



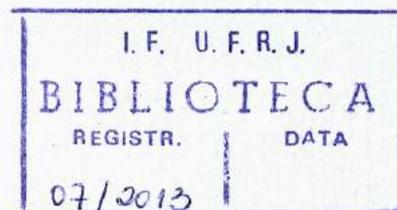
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO – UFRJ

INSTITUTO DE FÍSICA – UFRJ

**MONOGRAFIA DE CONCLUSÃO DO CURSO DE
LICENCIATURA EM FÍSICA**

**EXPERIÊNCIAS COM A PLACA ARDUINO:
CONDUÇÃO DE CALOR**

THIAGO ANTUNES FERREIRA



Rio de Janeiro
Junho de 2013

THIAGO ANTUNES FERREIRA
DRE:110085451

**EXPERIÊNCIAS COM A PLACA ARDUINO:
CONDUÇÃO DE CALOR**

Monografia apresentada ao Curso de Licenciatura em Física, da Universidade Federal do Rio de Janeiro, como requisito parcial à obtenção do título de Licenciado em Física.

Orientador: Hélio Salim

Rio de Janeiro

THIAGO ANTUNES FERREIRA
DRE:110085451

**EXPERIÊNCIAS COM A PLACA ARDUINO:
CONDUÇÃO DE CALOR**

Monografia apresentada ao Curso de Licenciatura em Física, da Universidade Federal do Rio de Janeiro, como requisito parcial à obtenção do título de Licenciado em Física.

COMISSÃO EXAMINADORA

Prof.

Prof.

Prof.

Rio de Janeiro, ____ de _____ de 2013

Dedicatória e agradecimentos

Agradeço,

A Deus;

A minha família, pela assistência e carinho;

Aos meus amigos que sempre me deram força;

A minha namorada Thaís, que sempre esteve ao meu lado;

Ao meu orientador, Hélio Salim, pela ajuda e orientação no meu trabalho;

Aos professores do Instituto de Física que sempre se disponibilizaram a ajudar.

Resumo

No ensino de física, temos a necessidade de realizar experimentos a fim de descobrir, comprovar ou facilitar o aprendizado de determinado fenômeno para os alunos. Devido à falta de práticas experimentais em sala de aula aliado ao baixo aprofundamento teórico disponível nos livros didáticos, encontramos dificuldades em realizar uma prática experimental que ajude a compreensão de fenômenos físicos.

Com a plataforma Arduino temos a opção de desenvolver aplicações de baixo custo, não apenas para o ensino de física, assim como, para as demais áreas do conhecimento.

Por ser um micro controlador de placa única e de código aberto e com um extenso número de usuários que disponibilizam seus códigos de forma livre, o Arduino se torna um incrível instrumento, possibilitando montar um amplo laboratório didático no ensino de física.

Abordaremos neste trabalho a utilização do Arduino como ferramenta didática para o ensino de física e a montagem de um experimento para o tema de condução de calor.

Índice

1. INTRODUÇÃO	7
2. A FÍSICA NO ENSINO MÉDIO	8
3. O ARDUINO	10
3.1 O HARDWARE	11
3.2 O SOFTWARE	12
3.3 APLICAÇÕES	14
4. SENSORES	15
4.1. SENSOR DE TEMPERATURA	16
5. ESTUDO DA TEMPERATURA E DO CALOR	19
5.1. CONDUÇÃO TÉRMICA	21
6. A EXPERIÊNCIA	24
6.1. AQUISIÇÃO E TRATAMENTO DOS DADOS	27
7. COMENTÁRIOS FINAIS	30
REFERÊNCIAS	31
APÊNDICE	33

1. Introdução

Um dos problemas encontrados no ensino de Física é a falta de práticas experimentais em sala de aula e a superficialidade com que alguns temas são abordados. Alguns assuntos são abordados com pouca profundidade ou apresentados em uma seção à parte nos livros didáticos, juntamente com a falta de experiências práticas, tornando o ensino de Física ainda mais abstrato do ponto de vista do aluno. Devido à falta de recursos em sala de aula aliada a pouca literatura disponível, fica difícil para o docente ter alternativas para contornar os problemas na aprendizagem do aluno e ter outros meios de ensino. A partir deste problema, veremos então, como é possível relacionar a teoria com a prática utilizando uma rica ferramenta de baixo custo que possibilita a abordagem, em sala de aula, de diversos temas e não apenas relacionados à Física.

O presente trabalho consiste em uma proposta de uma prática experimental para ser usada em sala de aula abordando o tema transferência de calor utilizando uma placa micro controladora Arduino. Essa proposta tem em vista o fato de que o assunto é pouco abordado em livros didáticos e com pouca, ou nenhuma, prática experimental no Ensino Médio.

Devido à falta de versatilidade e ao alto custo de equipamentos didáticos comerciais, a utilização da Arduino se mostra uma excelente solução para falta de práticas experimentais em sala de aula para diversos temas.

2. A Física no Ensino Médio

O ensino de Física vem sendo abordado de forma abstrata sem correlação com o cotidiano, dando mais ênfase a teoria matemática do que ao conhecimento físico e seus conceitos. Devemos levar em consideração os Parâmetros Curriculares Nacionais para o Ensino Médio (PCNEM, 1999), que a matemática é uma linguagem que busca dar conta de aspectos do real e que é instrumento formal de expressão e comunicação para diversas ciências, sendo assim apenas um instrumento.

Ainda de acordo com os Parâmetros Curriculares Nacionais para o Ensino Médio (PCNEM, 1999) o Ensino de Física no nível médio deve desenvolver a capacidade de investigação física, capacidade de interpretar fatos, fenômenos e processos naturais levando-os a compreensão dos avanços tecnológicos do cotidiano, além de privilegiar a aplicação da teoria na prática. O que vemos hoje no ensino de Física é o contrário do que é esperado pelo PCNEM, além de aulas teóricas com ênfase apenas matemática, não temos práticas experimentais que instigue e possibilite a investigação de determinado fenômeno pelo aluno.

Na análise de dados de trabalhos publicados entre 1992 e 2001, na Revista Brasileira de Ensino de Física (RBEF), no Caderno Brasileiro de Ensino de Física (CBEF) e na Revista a Física na Escola (FnE), Araújo e Abib (2003), declaram que a utilização de atividades experimentais como estratégia de ensino de Física tem sido apontada por professores e alunos como o melhor caminho no ensino de Física, minimizando as dificuldades de aprendizagem e ensino de modo significativo e consistente. Porém este estudo mostra ainda que essas propostas estão longe dos trabalhos realizados na grande maioria das escolas.

Diversos problemas são apontados para a falta de práticas experimentais como a pouca repercussão de novas propostas curriculares, falta de tempo para a prática, além da falta de atividades experimentais. Podemos dizer também que um dos principais obstáculos para utilização de experimentos no ensino de Física, ainda de acordo com Pena e Filho (2009) são, a falta ou carência de pesquisas

sobre o que os alunos realmente aprendem por meio de experimentos, o despreparo do professor para trabalhar com práticas experimentais e a falta de condições de trabalho.

Devido a estes fatores que dificultam a utilização de práticas experimentais, o ensino de Física fica visivelmente prejudicado, pois perde um importante e poderoso meio de investigação e ensino que facilita a aprendizagem do aluno, possibilitando apenas que métodos tradicionais de ensino, que visam apenas a teoria matemática e a resolução de problemas dos livros didáticos, de serem aplicadas em aula.

Este trabalho é uma proposta de prática experimental que vai ao encontro a Leis de Diretrizes e Bases da Educação Nacional (LDB) que visa à compreensão dos fundamentos científico-tecnológicos dos processos produtivos, relacionando a teoria com a prática, no ensino de cada disciplina (Art. 35. IV), e ao PCNEM, pois irá privilegiar a aplicação da teoria na prática, além de promover o desenvolvimento de habilidades e estimular o surgimento de novas aptidões.

3. O Arduino

O projeto Arduino iniciou-se em 2005 na cidade de Ivrea, Itália, com o intuito de criar projetos escolares de baixo custo, com orçamento menor que outros sistemas de prototipagem que utilizavam outros microcontroladores disponíveis na época.

O Arduino é uma plataforma de prototipagem projetada com um micro controlador *Atmel AVR* de placa única, de *hardware* aberto com entradas e saídas embutidas e um conjunto de *software* para programa-lo com uma linguagem de programação própria no qual é essencialmente C/C++. De forma prática, podemos dizer que o Arduino é um pequeno computador que permite programar suas entradas e saídas entre o dispositivo e os componentes externos conectados a ele. Sua utilização é fácil e ideal para a elaboração e criação de dispositivos que permitam a interação com o ambiente que utilizem como entrada sensores de temperatura, luz, som, proximidade etc., tendo como saída leds, motores, displays, autofalantes etc., criando desta forma ilimitadas possibilidades de utilização.

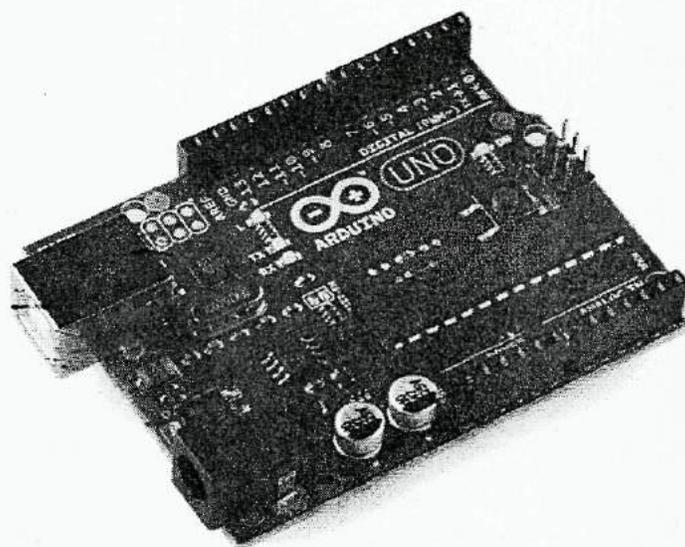


Figura 3.1 – Arduino UNO.

3.1. O hardware

A placa Arduino é composta por um micro controlador *Atmel AVR* de 8 bits e componentes eletrônicos necessários para a comunicação com o micro controlador (regulador, cristal ou oscilador - relógio simples que envia pulsos de tempo em uma frequência especificada que permite a operação na velocidade correta). Modelos antigos do Arduino contam com uma interface serial RS232 enquanto modelos mais novos contam com uma interface USB para conectá-los diretamente a um PC ou Mac. Isso se dá devido à substituição do chip FTDI pelo chip *Atmega8U2* programado como um conversor USB para serial, além de uma interface SPI de programação para substituir o *Bootloader*. A forma padrão de como seus conectores estão dispostos permite integrar outros módulos de expansão a fim de aumentar a capacidade de recursos e comunicação. Estas expansões são conhecidas como *shields* e possibilitam integrar conexões de rede, com e sem fio, GSM, motores, GPS, entre outras funcionalidades adicionais de forma prática. Como o hardware do Arduino é de fonte-aberta, qualquer placa clone é compatível com ele, assim os *shields* também são compatíveis.

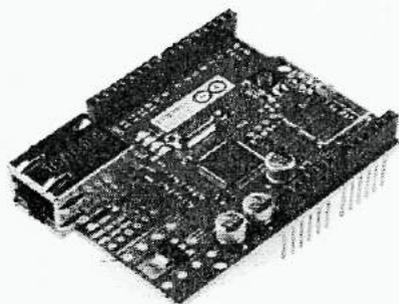


Figura 3.2 – Ethernet Shield.

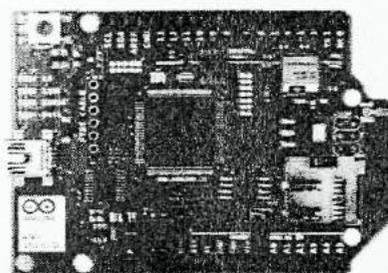


Figura 3.3 – Wifi Shield.

O chip é pré-programado com um *bootloader*, isto é, um gerenciador de inicialização, que facilita o carregamento de programas para sua memória flash integrado, o que dispensa a utilização de um programador externo.

Por ser um *software e hardware* de fonte aberta, seus esquemas, códigos, projetos, etc., podem ser utilizados por qualquer pessoa. Assim, existem no mercado muitas placas clones baseadas no Arduino (freeduino, Seeduino, Brasuíno, Roboduino etc). O nome Arduino não pode ser atribuído a estes clones, pois este é reservado à placa oficial. Como as placas clones são 100% compatíveis com o Arduino, a utilização dos programas, *shields* etc., também são compatíveis.

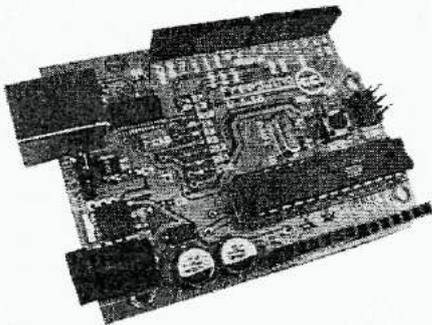


Figura 3.4 – Freeduino.

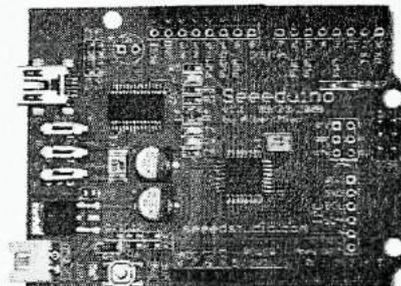


Figura 3.5 – Seeduino.

Existe uma ampla comunidade internacional de utilizadores da placa Arduino extremamente cooperativa, onde as diversas experiências de utilização da placa assim como seus projetos são disponibilizadas de forma livre e gratuita, contando ainda com um fórum disponível em sua página oficial www.arduino.cc, onde podemos obter ajuda dos utilizadores.

O Arduino UNO R3, versão utilizada neste projeto, utiliza o micro controlador Atmega328, possui 32Kb em sua memória flash, sendo 0,5 Kb utilizado pelo *bootloader*, possui 14 entradas/saídas digitais sendo 6 PWM (modulação por largura de pulso), 6 entradas analógicas, um ressonador cerâmico de 16 Mhz (produz oscilações em uma frequência específica), uma conexão USB para ser conectado diretamente a um PC), um conector de alimentação, um

cabeçalho ICSP (pode-se programar o Arduino através do ICSP ignorando o bootloader) , e um botão de reset.

3.2. O software

O Arduino conta com uma interface IDE (*Integrated Development Environment*) de programação, um software livre, baseado em C/C++, utilizado para escrever o código na linguagem que o Arduino “compreende”. Esta interface de programação conta com exemplos de programas prontos que são um conjunto de instruções passo a passo que será enviado ao Arduino. Os programas escritos para Arduino são conhecidos como *sketches* (rascunhos ou esboço). Ainda é possível selecionar o tipo de placa Arduino utilizada, gravar um novo *bootloader* no microcontrolador para ser reconhecido como outro dispositivo no PC, como joystick, mouse etc., fazer o carregamento dos *sketches* para o Arduino e ter uma leitura de dados em um monitor serial, onde é possível selecionar a saída desejada.

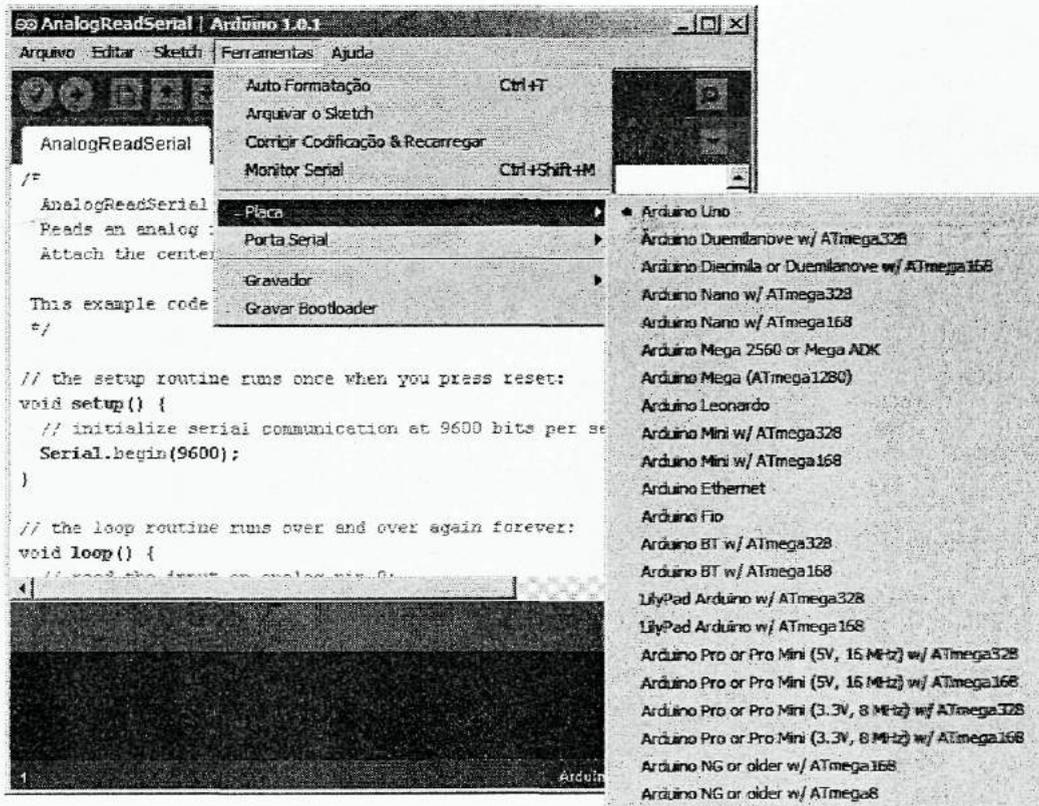


Figura 3.7 – IDE do Arduino.

Podemos ver que a IDE é dividida em partes, o menu principal superior que conta com todas as opções do programa, abaixo um outro menu conhecido como *toolbar* que concentra opções para rápida análise, a janela de código (Sketch window) ao centro, onde os códigos são escritos, e a janela de mensagens na parte inferior informando as mensagens que o Arduino ~~exibe ao~~ conectar a placa, realizar upload do código ou verificá-lo. Temos ainda o monitor serial que é uma ferramenta muito útil para depuração de código e que permite obter leituras diretas enviadas do Arduino, podendo também enviar dados de volta a ele.

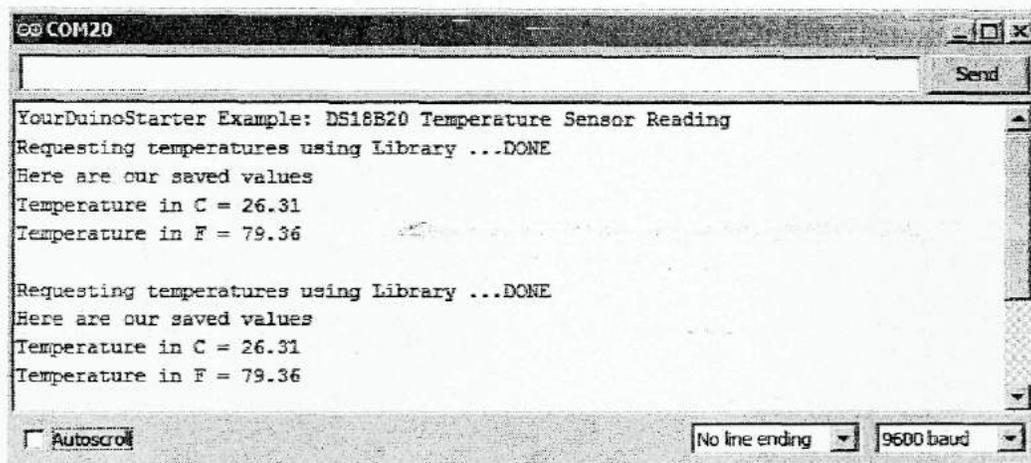


Figura 3.6 – Monitor serial de leitura – IDE Arduino.

3.3. Aplicações

Devido à ampla escala de recursos e suporte disponíveis, podemos fazer uso do Arduino em diversos cenários que envolvem a coleta de dados dinâmica e interação com o ambiente, como exemplo em telecomunicações, na robótica e no ensino de Física e ciências afins. Além da utilização do Arduino para este projeto específico, podemos fazer uso para outros trabalhos utilizando os sensores de distância/proximidade ou acelerômetros no estudo da cinemática, os sensores de força/pressão para trabalhos com a dinâmica, montar pequenos laboratórios de eletricidade utilizando componentes eletrônicos básicos, além de muitas outras aplicações de baixo custo no ensino de Física que antes eram limitados devido aos altos custos dos equipamentos.

4. Sensores

Sensores são dispositivos sensíveis a alguma forma de energia do ambiente, que pode ser mecânica, luminosa, cinética etc., permitindo a análise de uma determinada condição do ambiente. Podemos classificar um sensor como uma forma de transdutor que é o componente que transforma um tipo de energia em outro, no caso específico, transformando algum tipo de energia do ambiente (luz, calor, mecânica) em energia elétrica.

Sensores analógicos são baseados em sinais analógicos, podendo assumir infinitos valores intermediários, entre seus limites de operação, no seu sinal de saída ao longo do tempo.

Sensores digitais baseiam-se em valores discretos bem definidos, podendo ser descritos como alto (High) ou baixo (Low), ou apenas "1" e "0". Tais sensores utilizam a lógica binária para operar.

Como podemos verificar na figura 4.1 o sinal de um sensor analógico em vermelho e o sinal de um sensor digital em preto.

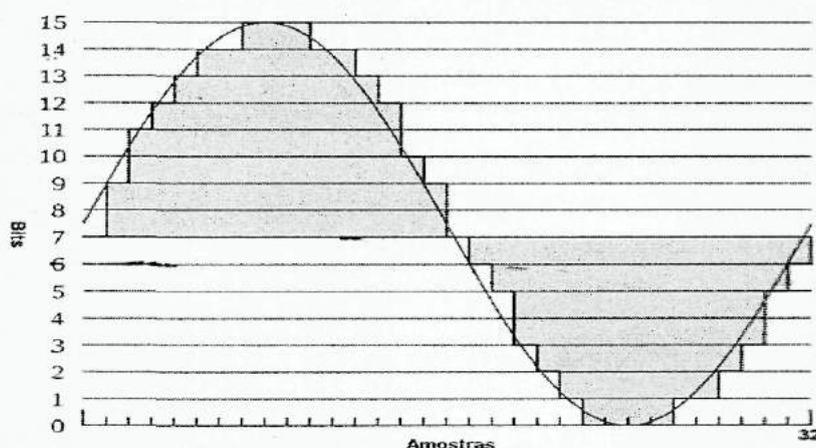


Figura 4.1 – Sinal Analógico e Digital.

Cada aplicação requer uma análise para identificar qual o melhor tipo de sensor a ser utilizado. Para isso, podemos levar em consideração algumas características importantes como os tipos de saída (Digital/Binária, Analógica), a sensibilidade que é a relação de quanto o sinal saída varia em relação ao sinal de entrada, a faixa (*range*) que é o intervalo de valores que a grandeza monitorada pode assumir (o sensor irá responder de acordo com o especificado pelo fabricante), a tolerância que é a discrepância máxima entre o valor real e o valor medido pelo sensor. No caso dos sensores digitais, a resolução que é menor variação da grandeza de entrada que irá gerar uma variação na saída, é um fator importante. Para sensores que fornecem a saída em número binário, a resolução é dada por $1/(2^n)$ em que n é o número de bits que o sensor trabalha.

4.1. Sensor de Temperatura

Sensores de temperatura são amplamente utilizados em indústrias, eletrônicos, veículos e uso doméstico para diversos fins. Tem por sua finalidade “sentir” a temperatura e repassar essa informação a um dispositivo capaz de interpretá-la. Existem vários tipos de sensores de temperatura, entre alguns podemos citar os termistores, que são resistores termicamente sensíveis; o termopar, que é constituído por dois metais que geram uma tensão elétrica em função da temperatura; sensores eletrônicos como diodos e transistores; circuitos integrados como o sensor analógico LM35 e circuitos integrados inteligentes digitais, como o DS1820 que incorpora características dos circuitos integrados e conta ainda com parâmetros programáveis que transformam esse dispositivo em um transmissor inteligente de temperatura.

Para este projeto, dispomos de dois sensores, o circuito integrado LM35, de baixo custo, que fornece um sinal analógico e a tensão de saída é linearmente proporcional à variação de temperatura em Celsius. Porém, sua utilização é comprometida devido a grande quantidade de portas analógicas requeridas pelo micro controlador e por isso, optamos por utilizar o sensor DS18B20 devido à sua versatilidade.

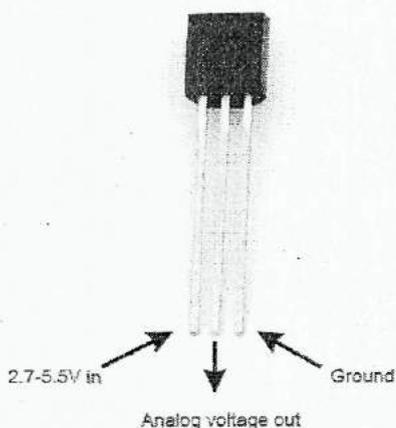


Figura 4.2 – Sensor LM35.

O sensor DS18B20 é um sensor digital que de acordo com seu *datasheet*, é capaz de medir a temperatura em graus Celsius, com resolução configurável de 9 bits a 12 bits, possuindo uma função de alarme programável para valores abaixo ou acima de uma temperatura alvo. A comunicação é feita pela tecnologia *one-Wire* ou *1-Wire*, ou seja, necessitando apenas de um pino digital do microcontrolador para transferir os dados. Opera entre -55°C até $+125^{\circ}\text{C}$ com precisão de $\pm 0.5^{\circ}\text{C}$ (se estiver operando entre -10°C a $+85^{\circ}\text{C}$).

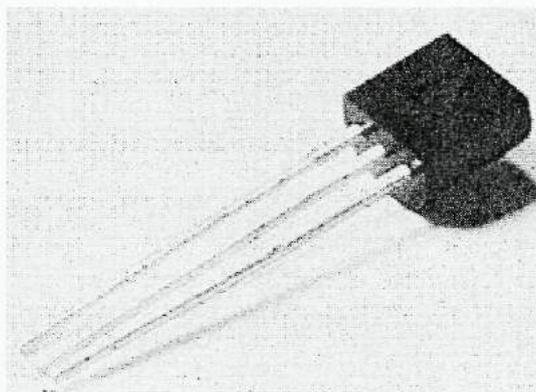


Figura 4.3 – Sensor Digital DS18B20.

O DS18B20 possui um número serial único de 64 bits de identificação, podendo vários DS18B20 funcionar no mesmo barramento *1-Wire*, o que permite conectar diversos sensores em um mesmo microcontrolador e independente da ordem em que eles se encontrem. Por meio do programa de aquisição de dados é possível obter o número serial de cada sensor e ordená-los de acordo com a necessidade do projeto.

Neste projeto iremos utilizar cerca de seis sensores DS18B20 conectados no mesmo barramento, o que impossibilita a utilização de qualquer outro tipo de sensor térmico.

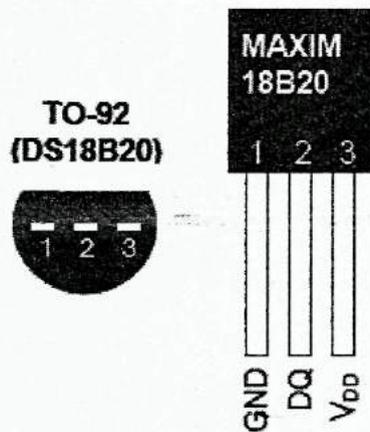


Figura 4.4 – Configuração dos pinos DS18B20.

5. Estudo da temperatura e do calor

Nos estudos relacionados à transferência de calor, devemos considerar os conceitos sobre temperatura, calor e dilatação como pré-requisitos. Para que possamos ter um entendimento sobre o fenômeno físico da condução térmica não basta apenas relatar as equações, devemos entender como ocorre o fenômeno.

Podemos perceber entre dois objetos qual é o mais quente e qual é mais frio pelo contato com eles. A temperatura é uma grandeza física que informa quão quente ou frio está um objeto em relação a algum padrão, assim, podemos dizer que um objeto está com a temperatura mais elevada ou mais baixa que o outro. Para efetuar a medição da temperatura, é necessário um medidor apropriado, um termômetro. Como praticamente todos os objetos se dilatam com a elevação de temperatura e se contraem com a diminuição de sua temperatura, a maioria dos termômetros funcionam por meio da dilatação. Com o aumento de temperatura de um corpo, suas moléculas ou átomos passam a oscilar mais rapidamente, fazendo com que eles, ao vibrar, se afastem mais da posição de equilíbrio, tendo como resultado a dilatação.

Precisamos ter um padrão de medição de temperatura. A escala kelvin é a escala termométrica adotada pelo Sistema Internacional e utilizada pelos físicos. Para a criação de uma escala, precisamos ter um fenômeno térmico reprodutível. Usamos os chamados *pontos de temperatura constante* ao qual atribuímos valores convencionados de temperatura na escala kelvin. Entre os fenômenos reprodutíveis escolhemos, através de acordo internacional, o ponto triplo da água, que é a coexistência de um equilíbrio termodinâmico para a água nos três estados da matéria (sólido, líquido e gasoso). Neste acordo internacional, foi atribuído o valor de 273,16 K ao ponto triplo da água a pressão de 0,001 atm. Um kelvin é assim $1/273,16$ da diferença entre o zero absoluto e a temperatura do ponto triplo.

A escala Celsius é a mais usada no dia-a-dia em quase todos os países do mundo. Ainda por convenção internacional, a escala Celsius é definida a partir da

escala kelvin através da relação $T_c = T_k - 273,15^\circ$, em que T_c representa a temperatura em Celsius e T_k a temperatura em kelvin. Podemos ver pela tabela abaixo a correspondência entre as duas escalas termométricas.

Tabela 5.1

Temperatura	°C	K
Ponto de fusão da água (1 atm)	0,00	273,15
Ponto de ebulição da água (1 atm)	99,975	373,15
Ponto Triplo da Agua	0,01	273,16
Zero Absoluto	-273,15	0

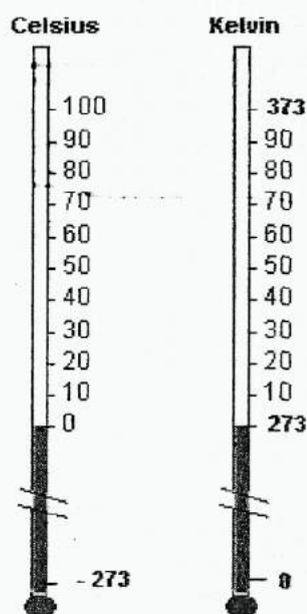
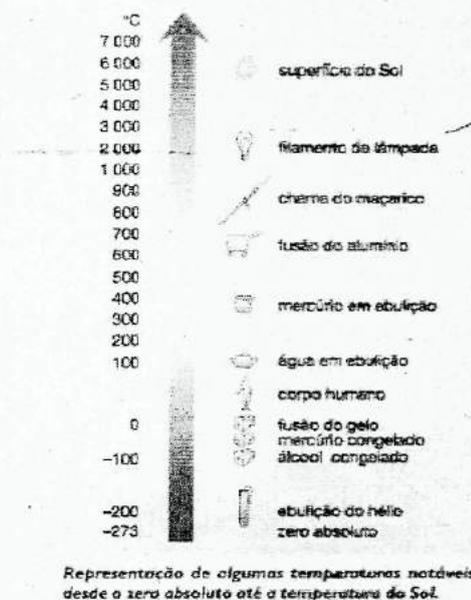


Figura 5.1 – Escala Celsius e kelvin.



Representação de algumas temperaturas notáveis, desde o zero absoluto até a temperatura do Sol.

Figura 5.2 – Temperaturas notáveis.

Quando dois corpos de diferentes temperaturas são postos em contato, após passar determinado tempo eles atingem a mesma temperatura, tal fenômeno ocorre devido à transferência calor e ocorre até os corpos alcançarem o equilíbrio térmico. Assim, o calor é definido como sendo a transferência de energia entre corpos com diferentes temperaturas, do corpo com maior temperatura para o corpo com menor temperatura.

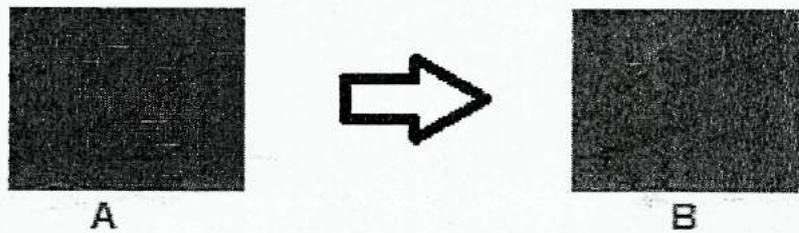


Figura 5.3 – Transferência de energia do corpo A para o corpo B.

Da figura 5.1, podemos entender que o corpo A, que possui maior temperatura, transfere energia para o corpo B, de menor temperatura. Um corpo não possui calor e sim energia interna, quanto maior sua temperatura maior será sua energia interna. O termo calor deve ser usado para designar a energia em trânsito, quando esta é transferida deixa de ser calor. Mesmo sem receber calor a energia de um corpo pode aumentar se o mesmo receber outras formas de energia.

O calor pode ser transferido de um corpo para outro por três meios diferentes que são: a condução, a convecção e a radiação. Neste trabalho estudaremos apenas a transferência por meio de condução, que é ocasionado devido a seu contato.

5.1. Condução térmica

A transferência de calor por meio da condução térmica ocorre através de um meio material, sem que haja movimento do próprio meio, diferentemente da convecção que ocorre devido à movimentação do próprio fluido. Quando colocamos uma agulha de metal no fogo e a seguramos, logo ela esquentará e não será possível segurá-la. O calor penetra na extremidade da agulha e logo é transferida a toda ela. Isto ocorre, pois os átomos da extremidade aquecida oscilam cada vez mais rápido, colidindo assim com os átomos e elétrons livres. Este processo continua até que o aumento no movimento seja transmitido a todos os átomos e o corpo se torne mais quente.

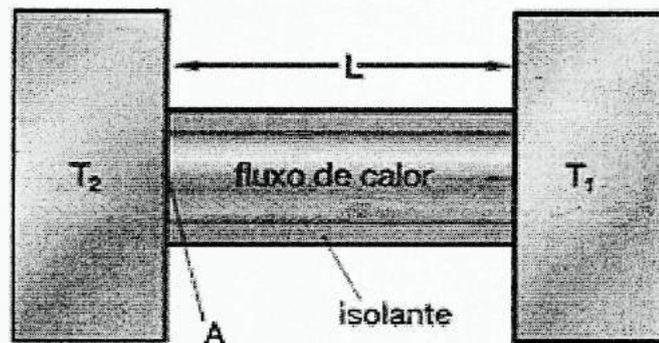


Figura 5.4 – Condução térmica por uma barra ($T_2 > T_1$).

Se considerarmos a montagem experimental indicada na Figura 5.4, a quantidade de calor ΔQ , que passa por uma secção qualquer da barra durante em um intervalo de tempo Δt é chamado de fluxo de calor Φ , assim:

$$\Phi = \frac{\Delta Q}{\Delta t} \quad (1)$$

Caso a barra seja envolvida por um isolante térmico, não permitindo as trocas de calor entre ela e o ambiente, verificaremos que após determinado tempo, ela atingirá uma situação que é denominado estado estacionário.

“Em regime estacionário, o fluxo de calor por condução em um material homogêneo é diretamente proporcional à área da secção transversal atravessada e à diferença de temperatura entre os extremos e inversamente proporcional à espessura da camada considerada.”
(Ramalho, 1987).

Isto ocorre quando a condução térmica é constante, de modo que após um tempo dado, a distribuição espacial de temperatura (gradiente) não se altera mais. Entende-se por gradiente um vetor cuja intensidade é a taxa de variação espacial de uma grandeza, e sua direção e sentido é aquele da maior variação da grandeza no espaço.

Nesse estado, a temperatura em qualquer ponto da barra permanece constante e varia linearmente no espaço, ao longo da direção de transferência de calor. Pela conservação do calor, verificamos que a quantidade de calor que entra na barra é a mesma que sai.

Experimentalmente podemos verificar que o fluxo de calor é:

- Diretamente proporcional à área A da secção reta da barra:

$$\Phi \propto A ; \quad (5.1)$$

- Diretamente proporcional à diferença de temperatura entre os dois mananciais térmicos:

$$\Phi \propto T_2 - T_1 ; \quad (5.3)$$

- Inversamente proporcional ao comprimento L da barra

$$\Phi \propto \frac{1}{L} ; \quad (5.4)$$

Assim, temos que:

$$\Phi = k * A * \frac{T_2 - T_1}{L} \quad (5.5)$$

onde k é uma constante de proporcionalidade característica do meio condutor que se chama *condutividade térmica* do material ($k > 0$). Adicionaremos um sinal negativo na equação (5.5) devido ao fato que o calor flui de temperaturas mais altas para temperaturas mais baixas. Assim a equação (5.5) fica:

$$\Phi = -k * A * \frac{T_2 - T_1}{L} \quad (5.6)$$

A equação (5.6) é uma expressão da Lei de Fourier da condução térmica. Através da Lei de Fourier, podemos quantificar a transmissão de calor entre as duas fontes colocadas nos extremos de uma barra metálica.

No caso em que o material da barra não seja uniforme é necessário escrever a equação (5.6) num ponto definido da barra e para isso temos que considerar a taxa $\Delta T/L$ em termos de limite quando entendemos L como uma variação infinitesimal. Se considerarmos um eixo $-x$ de coordenadas ao longo da barra, cujo sentido é o do fluxo de calor (temperatura mais alta para o de temperatura mais baixa) podemos escrever a lei de Fourier de uma forma mais geral.

$$\Phi = -k * A * \frac{dT}{dx} \quad (5.7)$$

Particularmente, se consideramos o sistema da Figura 5.4 em estado estacionário podemos dizer que:

$$\Phi = \Phi(X) = cte. \Rightarrow \frac{dT}{dx} = b \text{ (cte)}. \quad (5.8)$$

$$T = a + bx \Rightarrow T = T_2 - \frac{T_2 - T_1}{L} * x \quad (5.9)$$

6. A Experiência

A experiência consiste em estudar a transferência de calor por condução através de uma barra metálica entre duas fontes, uma resfriada à temperatura de zero grau celsius e a outra aquecida à temperatura de cem graus celsius.

Os equipamentos necessários estão listados abaixo. São simples, fáceis de serem encontrados e de baixo custo.

A - 1 barra de latão (liga metálica de cobre e zinco);

B - 2 recipientes (fonte aquecida e resfriada);

C - 1 aquecedor elétrico portátil, também conhecido como "rabo quente".

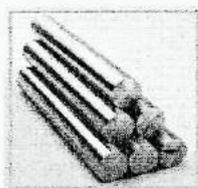
D - 1 termômetro;

E - 6 sensores de temperatura 1- wire digital;

F - 1 matriz de Contatos / 1 Placa Arduino / 1 resistor de 4,7K Ω ;

G - 1 computador (desktop, laptop ou netbook);

H - isopor e fita adesiva.



A



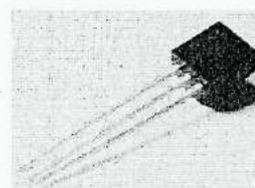
B



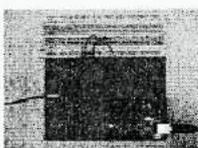
C



D



E



F



G



H

Abaixo temos uma ilustração de como é feita a montagem dos equipamentos.

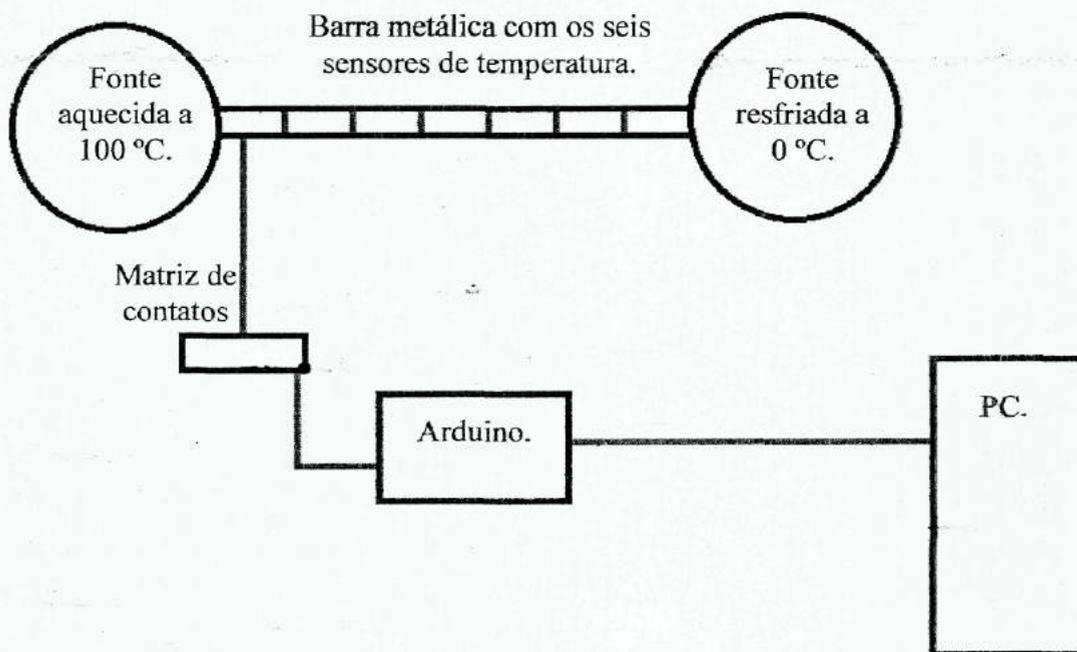


Figura 6.1 - Ilustração experiência.

A montagem é feita acoplando a barra metálica aos dois recipientes e deixando 35,0 cm entre elas. Os sensores de temperatura são colocados a uma distância de 5,0 cm da fonte e de cada sensor. Após esta etapa conectamos os sensores à matriz de contatos como ilustra a figura 6.2.

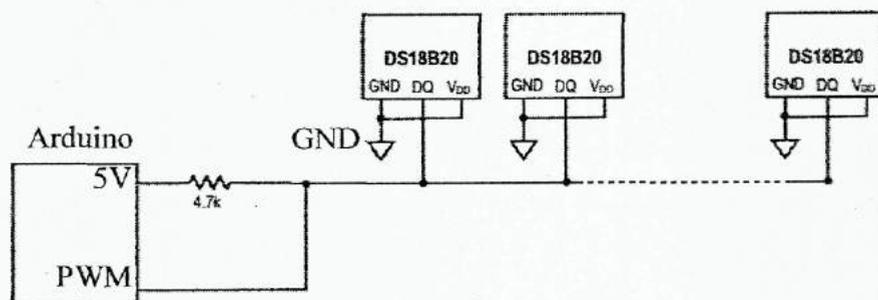


Figura 6.2 – Esquema de conexão dos sensores e a placa Arduino.

A seguir temos a montagem final da experiência em operação.

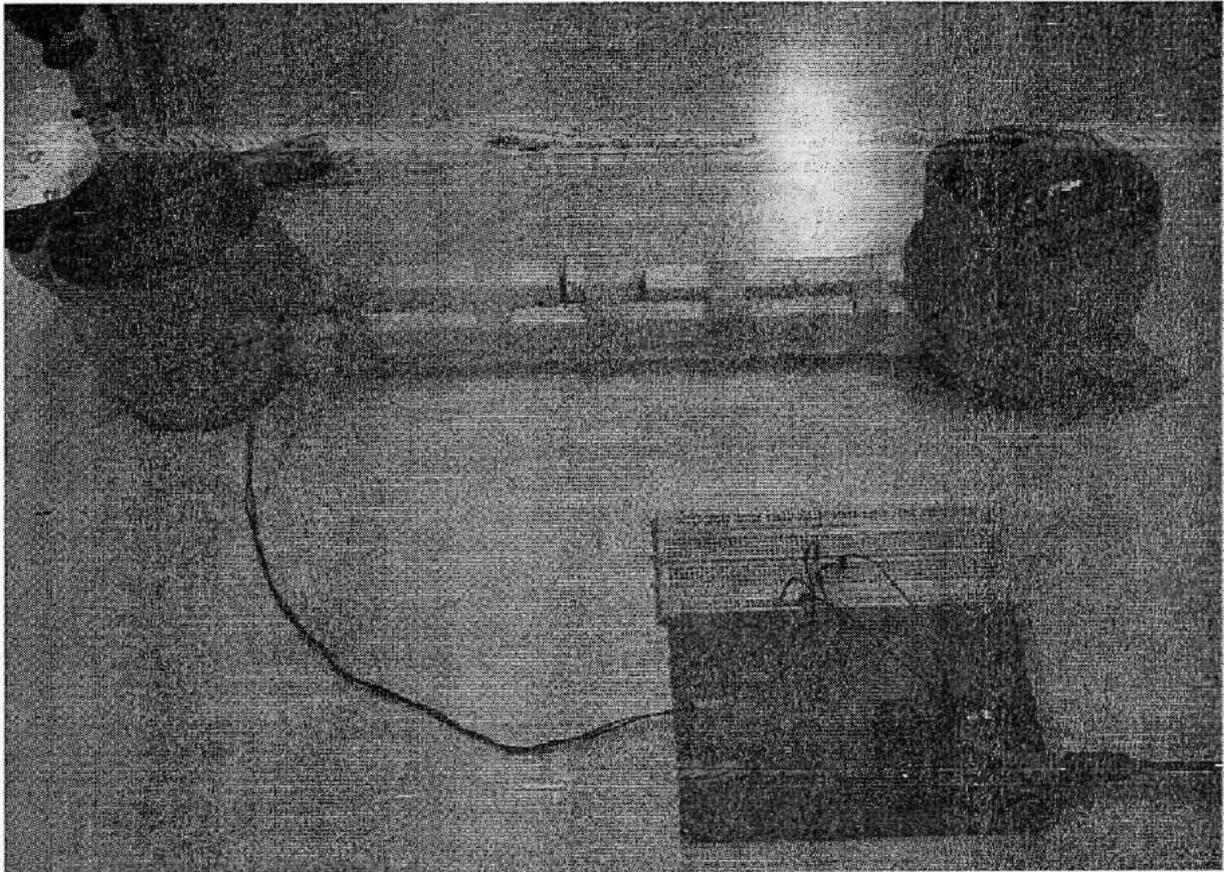


Figura 6.3 – A experiência: sensores (abaixo do isopor) conectados ao Arduino.

Em um primeiro momento, após os equipamentos montados, ligamos a placa Arduino ao computador e executamos um programa de captura de dados da porta serial, a fim de verificar as temperaturas lidas pelos sensores.

Ao decorrer do tempo, percebemos que as temperaturas em cada sensor se altera, se elevando próximo a fonte aquecida e decaindo próximo a fonte resfriada, e aproximadamente após 1 hora, não ocorre mais alteração das temperaturas em cada sensor, chegando então a um estado estacionário.

Neste ponto, o sistema alcançou um estado de condução estacionário após um tempo em equilíbrio, ou seja, a distribuição espacial das temperaturas (o gradiente de temperatura) na barra metálica não mais se altera, permanecendo constante.

6.1. Aquisição e tratamento dos dados.

O *sketch* do Arduino foi construído de tal maneira a enviar as temperaturas lidas pelos sensores para a porta serial separadas por ponto e vírgula e em linha única, o que permite a visualização em um formato compatível com o programa MS Excel®, tornando fácil a sua manipulação. Podemos utilizar também o Openoffice (gratuito e de código aberto) além de outros programas como o Origin® (programa para análise de dados e estatística).

Para a captura de dados da porta serial, utilizamos um programa feito em linguagem BASIC que pode ser executado dentro de um compilador gratuito (LibertyBasic v.4.04 – <http://www.libertybasic.com/>) facilitando quaisquer alterações em seu código. Com ele exportamos os dados para um arquivo em formato texto que pode ser lido pelo programa MS Excel de modo simples. Além deste programa que disponibilizamos o código fonte, podemos utilizar também os diversos programas de código aberto disponibilizados gratuitamente pelo site <http://sourceforge.net/>. Os programas utilizados para coleta de dados da porta serial e o *sketch* do Arduino, estão disponíveis no Apêndice.

Após ter os dados em uma planilha do Excel, construímos os gráficos das temperaturas pelo tempo, onde conseguiremos visualizar o estado estacionário, ou seja, em que a temperatura em cada ponto da barra não se altera com o decorrer do tempo.

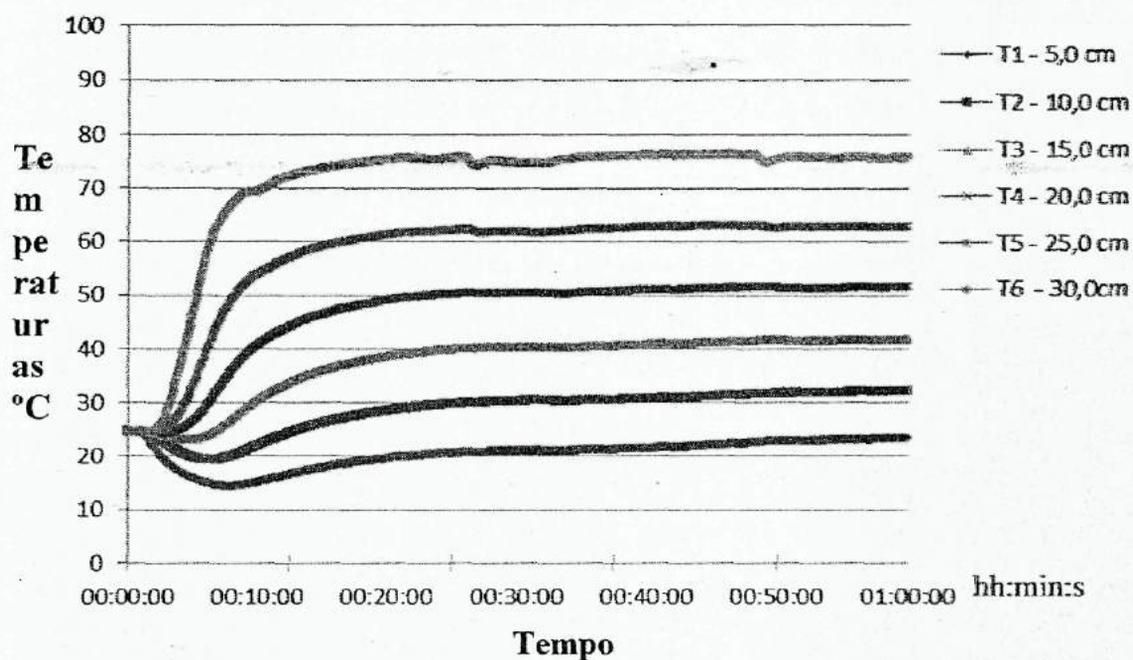


Gráfico 6.1 – Temperatura de cada sensor na barra no decorrer da experiência.

Pelo gráfico podemos verificar o estado estacionário que ocorre após aproximadamente 50 minutos de experiência. A Tabela 1 mostra o tempo decorrido, a temperatura e a distância dos sensores em relação à fonte resfriada.

Tabela 1

Tempo (h:min:s)	Temperatura (°C)	Distância (cm)
00:58:35	23,25	5,0
00:58:35	32,13	10,0
00:58:35	41,69	15,0
00:58:35	51,50	20,0
00:58:35	62,94	25,0
00:58:35	75,69	30,0

De posse das seis temperaturas, faremos um outro gráfico das temperaturas pela distância em que os sensores se situavam na barra de latão.

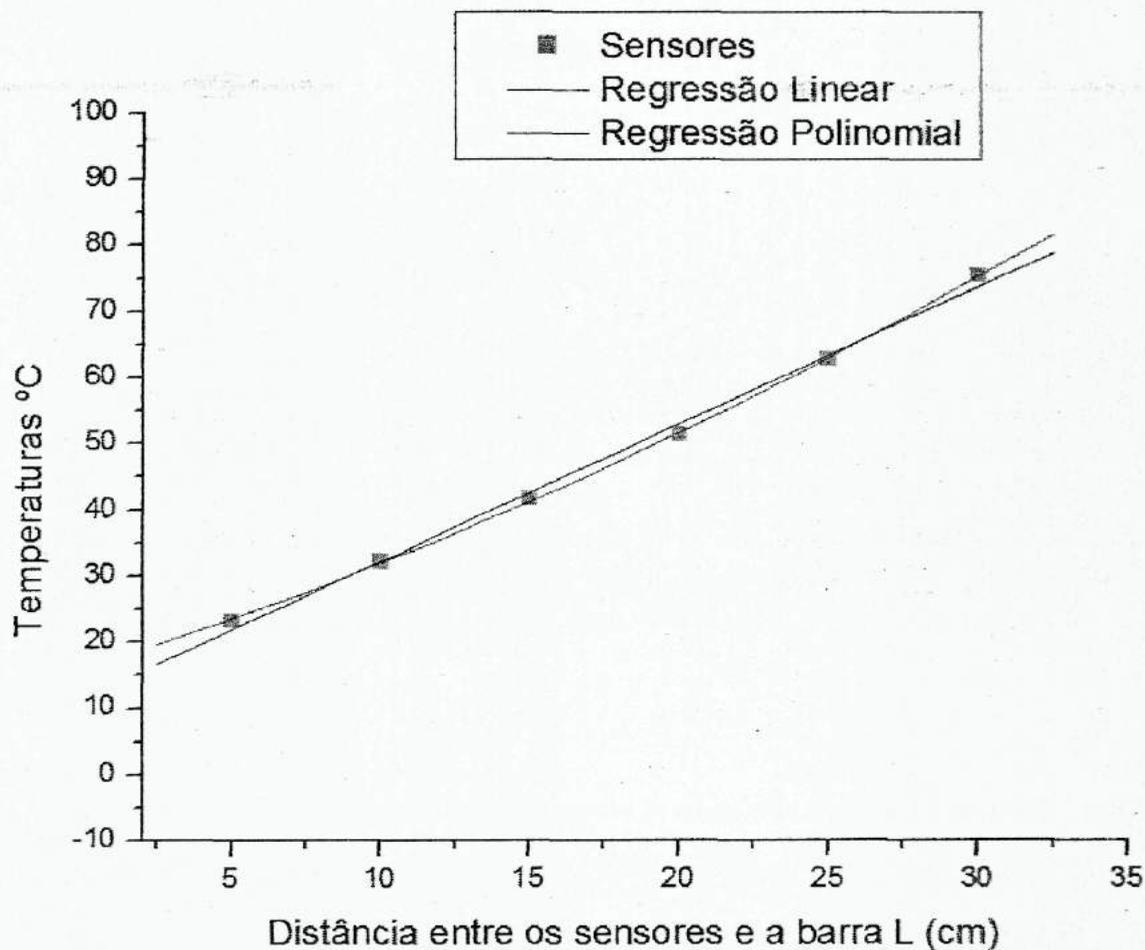


Gráfico 6.2 – Linearidade das Temperaturas.

Podemos verificar que existe uma distribuição uniforme de temperatura como deduzimos mais acima. Verificaremos assim que o fluxo de calor tem o mesmo valor (dentro do erro da experiência) em diferentes posições da barra.

7. Comentários Finais.

Conseguimos realizar uma experiência de transferência de calor por condução utilizando apenas ferramentas de baixo custo. Este projeto mostra como é possível realizar experiências em sala de aula, eliminando problemas como a falta de recursos e compra de equipamentos experimentais. A placa Arduino se mostra uma incrível ferramenta no ensino de Física, tornando possível realizar diversas práticas em sala de aula ou em um laboratório dispendendo de poucos recursos. Permite ao docente planejar uma atividade fazendo conexões com questões do cotidiano, possibilitando ao aluno pensar sobre o fenômeno, medir, controlar grandezas, construir gráficos, tornando-se assim uma ferramenta didática.

Neste trabalho apresentamos uma alternativa à falta de conteúdo e práticas experimental para que os professores possam utilizar de uma prática em sala de aula ou laboratório, que segue um pensamento alinhado as Leis de Diretrizes e Bases da Educação Nacional (LDB), visando à compreensão dos fundamentos científico-tecnológicos dos processos produtivos, fazendo a relação direta da teoria com a prática.

Referências

PENA, F., RIBEIRO FILHO, A.. **Obstáculos para o uso da experimentação no ensino de Física: um estudo a partir de relatos de experiências pedagógicas brasileiras publicados em periódicos nacionais da área (1971-2006)**. Revista Brasileira de Pesquisa em Educação em Ciências, América do Norte, 9, fev. 2011. Disponível em: <http://revistas.if.usp.br/rbpec/article/view/37/33>. Acesso em: 18 Fev. 2013.

ARAÚJO, M. S. T.; ABIB, M. L. V. S. **Atividades experimentais no ensino de Física: diferentes enfoques, diferentes finalidades**. Revista Brasileira de Ensino de Física, São Paulo, v. 25, n. 2, p. 176 – 194, jun. 2003.

BRASIL. Ministério de Educação e Cultura. **LDB - Lei nº 9394/96**, de 20 de dezembro de 1996. Estabelece as diretrizes e bases da Educação Nacional. Brasília: MEC, 1996.

BRASIL. MEC. Secretaria da Educação Média e Tecnológica. **Parâmetros curriculares nacionais: ensino médio (1ª parte)**. Brasília: MEC/Secretaria da Educação Média e Tecnológica, 1999.

Página oficial **Arduino** - <http://www.arduino.cc/> - Acessado entre janeiro de 2012 e julho de 2013.

McRoberts, Michael. **Arduino Básico**. 1º ed. São Paulo: Novatec Editora, 2011.

Anderson R. de Souza, Alexsander C. Paixão, Diego D. Uzêda, Marco A. Dias, Sergio Duarte e Helio S. de Amorim. **A placa Arduino: uma opção de baixo custo para experiências de física** - Rev. Bras. Ensino Fís. vol.33 no.1 São Paulo Jan./Mar. 2011.

Marisa A. Cavalcante, Cristiane R. Caetano Tavoraro e Elio Molisani. Revista Brasileira de Ensino de Física, v. 33, n. 4, 4503 (2011) – **Física com Arduino para iniciantes.**

THOMAZINI, D.; ALBUQUERQUE, P. U. B. **Sensores industriais: fundamentos e aplicações.** São Paulo: Érica, 2005.

DS18B20, **Programmable Resolution 1-Wire Digital Thermometer** - <http://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>

Nussenzveig, H. Moysés. **Curso de Física Básica 2: Fluidos, Oscilações e Ondas, Calor**, 4a edição, Editora Edgard Blücher, 2002.

Portal São Francisco - <http://www.portalsaofrancisco.com.br/> - acessado em janeiro de 2013.

HALLIDAY, D., RESNICK, R., WALKER, J. **Fundamentos de Física 2: Gravitação, Ondas e Termodinâmica.** 6ed. São Paulo: LTC

GASPAR, Alberto. **Física - volume único.** 1ª ed. São Paulo: Ática. 2001.

HEWITT, P. G. **Física Conceitual.** 9. ed., Porto Alegre: Bookman, 2002

NICOLAU, G. F.; TOLEDO, P^a.; RAMALHO JR., F.; IVAN, J. **Os Fundamentos da Física 2 – Termologia, óptica e ondas.** São Paulo, Editora Moderna 1985.

APÊNDICE A – Código fonte “Sketch” Arduino.

```
//Projeto: transferência de calor
#include <OneWire.h>
#include <DallasTemperature.h>
//Linha de dados vai para o pino 3
#define ONE_WIRE_BUS 3
//Prepara uma instância de oneWire para se comunicar com dispositivos OneWire
//(não apenas CIs de temperatura da Maxim/Dallas)
OneWire oneWire(ONE_WIRE_BUS);
//Passa nossa referencia de oneWire para DallasTemperature
DallasTemperature sensors(&oneWire);
// arrays para armazenar os endereços dos dispositivos
DeviceAddress Term1, Term2, Term3, Term4, Term5, Term6;
void setup() {
  //inicia a porta serial
  Serial.begin(9600);
  //inicia a biblioteca
  sensors.begin();
  //localiza os dispositivos no barramento
  Serial.print("Locating devices...");
  Serial.print("Found ");
  Serial.print(sensors.getDeviceCount(),DEC);
  Serial.println(" devices.");
  if(!sensors.getAddress(Term1,0))
  Serial.println("Unable to find address for Device 0");
  if(!sensors.getAddress(Term2,1))
  Serial.println("Unable to find address for Device 1");
  if(!sensors.getAddress(Term3,2))
  Serial.println("Unable to find address for Device 2");
  if(!sensors.getAddress(Term4,3))
  Serial.println("Unable to find address for Device 3");
  if(!sensors.getAddress(Term5,4))
  Serial.println("Unable to find address for Device 4");
  if(!sensors.getAddress(Term6,5))
  Serial.println("Unable to find address for Device 5");
  //imprime os endereços para os quatro dispositivos
```

```

Serial.print("Device 0 Address: ");
printAddress(Term1);
Serial.println();
Serial.print("Device 1 Address: ");
printAddress(Term2);
Serial.println();
Serial.print("Device 2 Address: ");
printAddress(Term3);
Serial.println();
Serial.print("Device 3 Address: ");
printAddress(Term4);
Serial.println();
Serial.print("Device 4 Address: ");
printAddress(Term5);
Serial.println();
Serial.print("Device 5 Address: ");
printAddress(Term6);
Serial.println();
}
//Função para imprimir o endereço de um dispositivo
void printAddress(DeviceAddress deviceAddress)
{ for(int i=0; i<8; i++) {
  //preenche o endereço com zeros, se necessário
  if(deviceAddress[i] < 16) Serial.print("0");
  Serial.print(deviceAddress[i], HEX);
}
}
void loop() {
  //chama sensor.requestTemperature() para emitir uma solicitação de temperatura
  //global a todos os dispositivos no barramento
  sensors.requestTemperatures();
  // imprime as informações do dispositivo
float T1 = sensors.getTempC(Term1);
Serial.print(T1);
  Serial.print(" ");
float T6 = sensors.getTempC(Term6);
Serial.print(T6);
  Serial.print(" ");
}

```

```
float T2 = sensors.getTempC(Term2);  
Serial.print(T2);  
  Serial.print(" ; ");  
float T4 = sensors.getTempC(Term4);  
Serial.print(T4);  
  Serial.print(" ; ");  
float T3 = sensors.getTempC(Term3);  
Serial.print(T3);  
  Serial.print(" ; ");  
float T5 = sensors.getTempC(Term5);  
Serial.print(T5);  
  Serial.println();  
  delay(5000);  
}
```

APÊNDICE B – Código fonte do programa de captura de dados da porta

serial.

```
'Lê_Serial.bas - Versão 1.0
```

```
'Este programa tem por finalidade ler a porta serial e armazenar as leituras
```

```
'num arquivo de texto, leserial.txt, armazenado no diretório raiz C:\Você pode mudar
```

```
'essa escolha: veja abaixo.
```

```
'Adaptado por Helio Salim de Amorim em 09/04/2009.
```

```
'Notes:
```

```
'#comm is the handle for our comm port (nós estamos usando a com4. Mude isto se a placa
```

```
'   Arduino não estiver em com4).
```

```
'#window is the handle for our window
```

```
'#window.te is the handle for our texteditor. To display on-screen, we print to this
```

```
'#window.gb is the handle for our graphicbox. This object is outside the borders of
```

```
'   our window so that it is hidden, and this is where we get the user keystrokes
```

```
'Also, we use lof(#comm) to find out how many characters are waiting to be read from
```

```
'   the com port, and we read those characters using the input$() function.
```

```
nomainwin
```

```
WindowWidth = 600
```

```
WindowHeight = 600
```

```
texteditor #window.te, 0, 0, 591, 454 'The handle for our texteditor is #window.te
```

```
graphicbox #window.gb, 800, 1, 10, 10
```

```
open "Temperatura" for window as #window 'The handle for our window is #window
```

```
print #window.gb, "when characterInput [getChar]" 'When the user presses a key go to
```

```
[getChar]
```

```
print #window, "trapclose [quit]" 'When the user closes our terminal window, go to [quit]
```

```
print #window.te, "!autoresize"; 'Tell the texteditor to resize with the terminal window
```

```
print #window, "font courier_new 9";
```

```
'Set the size of the communications buffer to 16K
```

```
Com = 8192 * 2
```

```
'Abre a porta de comunicação 16. It's handle is #comm.
```

```
'Se for necessário trocar a porta serial, isto é feito na próxima linha
```

```
'Vamos usar a COM16, onde se encontra a arduino.
```

```

open "COM4:9600,n,8,1,ds0,cs0,rs" for random as #comm
open "c:\temp\leserial.txt" for output as #f 'se quiser, troque aqui a opção c:\leserial.txt
prompt "Informe o nome do operador"; opera$
print #f, "Operador:"; opera$
print #f, "DATA:"; date$()
print #f, "Hora:"; time$()
if opera$="" then [loop]
i=0
[loop]
'This is where our program spends most of its time after a successful CONNECT
'If we want to capture keystrokes from our graphicbox, then it always needs
'to have the input focus. The next line ensures that it always does.
print #window.gb, "setfocus"
t$ = input$(#comm, lof(#comm)) 'Get com port data
if t$ <> "" then print #window.te, t$; 'Display com port data
    print #f, t$;
    scan 'Check for input events
    c$ = Inkey$
if len(c$) = 1 then goto [quit]
goto [loop]
[getChar]
'Whenever the user presses a key, we go here to process it.
'c$ = Inkey$
'Send the character to the com port only if it is not a special keycode
'if len(c$) = 1 then print #comm, c$;
'goto [loop]
[quit]
close #window
close #comm
close #f
end

```