

COMPARAÇÃO ENTRE TRÊS TIPOS DE SENSORES DE TEMPERATURA EM ASSOCIAÇÃO COM ARDUÍNO

Comparison between three types of temperature sensors
in association with arduino

Claodomir Antonio Martinazzo¹; Tailan Orlando²

¹ Professor do Departamento de Ciências Exatas e da Terra. URI Erechim. E-mail: mclao@uricer.edu.br

² Acadêmico do Curso de Engenharia Elétrica da URI Erechim

Data do recebimento: 18/04/2016 - Data do aceite: 24/06/2016

RESUMO: O emprego das placas Arduino para aquisição de dados é uma prática corrente para muitos profissionais e pessoas interessadas em tecnologia microcontrolada. Entre esses profissionais, estão muitos professores que a utilizam para fins didáticos. Neste trabalho, foram testados três sensores de temperatura, dois analógicos (um termistor e um LM35) e um digital (DS18B20). O teste com os três sensores, coletando dados isoladamente, apresentou bons resultados, no entanto, o sensor digital DS18B20 mostrou-se mais estável, seguido pelo termistor e por último, o LM35. No experimento em que todos os três sensores foram colocados a medir ao mesmo tempo, na mesma placa Arduino UNO R3, o LM35 teve um comportamento anormal (leitura com maior variabilidade) que foi minimizado com a utilização de um amplificador operacional no modo seguidor de tensão, trazendo as leituras para valores com menor variabilidade. Os dois sensores analógicos registraram valores de temperatura abaixo da lida em termômetro de mercúrio. O sensor digital DS18B20 apresentou leituras extremamente precisas com variabilidade zero. Concluindo, os três tipos de sensores podem ser utilizados, conforme a necessidade de precisão. Ou seja, os dois sensores analógicos fizeram, isoladamente ou em conjunto com amplificação de sinal, razoáveis leituras de temperatura, mas o sensor mais estável foi o sensor digital DS18B20.

Palavras-chave: Arduino. Sensores. Temperatura.

ABSTRACT: The use of Arduino boards for data acquisition is a common practice for many professionals and people interested in micro-controlled technology. Among these professionals are many teachers who use it for

teaching purposes. In this work, three temperature sensors were tested, two analog (a thermistor and LM35) and a digital (DS18B20) one. The test with the three sensors, collecting data alone, showed good results, however, the digital sensor DS18B20 proved to be most stable, followed by the thermistor and last the LM35. In the experiment in which all the three sensors were placed on the same plate Arduino Uno R3 to measure at the same time, the LM35 had abnormal behavior (reading with greater variability) which was minimized with the use of an operational amplifier in the voltage follower mode, bringing the readings to less variable values. The two analog sensors recorded temperature values below the ones read in the mercury thermometer. The digital sensor DS18B20 presented extremely accurate readings with zero variability. In conclusion, the three types of sensors can be used, according to the precision needed, that is, the two analog sensors did, alone or in conjunction with signal amplification, reasonable temperature readings, but the most stable sensor is the digital sensor DS18B20.

Keywords: Arduino. Sensors. Temperature.

Introdução

Com o advento da popularização da tecnologia, o acesso a dispositivos e componentes eletrônicos ficou mais fácil. Entre eles, o acesso a plataformas microprocessadas de baixo custo, como é o caso do Arduino. Segundo Martinazzo et al. (2014), a Plataforma Arduino, utilizada para aquisição automática de dados, é uma metodologia capaz de auxiliar o processo de ensino e aprendizagem de Física, levando a uma aprendizagem realmente significativa. Neste artigo, os autores versam sobre as possibilidades de utilização da referida tecnologia.

Neste trabalho, objetivou-se desenvolver uma comparação entre alguns modelos de sensores de temperatura disponíveis no mercado, a fim de comparar suas relações de custo-desempenho.

Com a grande importância e variedade de sensores de temperatura encontrados atualmente no mercado, encontrar um sensor que atenda as necessidades, seja preciso e tenha uma boa relação custo-benefício pode ser algo trabalhoso. Por meio de sensores, é

possível fazer a leitura das temperaturas em diversos casos, desde a temperatura de funcionamento de um motor até a temperatura em dado ambiente, e responder de acordo com elas, ou seja, criar um sistema capaz de interagir com o ambiente em questão.

Neste trabalho, são apresentados alguns tipos de sensores de temperatura, detalhando o seu funcionamento e demonstrando exemplos de utilização com a plataforma Arduino. São utilizados dois modelos analógicos e um digital, e a combinação destes para testes comparativos.

A escolha pela utilização da plataforma Arduino deu-se pela sua ampla disponibilidade, por ter um baixo custo e por ser uma plataforma de código aberto, com a possibilidade de fácil manipulação dos dados.

O Arduino é uma plataforma microcontrolada baseada nos microcontroladores da Atmel, com uma ampla disponibilidade de modelos para atender a inúmeras atividades. É uma placa flexível e de fácil manipulação, baseada em linguagem C/C++, que pode interagir com o ambiente por intermédio de sensores e atuadores, de modo autônomo ou

comunicando-se com um computador para realizar alguma tarefa, utilizando de um software específico.

A placa Arduino possui o microcontrolador Atmel como centro de comando. É ele que armazena os processos que deverão ser executados e é ele que executa esses comandos. A maioria das placas podem ser alimentadas pela porta USB (do inglês *Universal Serial Bus*, Barramento Serial Universal, em português), ou através de uma alimentação externa de 6 a 12 V em corrente contínua. Um regulador de tensão existente na própria placa faz o ajuste para a tensão de 5 V da placa.

As placas Arduino possuem portas digitais que podem ser utilizadas como entradas ou saídas digitais, além das portas analógicas que podem receber sinais de 0 a 5V. A quantidade de portas varia conforme o modelo de placa.

De acordo com Souza (2013), no caso das entradas analógicas, elas podem assumir diversos valores dentro de um intervalo de 0 a 5 V, porém, o microcontrolador da placa não trabalha com sinais analógicos, apenas digitais. Portanto, é preciso converter um sinal analógico para um valor digital. Esse processo é responsabilidade do conversor Analógico digital, ADC ou conversor A/D.

A quantidade de valores representáveis em um conversor A/D depende da quantidade de bits da sua resolução. A resolução de um conversor A/D pode ser obtida pela equação 01:

$$Resolução = \frac{V_{Ref}}{2^n} \quad (01)$$

onde:

V_{Ref} : tensão de referência do conversor A/D;

n: número de bits do conversor.

A resolução do conversor A/D do microcontrolador ATmega328, presente na placa Arduino UNO, é de 10 bits e a sua tensão de entrada pode variar de 0 V até V_{cc} ou o V_{Ref} . Quando está trabalhando com uma tensão V_{Ref} de 5V a resolução será de:

$$Resolução = 5/1024 = 4,88mV.$$

Para melhorar a precisão desse conversor existem duas opções: alterar a resolução do conversor ou diminuir a tensão de referência. Como a resolução do conversor analógico-digital do Arduino é fixa em 10 bits, opta-se por diminuir a tensão de referência.

Ao trabalhar com a tensão de referência externa de 3,3V, a resolução será:

$$Resolução = 3,3/1024 = 3,22mV.$$

O valor do passo fica menor para esse valor de tensão de referência.

Ao selecionar a referência externa, a resolução irá depender da tensão aplicada ao pino AREF. Para que haja a alteração da referência externa, é preciso utilizar a função *analogReference(EXTERNAL)*.

Uma das principais utilizações do Arduino é a leitura de sensores para uma análise ou para controlar uma interação com o meio, utilizado muitas vezes na automação, desde residências até processos industriais.

A palavra sensor pode ser definida como “aquilo que sente”. Para a eletrônica, um dispositivo ou componente que seja capaz de ler e extrair dados do ambiente pode ser classificado como um sensor. Os sensores podem detectar sinais dos mais variados fenômenos físicos, como temperatura, luminosidade, rotação, radiação, entre outros.

Os sensores podem ser analógicos ou digitais, sendo que os analógicos são os

mais comuns. Estes sensores comunicam-se com sinais analógicos, isto é, podem assumir infinitos valores dentro de um intervalo determinado, porém isso vai depender da resolução utilizada.

Os sensores digitais comunicam-se com níveis de tensão definidos. Estes níveis são relativos aos estados *LOW* (baixo) e *HIGH* (alto), ou também vistos como “0” ou “1”. Alguns sensores criam sinais mais complexos com sequências binárias para fazer a comunicação.

A temperatura é uma medida que relaciona o nível de agitação das moléculas com a transferência da energia cinética entre os átomos de um material, expressa em unidades de graus celsius, ou em outra escala.

Para a realização de medidas de temperatura, na maioria dos casos, é utilizada a propriedade física do aumento da resistência elétrica com o aumento da temperatura, em que um sistema monitorado é possível ler esta variação. Os sensores que se utilizam dessa propriedade são os chamados RTDs (do inglês *Resistance Temperature Detectors*, termorresistências em português).

Existe uma vasta gama de sensores de temperatura disponíveis no mercado, variando quanto à precisão, escala de temperatura, e, consequentemente, custos. Os sensores mais comuns são os termopares, os RTDs (Pt100, por exemplo) e os termistores.

Para os experimentos realizados neste projeto foram utilizados três modelos de sensores, o termistor (NTC 10k), o LM35 e o DS18B20.

Termistor

Termistores são dispositivos elétricos que possuem a sua resistência elétrica alterada com a temperatura. Ou seja, alteram sua resistência elétrica para diferentes valores de temperatura. São amplamente utilizados para o controle de temperatura em equipamentos

eletrônicos, como alarmes, termômetros, relógios, circuitos eletrônicos de compensação térmica, dissipadores de calor, ar condicionados, etc. Podem ser termistores do tipo Coeficiente Negativo de Temperatura ou Coeficiente Positivo de Temperatura. Segundo ADAFRUIT (2016), pode-se assumir que não existe termistor que, associado com um Arduino com resolução de 10 bits forneça leituras com precisão melhor do que $\pm 0,50^\circ\text{C}$.

LM35

O sensor de temperatura LM35 é um sensor preciso que possui uma resposta de tensão relativa à temperatura de forma linear que pode ser alimentado com até 40 V, apresentando uma variação de 10 mV por grau celsius de temperatura. Apresenta uma grande vantagem, pois não necessita de qualquer calibração externa para se obter exatidão e possui uma precisão de até $\pm 0,25^\circ\text{C}$ dentro da faixa de temperatura de -55°C a 150°C .

DS18B20

Segundo Maximintegrated (2015), o termômetro digital DS18B20 fornece 9-bit à 12-bit para as medições de temperatura em graus celsius e tem uma função de alarme não volátil programável com pontos de disparo superior e inferior. O DS18B20 comunica através de um barramento *1-Wire* que, por definição, requer apenas uma linha de dados para a comunicação com um microprocessador central. Tem uma temperatura de operação na gama de -55°C a $+125^\circ\text{C}$ e tem uma precisão de $\pm 0,5^\circ\text{C}$ ao longo da faixa de -10°C a $+85^\circ\text{C}$. Internamente, o DS18B20 pode derivar energia diretamente a partir da linha de dados (“energia parasita”), eliminando a necessidade de uma fonte de alimentação externa.

Ainda, segundo a Maximintegrated (2015), cada sensor DS18B20 tem um código

de série exclusivo de 64 bits, permitindo, assim, que vários sensores DS18B20s possam funcionar na mesma linha (*1-Wire*), ou seja, utilizando um único pino de dados do Arduino. Aplicações que podem se beneficiar desta funcionalidade incluem controles ambientais de aquecimento, ventilação e ar condicionado (HVAC), que permitem o monitoramento de temperatura em sistemas no interior de edifícios, equipamentos ou máquinas e processo de monitoramento e controle de sistemas. O DS18B20 utiliza a biblioteca “*OneWire*” para fazer a comunicação pelo protocolo 1^o C, disponível gratuitamente na internet.

Material e Métodos

Como apresentado na introdução foram testados três tipos de sensores de temperatura, cuja metodologias passam a ser descritas na sequência.

Leitura com termistor (10 kohm)

Como a sua resistência elétrica varia de acordo com a temperatura, o termistor permite que, ao associá-lo em série com outro resistor fixo (divisor de tensão), de preferência um valor próximo de sua resistência nominal à temperatura média ambiente, e calculando sua variação de tensão no circuito, é possível obter a variação da temperatura. Porém, essa variação não é linear. Para que a leitura da temperatura seja feita, algumas equações precisam ser utilizadas, conforme sequência.

A equação 02 fornece a tensão na saída do divisor. A equação 03 é utilizada para calcular o valor da resistência do termistor:

$$V_{out} = \frac{V_{in} \cdot (V_{leitura})}{1024} \quad (02)$$

$$R_{out} = \left(\frac{(R_{aux} \cdot V_{in})}{V_{out}} \right) - R_{aux} \quad (03)$$

onde:

V_{out} : É a tensão na saída do divisor.

$V_{leitura}$: É o valor lido na porta analógica do Arduino.

R_{out} : É o valor da resistência do termistor.

R_{aux} : É o valor do resistor auxiliar.

V_{in} : É a tensão de alimentação do circuito.

Para converter a resistência elétrica, medida no sensor, em temperatura, foram utilizadas as equações 02, 03, 04, 05, 06, culminando com a equação 07. Elas fazem parte de uma equação simplificada a partir da Equação de Stenhardt-Hart, cuja descrição detalhada pode ser vista em Wikipedia (2016), e que permite calcular, com boa aproximação, os valores de resistência elétrica de um semicondutor em diferentes temperaturas. A equação 06 fornece a temperatura em kelvin e a equação 07 faz a conversão da temperatura em kelvin para celsius. As equações 02, 03 04 e 05 fornecem os parâmetros para o cálculo da temperatura na equação 06.

$$Beta = \frac{\log\left(\frac{RT_1}{RT_2}\right)}{\frac{1}{T_1} - \frac{1}{T_2}} \quad (04)$$

$$R_{inf} = R_0 \cdot e^{\left(\frac{-Beta}{T_0}\right)} \quad (05)$$

$$Temp_K = \frac{Beta}{\log\left(\frac{R_{out}}{R_{inf}}\right)} \quad (06)$$

$$Temp_c = Temp_k - 273,15 \quad (07)$$

onde:

$Beta$: É um parâmetro.

RT_1 : Resistência medida na temperatura de calibração 1.

RT_2 : Resistência medida na temperatura de calibração 2.

T_1 : Temperatura de calibração 1.

T_2 : Temperatura de calibração 2.

R_{inf} : É um parâmetro.

R_0 : Valor nominal do RTC a 25°C.

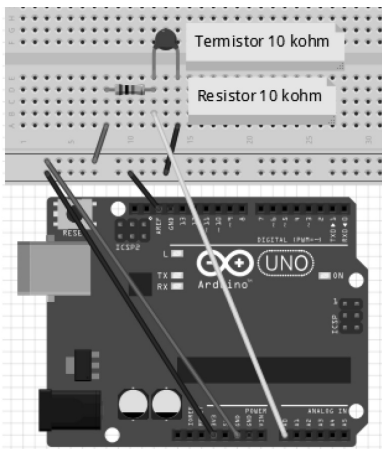
T_0 : Temperatura de 25 (298,15K).

$Temp_k$: É a temperatura de saída em kelvin.

$Temp_c$: É a temperatura de saída em °C.

A Figura 1 mostra o esquema de ligação do termistor e do resistor, ambos de 10 kohms a placa Arduino UNO R3.

Figura 1 - Simulação do circuito de aquisição de dados com termistor NTC de 10 kohms. Esquema elaborado no Fritzing.



O código da sequência foi baseado em códigos livres da internet, entre eles

ADAFRUIT, e adaptado aos parâmetros de calibração estimados, ou seja, o valor da resistência elétrica do termistor no ponto de fusão do gelo (RT_2) e no ponto de ebulição da água (RT_1).

```
#define Termistorpin A0 // pino
analógico conectado o termistor
float Vin=3.3; // [V]
float Raux=10000; // [ohm] Resistor
auxiliar
float R0=10000; // [ohm] valor
nominal do RTC a 25°C
float T0=298.15; // [K] (25°C)
float Vout=0.0; // [V] Tensão de saída
do divisor
float Rout=0.0; // [ohm] Valor de
resistência do NTC
//Ver Datasheet ou usar um multímetro
e medir a resistência do NTC nas duas
temperaturas
float T1=273; // [K] Temperatura de
calibração 1
float T2=373; // [K] Temperatura de
calibração 2
float RT1=29000; // [ohms] Resistência
medida na temperatura de calibração 1
float RT2=700; // [ohms] Resistência
medida na temperatura de calibração 2
float beta=0.0; // [K] parâmetro Beta
float Rinf=0.0; // [ohm] parâmetro Rinf
float TempK=0.0; // [K] Temperatura
de saída em Kelvin
float TempC=0.0; // [°C] Temperatura
de saída em °C
void setup() {
    analogReference(EXTERNAL);
    Serial.begin(9600);
    Serial.println("Temperatura
    Termistor:"); //parâmetros
    beta=(log(RT1/RT2))/((1/T1)-(1/
    T2));
    Rinf=R0*exp(-beta/T0);
}
```



```

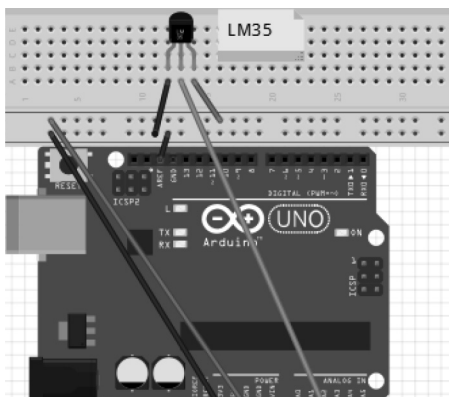
void loop(){
    //Cálculo da resistência do NTC
    Vout=Vin*((float)
    (analogRead(Termistorpin))/1024.0);
    Rout=((Vin/Vout)*Raux)-Raux;
    TempK=(beta/log(Rout/Rinf)); //
    Cálculo da temperatura
    TempC=TempK-273.15;
    Serial.println(TempC);
    delay(500);
}

```

Sensor LM35

Como foi descrito na introdução, o LM35 não necessita de qualquer calibração externa para se obter exatidão. A Figura 2 mostra a ligação do componente ao Arduino UNO R3.

Figura 2 - Simulação do circuito de aquisição de dados com o LM35. Esquema elaborado no Fritzing.



O código da sequência, foi baseado em códigos livres encontrados na internet e adaptado as necessidades deste trabalho, para fazer a leitura do sensor LM35.

```

constint LM35 = A2; // Pino Analógico
onde vai ser ligado ao pino 2 do LM35
floatVin=3.3; // [V]
float CELSIUS_BASE = 0.0;

```

```

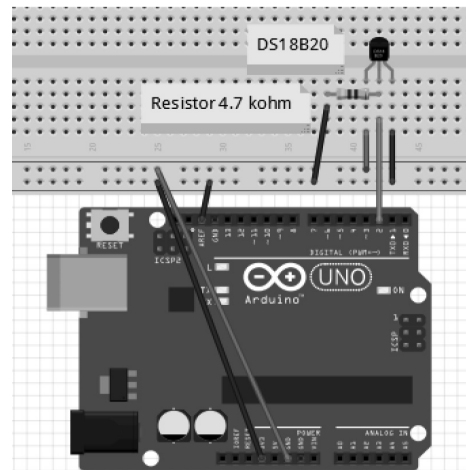
float temperaturalm35;
void setup() {
    analogReference(EXTERNAL);
    Serial.begin(9600);
    Serial.println("Temperatura LM35:");
    CELSIUS_BASE= ((Vin/1024)*100);
}
void loop() {
    temperaturalm35=(analogRead(LM35)
    * CELSIUS_BASE);
    Serial.println(temperaturalm35);
    delay(500);
}

```

Sensor DS18B20

O sensor DS18B20 necessita ser associado a um resistor de 4700 ohms. A Figura 3 mostra o esquema de ligação dos componentes a placa Arduino UNO R3.

Figura 3 - Simulação do circuito de aquisição de dados com o DS18B20. Esquema elaborado no Fritzing.



O código da sequência foi baseado em ROBOCORE, 2013 e adaptado as necessidades deste trabalho.

```

#include <OneWire.h>
int DS18S20_Pin = 2; //DS18S20 sinal no

```

```

pino digital 2
OneWireds(DS18S20_Pin); // liga o pino
digital 2
void setup(void) {
    analogReference(EXTERNAL);
    Serial.begin(9600);
}
void loop(void) {
    float temperature = getTemp();
    Serial.println(temperature);
    delay(500); //Retardo de 0,5 s.
}
float getTemp(){
    //a temperatura retorna de um
    DS18S20 em °C.
    byte data[12];
    byte addr[8];
    if (!ds.search(addr)) {
        //se não há mais sensores em
        cadeia, reinicia pesquisa
        ds.reset_search();
        return -1000;
    }
    if (OneWire::crc8(addr, 7) !=
    addr[7]) {
        Serial.println("CRC não é
        válido!");
        return -1000;
    }
    if (addr[0] != 0x10 && addr[0] !=
    0x28) {
        Serial.print("Dispositivo não
        reconhecido");
        return -1000;
    }
    ds.reset();
    ds.select(addr);
    ds.write(0x44,1); // iniciar a
    conversão, com energia parasita no
    final
    byte present = ds.reset();
    ds.select(addr);

```

```

ds.write(0xBE); // Lendo Scratchpad
for (inti = 0; i < 9; i++) { // necessita-
se 9 bytes
    data[i] = ds.read();
}
ds.reset_search();
byte MSB = data[1];
byte LSB = data[0];
floattempRead = ((MSB << 8) | LSB);
//usando complemento de dois
float TemperatureSum = tempRead /
16;
return TemperatureSum;
}

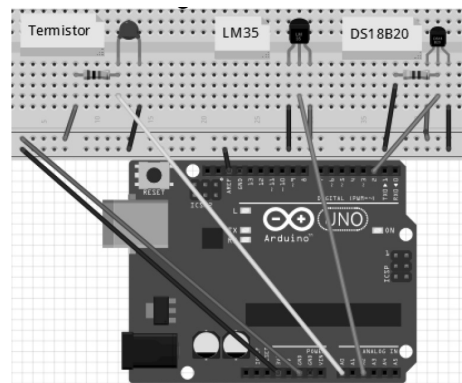
```

Aquisição de dados utilizando os três sensores ao mesmo tempo

Para fins de comparação da precisão e custo-benefício, foram implementados ao mesmo experimento os três sensores, para fazer a leitura dos dados ao mesmo tempo.

A Figura 4 apresenta o esquema da ligação dos três sensores ao Arduino UNO R3.

Figura 4 - Simulação do circuito de aquisição de dados com os três sensores. Esquema elaborado no Fritzing.



O código utilizado para a leitura dos três sensores ao mesmo tempo é a união dos três códigos citados anteriormente.

Resultados e Discussão

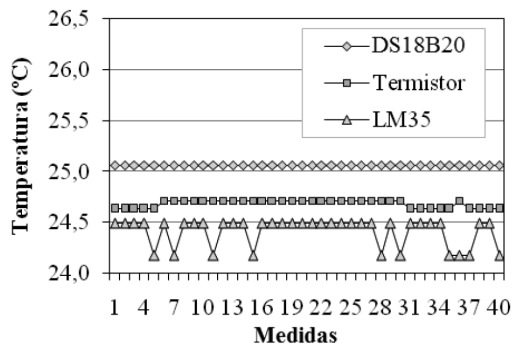
O primeiro experimento consistiu na leitura da temperatura do ambiente, pelos três sensores, um de cada vez.

A Tabela I apresenta a média, o desvio padrão (s) e o coeficiente de variação (CV) para os dados de temperatura (em segundos) para cada sensor. A Figura 5 mostra a distribuição dos valores de temperatura em um gráfico. Foram 40 medidas para cada sensor. Para esse experimento, o termômetro de mercúrio marcava 25 °C ($\pm 0,5$ °C).

Tabela I - Descrição estatística dos dados de temperatura dos três sensores.

Sensor	Temperatura (°C)		
	Média (s)	s (s)	CV (%)
Termistor	24,68	0,04	0,16
LM35	24,41	0,14	0,57
DS18B20	25,06	0,00	0,00

Figura 5 - Variação de temperatura nos sensores com leituras isoladas.



A análise dos dados mostra um pequeno coeficiente de variação, principalmente pelo fato de que as leituras não foram feitas em um ambiente com temperatura controlada, ficando, assim, os sensores sujeitos às variações da temperatura do ambiente, originadas, provavelmente, por correntes de ar ou até mesmo pela presença humana e ainda pela própria

precisão dos sensores. Observou-se que os coeficientes de variação das temperaturas ficaram em 0,16% para o Termistor, 0,57% para o LM35 e 0,00% para o DS18B20. Os valores das leituras estão muito próximos, mas, certamente, já é possível visualizar o sensor digital DS18B20 como perfeitamente estável.

Pode-se observar uma diferença na temperatura lida pelo termistor e pelo LM35 em relação à temperatura média, ou seja, abaixo da temperatura do DS18B20. Como a preocupação deste trabalho é com a variabilidade dos dados lidos por cada sensor, não foi feita uma tentativa de melhor calibração do termistor (o que poderia ser feito, no caso do termistor, medindo sua resistência num banho de gelo a 0 °C – o que não é fácil de conseguir e sua resistência a 100 °C, o que também não é fácil de conseguir na altitude de 760 m acima do nível do mar).

Com o intuito de observar a leitura da temperatura de todos os sensores, ao mesmo tempo, eles foram ligados juntos, pois poderia existir a necessidade de associar diversos tipos de sensores em um dispositivo prático, coletando as informações concomitantemente. A Tabela II mostra a descrição estatística dos dados de temperatura para cada sensor, fazendo leituras ao mesmo tempo.

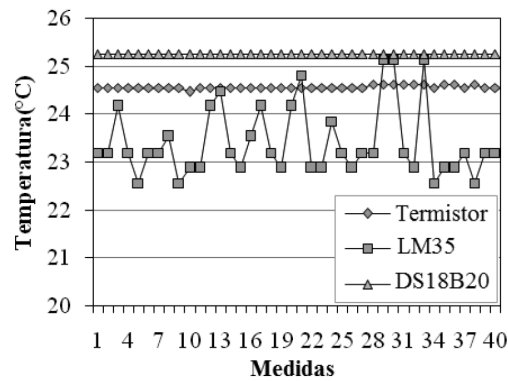
Tabela II - Descrição estatística dos dados de temperatura dos três sensores.

Sensor	Temperatura (°C)		
	Média (s)	s (s)	CV (%)
Termistor	24,57	0,04	0,15
LM35	23,40	0,73	3,13
DS18B20	25,25	0,00	0,00

Tanto a Tabela II quanto a Figura 6 mostram um problema com a leitura dos dados pelo LM35, de causa desconhecida a princípio. Observa-se, que ocorreu uma oscilação, variando em até 5 °C, enquanto os outros

sensores faziam sua leitura normalmente. O coeficiente de variação do LM35 foi para 3,13%.

Figura 6 - Variação de temperatura nos sensores fazendo leituras concomitantes, na mesma placa Arduino UNO R3.



Na tentativa de entender porque o LM35 comportava-se anormalmente em associação com os sensores DS18B20 e termistor, foram associados quatro sensores LM35. A Tabela III e Figura 7 mostram a leitura constante de três dos sensores (desvio padrão nulo), apesar de um deles diferir o valor de leitura em relação aos outros dois. Não obstante, o sensor C registrou leituras minimamente variadas (desvio padrão = 0,15). Observou-se que, na associação de sensores LM35, as leituras tornaram-se mais estáveis (variabilidade nula em três deles), apesar da discrepância na temperatura média, com exceção de um dos sensores que apresentou variabilidade, como já descrito. A explicação mais plausível, neste contexto experimental, é a de que esses valores discrepantes na média sejam normais, pois estão aproximadamente dentro da faixa de precisão de leitura do sensor LM35 ($\pm 0,25^{\circ}\text{C}$). Também não foi possível verificar a qualidade da fabricação dos sensores e seus testes de qualidade que poderiam garantir a precisão alegada dos mesmos. Portanto, os dados mostram que, na associação de sensores da mesma família, a leitura é mais

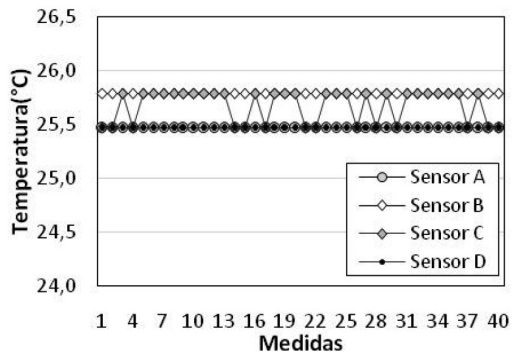
estável, em média, constatação importante, principalmente quando esses sensores estiverem controlando um ventilador ou sistema de aquecimento ou, no caso de experiências de laboratório, por exemplo, a Lei de Resfriamento de Newton.

Tabela III - Estatística descritiva parcial dos dados de temperatura dos quatro sensores LM35 fazendo leituras concomitantes, na mesma placa Arduino UNO R3.

Sensor	Temperatura ($^{\circ}\text{C}$)		
	Média (s)	s (s)	CV (%)
Sensor A	25,46	0,00	0,00
Sensor B	25,78	0,00	0,00
Sensor C	25,67	0,15	0,58
Sensor D	25,46	0,00	0,00

Fonte: Os autores.

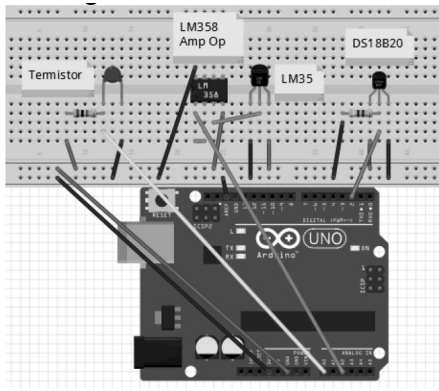
Figura 7 - Variação de temperatura nos sensores LM35 fazendo leituras concomitantes, na mesma placa Arduino UNO R3.



Outro experimento feito na tentativa de estabilizar a leitura do LM35 associado aos outros dois tipos de sensores foi a introdução de um amplificador operacional do tipo LM358, em que o sinal de saída do LM35 foi estabilizado com o uso do amplificador no modo seguidor de tensão. Assim, quem fornece a potência ao sinal é o amplificador e não a fonte de sinal, no caso, o LM35.

O código do *sketch* não foi alterado, apenas o amplificador foi adicionado no circuito, conforme Figura 8.

Figura 8 - Simulação do circuito de aquisição de dados com os três sensores com erro corrigido. Esquema elaborado no Fritzing.



A Tabela IV e Figura 9 mostram que a leitura do LM35 ainda apresenta variabilidade considerável em relação ao sensor DS18B20 e até mesmo ao termistor. No entanto, em relação ao valor da temperatura em si, está mais próxima da temperatura real, se comparado aos dados da tabela II e Figura 5. Observou-se relativa melhora na estabilidade da leitura de dados do LM35, ficando com um coeficiente de variação de apenas 0,57%, o que é aceitável, tratando-se de não estar em um ambiente de temperatura controlada.

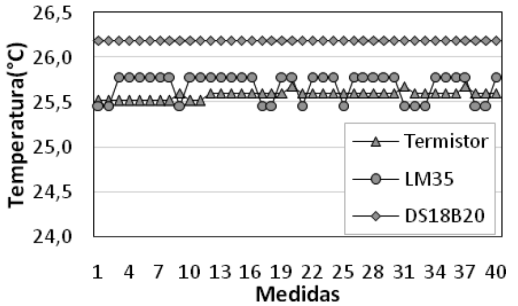
Tabela IV - Descrição estatística dos dados de temperatura dos três sensores.

Sensor	Temperatura (°C)		
	Média (s)	s (s)	CV (%)
Termistor	25,59	0,04	0,17
LM35	25,68	0,15	0,57
DS18B20	26,19	0,00	0,00

É inevitável inferir que a resolução de cada sensor interfere na leitura, na medida em que a combinação da resolução do conversor AD do Arduino com a resolução de cada sensor resultará no valor apresentado do fenômeno físico estudado. Dessa forma, parte das oscilações observadas nas leituras são explicadas pela resolução do conversor AD e das resoluções dos sensores. Parte das

diferenças observadas nas temperaturas médias dá-se por problemas de calibração dos sensores e que poderiam ser resolvidas por métodos apropriados de calibração.

Figura 9 - Variação da temperatura nos sensores com o amplificador operacional.



Em pesquisas nas lojas de componentes eletrônicos e lojas especializadas em Arduino, é possível verificar que o sensor digital DS18B20 é um sensor bem mais caro, na ordem de 16 vezes, em relação ao termistor, e 5 vezes em relação ao sensor LM35. Essas proporções de valores dependem das lojas.

Considerando os resultados dos experimentos e dos valores associados aos sensores, é possível considerar que, mesmo sendo mais oneroso, o sensor digital DS18B20 é mais apropriado em se tratando de controle de dispositivos e experimentos em que se exija maior precisão e repetitividade. Os sensores do tipo termistor e LM35 também podem ser utilizados em experimentos menos sensíveis, principalmente pelo seu baixo custo e em condições em que o valor da temperatura mais próximo possível do real não seja essencial. Os termistores em sua forma básica são muito mais baratos do que os LM35 e DS18B20. Entre termistor e LM35, considera-se que a preferência seja pelo termistor, não somente pelo menor custo, mas pela menor variabilidade nas leituras. Existe ainda a possibilidade de se obter os valores das variáveis de calibração em condições controladas de laboratório e com microcon-

troladores de melhor resolução (resolução de 12bits, por exemplo) e até mesmo testar termistores com melhor precisão.

Considerações Finais

Ao término dos experimentos, análises e observações, foi possível concluir que os três tipos de sensores de temperatura estudados poderão ser utilizados no desenvolvimento futuro de experimentos didáticos e em possíveis dispositivos práticos para o controle de temperatura. Entretanto, o melhor sensor, considerando que é digital, e que, portanto, não tem a necessidade de conversão do sinal

analogico para digital, é o DS18B20. Não obstante, seu custo relativamente elevado, é mais apropriado para aplicações em que se exija maior precisão e repetitividade.

Os sensores analógicos, LM35 e termistor de 10 kohms utilizados, tiveram resultados regulares para suas características de resolução e custos reduzidos. A necessidade de conversão de suas leituras analógicas para digitais, considerando a resolução do sistema Arduino utilizado (10 bits) e dos próprios sensores, proporcionou leituras com maior variabilidade, em relação ao DS18B20, sem, no entanto, inviabilizar sua utilização em sistemas menos críticos.

AGRADECIMENTOS

Agradecimento especial à URI que por meio dos programas de incentivo à pesquisa permitiu a realização deste trabalho.

REFERÊNCIAS

ADAFRUIT. **Using a Thermistor**. Disponível em: <<https://learn.adafruit.com/thermistor/overview?view=all#using-a-thermistor>>. Acesso em: 06 jan. 2016.

SOUZA, Fábio (2013). **Arduino – Entradas Analógicas**. Disponível em: <<http://www.embarcados.com.br/arduino-entradas-analogicas/>>. Acesso em: 18 nov. 2015.

MARTINAZZO, C. A. et al. **Arduino: uma tecnologia no Ensino de Física. Perspectiva**, Erechim, v. 38, n. 143, p. 21-30, setembro/2014. Disponível em: <http://www.uricer.edu.br/site/pdfs/perspectiva/143_430.pdf>. Acesso em: 10 abr. 2016.

MAXIMINTEGRATED. **DS18B20 – Programmable Resolution 1-Wire Digital Thermometer**. Disponível em: <<https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>>. Acesso em: 10 nov. 2015.

ROBOCORE. **Sensor de Temperatura Dallas DS18B20**. Disponível em: <https://www.roboconet.net/modules.php?name=Forums&file=viewtopic&t=3973>. Acesso em: 06 out. 2015.

WIKIPEDIA. **Steinhart–Hart equation**. Disponível em: <https://en.wikipedia.org/wiki/Steinhart%E2%80%93Hart_equation>. Acesso em: 10 jul. 2016.