Teaching digital control using a low-cost microcontroller-based temperature control kit

D. Ibrahim
Department of Computer Engineering, Near East University, Lefkosa, Turkey
E-mail: dogan@neu.edu.tr

Abstract The design of a low-cost digital temperature control kit is described. The system enables the students to implement various control strategies using a microcontroller. The kit is intended to be helpful in a control laboratory as a complement to the digital control system theory taught to undergraduate students.

Keywords microcontroller-based control; teaching digital control; temperature control

With the availability of low-cost computers and microcontrollers, digital control has gained popularity and most current control systems are based on digital techniques. The same is true for simulation. Digital simulation techniques have replaced analogue simulators. These simulators are in the form of interactive computer packages used in the industry, in research laboratories, colleges and universities. Early simulation packages such as ACSL[1] were designed for large mainframe computers and only large organizations could afford to purchase and use such packages. Currently, software packages such as TUTSIM,[2] 20-sim,[3] program CC,[4] VisSim,[5] Extend,[6] and MATLAB[7] are available for desktop PCs and they are within the budgets of most universities and colleges. Some packages (e.g. MATLAB and program CC) are also offered in the form of a low-cost student edition and students can purchase and use these packages on their own PCs, away from the university laboratories.

Simulation is an invaluable tool in teaching the theory of control systems. For example, the students can plot the accurate root-locus of a complex system in a matter of a few minutes rather than spending several hours. Similarly, the time and frequency responses of a system can very easily be plotted with the aid of a simulator package. Although the simulators are very useful tools they are not the same as real world solutions. There are also cases in which computer models may be inappropriate, or the system is too complex to describe by mathematical equations in a computer. It is the authors' experience that students learn an engineering topic best when they see the physical results of the experiments they perform. Simulation can still be used at the first level of the analysis but this should be supported with real physical laboratory experiments. Thus, simulation should be a complementary tool rather than the only tool when teaching an engineering topic.

One of the problems with commercially available physical laboratory experiments is that the experimental kits are usually very expensive, especially when a number of similar kits are purchased for teaching purposes. Such kits may also require frequent calibration and maintenance services as a result of component failures and ageing. Laboratory kits also do become obsolete quite rapidly as new products are developed.

This paper describes a low-cost temperature control kit which is designed and used in the engineering teaching laboratories of the Near East University. The kit is based on the popular PIC16F877 model microcontroller, manufactured by Microchip Inc.[8] The overall cost of the kit is less than $200, which is well below the cost of similar commercially available educational temperature control kits. The design, modelling and digital control of the kit are described in detail.

Temperature control kit

Educational temperature control kits are not new. Many companies manufacturing laboratory kits also offer some kind of general process control or temperature control kits. TCL-1 by Kuruganti[9] is a temperature control loop trainer which is intended to show how the temperature in a heat exchanger can be controlled. TCT-1, also by Kuruganti, is an on/off-based temperature control teaching kit. PROCON by Feedback Instruments[10] is a process control system, which includes rigs for level, flow, temperature and pH control. Here, the temperature control system uses water as the process fluid and the kit provides PID control with auto-tune facilities. G34/EV is a PID-based educational temperature control unit manufactured by Elettronica. The unit can be interfaced to a PC and consists of a PID controller, power amplifier, and signal conditioner for temperature sensors.

Near East University offers undergraduate and graduate level engineering courses and control engineering is one of these topics, which is taught for one semester. There are no practical experiments and students have been using the MATLAB package to design, simulate and test their control theory. It was felt necessary to provide some practical experiments to the students as a way of supporting the theoretical concepts taught in the classroom. The main reason to design a control kit rather than to buy a commercially available one was the cost. Process control is a very important field of automatic control engineering and as a first initiative it was decided to develop a digital temperature control experiment based upon a microcontroller. One of the aims during the implementation of this laboratory kit was to use low cost, but industrial equipment in order to ensure the necessary robustness for its use. It is hoped to develop more control-based experiments in the near future with the participation of members of the faculty and students.

The block diagram of the digital temperature control kit is shown in Fig. 1. The working principle of this experiment consists of heating the water in a small container using a low-voltage electric heating element and a simple MOSFET-based power controller circuit. A temperature sensor is immersed into the water whose output signal is sent to a PIC-type[8] microcontroller. This signal is compared with a reference temperature signal and a PID controller algorithm is implemented by the microcontroller to achieve the required temperature control action.

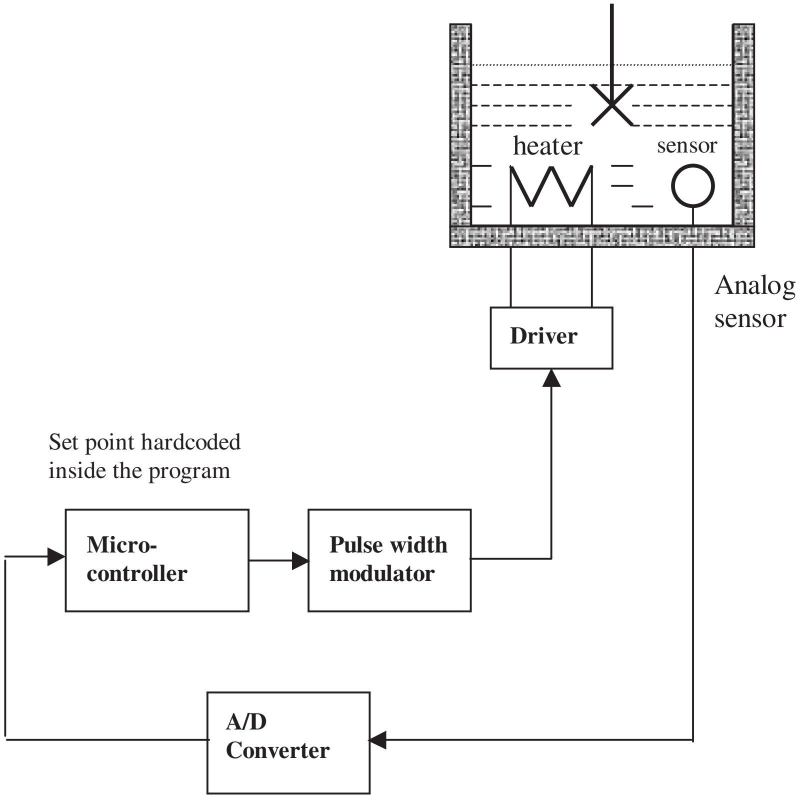

Fig. 1 *Schematic of the temperature control kit.*

This process is used to teach the following important concepts to the students:

- Modelling and identification of a real physical process
- Using the Ziegler-Nichols[11,12] tuning method
- Using microcontrollers in process automation
- Developing and experimenting with digital PID[13,14,15] controllers.

Figure 2 shows a picture of the prototype experiment kit. The kit is rather simple, consisting of only low cost materials. A round plastic container is used to store the water. The heater element and the sensor are immersed in this container. The temperature is sensed using a low cost semiconductor sensor, which is protected inside a glass tube. The heating element is the type which is used in camping and other outdoor activities in order to warm up liquid in a cup, for example for making coffee. The heater operates with 12 V, draws 10 A of current and provides a power of 120 W. A laboratory power supply to provide such high power is usually rather

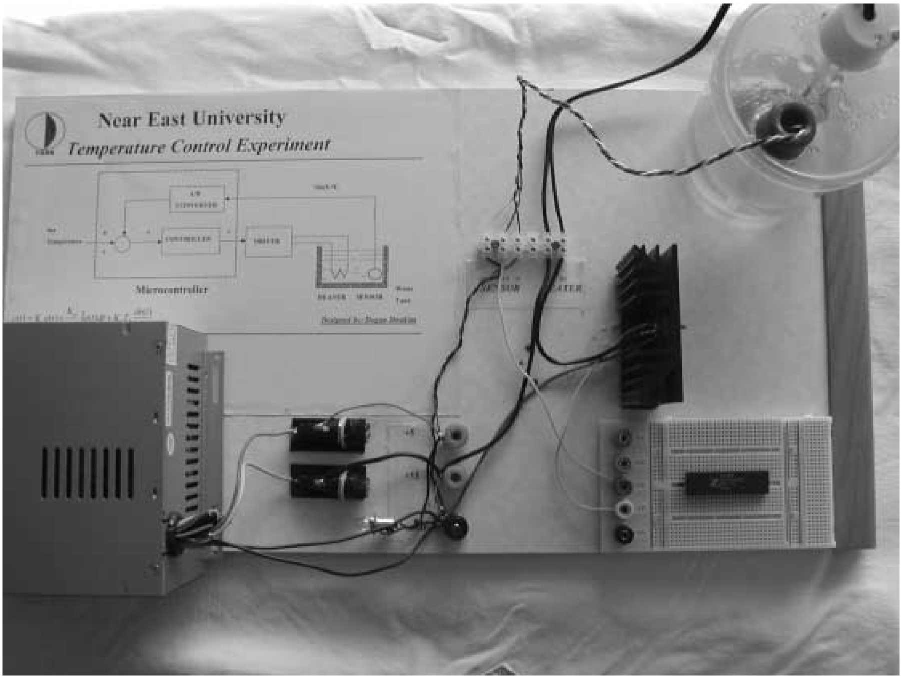

Fig. 2 *The temperature control kit.*

expensive so a standard 350 W PC power supply is used instead, costing no more than $50. Using a low voltage in an experimental kit has the advantage that the system is safe as there is no risk of electric shock.

Figure 3 shows the electrical circuit diagram of the kit. The circuit is rather simple, consisting of only a few parts. LM35DZ[16] is the analog semiconductor temperature sensor, PIC16F877 is the microcontroller, and IRL1004[17] is a power MOSFET switch, used to drive the heater element.

The temperature sensor

The temperature sensor used in the experiment is a 3-pin semiconductor sensor with an output voltage directly proportional to the temperature. The output of the sensor is connected to one of the A/D converter inputs of the microcontroller.

There was the option of using a digital output sensor, but they are usually more expensive and it was also felt necessary to use an analog sensor and teach the students the practical applications of A/D converters.

The microcontroller

In order to lower costs, we needed a microcontroller with a built-in A/D converter. Process control algorithms require the use of floating-point arithmetic and as a result,

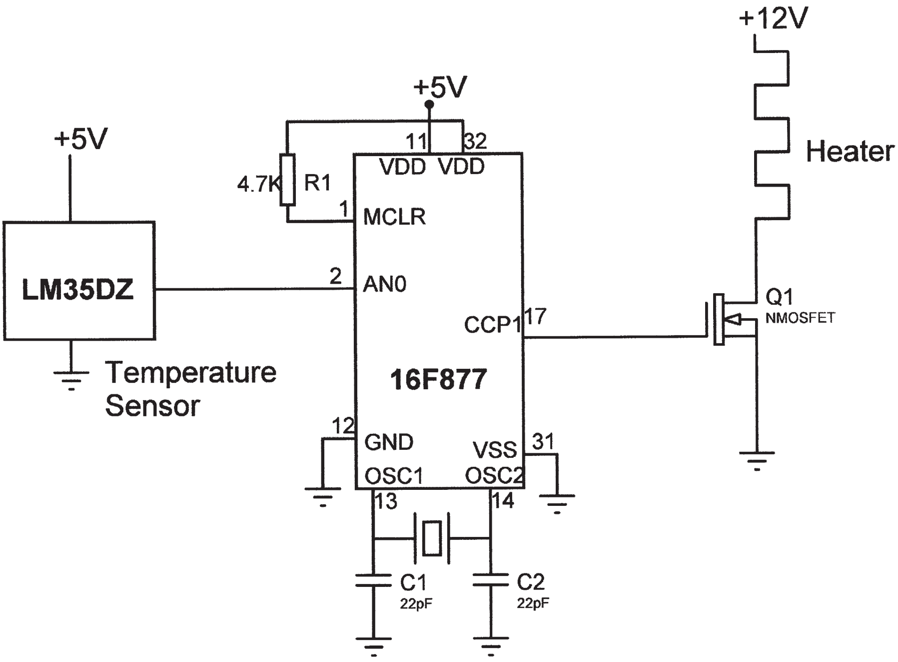

Fig. 3 *Circuit diagram of the kit.*

a microcontroller with a large data memory was also required. Another requirement to lower the cost was a built-in pulse width modulated (PWM) output, which was used to drive the heater circuitry linearly. The development of floating-point arithmetic routines is very complex using the assembly language and it was decided to program the microcontroller in a high-level language which also supported the floating-point arithmetic.

The PIC16F877 microcontroller satisfied all of our requirements. This is one of the most popular microcontrollers used in industry and it offers the following features:

- 8K × 14 flash program memory
- 368 bytes RAM memory
- 256 bytes EEPROM memory
- 8 × 10 bit A/D converters
- Pulse width modulated (PWM) output
- High-level language support

The FED C[18] compiler was used for program development. This is a Windows-based low-cost compiler for the PIC family of microcontrollers. The compiler offers a large number of standard C library functions, including support for floating-point arithmetic.

The heater driver

An IRL1004 power MOSFET switch is used to drive the heater element. This MOSFET can dissipate up to 200 W when mounted on a suitable heat sink. The heating element is connected to the drain pin of the MOSFET and the gate input is controlled from the microcontroller (see Fig. 3).

Large industrial temperature control systems are based on a.c. power control techniques using thyristors and triacs and appropriate theory is given to the students on this topic.

Modelling

The system can be approximated to a first-order system with a time lag. A simplified mathematical model of the overall system can be derived as described here.

Mathematical model of the tank

The heat-balance equation for the tank can be written as:

Heat input to the system = heat increase in the system + heat losses

If we let

m_1 = mass of water inside the tank
m_2 = mass of the tank
c_1 = specific heat capacity of water
c_2 = specific heat capacity of the tank

Ignoring the heat loss through the walls of the tank and the heat capacities of the heater element and the mixer, we can write the following equation

$$\text{Heat increase in the tank} = (m_1c_1 + m_2c_2)\frac{\mathrm{d}T}{\mathrm{d}t}$$

$$\text{Heat loss from the tank} = hA(T - T_a)$$

Where T_a is the ambient temperature, A is the tank top area, and h is a constant, which depends on the surface and the ambient temperature.

Thus, the heat input to the system is

$$E = (m_1c_1 + m_2c_2)\frac{\mathrm{d}T}{\mathrm{d}t} + hA(T - T_a) \tag{1}$$

if we assume that the ambient temperature is constant, and let

$$T_q = T - T_a$$

we can write eqn (1) as:

$$E = (m_1c_1 + m_2c_2)\frac{\mathrm{d}T_q}{\mathrm{d}t} + hAT_q$$

or, letting $k_1 = (m_1c_1 + m_2c_2)$ and $k_2 = hA$,

$$\frac{T_q(s)}{E(s)} = \frac{1}{sk_1 + k_2} \tag{2}$$

which is a first order system with time constant k_1/k_2.

Mathematical model of the heater

The relationship between the applied voltage and the energy generated by an electrical heating element is non-linear. In this experiment this relationship is linearised by driving the heater from a pulse width modulated (PWM) signal. A pulse width modulated signal is generated from the microcontroller as shown in Fig. 4 where M and S are the mark and the space of the waveform, and T is the period, i.e. $T = M + S$. This waveform is used to control a power MOSFET switch where the heater element is connected as the load of this device.

The r.m.s. value of the current through the heater can be calculated as

$$\begin{aligned} I_{\mathrm{rms}} &= \sqrt{\frac{1}{T}\int_0^T i^2(t)\,\mathrm{d}t} \\ &= \sqrt{\frac{1}{T}\int_0^M I_o^2} \\ &= \sqrt{\frac{MI_o^2}{T}} \end{aligned}$$

or,

$$I_{\mathrm{rms}} = I_o\sqrt{\frac{M}{T}} \tag{3}$$

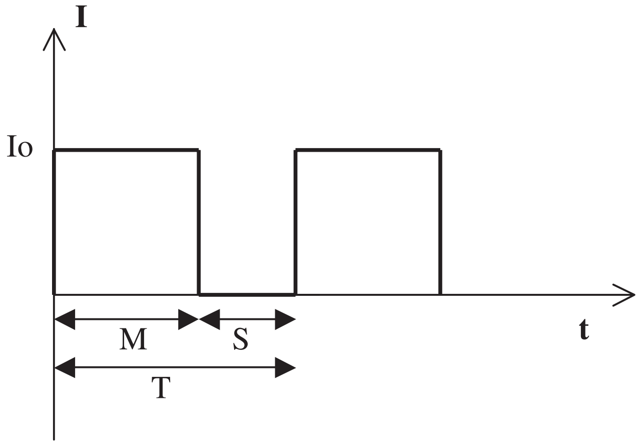

Fig. 4 *PWM heater waveform.*

Assuming the heating element has a pure resistance, R, the average power delivered to the heater can be calculated as:

$$\begin{aligned} P_{\mathrm{AV}} &= RI_{\mathrm{rms}}^2 \\ &= RI_{\mathrm{o}}^2 \frac{M}{T} \end{aligned}$$

if we let $\alpha = \frac{RI_{\mathrm{o}}^2}{T}$ then

$$P_{\mathrm{AV}} = \alpha \cdot M \tag{4}$$

Equation (4) shows that the average power delivered to the load is linearly proportional to the *on-time (M)* of the signal. We will call M the *duty cycle* of the waveform.

The frequency of the waveform must be well above the closed-loop bandwidth of the control system so that the process is only affected by the mean level of the waveform. In this project, we will assume a frequency of 1 kHz, i.e. the period is 1 ms.

In this project,

$$R = 1.2\,\Omega$$

$$I_{\mathrm{o}} = 10\,\mathrm{A}$$

$$T = 1\,\mathrm{ms} = 10^{-3}\,\mathrm{s}$$

Thus, the transfer function of the heater is, from eqn (4):

$$P_{\mathrm{AV}} = \frac{1.2 \times 100}{10^{-3}} M$$

or,

$$\frac{P_{\mathrm{AV}}}{M} = 1.2 \times 10^5 \tag{5}$$

where P_{AV} is in watts and M is in seconds.

Equation (5) shows the linear relationship between the duty cycle of the applied signal and the average power generated by the heater.

Mathematical model of the temperature sensor

The temperature sensor is a semiconductor device with a linear voltage-temperature relationship specified as 10 mV/°C, i.e.

$$\frac{V_{\mathrm{o}}}{T} = 0.01 \tag{6}$$

Where V_{o} is the sensor output voltage in volts, and T is the temperature in °C.

Experiment example

Identification of the system

The dynamic behaviour of the system is identified using non-parametric modelling, by using a reaction curve method. For this, the feedback loop is opened, and a step PWM input is applied to the heater driver by the microcontroller. The temperature of the water in the tank is then measured and recorded every second by connecting the output of the sensor to the voltage input of DrDaq[19] hardware and Picolog[19] software. Both of these products are manufactured by Pico Technology. DrDaq is a small card which is plugged into the parallel port of a PC. The card is equipped with sensors to measure the physical quantities such as the light intensity, sound level, voltage, humidity, and temperature. Picolog software runs on a PC and can be used to record the measurements of the DrDaq card in real time. The software includes a graphical option that enables the measurements to be plotted.

A Ziegler-Nichols tuning method is then used to identify the system, as shown in Fig. 5. The open-loop system transfer function was found to be

$$G(s) = \frac{0.826e^{-180s}}{(1+1800s)} \tag{7}$$

The system has a large time lag (180 seconds) and a time constant of 1800 seconds.

Choosing a controller algorithm

The PID algorithm was selected as the controller since it is probably the most extensively used method in industrial process control applications. A large number of

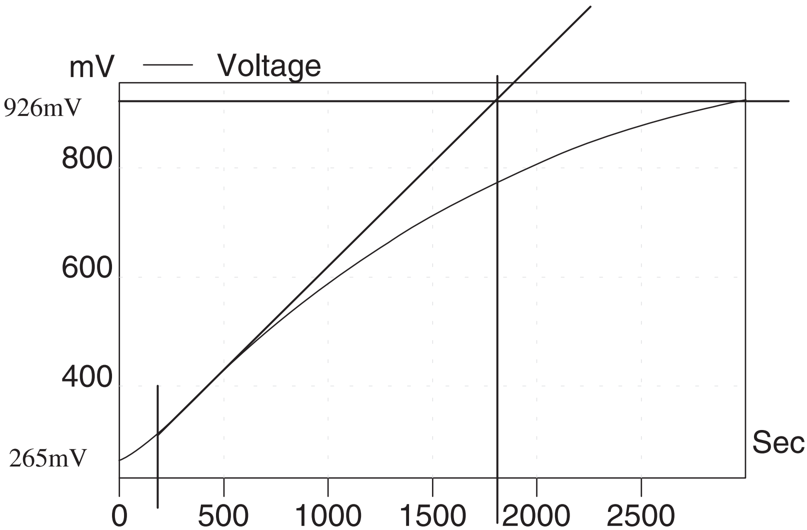

Fig. 5 *Using the Ziegler-Nichols method to find system parameters.*

references can be found which describe the continuous and digital forms of this controller, its performance evaluation, implementation and auto-tuning forms.

The transfer function of the standard PID algorithm is:

$$u(t) = K_\mathrm{p} e(t) + \frac{K_\mathrm{p}}{T_\mathrm{t}} \int_0^t e(t)\,\mathrm{d}t + K_\mathrm{p} T_\mathrm{d} \frac{\mathrm{d}e(t)}{\mathrm{d}t} \tag{8}$$

The block diagram of the continuous PID controller is shown in Fig. 6, where, K_p is the proportional gain, T_i is the integral time constant, and T_d is the derivative time constant.

In the s-domain, the PID controller can be written as

$$U(s) = K_\mathrm{p}\left[1 + \frac{1}{T_\mathrm{t} \cdot s} + TD \cdot s\right] E(s) \tag{9}$$

The discrete form of the PID controller can be derived by finding the z-transform of eqn (9):

$$U(z) = E(z) K_\mathrm{p}\left[1 + \frac{T}{T_\mathrm{i}\left(1 - z^{-1}\right)} + T_\mathrm{d} \frac{\left(1 - z^{-1}\right)}{T}\right] \tag{10}$$

Equation (10) is usually written as:

$$\frac{U(z)}{E(z)} = a + \frac{b}{1 - z^{-1}} + c\left(1 - z^{-1}\right) \tag{11}$$

where

$$a = K_\mathrm{p}, \qquad b = \frac{K_\mathrm{p} T}{T_\mathrm{i}}, \qquad c = \frac{K_\mathrm{p} T_\mathrm{d}}{T}.$$

PI Controller

The Ziegler-Nichols parameters for a PI controller are:

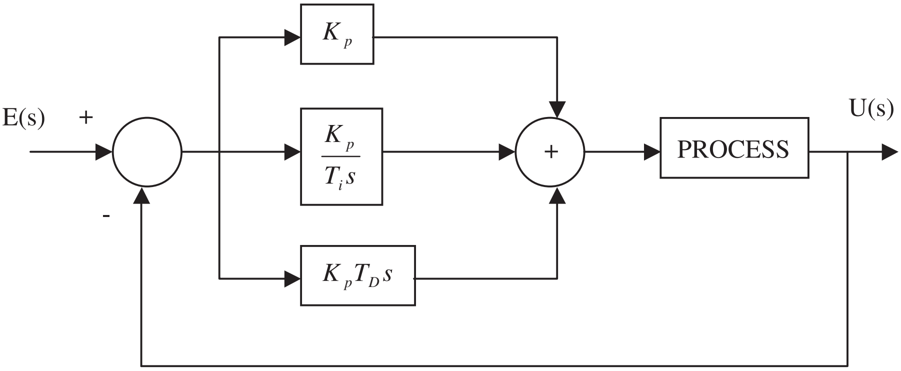

Fig. 6 *Block diagram of continuous PID controller.*

$$K_p = \frac{0.9T_i}{KT_d} \quad \text{and} \quad T_i = 3.3T_d$$

Taking a sampling time of $T = 20\,\text{s}$, a and b in eqn (10) are calculated to be a = 10.9, $b = 0.37$.

The PI algorithm implemented on the microcontroller is the following:

```
BEGIN
   Read a and b parameters of the controller
   Read MAX and MIN
   Read the set-point temperature
   Initialise the A/D converter

   DO FOREVER
      Read the set-point              r(kT)
      Read water temperature          y(kT)
      Calculate error                 e(kT) = r(kT) − y(kT)
      Calculate proportional term     q(kT) = ae(kT)
      Calculate integral term         p(kT) = be(kT) + p(kT-T)
      Calculate output                u(kT) = p(kT) + q(kT)
                                      if(u(kT) > MAX)
                                         p(kT) = p(kT-T)
                                         u(kT) = MAX
                                      else if(u(kT) < MIN)
                                         p(kT) = p(kT-T)
                                         u(kT) = MIN
                                      endif
      Save for next cycle             p(kT-T) = p(kT)
                                      e(kT-T) = e(kT)
      Wait for next cycle

   ENDDO

END
```

This algorithm was implemented using C language. The controller output is limited to be within MIN and MAX in order to avoid integral saturation.

The response of the system with the PI controller is shown in Fig. 7. In this example, the set point was 30°C and the temperature reached this value with no overshoot and no steady-state errors. One of the advantages of using the Ziegler-Nichols method is that it yields a satisfactory response with little effort.

Some computer packages such as the ExperTune[20] by Top Control run on a PC, analyse a system in real time and provide an optimum set of PID controller parameters. With the availability of such packages it should take much less time to tune a PID controller satisfactorily.

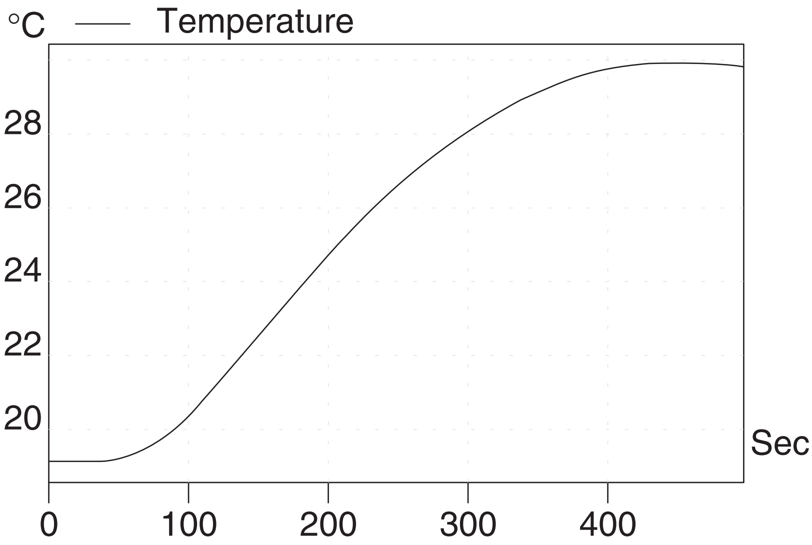

Fig. 7 *Response of the system with PI controller.*

Conclusion

The design of a low-cost digital temperature control kit has been described. The aim in designing this kit was to teach engineering students the practical applications of the theory they are taught in the classroom.

The kit is designed using standard low-cost components which are readily available in most electronic component shops. Another advantage of the kit is that it enables the students to experiment and learn the microcontrollers which are used extensively in most intelligent electronic control projects. The kit is complemented with a laboratory manual which is written to help the students follow the experiments in an orderly way.

Other control algorithms and design procedures such as state-space techniques can be developed for the kit. It is also hoped to develop other automation kits in the near future, such as level control systems, flow control systems, servo control systems and so on.

References

1 *ACSL User Guide* (Mitchell & Gauthier, Concord, MA, 1981).
2 M. H. Waller and P. E. Wellstead, 'Tuning of a d.c. motor position servo controller using TUTSIM', *IJEEE*, **24** (1987), 251–258.
3 Controllab products web site: http://www.20sim.com
4 P. M. Thompson, '*User's Guide Program CC*' (Systems Technology, 1985).
5 Adept Scientific web site: http://www.adeptscience.co.uk/products/mathsim/vissim/
6 Imagine That Inc. web site: http://www.imaginethatinc.com/aboutext.htm
7 Mathworks web site: http://www.mathworks.com
8 Microchip web site: http://www.microchip.com

9 Kuruganti web site: http://www.kuruganti.com/products.htm

10 Feedback Instruments web site: http://www.fbk.co.uk

11 J. G. Ziegler, N. B. Nichols and N. Y. Rochester, 'Optimum setting for Automatic Controllers', *Trans. ASME*, **64** (1942), 759–768.

12 J. R. Leigh, 'Temperature measurement & control', IEE Control Engineering Series, **33** (1991).

13 C. Kuo-Cheng, A. B. Corripio and C. L. Smith, 'Digital control algorithms. Part III. Tuning PI and PID controllers', *Instr. Control Systems*, (1973), 41–43.

14 J. P. Gerry, 'A comparison of PID control algorithms', *Control Engineering*, (1987), 102–105.

15 P. J. Gawthrop, 'Self-tuning PID controllers: algorithms and implementation', *IEEE Trans. Automatic Control*, **31** (1986), 201–209.

16 National Semiconductors web site: http://national.com/pf/LM/LM35.html

17 Kyungsung web site: http://kyungsung-elect.co/kr/irf-switch.htm

18 Forest Electronics Developments web site: http://fored.co.uk

19 Pico Technology web site: http://www.picotech.com

20 ExperTune web site: http://www.expertune.com