Physics Education

PAPER

A data acquisition system for water heating and cooling experiments

To cite this article: J E M Perea Martins 2017 *Phys. Educ.* **52** 015019

PAPER

Phys. Educ. **52** (2017) 015019 (7pp) iopscience.org/ped

A data acquisition system for water heating and cooling experiments

J E M Perea Martins

Computer Science Department (FC), São Paulo State University (UNESP), 17033–360, Bauru-SP, Brazil

E-mail: perea@fc.unesp.br

Abstract

This work presents a simple analogue waterproof temperature probe design and its electronic interfacing with a computer to compose a data acquisition system for water temperature measurement. It also demonstrates the system usage through an experiment to verify the water heating period with an electric heater and another to verify the Newton's law of cooling

Introduction

Water temperature is a subject that allows direct association with several physical problems and scientific concepts, composing an interesting topic for educational approaches and several practical experiments in the classroom. These frequently require lots of temperature measurements, justifying the use of an automatic data acquisition system [1, 2]. Historical reviews show the scientific interest in questions relating to this subject; the water cooling process generated several studies since its first discussion published by Newton in 1701 [3, 4]. This context motivated the present work whose main relevance is the proposal of a low cost and easy hardware mounting data acquisition system with free software. Besides, it allows for the introduction of the analogue to digital conversion principle that has a large importance in the electronic instrumentation area.

Figure 1 shows the proposed data acquisition system (DAQ) named Easy-DAQ whose parts are:

1. A waterproof temperature probe designed with the sensor LM35.
2. An Arduino board to operate as data acquisition hardware interface.
3. A personal computer (PC) for data recording.

The Easy-DAQ operation is summarised as:

1. The LM35 sensor generates an electrical output voltage signal proportional to temperature.
2. The Arduino board receives the sensor output signal and generates a number proportional to it. Note that, the number is also proportional to the measured temperature. Subsequently, the Arduino sends the number to a computer.
3. The computer receives the number and automatically shows the information and plots it in a temperature graph.

The Arduino board is a small and inexpensive processing board for building electronic projects and allows for easy programing. This work uses the Arduino board model UNO-R3 programmed to operate as an analogue to digital converter (ADC), interfacing the sensor and the computer.

The proposed system requires software for the Arduino board and another for the personal computer that is referred to as Easy-DAQ software. The author will send a free of charge copy of both to anyone who requests by email (the software has free versions in English, Portuguese

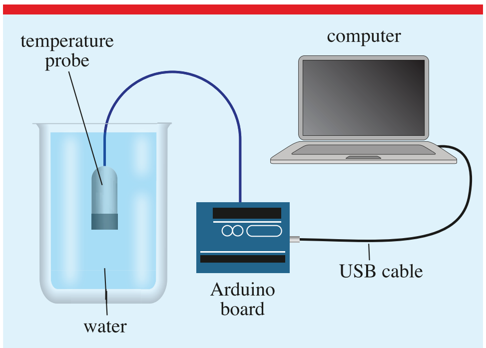

Figure 1. The Easy-DAQ structure.

and Spanish languages; there is second English version with Fahrenheit scale).

Material and methods

This section details the sensor assembling, the analogue to digital conversion system and the Easy-DAQ software.

The temperature probe

The proposed waterproof temperature probe uses the inexpensive LM35 temperature sensor that operates with the Celsius degree scale. It has two pins for power connection and a third that provides a linear output signal of 10 mV °C^{-1}. Figure 2 shows the LM35 pins and exemplifies its analysis with a simple voltmeter to verify the air temperature in the classroom. For example, a sensor output signal of 253 mV represents a temperature of 25.3 °C.

In fact, there are commercial waterproof temperature probes such as the DS18B20 model that is digital and more accurate. However, this work uses the LM35 because it allows for an educational approach about analogue to digital conversion and the proposed probe assembling can encourage students to use their own creativity for new apparatus design.

Figure 3 shows the probe is assembled covering the LM35 pins with a layer of epoxy putty to protect them from water effects. The assembling steps are:

1. Weld the sensor pins to wires with different colours.
2. Cover the sensor middle pin and the sensor body with insulating tapes to protect against short circuits and damages, respectively.

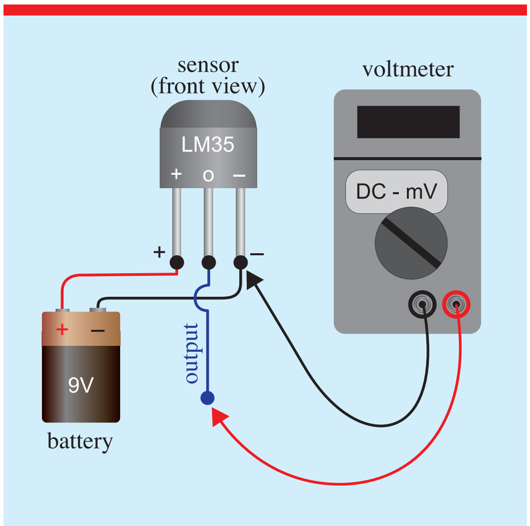

Figure 2. Demonstration of the LM35 operation.

3. Cover the pins with a layer of epoxy putty and wait for the epoxy drying. The pins cannot be in short circuit. The 'sensor base' is a small border area and requires a special attention to avoid a weak epoxy contact.
4. Sand lightly the dry epoxy and then paint it. The dry epoxy is an electrical insulator, but it is susceptible to a slow water infiltration that can disrupt the sensor operation after some hours of underwater operation. The paint avoids infiltration, but usually it is necessary at least three coats of paint.
5. Remove the insulating tape from the sensor body (yellow tape). Finally, the probe is ready.

The epoxy, wires and paint must withstand at least 100 °C that is the water boiling point.

A single LM35 can replace directly the waterproof probe for operations exclusively in dry environment.

Analogue to digital conversion

The LM35 is an *analogue sensor* and it means that its output signal (voltage) varies continuously over time, proportionally to the measured physical phenomenon (temperature). Its output signal is 10 mV °C^{-1} and its sensing range is from 0 °C to 100 °C. Therefore, at any particular time there is an output signal in the range from 0 V to 1 V.

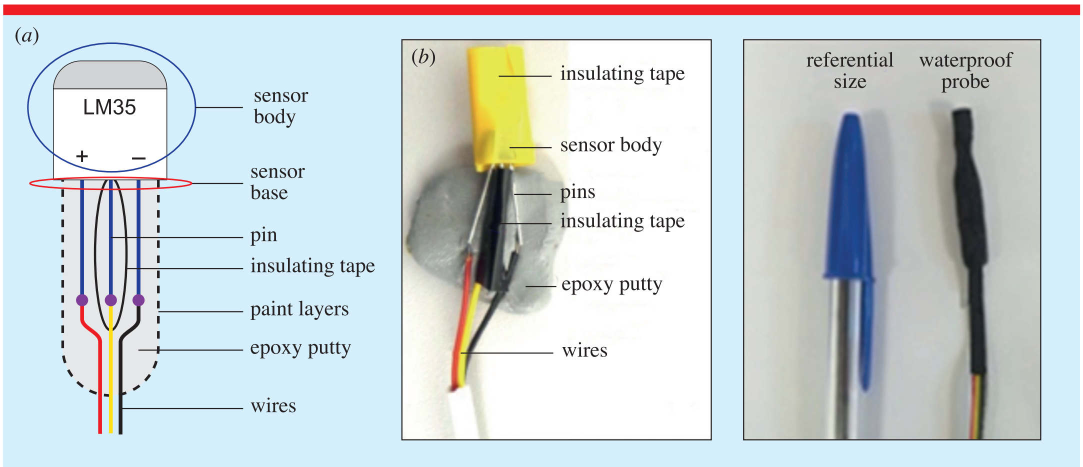

Figure 3. (A) The waterproof probe structure. (B) The assembling process. (C) The final probe.

Computers process only information represented by numbers, and the analogue sensor output is a voltage signal. Therefore, the hardware connection between both requires a special interface called ADC. It receives the sensor output voltage, generates an integer number proportional to the voltage, and sends this number to a computer where it is processed. Figure 4 exemplifies this structure.

A detailed ADC study involves several concepts of electronics and physics; however, its basic operation can be explicated through the concepts of *ADC Resolution* and the *ADC Reference Voltages* that define the relation between the ADC input voltage and the ADC output number.

Note that the ADC input signal is physically a voltage level that is mathematically analysed as a real number that theoretically can assume infinite values in a specific range. However, the ADC output is an integer number into a finite set.

The ADC output is a set of integer numbers with 2^N values, varying from 0 to $2^N - 1$, where N is the ADC Resolution, denoted in bits. It means that the ADC input voltage, which mathematically is a real number, is quantified into a discrete set of 2^N steps (Q) computed as:

$$Q = \frac{V_{\text{ref}} - V_{\text{refn}}}{2^N} \quad (1)$$

V_{ref} and V_{refn} are the ADC reference voltages. They are external voltages connected to ADC device and determine the maximum and the minimum accepted ADC input voltage, respectively.

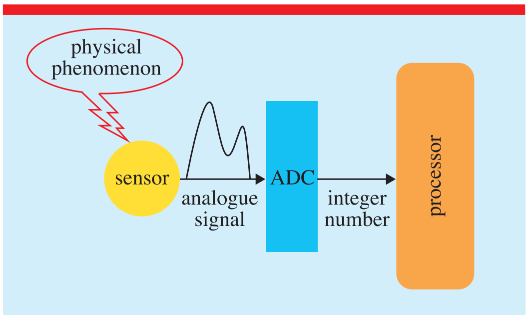

Figure 4. The ADC system structure.

Physically, they can be derived directly from the ADC power source or from special chips that provide more accurate voltage values. Usually, V_{refn} is fixed at 0 V in practical approaches.

Table 1 exemplifies the input and output relation of a hypothetical ADC whose resolution (N) is only 3 bits, V_{ref} is 5 V and V_{refn} is 0 V. It means the accepted ADC input voltage range is from 0 V to 5 V and its Q is 1.25 V that quantifies the input voltage in eight output intervals of 0.625 V each. For example, in this case, any ADC input signal in the range from 3.125 V to 3.75 V generates the same ADC output number 5.

Note that, a lower Q advantage is to ensure a better ADC ability to detect smaller input signal variations and, consequently, smaller physical phenomenon variations.

The Arduino model UNO-R3 has an internal ADC with 10 bits of resolution. In this work it is programmed to operate with V_{ref} of 5 V and V_{refn} of 0 V, which generated a Q value at about 4.9 mV

Table 1. Relation between ADC input and output.

ADC input voltage intervals (volts)	ADC output (integer number)
0.000 … 0.625	0
0.625 … 1.250	1
1.250 … 1.875	2
1.875 … 2.500	3
2.500 … 3.125	4
3.125 … 3.750	5
3.750 … 4.375	6
4.375 … 5.000	7

($5\ \text{V}/2^{10}$). The LM35 sensor output is 10 mV °C^{-1} and, therefore, it is easy to compute that this system detects temperature variations in steps of 0.49 °C, which can be rounded to 0.5 °C for an easier approach.

Figure 5 shows the sensor probe connection to the Arduino board that has header connectors. The sensor probe output is connected to the Arduino ADC input pins called 'A0' and the sensor power pins are connected to Arduino output pins called '5 V' and 'GND'. The A0 pin also has 10 μF capacitor connected to ground and 2 kΩ resistor in series to reduce noise. It can be better assembled with breadboard help.

The Easy-DAQ software

Figure 6 shows the Easy-DAQ software that has an intuitive operation. The user must define only the sampling interval and the serial interface (COM) number, and then click on the Start button to begin the data acquisition process. The Stop button finishes processing and allows the user to save the graph with a specific name.

The sampling interval is the time between two consecutive temperature measurements. The serial interface number is the virtual port number automatically created by the computer operating system when the Arduino is connected to the computer through USB.

The software screen shows:

1. The ADC input and output values.
2. The current measured temperature value.
3. A stopwatch to record experiment periods.
4. The date and time of the start and stop operations.
5. A temperature (°C) graph updated for each measurement.

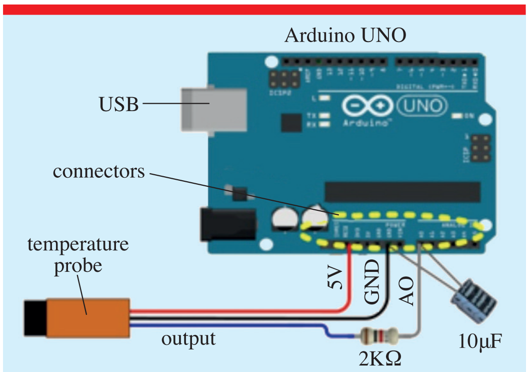

Figure 5. The temperature probe connection.

The software installation requires only its copy to a folder, without any complex operation. In spite of simplicity, the software has a user guide with detailed explanations.

Physical experiments and results

This section shows the results of two physical different experiments with the Easy-DAQ.

Water heating principle

The first proposed experiment is to verify experimentally the quantity of heat associated with the water temperature change according to the equation:

$$Q = m\ c\ \Delta T \tag{2}$$

Where: Q is the heat.
m is the mass,
c is the specific heat,
ΔT is the temperature variation.

Heat (Q) is the energy transferred from a hotter to a colder object when both are in contact and is expressed as joule (J) in the SI system. ΔT represents the difference between initial and final temperatures of the initially colder object. Mass (m) is the amount of matter in an object and is expressed in kilogram (kg). The mass of a specific volume (l) of water can be computed according to its density of 1000 kg m^{-3} or 1 kg l^{-1}.

The specific heat (c) is expressed as J (g °C)$^{-1}$ and represents as the amount of heat required to raise the temperature of one gram of a substance by one degree Celsius. The water specific heat is fixed at 4.186 J (g °C)$^{-1}$, which can be relatively large when compared with some other substances. It means

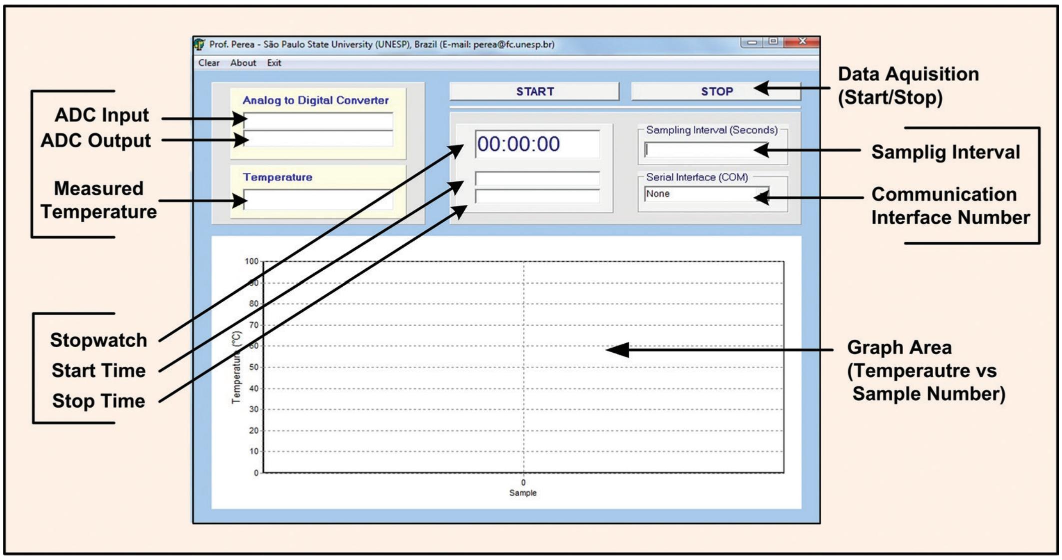

Figure 6. The user interface of the Easy-DAQ software.

that water requires a relative large amount of heat to increase its temperature and curiously it justifies why the water temperature in a lake or pool usually keeps below the ambient temperature on hot summer days.

Figure 7 shows the experiment apparatus, which includes the sensing probe and an electric immersion heater to heat the water in a jug.

The electric heater power (P) is expressed in watts (W) and represents the rate at which energy is delivered for a specific period (t). Considering that $1\ W = 1\ \mathrm{J\ s^{-1}}$, the relation between the electric heater power (P) and heat (Q) is expressed as:

$$Q = Pt \tag{3}$$

Therefore, equations (2) and (3) are associated as:

$$Pt = m\ c\ \Delta T \tag{4}$$

Note that, the water is inside a jug and it absorbs part of heat supplied by the electric heater and consequently influences directly the heating time. Therefore, equation (4) can be rewritten as:

$$P\ t = Q_{\mathrm{w}} + Q_{\mathrm{j}} \tag{5}$$

Where Q_{w} and Q_{j} are the heat required to increase the water and the jug by ΔT degrees, respectively.

This experiment used an electric heater with power of 500 watts to increase by 30 °C the temperature of 0.9 litre of water that was in an aluminium jug with specific heat of 0.9 J (g °C)$^{-1}$

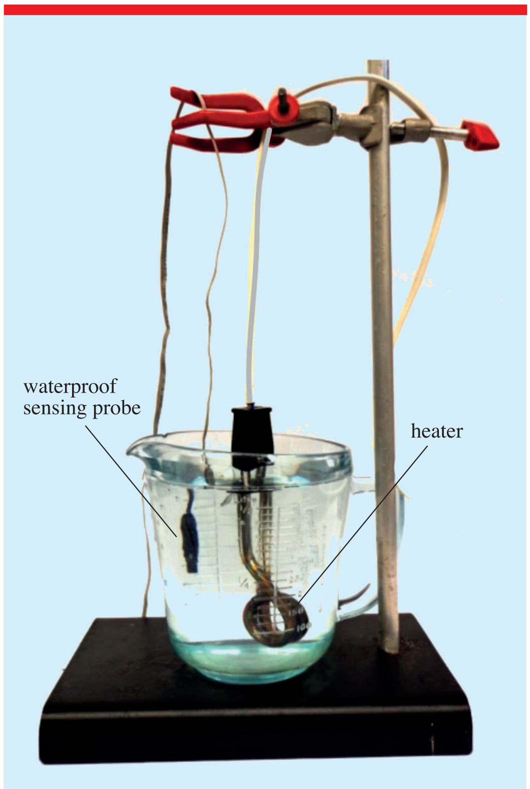

Figure 7. The water heating apparatus.

and mass of 130 g. Equation (5) computes the heating time as:

$$500 * t = (900\ g * 4.186\ J\ (g\ °C)^{-1} * 30\ °C)$$
$$+ (130\ g * 0.9\ J\ (g\ °C)^{-1} * 30\ °C)$$
$$t = 233\ s$$

This experiment steps are:

1. Connect the sensor, the Arduino and the Computer.
2. Put the sensor probe and the heater in the water.
3. Start the Easy-DAQ software.
4. Turn on the electric heater and note this time according to the software stopwatch.
5. Wait for a period (t).
6. Turn off the heater.
7. Stop the software and save the graph.

Figure 8 shows the water heating graph generated by the software. The red curve is the temperature variation as a function of the sample number. The graph also automatically includes the data/time when the process was started and the sampling interval and, therefore, the moment of each sample can be directly computed. Figure 8 was edited to highlight the heater on/off moments and the heating period.

Figure 8 shows that temperatures varied from 23 °C to 56 °C in 233 s, but the final temperature should be 53 °C. Immediately after the chart plot, the sensor was positioned at different points in the water and recorded final temperatures from 47 °C to 59 °C. The differences can be justified due to the convection in the water heating process, where the water density decreases when its temperature increases and, therefore, the hot water rises and the cold water descends, causing convection current with transport of energy.

Besides the convection currents influence, the real final temperature can be influenced by factors as:

- Heater power accuracy and physical location.
- Differences between theoretical and real values fixed for the jug specific heat (c), which that can vary significantly according to its real material composition.
- Natural heat loss due to materials and environmental interactions.
- Sensor probe accuracy and location.

After the heating period, the hot water was stirred in the horizontal and vertical directions as an attempt to reduce the temperature differences. It temporarily reached 53.5 °C at various measurement points, which is only 0.5 °C greater than the theoretical value and represents an error at about 1.0%. The experiment was repeated with water inside a glass jug with 543 grams and specific heat theoretically fixed at 0.837 J (g °C)$^{-1}$, whose final temperature error for stirred water was 7.3%. Note that, the spoon characteristics and its movement add new physical elements to the system that could be concerned for a detailed scientific study, but for an empirical approach it allowed a satisfactory result with an acceptable error level.

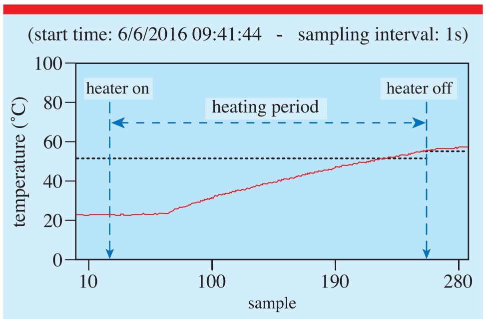

Figure 8. The water heating curve.

The Newton's law of cooling

This experiment compares theoretical and practical approaches of the Newton's Law of Cooling that computes an object temperature in a cooling process over time as:

$$T(t) = T_a + (T_h - T_a)\, e^{-kt} \quad (6)$$

Where: $T(t)$ is the object temperature at time t.
T_a is the ambient temperature.
T_h is the hot object temperature at initial time.
K is an experimental constant.
t is the time.

The procedure to verify the real water cooling process is:

1. Measure the ambient temperature (T_a).
2. Put the sensing probe into the jar and then fill it with hot water.
3. Start the Easy-DAQ software.
4. Wait until the water temperature becomes equal to the air temperature.
5. Stop the software operation and save the graph.

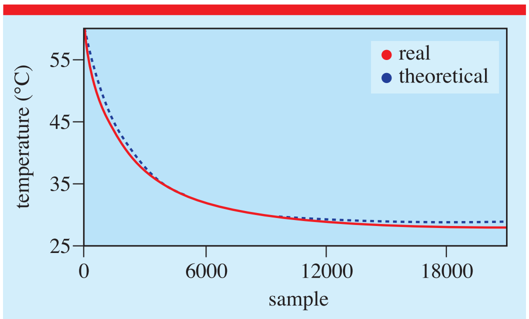

Figure 9. Real and theoretical water cooling curves.

Figure 9 shows the cooling curve (red) generated by the system and the theoretical curve (blue) computed with a spreadsheet according to equation (6).

Firstly, the software was started to plot the real cooling curve of 0.5 litre of water at 60 °C stored in an aluminium jug. The interval between samples was one second and the ambient temperature was 28.5 °C. Later, a temperature (T) and its respective time (t) were chosen in the graph to compute the k value from equation (6). Considering that k is computed and Ta is known, then a theoretical curve can be plotted with a spreadsheet and superimposed with the real curve.

Figure 9 shows that theoretical and experimental curves do not fit perfectly. In fact, the cooling process includes several physical parameters such as evaporation from water surface, transport of heat between liquid and jug, transport between jug and surroundings, ambient temperature variations during the cooling period, and the influence of water physical characteristics. It creates a complex phenomenon that does not vary linearly with temperature differences and whose accurate mathematical representation becomes hard [3, 4]. In spite of the discrepancies between theoretical and real values, the Newton's law of cooling indicates satisfactorily the phenomenon behaviour and is widely used in physical studies and textbooks.

The jug material is another component that influences directly the cooling curve slope due to its thermal conductivity that indicates the material ability to conduct heat. A low thermal conductivity indicates that the material is a thermal insulator and it justifies serving hot tea in a mug of glass and not in an aluminium mug, whose thermal conductivities are fixed at about 0.8 W m^{-1} °C^{-1} and 205 W m^{-1} °C^{-1}, respectively.

Conclusion

This work presented an analogue waterproof temperature probe design to operate with a low cost data acquisition system and proved its potential in physics education through experiments for water heating and cooling verification, which allow several physical concepts discussion in classroom.

Received 11 July 2016
Accepted for publication 3 August 2016
doi:10.1088/1361-6552/52/1/015019

References

[1] Walker J 2010 'How science works' and data logging: eleven quick experiments with a kettle *Phys. Educ.* **45** 658–69
[2] Pathare S *et al* 2015 Understanding thermal equilibrium through activities *Phys. Educ.* **50** 146–58
[3] Cheng K C and Fujii T 1998 Heat in history Isaac Newton and heat transfer *Heat Transfer Eng.* **19** 9–21
[4] Besson U 2010 The history of the cooling law: when the search for simplicity can be an obstacle *Sci. Educ.* **21** 1085–110

J E M Perea Martins is assistant professor in the Computer Science Department at the Sao Paulo State University (UNESP), in Brazil. He received his bachelor degree in computer science from UNESP in 1988 and his master and PhD degrees in physics from the University of São Paulo (USP) in 1993 and 1998, respectively. Currently, his research interest areas include electronic instrumentation, microcontrollers, computer architecture, computer simulation, and physics education.