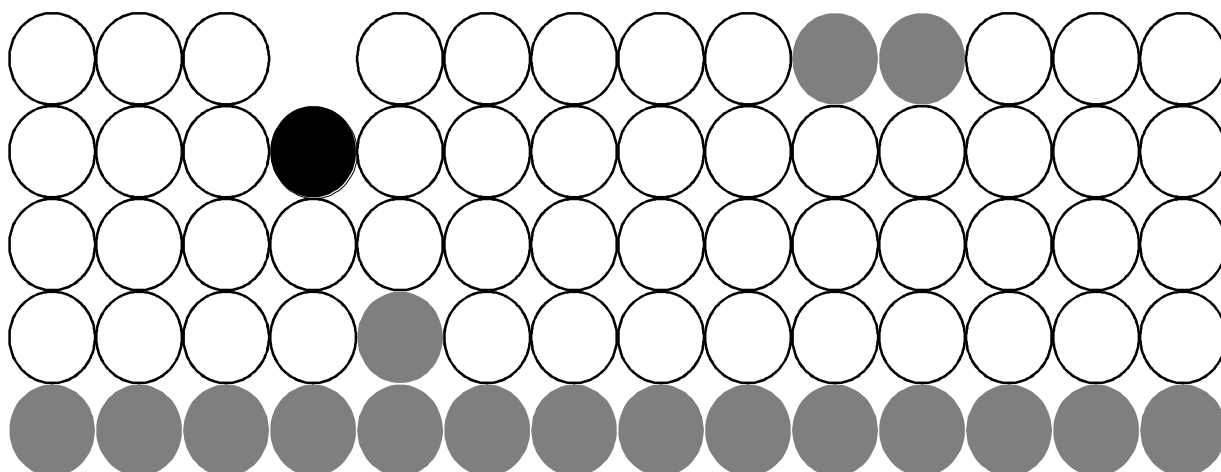


TEXTOS DE APOIO AO PROFESSOR DE FÍSICA

v.25 n.4 2014

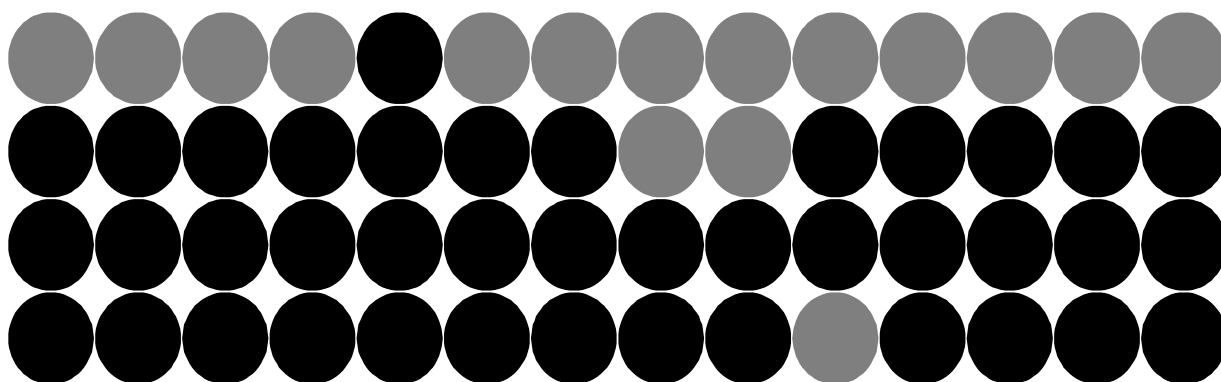
ISSN 1807-2763



Arduino para físicos

Uma ferramenta prática para aquisição de dados automáticos.

Rafael Frank de Rodrigues
Silvio Luiz Souza Cunha



Programa de Pós-Graduação em Ensino de Física

Textos de Apoio ao Professor de Física, v.25 n.4, 2014.
Instituto de Física – UFRGS
Programa de Pós – Graduação em Ensino de Física
Mestrado Profissional em Ensino de Física

Editores: Marco Antonio Moreira
Eliane Angela Veit

Dados Internacionais de Catalogação na Publicação (CIP)
Setor de Processamento Técnico
Biblioteca Professora Ruth de Souza Schneider
Instituto de Física/UFRGS

R696a Rodrigues, Rafael Frank de

Arduino para físicos : uma ferramenta prática para
aquisição de dados automáticos / Rafael Frank de Rodrigues
e Silvio Luiz Souza Cunha – Porto Alegre: UFRGS, Instituto
de Física, 2014.

34 p.; il. (Textos de apoio ao professor de física / Marco
Antonio Moreira, Eliane Angela Veit, ISSN 1807-2763; v. 25 ,
n.4)

1. Ensino de Física	2. Ensino Médio	3. Arduino
4. Sensores	I. Cunha, Silvio Luiz Souza	II. Título
II. Série.		

Impressão: Waldomiro da Silva Olivo
Intercalação: João Batista C. da Silva



Arduino para físicos

Uma ferramenta prática para aquisição de dados automáticos.

Rafael Frank de Rodrigues

Silvio Luiz Souza Cunha

Porto Alegre

2014

Prefácio

Este material é uma breve introdução ao uso do Arduino como ferramenta do Ensino de Física. Neste material vocês encontrarão definições, explicações e utilizações de sensores com a placa Arduino. Este produto é fruto de um projeto de Mestrado Profissional em Ensino de Física no qual foi desenvolvida, com alunos do Ensino Médio, uma estação meteorológica. Os esquemas e as programações contidas neste material fizeram parte deste trabalho e são apresentados como exemplos do uso do Arduino com alguns sensores. O material foi escrito supondo que o leitor tenha algum conhecimento ainda que rudimentar de programação e de eletrônica.

Sumário

I - Arduino.....6

1.1 A placa.....	7
1.2 Programação.....	8
1.2.1 Exemplo 1: Porta digital.....	9
1.2.2 Exemplo 2: Porta analógica.....	11
1.2.3 Exemplo 3: Função PWM.....	13

2 - Sensores: Exemplos De Aplicação.....16

2.1 LM35: Sensor para temperatura.....	16
2.2 LDR: Sensor para luminosidade.....	18
2.3 HIH-4000-001: Sensor para umidade.....	20
2.4 Reed switch: Sensor para biruta.....	21
2.5 Fotodiodo: Sensor para vento.....	25
2.6 BMP085: Sensor para pressão.....	27

3 – Arduino e o Aluno.....29

Referencias Bibliográfica30

Arduino

O Arduino foi desenvolvido inicialmente como uma ferramenta para profissionais das Artes-Plásticas e, pelo seu fácil uso e baixo custo, caiu no

gosto das pessoas que precisavam desenvolver projetos com aquisição automática de dados. O seu caráter *open source* levou a uma rápida disseminação, com o surgimento de várias placas genéricas a preços muito acessíveis.



Figura 1- Logo marca do Arduino

entradas e saídas, digitais e analógicas. A programação do Arduino é baseada na linguagem *Wiring*¹, que lembra muito a linguagem C++ e pode ser feita através de um aplicativo próprio o IDE - *Integrated Development Environment* (Ambiente Integrado de Desenvolvimento) - que por sua vez é baseado no *Processing*².

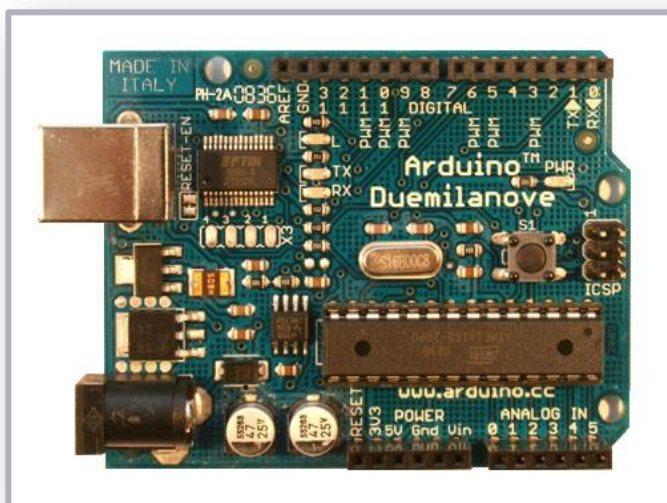


Figura 2- Foto do Arduino Duemilanove

¹Mais informações em <http://pt.wikipedia.org/wiki/Wiring>

²Mais informações em: [http://pt.wikipedia.org/wiki/Processing_\(linguagem_de_programa%C3%A7%C3%A3o\)](http://pt.wikipedia.org/wiki/Processing_(linguagem_de_programa%C3%A7%C3%A3o)).

I.1 – A placa

Existem no mercado vários modelos de placas Arduino. As mais utilizadas são a Uno, Duemilanove e Mega. No quadro que segue estão as principais características de cada placa.

Tabela 1 – Características do hardware do Arduino Uno, Duemilanove e Mega

Placa	Uno	Duemilanove	Mega
Microcontrolador	ATmega328	ATmega168	ATmega1280
Tensão de funcionamento	5V	5V	5V
Tensão de entrada	6-20V	6-20V	6-20V
E/S Digitais	14	14	54
Entradas analógicas	6	6	16
Flash Memory	32k	16k	128k
Clock	16Hz	16Hz	16Hz

A placa Mega sendo mais robusta, com mais memória e mais portas de entrada e saída é mais adequada para uso em projetos mais elaborados. Já as placas Uno e Duemilanove são mais baratas, mas oferecem recursos suficientes para desenvolver a maioria dos projetos. Portanto, vamos descrever as portas destas placas, que são praticamente iguais, pois quase não há diferença no *Hardware* das mesmas.

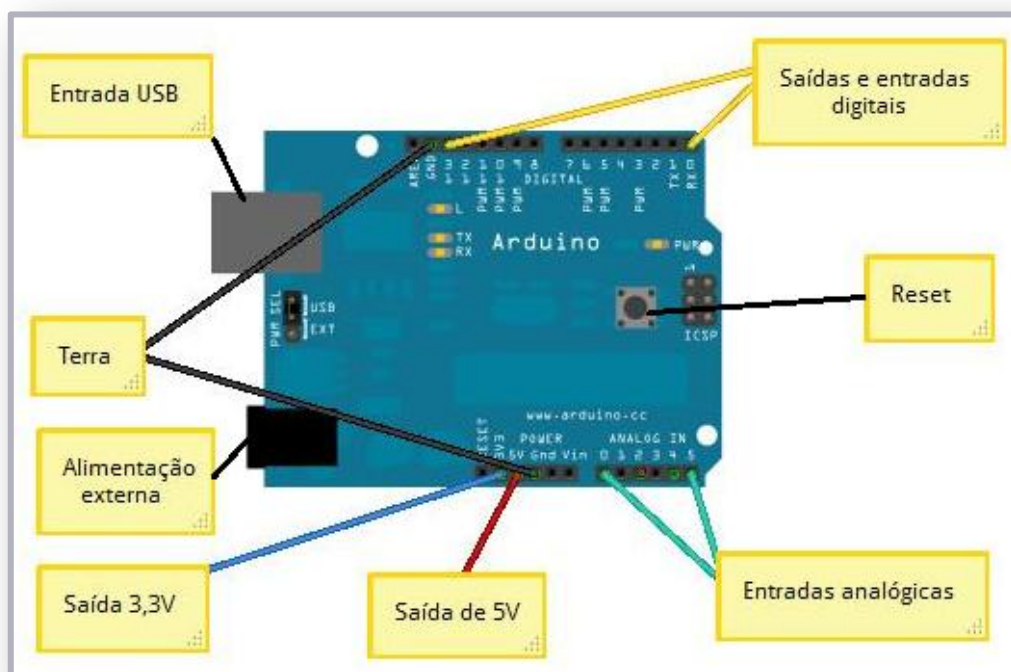


Figura 3 – Esquema de entradas e saídas do Arduino Uno ou Duemilanove

A placa Arduino pode ser alimentada através da conexão USB ou com uma fonte de alimentação externa de 5V e 500mA. A fonte de energia é selecionada automaticamente. A conexão com o micro dá-se através da porta USB do Arduino.

Os pinos digitais 3, 5, 6, 9, 10 e 11 podem usar a função PWM – *Pulse-Width Modulation* (modulação por largura de pulso). Os pinos digitais 0 e 1, servem para a comunicação serial com o computador RX e TX. Já os pinos 2 e 3 podem ser usados para gerar uma interrupções.

I.2 - Programação

Para programar o Arduino utilizamos o aplicativo IDE, figura 4 e 5. O IDE em suas várias versões pode ser encontrado no site <http://arduino.cc/en/Main/Software>.

A programação do Arduino dá-se através de uma linguagem própria, baseada

e *Wiring* que lembra muito a linguagem C++. No *site* do Arduino encontramos um vasto material de apoio, histórico do Arduino, modelos, fóruns, referências para a linguagem e exemplos. No próprio aplicativo IDE do Aduino encontramos alguns exemplos.

Antes de programar é importante configurar o aplicativo IDE para o modelo de placa a ser utilizada, como mostrado na figura 5 e este estará pronto para receber o código da programação. Quando a programação estiver completa basta clicar em *Upload* para finalizar e

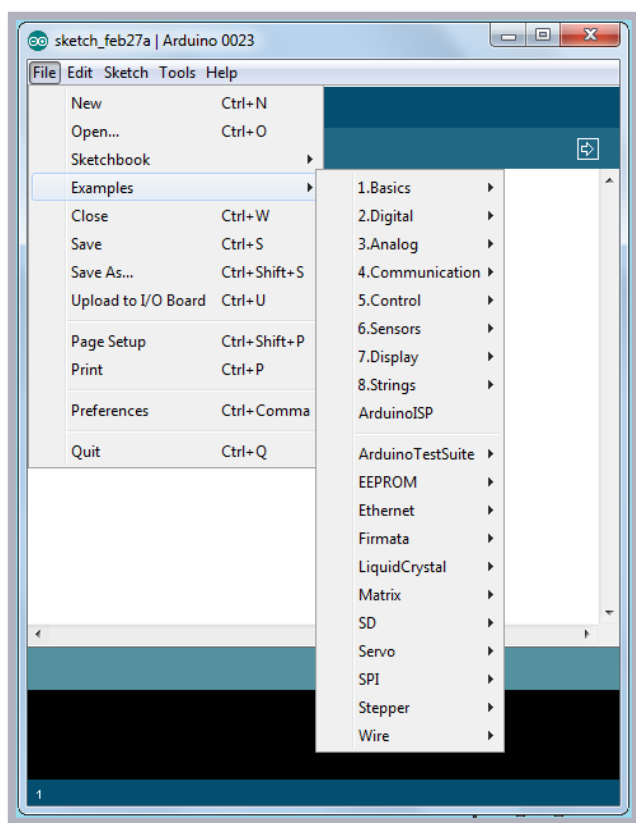


Figura 4- IDE do Arduino versão 023

carregar o programa no Arduino. O IDE faz uma compilação do código, verifica se não há erros de comandos na programação e caso haja ele cancela o *Upload* e gera um aviso na tela, do contrário carrega o código compilado no Arduino.

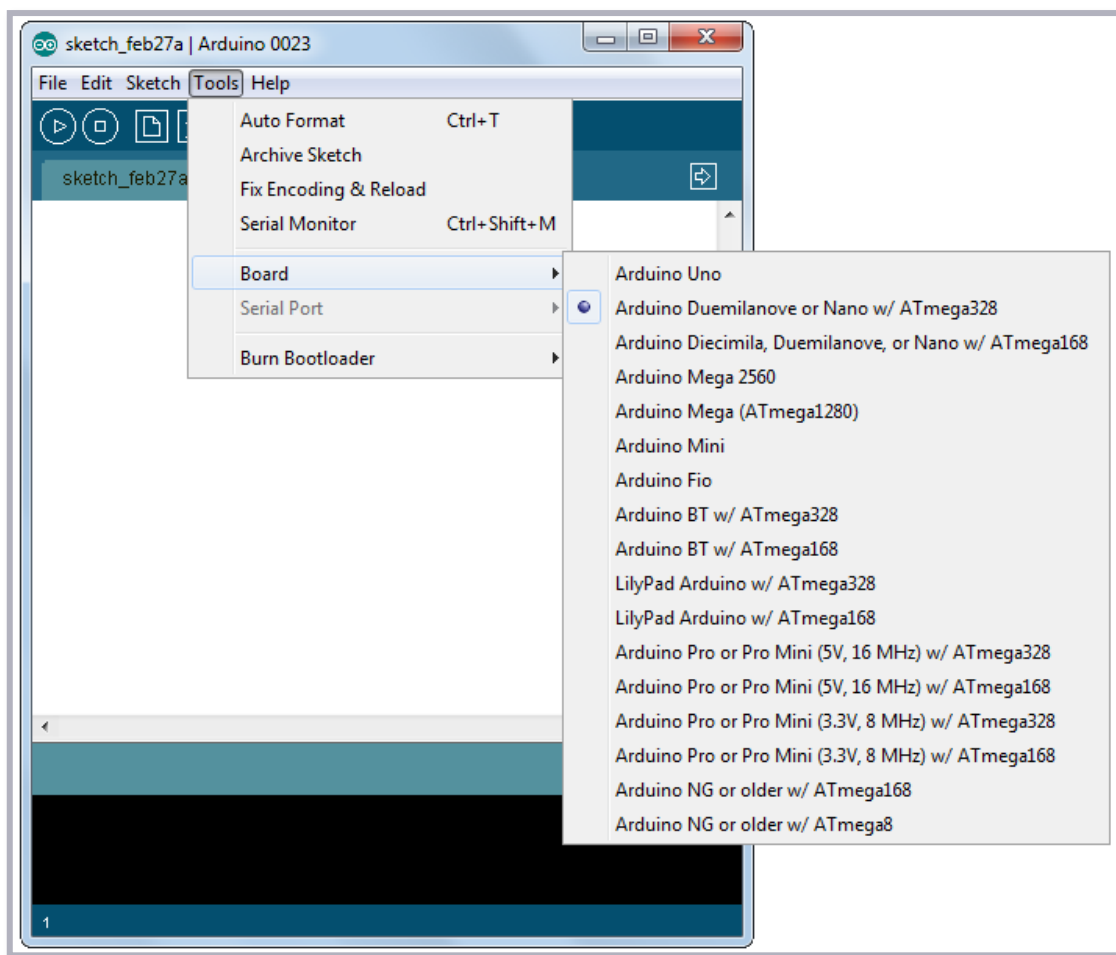


Figura 5 – Utilizando o programa IDE para escolher o modelo de placa a ser utilizada

1.2.1 –Exemplo I: Porta digital

Para demonstrar a utilização da porta digital do Arduino usaremos um exemplo retirado do aplicativo IDE. Este exemplo é o *Blink* que tem por finalidade fazer piscar um LED em intervalos de tempo iguais. O esquema elétrico necessário para ativar o LED através de uma porta do Arduino está mostrado na Figura 6.

Material:

- LED
- Resistor de 100Ω
- Fios
- *Protoboard*
- Arduino

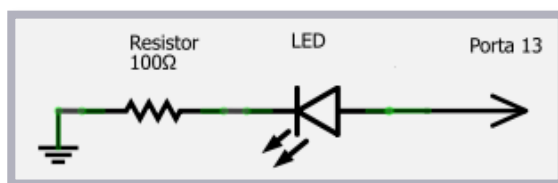


Figura 6 – Esquema elétrico para para ativar um LED.

A montagem do circuito é bem simples a perna maior do LED é conectada na porta 13 do Arduino. A outra perna conecta-se a resistência e esta por sua vez no GND do Arduino, figuras 6 e 7.

Resistência elétrica é uma oposição à corrente elétrica em um condutor. Essa oposição poderá ser maior ou menor dependendo da natureza do condutor. Observa-se que para a mesma tensão elétrica obtêm-se correntes diferentes para diferentes condutores.

Código: Na programação para o Arduino o código é sempre dividido em três partes. A primeira é a declaração de variáveis. A segunda parte é a definição dos parâmetros das portas. A terceira é o bloco de comandos. Na linguagem de programação para o Arduino qualquer texto que for escrito após “//” até o final de cada linha de código será considerado como comentário e será ignorado durante a compilação pelo IDE.

```
// sempre se inicia com as declarações das variáveis

Int LED =13; // O valor inteiro LED é igual a 13

void setup() { // Definições das portas do Arduino

  pinMode(LED, OUTPUT); //Define LED(13) como uma porta
  de saída

}

void loop() { // Corpo do programa

  digitalWrite(LED, HIGH); // Ativa a porta digital 13
  delay(1000); // Espera 1000ms
  digitalWrite(LED, LOW); // Desativa a porta digital 13
  delay(1000); // Espera 1000ms

}
```

Montagem: A figura 7 está mostrando a montagem sobre o *protoboard* do circuito do LED com o Arduino.

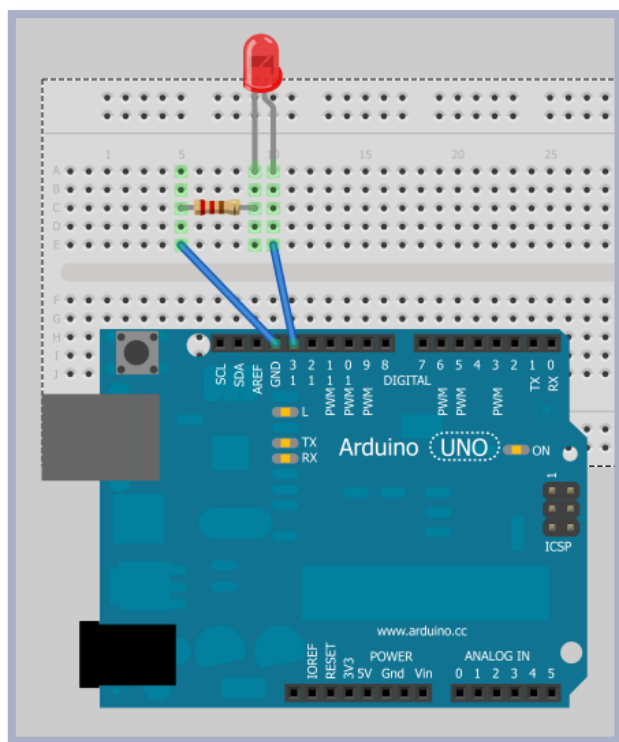


Figura 7 – Esquema de montagem para o exemplo *Blink*

1.2.2 –Exemplo 2: Porta analógica

Com este exemplo iremos ver duas novas funções; uma de leitura da porta analógica e outra de impressão de valores. Para tanto usaremos um LDR que é um resistor que tem seu valor alterado dependendo da luminosidade incidente sobre ele. O LDR pode então ser utilizado como um sensor de LUZ.

Material:

- LDR³
- Resistor de 100Ω
- Fios
- *Protoboard*
- Arduino

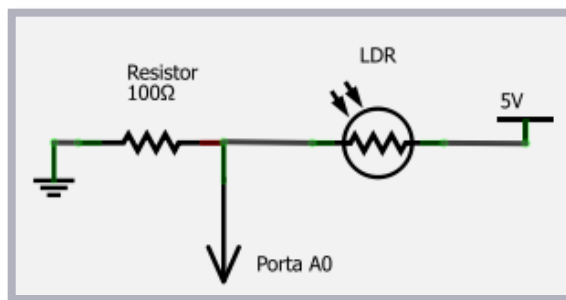


Figura 8. Esquema elétrico para a leitura da ativação de um LDR através da Porta A0.

³ Datasheet do LDR em <http://pdf1.alldatasheet.net/datasheet-pdf/view/124422/ETCI/NORPI2.html>

A figura 8 mostra o esquema elétrico para a leitura da ativação de um LDR através da Porta A0. Esta porta lê a DDP (diferença de potencial elétrico) sobre o LDR. O sinal lido na porta A0 será proporcional à intensidade luminosa incidente sobre o LDR

Tensão elétrica ou **diferença de potencial** (DDP) é a diferença de potencial elétrico entre dois pontos ou a diferença em energia elétrica potencial por unidade de carga elétrica entre dois pontos. A diferença de potencial é igual ao trabalho que deve ser feito, por unidade de carga contra um campo elétrico para se movimentar uma carga qualquer.

Código:

```
Int LDR;    // Declara que a variável LDR é um número
inteiro

void setup() {    // Definições das portas do Arduino

  pinMode(A0, INPUT); //Define A0 como uma porta de
  entrada de valores.O A0 representa porta analógica 0
  Serial.begin(9600); // Ativa a impressão de valores na tela
  do micro, 9600 é a velocidade de transferência

}

void loop() {    // Corpo do programa

  LDR = analogRead(A0);    // Lê o valor da porta A0 e
  armazena na variável LDR
  Serial.print("Valor lido: ");    //Imprime na tela o que
  estiver entre aspas
  Serial.println(LDR);    // imprime o valor de
  armazenado em LDR e vai para uma nova linha
  delay(200);    // espera 200ms
}
```

OBS: Para visualizar os resultados do comando *serial.print()* você deve clicar no botão *Serial Monitor*, mostrado na figura 9. Este irá abrir um tela auxiliar com a impressão dos dados.

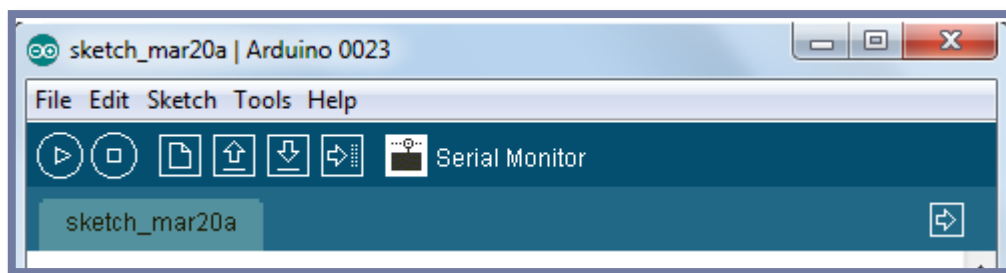


Figura 9 – Imagem da parte superior do programa IDE do Arduino

Montagem: A figura 10 está mostrando a montagem sobre o *protoboard* do circuito do LDR com o Arduino.

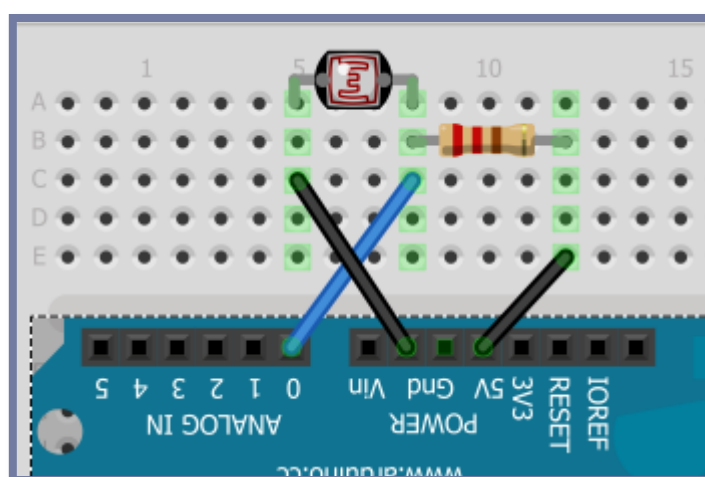


Figura 10 – Esquema de montagem para o exemplo *porta analógica*

1.2.3 – Exemplo 3: Função PWM

Neste exemplo iremos variar o brilho de um LED através de um potenciômetro. Para podermos fazer isso iremos usar duas novas funções: a *map* que faz uma regra de três, convertendo o valor lido na porta analógica entre 0 e 255. E a função PWM que está associada às portas digitais do 3, 5, 6, 9, 10 e 11 do Arduino. A função PWM faz variar a largura do pulso de tensão de saída da porta digital. Quando se utiliza a porta 3, 9, 10 e 11 a frequência dos pulsos é de 490Hz e nas portas 5 e 6 é de 98Hz. O valor desta função vai de 0 a 255, onde 255 corresponde a 100% de pulso de saída, figura 11.

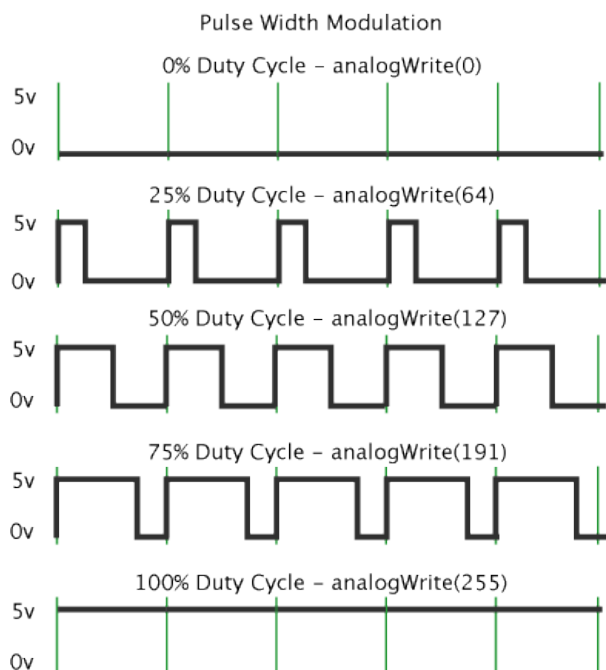


Figura 11 – Representação da saída de tensão quando se usa a função PWM. Para valor 0 temos tensão igual a zero, mas para 255 temos uma tensão contínua de 5V. Qualquer outro valor vai gerar um onda quadrada de 5V, com intervalos de tempo iguais. Imagem retirada de <http://arduino.cc/en/Tutorial/PWM>.

Material:

- LDR
- Resistor de 100Ω
- Fios
- Potenciômetro de 10k
- *Protoboard*
- Arduino

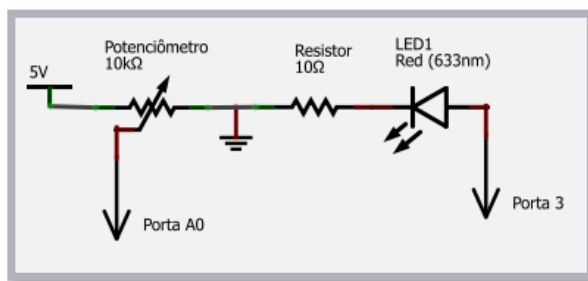


Figura 12 – O esquema elétrico apresenta a leitura do potenciômetro pela porta A0 e o controle do LED pela porta 3.

A porta A0 será usada para ler o valor de um potenciômetro submetido a uma diferença de potencial de 5V. O brilho do LED será ajustado em função desta leitura. O valor da porta A0 será responsável pelas alterações dos pulsos de tensão da função PWM. É conveniente colocar uma resistência em série com o LED para que a corrente elétrica não seja demasiadamente alta e o danifique.

Corrente elétrica é o fluxo ordenado de partículas portadoras de carga elétrica, ou também, é o deslocamento de cargas dentro de um condutor, quando existe uma diferença de potencial elétrico entre as extremidades.

Código:

```
Int POT; // Declara que a variável POT é um número
Int LED; // Declara que a variável LED é um número
```

```
void setup() {    // Definições das portas do Arduino

pinMode(A0, INPUT); //Define A0 como uma porta de
  entrada de valores.O A0 representa porta analógica 0

pinMode(3, OUTPUT); //Define 3 como uma porta de saída.

}

void loop() {    // Corpo do programa

POT = analogRead(A0); // Lê o valor da porta A0 e
  armazena na variável POT
LED = map(POT,0,1023,0,255); // Usa o valor da variável
  POT que varia de 0 a 1023 para converter em número de 0
  a 255. Que são os valores que a função PWM suportam
analogWrite(3, LED); // Manda a porta 3 liberar uma tensão
  na forma de onda quadrada onde a largura é definida pelo
  valor da variável LED
```

Montagem: Na figura 13 está mostrando a montagem sobre o *protoboard* do exemplo 3.

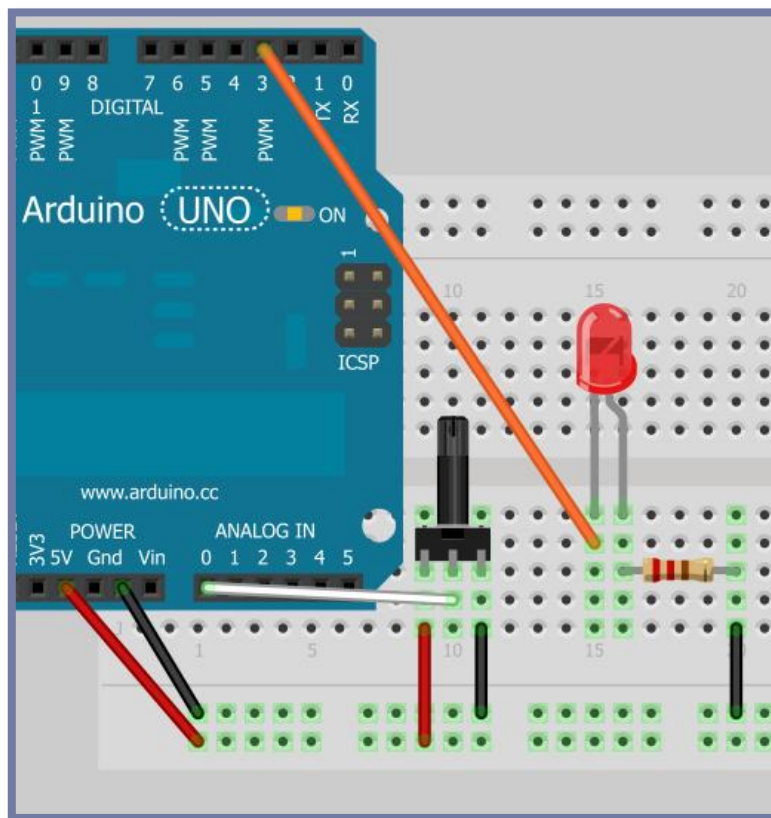


Figura 13 – Esquema da montagem do exemplo 3, função PWM

Sensores: Exemplo de Aplicação

A seguir serão mostrados alguns exemplos do uso de diferentes sensores com o Arduino que podem ser usados em aulas ou na construção de equipamentos e experimentos com aquisição automáticos de dados.

2.1 – LM35: Sensor para temperatura

O LM35 é um sensor de temperatura que nos remete valores na escala celsius, linear ($10\text{mV}/^{\circ}\text{C}$) e pode medir valores entre -55°C a 150°C . Para medir valores negativos de temperatura é necessário de uma fonte com tensão negativa.

Temperatura é uma medida da energia cinética média de translação das moléculas de um gás ideal. Ou podemos dizer que é a quantidade que diz quão quente ou frio um corpo se encontra, em relação a um determinado padrão.

Material:

- LM35⁴
- *Protoboard*
- Arduino

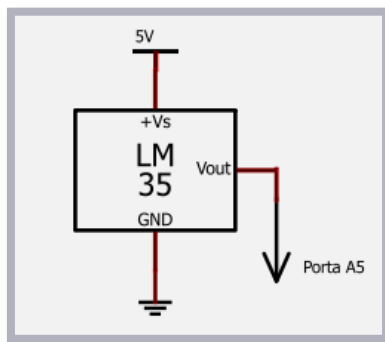


Figura 13 – Esquema de ligação do LM35

A ligação é bem simples, não há necessidade de se conectar outros componentes para medir temperaturas positivas, figura 13 e 14. Para medir temperaturas negativas deve-se substituir o GND por uma fonte de tensão negativa, -5V.

⁴ Datasheet do LM35 em <http://pdf1.alldatasheet.net/datasheet-pdf/view/8866/NSC/LM35.html>

Código:

```

float sensor; // declara a variável sensor
float cont;   // declara a variável cont

void setup(){

  pinMode(A5, INPUT); //Define A5 como uma porta de
  entrada de valores.O A5 representa porta analógica 5
  Serial.begin(9600); // Ativa a impressão de valores na tela
  do micro, 9600 é a velocidade de transferência

}

void loop(){

  sensor = analogRead(A5); // Lê o valor da porta A5 e
  armazena na variável sensor
  cont=(sensor*5)/1023;    // o valor lido de tensão na porta
  analógica "sensor" será armazenado com uma precisão de 1
  Kbyte (1024 bytes) de variação. Então 0V está 0 e 5V para
  1023
  Serial.print("Temperatura: "); //Imprime na tela o que
  estiver entre aspas
  Serial.println(cont*100); // como cada 0,01V é um 1°C
  multiplicamos por 100 o cont que é o valor da tensão.
  Imprime o valor e pula uma linha
  delay(1000); // espera 1s

}

```

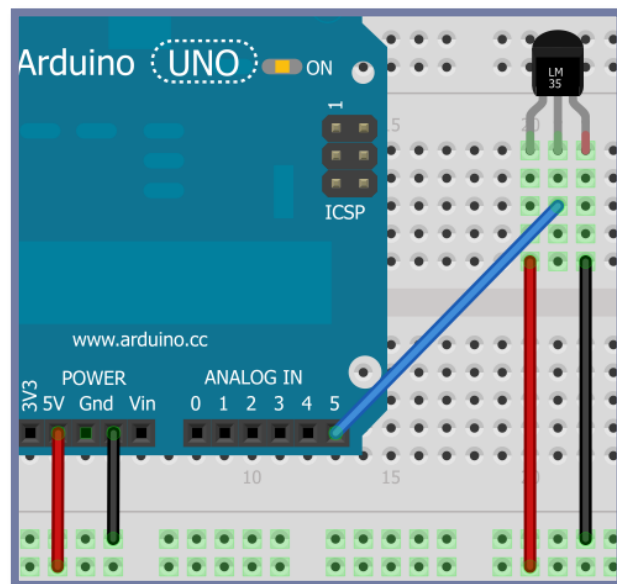
Montagem:

Figura 14 – Ilustração da montagem do sensor LM35 no *ProtoBoard* com o Arduino.

2.2 – LDR: Sensor para luminosidade

O LDR é um resistor que varia sua resistência quando exposto à luz. Com três LDRs em série ou mais, dispostos em arco, podemos verificar a variação da luminosidade do local. Determinando assim se é um dia claro, escuro ou encoberto. Neste exemplo é sugerida a montagem dos LDRs em um arco de modo a captar a luz do amanhecer ao entardecer, este arco deve estar alinhado com o movimento solar.

Associação de resistores:

Associação em série ocorre quando os resistores estão ligados um ao outro sem haver nenhum outro elemento de circuito entre eles.

Associação em paralelo ocorre quando os dois terminais de cada uma das resistências estiverem ligados aos mesmos pontos do circuito.

Material:

- *Protoboard*
- Arduino
- Resistor de 300Ω
- 3 LDRs

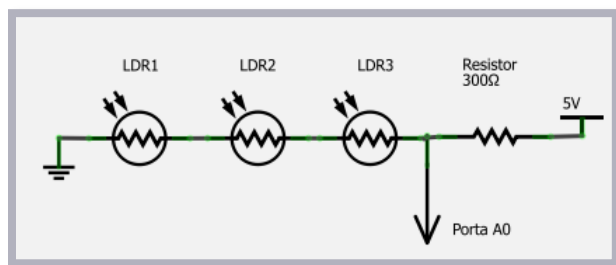


Figura 15 – Esquema elétrico de três LDRs em série.

A porta A0 lê a diferença de potencial nos LDR3, com este valor estabelecemos uma comparação que nos permite dizer se é claro, nublado ou escuro.

Código

```
int resistor; // cria variável resistor
char* myStrings[ ]={"escuro", "encoberto", "claro"}; //
cria uma matriz de caracteres onde o valor 0 é a palavra
escuro, 1 encoberto e 2 claro
int luz; // a variável luz

void setup() {
  pinMode(A0, INPUT); //Define A0 como uma porta de entrada
de valores.O A0 representa porta analógica 0
```

```

Serial.begin(9600); // Ativa a impressão de valores na tela do
micro, 9600 é a velocidade de transferência

}

void loop() {
  resistor=analogRead(A0); // Lê o valor da porta A0 e
armazena na variável resistor

  if (resistor <= 10){ // se o valor da variável for menor que 10,
variável luz igual a 0
  luz=0;
  }

  if (resistor > 10 && resistor < 400){ // se o valor da variável
for entre 10 e 400, variável luz igual a 1

  luz=1;
  }

  if (resistor >= 400){ // se o valor da variável for maior que
400, variável luz igual a 2

  luz=2;
  }

  Serial.print(resistor); // imprime o valor da variável
resistência para calibração das funções if
  Serial.print(" "); // imprime um espaço
  Serial.print(myStrings[luz]); // imprime a matriz de
caracteres escuro, encoberto e claro
  Serial.println(" "); // pula para próxima linha
  delay(500); // dá um tempo de 500
}

```

Montagem:

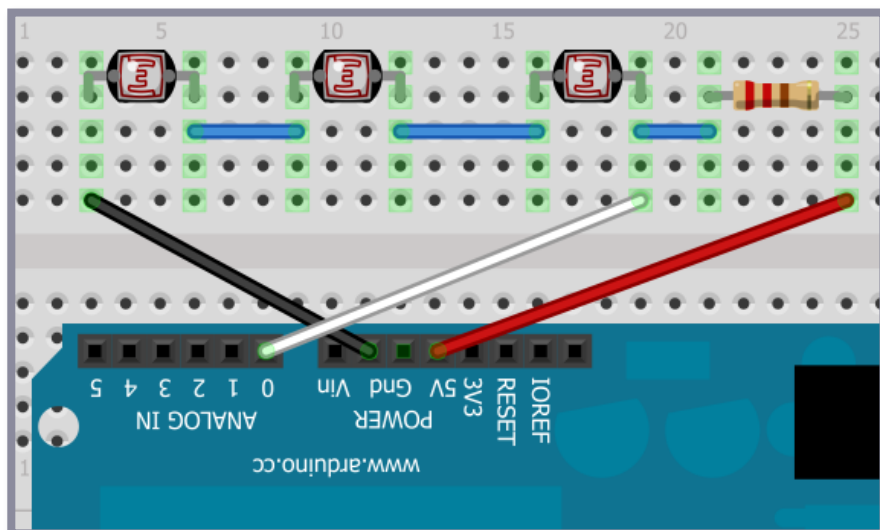


Figura 16 – Montagem do LDRs no *Protoboard* para determinação da incidência da luz no transcorrer do *dia*.

2.3 – HIH-4000-001: Sensor para umidade

O sensor HIH-400-001 é um sensor linear de umidade relativa do ar, pouco comum no mercado brasileiro. Quanto maior a umidade maior é a tensão de saída.

Umidade relativa do ar, em termos simplificados, o quanto de água na forma de vapor existe na atmosfera no momento em relação ao total máximo que poderia existir, na temperatura observada.

Material:

- *Protoboard*
- *Arduino*
- HIH-4000-001⁵

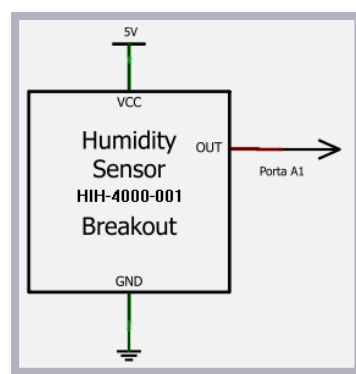


Figura 17 – Esquema elétrico do sensor de umidade

Sensor de fácil montagem pode ser conectado diretamente no Arduino e sua saída é linear. Com 0 de umidade o sensor acusa uma tensão de 0,826V, e a cada 0,0314V equivale a um ponto percentual de umidade relativa. Então 10% de umidade acusariam no sensor 1,14V ($1,14V = 0,826 + 0,0314 \times 10\%$).

Código:

```
float sensor; // declara a variável sensor
float cont;   // declara a variável cont

void setup(){
  pinMode(A1, INPUT); // ativa a porta analógica A1 como
  entrada de dados
  Serial.begin(9600); // ativa a porta serial
}

void loop(){
  sensor = analogRead(A1); // lê o valor da porta A1 e
  armazena na variável sensor
```

⁵ Datasheet do HIH-4000-001 em <http://pdf1.alldatasheet.net/datasheet-pdf/view/227310/HONEYWELL/HIH-4000-001.html>

```

cont=(sensor*5)/1023; // converte os valor em volts
cont= cont-0.826; // desconto de tensão, pois 0 de
umidade relativa corresponde a 0,826v na saída do sensor
cont=cont/0.031483; // cada um ponto percentual de
umidade relativa é igual a 0,031483V

Serial.print(cont); // imprime o valor de cont
Serial.println("%"); // imprime % e pula a linha
delay(250); // dá um tempo de 0,25s
}
    
```

Montagem:

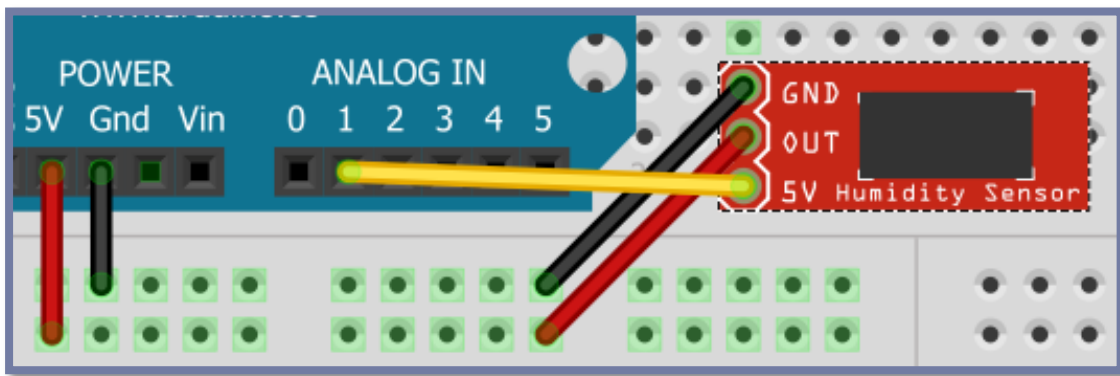


Figura 18– Montagem do sensor de umidade no *Protoboard*

2.4 – Reed switch: Sensor para Biruta

O Reed switch funciona como uma chave magnética, que fecha com a aproximação de um ímã e a abre quando este se afasta. Largamente utilizado em alarmes de portas e janelas.

Material:

- 8 Resistor de 1kΩ
- *Protoboard*
- Arduino
- 8 Reed Switch⁶
- Ímã

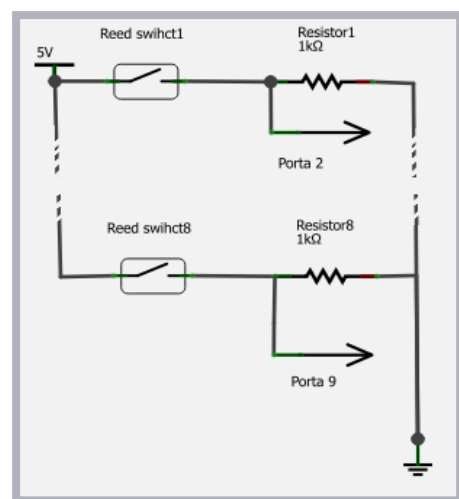


Figura 19 – Esquema elétrico das ligações dos Reed switch para fazer a biruta.

⁶ Datasheet do Read Switch em <http://pdf1.alldatasheet.net/datasheet-pdf/view/319728/MEDER/ORD213.html>

Para construir uma biruta podemos utilizar oito chaves magnéticas (reed switches). As chaves foram dispostas de acordo com os pontos cardeais principais e os secundários sobre um disco circular de 30cm de diâmetro. Sobre o disco, preso a um eixo vertical, está uma biruta que pode girar livremente de acordo com a direção do vento. No braço da biruta está fixo um ímã que ativa as chaves magnéticas quando passa por sobre elas enquanto a biruta gira.

Biruta é um equipamento que serve para indicar a direção do vento. As birutas que encontramos no dia-a-dia também nos fornecem a intensidade do vento. Quanto maior a **velocidade** do vento mais reto fica o cone de tecido que a constitui.

Velocidade relaciona a variação da posição de um ponto ou corpo em relação ao tempo. Podemos dizer que é a distância percorrida por um corpo num determinado intervalo temporal.

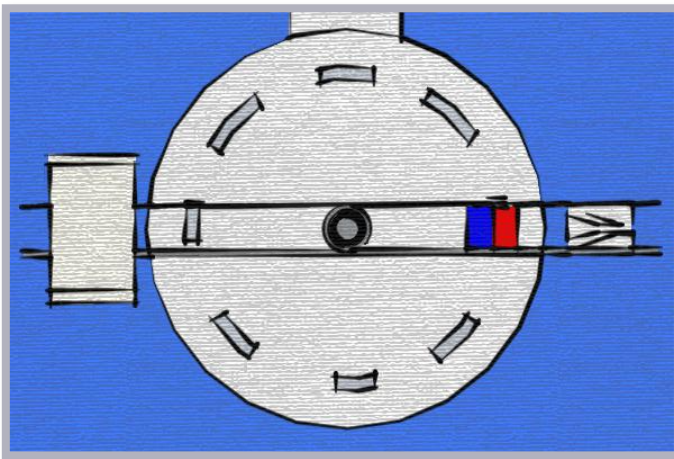


Figura 20– Desenho da montagem da biruta. Os Reeds switches são oitos e ficam posicionados de acordo com pontos cardeais. Sobre eles gira um eixo com um ímã acoplado.

Código:

```
int N; // declara a variável N
int posicao; // declara a variável posição

void setup() {
  pinMode(2, INPUT); // declara que as portas 2, 3, 4, 5, 6,
  7, 8 e 9 são de entradas de dados
  pinMode(3, INPUT);
  pinMode(4, INPUT);
  pinMode(5, INPUT);
  pinMode(6, INPUT);
  pinMode(7, INPUT);
  pinMode(8, INPUT);
  pinMode(9, INPUT);
  Serial.begin(9600); // ativa a porta serial
}
```

```

void loop() {

    digitalWrite(2, LOW);    // declara que o estado inicial das
    portas 2, 3, 4, 5, 6, 7, 8 e 9 é baixo, sem tensão
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
    digitalWrite(5, LOW);
    digitalWrite(6, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);

    posicao = digitalRead(2);    // faz a leitura da porta e
    armazena o valor na variável posição
    if ( posicao == HIGH) // se o estado da variável posição for
    alto ativa a condição
    {
        N=0; // a variável N indica a direção em graus, 0 é norte.
    }

    posicao = digitalRead(3);    // faz a leitura da porta e
    armazena o valor na variável posição
    if ( posicao == HIGH) // se o estado da variável posição for
    alto ativa a condição
    {
        N=45; // a variável N indica a direção em graus, 45 é
    nordeste
    }

    posicao = digitalRead(4);    // faz a leitura da porta e
    armazena o valor na variável posição
    if ( posicao == HIGH) // se o estado da variável posição for
    alto ativa a condição
    {
        N=90; // a variável N indica a direção em graus, 90 é leste.
    }

    posicao = digitalRead(5);    // faz a leitura da porta e
    armazena o valor na variável posição
    if ( posicao == HIGH) // se o estado da variável posição for
    alto ativa a condição
    {
        N=135; // a variável N indica a direção em graus, 135 é
    sudeste
    }

    posicao = digitalRead(6);    // faz a leitura da porta e
    armazena o valor na variável posição
    if ( posicao == HIGH) // se o estado da variável posição for
    alto ativa a condição
    {
        N=180; // a variável N indica a direção em graus, 180 é sul
    }
}

```

```

posicao = digitalRead(7);    // faz a leitura da porta e
armazena o valor na variável posição

if ( posicao == HIGH)    // se o estado da variável posição
for alto ativa a condição

{
    N=225;    // a variável N indica a direção em graus, 225 é
sudoeste
}
posicao = digitalRead(8);    // faz a leitura da porta e
armazena o valor na variável posição

if ( posicao == HIGH)    // se o estado da variável posição
for alto ativa a condição
{
    N=270;    // a variável N indica a direção em graus, 270 é
oeste
}

posicao = digitalRead(9);    // faz a leitura da porta e
armazena o valor na variável posição
if ( posicao == HIGH)    // se o estado da variável posição
for alto ativa a condição
{
    N=315;    // a variável N indica a direção em graus, 315 é
noroeste
}

}

Serial.println(N); // imprime o valor da variável N
delay(500);    //dá um tempo de 0,5s
}
    
```

Montagem:

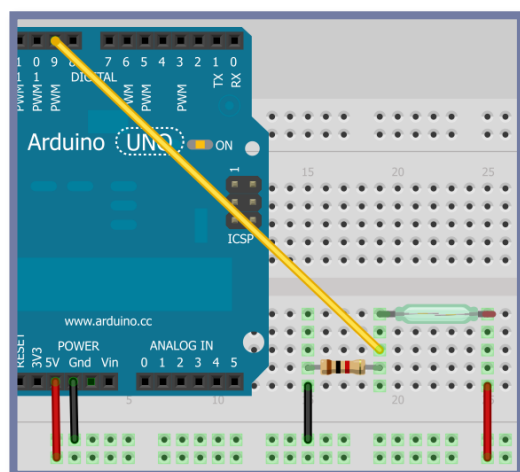


Figura 21 –Montagem da biruta. Por conveniência foi mostrado apenas um *Reed swihcts*, mas são 8 no total e cada um se liga a uma porta digital distinta.

2.5 – Fotodiodo: Usado como sensor para determinar a velocidade do vento

O fotodiodo funciona com uma chave, quando exposto a luz infravermelha conduz. Estas mudanças de estado podem ser lidas pelo Arduino.

Material:

- *Protoboard*
- Arduino
- 3 Fotodiodos⁷
- 3 LEDs infravermelho⁸
- 3 Resistores de 300Ω

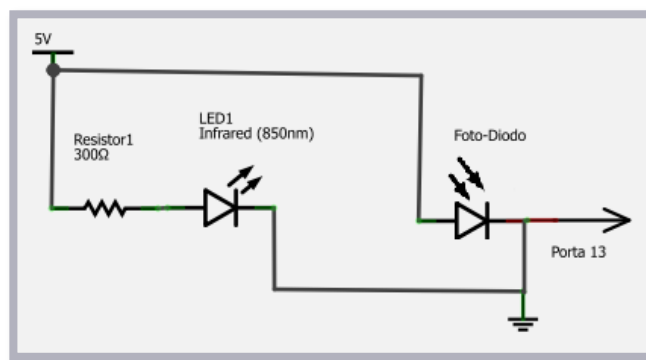


Figura 22 – Esquema de ligação do fotodiodo.

Para medir a intensidade do vento podemos utilizar três emissores e três receptores de infravermelho, tipo fotodiodos. Construímos um sistema com três pás em ângulos de 120° que giram livremente na horizontal. Conforme giram as hastes das pás, estas bloqueiam o emissor de infravermelho e o receptor muda de estado. O tempo de mudança do estado é medido e com isso se obtém a velocidade angular das pás e a partir da qual podemos determinar a intensidade do vento. A montagem proposta é mostrada na figura 23.

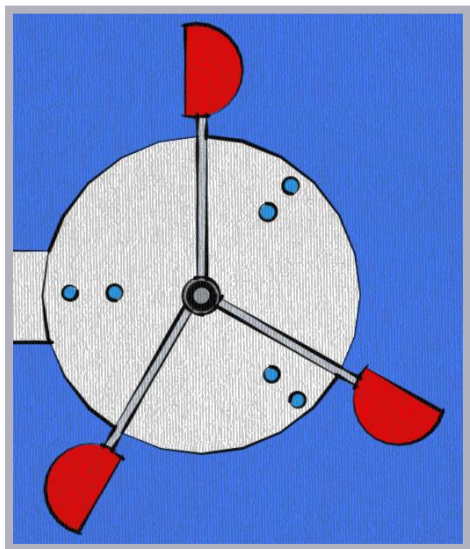


Figura 23- Imagem da montagem do sistema para determinar a intensidade do vento

Velocidade angular tem uma definição análoga ao conceito de velocidade, mas relaciona variação angular em relação ao tempo. A velocidade angular descreve a velocidade de uma rotação.

⁷ Datasheet do fotodiodo em <http://pdf1.alldatasheet.com/datasheet-pdf/view/45628/SIEMENS/SFH229.html>

⁸ Datasheet do LED infravermelho em <http://pdf1.alldatasheet.com/datasheet-pdf/view/219282/EVERLIGHT/IR383.html>

Código:

```

float velocidade; //declara variável do tipo fracionaria
int ini;
int fim;
int angulo;
int sinal;
long int tempo; //declara variável do tipo inteira com
capacidade para armazenar números muitos grandes

void setup(){

    pinMode(13,INPUT); //declara que a porta é de entrada
    Serial.begin(9600); //ativa a porta serial

}

void loop(){
    angulo= 0; // zera a variável ângulo
    ini=millis(); // função armazena o tempo atual na variável
ini, definindo o tempo inicial
    for (int i=1; i <= 100; i++) // repete o processo abaixo 100
vezes
    {
        sinal=digitalRead(13); //lê o valor da porta e armazena
na variável
        if (sinal == LOW) //se variável tiver valor baixo ativa o if
        {
            angulo= angulo + 1; //adiciona +1 na variável ângulo
            delay(200); // dá um tempo de 0,2s
        }
        delay(10); // dá um tempo de 0,01s
    }
    fim=millis();// armazena o valor atual do tempo na variável
ini, definindo o tempo final
    tempo= fim - ini; // calcula o intervalo de tempo
    velocidade= (angulo/tempo)*1000; // ângulo dividido
pelo tempo em milissegundo

    Serial.println(velocidade); // imprime o valor da
velocidade

}

```

Montagem:

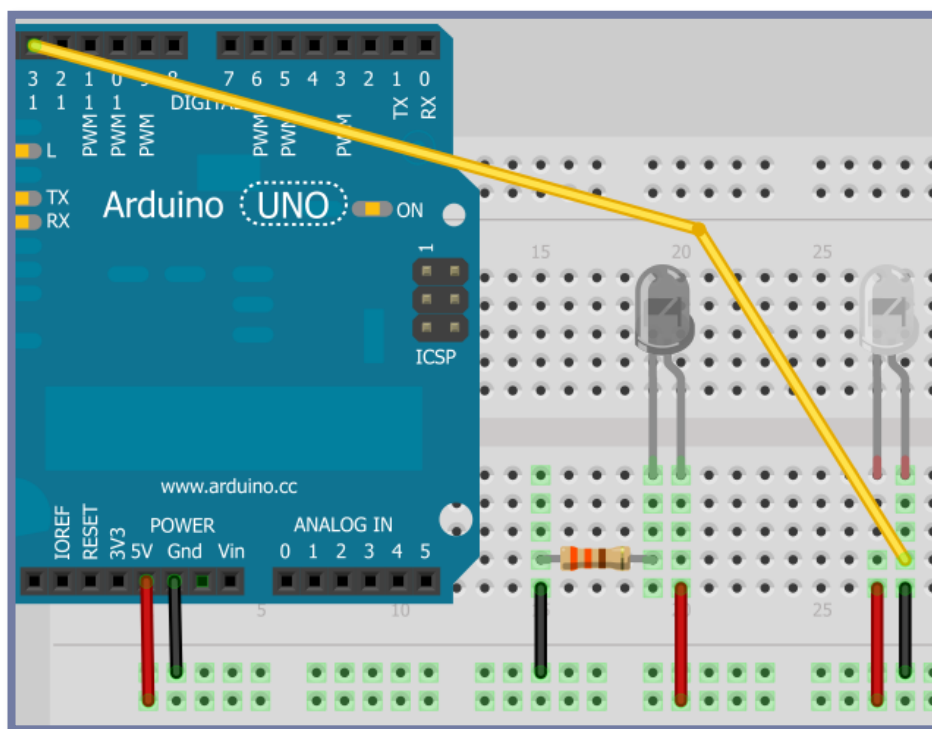


Figura 24 – Esquema da montagem. O sensor cinza escuro é o emissor e cinza claro o receptor

2.6 – BMP085: Sensor para pressão

O sensor BMP085 é bem preciso, largamente utilizado na aviação comercial como altímetro. Além de medir a pressão permite medir a temperatura. Outra vantagem na utilização deste sensor com o Arduino é que já existe uma biblioteca pronta, não havendo necessidade de programação. A biblioteca do BMP085 para Arduino retorna os valores de pressão e temperatura. A montagem para esta aplicação é mostrada na figura 25.

Material:

- Protoboard
- Arduino
- BMP085⁹

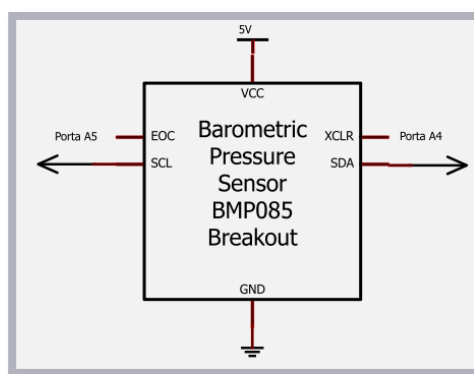


Figura 25 – Esquema de ligação do sensor de pressão

⁹ Datasheet do sensor BMP085 em http://www.adafruit.com/datasheets/BMP085_DataSheet_Rev.1.0_01July2008.pdf

Para podermos medir a pressão atmosférica e a temperatura devemos ligar o sensor BMP085 como mostra a figura 25 ou 26.

Pressão atmosférica é a pressão exercida pela camada de moléculas de ar sobre a superfície. A pressão é a força exercida por unidade de área, neste caso a força exercida pelo ar em um determinado ponto da superfície. Essa pressão pode mudar de acordo com a variação de altitude, ou seja, quanto maior a altitude menor a pressão e, conseqüentemente, quanto menor a altitude maior a pressão exercida pelo ar na superfície terrestre.

Montagem:

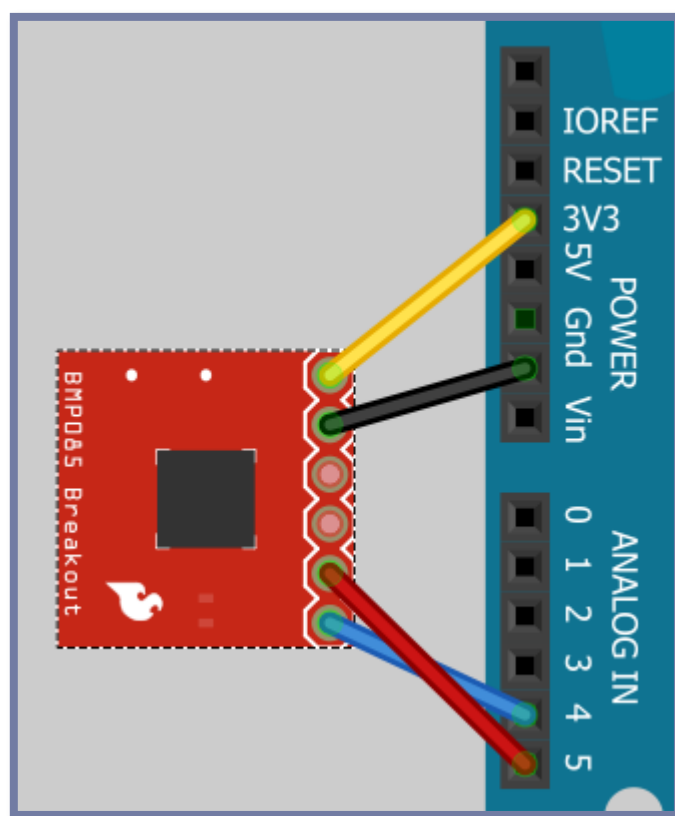


Figura 26 – Esquema de ligação do sensor BMP085 no Arduino

Arduino e o Aluno

A maioria dos educandos quando entra em contato com o Arduino normalmente não possui conhecimento algum de eletrônica. Então, espera-se que este conhecimento seja adquirido primeiro para que possamos trabalhar, porém não há esta necessidade. Podemos começar com uma breve explicação de conceitos como diferença de potencial elétrico (tensão), corrente elétrica, resistores e código de cores. Ao longo das atividades esses conceitos e outros poderão ser aprofundados.

Os comandos de programação do Arduino são muito simples, de modo que os alunos podem aprender o seu uso através de sua aplicação em exemplos simples por tentativa e erro, ainda que se recomende o uso das melhores práticas de programação, como o uso de fluxograma,

O exemplo 1 (pág. 7) permite que o aluno manipule a frequência de acendimento de um LED. Com esta atividade podemos explicar as três partes que compõem a estrutura de programação: declarar variáveis, configurar as portas e a programação. Outra atividade importante com este exemplo é fazer com que o aluno manipule o programa para acender múltiplos LEDs, como construir um semáforo, por exemplo.

Os exemplos 2 (pág. 9) e 3 (pag 11) permitem o aluno exercitar leitura, ativação ou controle de diferentes componentes como, potenciômetros, LEDs, lâmpadas e motores DC.

Com a prática desses três exemplos e sua manipulação, espera-se que a atividade com Arduino se desenvolva sem muitas complicações. Sempre estimule o aluno a pesquisar outras soluções para os desafios ou problemas propostos e a trabalhar em grupo.

REFERÊNCIAS BIBLIOGRÁFICAS

ARDUINO - <http://arduino.cc/>

DATA SHEETS - <http://www.alldatasheet.com/>

LABORATÓRIO DE GARAGEM - <http://labdegaragem.com/>

MCROBERTS, M. *Arduino Básico*, São Paulo, Novatec, 2011

PROCESSING -

[http://pt.wikipedia.org/wiki/Processing_\(linguagem_de_programação\)](http://pt.wikipedia.org/wiki/Processing_(linguagem_de_programação))

SOUZA, A, R. PAIXÃO, A, C. UZÊDA, D, D. DIAS, M, A. DUARTE, S. AMORIM, H, S. *A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC*. Revista Brasileira de Ensino de Física, v. 33, n. 1, 1702 (2011)

UNESCO/COL. *Open Educational Resources and Change in Higher Education: Reflections from Practice*. Commonwealth of Learning: Vancouver, 2012

WIRING - <http://pt.wikipedia.org/wiki/Wiring>

TEXTOS DE APOIO AO PROFESSOR DE FÍSICA

Disponíveis em: http://www.if.ufrgs.br/ppgenfis/mostra_ta.php

- nº. 1 Um Programa de Atividades sobre de Física para a 8ª Série do 1º Grau
Rolando Axt, Maria Helena Steffani e Vitor Hugo Guimarães, 1990.
- nº. 2 Radioatividade
Magale Elisa Brückmann e Susana Gomes Fries, 1991.
- nº. 3 Mapas Conceituais no Ensino de Física
Marco Antonio Moreira, 1992.
- nº. 4 Um Laboratório de Física para Ensino Médio
Rolando Axt e Magale Elisa Brückmann, 1993.
- nº. 5 Física para Secundaristas – Fenômenos Mecânicos e Térmicos
Rolando Axt e Virgínia Mello Alves, 1994.
- nº. 6 Física para Secundaristas – Eletromagnetismo e Óptica
Rolando Axt e Virgínia Mello Alves, 1995.
- nº. 7 Diagramas V no Ensino de Física
Marco Antonio Moreira, 1996.
- nº. 8 Supercondutividade – Uma proposta de inserção no Ensino Médio
Fernanda Ostermann, Letície Mendonça Ferreira, Claudio de Holanda Cavalcanti, 1997.
- nº. 9 Energia, entropia e irreversibilidade
Marco Antonio Moreira, 1998.
- nº. 10 Teorias construtivistas
Marco Antonio Moreira e Fernanda Ostermann, 1999.
- nº. 11 Teoria da relatividade especial
Trieste Freire Ricci, 2000.
- nº. 12 Partículas elementares e interações fundamentais
Fernanda Ostermann, 2001.
- nº. 13 Introdução à Mecânica Quântica. Notas de curso
Ileana Maria Greca e Victoria Elnecape Herscovitz, 2002.
- nº. 14 Uma introdução conceitual à Mecânica Quântica para professores do ensino médio
Trieste Freire Ricci e Fernanda Ostermann, 2003.
- nº. 15 O quarto estado da matéria
Luiz Fernando Ziebell, 2004.
- v. 16, n. 1 Atividades experimentais de Física para crianças de 7 a 10 anos de idade
Carlos Schroeder, 2005.
- v. 16, n. 2 O microcomputador como instrumento de medida no laboratório didático de Física
Lucia Forgiarini da Silva e Eliane Angela Veit, 2005.
- v. 16, n. 3 Epistemologias do Século XX
Neusa Teresinha Massoni, 2005.

- v. 16, n. 4 Atividades de Ciências para a 8a série do Ensino Fundamental: Astronomia, luz e cores
Alberto Antonio Mees, Cláudia Teresinha Jraige de Andrade e Maria Helena Steffani, 2005.
- v. 16, n. 5 Relatividade: a passagem do enfoque galileano para a visão de Einstein
Jeferson Fernando Wolff e Paulo Machado Mors, 2005.
- v. 16, n. 6 Trabalhos trimestrais: pequenos projetos de pesquisa no ensino de Física
Luiz André Mützenberg, 2005.
- v. 17, n. 1 Circuitos elétricos: novas e velhas tecnologias como facilitadoras de uma
aprendizagem significativa no nível médio
Maria Beatriz dos Santos Almeida Moraes e Rejane Maria Ribeiro-Teixeira, 2006.
- v. 17, n. 2 A estratégia dos projetos didáticos no ensino de física na educação de jovens e adultos
(EJA)
Karen Espindola e Marco Antonio Moreira, 2006.
- v. 17, n. 3 Introdução ao conceito de energia
Alessandro Bucussi, 2006.
- v. 17, n. 4 Roteiros para atividades experimentais de Física para crianças de seis anos de idade
Rita Margarete Grala, 2006.
- v. 17, n. 5 Inserção de Mecânica Quântica no Ensino Médio: uma proposta para professores
Márcia Cândida Montano Webber e Trieste Freire Ricci, 2006.
- v. 17, n. 6 Unidades didáticas para a formação de docentes das séries iniciais do ensino
fundamental
Marcelo Araújo Machado e Fernanda Ostermann, 2006.
- v. 18, n. 1 A Física na audição humana
Laura Rita Rui, 2007.
- v. 18, n. 2 Concepções alternativas em Óptica
Voltaire de Oliveira Almeida, Carolina Abs da Cruz e Paulo Azevedo Soave, 2007.
- v. 18, n. 3 A inserção de tópicos de Astronomia no estudo da Mecânica em uma abordagem
epistemológica
Érico Kemper, 2007.
- v. 18, n. 4 O Sistema Solar – Um Programa de Astronomia para o Ensino Médio
Andréia Pessi Uhr, 2007.
- v. 18, n. 5 Material de apoio didático para o primeiro contato formal com Física; Fluidos
Felipe Damasio e Maria Helena Steffani, 2007.
- v. 18, n. 6 Utilizando um forno de microondas e um disco rígido de um computador como
laboratório de Física
Ivo Mai, Naira Maria Balzaretto e João Edgar Schmidt, 2007.
- v. 19, n. 1 Ensino de Física Térmica na escola de nível médio: aquisição automática de dados
como elemento motivador de discussões conceituais
Denise Borges Sias e Rejane Maria Ribeiro-Teixeira, 2008.
- v. 19, n. 2 Uma introdução ao processo da medição no Ensino Médio
César Augusto Steffens, Eliane Angela Veit e Fernando Lang da Silveira, 2008.

- v. 19, n. 3 Um curso introdutório à Astronomia para a formação inicial de professores de Ensino Fundamental, em nível médio
Sônia Elisa Marchi Gonzatti, Trieste Freire Ricci e Maria de Fátima Oliveira Saraiva, 2008.
- v. 19, n. 4 Sugestões ao professor de Física para abordar tópicos de Mecânica Quântica no Ensino Médio
Sabrina Soares, Iramaia Cabral de Paulo e Marco Antonio Moreira, 2008.
- v. 19, n. 5 Física Térmica: uma abordagem histórica e experimental
Juleana Boeira Michelena e Paulo Machado Mors, 2008.
- v. 19, n. 6 Uma alternativa para o ensino da Dinâmica no Ensino Médio a partir da resolução qualitativa de problemas
Carla Simone Facchinello e Marco Antonio Moreira, 2008.
- v. 20, n. 1 Uma visão histórica da Filosofia da Ciência com ênfase na Física
Eduardo Alcides Peter e Paulo Machado Mors, 2009.
- v. 20, n. 2 Relatividade de Einstein em uma abordagem histórico-fenomenológica
Felipe Damasio e Trieste Freire Ricci, 2009.
- v. 20, n. 3 Mecânica dos fluidos: uma abordagem histórica
Luciano Dernadin de Oliveira e Paulo Machado Mors, 2009.
- v. 20, n. 4 Física no Ensino Fundamental: atividades lúdicas e jogos computadorizados
Zilk M. Herzog e Maria Helena Steffani, 2009.
- v. 20, n. 5 Física Térmica
Nelson R. L. Marques e Ives Solano Araujo, 2009.
- v. 20, n. 6 Breve introdução à Física e ao Eletromagnetismo
Marco Antonio Moreira, 2009.
- v. 21, n. 1 Atividades experimentais de Física à luz da epistemologia de Laudan: ondas mecânicas no ensino médio
Lizandra Botton Marion Morini, Eliane Angela Veit, Fernando Lang da Silveira, 2010.
- v. 21, n. 2 Aplicações do Eletromagnetismo, Óptica, Ondas, da Física Moderna e Contemporânea na Medicina (1ª Parte)
Mara Fernanda Parisoto e José Túlio Moro, 2010.
- v. 21, n. 3 Aplicações do Eletromagnetismo, Óptica, Ondas, da Física Moderna e Contemporânea na Medicina (2ª Parte)
Mara Fernanda Parisoto e José Túlio Moro, 2010.
- v. 21, n. 4 O movimento circular uniforme: uma proposta contextualizada para a Educação de Jovens e Adultos (EJA)
Wilson Leandro Krummenauer, Sayonara Salvador Cabral da Costa e Fernando Lang da Silveira, 2010.
- v. 21, n. 5 Energia: situações para a sala de aula
Marcia Frank de Rodrigues, Flávia Maria Teixeira dos Santos e Fernando Lang da Silveira, 2010.
- v. 21, n. 6 Introdução à modelagem científica
Rafael Vasques Brandão, Ives Solano Araujo e Eliane Angela Veit, 2010.
- v. 22, n. 1 Breve introdução à Lei de Gauss para a eletricidade e à Lei de Âmpere-Maxwell
Ives Solano Araujo e Marco Antonio Moreira, 2011.

- v. 22, n. 4 Visões epistemológicas contemporâneas: uma introdução
Marco Antonio Moreira e Neusa Teresinha Massoni, 2011.
- v. 22, n. 5 Introdução à Física das Radiações
Rogério Fachel de Medeiros e Flávia Maria Teixeira dos Santos, 2011.
- v. 22, n. 6 O átomo grego ao Modelo Padrão: os indivisíveis de hoje
Lisiane Araujo Pinheiro, Sayonara Salvador Cabral da Costa e Marco Antonio Moreira, 2011.
- v. 23, n. 1 Situações-problema como motivação para o estudo de Física no 9o ano
Terrimar I. Pasqualetto, Rejane M. Ribeiro-Teixeira e Marco Antonio Moreira, 2012.
- v. 23, n. 3 Universo, Terra e Vida: aprendizagem por investigação
Roberta Lima Moretti, Maria de Fátima Oliveira Saraiva e Eliane Angela Veit, 2012.
- v. 23, n. 4 Ensinando Física através do radioamadorismo
Gentil César Bruscatto e Paulo Machado Mors, 2012.
- v. 23, n. 5 Física na cozinha
Lairane Rekovvsky, 2012.
- v. 23, n. 6 Inserção de conteúdos de Física Quântica no Ensino Médio através de uma unidade de ensino potencialmente significativa
Adriane Griebeler e Marco Antonio Moreira, 2013.
- v. 24, n. 1 Ensinando Física Térmica com um refrigerador
Rodrigo Pogliani e Maria Helena Steffani, 2013.
- v. 24, n. 2 Einstein e a Teoria da Relatividade Especial: uma abordagem histórica e introdutória
Melina Silva de Lima, 2013.
- v. 24, n. 3 A Física dos equipamentos utilizados em eletrotermofototerapia
Alexandre Novicki, 2013.
- v. 24, n. 4 O uso de mapas e esquemas conceituais em sala de aula
Angela Denise Eich Müller e Marco Antonio Moreira, 2013.
- v. 24, n. 5 Evolução temporal em Mecânica Quântica: conceitos fundamentais envolvidos
Glaucio Cohen F. Pantoja e Victoria Elnecave Herscovitz, 2013.
- v. 24, n. 6 Aprendizagem significativa em mapas conceituais
Marco Antonio Moreira, 2013.
- v. 25, n. 1 Introdução ao uso de tecnologias no Ensino de Física experimental dirigida a licenciandos de Física
Leandro Paludo, Eliane Angela Veit e Fernando Lang da Silveira, 2014.
- v. 25, n. 2 Uma proposta para a introdução dos plasmas no estudo dos estados físicos da matéria no Ensino Médio
Luis Galileu G. Tonelli, 2014.
- v. 25, n. 3 Abordagem de conceitos de Termodinâmica no Ensino Médio por meio de Unidades de Ensino Potencialmente Significativas
Marcos Pradella e Marco Antonio Moreira, 2014.