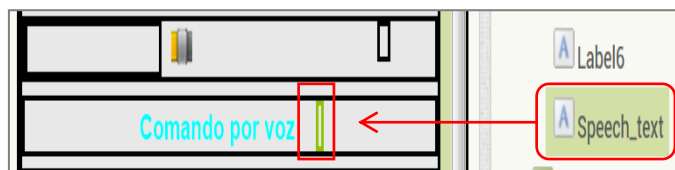
25º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo “HorizontalScrollArrangement” adicionado no 23º passo anterior;

26º Passo – Em seguida faça a edição das propriedades do componente “Legenda”, adicionado no passo anterior, para: Altura (Automático); Largura (Automático); Texto (Comando por voz); AlinhamentoDoTexto (esquerdo: 0); CorDeTexto (de sua preferência);

27º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo “HorizontalScrollArrangement”, adicionado no 23º passo anterior, logo depois do texto “Comando por voz”;

28º Passo – Na seção (Componentes) renomear o componente “Legenda”, adicionado no passo anterior, para “Speench\_text” ou o nome de sua preferência. Na figura 19, em destaque dentro do organizador “HorizontalScrollArrangement”, o componente “Speench\_text”;

Figura 19 – Destaque do componente do comando por voz, “Speench\_text”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

29º Passo – Em seguida faça a edição das propriedades do componente renomeado “Speench\_text” para: Altura (Automático); Largura (Automático); Texto (Atenção: apague o texto); AlinhamentoDoTexto (esquerdo: 0); CorDeTexto (de sua preferência). O componente “Speench\_text” mostrará na tela do celular, o texto referente à voz gravada pelo microfone durante o comando por voz.

30º Passo – Acesse o menu “Organização”, na seção *Paleta*, e em seguida clique e arrastar o componente “OrganizaçãoHorizontal” para dentro da área Screen1 na seção Visualizador, depois do retângulo do organizador “HorizontalScrollArrangement” contendo o componente “Speench\_text”;

31º Passo – Em seguida faça a edição das propriedades do “OrganizaçãoHorizontal” adicionado no passo anterior para: Altura (Preencher principal); Largura (Preencher principal); AlinhamentoHorizontal (centro:3); AlinhamentoVertical (Topo:1);

32º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Botão” para dentro do retângulo “OrganizaçãoHorizontal”, adicionado no passo anterior. Na sequência para facilitar a programação de blocos, na seção (componentes), renomear o componente “Botão” para “Ir\_para\_ Tela2”;

33º Passo – Em seguida faça a edição das propriedades do componente renomeado “Ir\_para\_ Tela2” para: Altura (Preencher principal); Largura (Preencher principal); Forma (arredondado); Texto (Ir para Tela 2); AlinhamentoDoTexto (centro: 1); CorDeTexto (de sua preferência);

34º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo “OrganizaçãoHorizontal”, depois do botão “Ir para Tela 2”. O componente “Legenda” adicionado servirá para separar o botão “Ir para Tela 2” do botão “Microfone”;

35º Passo – Faça a edição das propriedades do componente “Legenda”, adicionado no passo anterior para: TamanhoDaFonte (14.0); Altura (Preencher principal); Largura (Automático); Texto (Atenção: apague o texto); AlinhamentoDoTexto (centro: 1);

36º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Botão” para dentro do retângulo “OrganizaçãoHorizontal”, depois do componente “Legenda” adicionado no 34º. Em seguida, na seção (Componentes), renomear o “Botão” para “Mic”;

37º Passo – Faça a edição das propriedades do componente renomeado “Mic” para: Altura (Preencher principal); Largura (Preencher principal); Forma (arredondado); Texto (Microfone); AlinhamentoDoTexto (centro: 1); CorDeTexto (de sua preferência). Na figura 20, temos em destaque os botões “Ir para Tela 2” e “Microfone”;

Figura 20 – Os botões “Ir para Tela 2” e “Microfone”



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

38º Passo – Acesse o menu “Organização”, na seção *Paleta*, e em seguida clique e arrastar o componente “OrganizaçãoVertical” para dentro da área Screen1, na seção Visualizador, depois do retângulo da “Tela\_inicial”.

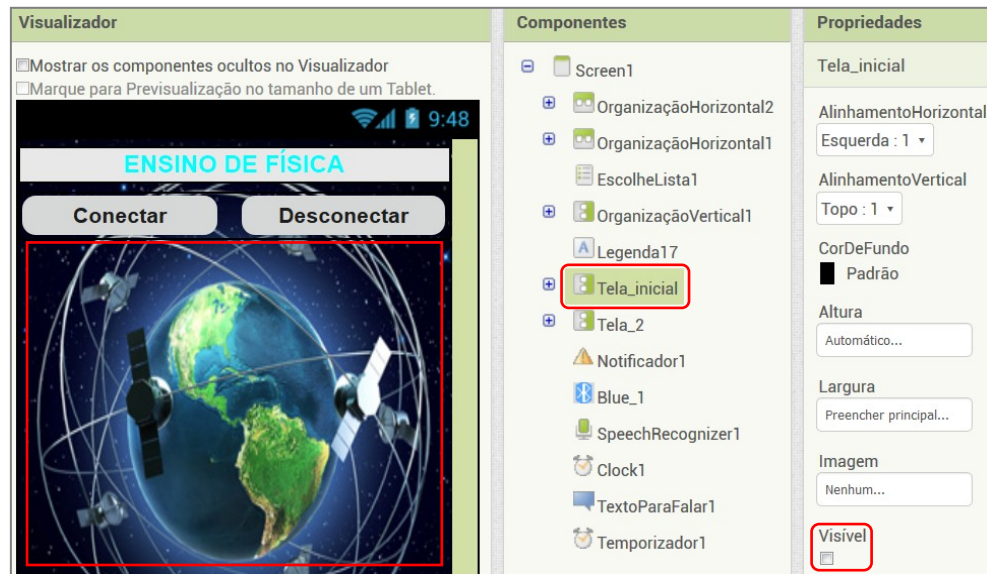
39º Passo – Em seguida renomear o componente “OrganizaçãoVertical” adicionado no passo anterior para “Tela\_2” na seção Componentes;

40º Passo – Em seguida faça a edição das propriedades do componente renomeado, “Tela\_2”, para: Altura (Preencher principal); Largura (Preencher principal); Alinhamentohorizontal (esquerda: 1); AlinhamentoVertical (Topo:1);

41º Passo – Desmarque a opção visível nas propriedades da tela 1 (“Tela\_inicial”), como destacado na figura 21. Deixe visível somente a tela 2 até o final da construção do aplicativo. Dessa forma os componentes da “Tela\_inicial” ficarão ocultos, facilitando o desenvolvimento dos trabalhos na tela 2 (“Tela\_2”). No final da criação do aplicativo, deixe as telas com a opção visível desmarcado. As telas inicial e tela 2, só aparecerão quando o dispositivo estiver conectado. A imagem de fundo da “Tela inicial” do aplicativo, em destaque na área do visualizador do ambiente de blocos (figura 21), está disponível em no site: <<https://br.pinterest.com/pin/596938125580364180/>>;



Figura 21 – “Tela\_inicial” com os componentes ocultos.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

42º Passo – Acesse o menu “Interface de Usuário”, na seção *Paleta*, e em seguida clique e arrastar o componente “Notificador” para dentro da área Screen1, na seção Visualizador. O componente “Notificador” será usado para notificação de conexão via Bluetooth;

43º Passo – Acesse o menu “Conectividade”, na seção *Paleta*, e em seguida clique e arrastar o componente “ClienteBluetooth” para dentro da área Screen1, na seção Visualizador. O componente “ClienteBluetooth” será usado para conectar o smartphone ou tablet com o Arduino via módulo Bluetooth;

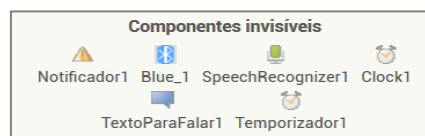
44º Passo – Acesse o menu “Mídia”, na seção *Paleta*, e em seguida clique e arrastar o componente “ReconhecedorDeVoz” para dentro da área Screen1, na seção Visualizador. O componente “ReconhecedorDeVoz” será usado para controlar Arduino, via módulo Bluetooth, por comando de voz;

45º Passo – Acesse o menu “Mídia”, na seção *Paleta*, e em seguida clique e arrastar o componente “TextoParaFalar” para dentro da área Screen1, na seção Visualizador. O componente “TextoParaFalar” será usado para, através do smartphone ou tablet, falar frases escritas na lógica de programação de blocos ou enviadas pelo Arduino.

46º Passo – Acesse o menu “Sensores”, na seção *Paleta*, e em seguida clique e arrastar dois componentes “Temporizador” para dentro da área Screen1, na seção

Visualizador. Renomear o primeiro “Temporizador” para “Clock1” com as seguintes propriedades: “DisparosContínuos” (marcado); “Ativado” (marcado); “Intervalo” (400). Na sequência, renomear o segundo para “Temporizador” para “Temporizador1” com as seguintes propriedades: “DisparosContínuos” (marcado); “Ativado” (marcado); “Intervalo” (1000). Os componentes “Temporizadores” serão usados na lógica de programação de blocos dos botões deslizantes (Slider) e no monitor serial Arduino. Na figura 22, os componentes adicionados na área Screen1 do aplicativo;

Figura 22 – Componentes presentes na lógica de programação do aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

47º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Botão” para dentro do retângulo renomeado “Tela\_2”. Esse botão será utilizado voltar para tela 1 (“Tela\_inicial”);

48º Passo – Na seção (Componentes) renomear o botão adicionado no passo anterior para “Voltar” ou o nome de sua preferência. Em seguida faça a edição das propriedades do botão “Voltar” para: Altura (Preencher principal); Largura (Preencher principal); Forma (arredondado); Texto (Voltar); AlinhamentoDoTexto (centro: 1); CorDeTexto (de sua preferência);

49º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo renomeado “Tela\_2”, logo abaixo do botão “Voltar”;

50º Passo – Na seção (Propriedades), faça a edição das propriedades do componente “Legenda” adicionado no passo anterior para: Altura (Automático); Largura (Automático); Texto (Caixa de textos); AlinhamentoDoTexto (esquerda: 0); CorDeTexto (de sua preferência);

51º Passo – Acesse o menu “Organização”, na seção *Paleta*, clique e arraste um componente “OrganizaçãoHorizontal” para dentro do retângulo renomeado “Tela\_2”, logo abaixo da legenda (Caixa de textos), na seção Visualizador;

52º Passo – Faça a edição das propriedades do componente “OrganizaçãoHorizontal” adicionado no passo anterior para: Altura (Preencher principal); Largura (Preencher principal); AlinhamentoHorizontal (esquerda: 1); AlinhamentoVertical (Topo:1);

53º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “CaixaDeTexto” para dentro do retângulo do componente “OrganizaçãoHorizontal” adicionado no 51º passo, na seção Visualizador;

54º Passo – Faça a edição das propriedades do componente “CaixaDeTexto” adicionado no passo anterior para: CorDefundo (Padrão); Ativado (marcada); TamanhoDaFonte (14.0); Altura (Automático); Largura (Preencher principal); Dica (Em braço, apague o texto); MultiLinha (desmarcado); SomenteNúmeros (desmarcado); Texto (Em branco, apague o texto); AlinhamentoDoTexto (esquerda: 0); CorDeTexto (Padrão);

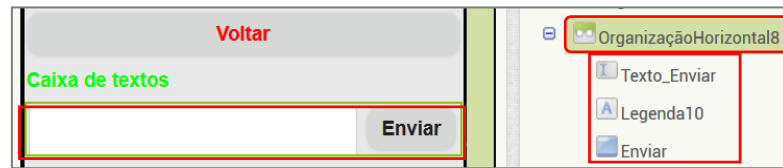
55º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo do componente “OrganizaçãoHorizontal”, adicionado no 51º passo, e solte-o depois do componente “CaixaDeTexto”. Em seguida faça a edição do componente “Legenda” para: TamanhoDaFonte (14.0); Altura (Automático); Largura (Automático); Texto (atenção apague o texto); AlinhamentoDoTexto (esquerda: 0);

56º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Botão” para dentro do retângulo do componente “OrganizaçãoHorizontal” e solte-o depois do componente “Legenda” adicionado no passo anterior. Em seguida renomear o componente “Botão” para “Enviar”. A finalidade do componente “Legenda” é separar a caixa de textos do botão “Enviar”;

57º Passo – Faça a edição das propriedades do botão “Enviar” para: Altura (Automático); Largura (Automático); Forma (arredondado); Texto (Enviar);

AlinhamentoDoTexto (centro: 1); CorDeTexto (de sua preferência). Na figura 23, o organizador com seus componentes em destaque;

Figura 23 – Os componentes “Texto\_Enviar” e “Enviar”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

58º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “Legenda” para dentro do retângulo do componente “OrganizaçãoHorizontal”, logo a baixo do organizador contendo os componentes “Texto\_Enviar” e “Enviar”;

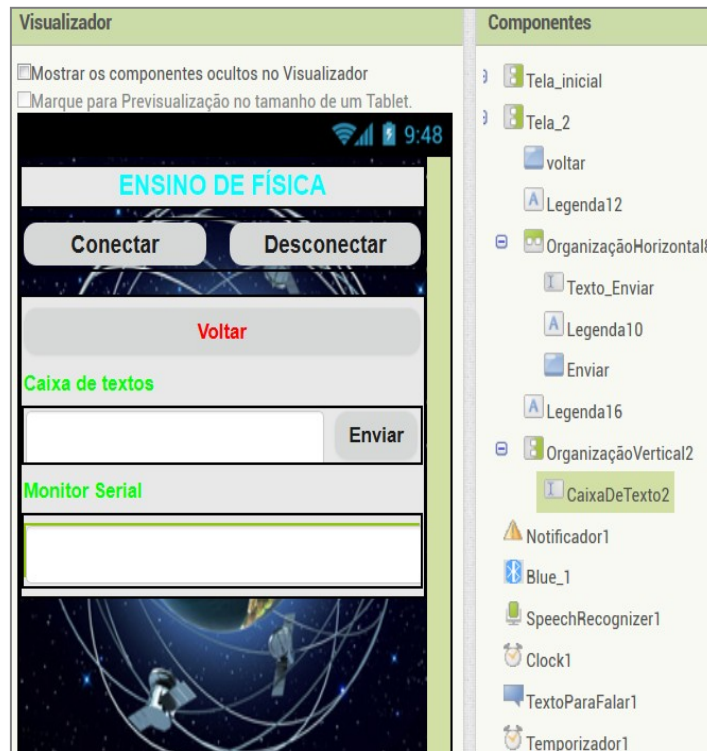
59º Passo – Faça a edição do componente “Legenda”, adicionado no passo anterior, para: TamanhoDaFonte (16.0); Altura (Automático); Largura (Automático); Texto (Monitor Serial); AlinhamentoDoTexto (esquerda: 0); CorDeTexto (de sua preferência);

60º Passo – Acesse o menu “Organização”, na seção *Paleta*, clique e arraste um componente “OrganizaçãoVertical” para dentro do retângulo renomeado “Tela\_2”, logo abaixo da legenda (Monitor Serial), na seção Visualizador;

61º Passo – Faça a edição das propriedades do componente “OrganizaçãoVertical” adicionado no passo anterior para: Altura (Preencher principal); Largura (Preencher principal); AlinhamentoHorizontal (esquerda: 1); AlinhamentoVertical (Topo:1); CorDeFundo (Padrão);

62º Passo – Acesse o menu “Interface do Usuário”, na seção *Paleta*, clique e arraste um componente “CaixaDeTexto” para dentro do retângulo do componente “OrganizaçãoVertical”, adicionado no 60º passo. Em seguida faça a edição das propriedades do componente “CaixaDeTexto” para: CorDeFundo (Branco); Ativado (desmarcado); TamanhoDaFonte (14.0); Altura (Automático); Largura (Preencher principal); Dica (Em branco, apague o texto); MultiLinha (marcado); SomenteNúmeros (desmarcado); Texto (Em branco, apague o texto); AlinhamentoDoTexto (esquerda: 0); CorDeTexto (Padrão). Na figura 24 a tela 2 no ambiente de *designer* do AI2.

Figura 24 – Detalhes da tela 2 no ambiente de *Designer* do AI2



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

### 2.2.2 Desenvolvimento da Lógica de Blocos do Aplicativo Ensino de Física

O desenvolvimento da programação do aplicativo *Ensino de Física* seguirá a seguinte ordem: Começando pela lógica de inicialização do aplicativo; depois pela conectividade via Bluetooth; Lógica de controle dos botões Liga/desliga; Lógica de controle dos botões deslizantes (Slider); Controle através do comando por voz; Lógica dos botões de mudança de tela; Controle através do comando por texto e por último o Serial Monitor Arduino.

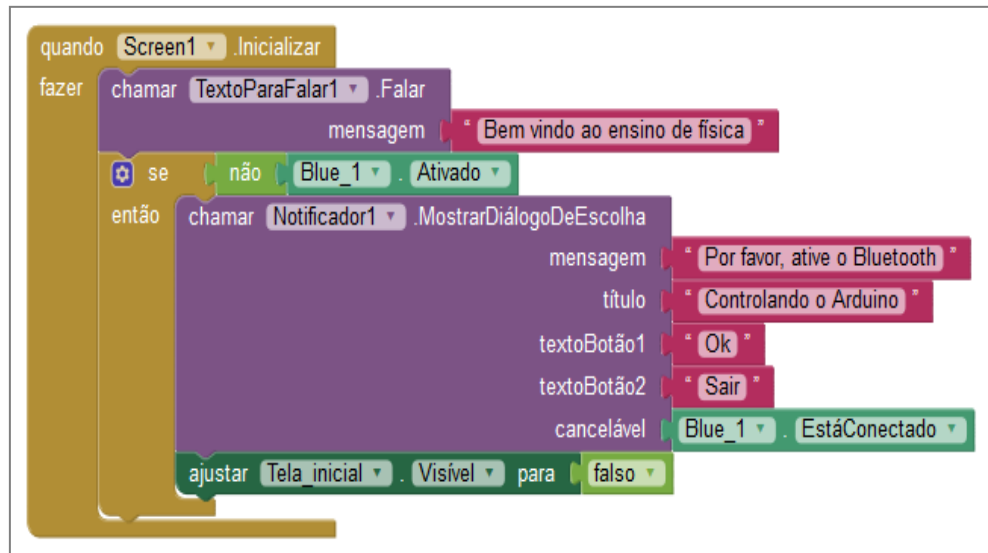
A seguir o passo a passo do desenvolvimento da lógica de programação do aplicativo no ambiente de blocos do AI2. A sequência de ligação dos blocos, referentes aos componentes presentes no ambiente de designer, começa pelo componente de inicialização Screen1, depois pelos elementos de conexão via Bluetooth.

1º Passo – Na inicialização do aplicativo, o smartphone ou tablet emitirá um áudio com a seguinte frase: “Bem vindo ao ensino de Física” e, também, mostrará uma caixa de texto com título “Controlando o Arduino” e com os botões de escolha: “OK” e “Sair”. Se a opção escolhida for “OK” e se o Bluetooth não estiver ligado, a tela de controles (Tela\_inicial) permanecerá oculta e aparecerá a seguinte mensagem de texto com a frase: “Por

favor, ative o Bluetooth”. Na figura 25, temos a lógica de blocos de inicialização do aplicativo.

Observação: As frases escritas nos “Blocos de textos”, blocos vermelhos da figura 25, são editáveis;

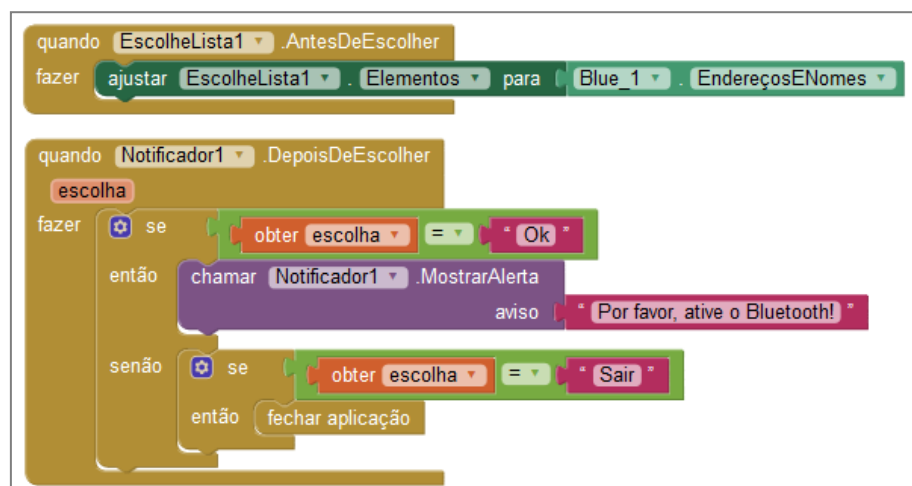
Figura 25 – Lógicas de inicialização do aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

2º Passo – Se o Bluetooth estiver ativado e a escolha for “OK”, uma lista de dispositivos pareados com o smartphone ou tablet será apresentada na tela. Se a escolha for “Sair”, o aplicativo será finalizado. Na figura 26, temos a lógica de escolha de inicialização de aplicação e lista de dispositivos pareados com Bluetooth;

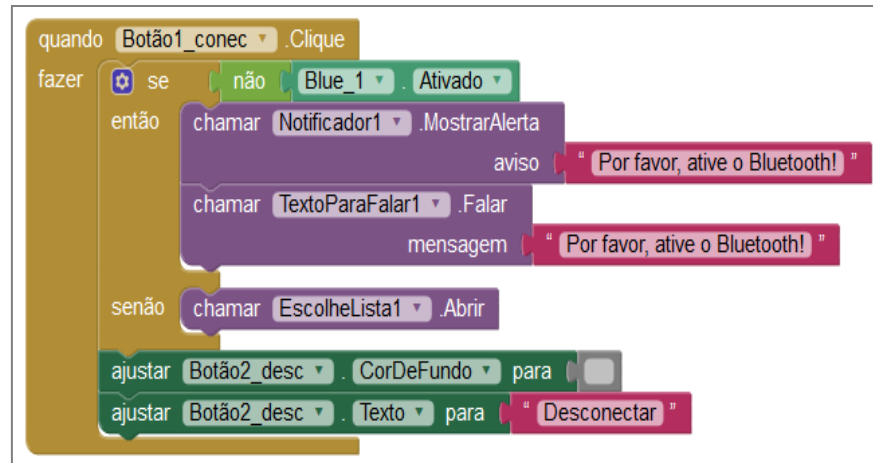
Figura 26 – Lógica de inicialização e lista de dispositivos pareados.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

3º Passo – A figura 27 mostra a lógica do botão “Conectar”. Se o Bluetooth não estiver ativado quando o botão “Conectar” for tocado, o smartphone ou tablet emitirá as mensagens de texto e de áudio: “Por favor, ative o Bluetooth”. Porém, se o Bluetooth estiver ligado, uma lista de dispositivos pareados com o smartphone ou tablet será apresentada na tela. A cor do botão “Desconectar” ficará cinza.

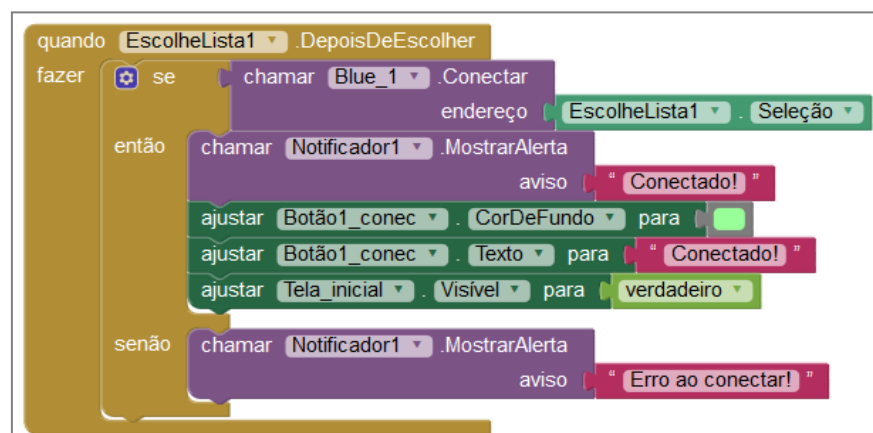
Figura 27 – Lógica de conectividade do botão “Conectar”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

4º Passo – A figura 28 mostra a lógica do botão “Conectar”. Depois que o usuário fizer a escolha do módulo Bluetooth ligado ao Arduino, as configurações do botão “Conectar” mudará: o texto para “Conectado!”, a cor do botão para verde e aparecerá uma mensagem na tela, “Conectado!”, indicando que o smartphone ou tablet foi conectado ao Arduino. O botão “Desconectar” não mudará suas configurações iniciais e a tela de comandos ficará visível para o usuário.

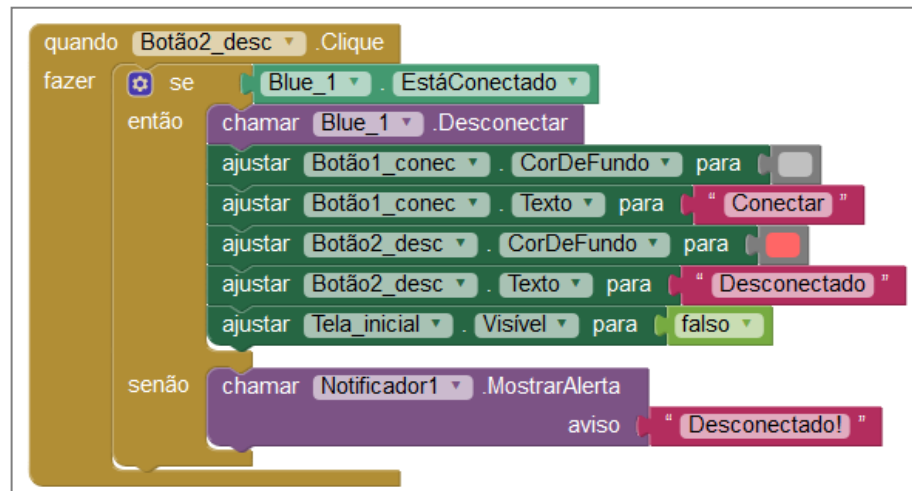
Figura 28 – Lógica de blocos do botão “Conectar”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

Caso o smartphone ou tablet não obtenha sucesso na conexão com o módulo Bluetooth escolhido, a mensagem de texto “Erro ao conectar!” aparecerá na tela (ver figura 28).

Figura 29 – Lógica de blocos do botão “Desconectar”.

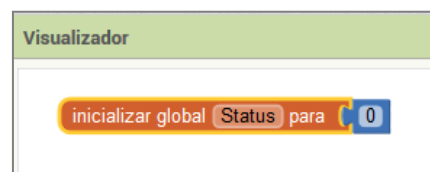


Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

5º Passo – A figura 29 mostra a lógica do botão “Desconectar”. Depois que smartphone ou tablet estiver conectado ao Arduino, se o usuário quiser desconectar basta apertar o botão “Desconectar” e o Arduino será desconectado, a tela de comandos ficará oculta, a cor do botão “Conectar” ficará cinza. O texto do botão “Desconectar” ficará “Desconectado”, a cor mudará para vermelho e aparecerá a seguinte mensagem de texto na tela: “Desconectado”;

6º Passo – A figura 30 mostra os blocos da lógica de inicialização dos botões “Liga/Desliga”. A variável “Status” do bloco “Inicializar global” mudará as configurações do botão “Liga/Desliga” toda vez que o mesmo for tocado;

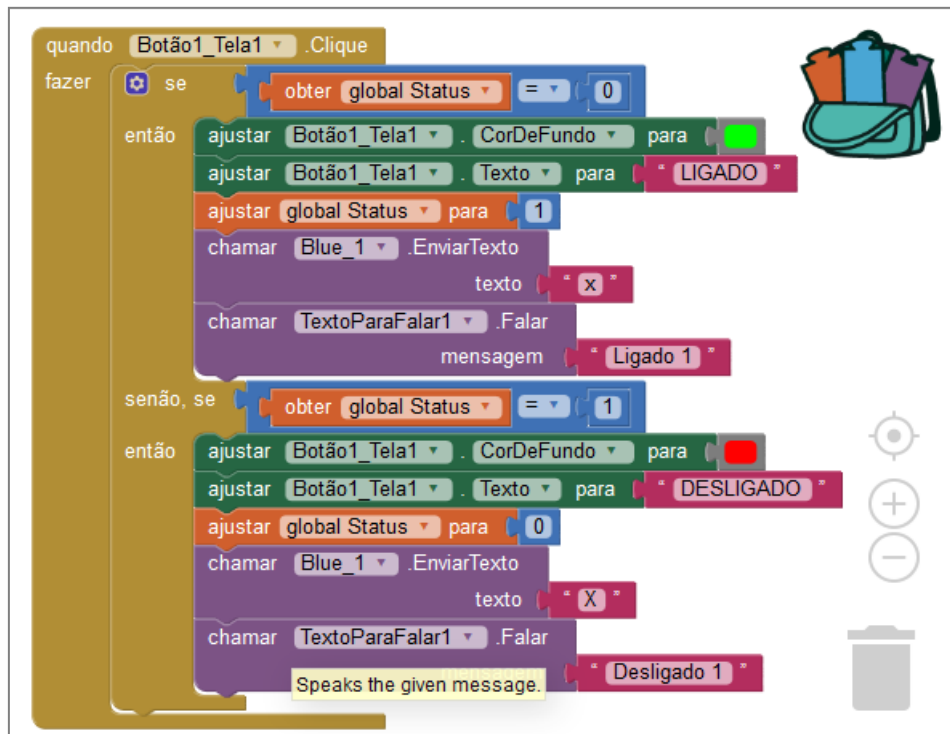
Figura 30 – Lógica de blocos dos botões “Liga/Desliga”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)



Figura 31 – Lógica de blocos do primeiro botão “Liga/Desliga”.

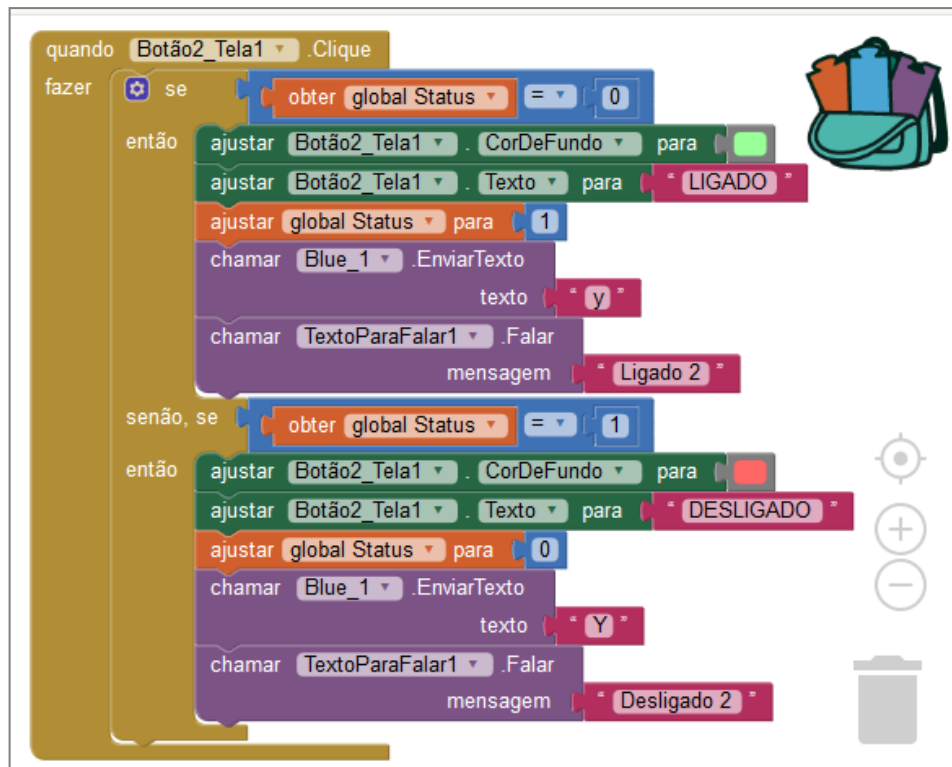


Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

7º Passo – Na figura 31, temos a lógica de blocos do primeiro botão “Liga/Desliga”. Quando o “Botão 1”, na tela de comandos, for tocado: a cor do botão ficará verde; o texto passará para “Ligado”; a variável “Status” mudará para “1”; o smartphone ou tablet enviará via Bluetooth o caractere “x” para o Arduino e emitirá a seguinte mensagem de áudio “Ligado 1”. Se o “Botão 1” for tocado novamente: a cor do botão ficará vermelha; o texto passará para “Desligado”; a variável “Status” mudará para “0”; o smartphone ou tablet enviará via Bluetooth o caractere “X” para o Arduino e emitirá a seguinte mensagem de áudio “Desligado 1”;

8º Passo – Na figura 32, temos a lógica de blocos do segundo botão “Liga/Desliga”. Quando o “Botão 2”, na tela de comandos, for tocado: a cor do botão ficará verde; o texto passará para “Ligado”; A variável “Status” mudará para “1”; o smartphone ou tablet enviará via Bluetooth o caractere “y” para o Arduino e emitirá a seguinte mensagem de áudio “Ligado 2”. Se o “Botão 2” for tocado novamente: a cor do botão ficará vermelha; o texto passará para “Desligado”; a variável “Status” mudará para “0”; o smartphone ou tablet enviará via Bluetooth o caractere “Y” para o Arduino e emitirá a seguinte mensagem de áudio “Desligado 2”;

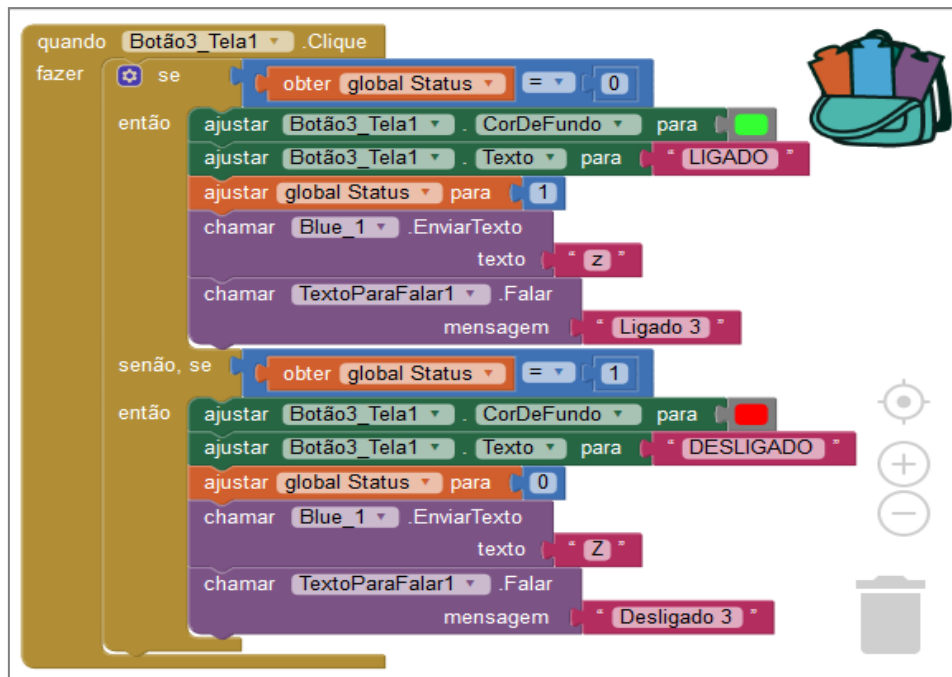
Figura 32 – Lógica de blocos do segundo botão “Liga/Desliga”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

9º Passo – Na figura 33, temos a lógica de blocos do terceiro botão “Liga/Desliga”. Quando o “Botão 3”, na tela de comandos, for tocado: a cor do botão ficará verde; o texto passará para “Ligado”; a variável “Status” mudará para “1”; o smartphone ou tablet enviará via Bluetooth o caractere “z” para o Arduino e emitirá a seguinte mensagem de áudio “Ligado 3”. Se o “Botão 3” for tocado novamente: a cor do botão ficará vermelha; o texto passará para “Desligado”; a variável “Status” mudará para “0”; o smartphone ou tablet enviará via Bluetooth o caractere “Z” para o Arduino e emitirá a seguinte mensagem de áudio “Desligado 3”;

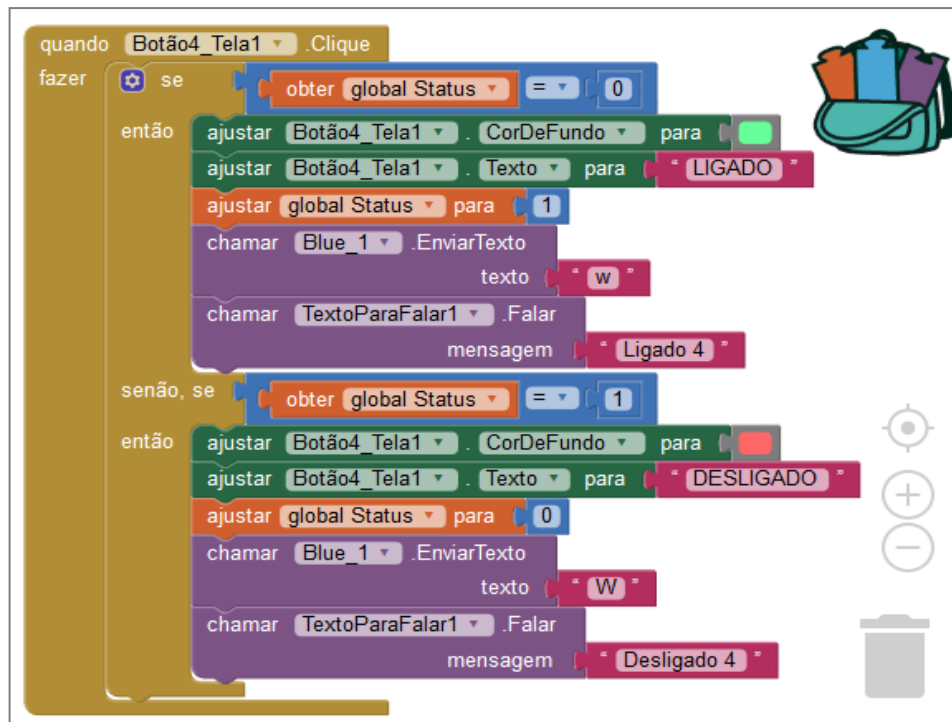
Figura 33 – Lógica de blocos do terceiro botão “Liga/Desliga”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

10º Passo – Na figura 34, temos a lógica de blocos do quarto botão “Liga/Desliga”. Quando o “Botão 4”, na tela de comandos, for tocado: a cor do botão ficará verde; o texto passará para “Ligado”; a variável “Status” mudará para “1”; o smartphone ou tablet enviará via Bluetooth o caractere “w” para o Arduino e emitirá a seguinte mensagem de áudio “Ligado 4”. Se o “Botão 4” for tocado novamente: a cor do botão ficará vermelha; o texto passará para “Desligado”; a variável “Status” mudará para “0”; o smartphone ou tablet enviará via Bluetooth o caractere “W” para o Arduino e emitirá a seguinte mensagem de áudio “Desligado 4”;

Figura 34 – Lógica de blocos do quarto botão “Liga/Desliga”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

11º Passo – A figura 35 mostra os blocos da lógica de inicialização dos botões deslizantes “Slider”. As variáveis dos blocos “Inicializar global” mudará as configurações dos botões deslizantes toda vez que os mesmos forem tocados;

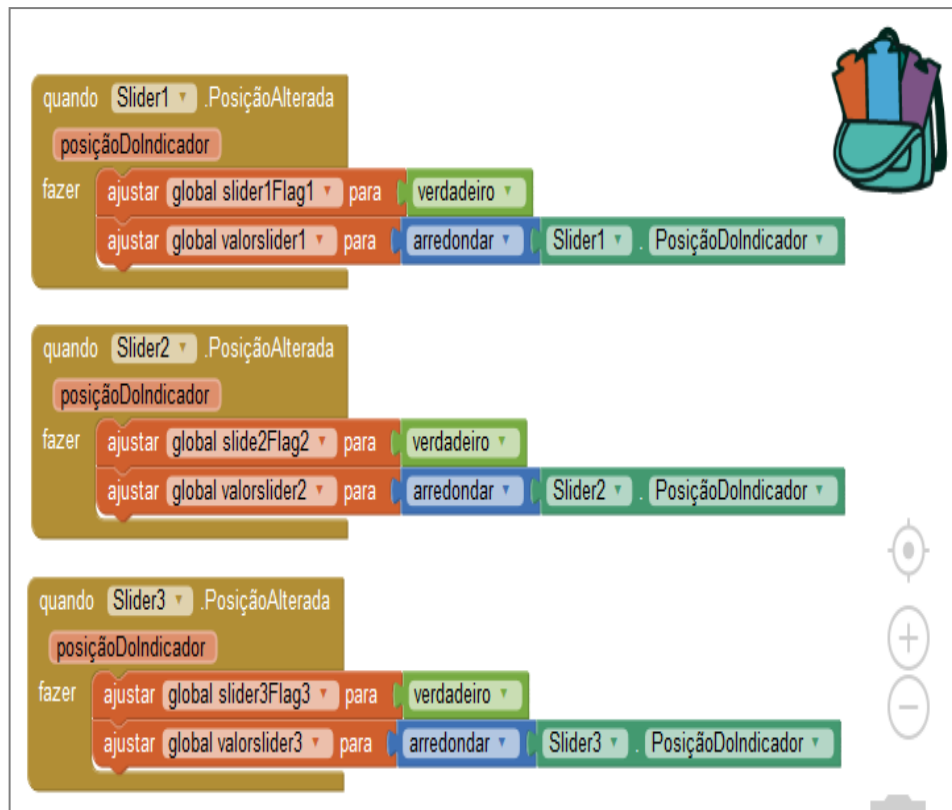
Figura 35 – Lógica de inicialização dos botões deslizantes “Slider”.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

12º Passo – Na figura 36, temos a lógica de blocos que fará o armazenamento dos valores das posições do cursor para cada botão deslizante. Os valores serão armazenados nas variáveis “inicializar global” de cada botão deslizante;

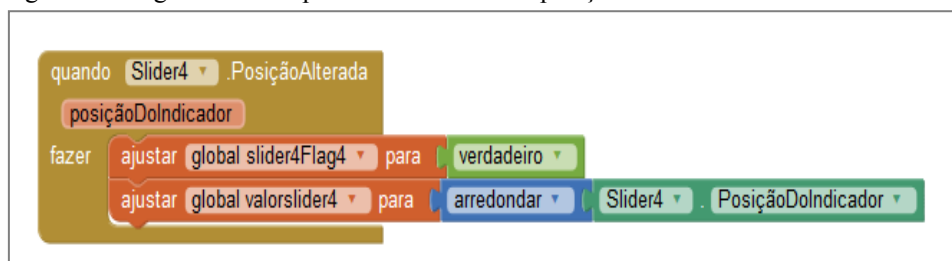
Figura 36 – lógica de blocos para obter os valores das posições dos botões deslizantes.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

13º Passo – Na figura 37, temos a lógica de blocos que fará o armazenamento dos valores das posições do botão deslizante “Slider 4”;

Figura 37 – lógica de bloco para obter o valor das posições do cursor do botão “Slider 4”.



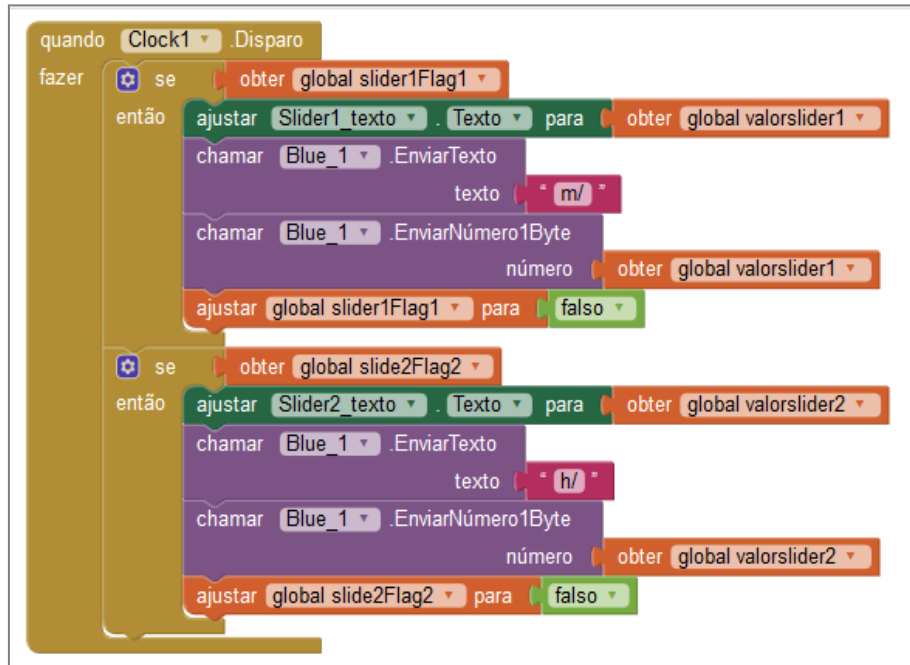
Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

14º Passo – Na figura 38, apresentamos a lógica de blocos que enviará via Bluetooth as informações de cada botão deslizante para o Arduino. Nessa lógica, toda vez que o cursor de um botão deslizante mudar de posição, o componente “Clock1” dispara selecionado os valores armazenados na variável “inicializar global” e enviará para Arduino, um caractere de seleção de um botão deslizante, juntamente com os valores das posições para cada botão “Slider”. Os caracteres de seleção dos botões deslizantes enviados serão: “m” para o botão

deslizante 1, “h” para o botão deslizante 2, “q” para o botão deslizante 3, “n” para o botão deslizante 4.

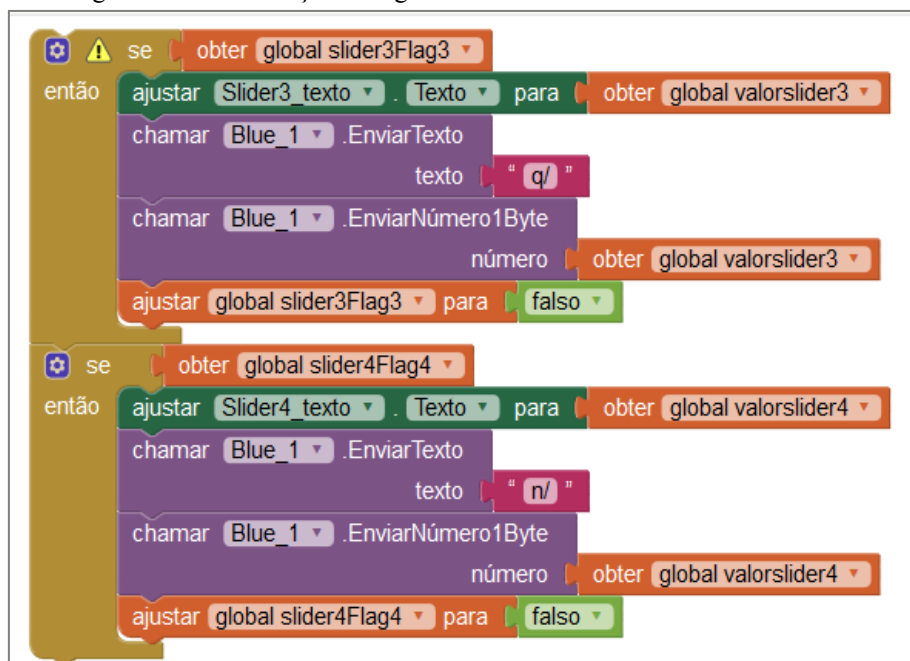
Observação: A função do caractere “/”, que aparece nos blocos de textos das figuras 38 e 39, será para separar a seleção de um botão “Slider”, dos seus respectivos valores de posições na hora do envio por Bluetooth para o Arduino.

Figura 38 – Lógica de blocos do envio de informações dos botões “Slider” para o Arduino.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

Figura 39 – Continuação da lógica de blocos dos botões deslizantes 3 e 4.

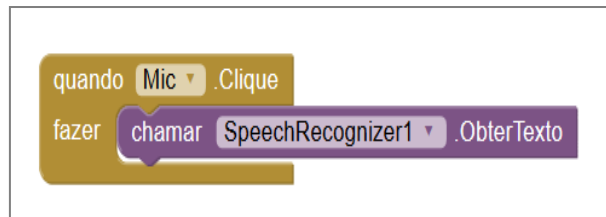


Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

15º Passo – A figura 39, mostra a continuação da lógica de blocos para os botões deslizantes 3 e 4. Essa lógica deverá ser feita logo abaixo do bloco “Se então”, referente ao botão deslizante 2, mostrado na figura 38;

16º Passo – A figura 40 mostra os blocos da lógica do botão de microfone, utilizado no controle do Arduino através do comando por voz. Nessa lógica, quando o botão “Microfone” for tocado, na tela inicial do aplicativo, e se o smartphone ou tablet estiver conectado na internet, o componente “SpeechRecognizer1” gravará a voz do usuário e em seguida enviará para o servidor do Google converte-la em texto. Depois que a voz for convertida em texto, o componente “SpeechRecognizer1” guardará o resultado;

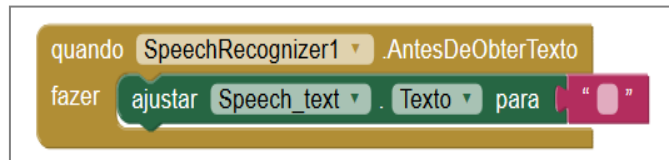
Figura 40 – Lógica do botão de microfone usado no comando por voz.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

17º Passo – Antes do componente “SpeechRecognizer1” obter o texto (resultado), o local (Speech\_text) do lado do nome “Comando por voz”, na tela inicial, será limpo. Na figura 41, temos a lógica de blocos antes do componente “SpeechRecognizer1” obter o resultado (texto) do comando de voz gravado pelo usuário;

Figura 41 – Lógica de blocos para o com componente “SpeechRecognizer1”.

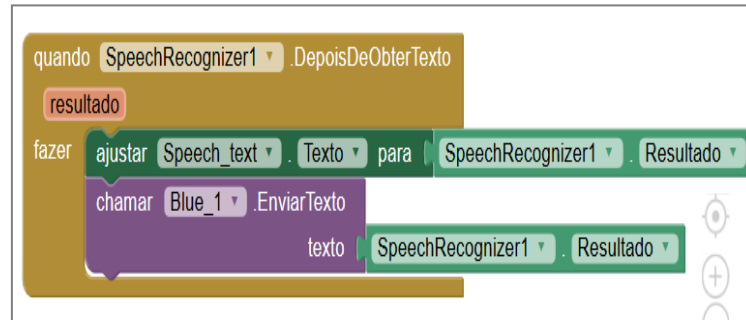


Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

18º Passo – Depois de obter, do servidor Google, o resultado na forma de texto do comando por voz gravado pelo usuário, o componente “SpeechRecognizer1” mostrará o texto (comando por voz) na tela do smartphone ou talet do lado do nome “Comando por Voz” e, também, enviará via Bluetooth para o Arduino. A figura 42 mostra a lógica do componente

“SpeechRecognizer1” depois de obter o resultado (texto) do comando por voz gravado pelo usuário;

Figura 42 – O resultado do comando por voz sendo enviado para o Arduino.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

19º Passo – A figura 43 mostra a lógica de blocos do botão “Ir para a Tela 2”, presente na tela inicial. Se o usuário tocar no botão “Ir para a Tela 2”, o aplicativo mudará da tela inicial para tela de monitoramento (Tela\_2);

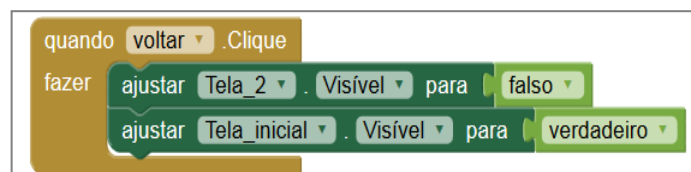
Figura 43 – Lógica de mudança da tela inicial para tela de monitoramento.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

20º Passo – A figura 44 mostra a lógica de blocos do botão “Voltar”. Se o usuário tocar no botão “Ir para a Tela 2”, o aplicativo mudará da tela inicial para a segunda tela;

Figura 44 – Lógica de mudança da tela de monitoramento para tela inicial.



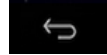
Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

21º Passo – A figura 46 mostra a lógica de blocos quando o botão voltar (figura 45) para tela anterior for pressionado nos dispositivos com sistema operacional Android. Se o aplicativo estiver na tela de monitoramento (tela 2) e o botão voltar, do smartphone ou tablet,



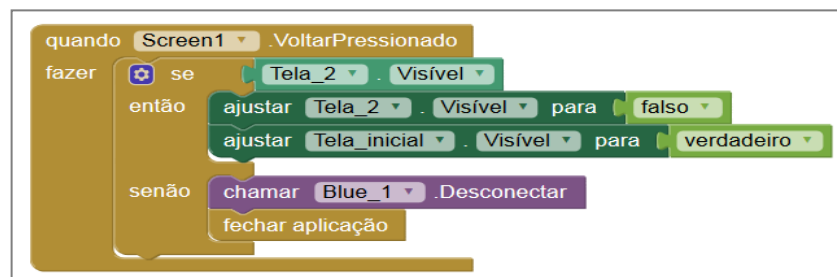
for pressionado aparecerá a tela inicial e a tela 2 ficará oculta. Quando o aplicativo estiver na tela inicial e o botão voltar, do smartphone ou tablet, for pressionado o aplicativo será fechado (fechar aplicação);

Figura 45 – Botão “voltar” dos dispositivos com Andoird.



Fonte: Do Próprio Autor.

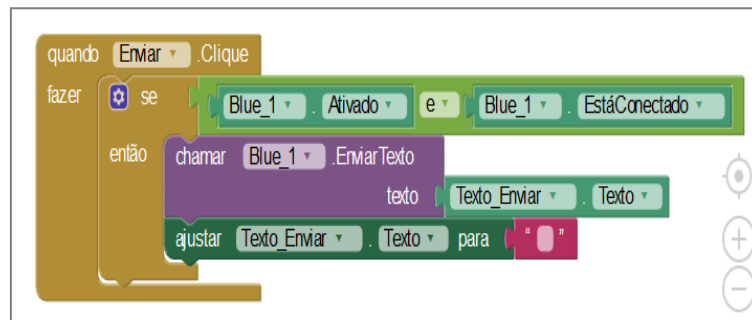
Figura 46 – Lógica de blocos de voltar a tela e encerrar o aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

22º Passo – A figura 47 mostra a lógica de blocos para controlar o Arduino através do envio de texto. Nessa lógica, depois que o usuário tocar no retângulo abaixo do nome “Caixa de textos”, o teclado do smartphone ou tablet aparecerá para receber os comandos. Depois de digitar os comandos, o usuário poderá tocar no botão “Enviar”. Se o Bluetooth estiver ativado e conectado, o texto digitado será enviado para o Arduino e em seguida a caixa de textos será esvaziada;

Figura 47 – Lógica do botão “Enviar” presente na tela 2 do aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

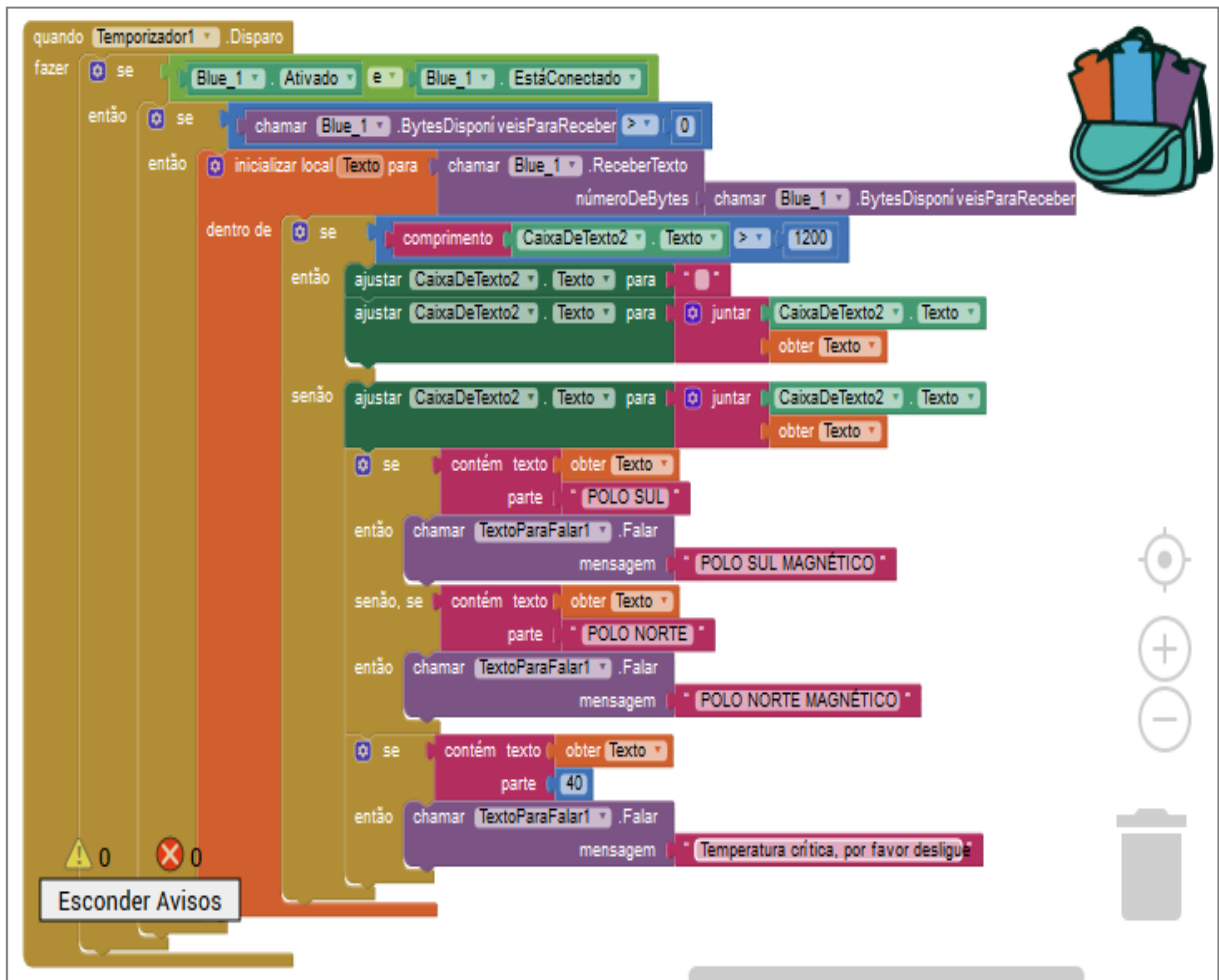
23º Passo – Quando o Bluetooth estiver ativado e conectado e o componente “Blue\_1” receber dados enviados pelo Arduino, o elemento “Temporizador” será executado com pulsos de duração de um segundo. Nesse intervalo de tempo, as informações recebidas do Arduino

serão atualizadas na caixa de textos a cada um segundo. Depois que a quantidade de Bytes recebidos atingir o valor de 1200, a caixa de textos será esvaziada e atualizada novamente a cada um segundo.

No experimento “Campo Magnético”, o Arduino enviará informações contendo as frases “POLO SUL” ou “POLO NORTE”. Se o Arduino enviar um texto contendo a frase “POLO SUL”, o smartphone ou tablet emitirá uma mensagem de voz com a seguinte frase: “Polo sul magnético”. Quando parte do texto recebido pelo aplicativo for “POLO NORTE”, a mensagem de voz emitida pelo smartphone ou tablet será: “Polo norte magnético”.

No experimento “Efeito Joule”, o Arduino enviará informações de temperatura, voltagem e tempo. Quando o valor da temperatura do resistor no experimento atingir o valor de 40°C, o aplicativo receberá os caracteres “40” e emitirá um aviso através de uma mensagem voz com a seguinte frase: “Temperatura crítica, por favor desligue”. Esse aviso será usado para que o aluno desligue a voltagem do resistor no experimento.

Figura 48 – Lógica de blocos do “Serial Monitor Arduino” presente na tela 2 do aplicativo.



Fonte: [http://ai2.appinventor.mit.edu/?locale=pt\\_BR#5344539176599552](http://ai2.appinventor.mit.edu/?locale=pt_BR#5344539176599552)

## 2.3 O Aplicativo Ensino de Física

O aplicativo Ensino de Física foi desenvolvido para ser utilizado, associado ao Arduino, nas atividades demonstrativas e experimentais de eletromagnetismo. Porém o App Ensino de Física pode ser usado em outras áreas da Física e na robótica educacional.

O aplicativo pode ser editado e aperfeiçoado na plataforma MIT App Inventor. Para isso basta seguir o passo a passo do desenvolvimento descrito nesse Produto Educacional.

A seguir, como exemplo, a aparência das telas do aplicativo nas atividades experimentais: “Campo Magnético” e “Efeito Joule”.

1ª Tela – Na figura 49, a tela de inicialização do aplicativo. Além da caixa de texto na tela de inicialização do aplicativo, o smartphone ou tablet emitirá, também, uma mensagem de voz com a seguinte: “Bem vindo ao Ensino de Física”;

Figura 49 – Tela de inicialização do aplicativo.



Fonte: Do próprio autor.

2ª Tela – Na figura 50, a tela do aplicativo antes da conexão com o Arduino;

Figura 50 – Tela do aplicativo antes da conexão via Bluetooth.



Fonte: Do próprio autor.

3ª Tela – A figura 51 mostra a tela com a lista de dispositivos pareados com o Bluetooth do celular, depois que o usuário tocar no botão “Conectar”;

Figura 51 – Tela com a lista de dispositivos pareados com Bluetooth.



Fonte: Do próprio autor.

4ª Tela – A figura 52 mostra a tela inicial (Tela 1) do aplicativo depois que o smartphone ou tablet estiver conectado ao módulo Bluetooth ligado ao Arduino. Os botões “Liga/Desliga” e os botões “Slider”, podem ser utilizados para controlar sensores e atuadores eletrônicos em diversas atividades experimentais de Física;

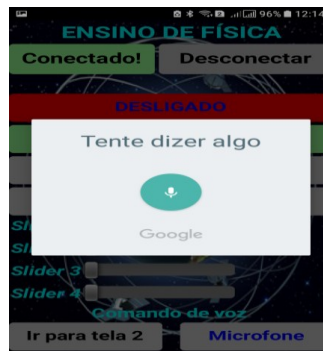
Figura 52 – Tela do aplicativo depois de conectado com o Arduino.



Fonte: Do próprio autor.

5ª Tela – A figura 53 mostra a Tela de comando por voz. Depois que o botão “Microfone” for tocado o usuário poderá gravar um comando que será enviado para o servidor Google que converterá a voz em texto que em seguida aparecerá na tela do aplicativo e enviado para o Arduino. O controle por voz só funcionará se smartphone ou tablet estiver conectado na internet. Com o controle por voz, o usuário liga e desliga os sensores e atuadores eletrônicos;

Figura 53 – Tela referente ao comando por voz.



Fonte: Do próprio autor.

6ª Tela – A figura 54 mostra a (tela 2) tela de controle por texto e monitoramento das atividades experimentais. Nessa tela o usuário verificará os dados obtidos de forma automática com o Arduino;

Figura 54 – Tela de monitoramento e controle por texto.



Fonte: Do próprio autor.

7ª Tela – A figura 55 mostra as informações de uma atividade experimental, utilizando ímãs de diversos formatos e uma bobina solenoide;

Figura 55 – Tela de monitoramento da polaridade e intensidade do campo magnético.



Fonte: Do próprio autor.

8ª Tela – A figura 56 mostra o monitoramento do campo magnético com a inversão da corrente elétrica em uma bobina solenoide em uma atividade de Eletromagnetismo;

Figura 56 – Tela de monitoramento da polaridade e intensidade do campo magnético.



Fonte: Do próprio autor.

9ª Tela – A figura 57 mostra a tela de monitoramento de uma atividade experimental envolvendo a transformação de energia elétrica em energia térmica;

Figura 57 – Tela monitoramento de uma atividade sobre o consumo de energia elétrica.



Fonte: Do próprio autor.

### 3. Construção do Kit Experimental Eletroduino

#### 3.1 A Plataforma Arduino

O Arduino consiste de uma plataforma de prototipagem eletrônica de *software* e *hardware* de código aberto (*open source*) que oferece subsídios para profissionais e amadores

desenvolverem suas criatividade com relação ao uso da tecnologia de forma mais fácil e flexível. Nesse âmbito, o Arduino é uma ferramenta muito utilizada por designers, engenheiros, estudantes, artistas, desenvolvedores e entre outros em todo o mundo. A plataforma Arduino é bastante utilizada na inovação de músicas, jogos, brinquedos, automação residencial, agricultura, veículos autônomos, na área da robótica, na educação e muito mais. O Arduino surgiu da necessidade de implementação de um projeto de pesquisa de Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis no *Interaction Design Institute* (Instituto de Design de Interação) da cidade de Ivrea no início do ano 2000. A plataforma Arduino, inicialmente, foi criada para ajudar os estudantes sem experiências em eletrônica e programação (ARDUINO, 2017). Nesse âmbito, conforme Blum (2016).

O Arduino é uma plataforma de desenvolvimento associada com uma linguagem de programação intuitiva que você desenvolve utilizando o ambiente de desenvolvimento integrado (IDE) Arduino. Ao equipar o Arduino com sensores, atuadores, luzes, alto-falantes, módulos adicionais (chamados de Shields), e outros circuitos integrados, você pode transformar o Arduino em um “cérebro” programável para qualquer sistema de controle. (BLUM, 2016, p. xxiv)

O *hardware* do Arduino é representado por uma placa eletrônica que utiliza o microcontrolador programável Atmega AVR da empresa Atmel. Os microcontroladores Atmega AVR presentes nas placas Arduino, apresentam entradas e saídas (digitais e analógicas) que interagem com os sensores e atuadores eletrônicos externos. A placa Arduino pode trabalhar de forma independente ou associada a um computador, que pode ser um *smartphone* ou *tablet*.

Para Oliveira et al. (2015), a placa Arduino é muito similar à de um computador de pequeno porte, sendo composto por um microcontrolador, memória RAM, armazenamento secundário (memória flash) e clock, entre outras funcionalidades que podem ser exploradas com a plataforma Arduino. E mais, ainda, conforme Oliveira et al.:

Mesmo tendo similaridade com computadores convencionais, a principal diferença do Arduino está na maneira como utilizamos as entradas e saídas. Por exemplo, em um computador pessoal, emprega-se teclado e mouse para entradas e monitores e impressoras para saídas. Já no Arduino, as entradas e saídas estão diretamente ligadas a componentes eletrônicos. Por isso, uma entrada do Arduino pode estar lendo o valor de um sensor de temperatura, interpretando-o e dando como resposta, por exemplo, o acendimento de um LED. (OLIVERIA et al., 2015, p. 18).

A placa mais utilizada mundialmente por desenvolvedores iniciantes é a Arduino UNO (figura 58). O Arduino UNO utiliza o microcontrolador Atmega328 e apresenta 6 pinos

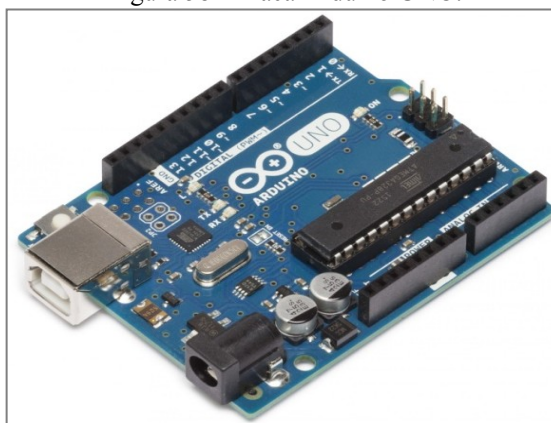
com entradas analógicas e 14 pinos que podem ser usados como entradas ou saídas digitais. Os pinos digitais trabalham com os valores de voltagens (0v ou 5v) e os pinos analógicos com os valores entre 0v e 5v. Na placa Arduino UNO, os pinos digitais (3, 5, 6, 9, 10 e 11) podem gerar valores inteiros entre 0 e 255, usando a técnica PWM (do inglês, Pulse Width Modulation ou Modulação de Largura de Pulso ).

A técnica PWM permite, a partir de um sinal analógico, produzir um conjunto de sinais digitais enviados durante um determinado período. Quando um sinal digital é plotado, em um gráfico, ao longo do tempo, será obtida uma forma de onda quadrada que alterna seu estado entre os níveis lógico *HIGH* (1) e *LOW* (0). (OLIVEIRA et al., 2015, P. 59).

Segundo BLUM (2016) a técnica PWM funciona controlando o tempo em que a onda quadrada, de frequência fixa, permanece em nível lógico alto (*HIGH*). Esse tempo é chamado de *ciclo de trabalho* (*Duty Cycle*). A frequência da onda quadrada gerada, no caso da placa Arduino, é de cerca de 490Hz. Isto é, o sinal varia entre alto (5v) e baixo (0v) cerca de 490 vezes por segundo. Nesse caso, quando o *Duty Cycle* for 0%, o valor médio da voltagem de saída do pino PWM do Arduino será 0v. Para o valor de *Duty Cycle* 50%, a voltagem de saída será 2,5v e para um valor de *Duty Cycle* 100% a saída do pino PWM será 5v.

Com a utilização dos pinos PWM do Arduino é possível controlar, por exemplo, a velocidade de um motor de corrente contínua, a luminosidade de um LED e até mesmo fazer o controle da intensidade do campo magnético em uma bobina solenoide.

Figura 58 – Placa Arduino UNO.

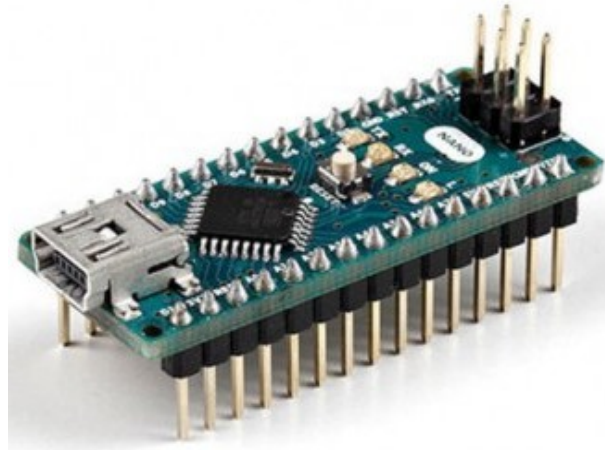


Fonte: <https://store.arduino.cc/usa/arduino-uno-rev3>.

O Kit experimental Eletroduino, descrito nesse material, foi desenvolvido com a placa Arduino Nano porque ela pode ser conectada diretamente em um *protoboard*, facilitando na hora da montagem do aparato experimental. Na figura 59, temos a placa Arduino Nano, utilizada na construção dos kits experimentais didáticos.



Figura 59 – Placa Arduino Nano.



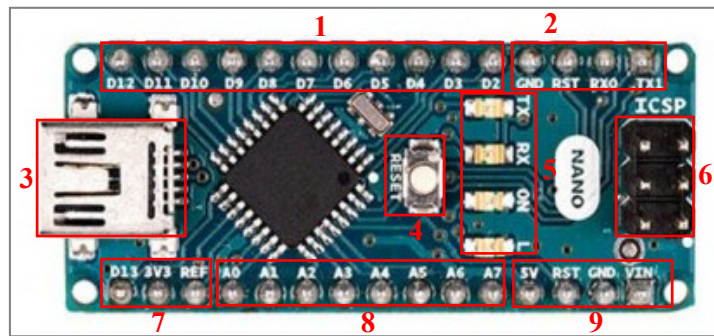
Fonte: <https://store.arduino.cc/usa/arduino-nano>.

Segundo ARDUINO (2017), o Arduino Nano é uma pequena placa eletrônica, muito versátil, baseada no microcontrolador Atmega328P e apresenta uma entrada USB Mini-B para comunicação com o *software*. As características técnicas da placa Arduino Nano são:

- Microcontrolador: Atmega328.
- Arquitetura: AVR.
- Tensão de operação: 5v.
- Tensão de entrada: 7v – 12v
- Memória Flash: 32KB, dos quais 2KB são usados pelo *bootloader*.
- Memória SRAM: 2KB.
- Memória EEPROM: 1KB.
- Frequência de *clock*: 16MHz.
- Hardware de comunicação: ICSP.
- Pinos com entrada analógica: 8.
- Pinos digitais de entrada ou saída (E/S): 22.(6 podem ter sinais PWM).
- Corrente DC máxima por E/S: 40mA.
- Consumo de energia: 19mA.
- Tamanho da placa PCB: 18mm x 45mm.
- Peso: 7g.

As características físicas da placa Arduino Nano (figura 60) estão apresentadas a seguir.

Figura 60 – Placa Arduino Nano.

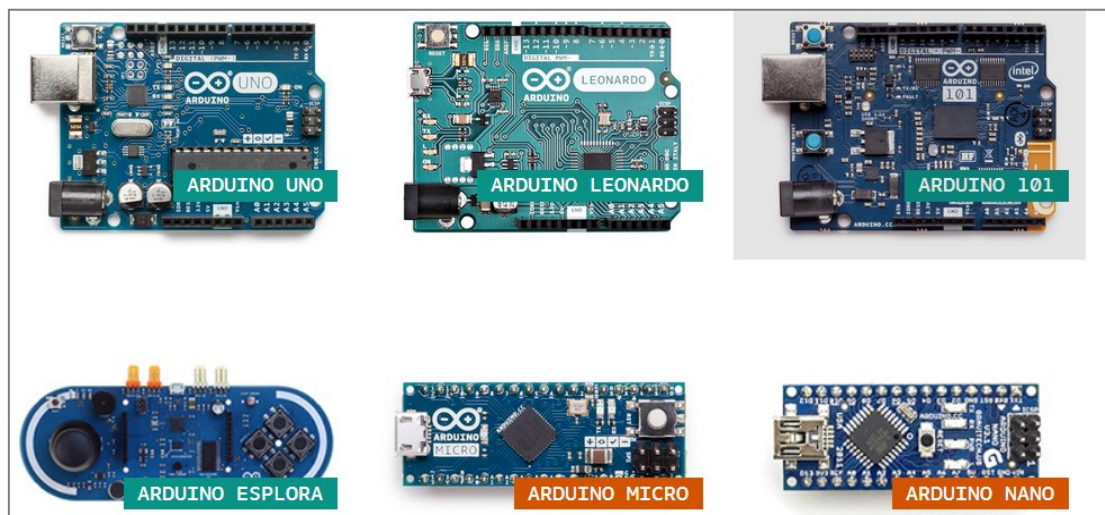


Fonte: Do próprio autor.

- 1 – Pinos digitais de E/S. (Os pinos: 3, 5, 6, 9, 10 e 11, geram sinais PWM).
- 2 – Pinos: (GND); (RST) reset externo; comunicação serial RX (0) e TX (1).
- 3 – Porta USB Mini-B.
- 4 – Botão reset da placa.
- 5 – LED's de sinalização: (RX, TX); On (ligado) e L (LED interno do pino 13).
- 6 – Porta de comunicação ICSP.
- 7 – Pino digital 13; Pino de saída de 3,3v e pino referência.
- 8 – Pinos com entrada analógica.
- 9 – Pino de saída 5v; pino reset; pino GND e pino de entrada (7V<sub>DC</sub> – 12V<sub>DC</sub>).

Na figura 61 apresentamos alguns modelos de placas Arduino disponíveis no mercado de prototipagem eletrônica.

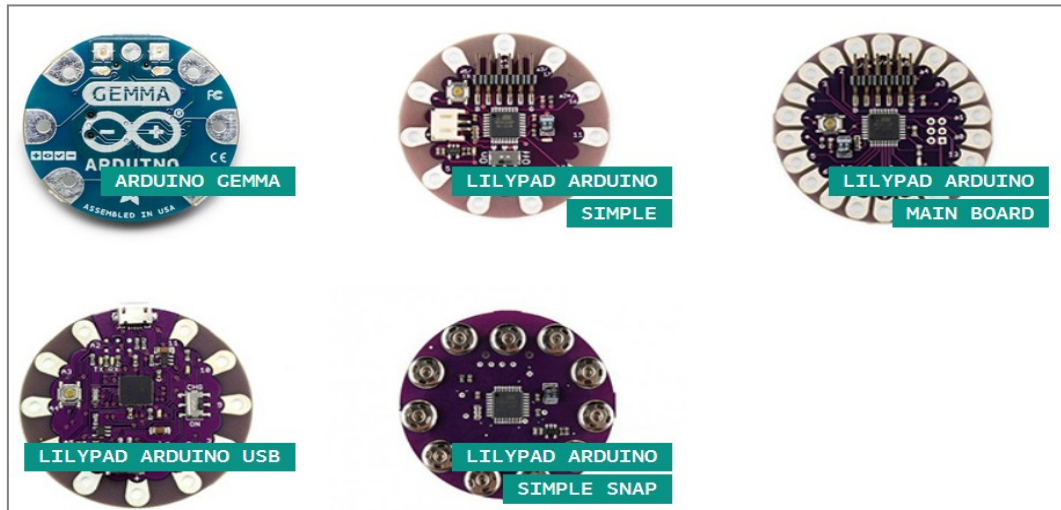
Figura 61 – Placas Arduino.



Fonte: <https://www.arduino.cc/en/Main/Products>.

Na figura 62, alguns modelos de placas utilizadas em projetos nos quais é possível costurá-las em tecidos de trajeis.

Figura 62 – Placas que podem ser costuradas em tecidos.



Fonte: <https://www.arduino.cc/en/Main/Products>.

O Arduino pode ser programado através de um *software* que representa o ambiente integrado de programação. O software está disponível gratuitamente para download na página oficial do Arduino: <https://www.arduino.cc/en/main/software>. O software Arduino está disponível para vários sistemas operacionais para computadores. (Figura 63).

O *software*, instalado em um computador, para programar a placa Arduino é comumente chamado de IDE (do inglês, Integrated Development Environment). O IDE Arduino é baseado na linguagem programação *Processing*<sup>2</sup> que torna possível escrever o código (sketch) em uma linguagem de programação que se parece com linguagem C/C++. O código escrito será processado e depois transferido do computador para o microcontrolador Atmega, na placa Arduino, através da comunicação serial USB. Por fim e não menos importante, temos outro *software* chamado de *bootloader*, que roda no microcontrolador Atmega AVR. Esse *software* grava o sketch enviado pela IDE Arduino na memória da placa Arduino.

<sup>2</sup> Processing: É uma linguagem de programação de código aberto que apresenta um ambiente de desenvolvimento que facilita a criação de projetos para iniciantes dentro das artes visuais e na alfabetização visual dentro da tecnologia. Há dezenas de milhares de estudantes, artistas, designers, pesquisadores e amadores que usam o Processing para aprender e criar protótipos (PROCESSING, 2017).

Figura 63 – Página para download do IDE Arduino.



Fonte: <https://www.arduino.cc/en/main/software>.

### 3.1.1 Instalação e Utilização do IDE Arduino

Depois de fazer o download do arquivo em “ZIP” do IDE Arduino, referente ao sistema operacional do seu computador, por exemplo, (arduino-nightly-windows), conecte a placa Arduino Nano ao computador através de um cabo USB Mini-B. O sistema operacional automaticamente tentará fazer o reconhecimento da placa Arduino e falhará por falta dos drives necessários. Feche a janela de instalação automática e faça a descompactação do arquivo em “ZIP”. Na sequência, dentro da pasta (arduino-nightly-windows), através de um duplo clique no ícone Arduino, você poderá utilizar o IDE Arduino. Um atalho do ícone Arduino poderá ser criado na área de trabalho do computador para facilitar a acessibilidade ao IDE Arduino. Em seguida, no computador com Windows, abra o menu iniciar e acesse as propriedades “Gerenciador de Dispositivos” e observe na seção (portas COM e LPT) em qual porta o placa Arduino foi conectada. Depois que a porta for identificada, clique com o botão direito do mouse e escolha a opção “Atualizar driver”. Para atualizar os drivers, utilize a opção “Procurar software no computador” e selecione o controlador a partir da pasta de drivers do IDE Arduino que foi baixado, espere o Windows concluir a instalação.

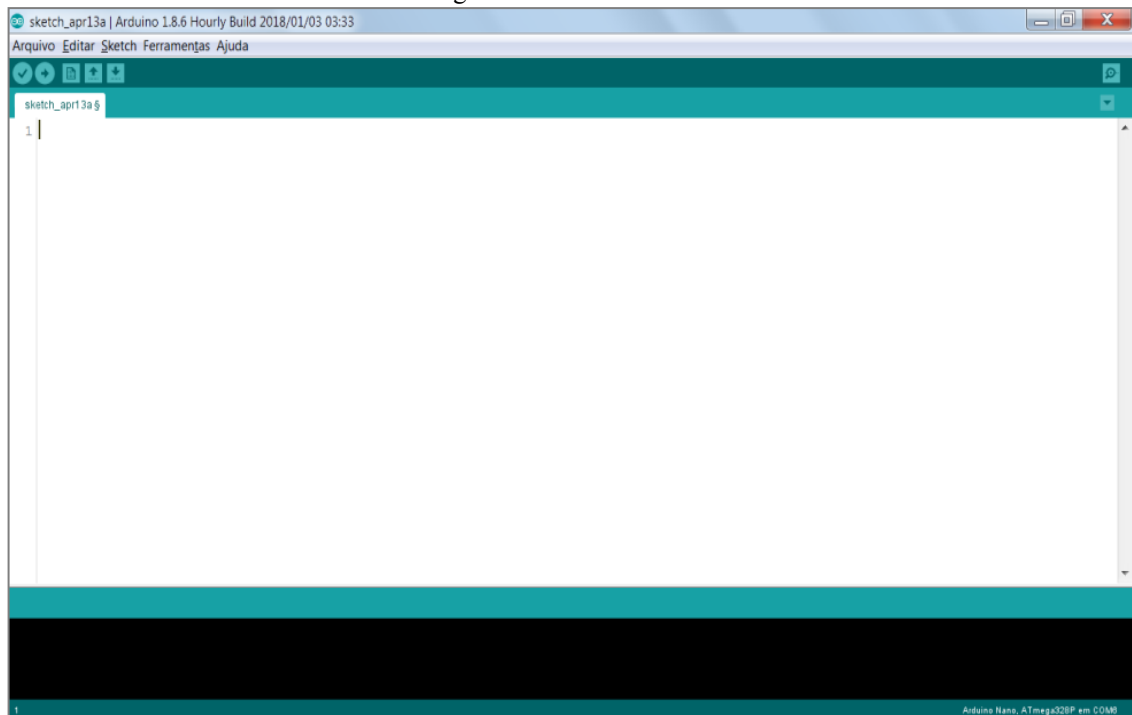
Outra opção é baixar um instalador do IDE Arduino (figura 63) equivalente com o sistema operacional do seu computador e seguir todos os passos da instalação.

A placa Arduino Nano possui um chip FTDI que faz comunicação com o computador. Nesse caso será preciso fazer a instalação dos drives de reconhecimento da placa pelo o sistema operacional. Conforme Oliveira et al.(2015),

Quando utilizar as placas Arduino Uno e Mega, não é necessário baixar e instalar nenhum drive adicional. Porém, caso você esteja utilizando alguma placa com chip FTDI, como, por exemplo, os modelos Duemilanove, Diecimila e Nano, torna-se necessário realizar o download de um instalador para o grupo de drives que está disponível no endereço <http://www.ftdichip.com/Drivers/VCP.htm>. Após realizar o download do instalador, dê um duplo clique no arquivo e siga as instruções do instalador. Ao final do processo, é necessário reiniciar o computador. (OLIVERIA et al., 2015, p. 22).

Na figura 64 podemos observar a janela de trabalho do aplicativo IDE Arduino depois de um duplo clique no ícone Arduino.

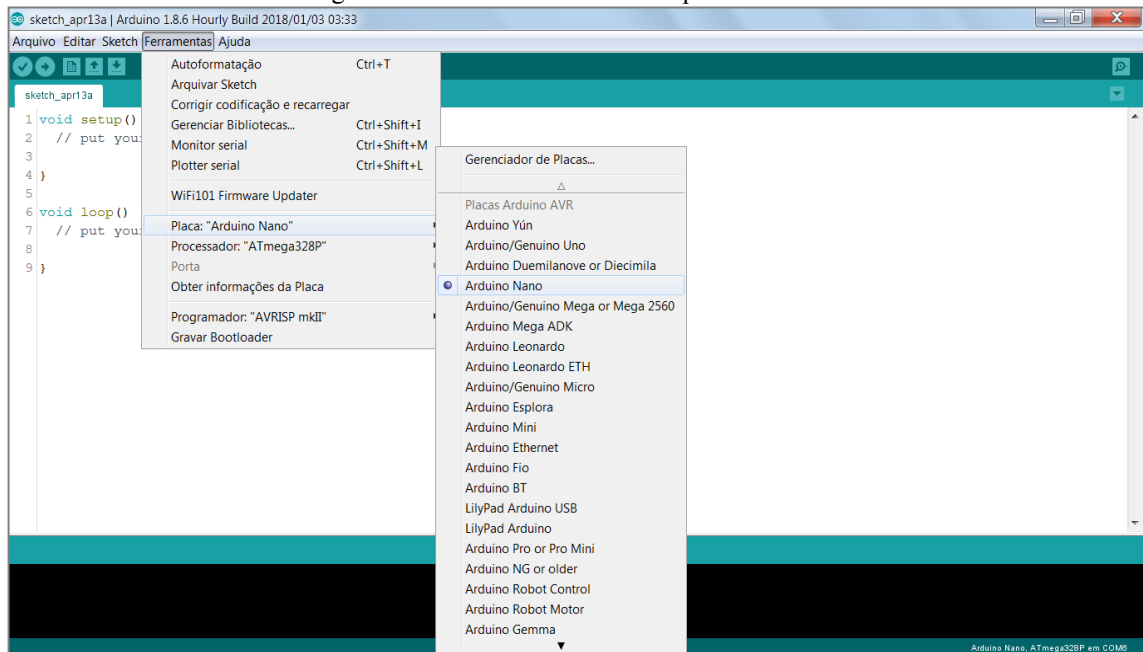
Figura 64 – IDE Arduino.



Fonte: Do próprio autor. (Feito no IDE Arduino).

Na sequência, no ambiente de desenvolvimento do Arduino, acesse o menu “Ferramentas” e depois em “Placas”, escolha a opção “Arduino Nano”, conforme mostrado na figura 65.

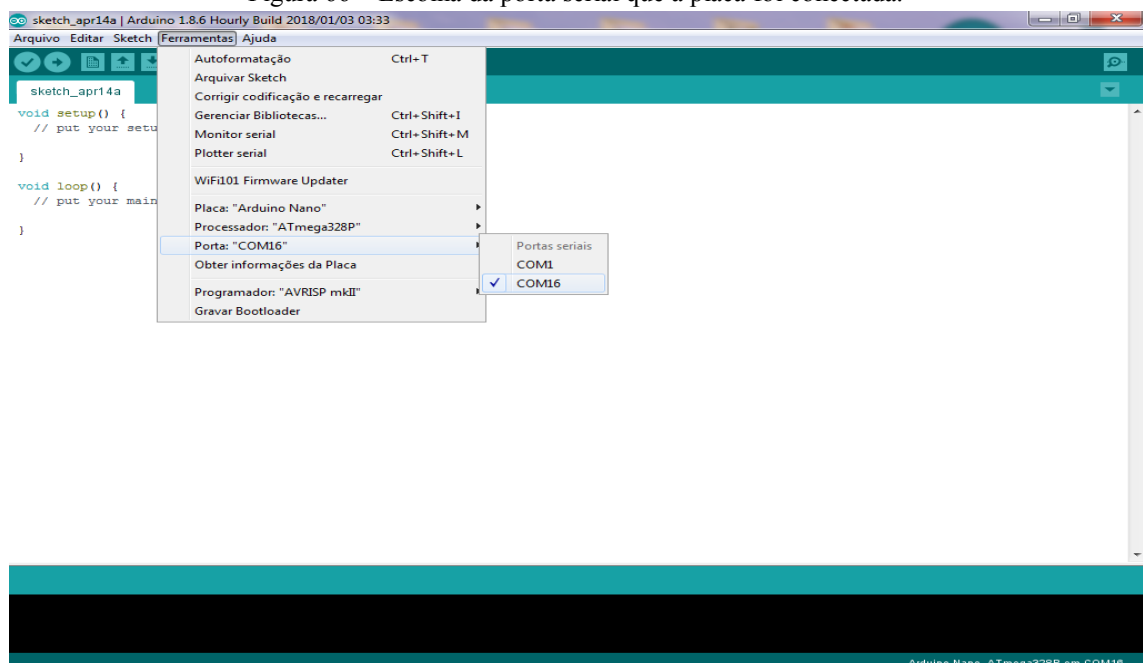
Figura 65 – Escolha do modelo de placa Arduino.



Fonte: Do próprio autor. (Feito no IDE Arduino).

Em seguida, será necessário escolher a porta “COM” que o sistema operacional reconheceu para a placa Arduino Nano, depois que os drives da mesma foram instalados. Para isso, será preciso entrar novamente no menu “Ferramentas” e depois em “Porta” para selecionar a porta “COM” que a placa foi conectada. Na figura 66 temos, em destaque, a porta que a placa foi conectada.

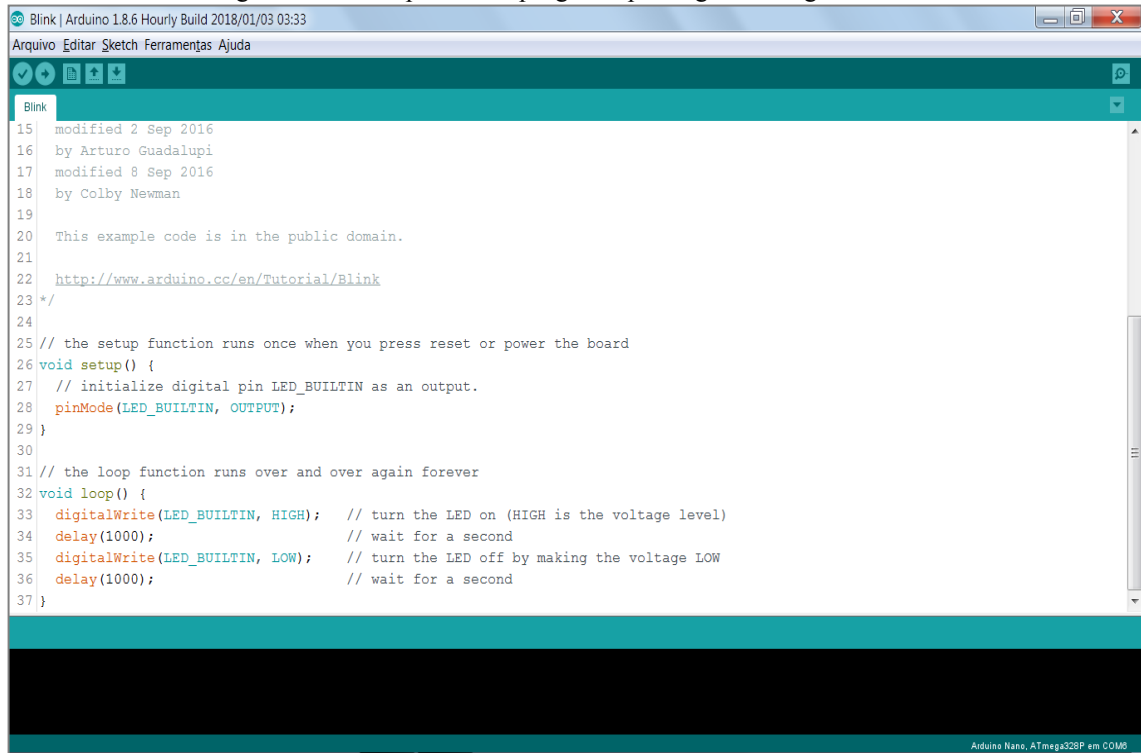
Figura 66 – Escolha da porta serial que a placa foi conectada.



Fonte: Do próprio autor (Feito no IDE Arduino).

A próxima etapa será escrever um programa (*Sketch*) no ambiente de desenvolvimento do Arduino. Na figura 67 temos um exemplo do próprio IDE Arduino, usado para acender e apagar um LED, presente nas placas Arduino, ligado ao pino digital13.

Figura 67 – Exemplo de um programa para ligar e desliga um LED.



Fonte: Do próprio autor. (Feito no IDE Arduino).

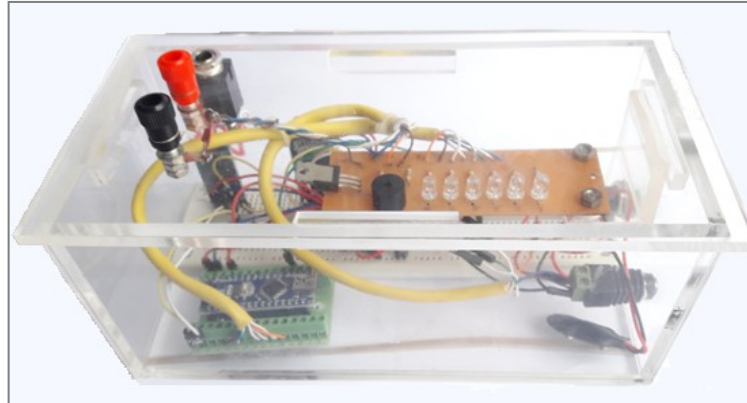
Para testar o programa de controle que será usado nas aulas experimentais de Eletromagnetismo, basta copiar o sketch (presente no apêndice A) e colá-lo dentro da área, mostrada na figura 64, de desenvolvimento de programação do IDE Arduino.

### 3.2 Materiais e Montagem

Os materiais utilizados para confeccionar os kits (Eletroduino) são encontrados com facilidade em lojas de componentes eletrônicos e na internet. Para montar o kit experimental Eletroduino, sugerimos que os componentes sejam montados em uma caixa de acrílico de 3mm de espessura (com tampa), apresentando as seguintes medidas: 20cm de comprimento, 10cm de largura e 9cm de altura. Conforme ilustrado, em um dos kits Eletroduino, na figura 68. Para fazer a caixa, existem empresas especializadas na fabricação de produtos feitos de acrílico.



Figura 68 – Kit experimental Eletroduino (campo magnético).



Fonte: Do próprio autor.

Para fazer a montagem dos kits experimentais serão necessárias as seguintes ferramentas:

- Multímetro digital.
- Ferro de soldar
- Alicates: universal, de corte e de bico.
- Chaves de fendas e/ou Philips.
- Furadeira.
- Pistola de cola quente.

### 3.2.1 Materiais para o Kit - Campo magnético

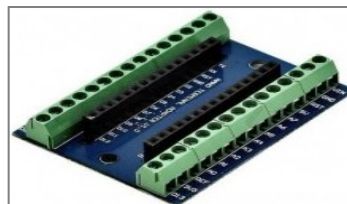
- 02 Arduino Nano.
- 01 Soquete adaptador para Arduino Nano.
- 02 Módulos Bluetooth HC 05 ou similar.
- 07 Resistores de  $1K\Omega$  (código de cores: marrom, preto, vermelho e dourado).
- 02 Resistores de  $10K\Omega$  (marrom, preto, laranja e dourado).
- 01 Resistores de  $220K\Omega$  (vermelho, vermelho, amarelo e dourado).
- 04 Resistores de  $220\Omega$  (vermelho, vermelho, marrom e dourado).
- 01 Transistor mosfet IRF640.
- 01 Sensor de campo magnético 49E.
- 01 Buzzy passivo de 5v.
- 02 LED's alto brilho de cor laranja (tamanho 5mm).



- 02 LED's alto brilho de cor verde (tamanho 5mm).
- 02 LED's alto brilho de cor vermelha (tamanho 5mm).
- 01 Chave liga/desliga.
- 01 Conector fêmea para plugue de fonte de alimentação de 12v.
- 20 Fios condutores (Jumper) do tipo macho-macho para Arduino.
- 01 Plugue fêmea tamanho P10 estéreo para fixar na caixa de acrílico.
- 02 Plugue macho (tamanho P10, estéreo).
- 01 Borne de cor vermelha.
- 01 Borne de cor preta.
- 01 Pedaco de cabo de rede (tamanho 50cm).
- 01 Placa perfurada de circuito impresso (tamanho 10cm x 4cm).
- 01 Pedaco de fio flexível (microfone) de três vias – tamanho 50cm.
- 01 *Protoboard* de 830 pontos.
- 01 Caixa de acrílico com tampa (comprimento: 20cm, largura: 10cm, altura: 9cm).
- 01 Caixa de acrílico com tampa (comprimento: 10cm, largura: 10cm, altura: 3cm).
- Limalha de ferro.
- Ímãs de diversos formatos.
- 01 bússola.
- 01 Bobina solenoide tirada de um relé de 12v.
- 01 Diodo comum - 1N004.

Na figura 69 temos um adaptador para a placa Arduino Nano (placa de leitura de sensor). Esse adaptador será necessário para fixar os fios da placa (BarGraf\_bip) do kit experimental com campo magnético.

Figura 69 – Adaptador para Arduino Nano.



Fonte: Do próprio autor.

A limalha de ferro será utilizada dentro da caixa fechada de acrílico com as seguintes dimensões: comprimento: 10cm, largura: 10cm, altura: 3cm. Essa caixa será usada pelos

alunos para visualizar as configurações das linhas de campo magnético, antes de utilizar o Eletroduino para identificar o campo magnético em ímãs de diversos formatos, mostrados na figura 70.

Figura 70 – Ímãs de diversos formatos.



Fonte: Do próprio autor.

Aplaca “BarGraf\_bip” (figura 71) será usada para mostrar a intensidade do campo magnético (efeito visual) através de um gráfico de LEDs e, também, por meio da intensidade sonora (bipes).

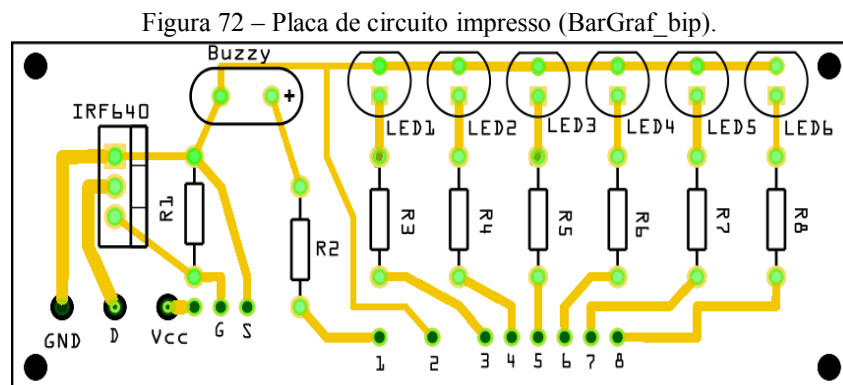
Figura 71 – Placa “BarGraf\_bip” usada no kit experimental.



Fonte: Do próprio autor.

A figura 72 mostra o layout do circuito impresso (placa PCB) usado para montar a placa “BarGraf\_bip” do kit experimental. O *layout* foi desenvolvido no programa Fritzing (disponível para download em: <http://fritzing.org/download/>). As entradas: (1 até 8) e “S”, da placa “BarGraf\_bip”, deverão ser conectadas ao adaptador do Arduino Nano (Arduino de leitura do sensor), através de um pedaço de cabo de rede (cabo flexível) para facilitar a abertura e fechamento da tampa da caixa de acrílico, quando for necessário editar o código nas placas Arduino. As entradas da placa “BarGraf\_bip” (8, 7, 6, 5, 4 e 3) deverão ser ligadas, respectivamente, aos pinos digitais (D2, D3, D4, D5, D6 e D7) do Arduino Nano (leitura de sensor) no adaptador, para acendimento dos LEDs do verde ao vermelho.

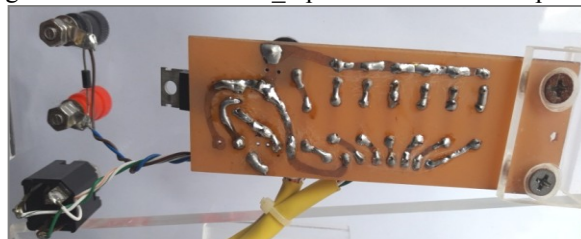
A entrada “2” presente na placa “BarGraf\_bip”, deverá ser ligada ao pino GND do adaptador Arduino Nano (Arduino de leitura ), a entrada “S” precisará ser ligada ao barramento negativo do *proto board*, a entrada “1” (pino positivo do *Buzzy* passivo de 5v) deverá ser ligada ao pino “D8” do Adaptador (Arduino de leitura), a entrada “GND” ligada a saída negativa (GND) da fonte de tensão de 12v e a entrada “G” deverá ser ligada a um resistor de 1K $\Omega$  conectado ao pino “D3” da placa Arduino Nano (placa de controle) para controlar a saída do Eletroduino.



Fonte: Do próprio autor (desenvolvida no Fritzing).

A figura 73 mostra a placa “BarGraf\_bip” feita em circuito impresso e montada na caixa de acrílico.

Figura 73 - Placa “BarGraf\_bip” feita em circuito impresso.



Fonte: Do próprio autor.

A entrada “VCC”, da placa “BarGraf\_bip”, será usada para fazer uma ponte entre a saída positiva (12v) da chave “Liga / Desliga” (figura 74) e saída positiva do Eletroduino (borne vermelho).

Figura 74 – Chave “Liga / Desliga” usada no Eletroduino.



Fonte: Do próprio autor.

Na figura 75 temos os bornes de saída do Eletroduino. O borne vermelho utilizado para o polo positivo e borne preto para o polo negativo.

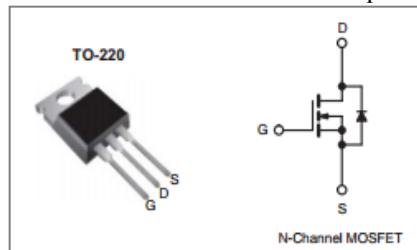
Figura 75 – Os bornes de saída usados no Eletroduino.



Fonte: Do próprio autor.

A saída “D” da placa “BarGraf\_bip” corresponde ao pino D (dreno) do transistor de efeito de campo IRF640 (figura 76). Essa saída deverá ser ligada diretamente na saída negativa do Eletroduino (borne preto).

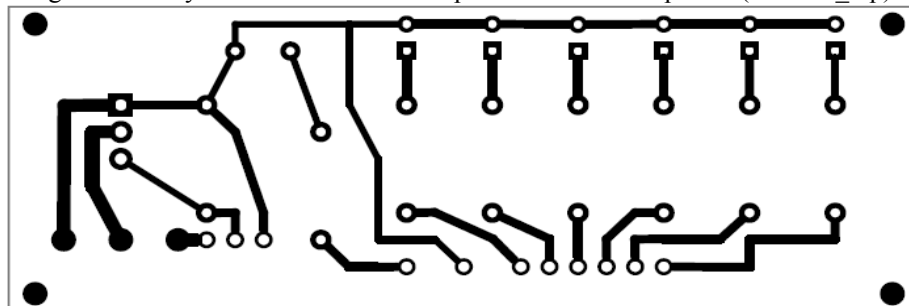
Figura 76 – Transistor de efeito de campo IRF640.



Fonte: <http://pdf1.alldatasheet.com/datasheet-pdf/view/250779/VISHAY/IRF640.html>

Na figura 77 temos um *layout* da placa PCB, em tamanho real, que poderá ser utilizado para confeccionar a placa de circuito impresso “BarGraf\_bip” do kit experimental.

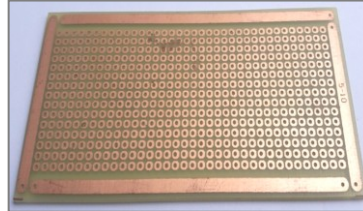
Figura 77 – Layout em tamanho real da placa de circuito impresso (BarGraf\_bip).



Fonte: Do próprio autor (desenvolvida no Fritzing).

Outra opção para montagem da placa “BarGraf\_bip”, para os que não possuem um kit de circuito impressos ou necessitam de uma montagem rápida, é a utilização de uma placa universal de circuito impresso (placa perfurada), mostrada na figura 78.

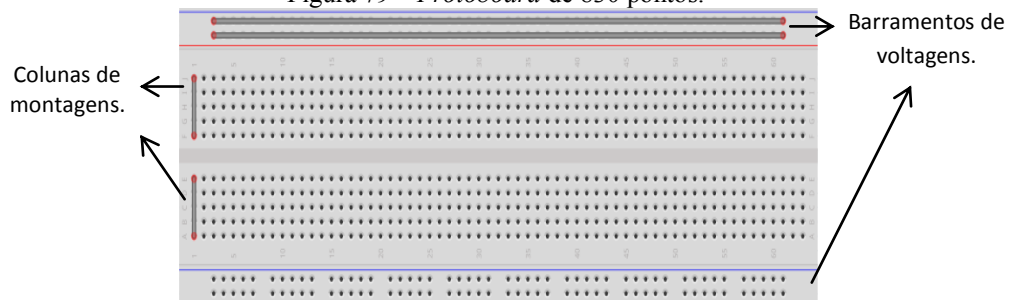
Figura 78 – Placa universal de circuito impresso de 10 cm x 4 cm).



Fonte: Do próprio autor.

Como sugestão, utilize um *protoboard* de 830 pontos para montar os circuitos do kit experimental. Um *protoboard* (também chamado de: *breadboard*, protoplaca e matriz de contados) é uma placa, com orifícios, utilizada na montagem de circuitos eletrônicos simples, sem precisar soldar componentes em uma placa de circuito impresso. No *protoboard*, mostrado na figura 79, os orifícios das linhas horizontais superiores (ou inferiores) são interligados eletricamente. Essas linhas independentes são utilizadas para os barramentos de voltagens positivas e negativas. Na área de montagem, no *protoboard*, os orifícios das colunas verticais, também, são interligados eletricamente.

Figura 79 – *Protoboard* de 830 pontos.



Fonte: Do próprio autor.

Monte a placa Arduino Nano, que será usada no controle nas atividades experimentais, e os módulos Bluetooth no *protoboard*. Depois de montar os circuitos, utilizando fios condutores (jumper para Arduino, mostrados na figura 80), cole o *protoboard* dentro da caixa de acrílico usada para montar o kit experimental.

Figura 80 – Fios *Jumpers* para Arduino.

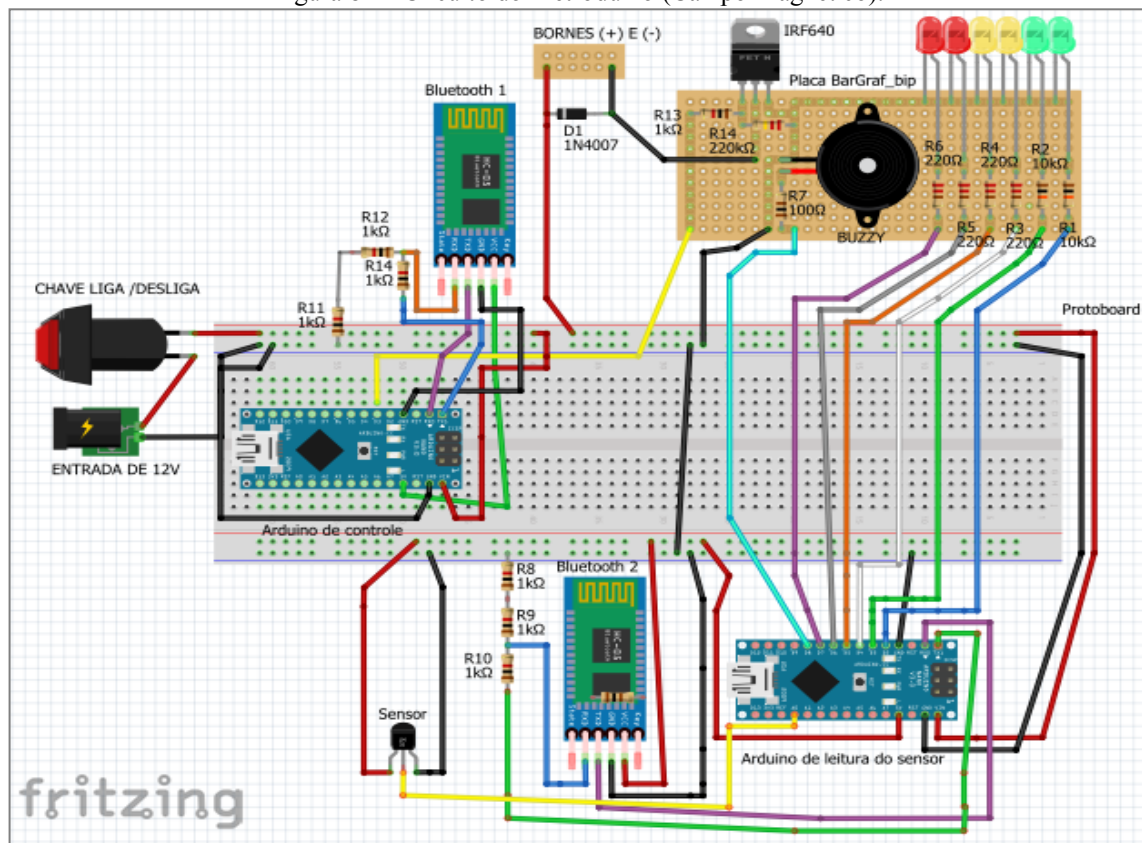


Fonte: Do próprio autor.

### 3.2.2 Circuito do Kit - Campo Magnético

O circuito do Kit Campo Magnético, mostrado da figura 81, foi desenvolvido no aplicativo *Fritzing*. Nesse circuito a placa Arduino Nano (Arduino de controle), receberá no seu pino de entra “V<sub>IN</sub>” a tensão de 12v, diretamente do barramento superior (+) do *protoboad*, através de um fio *jumper* vermelho e o pino “GND” é ligado diretamente no barramento superior (-), através de um fio *jumper* preto.

Figura 81 – Circuito do Eletroduino (Campo Magnético).

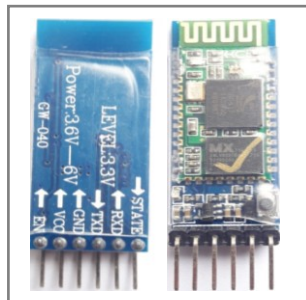


Fonte: Do próprio autor (desenvolvida no Fritzing).

No circuito do Kit Campo Magnético o pino de entra “V<sub>CC</sub>” do módulo Bluetooth 1 receberá a tensão de 5v diretamente do pino de saída 5v da placa Arduino de controle, através

do fio *jumper* verde. O pino “GND” do módulo é ligado diretamente (*jumper* preto) ao pino “GND” da placa Arduino de controle. No circuito os pinos de comunicação “TX” e “RX” do módulo Bluetooth 1 conectado ao Arduino de controle são ligados da seguinte forma: o pino de saída “TX” é ligado diretamente ao pino de entrada RX do Arduino de controle, o pino “RX” é ligado ao nó entre os resistores R12 e R14 do divisor de tensão. O pino TX da placa Arduino Nano (Arduino de controle) é ligado na extremidade livre do resistor R14 e a outra extremidade do resistor R11 é ligada no barramento GND do *proto board*. O divisor de tensão, mostrado na figura 81, formado pelos resistores (R11, R12 e R 14) deverá ser utilizado para evitar que o módulo Bluetooth não seja danificado, uma vez que o pino “RX” (do módulo) trabalha com tensão de 3.3v. A mesma ligação deverá ser feita para o módulo Bluetooth 2, conectado ao Arduino de leitura. Nessa ligação o divisor de tensão será feito através dos resistores (R8, R9 e R10). E mais ainda, no circuito da figura 81, os pinos digitais (D2, D3, D4, D5, D6 e D7), no “Arduino de leitura do sensor”, estão ligados, respectivamente, aos resistores (R1, R2, R3, R4, R5 e R6) que acendem os LEDs, do verde ao vermelho, na placa “BarGraf\_bip”. O pino “D3” do “Arduino de controle” enviará os sinais de PWM e sinais para ligar e desligar a saída do Eletroduino (bornes vermelho e preto). Esse controle será feito através do transistor mosfet IRF640. No circuito o pino “D3” é ligado, através de um fio *jumper* amarelo, ao resistor R13 (1K $\Omega$ ) conectado ao pino “G” (gatilho) do IRF640 na placa “BarGraf\_bip”. A saída positiva (borne vermelho) do Eletroduino é ligada, através de um fio *jumper* vermelho, a saída positiva da chave “Liga / Desliga” ou no barramento positivo (linha de 12v no *proto board*) e a saída negativa (borne preto) é ligada ao pino “D” (dreno) do IRF640. O pino “S” (porta) do IRF640 é ligado diretamente no barramento negativo do *proto board*. O pino “D8” da placa “Arduino de leitura do sensor” (no adaptador), no circuito, é ligado na entrada do resistor R7 que está conectado ao pino positivo do “Buzzy”. O módulo Bluetooth 1 (figura 82) ligado à placa Arduino Nano (Arduino de controle) servirá para fazer a conexão com o celular ou tablet nas atividades experimentais de Eletromagnetismo.

Figura 82 – Módulo Bluetooth HC-05.

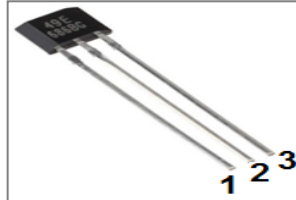


Fonte: Do próprio autor.



O sensor de campo magnético utilizado no kit experimental foi o SS49E, mostrado na figura 81, (*datasheet* do sensor no Anexo A). Esse sensor apresenta uma saída analógica que muda conforme a intensidade do campo magnético detectado.

Figura 81 - Sensor de campo magnético SS49E.



Fonte: Do próprio autor.

O pino 1 (entra  $V_{CC}$ ) do sensor de campo magnético (49E) precisará ser ligado a saída de 5v do “Arduino de leitura do sensor”. O pino 2 (saída do sensor) ligado ao pino de entrada analógica  $A_1$  do “Arduino de leitura do sensor” e o pino 3 (entrada GND do sensor) ligado ao barramento GND do *protoboard*.

A entrada do sensor de campo magnético, no Eletroduino, deverá ser feita através dos conectores (plugues) macho e fêmea de tamanhos (P10) mostrados na figura 82.

Figura 82 – Conector de entrada para plugue tamanho P10 (estéreo)



Fonte: Do próprio autor.

O plugue estéreo P10 (fêmea) deverá ser instalado na tampa da caixa de acrílico e o plugue P10 (macho) na ponteira sensor de campo magnético. Essa ponteira sensor de campo magnético deverá ser montada utilizando: dois plugues estéreos (P10, macho), um sensor 49E e um cabo flexível de três vias (três fios condutores no interior do cabo) (figura 83). Um desses plugues (P10) deverá ser soldado em uma das extremidades do cabo flexível e a outra extremidade será usada para instalar o sensor 49E dentro da capa (tampa) de um plugue P10. Para montar o sensor de campo magnético no cabo flexível, utilize um pequeno pedaço de placa perfurada (para soldar o sensor 49E) e cola quente para fazer o isolamento elétrico, antes de montar o sensor dentro da tampa do plugue P10.

Os plugues de tamanho P10 (tipo estéreo) possuem três terminais de ligações. Cada terminal do plugue P10 deverá ser utilizado para soldar os pinos (1, 2 e 3) do sensor 49E,



através o cabo flexível de três vias. As ligações precisarão ser feitas da seguinte forma: o terminal central (3) do plugue P10 macho, em destaque na figura 82, por exemplo, será ligado ao pino 2 (saída) do sensor de campo magnético; o terminal (2) do P10 deverá ser ligado ao pino 1 ( $V_{CC}$ ) do sensor 49E e o terminal 1 (P10) ligado ao pino 3 (GND) do sensor. Dessa forma as ligações para o conector estéreo P10 (fêmea) deverão ser feitas da seguinte maneira: o terminal central do plugue estéreo P10 (fêmea), correspondente ao terminal 3 (central) do plugue P10 macho, deverá ser ligado ao pino de entrada  $A_1$  do “Arduino de leitura do sensor”; o terminal do plugue P10 (fêmea), correspondente ao terminal 2 do plugue P10 (macho), deverá ser ligo ao barramento de saída de 5v do Arduino de leitura e o terminal do P10 fêmea (correspondente ao terminal 1 do P10 macho), precisará ser ligado ao barramento GND, no *proto board*. Para fazer as ligações correspondentes entre os plugues P10 macho e fêmea, será necessária a utilização de um multímetro na escala de continuidade.

Na figura 83 temos um cabo flexível de três vias utilizado para montar a ponteira do sensor de campo magnético.

Figura 83 – Cabo flexível de três vias (cabo de áudio para microfone).



Fonte: Do próprio autor.

Na figura 84 mostramos as ponteiras de sensores de campo magnético montadas para o Kit experimental.

Figura 84 – Ponteiras de sensores de campo magnético.



Fonte: Próprio autor

### 3.2.3 Materiais para o Kit - Efeito Joule

- 02 Arduino Nano.
- 01 Soquete para Arduino Nano.
- 02 Módulos Bluetooth HC 05 ou similar.
- 07 Resistores de  $1K\Omega$  (código de cores: marrom, preto, vermelho e dourado).
- 01 Resistores de  $10K\Omega$  (marrom, preto, laranja e dourado).
- 01 Resistores de  $220K\Omega$  (vermelho, vermelho, amarelo e dourado).
- 01 Transistor mosfet IRF640.
- 01 Sensor de temperatura LM35
- 01 LED's alto brilho de cor verde (tamanho 5mm).
- 01 Chave liga/desliga.
- 01 Conector fêmea para plugue de fonte de alimentação de 12v.
- 20 Fios condutores (Jumper) do tipo macho-macho para Arduino.
- 01 Plugue fêmea tamanho P10 estéreo para fixar na caixa de acrílico.
- 02 Plugue macho (tamanho P10, estéreo).
- 01 Borne de cor vermelha.
- 01 Borne de cor preta.
- 01 Peça de cabo de rede (tamanho 30cm).
- 01 Placa perfurada de circuito impresso (tamanho 4cm x 4cm).
- 01 Peça de fio flexível (microfone) de três vias – tamanho 50cm.
- 01 Protoboard (18x6).
- 01 Caixa de acrílico com tampa (comprimento: 20cm, largura: 10cm, altura: 9cm).
- Resistores de potências acima de 1w (  $47\Omega$ ,  $100\Omega$ ,  $220\Omega$ ,  $470\Omega$ ,  $1K\Omega$ ).

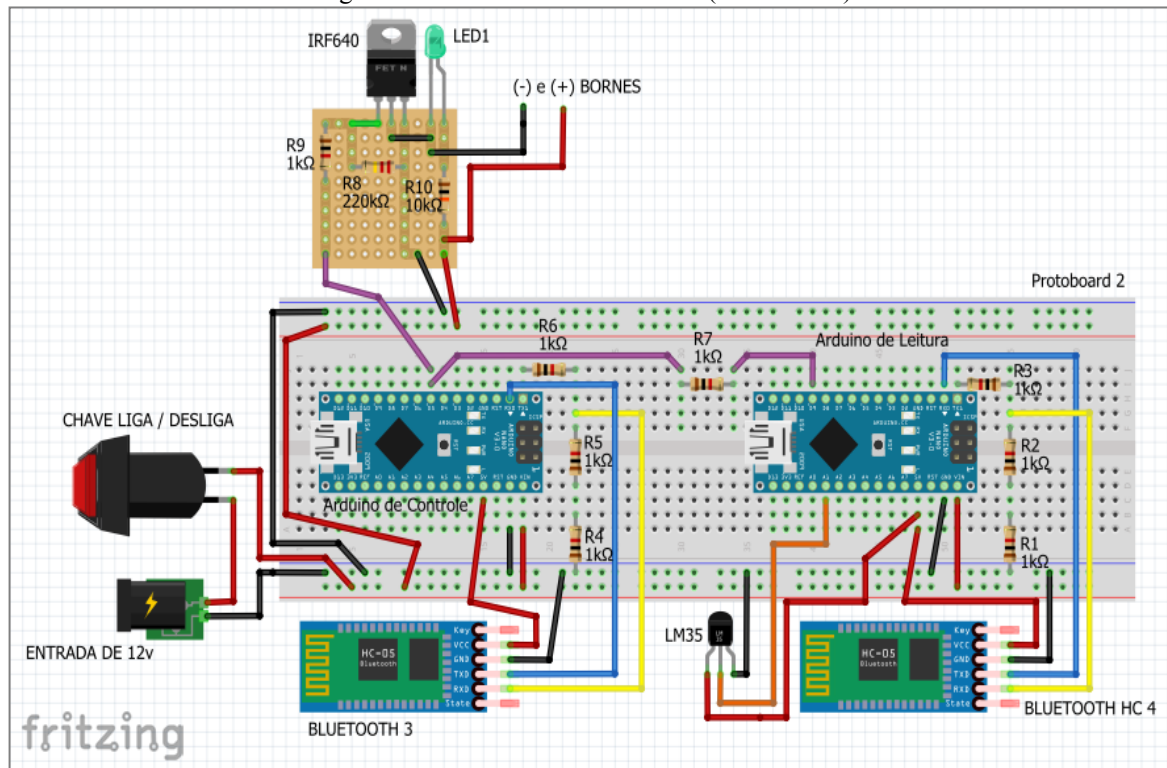
### 3.2.4 Circuito do Kit - Efeito Joule

O circuito do Kit Efeito Joule, mostrado na figura 85, deverá ser montado na caixa de acrílico com tampa (comprimento: 20cm, largura: 10cm, altura: 9cm). No circuito a placa Arduino Nano (Arduino de controle), receberá no seu pino de entrada “ $V_{IN}$ ” a tensão de 12v, diretamente do barramento inferior (+) do *protoboard*, através de um fio *jumper* vermelho e o pino “GND” é ligado diretamente no barramento inferior (-), através de um fio *jumper* preto.

O módulo Bluetooth 3 ligado à placa Arduino Nano (Arduino de controle) servirá para fazer a conexão com o celular ou tablet nas atividades experimentais de Eletromagnetismo.

No circuito do Kit Efeito Joule o pino de entrada “V<sub>CC</sub>” do módulo Bluetooth 3 receberá a tensão de 5v diretamente do pino de saída 5v da placa Arduino de controle, através do fio *jumper* vermelho. O pino “GND” do módulo é ligado diretamente (*jumper* preto) ao pino “GND” da placa Arduino de controle.

Figura 85 – Circuito do Eletroduino (Efeito Joule).

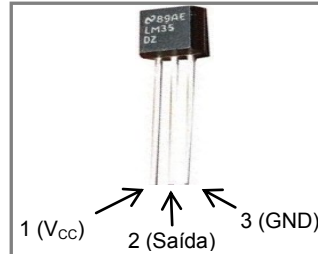


Fonte: Do próprio autor desenvolvida no Fritzing.

No circuito da figura 85 os pinos de comunicação “TX” e “RX” do módulo Bluetooth 3, conectado ao Arduino de controle, são ligados da seguinte maneira: o pino de saída “TX” é ligado diretamente ao pino de entrada RX do Arduino de controle, o pino “RX”, do módulo, é ligado ao nó entre os resistores R5 e R6 do divisor de tensão. A outra extremidade do resistor R6 é ligada diretamente no pino TX da placa Arduino Nano (Arduino de controle) e a extremidade do resistor R4 é ligada no barramento GND da *protoboard*. O divisor de tensão formado pelos resistores (R4, R5 e R6), mostrado no circuito (figura 85), deve ser utilizado para evitar que o módulo Bluetooth não seja danificado. O divisor de tensão garantirá um nível de 3.3v ao pino “RX” do módulo Bluetooth 3. A mesma ligação deverá ser feita para o módulo Bluetooth 4, conectado ao “Arduino de leitura”. O divisor de tensão do Bluetooth 4 será feito através dos resistores (R1, R2 e R3).

O sensor de temperatura utilizado no kit experimental, Efeito Joule, foi o LM35, mostrado na figura 86. Esse sensor apresenta uma saída analógica que muda conforme a variação de temperatura.

Figura 86 - Sensor de temperatura LM35.



Fonte: Do próprio autor.

O pino 1 (entra  $V_{CC}$ ) do sensor de temperatura (LM35) precisará ser ligado a saída de 5v do “Arduino de leitura”. O pino 2 (saída do sensor) ligado ao pino de entrada analógica “A<sub>1</sub>” do “Arduino de leitura” e o pino 3 (entrada GND do sensor) ligado ao barramento GND do *proto*board.

A entrada do sensor de temperatura (LM35), no Eletroduino, deverá ser feita através de conectores (plugues) macho e fêmea de tamanhos (P10). Nesse caso, as ligações e a construção da ponteira sensor de temperatura (figura 87) deverão ser feitas da mesma forma descrita para o Kit experimental, Campo Magnético, uma vez que há uma coincidência entre os pinos dos sensores 49E e LM35.

Figura 87 – Ponteiras de sensores de temperatura.



Fonte: Próprio autor.

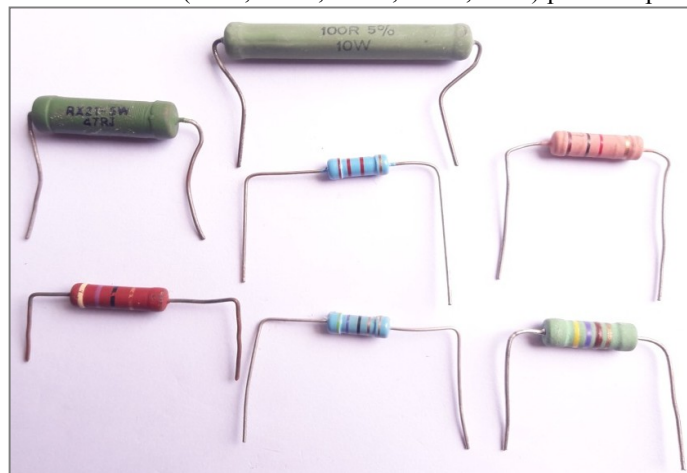
No circuito da figura 85, o pino “D9” do “Arduino de leitura” receberá através do resistor R7 ( $1K\Omega$ ), ligado ao pino “D5”, os sinais de valores (PWM) enviados pelo “Arduino de controle”. Quando um sinal (PWM) for recebido pelo “Arduino de leitura”, o tempo que o resistor passará conectado aos bornes de saída do Eletroduino, será cronometrado no aplicativo Ensino de Física. Esse tempo que o resistor ficará energizado será utilizado para calcular o consumo de energia elétrica no experimento.

O Arduino de controle, também, enviará através do pino “D5” os sinais de PWM e os sinais para ligar e desligar a saída do Eletroduino (bornes vermelho e preto). Esse controle será feito através do transistor mosfet IRF640. No circuito, da figura 85, o pino “D5” é ligado, através de um fio *jumper* lilás, ao resistor R9 (1K $\Omega$ ) conectado ao pino “G” (gatilho) do IRF640 na placa perfurada. A saída positiva (borne vermelho) do Eletroduino é ligada, através de um fio *jumper* vermelho, no barramento positivo (linha de 12v no *proto board*) ou, também, poderá ser ligada na saída positiva da chave “Liga / Desliga”. A saída negativa (borne preto) é ligada ao pino “D” (dreno) do IRF640, na placa perfurada. O pino “S” (porta) do IRF640 é ligado diretamente no barramento negativo do *proto board*.

O LED verde, presente na placa perfurada, na figura 85 será utilizado para sinalizar a saída do aparato experimental.

A figura 88 mostra os resistores ( 47 $\Omega$ , 100 $\Omega$ , 220 $\Omega$ , 470 $\Omega$ , 1K $\Omega$ ) que deverão ser utilizados nas atividades experimentais com o kit Efeito Joule. Os resistores precisarão ter potências a partir de 1w.

Figura 88 - resistores ( 47 $\Omega$ , 100 $\Omega$ , 220 $\Omega$ , 470 $\Omega$ , 1K $\Omega$ ) para o experimento.



Fonte: Próprio autor.

## 4 Fundamentos Teóricos de Física

### 4.1 A lei de Ohm e o Efeito Joule

Quando uma voltagem é estabelecida em um condutor metálico, o campo elétrico criado no seu interior faz com que um número muito elevado de elétrons livres tenha um movimento médio resultante em um sentido determinado nesse condutor, ou seja, uma corrente elétrica é estabelecida. Nesse movimento, os elétrons colidem entre si e também contra os átomos que constituem o material. Portanto, os elétrons encontram certa dificuldade para se deslocar, isto é, existe uma resistência à passagem da corrente elétrica no condutor. Nesse sentido, a resistência elétrica pode determinada pela lei OHM, e tem como unidade de medida, segundo o Sistema internacional de Unidades (SI), o Ohm ( $\Omega$ ). Assim, defini-se resistência elétrica  $R$  pela razão:

$$R = \frac{V}{i} . \quad (1)$$

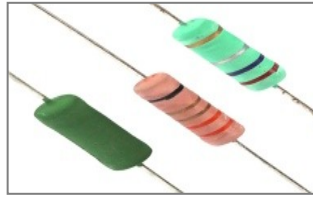
Na equação acima  $V$  é a voltagem estabelecida no condutor e  $i$  é a intensidade de corrente elétrica que o atravessa.

A passagem de corrente elétrica por um condutor gera calor esse fenômeno é conhecido como *efeito Joule*, que é causado pelos choques dos elétrons de condução com os átomos do condutor, aumentando a amplitude das vibrações térmicas da estrutura do material condutor, isto é, causando um aumento de temperatura. O *efeito joule* pode ser evidenciado pela relação entre a potência elétrica dissipada, na transformação de energia elétrica em térmica, com o quadrado da intensidade de corrente elétrica que atravessa um condutor, e tem como unidade de medida, segundo o (SI), o watt (W). Dessa forma, define-se a lei de Joule por:

$$P = R . i^2 . \quad (2)$$

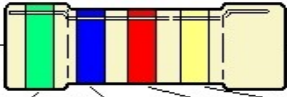
Os resistores elétricos são componentes destinados a limitação de corrente nos circuitos elétricos com a dissipação de energia em calor. Na figura 89, temos alguns resistores comerciais. E na figura 90, um código de cores para identificação do valor da resistência elétrica de resistor a partir do seu código de cor.

Figura 89 – Resistores comerciais.



Fonte: < <https://www.eletopecas.com/Produto/resistor-de-fio-5w-022> > Acesso em dez. 2017.

Figura 90 – Código de cores para determinação da resistência elétrica de um resistor.

				
COR	1ª FAIXA NÚMERO	2ª FAIXA NÚMERO	3ª FAIXA ZEROS	4ª FAIXA TOLERÂNCIA
PRETO	—	0	—	—
MARROM	1	1	0	1%
VERMELHO	2	2	00	2%
LARANJA	3	3	000	—
AMARELO	4	4	0000	—
VERDE	5	5	00000	—
AZUL	6	6	000000	—
VIOLETA	7	7	—	—
CINZA	8	8	—	—
BRANCO	9	9	—	—
OURO	—	—	X 0,1	5%
PRATA	—	—	X 0,01	10%

Fonte: Disponível em: < <https://mmalondrina.wordpress.com/2014/09/05/codigo-de-cores-e-leitura-de-resistores/> > Acesso em: dez. 2017.

No exemplo da figura a cima as cores do resistor são: verde que representa o número “5”, a segunda cor é o azul que representa o número “6” a terceira faixa de cor é a vermelha que vale “00” e a quarta faixa é a cor ouro que corresponde ao erro de leitura. Portanto o valor do resistor é  $56000\Omega$  ou  $5,6K\Omega$  com erro de leitura de  $\pm 5\%$ .

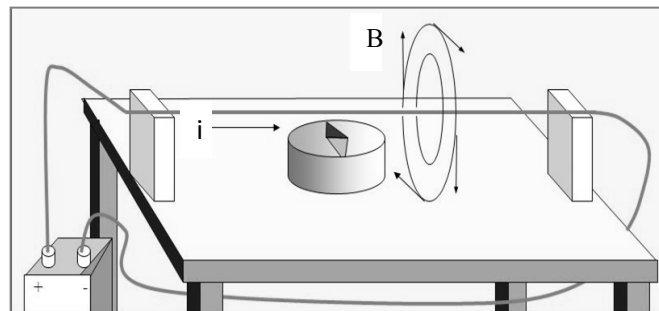
## 4.2 O Eletromagnetismo

Desde a época dos filósofos atomistas gregos já era observado que algumas pedras podiam atrair outras pedras e até mesmo o ferro. Essas pedras eram chamadas pelos gregos de magneto (pedra de magnetita). A magnetita, ímã natural permanente, é um minério naturalmente magnetizado, um tetróxido de triferro ( $Fe_3O_4$ ), encontrado na Ásia Menor na antiga cidade grega Magnésia (BASSALO, 2007). Os ímãs são corpos de materiais ferromagnéticos. Eles têm a propriedade de atrair fortemente ou repelir outros materiais ferromagnéticos, como por exemplo: ferro, níquel, cobalto, neodímio (terras raras) e etc.

O físico inglês Michael Faraday (1791 – 1867) propôs o conceito de campo magnético, analisando as linhas formadas por limalhas de ferro em uma folha de papel colocada sobre um ímã. Faraday descreveu o campo magnético como sendo a região do espaço na qual se realiza a interação magnética entre objetos magnetizados, separados a certa distância.

Em 1820 o físico dinamarquês Hans Christian Oersted (1777-1851) realizou um experimento com uma agulha magnética (bússola) perpendicularmente a um fio condutor percorrido por uma corrente elétrica. (figura 91). Ele percebeu uma pequena deflexão da agulha causada por uma força magnética, quando colocada próxima a um fio energizado e ao interromper a passagem da corrente elétrica a agulha voltava para sua posição inicial. A experiência de Oersted mostrou que um fio condutor percorrido por uma corrente elétrica produz um campo magnético ao seu redor. Portanto podemos concluir que as cargas elétricas em movimento geram um campo magnético próximo a elas.

Figura 91 - A experiência de Oersted.



Fonte:Disponível em:< <http://slideplayer.es/slide/135575/>> Acesso em dez, 2017

Campo magnético em um solenoide pode ser determinado através da lei de Ampère.

$$B = \mu \cdot n \cdot i, \quad (3)$$

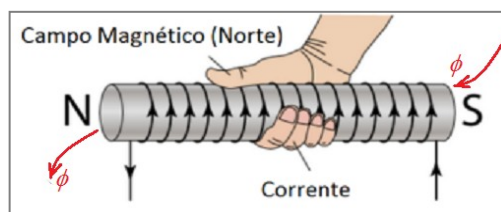
em que  $B$  é a intensidade do campo magnético;  $n$  é o número de espiras por unidade de comprimento;  $i$  é a intensidade de corrente elétrica que percorre o solenoide e  $\mu$  é a permeabilidade magnética do meio no interior do solenoide, no ar

$$\mu_0 = 4\pi \times 10^{-7} \text{ Tm/A}. \quad (4)$$

O sentido do vetor campo magnético pode ser determinado pela regra da mão direita, em os dedos envolvendo o solenoide indicam o sentido da corrente elétrica e o polegar indica o sentido do fluxo magnético, figura 92.



Figura 92 – Regra da mão direita para o solenoide.



Fonte: [https://wiki.ifsc.edu.br/mediawiki/index.php/AULA\\_13\\_-\\_Circuitos\\_2\\_-\\_Engenharia](https://wiki.ifsc.edu.br/mediawiki/index.php/AULA_13_-_Circuitos_2_-_Engenharia)

## 5 Sequência didática para aplicação do kit Eletroduino

Esta proposta didática deverá ser aplicada em turmas de terceiro ano do Ensino Médio. Sua aplicação poderá ser efetivada em sala de aula ou em um laboratório de ciências da escola. Com a finalidade de otimizar o uso da proposta didática, será necessário construir pelo menos 2 (dois) kits experimentais Eletroduinos para demonstração dos experimentos de campo magnético e efeito Joule. Com esses experimentos será possível trabalhar os conceitos de campo magnético, voltagem, resistência elétrica, corrente elétrica e transformação de energia elétrica em energia térmica.

Para uma primeira aula, utilizando o kit Eletroduino, sugerimos uma aula introdutória fazendo uma abordagem teórica expositiva sobre campo magnético, destacamos suas manifestações no cotidiano. Em seguida o professor pode fazer uma apresentação e demonstração do kit experimental (Eletroduino) para deixar os alunos familiarizados com o aplicativo Ensino de Física e com o equipamento feito com o microcontrolador Arduino. Para a demonstração, pode-se utilizar um ímã de qualquer formato e uma bobina solenoide. Na segunda aula, o professor poderá trabalhar as atividades experimentais dividindo a turma em equipes de 4 a 5 alunos. Em que cada equipe deverá receber um guia das atividades experimentais e um kit Eletroduino. Serão necessários, também, dois aparelhos celulares com o aplicativo Ensino de Física instalado, um para fazer a leitura dos dados experimentais obtidos de forma automática e o outro para fazer o controle do experimento.

Depois de realizar as atividades experimentais, o professor poderá aplicar os questionários sugeridos no apêndice D, sobre os experimentos, como forma de se verificar a aprendizagem referente a fenomenologia investigada.

Nas próximas seções apresentamos as sugestões dos guias experimentais das atividades “Lei de Ohm e Efeito Joule” e “Campo Magnético”.

## 5.1 Atividade prática 01 – Lei de OHM e Efeito Joule

A atividade prática “Lei de Ohm e efeito Joule” tem como objetivos, fazer com que os alunos compreendam a relação entre voltagem, corrente e resistência elétrica e entendam a transformação de energia elétrica em energia térmica de forma contextualizada com a realidade cotidiana.

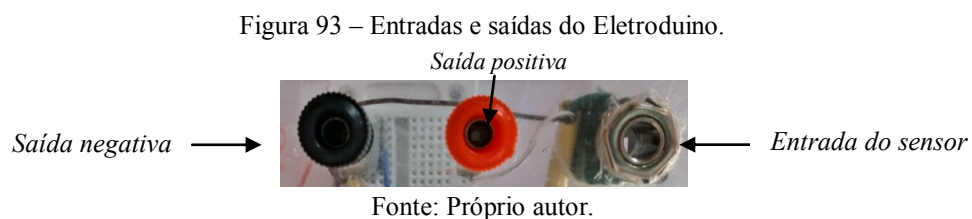
Nessa atividade experimental será possível: observar as variações de temperatura nos elementos resistivos quando submetidos a uma determinada voltagem; verificar as transformações de energia elétrica em energia térmica e fazer o levantamento de dados para que os alunos possam calcular as energias dissipadas nos circuitos elétricos.

### 5.1.1 Materiais para a atividade prática

- ✓ *Eletrouino*: equipamento de aquisição automática de dados experimentais;
- ✓ App “*Ensino de Física*”: Aplicativo de celular para controle do experimento;
- ✓ Fonte de alimentação de  $12V_{cc}$ ;
- ✓ Ponteira com sensor de temperatura;
- ✓ Resistores de potências acima de 1w (  $47\Omega$ ,  $100\Omega$ ,  $220\Omega$ ,  $470\Omega$ ,  $1K\Omega$ ).

### 5.1.2 Procedimentos Experimentais

1º passo – Conectar a ponteira do sensor de temperatura na entrada do equipamento Eletrouino, figura 93;



2º passo – Conectar a fonte de  $12V_{cc}$  na entrada de alimentação do equipamento Eletrouino e depois ligar a chave (*botão liga/desliga*), figura 94;



3º passo – Abrir o aplicativo *Ensino de Física* no celular e aperte o botão *OK* e em seguida, ative o Bluetooth do celular se não estiver ligado.

4º passo – Aperte o botão *conectar* e em seguida, na lista de módulos Bluetooths pareados, selecione o módulo Bluetooth indicado na atividade experimental.

5º passo – Conectar com os módulos Bluetooth (controle) e Bluetooth (leitura).

6º passo – Conectar um dos resistores ( $47\Omega$ ,  $100\Omega$ ,  $220\Omega$ ,  $470\Omega$ ,  $1K\Omega$ ) nas saídas positiva e negativa, e em seguida posicionar o sensor de temperatura colado (encostado) no resistor.

7º passo – Depois apertar o botão “*Ir para tela 2*” no aplicativo de leitura, em seguida ativar o “*botão 2*” no aplicativo de controle para energizar o resistor.

8º passo – Observar os valores e temperatura. Atenção, depois que a temperatura do resistor atingir  $40^{\circ}\text{C}$ , por favor, desligue o “*botão 2*” no aplicativo de controle, sem tocar no resistor e em seguida aperte o botão “Desconectar” no aplicativo de leitura para fazer as anotações na tabela 1.

9º passo – Repetir o 6º passo, anterior, e em seguida deslizar o botão “*Slider 2*” no aplicativo de controle para variar os valores de voltagem no resistor. Atenção, depois que a temperatura do resistor atingir  $40^{\circ}\text{C}$  (quarenta graus Celsius), por favor, desligue o “*botão 2*” no aplicativo de controle, sem tocar no resistor. Fazer as anotações na tabela 2, durante as observações.

10º passo – Anotar os valores de temperatura, voltagem e de tempo para cada resistor nas tabelas.

Tabela 1 – Anotações dos valores obtidos nos procedimentos com botões “liga/desliga”.

Resistores ( $\Omega$ )	Voltagem (v)	Temperatura ( $^{\circ}\text{C}$ )
1º _____		
2º _____		
3º _____		
4º _____		
5º _____		

11º passo - Verificar o efeito térmico nos resistores com o aumento da voltagem e analisar a potência dissipada e a energia consumida no processo de transformação de energia elétrica em térmica.

Tabela 2 – Anotações dos valores obtidos nos procedimentos com o botão “Slider”.

Resistor $\Omega$	Voltagem (v)	Temperatura ( $^{\circ}\text{C}$ )	Tempo (s)

## 5.2 Atividade prática 02 – Campo Magnético

A atividade prática “Campo Magnético” tem como objetivos fazer com que os alunos observem de forma fenomenológica o campo magnético em ímãs e condutores percorridos por corrente elétrica.

Nessa atividade experimental será possível observar as regiões de maiores interações dos campos magnéticos produzidos por: ímãs e por uma bobina solenoide; identificar os polos magnéticos de ímãs e de um solenoide; medir as intensidades dos campos magnéticos produzidos por ímãs e por uma bobina solenoide e, ainda, verificar o comportamento do campo magnético no interior de uma bobina solenoide sem núcleo ou com núcleo de diversos materiais.

### 5.2.1 Materiais para a atividade prática

- ✓ *Eletroduino*: equipamento de aquisição automática de dados experimentais.
- ✓ App “*Ensino de Física*”: Aplicativo de celular para controle do experimento.
- ✓ Fonte de alimentação de  $12V_{cc}$ .
- ✓ Ponteira com sensor de campo magnético.
- ✓ Ímãs de ferrite de diversos formatos.
- ✓ Ímãs de neodímio.
- ✓ Bobina solenoide.
- ✓ Núcleo de ferro ou outros materiais.
- ✓ Núcleo de ferrite.
- ✓ Caixa de acrílico com limalha de ferro.
- ✓ Bússola.

### 5.2.2 Procedimentos Experimentais

1º passo – Verificar a manifestação do campo magnético produzido por diversos ímãs, utilizando a caixa de acrílico com limalha de ferro.

2º passo – Conectar a ponteira do sensor de campo magnético (figura 95) na entrada (figura 96) do equipamento *Eletroduino*.

Figura 95 – Ponteiras de sensores de campo magnético.



Fonte: Próprio autor.

Figura 96 – Entradas e saídas do Eletroduino.



Fonte: Próprio autor.

3º passo - Conectar a fonte de 12V<sub>cc</sub> (figura 97) na entrada de alimentação do equipamento *Eletroduino* e depois ligar a chave, figura 98;

Figura 97 – Fonte de alimentação de 12v.



Fonte: Próprio autor.

Figura 98 – Entrada de fonte de 12V<sub>cc</sub> e botão liga/desliga do Eletroduino.



Fonte: Próprio autor

4º passo - Abrir o aplicativo *Ensino de Física* no celular e aperte o botão *OK* e em seguida, ative o Bluetooth do celular se não estiver ligado.

5º passo - Aperte o botão *conectar* e em seguida, na lista de módulos Bluetooths pareados, selecione o módulo Bluetooth indicado na atividade experimental.

6º passo – Conectar com os módulos Bluetooth (controle) e Bluetooth (leitura).

7º passo – Conectar o circuito do fio reto (fio da bobina solenoide) nas saídas positiva e negativa e em seguida posicionar a bússola para verificar o comportamento da agulha magnética. Depois ativar o “*Botão 1*” do aplicativo de controle para energizar o fio reto (bobina solenoide). Observar e anotar o resultado.

8º passo – Apertar o botão “*Ir para tela 2*” do aplicativo e em seguida aproximar o sensor de um ímã. Observar e anotar.

9º passo – Conectar os terminais da bobina solenoide, sem núcleo, nas saídas positiva e negativas. Depois apertar o “*botão 1*” do aplicativo ou apertar o botão “*Ir para tela 2*” e escrever na caixa de texto “Ligar campo magnético” para energizar a bobina. Observar e anotar o resultado quando o sensor estiver bem próximo do centro da bobina.

10º passo – Repetir o procedimento anterior com a bobina solenoide com núcleo de ferro. Observar e anotar o resultado.

11º passo – Inverter as polaridades da bobina solenoide e em seguida apertar o “*Botão 1*” do aplicativo de controle. Depois com o aplicativo na tela 2, aproximar o sensor da bobina e observar e anotar os resultados.

## Referências Bibliográficas

ARDUINO. **Arduino**. Disponível em: < <https://www.arduino.cc> >. Acesso em: 17/08/2017.

BASSALO, José Maria Filardo. **ELETRODINÂMICA CLÁSSICA**. 1.ed. São Paulo: Editora Livraria da Física, 2007.

BLUM, Jeremy. **Explorando o Arduino**: Técnicas e ferramentas para mágicas de engenharia. Rio de Janeiro: Alta Books, 2016.

CLARK, Andrew. **App Inventor launches second iteration**. MIT News correspondent, 2013. Disponível em: < <http://news.mit.edu/2013/app-inventor-launches-second-iteration> >. Acesso em: 30/07/2017.

FRY, Ben; REAS, Casey **Processing Overview**. Processing. Disponível em: < <https://processing.org/tutorials/overview/> >. Acesso em: 07/04/2018

GRAÇAS, Sergio. **Integração de Arduino, Bluetooth, Android com AppInventor**. Disponível em: < <https://www.youtube.com/watch?v=blvkJBAcGY0> >. Acesso em: 21/09/2016.

MCROBERTS, Michael. **Arduino Básico**. São Paulo: Novatec, 2011.

MIT – Massachusetts Institute of Technology. **MIT App Inventor – about us**. Disponível em: <http://appinventor.mit.edu/explore/about-us.html>. Acesso em: 30/07/2017.

OLIVEIRA, Cláudio; ZANETTI, Humberto. **Arduino Descomplicado**: Como elaborar projetos de eletrônica. 1 ed. São Paulo: Érica / Saraiva, 2015.

ROVAI, Marcelo. Tutorial: **Controlando o Arduino via Bluetooth**. Em 31 janeiros 2016. Disponível em: < <http://labdegaragem.com/forum/topics/tutorial-controlando-o-arduino-via-bluetooth> >. Acesso em: 01/05/2017.

SEC:Eletronics INC. IN: **Datasheet do sensor ss49e**. v2. Mar, 2008 Disponível em: <[http://files.amperka.ru/store-media/products/troyka-hall-sensor/media/SS49e\\_Hall\\_Sensor\\_Datasheet.pdf](http://files.amperka.ru/store-media/products/troyka-hall-sensor/media/SS49e_Hall_Sensor_Datasheet.pdf)>. Acesso em: 13/01/2018.

SILVEIRA, Sérgio. **Desenvolvimento de um kit experimental com arduino para o ensino de física moderna no ensino médio**. Disponível em: < [http://www1.fisica.org.br/mnpf/sites/default/files/dissertacao\\_sergio.pdf](http://www1.fisica.org.br/mnpf/sites/default/files/dissertacao_sergio.pdf) >. Acesso em: 22/06/2017.

TEXAS INSTRUNETS. IN: **DATASHEET DO SENSOR DO LM35**. Disponível em: <<http://www.ti.com/lit/ds/symlink/lm35.pdf>> Acesso em: 13/01/2018.

## **Apêndice A**

### **Códigos dos Programas para o Arduino**



## Programa de Controle

O programa de controle deverá ser copiado para o ambiente de desenvolvimento de programas do Arduino (o IDE Arduino). O programa (código) abaixo será utilizado para controle dos experimentos de Física. As portas 3, 5, 6, e 11 (no Arduino Nano), possuem a função PWM. A porta 11 pode ser usada para controle de liga/Desliga em futuros projetos. O código deverá ser compilado para a placa Arduino Nano, no Eletroduino, que fará o controle nas atividades experimentais.

```
// PROGRAMA DE CONTROLE
#include <SoftwareSerial.h>

String controle;

int controle1 = 3;
int controle2 = 5;
int controle3 = 6;
int controle4 = 9;
int controle5 = 11;

void setup () {
  Serial.begin(9600);
  pinMode(controle1, OUTPUT);
  pinMode(controle2, OUTPUT);
  pinMode(controle3, OUTPUT);
  pinMode(controle4, OUTPUT);
  pinMode(controle5, OUTPUT); //Controle só por voz
}

void loop()
{
  while (Serial.available()) //Verifique se há um byte disponível para ser lido.
  {
    delay(10); //Atraso adicionado para tornar o programa estável.
    char c = Serial.read(); //Armazenar uma leitura em série
    controle += c; //Construir a string.
```

```

}
if (controle.length() > 0)
{
    Serial.println(controle);

    // Controle dos botões:

        if (controle == "x"){digitalWrite(controle1, HIGH);}
    else if (controle == "X"){digitalWrite(controle1, LOW);}
    else if (controle == "y"){digitalWrite(controle2, HIGH);}
    else if (controle == "Y"){digitalWrite(controle2, LOW);}
    else if (controle == "z"){digitalWrite(controle3, HIGH);}
    else if (controle == "Z"){digitalWrite(controle3, LOW);}
    else if (controle == "w"){digitalWrite(controle4, HIGH);}
    else if (controle == "W"){digitalWrite(controle4, LOW);} /*Controle só por voz */

    // Controle liga/desliga por voz:

        else if (controle == "Ligar campo magnético" || controle == "ligar campo magnético")
        {digitalWrite(controle1, HIGH);}
        else if (controle == "desligar campo magnético" || controle == "Desligar campo
magnético") {digitalWrite(controle1, LOW);}
        else if (controle == "ligar resistor" || controle == "Ligar resistor")
        {digitalWrite(controle2, HIGH);}
        else if (controle == "desligar resistor" || controle == "Desligar resistor")
        {digitalWrite(controle2, LOW);}
        else if (controle == "ligar sensor" || controle == "Ligar sensor") {digitalWrite(controle3,
HIGH);}
        else if (controle == "desligar sensor" || controle == "Desligar sensor")
        {digitalWrite(controle3, LOW);}
        else if (controle == "Ligar motor" || controle == "Ligar motor") {digitalWrite(controle4,
HIGH);}
        else if (controle == "Desligar motor" || controle == "desligar motor")
        {digitalWrite(controle4, LOW);}

```

```

        else if (controle == "Ligar som" || controle == "Ligar som") {digitalWrite(controle5,
HIGH);}
        else if (controle == "Desigar som" || controle == "desligar som") {digitalWrite(controle5,
LOW);}

    /* Controle dos botões deslizantes. Controla a intensidade das portas 3, 5, 6, e 11 (no
    Arduino UNO ou Arduino Nano).*/

    char slider = controle[0];
    int value = controle[2];
    Serial.print("  ");
    Serial.println(value);
    if ( slider == 'm')
    {
        analogWrite(controle1, value);
    }
    if ( slider == 'h')
    {
        analogWrite(controle2, value);
    }
    if ( slider == 'q')
    {
        analogWrite(controle3, value);
    }
    if ( slider == 'n')
    {
        analogWrite(controle4, value);
    }
    controle=""; //Redefinir a variável.
}
}

```

## Programa Para Efeito Joule

O programa para o experimento “Efeito Joule” deverá ser copiado para o ambiente de desenvolvimento de programas do Arduino (o IDE Arduino). O programa (código) abaixo será utilizado para aquisição automática de dados dos experimentos: Lei Ohm; Efeito Joule e Consumo de energia elétrica.

O código deverá ser compilado para a placa Arduino Nano no adaptador, no Eletroduino. A mesma que utilizará o sensor LM35 (sensor de temperatura).

```
// Sketch para o Eletroduino.
```

```
// PROGRAMA EFEITO JOULE
```

```
// Temperatura e voltagem no resistor do experimento
```

```
#include <Wire.h>
```

```
const int LM35 = A1; // pino A1 ligado ao sensor LM35
```

```
float Temp = 0;
```

```
int valor_temp = 0;
```

```
int entradaPWM = 9; /* O pino 9 recebe o valor do PWM da outra placa Arduino que controla o experimento */
```

```
int valorPWM;
```

```
unsigned int tempo1, tempo2;
```

```
void setup() {
```

```
    Serial.begin (9600);
```

```
    analogReference(INTERNAL);
```

```
}
```

```
void loop() {
```

```
    // Leitura de tensão PWM
```

```
valorPWM = pulseIn(entradaPWM, HIGH);

/* Para calibração do valor da voltagem, retire as // nas duas linhas abaixo. */

//Serial.print("Valor do PWM = ");

//Serial.println(valorPWM);

int voltagem = map(valorPWM,0,1970,0,12);

Serial.print("Voltagem = ");

Serial.print(voltagem,1);

Serial.println(" v");

delay(300);

//Leitura do Tempo que o resistor permanece energizado.

if (valorPWM <=12){

    tempo1 = millis();

    }

else { tempo2 = millis()-tempo1;

Serial.print("Tempo = ");

Serial.print (tempo2/1000);

Serial.println("s");

}

// Leitura da Temperatura

valor_temp = analogRead (LM35);

Temp = valor_temp*0.1075268817204301;

Serial.print("TEMPERATURA DO RESISTOR = ");

Serial.print(Temp,1);
```

```

Serial.println(" °C");

}

```

### **Programa Para Efeito Joule Usando um Kit com placa BarGraf\_bip**

O programa para o experimento “Efeito Joule” deverá ser copiado para o ambiente de desenvolvimento de programas do Arduino (o IDE Arduino). O programa (código) abaixo será utilizado para aquisição automática de dados dos experimentos: Lei Ohm; Efeito Joule e Consumo de energia elétrica.

O código deverá ser copilado para a placa Arduino Nano no adaptador, no Eletroduino. A mesma que utilizará o sensor LM35 (sensor de temperatura).

```

// Sketch para o Eletroduino.

// NewTone(Pinofalante, frequencia, tempo)

# include <NewTone.h>

#include <Wire.h>

float Temp = 0;

int valor_temp = 0;

int entradaPWM = 9;

int  valorPWM;

unsigned int tempo1, tempo2;

const int Buzzer = 8;

int sensorReading = 0;

int frequencia = 3000;

int Tempo;

```

```
const int analogPin = A1; // the pin that the potentiometer is attached to

const int ledCount = 6; // the number of LEDs in the bar graph

int ledPins[] = {

    2, 3, 4, 5, 6, 7,

}; // an array of pin numbers to which LEDs are attached


void setup() {

    pinMode(Buzzer, OUTPUT);

    pinMode(analogPin, INPUT);

    Serial.begin(9600);

    analogReference(INTERNAL);

    // loop over the pin array and set them all to output:

    for (int thisLed = 0; thisLed < ledCount; thisLed++) {

        pinMode(ledPins[thisLed], OUTPUT);

    }

}


void loop() {

    // read the potentiometer:

    int sensorReading = analogRead(analogPin);

    // map the result to a range from 0 to the number of LEDs:

    int ledLevel = map(sensorReading, 270, 370, 0, ledCount);
```

```
// loop over the LED array:

for (int thisLed = 0; thisLed < ledCount; thisLed++) {

    // if the array element's index is less than ledLevel,

    // turn the pin for this element on:

    if (thisLed < ledLevel) {

        digitalWrite(ledPins[thisLed], HIGH);

    }

    // turn off all pins higher than the ledLevel:

    else {

        digitalWrite(ledPins[thisLed], LOW);

    }

}

if (Tempo = map(sensorReading,270,370,0,200)){

    NewTone(Buzzer,frequencia, Tempo);

    delay(100);

}

valorPWM = pulseIn(entradaPWM, HIGH);

//Serial.print("Valor do PWM = ");

//Serial.println(valorPWM);

int voltagem = map(valorPWM,0,1970,0,12);

Serial.print("Voltagem = ");

Serial.print(voltagem,1);

Serial.println(" v");
```



```
    delay(300);

//Leitura do Tempo

    if (valorPWM <=12){

        tempo1 = millis();

    }

else { tempo2 = millis()-tempo1;

    Serial.print("Tempo = ");

    Serial.print (tempo2/1000);

    Serial.println("s");

}

//Leitura da temperatura através do sensor LM35

//O Sensor LM35 enviará valores para a placa Arduino através do pino A1

// Leitura da Temperatura

valor_temp = analogRead (analogPin);

Temp = valor_temp*0.1075268817204301;

Serial.print("TEMPERATURA DO RESISTOR = ");

Serial.print(Temp,1);

Serial.println(" °C");

//Serial.print("Leitura do LM35 = ");

//Serial.println(valor_temp);

}
```

## Programa Para Campo Magnético

O programa para o experimento “Campo Magnético” deve ser copiado para o ambiente de desenvolvimento de programas do Arduino (o IDE Arduino). O programa (código) abaixo será utilizado para: detectar e medir a intensidade de campo magnético nos experimentos de Física; Identificar os polos magnéticos de ímãs de diversos formatos e bobinas solenoides. O programa deverá ser compilado para a placa Arduino Nano, no Eletroduino. A mesma placa Arduino Nano que utilizará um sensor 49E (sensor de campo magnético).

```
// PROGRAMA PARA CAMPO MAGNÉTICO
```

```
// NewTone(Pinofalante, frequencia, tempo)
```

```
# include <NewTone.h>
```

```
const int Buzzer = 8;
```

```
int sensorReading = 0;
```

```
int frequencia = 3000;
```

```
int Tempo;
```

```
const int analogPin = A1; /* Pino em que o sensor está ligado ao número de LEDs no gráfico de barras */
```

```
const int ledCount = 6;
```

```
int ledPins[] = {
```

```
  2, 3, 4, 5, 6, 7,
```

```
}; // Sequência de pinos ligados aos LEDs.
```

```
void setup() {
```

```
  pinMode(Buzzer, OUTPUT);
```

```
  pinMode(analogPin, INPUT);
```

```
Serial.begin(9600);

// Loop sobre a matriz de pinos e definição de todos como saída:

for (int thisLed = 0; thisLed < ledCount; thisLed++) {

    pinMode(ledPins[thisLed], OUTPUT);

}

}

void loop() {

    int sensorReading = analogRead(analogPin);

    // Para calibração do sensor, retire as // nas duas linhas abaixo.

    //Serial.print(" Leitura do sensor = ");

    //Serial.println(sensorReading);

    if (sensorReading > 535) {

        Serial.println("POLO SUL");

    }

    if (sensorReading < 500){

        Serial.println("POLO NORTE");

    }

    int ledLevel = map(sensorReading, 535, 650, 0, ledCount);

    for (int thisLed = 0; thisLed < ledCount; thisLed++) {

        if (thisLed < ledLevel) {

            digitalWrite(ledPins[thisLed], HIGH);

        }

        else {
```

```

    digitalWrite(ledPins[thisLed], LOW);

}

}

if (Tempo = map(sensorReading,535,680,0,200)){

    NewTone(Buzzer,frequencia, Tempo);

    delay(100);

}

//media de 10 medidas para filtrar ruídos.

long measure = 0;

for(int i = 0; i < 10; i++){

    measure += analogRead(analogPin);

    //value = measure;

}

measure /= 10;

//calculo da voltagem em V

float Voltagem = measure * 5.0 / 1023; /* Voltagem de leitura do sensor */

// Para calibração do sensor, retire as // nas três linhas abaixo para encontrar a equação para
“B”.

//Serial.print("Voltagem = ");
//Serial.print(Voltagem,2);
//Serial.println(" V  ");

//Campo Magnético para voltagem de V= 2,52v, sensor sem campo.

float B = ((Voltagem * 0.088757396) - (0.223668639)); /* Equação “B”, para medir
intensidade de campo magnético */

```

```
Serial.print(" Campo Magnético = ");  
Serial.print(B,2);  
Serial.println(" Tesla");  
delay(50);  
}
```

## **Apêndice B**

### **Calibração do Sensor 49E**

### A calibração do Sensor 49E

Segundo as informações contidas no datasheet (Anexo A) do sensor 49E, na ausência de campo magnético a voltagem no pino de saída do sensor é de 2,5v e 4.21v quando o campo magnético detectado for 1500 Gauss ou 0,15 Tesla.

A equação para medir a intensidade de campo magnético, foi obtida utilizando a leitura da voltagem do sensor do experimento e as informações contidas no datasheet do sensor 49E. Para calibração e determinação da equação para medir a intensidade de campo magnético, utiliza-se o valor da voltagem do sensor, na ausência de campo magnético, obtida no monitor serial (Tela 2) do aplicativo Ensino de Física.

A equação do experimento foi determina da seguinte forma:

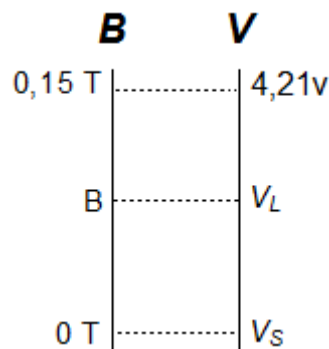


Figura – Relação entre a voltagem e a intensidades de campo magnético do sensor.

Na figura o termo  $V_L$  é a voltagem medida na saída do sensor 49E,  $V_S$  é a voltagem de referência do sensor na ausência de campo magnético e B é a intensidade do campo magnético. Portanto,

$$\frac{B}{0,15} = \frac{V_L - V_S}{4,21 - V_S} \quad (1)$$

Logo,

$$B = \frac{0,15V_L}{4,21 - V_S} - \frac{0,15V_S}{4,21 - V_S} \quad (2)$$

O valor de  $V_S$  será obtido, diretamente, na tela do celular, para ser usado na calibração do sensor. Por exemplo, para  $V_S = 2,52v$ , obtido do monitor serial do aplicativo, sensor na ausência de campo magnético, temos a equação para medir a intensidade do campo magnético com o Arduino e o sensor 49E:

$$B = 0,0887573964V_L - 0,2236686391 \quad (3)$$

## **Apêndice C**

### **Guia de Atividades Experimentais de Física**



## Guia de Atividades Experimentais de Física

Prof<sup>o</sup>.: Heberval M. Nunes

As atividades experimentais serão realizadas em grupo com a orientação do professor.

Escola: \_\_\_\_\_ Turma: \_\_\_\_ Data: \_\_/\_\_/\_\_\_\_

Nomes: \_\_\_\_\_  
\_\_\_\_\_

### Atividade prática 02 – Campo Magnético

#### Objetivos

##### Objetivo geral

- Compreender o que é campo magnético

##### Objetivos específicos

- Observar as regiões de maiores interações dos campos magnéticos produzidos por: ímãs e por uma bobina solenoide.
- Identificar os polos magnéticos de ímãs e de um solenoide.
- Medir as intensidades dos campos magnéticos produzidos por ímãs e por uma bobina solenoide.
- Calcular o valor teórico do número de espiras por unidade de comprimento de uma bobina solenoide.

#### Materiais

- *Eletroduino*: equipamento de aquisição automática de dados experimentais.
- App “*Ensino de Física*”: Aplicativo de celular para controle do experimento.
- Fonte de alimentação de  $12V_{cc}$ .
- Ponteira com sensor de campo magnético.
- Ímãs de ferrite de diversos formatos.
- Ímãs de neodímio

- Bobina solenoide.
- Núcleo de ferro.
- Núcleo de ferrite.
- Caixa de acrílico com limalha de ferro.
- Bússola.
- Circuito de um fio reto.

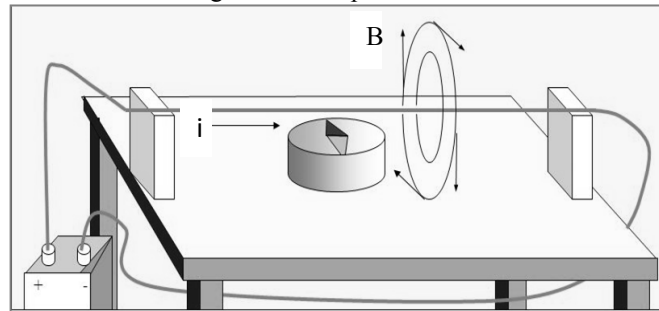
## Resumo Teórico

Desde a época dos filósofos atomistas gregos já era observado que algumas pedras podiam atrair outras pedras e até mesmo o ferro. Essas pedras eram chamadas pelos gregos de magneto (pedra de magnetita). A magnetita, ímã natural permanente, é um minério naturalmente magnetizado, um tetróxido de triferro ( $\text{Fe}_3\text{O}_4$ ), encontrado na Ásia Menor na antiga cidade grega Magnésia. Os ímãs são corpos de materiais ferromagnéticos. Eles têm a propriedade de atrair fortemente ou repelir outros materiais ferromagnéticos, como por exemplo: ferro, níquel, cobalto, neodímio (terras raras) e etc.

O físico inglês Michael Faraday (1791 – 1867) propôs o conceito de campo magnético, analisando as linhas formadas por limalhas de ferro em uma folha de papel colocada sobre um ímã. Faraday descreveu o campo magnético como sendo a região do espaço na qual se realiza a interação magnética entre objetos magnetizados, separados a certa distância.

Em 1820 o físico dinamarquês Hans Christian Oersted (1777-1851) realizou um experimento com uma agulha magnética (bússola) perpendicularmente a um fio condutor percorrido por uma corrente elétrica. Figura 1. Ele percebeu uma pequena deflexão da agulha causada por uma força magnética, quando colocada próxima a um fio energizado e ao interromper a passagem da corrente elétrica a agulha voltava para sua posição inicial. A experiência de Oersted mostrou que um fio condutor percorrido por uma corrente elétrica produz um campo magnético ao seu redor. Portanto podemos concluir que as cargas elétricas em movimento geram um campo magnético próximo a elas.

Figura 1 - A experiência de Oersted.



Fonte: Disponível em: < <http://slideplayer.es/slide/135575/> > Acesso em dez, 2017

Campo magnético em um solenoide pode ser determinado através da lei de Ampère.

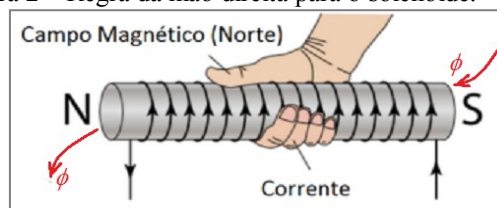
$$B = \mu \cdot n \cdot i \quad (01)$$

Onde:  $B$  é a intensidade do campo magnético;  $n$  é o número de espiras por unidade de comprimento;  $i$  é a intensidade de corrente elétrica que percorre o solenoide e  $\mu$  é a permeabilidade magnética do meio no interior do solenoide, no ar

$$\mu_0 = 4\pi \cdot 10^{-7} \text{ T} \cdot \text{m/A}. \quad (02)$$

O sentido do vetor campo magnético pode ser determinado pela regra da mão direita, em os dedos envolvendo o solenoide indicam o sentido da corrente elétrica e o polegar indica o sentido do fluxo magnético, figura 2.

Figura 2 – Regra da mão direita para o solenoide.



Fonte: [https://wiki.ifsc.edu.br/mediawiki/index.php/AULA\\_13\\_-\\_Circuitos\\_2\\_-\\_Engenharia](https://wiki.ifsc.edu.br/mediawiki/index.php/AULA_13_-_Circuitos_2_-_Engenharia)

## Procedimentos Experimentais

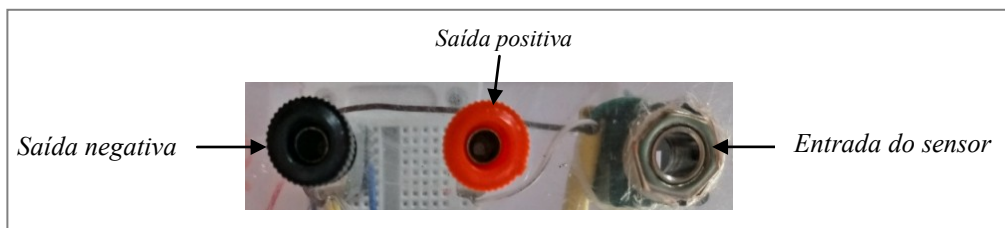
1º passo - Conectar a ponteira do sensor (figura 3) de campo magnético na entrada do equipamento *Eletroduino*, figura 4;

Figura 3 – Ponteiras de sensores de campo magnético.



Fonte: Próprio autor.

Figura 4 – Entradas e saídas do Eletroduino.



Fonte: Próprio autor.

2º passo - Conectar a fonte de  $12V_{cc}$  (figura 5) na entrada de alimentação do equipamento *Eletroduino* e depois ligar a chave, figura 6;

Figura 5 – Fonte de alimentação de 12v.



Fonte: Próprio autor.

Figura 6 – Entrada de fonte de  $12V_{cc}$  e botão liga/desliga do Eletroduino.



Fonte: Próprio autor

3º passo - Abrir o aplicativo *Ensino de Física* no celular e aperte o botão *OK* e em seguida, ative o Bluetooth do celular se não estiver ligado.

4º passo - Aperte o botão *conectar* e em seguida, na lista de módulos Bluetooth pareados, selecione o módulo Bluetooth indicado na atividade experimental.

5º passo – Conectar com os módulos Bluetooth 01 e 04 (grupo 01), com os módulos Bluetooth 03 e 06 (grupo 02) e com os módulos Bluetooth 02 e 08 (grupo 03).

6º passo - Verificar a manifestação do campo magnético produzido por: um fio reto, por um ímã e por uma bobina solenoide.

1. Verificar a manifestação do campo magnético produzido por diversos ímãs, utilizando a caixa de acrílico com limalha de ferro.
2. Conectar o circuito do fio reto nas saídas positiva e negativa e em seguida posicionar a bússola para verificar o comportamento da agulha magnética, conforme a figura 01. Depois ativar o “*Botão 1*” do aplicativo de controle para energizar o fio reto. Observar e anotar o resultado.
3. Apertar o botão “*Ir para tela 2*” do aplicativo e em seguida aproximar o sensor de um ímã. Observar e anotar.
4. Conectar os terminais da bobina solenoide, sem núcleo, nas saídas positiva e negativas. Depois apertar o “*botão 1*” do aplicativo ou apertar o botão “*Ir para tela 2*” e escrever na caixa de texto “Ligar campo magnético” para energizar a bobina. Observar e anotar o resultado quando o sensor estiver bem próximo do centro da bobina.
5. Repetir o procedimento anterior (3) com a bobina solenoide com núcleo de ferro. Observar e anotar o resultado.
6. Inverter as polaridades da bobina solenoide e em seguida apertar o “*Botão 1*” do aplicativo de controle. Depois com o aplicativo na tela 2, aproximar o sensor da bobina e observar e anotar os resultados.

## REFERÊNCIAS

FILHO, Aurélio Gonçalves; TOSCANO, Carlos. **Física e Realidade**. 1 ed. São Paulo: Scipione, v 3, 2011.

GASPAR, Alberto. **Compreendendo a Física**. 1. ed 1. São Paulo: Ática, v 3, 2011.

MÁXIMO, Antonio; ALVARENGA, Beatriz. **Curso de Física**. Ed. 1. São Paulo: Scipione, v 3, 2011.

KANTOR, Carlos A; et al. **Coleção Quanta Física**. 1 ed. São Paulo: Editora PD, v 3, 2010.

POGIBIN, Alexander; ANDRADE, Renata de; ROMERO, Talita Raquel. **Física em Contextos**. 1 ed. São Paulo: FTD, v 3, 2011.

SANT'ANNA, Blaidi; et al. **Conexões com a Física**. 1 ed. São Paulo: Moderna, v 3, 2011.

## Guia de Atividades Experimentais de Física

Prof<sup>o</sup>.: Heberval M. Nunes

As atividades experimentais serão realizadas em grupo com a orientação do professor.

Escola: \_\_\_\_\_ Turma: \_\_\_\_ Data: \_\_/\_\_/\_\_\_\_

Nomes: \_\_\_\_\_  
\_\_\_\_\_

### Atividade prática 01 – Efeito Joule e Lei de OHM

#### Objetivos

##### Objetivo geral

Compreender a transformação energia elétrica em energia térmica

##### Objetivos específicos

- Observar as variações de temperatura nos elementos resistivos quando submetidos a uma determinada voltagem;
- Verificar as transformações de energia elétrica em energia térmica;
- Calcular as energias dissipadas nos circuitos elétricos.

##### Materiais

- *Eletrôduino*: equipamento de aquisição automática de dados experimentais;
- App “*Ensino de Física*”: Aplicativo de celular para controle do experimento;
- Fonte de alimentação de 12V<sub>cc</sub>;
- Ponteira com sensor de temperatura;
- Resistores de diversos valores.

## Resumo Teórico

Quando uma voltagem é estabelecida em um condutor metálico, o campo elétrico criado no seu interior faz com que um número muito elevado de elétrons livres tenha um movimento médio resultante em um sentido determinado nesse condutor, ou seja, uma corrente elétrica é estabelecida. Nesse movimento, os elétrons colidem entre si e também contra os átomos que constituem o material. Portanto, os elétrons encontram certa dificuldade para se deslocar, isto é, existe uma resistência à passagem da corrente elétrica no condutor. Nesse sentido, a resistência elétrica pode determinada pela lei de Ohm, e tem como unidade de medida, segundo o Sistema Internacional de Unidades (SI), o Ohm ( $\Omega$ ). Assim, defini-se resistência elétrica,  $R$ , pela razão:

$$R = \frac{V}{i} \quad (01)$$

Na equação acima  $V$  é a voltagem estabelecida no condutor e  $i$  é a intensidade de corrente elétrica que o atravessa.

A passagem de corrente elétrica por um condutor gera calor esse fenômeno é conhecido como *efeito Joule*, que é causado pelos choques dos elétrons de condução com os átomos do condutor, aumentando a amplitude das vibrações térmicas da estrutura do material condutor, isto é, causando um aumento de temperatura. O *efeito joule* pode ser evidenciado pela relação entre a potência elétrica dissipada, na transformação de energia elétrica em térmica, com o quadrado da intensidade de corrente elétrica que atravessa um condutor. Dessa forma, define-se a lei de Joule por:

$$P = R \cdot i^2 \quad (2)$$

Os resistores elétricos são componentes destinados a limitação de corrente nos circuitos elétricos com a dissipação de energia em calor. Na figura 1, temos alguns resistores comerciais. E na tabela 1, um código de cores para determinação de valores da resistência elétrica de resistores.

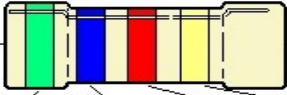
Figura 1 – Resistores comerciais



Fonte: < <https://www.eletronpeças.com/Produto/resistor-de-fio-5w-022> > Acesso em dez. 2017.



Tabela 1 – Código de cores para determinação da resistência elétrica de um resistor.



<u>COR</u>	<u>1ª FAIXA NÚMERO</u>	<u>2ª FAIXA NÚMERO</u>	<u>3ª FAIXA ZEROS</u>	<u>4ª FAIXA TOLERÂNCIA</u>
PRETO	—	0	—	—
MARROM	1	1	0	1%
VERMELHO	2	2	00	2%
LARANJA	3	3	000	—
AMARELO	4	4	0000	—
VERDE	5	5	00000	—
AZUL	6	6	000000	—
VIOLETA	7	7	—	—
CINZA	8	8	—	—
BRANCO	9	9	—	—
OURO	—	—	X 0,1	5%
PRATA	—	—	X 0,01	10%

Fonte: Disponível em: < <https://mmalondrina.wordpress.com/2014/09/05/codigo-de-cores-e-leitura-de-resistores/> > Acesso em: dez. 2017.

No exemplo da figura a cima as cores do resistor são: verde que representa o número “5”, a segunda cor é o azul que representa o número “6” a terceira faixa de cor é a vermelha que vale “00” e a quarta faixa é a cor ouro que corresponde ao erro de leitura. Portanto o valor do resistor é  $5600\Omega$  ou  $5,6K\Omega$  com erro de leitura de  $\pm 5\%$ .

## Procedimentos Experimentais

1º passo - Conectar a ponteira do sensor de temperatura na entrada do equipamento Eletroduino, figura 2;

Figura 2 – Entradas e saídas do Eletroduino.



Fonte: Próprio autor.

2º passo - Conectar a fonte de  $12V_{cc}$  na entrada de alimentação do equipamento Eletroduino e depois ligar a chave, figura 3;

Figura 3 – Entrada de fonte de 12V<sub>cc</sub> e botão liga/desliga do Eletroduino.



Fonte: Próprio autor.

3º passo - Abrir o aplicativo *Ensino de Física* no celular e aperte o botão *OK* e em seguida, ative o Bluetooth do celular se não estiver ligado.

4º passo - Aperte o botão *conectar* e em seguida, na lista de módulos Bluetooth pareados, selecione o módulo Bluetooth indicado na atividade experimental.

5º passo – Conectar com os módulos Bluetooth 02 (controle) e Bluetooth 08 (leitura).

6º passo - Verificar o efeito térmico nos resistores com o aumento da voltagem e analisar a potência dissipada e a energia consumida no processo de transformação de energia elétrica em térmica.

1. Conectar um resistor nas saídas positiva e negativa, e em seguida posicionar o sensor de temperatura colado no resistor.
2. Depois apertar o botão “*Ir para tela 2*” no aplicativo de leitura, em seguida ativar o “*botão 2*” no aplicativo de controle para energizar o resistor.
3. Observar os valores e temperatura. Atenção, depois que a temperatura do resistor atingir 40°C, por favor, desligue o “*botão 2*” no aplicativo de controle, sem tocar no resistor e em seguida aperte o botão “*Desconectar*” no aplicativo de leitura para fazer as anotações na tabela 2.
4. Conectar outro resistor nas saídas positiva e negativa e em seguida, deslizar o botão “*Slider 2*” no aplicativo de controle para variar os valores de voltagem no resistor. Atenção, depois que a temperatura do resistor atingir 40°C (quarenta graus Celsius), por favor, desligue o “*botão 2*” no aplicativo de controle, sem tocar no resistor. Fazer as anotações na tabela 3, durante as observações.
5. Anotar os valores de temperatura, voltagem e de tempo para cada resistor nas tabelas.

Tabela 2 – Anotações dos valores obtidos no procedimento 3.

Resistor ( $\Omega$ )	Voltagem (v)	Temperatura ( $^{\circ}\text{C}$ )	Tempo (s)

Tabela 3 – Anotações dos valores obtidos no procedimento 4.

Resistor: _____	Voltagem (v)	Temperatura ( $^{\circ}\text{C}$ )
1		
2		
3		
4		
5		
Tempo (s) _____		

## REFERÊNCIAS

FILHO, Aurélio Gonçalves; TOSCANO, Carlos. **Física e Realidade**. 1 ed. São Paulo: Scipione, v 3, 2011.

GASPAR, Alberto. **Compreendendo a Física**. 1. ed 1. São Paulo: Ática, v 3, 2011.

MÁXIMO, Antonio; ALVARENGA, Beatriz. **Curso de Física**. Ed. 1. São Paulo: Scipione, v 3, 2011.

KANTOR, Carlos A; et al. **Coleção Quanta Física**. 1 ed. São Paulo: Editora PD, v 3, 2010.

POGIBIN, Alexander; ANDRADE, Renata de; ROMERO, Talita Raquel. **Física em Contextos**. 1 ed. São Paulo: FTD, v 3, 2011.

SANT'ANNA, Blaidi; et al. **Conexões com a Física**. 1 ed. São Paulo: Moderna, v 3, 2011.

## **Apêndice D**

### **Questionários Referentes às Atividades Experimentais**

**ATIVIDADES EXPERIMENTAIS DE FÍSICA**

Profº.: Heberval M. Nunes

Escola: \_\_\_\_\_ Turma: \_\_\_\_ Data: \_\_/\_\_/\_\_\_\_

Nome do aluno(a): \_\_\_\_\_ Turno: \_\_\_\_\_

**Questões das Atividades práticas 01- Campo Magnético**

1ª O que aconteceu com a agulha da bússola no experimento com o fio reto?

---

---

---

---

---

2ª O que aconteceu com a intensidade do campo magnético do ímã quando o sensor era aproximado?

---

---

---

---

---

3ª - O que acontece com as intensidades do campo magnético sem e com a presença de um núcleo de material de ferro no interior de uma bobina solenoide energizada?

---

---

---

---

4ª - O que aconteceu com o campo magnético quando se inverte a polaridade de ligação da bobina solenoide?

---

---

---

---

5ª - Por que o campo magnético nas extremidades dos ímãs é mais forte que nas outras partes?

---

---

---

---

6ª - Por que o campo magnético no eixo central do solenoide é mais forte que nas laterais?

---

---

---

---

7ª - Na bobina solenoide do experimento a corrente elétrica que circulava era de 0,7A. Use a equação da lei de Ampère para determinar: a) o número de espiras por unidade de comprimento do solenoide sem o núcleo. Use o valor campo magnético medido no experimento ( $B = \text{_____}$ ) e a permeabilidade magnética do ar  $\mu_0 = 4\pi \cdot 10^{-7} \text{T.m/A}$ . b) O campo magnético na extremidade da bobina solenoide com núcleo de ferro.

## ATIVIDADES EXPERIMENTAIS DE FÍSICA

Profº.: Heberval M. Nunes

Escola: \_\_\_\_\_ Turma: \_\_\_\_ Data: \_\_/\_\_/\_\_\_\_

Nome do aluno(a): \_\_\_\_\_ Turno: \_\_\_\_\_

### Questões das Atividades práticas 02 – Efeito Joule

1ª – Explique, com suas próprias palavras, o efeito observado na prática realizada em sala de aula.

---

---

---

---

---

2ª – Com base no fenômeno observado na prática experimental, quais as consequências do aumento do uso de muitos aparelhos elétricos ligados em uma única extensão elétrica?

---

---

---

---

---

---

---

3ª – Por que os fios das ligações elétricas esquentam quando são percorridos por corrente elétrica?

---

---



---

---

---

4ª – Utilize os valores obtidos através do experimento para calcular a potência dissipada em cada resistor.

5ª – Utilizando os valores obtidos na prática experimental, determine a energia elétrica consumida por cada resistor.

6ª – A intensidade de corrente elétrica que foi estabelecida no resistor de  $100\Omega$  do experimento foi de  $0,12A$ . Supondo que essa corrente se manteve durante 10 segundos, determine: a) a quantidade total de cargas elétricas que passou pela secção transversal do resistor; b) o número de elétrons que passou pela secção, considerando o módulo de carga do elétron  $1,6 \times 10^{-19}C$ .

**Anexo A*****DATASHEET* SS49E**



## SS49E

### Linear Hall Effect Sensor

#### Features

Miniature construction  
 Low-Noise Output  
 4.5 V to 6 V Operation  
 Magnetically Optimized Package  
 Linear output for circuit design flexibility  
 Temperature range of -40 °C to 85 °C



3 pin SIP (suffix UA)

#### Description

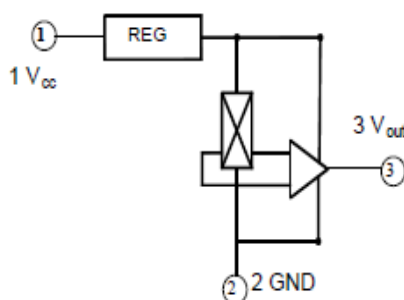
SS49E Linear Hall-effect sensor is small, versatile linear Hall-effect device that is operated by the magnetic field from a permanent magnet or an electromagnet. The linear sourcing output voltage is set by the supply voltage and varies in proportion to the strength of the magnetic field. The integrated circuitry features low

noise output, which makes it unnecessary to use external filtering. It also includes thin film resistors to provide increased temperature stability and accuracy. The linear Hall sensor has an operating temperature range of -40 °C to 85 °C appropriate for commercial, consumer and industrial environments.

#### Typical Applications

Motor control  
 Magnetic code reading  
 Ferrous metal detector  
 Current sensing  
 Position sensing

#### Functional Block Diagram

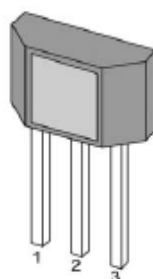




## SS49E

*Linear Hall Effect Sensor*

### Pinning



### Pin Description

Name	No	Status	Description
Vdd	1	P	Power Supply
Gnd	2	P	IC Ground
Output	3	O	Output

### Absolute Maximum Ratings

Parameter	Symbol	Value	Units
Supply Voltage (operating)	$V_{CC}$	8.0	V
Output Current	$I_{OUT}$	20	mA
Operating Temperature Range	$T_A$	-40~85	°C
Storage Temperature Rang	$T_S$	-65~150	°C



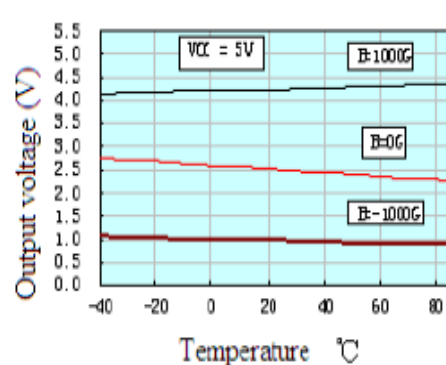
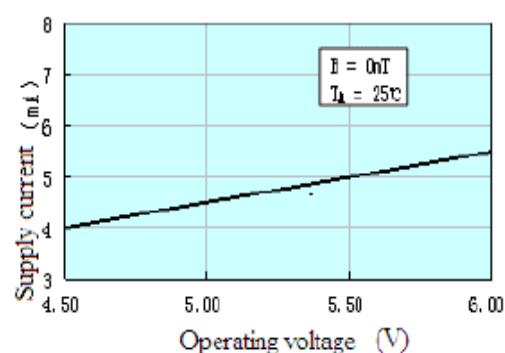
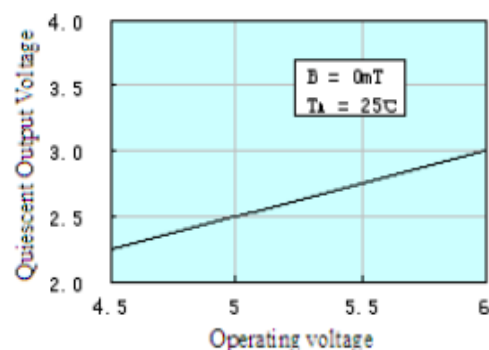
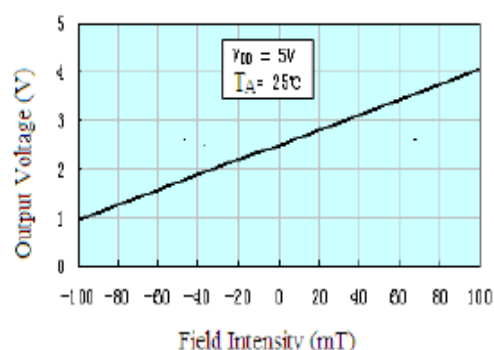
## SS49E

### Linear Hall Effect Sensor

Electrical Characteristics ( $T_A = 25^\circ\text{C}$ ,  $V_{CC} = 5.0\text{V}$ )

Parameter	Symbol	Test Conditions	Min	Typ	Max	Units
Operating voltage	$V_{CC}$	Operating	3.0		6.5	V
Supply current	$I_{CC}$	Average		4.2	8.0	mA
Output Current	$I_{OUT}$		1.0	1.5		mA
Response Time	$T_{ack}$			3		$\mu\text{s}$
Quiescent Output Voltage	$V_o$	$B=0\text{G}$	2.25	2.5	2.75	V
Sensitivity	$\Delta V_{out}$	$T_A = 25^\circ\text{C}$	1.6	1.8	2.0	mV/G
Min Output Voltage		$B = -1500\text{G}$		0.86		V
Max Output Voltage		$B = 1500\text{G}$		4.21		V

### Performance Characteristics



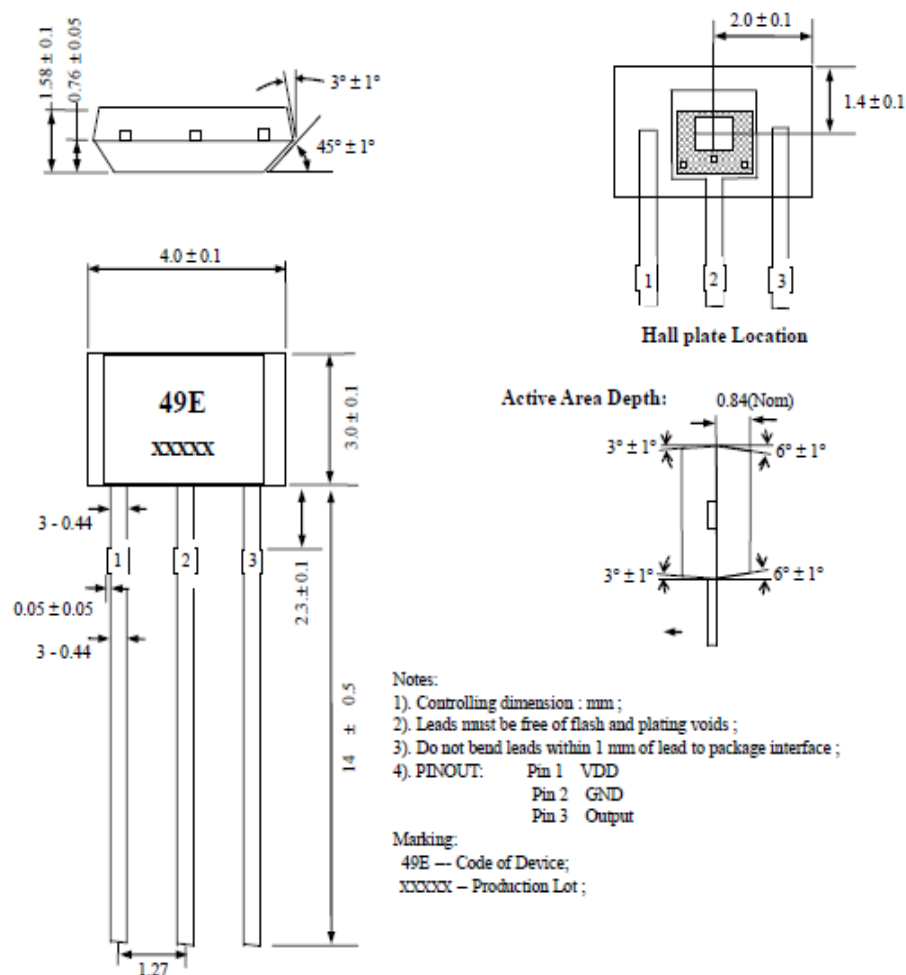


## SS49E

### Linear Hall Effect Sensor

#### Package Information

Package UA, 3-Pin SIP:



#### Ordering Information

Part No.	Pb-free	Temperature Code	Package Code	Packing
SS49E	YES	-40°C to 85°C	TO-92	Bulk, 1000 pieces/bag