1 Introduction

1.1 THE IDEA OF SYSTEM CONTROL

Control engineering is concerned with controlling a dynamic system or plant. A dynamic system can be a mechanical system, an electrical system, a fluid system, a thermal system, or a combination of two or more types of system. The behaviour of a dynamic system is described by differential equations. Given the model (differential equation), the inputs and the initial conditions, we can easily calculate the system output.

A plant can have one or more inputs and one or more outputs. Generally a plant is a continuous-time system where the inputs and outputs are also continuous in time. For example, an electromagnetic motor is a continuous-time plant whose input (current or voltage) and output (rotation) are also continuous signals. A control engineer manipulates the input variables and shapes the response of a plant in an attempt to influence the output variables such that a required response can be obtained.

A plant is an *open-loop* system where inputs are applied to drive the outputs. For example, a voltage is applied to a motor to cause it to rotate. In an open-loop system there is no knowledge of the system output. The motor is expected to rotate when a voltage is applied across its terminals, but we do not know by how much it rotates since there is no knowledge about the output of the system. If the motor shaft is loaded and the motor slows down there is no knowledge about this. A plant may also have disturbances affecting its behaviour and in an open-loop system there is no way to know, or to minimize these disturbances.

Figure 1.1 shows an open-loop system where the system input is expected to drive the system output to a known point (e.g. to rotate the motor shaft at a specified rate). This is a single-input, single-output (SISO) system, since there is only one input and also only one output is available.

In general, systems can have multiple inputs and multiple outputs (MIMO). Because of the unknowns in the system model and the effects of external disturbances the open-loop control is not attractive. There is a better way to control the system, and this is by using a sensor to measure the output and then comparing this output with what we would like to see at the system output. The difference between the desired output value and the actual output value is called the *error signal*. The error signal is used to force the system output to a point such that the desired output value and the actual output value are equal. This is termed *closed-loop* control, or feedback control. Figure 1.2 shows a typical closed-loop system. One of the advantages of closed-loop control is the ability to compensate for disturbances and yield the correct output even in the presence of disturbances. A *controller* (or a *compensator*) is usually employed to read the error signal and drive the plant in such a way that the error tends to zero.

Microcontroller Based Applied Digital Control D. Ibrahim

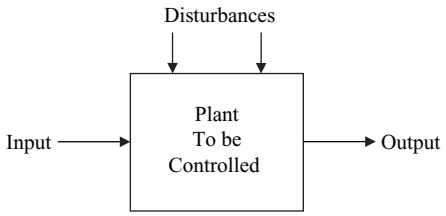

Figure 1.1 Open-loop system

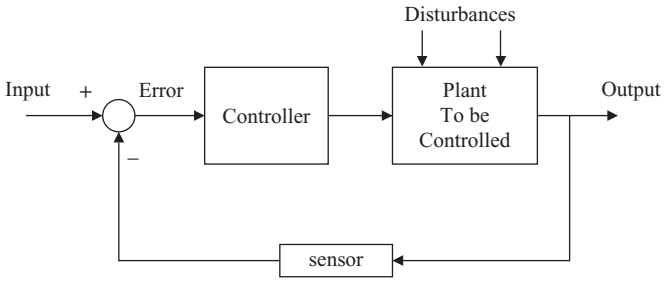

Figure 1.2 Closed-loop system

Closed-loop systems have the advantage of greater accuracy than open-loop systems. They are also less sensitive to disturbances and changes in the environment. The time response and the steady-state error can be controlled in a closed-loop system.

Sensors are devices which measure the plant output. For example, a thermistor is a sensor used to measure the temperature. Similarly, a tachogenerator is a sensor used to measure the rotational speed of a motor, and an accelerometer is used to measure the acceleration of a moving body. Most sensors are analog devices and their outputs are analog signals (e.g. voltage or current). These sensors can be used directly in continuous-time systems. For example, the system shown in Figure 1.2 is a continuous-time system with analog sensors, analog inputs and analog outputs. Analog sensors cannot be connected directly to a digital computer. An analog-to-digital (A/D) converter is needed to convert the analog output into digital form so that the output can be connected to a digital computer. Some sensors (e.g. temperature sensors) provide digital outputs and can be directly connected to a digital computer.

With the advent of the digital computer and low-cost microcontroller processing elements, control engineers began to use these programmable devices in control systems. A digital computer can keep track of the various signals in a system and can make intelligent decisions about the implementation of a control strategy.

1.2 COMPUTER IN THE LOOP

Most control engineering applications nowadays are computer based, where a digital computer or a microcontroller is used as the controller. Figure 1.3 shows a typical computer controlled system. Here, it is assumed that the error signal is analog and an A/D converter is used to convert the signal into digital form so that it can be read by the computer. The A/D converter

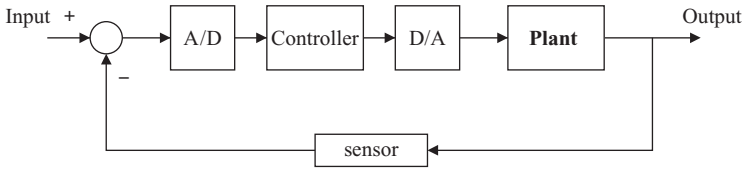

Figure 1.3 Typical digital control system

samples the signal periodically and then converts these samples into a digital word suitable for processing by the digital computer. The computer runs a controller algorithm (a piece of software) to implement the required actions so that the output of the plant responds as desired. The output of a digital computer is a digital signal, and this is normally converted into analog form by using a digital-to-analog (D/A) converter. The operation of a D/A converter is usually approximated by a zero-order hold transfer function.

There are many microcontrollers that incorporate built-in A/D and D/A converter circuits. These microcontrollers can be connected directly to analog signals, and to the plant.

In Figure 1.3 the reference set-point, sensor output, and the plant input and output are all assumed to be analog. Figure 1.4 shows the block diagram of the system in Figure 1.3 where the A/D converter is shown as a sampler. Most modern microcontrollers include built-in A/D and D/A converters, and these have been incorporated into the microcontroller in Figure 1.4.

There are other variations of the basic digital control system. In Figure 1.5 another type of digital control system is shown where the reference set-point is read from the keyboard or is hard-coded into the control algorithm. Since the sensor output is analog, it is converted into digital form using an A/D converter and the resulting digital signal is fed to the computer where the error signal is calculated and is used to implement the control algorithm.

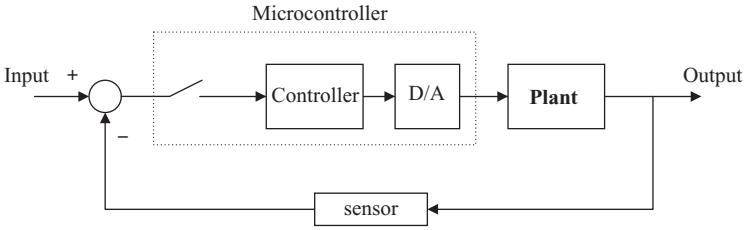

Figure 1.4 Block diagram of a digital control system

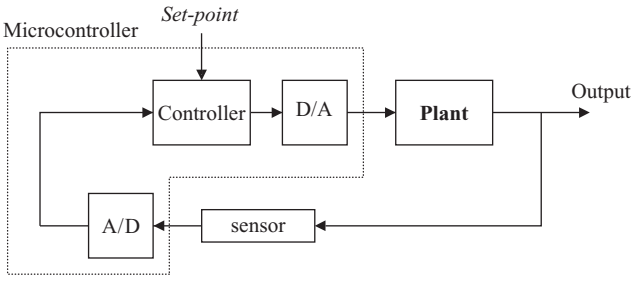

Figure 1.5 Another form of digital control

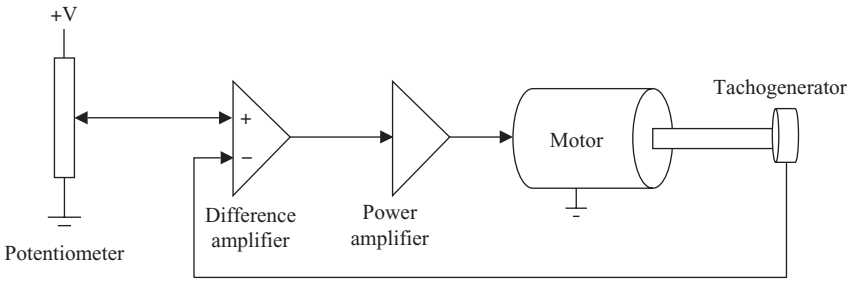

Figure 1.6 Typical analog speed control system

The purpose of developing the digital control theory is to be able to understand, design and build control systems where a computer is used as the controller in the system. In addition to the normal control task, a computer can perform supervisory functions, such as reading data from a keyboard, displaying data on a screen or liquid crystal display, turning a light or a buzzer on or off and so on.

Figure 1.6 shows a typical closed-loop analog speed control system where the desired speed of the motor is set using a potentiometer. A tachogenerator produces a voltage proportional to the speed of the motor, and this signal is used in a feedback loop and is subtracted from the desired value in order to generate the error signal. Based on this error signal the power amplifier drives the motor to obtain the desired speed. The motor will rotate at the desired speed as long as the error signal is zero.

The equivalent digital speed control system is shown in Figure 1.7. Here, the desired speed is entered from the keyboard into the digital controller. The controller also receives the converted output signal of the tachogenerator. The error signal is calculated by the controller by subtracting the tachogenerator reading from the desired speed. A D/A converter is then used to convert the signal into analog form and feed the power amplifier. The power amplifier then drives the motor.

Since the speed control can be achieved by using an analog approach, one is tempted to ask why use digital computers. Digital computers in 1960s were very large and very expensive devices and their use as controllers was not justified. They could only be used in very large

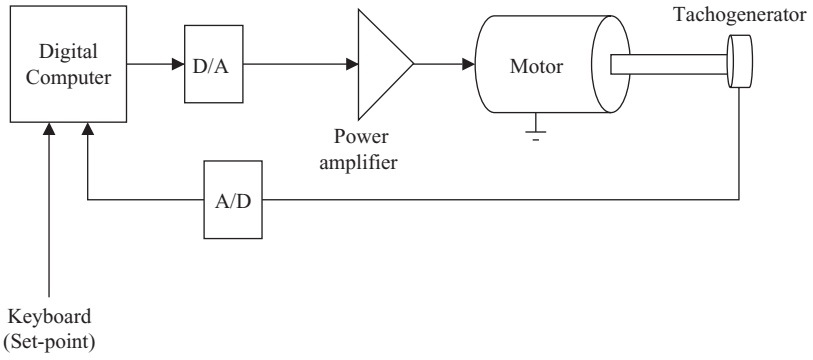

Figure 1.7 Digital speed control system

and expensive plants, such as large chemical processing plants or oil refineries. Since the introduction of microprocessors in the early 1970s the cost and size of digital computers have been greatly reduced. Early microprocessors, such as the Intel 8085 or the Mostek Z80, were very limited and required several chips before they could be used as processing elements. The required chips were read-only memory (ROM) to store the user program, random-access memory (RAM) to store the user data, input–output (I/O) circuitry, A/D and D/A converters, interrupt logic, and timer circuits. By the time all these chips were put together the chip count, power consumption, and complexity of the basic hardware were considerable. These controllers were in the form of microcomputers which could be used in many medium and large digital control applications.

Interest in digital control has grown rapidly in the last several decades since the introduction of *microcontrollers*. A microcontroller is a single-chip computer, including most of a computer's features, but in limited sizes. Today, there are hundreds of different types of microcontrollers, ranging from 8-pin devices to 40-pin, or even 64- or higher pin devices. For example, the PIC16F877 is an 8-bit, 40-pin microcontroller with the following features:

- operation up to 20 MHz;
- 8K flash program memory;
- 368 bytes RAM memory;
- 256 bytes electrically erasable programmable read-only memory (EEPROM) memory;
- 15 types of interrupts;
- 33 bits of parallel I/O capability;
- 2 timers;
- universal synchronous–asynchronous receiver/transmitter (USART) serial communications;
- 10-bit, 8-channel A/D converter;
- 2 analog comparators;
- 33 instructions;
- programming in assembly or high-level languages;
- low cost (approximately $10 each).

Flash memory is nonvolatile and is used to store the user program. This memory can be erased and reprogrammed electrically. EEPROM memory is used to store nonvolatile user data and can be written to or read from under program control. The microcontroller has 8K program memory, which is quite large for control based applications. In addition, the RAM memory is 368 bytes, which again is quite large for control based applications.

1.3 CENTRALIZED AND DISTRIBUTED CONTROL SYSTEMS

Until the beginning of 1980s, computer control was strictly *centralized*. Usually a single large computer or minicomputer (e.g. the DEC PDP11 series) was used to control the plant. The computer, associated power supplies, input–output, keyboard and display unit were all situated in a central location. The advantages of centralized control are as follows:

- It is easy to manage the computer.
- Only one computer is used.
- Less number of people are required.

In a centralized control system, the controller algorithm is implemented in a single central computer. Hence, all sensors, actuators, input units and output units must be connected directly to this central computer.

Today, *distributed control* is more widely used. A distributed control system (DCS) consists of a number of computers installed at different locations, each performing an independent control action. Distributed control has emerged as a result of the sharp decrease in price, and the consequent widespread use, of computers. Also, the development of computer networks has made it possible to interconnect computers in a local area network (LAN), as well as in a wide area network (WAN). The main advantages of DCSs are as follows:

- A higher performance is obtained from a distributed system than from a centralized control system.
- A distributed system is more reliable than a centralized system. In the case of a centralized system, if the computer fails, the whole plant becomes unusable. In a DCS, if one computer fails, only a small part of the plant will be affected and the load of the failed computer can usually be distributed among the other computers.
- A DCS can easily be expanded by adding more computers to the network. For example, if 10 computers are used to control the temperature of 10 ovens, then if the number of ovens is increased to 15, it is easy to add five more computers to the network.
- A DCS is more flexible than a centralized control system as it can be easily adjusted to plant requirements.

In a DCS the sensors and actuators can be connected to local computers which can execute localized controller algorithms. Thus, the local computers in a distributed control environment are usually used for direct digital control (DDC). In a DDC application the computer is used only to carry out the control action for the plant. It is also possible to add some level of supervisory control action to a DDC computer, such as displaying the values of sensors, inputs and outputs.

Distributed control systems are generally used as client–server systems. In such a system one computer (or more if necessary) is designated as the *server* and carries out the common control operations. Other computers in the system are called *clients* and they obey and implement the instructions they receive from the master computer. For example, the task of a client computer could be to receive and format analog data from a sensor, and then pass this data to the server computer every second.

Distributed control systems usually exist within finite boundaries, such as within a factory complex, and all the computers communicate with each other using a LAN cable. Wireless LAN systems are becoming popular, and there is no reason why a DCS cannot be constructed using wireless LAN technology. Using wireless, system reconfiguration is as easy as just adding or removing a computer.

1.4 SCADA SYSTEMS

The term SCADA is an abbreviation for *supervisory control and data acquisition*. SCADA systems integrate the data acquisition and system monitoring and control activities using graphical software packages. A SCADA system is nothing but a customized graphical applications

program with all the necessary hardware components. It can be developed using the popular visual programming languages such as *Visual C++* or *Visual Basic*. Good human–computer interface techniques should be employed in the design of the user interface. Alternatively, graphical programming languages such as *Labview* or *VisiDaq* can be used to create powerful, user-friendly SCADA systems.

In a SCADA system the user can have access to a graphical screen in order to monitor or change a setting in the plant. SCADA systems consist of both hardware and software and are usually implemented using personal computers (PCs). Typical hardware includes the computer, keyboard, touch screen, sensors and actuators. The software is in the form of a graphical user interface, where parts of the plant, sensor data and actuator data can all be displayed in various colours on a screen. The advantage of a SCADA system is that the user can easily monitor the status of the overall system. It is important that a SCADA system should be secure and password protected to avoid unauthorized access to the control screens.

1.5 HARDWARE REQUIREMENTS FOR COMPUTER CONTROL

1.5.1 General Purpose Computers

In general, although almost any digital computer can be used for digital control there are some requirements that should be satisfied before a computer is used for such an application. Today, the majority of small and medium scale DDC-type applications are based on microcontrollers which are used as embedded controllers. Applications where user interaction and supervisory control are required are commonly designed around the standard PC hardware.

As shown in Figure 1.8, a general purpose computer consists of the following basic building elements:

- central processing unit (CPU);
- program memory;
- data memory;
- input–output devices.

The CPU is the part which contains the arithmetic and logic unit (ALU), the control unit (CU) and the general purpose registers (GPR). The ALU consists of the logic circuitry necessary to carry out arithmetic and logic operations, for example to add or subtract numbers, to compare numbers and so on. Some ALU units are equipped to carry out multiplication and division and floating point mathematical operations. The CU supervises the operations within the CPU, fetches instructions from the program memory, decodes these instructions and controls the ALU and other parts of the computer so that the required operations can be implemented. The GPR are a set of fast registers which are generally used to carry out fast operations within the CPU.

The program memory of a general purpose computer is usually an external unit and attached to the computer via the data bus and the address bus. A bus is a collection of conductors which carry electrical signals. The data bus is a bidirectional bus which carries the data to be sent or received between the CPU and the other parts of the computer. The size of this bus is 8 bits in most microprocessors and microcontrollers. Some microcontrollers have data buses that are 16 or even 32 bits wide. Minicomputers and mainframe computers usually have 64 or even higher data widths. The address bus is a unidirectional bus which is used to address the peripheral

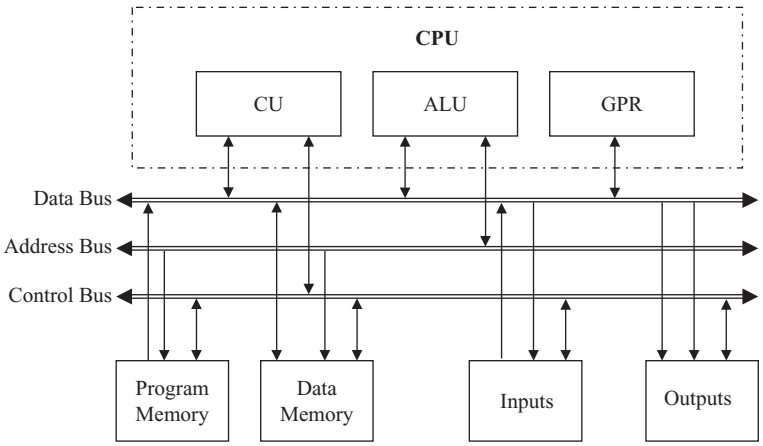

Figure 1.8 Schematic of a general purpose computer

devices attached to the computer. For example, when data is to be written to the memory the address of the memory location is sent on the address bus and the actual data byte is sent on the data bus. The program memory is usually a nonvolatile memory, such as electrically programmable read-only memory (EPROM), EEPROM or flash memory. EPROM memory can be programmed using a suitable programmer device. This type of memory has to be erased using an ultraviolet light source before the contents can be changed. EEPROM memory can be programmed and erased by sending electrical signals to the memory. The disadvantage of this memory is that it is usually a slow process to write or read data from an EEPROM memory. Currently, flash memory is one of the most popular types of nonvolatile memory used. Flash memory is fast and can be erased under program control.

The data memory is usually a volatile memory, used to store the user data. RAM type memories are commonly used for this purpose. The size of this memory can vary from several tens of kilobytes to tens of gigabytes.

Minicomputers and larger computers are equipped with auxiliary storage mediums such as hard disks and magnetic tapes. These devices provide bulk storage for programs and data. Magnetic tape is usually used to store the entire contents of a hard disk for backup purposes.

Input–output devices are also known as the peripheral devices. Many different types of input devices – scanner, camera, keyboard, microphone and mouse – can be connected to the computer. The output devices can be printers, plotters, speakers, visual display units and so on.

General purpose computers are usually more suited to data processing type applications. For example, a minicomputer can be used in an office to provide word processing. Similarly, a large computer can be used in a bank to store and manipulate the accounts of thousands of customers.

1.5.2 Microcontrollers

A microcontroller is a single-chip computer that is specifically manufactured for embedded computer control applications. These devices are very low-cost and can be used very easily

in digital control applications. Most microcontrollers have the built-in circuits necessary for computer control applications. For example, a microcontroller may have A/D converters so that the external signals can be sampled. They also have parallel input–output ports so that digital data can be read or output from the microcontroller. Some devices have built-in D/A converters and the output of the converter can be used to drive the plant through an actuator (e.g. an amplifier). Microcontrollers may also have built-in timer and interrupt logic. Using the timer or the interrupt facilities, we can program the microcontroller to implement the control algorithm accurately.

Microcontrollers have traditionally been programmed using the assembly language of the target device. As a result, the assembly languages of the microcontrollers manufactured by different firms are totally different and the user has to learn a new language before being able program a new type of device. Nowadays microcontrollers can be programmed using high-level languages such as BASIC, PASCAL or C. High-level languages offer several advantages compared to the assembly language:

- It is easier to develop programs using a high-level language.
- Program maintenance is much easier if the program is developed using a high-level language.
- Testing a program developed in a high-level language is much easier.
- High-level languages are more user-friendly and less prone to making errors.
- It is easier to document a program developed using a high-level language.

In addition to the above advantages, high-level languages have some disadvantages. For example, the length of the code in memory is usually larger when a high-level language is used, and the programs developed using the assembly language usually run faster than those developed using a high-level language.

In this book, PIC microcontrollers are used as digital controllers. The microcontrollers are programmed using the high-level C language.

1.6 SOFTWARE REQUIREMENTS FOR COMPUTER CONTROL

Computer hardware is nowadays very fast, and control computers are generally programmed using a high-level language. The use of the assembly language is reserved for very special and time-critical applications, such as fast, real-time device drivers. C is a popular language used in most computer control applications. It is a powerful language that enables the programmer to perform low-level operations, without the need to use the assembly language.

The software requirements in a control computer can be summarized as follows:

- the ability to read data from input ports;
- the ability to send data to output ports;
- internal data transfer and mathematical operations;
- timer interrupt facilities for timing the controller algorithm.

All of these requirements can be met by most digital computers, and, as a result, most computers can be used as controllers in digital control systems. The important point is that it is not justified and not cost-effective to use a minicomputer to control the speed of a motor, for example. A microcontroller is much more suitable for this kind of control application. On

the other hand, if there are many inputs and many outputs, and if it is required to provide supervisory tasks as well then the use of a minicomputer can easily be justified.

The controller algorithm in a computer is implemented as a program which runs continuously in a loop which is executed at the start of every sampling time. Inside the loop, the desired reference value is read, the actual plant output is also read, and the difference between the desired value and the actual value is calculated. This forms the error signal. The control algorithm is then implemented and the controller output for this sampling instant is calculated. This output is sent to a D/A converter which generates an analog equivalent of the desired control action. This signal is then fed to an actuator which in turn drives the plant to the desired point.

The operation of the controller algorithm, assuming that the reference input and the plant output are digital signals, is summarized below as a sequence of simple steps:

Repeat Forever

When it is time for next sampling instant

- Read the desired value, R
- Read the actual plant output, Y
- Calculate the error signal, $E = R - Y$
- Calculate the controller output, U
- Send the controller output to D/A converter
- Wait for the next sampling instant

End

Similarly, if the reference input and the plant output are analog signals, the operation of the controller algorithm can be summarized as:

Repeat Forever

When it is time for next sampling instant

- Read the desired value, R, from A/D converter
- Read the actual plant output, Y, from the A/D converter
- Calculate the error signal, $E = R - Y$
- Calculate the controller output, U
- Send the controller output to D/A converter
- Wait for the next sampling instant

End

One of the important features of the above algorithms is that once they have been started they run continuously until some event occurs to stop them or until they are stopped manually by an operator. It is important to make sure that the loop is run continuously and exactly at the same times, i.e. exactly at the sampling instants. This is called synchronization and there are several ways in which synchronization can be achieved in practice, such as:

- using polling in the control algorithm;
- using external interrupts for timing;
- using timer interrupts;

- ballast coding in the control algorithm;
- using an external real-time clock.

These methods are discussed briefly here.

1.6.1 Polling

Polling is the software technique where we keep waiting until a certain event occurs, and only then perform the required actions. This way, we wait for the next sampling time to occur and only then run the controller algorithm.

The polling technique is used in DDC applications since the controller cannot do any other operation during the waiting of the next sampling time. The polling technique is described below as a sequence of steps:

Repeat Forever
While Not sampling time
Wait
End

- Read the desired value, R
- Read the actual plant output, Y
- Calculate the error signal, $E = R - Y$
- Calculate the controller output, U
- Send the controller output to D/A converter

End

1.6.2 Using External Interrupts for Timing

The controller synchronization task can easily be performed using an external interrupt. Here, the controller algorithm can be written as an interrupt service routine (ISR) which is associated with an external interrupt. The external interrupt will typically be a clock with a period equal to the required sampling time. Thus, the computer will run the interrupt service (i.e. the algorithm) routine at every sampling instant. At the end of the ISR control is returned to the main program where the program either waits for the occurrence of the next interrupt or can perform other tasks (e.g. displaying data on a LCD) until the next external interrupt occurs.

The external interrupt approach provides accurate implementation of the control algorithm as far as the sampling time is concerned. One drawback of this method is that an external clock is required to generate the interrupt pulses.

The external interrupt technique has the advantage that the controller is not waiting and can perform other tasks in between the sampling instants.

The external interrupt technique of synchronization is described below as a sequence of steps:

Main program:
Wait for an external interrupt (or perform some other tasks)
End

Interrupt service routine (ISR):

- Read the desired value, R
- Read the actual plant output, Y
- Calculate the error signal, $E = R - Y$
- Calculate the controller output, U
- Send the controller output to D/A converter

Return from interrupt

1.6.3 Using Timer Interrupts

Another popular way to perform controller synchronization is to use the timer interrupt available on most microcontrollers. Here, the controller algorithm is written inside the timer interrupt service routine, and the timer is programmed to generate interrupts at regular intervals, equal to the sampling time. At the end of the algorithm control returns to the main program, which either waits for the occurrence of the next interrupt or performs other tasks (e.g. displaying data on an LCD) until the next interrupt occurs.

The timer interrupt approach provides accurate control of the sampling time. Another advantage of this technique is that no external hardware is required since the interrupts are generated by the internal timer of the microcontroller.

The timer interrupt technique of synchronization is described below as a sequence of steps:

Main program:
Wait for a timer interrupt (or perform some other tasks)
End
Interrupt service routine (ISR):

- Read the desired value, R
- Read the actual plant output, Y
- Calculate the error signal, $E = R - Y$
- Calculate the controller output, U
- Send the controller output to D/A converter

Return from interrupt

1.6.4 Ballast Coding

In this technique the loop timing is made to be independent of any external or internal timing signals. The method involves finding the execution time of each instruction inside the loop and then adding *dummy* code to make the loop execution time equal to the required sampling time.

This method has the advantage that no external or internal hardware is required. But one big disadvantage is that if the code inside the loop is changed, or if the CPU clock rate of the microcontroller is changed, then it will be necessary to readjust the execution timing of the loop.

The ballast coding technique of synchronization is described below as a sequence of steps. Here, it is assumed that the loop timing needs to be increased and some dummy code is added to the end of the loop to make the loop timing equal to the sampling time:

Do Forever:

- Read the desired value, R
- Read the actual plant output, Y
- Calculate the error signal, $E = R - Y$
- Calculate the controller output,U
- Send the controller output to D/A converter

Add dummy code
...
...
Add dummy code

End

1.6.5 Using an External Real-Time Clock

This technique is similar to using an external interrupt to synchronize the control algorithm. Here, some real-time clock hardware is attached to the microcontroller where the clock is updated at every *tick*; for example, depending on the clock used, 50 ticks will be equal to 1 s if the tick rate is 20 ms. The real-time clock is then read continuously and checked against the time for the next sample. Immediately on exiting from the wait loop the current value of the time is stored and then the time for the next sample is updated by adding the stored time to the sampling interval. Thus, the interval between the successive runs of the loop is independent of the execution time of the loop.

Although the external clock technique gives accurate timing, it has the disadvantage that real-time clock hardware is needed.

The external real-time clock technique of synchronization is described below as a sequence of steps. T is the required sampling time in ticks, which is set to n at the beginning of the algorithm. For example, if the clock rate is 50 Ticks per second, then a Tick is equivalent to 20 ms, and if the required sampling time is 100 ms, we should set $T = 5$:

$T = n$
Next_Sample_Time = Ticks + T
Do Forever:
While Ticks < Next_Sample_Time
Wait
End
Current_Time = Ticks

- Read the desired value, R
- Read the actual plant output, Y
- Calculate the error signal, $E = R - Y$
- Calculate the controller output, U

- Send the controller output to D/A converter
- Next_Sample_Time = Current_Time + T

End

1.7 SENSORS USED IN COMPUTER CONTROL

Sensors are an important part of closed-loop systems. A sensor is a device that outputs a signal which is related to the measurement of (i.e. is a function of) a physical quantity such as temperature, speed, force, pressure, displacement, acceleration, torque, flow, light or sound. Sensors are used in closed-loop systems in the feedback loops, and they provide information about the actual output of a plant. For example, a speed sensor gives a signal proportional to the speed of a motor and this signal is subtracted from the desired speed reference input in order to obtain the error signal.

Sensors can be classified as analog or digital. Analog sensors are more widely available, and their outputs are analog voltages. For example, the output of an analog temperature sensor may be a voltage proportional to the measured temperature. Analog sensors can only be connected to a computer by using an A/D converter. Digital sensors are not very common and they have logic level outputs which can directly be connected to a computer input port.

The choice of a sensor for a particular application depends on many factors such as the cost, reliability, required accuracy, resolution, range and linearity of the sensor. Some important factors are described below.

Range. The range of a sensor specifies the upper and lower limits of the measured variable for which a measurement can be made. For example, if the range of a temperature sensor is specified as 10–60 °C then the sensor should only be used to measure temperatures within that range.

Resolution. The resolution of a sensor is specified as the largest change in measured value that will not result in a change in the sensor's output, i.e. the measured value can change by the amount quoted by the resolution before this change can be detected by the sensor. In general, the smaller this amount the better the sensor is, and sensors with a wide range have less resolution. For example, a temperature sensor with a resolution of 0.001 K is better than a sensor with a resolution of 0.1 K.

Repeatability. The repeatability of a sensor is the variation of output values that can be expected when the sensor measures the same physical quantity several times. For example, if the voltage across a resistor is measured at the same time several times we may get slightly different results.

Linearity. An ideal sensor is expected to have a linear transfer function, i.e. the sensor output is expected to be exactly proportional to the measured value. However, in practice all sensors exhibit some amount of nonlinearity depending upon the manufacturing tolerances and the measurement conditions.

Dynamic response. The dynamic response of a sensor specifies the limits of the sensor characteristics when the sensor is subject to a sinusoidal frequency change. For example, the dynamic response of a microphone may be expressed in terms of the 3-dB bandwidth of its frequency response.

In the remainder of this chapter, the operation and the characteristics of some of the popular sensors are discussed.

1.7.1 Temperature Sensors

Temperature is one of the fundamental physical variables in most chemical and process control applications. Accurate and reliable measurement of the temperature is important in nearly all process control applications.

Temperature sensors can be analog or digital. Some of the most commonly used analog temperature sensors are: thermocouples, resistance temperature detectors (RTDs) and thermistors. Digital sensors are in the form of integrated circuits. The choice of a sensor depends on the accuracy, the temperature range, speed of response, thermal coupling, the environment (chemical, electrical, or physical) and the cost.

As shown in Table 1.1, thermocouples are best suited to very low and very high temperature measurements. The typical measuring range is from $-270\,^{\circ}$C to $+2600\,^{\circ}$C. In addition, thermocouples are low-cost, very robust, and they can be used in chemical environments. The typical accuracy of a thermocouple is $\pm 1\,^{\circ}$C. Thermocouples do not require external power for operation.

RTDs are used in medium-range temperature measurements, ranging from $-200\,^{\circ}$C to $+600\,^{\circ}$C. They can be used in most chemical environments but they are not as robust as thermocouples. The typical accuracy of RTDs is $\pm 0.2\,^{\circ}$C. They require external power for operation.

Thermistors are used in low- to medium-temperature applications, ranging from $-50\,^{\circ}$C to about $+200\,^{\circ}$C. They are not as robust as thermocouples or RTDs and they cannot easily be used in chemical environments. Thermistors are also low-cost devices, they require external power for operation, and they have an accuracy of $\pm 0.2\,^{\circ}$C.

Integrated circuit temperature sensors are used in low-temperature applications, ranging from $-40\,^{\circ}$C to $+125\,^{\circ}$C. These devices can be either analog or digital, and their coupling with the environment is not very good. The accuracy of integrated circuit sensors is around $\pm 1\,^{\circ}$C. Integrated temperature sensors differ from other sensors in some important ways:

- They are relatively small.
- Their outputs are highly linear.
- Their temperature range is limited.
- Their cost is very low.
- Some models include advanced features, such as thermostat functions, built-in A/D converters and so on.
- An external power supply is required to operate them.

Table 1.1 Temperature sensors

Sensor	Temperature range (°C)	Accuracy (±°C)	Cost	Robustness
Thermocouple	−270 to +2600	1	Low	Very high
RTD	−200 to +600	0.2	Medium	High
Thermistor	−50 to +200	0.2	Low	Medium
Integrated circuit	−40 to +125	1	Low	Low

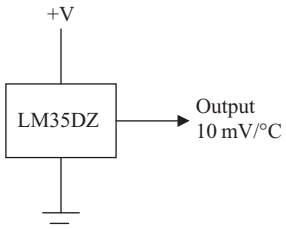

Figure 1.9 LM35DZ temperature sensor

Analog integrated circuit temperature sensors can be voltage output or current output devices. Voltage output sensors give a voltage which is directly proportional to the measured temperature. Similarly, current output sensors act as high-impedance current sources, giving an output current which is proportional to the temperature.

A popular voltage output analog integrated circuit temperature sensor is the LM35DZ, manufactured by National Semiconductors Inc. (see Figure 1.9). This is a 3-pin analog output sensor which provides a linear output voltage of 10 mV/°C. The temperature range is 0 °C to +100 °C, with an accuracy of ±1.5 °C.

The AD590 is an analog integrated circuit sensor with a current output. The device operates in the range −55 °C to +150 °C and produces an output current of 1 μA/°C.

Digital integrated temperature sensors produce digital outputs which can be interfaced to a computer. The output data format is usually nonstandard and the measured temperature can be extracted by using suitable algorithms. The DS1620 is a popular digital temperature sensor which also incorporates digitally programmable thermostat outputs. The device provides a 9-bit serial data to indicate the measured temperature. Data is extracted from the device by sending clock pulses and then reading the data after each pulse. Table 1.2 shows the sensor's measured temperature–output relationship.

There can be several sources of error during the measurement of temperature. Some important possible errors are described below.

Sensor self-heating. RTDs, thermistors and integrated circuit sensors require an external power supply for their operation. The power supply can cause the sensor to heat, leading to an error in the measurement. The effect of self-heating depends on the size of the sensor and the amount of power dissipated by the sensor. Self-heating can be avoided by using the lowest possible external power, or by considering the heating effect in the measurement.

Table 1.2 Temperature–data relationship of DS1620

Temperature (°C)	Digital output
+125	0 11111010
+25	0 00110010
0.5	0 00000001
0	0 00000000
−0.5	1 11111111
−25	1 11001110
−55	1 10010010

Electrical noise. Electrical noise can introduce errors into the measurement. Thermocouples produce very low voltages (of the order of tens of microvolts) and, as a result, noise can easily enter the measurement. This noise can usually be minimized by using low-pass filters, and by keeping the sensor leads as short as possible and away from motors and other electrical machinery.

Mechanical stress. Some sensors such as RTDs are sensitive to mechanical stress and should be used carefully. Mechanical stress can be minimized by avoiding deformation of the sensor.

Thermal coupling. It is important that for accurate and fast measurements the sensor should make a good contact with the measuring surface. If the surface has a thermal gradient then incorrect placement of the sensor can lead to errors. If the sensor is used in a liquid, the liquid should be stirred to cause a uniform heat distribution. Integrated circuit sensors usually suffer from thermal coupling since they are not easily mountable on surfaces.

Sensor time constant. The response time of the sensor can be another source of error. Every type of sensor takes a finite time to respond to a change in its environment. Errors due to the sensor time constant can be minimized by improving the coupling between the sensor and the measuring surface.

1.7.2 Position sensors

Position sensors are used to measure the position of moving objects. These sensors are basically of two types: sensors to measure linear movement, and sensors to measure angular movement.

Potentiometers are available in linear and rotary forms. In a typical application, a fixed voltage is applied across the potentiometer and the voltage across the potentiometer arm is measured. This voltage is proportional to the position of the arm, and hence by measuring the voltage we know the position of the arm. Figure 1.10 shows a linear potentiometer. If the applied voltage is V_i, the voltage across the arm is given by

$$V_a = kV_i y$$

where y is the position of the arm from the beginning of the potentiometer, and k is a constant.

Figure 1.11 shows a rotary potentiometer which can be used to measure angular position. If V_i is again the applied voltage, the voltage across the arm is given by

$$V_a = kV_i\theta$$

where θ is the angle of the arm, and k is a constant.

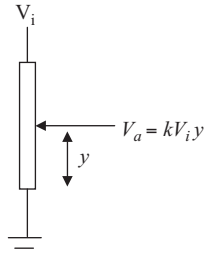

Figure 1.10 Linear potentiometer

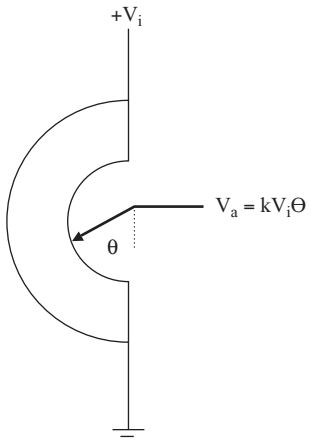

Figure 1.11 Rotary potentiometer

Potentiometer type position sensors are low-cost, but they have the disadvantage that the range is limited and also that the sensor can be worn out by excessive movement of the arm.

Among other types of position sensors are capacitive sensors, inductive sensors, linear variable differential transformers (LVDTs) and optical encoders. Capacitive position sensors rely on the fact that the capacitance of a parallel plate capacitor changes as the distance between the plates is changed. The formula for the capitance, C, of a parallel plate capacitor is

$$C = \varepsilon \frac{A}{d}, \tag{1.1}$$

where ε is the dielectric constant, A the area of the plates and d the distance between the plates.

Typically, the capacitor of the sensor is used in the feedback loop of an operational amplifier as shown in Figure 1.12, and a reference capacitor is used at the input. If a voltage V_i is applied, the output voltage V_o is given by

$$V_o = -V_i \frac{C}{C_{\text{ref}}}. \tag{1.2}$$

Using equation (1.1), we obtain

$$V_o = -V_i \frac{C_{\text{ref}} d}{\varepsilon A}. \tag{1.3}$$

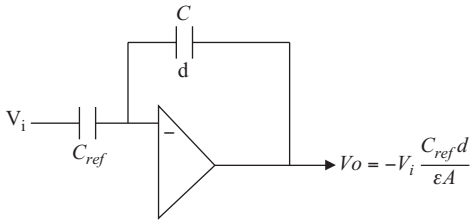

Figure 1.12 Position sensor using capacitors

Figure 1.13 Commercially available LVDT sensor

From (1.3), if we apply a constant amplitude sinusoidal signal as the input, the amplitude of the output voltage is proportional to the distance between the plates.

LVDT sensors (see Figure 1.13) consist of one primary and two secondary windings on a hollow cylinder. The primary winding is in the middle, and the secondary windings have equal number of turns, series coupled, and they are at the ends of the cylinder (see Figure 1.14). A sinusoidal signal with a voltage of 0.5–5 V and frequency 1–20 kHz is applied to the primary winding. A magnetic core which measures the position moves inside the cylinder, and the movement of this core varies the magnetic field linking the primary winding to the secondary windings. Because the secondary windings are in opposition, the movement of the core to one position increases the induced voltage in one secondary coil and decreases the induced voltage in the other secondary coil. The net voltage difference is proportional to the position of the core inside the cylinder. Thus, by measuring the induced voltage we know the position of the core. The strong relationship between the core position and the induced voltage yields a design that exhibits excellent resolution. Most commercially available LVDTs come with built-in signal-conditioning circuitry that provides an easy interface to a computer. The device operates from a d.c. supply and the signal conditioner provides the a.c. signal required for the operation of the circuit, as well as the demodulation of the output signal to give a useful d.c.

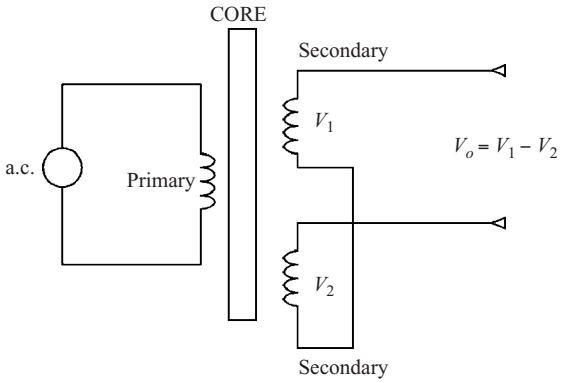

Figure 1.14 LVDT sensor circuit diagram

voltage output. The range of an LVDT is from $\pm 125\ \mu$m to ± 75 mm and the sensitivity ranges from 0.6 to 30 mV per 25 μm under normal excitation of 3–6 V.

The advantages of LVDT are:

- low cost;
- robust design;
- no hysteresis effect;
- fast response time;
- no friction resistance;
- long life.

The main disadvantage of the LVDT is that the core must have direct contact with the measured surface, which may not always be possible.

1.7.3 Velocity and acceleration sensors

Velocity is the differentiation of position, and in general position sensors can be used to measure velocity. The required differentiation can be done either in hardware (e.g. using operational amplifiers) or by the computer. For more accurate measurements velocity sensors should be used. There are two types of velocity sensors: linear sensors, and rotary sensors.

Linear velocity sensors can be constructed using a pair of coils and a moving magnet. When the coils are connected in series, the movement of the magnet produces additive voltage which is proportional to the movement of the magnet.

One of the most widely used rotary velocity sensors is the tachometer (or tachogenerator). A tachometer (see Figure 1.15) is connected to the shaft of a rotating device (e.g. a motor) and produces an analog d.c. voltage which is proportional to the speed of the shaft. If ω is the angular velocity of the shaft, the output voltage of the tachometer is given by

$$V_o = k\omega,$$

where k is the gain constant of the tachometer.

Another popular velocity sensor is the optical encoder. This basically consists of a light source and a disk with opaque and transparent sections where the disk is attached to the rotating shaft. A light sensor at the other side of the wheel detects light and a pulse is produced when the transparent section of the disk comes round. The encoder's controller counts the pulses in a given time, and this is proportional to the speed of the shaft. Figure 1.16 shows a typical commercial encoder.

Figure 1.15 Commercially available tachometer

Figure 1.16 Commercially available encoder

Acceleration is the differentiation of velocity, or the double differentiation of position. Thus, in general, position sensors can be used to measure acceleration. The differentiation can be done either by using operational amplifiers or by a computer program. For accurate measurement of the acceleration, semiconductor accelerometers can be used. For example, the ADXL202 is an accelerometer chip manufactured by Analog Devices Inc. This is a low-cost 8-pin chip with two outputs to measure the acceleration in two dimensions. The outputs are digital signals whose duty cycles are proportional to the acceleration in each of the two axes. These outputs can be connected directly to a microcontroller and the acceleration can be measured very easily, requiring no A/D converter. The measurement range of the ADXL202 is $\pm 2\,g$, where g is acceleration due to gravity, and the device can measure both dynamic acceleration (e.g. vibration), and static acceleration (e.g. gravity).

1.7.4 Force sensors

Force sensors can be constructed using position sensors. Alternatively, a strain gauge can be used to measure force accurately. There are many different types of strain gauges. A strain gauge can be made from capacitors and inductors, but the most widely used types are made from resistors. A wire strain gauge is made from a resistor, in the form of a metal foil. The principle of operation is that the resistance of a wire increases with increasing strain and decreases with decreasing strain.

In order to measure strain with a strain gauge, it must be connected to an electrical circuit, and a Wheatstone bridge is commonly used to detect the small changes in the resistance of the strain gauge.

Strain gauges can be used to measure force, load, weight pressure, torque or displacement.

Force can also be measured using the principle of piezoelectricity. A piezoelectric sensor produces voltage when a force is applied to its surface. The disadvantage of this method is that the voltage decays after the application of the force and thus piezoelectric sensors are only useful for measuring dynamic force.

1.7.5 Pressure sensors

Early pressure measurement was based on using a flexible device (e.g. a diaphragm) as a sensor; the pressure changed as the device moved and caused a dial connected to the device to move

and indicate the pressure. Nowadays, the movement is converted into an electrical signal which is proportional to the applied pressure. Strain gauges, capacitance change, inductance change, piezoelectric effect, optical pressure sensors and similar techniques are used to measure the pressure.

1.7.6 Liquid sensors

There are many different types of liquid sensors. These sensors are used to:

- detect the presence of liquid;
- measure the level of liquid;
- measure the flow rate of liquid, for example through a pipe.

The presence of a liquid can be detected by using optical, ultrasonic, change of resistance, change of capacitance or similar techniques. For example, optical technique is based on using an LED and a photo-transistor, both housed within a plastic dome and at the head of the device. When no liquid is present, light from the LED is internally reflected from the dome to the photo-transistor and the output is designed to be off. When liquid is present the dome is covered with liquid and the refractive index at the dome–liquid boundary changes, allowing some light to escape from the LED. As a result of this, the amount of light received by the photo-transistor is reduced and the output is designed to switch on, indicating the presence of liquid.

The level of liquid in a tank can be measured using immersed sensor techniques, or non-touching ultrasonic techniques. The simplest technique is to immerse a rod in the liquid with a potentiometer placed inside the rod. The potentiometer arm is designed to move as the level of the liquid is changed. The change in the resistance can be measured and hence the level of the liquid is obtained.

Pressure sensors are also used to measure the level of liquid in a tank. Typically, the pressure sensor is mounted at the bottom of the tank where change of pressure is proportional to the height of the liquid. These sensors usually give an analog output voltage proportional to the height of the liquid inside the tank.

Nontouching ultrasonic level measurement is very accurate, but more expensive than the other techniques. Basically, an ultrasonic beam is sent to the surface of the water and the echo of the beam is detected. The time difference between sending the beam and the echo is proportional to the level of the liquid in the tank.

The liquid flow rate can be measured by several techniques:

- paddlewheel sensors;
- displacement flow meters;
- magnetic flow meters;

Paddlewheel sensors are cost-effective and very popular for the measurement of liquid flow rate. A wheel is mounted inside the sensor whose speed of rotation is proportional to the flow rate. As the wheel rotates a voltage is produced which indicates the flow rate.

Displacement flow meters measure the flow rate of a liquid by separating the flow into known volumes and counting them over time. These meters provide good accuracy. Displacement flow meters have several types such as sliding vane meters, rotary piston meters, helix flow meters and so on.

Figure 1.17 Commercially available magnetic flow rate sensor (Sparling Instruments Inc.)

Magnetic flow meters are based on Faraday's law of magnetic induction. Here, the liquid acts as a conductor as it flows through a pipe. This induces a voltage which is proportional to the flow rate. The faster the flow rate, the higher is the voltage. This voltage is picked up by the sensors mounted in the meter tube and electronic means are used to calculate the flow rate based on the cross-sectional area of the tube. Advantages of magnetic flow rates are as follows:

- Corrosive liquids can be used.
- The measurement does not change the flow stream.

Figure 1.17 shows a typical magnetic flow meter.

1.7.7 Air flow sensors

Air flow is usually measured using anemometers. A classical anemometer (see Figure 1.18) has a rotating vane, and the speed of rotation is proportional to the air flow. Hot wire anemometers

Figure 1.18 Classical anemometer

Figure 1.19 Hot wire anemometer (Extech Instruments Corp.)

have no moving parts (Figure 1.19). The sensor consists of an electrically heated platinum wire which is placed in the air flow. As the flow velocity increases the rate of heat flow from the heated wire to the flow stream increases and a cooling occurs on the electrode, causing its resistance to change. The flow rate is then determined from the change in the resistance.

1.8 EXERCISES

1. Describe what is meant by accuracy, range and resolution.
2. Give an example of how linear displacement can be measured.
3. A tachometer is connected to a motor shaft in a speed control system. If the tachometer produces 100 mV per revolution, write an expression for its transfer function in terms of volts/radians-per-second.
4. What is polling in software? Describe how a control algorithm can be synchronized using polling.
5. What are the differences between polling and interrupt based processing? Which method is more suitable in digital controller design.
6. Explain why an A/D converter may be required in digital control systems.

7. Explain the operation of an LVDT sensor. You are required to design a motor position control system. Explain what type of position transducer you would use in your design.
8. You are required to measure the flow rate of water entering into a tank. Explain what type of sensor you can use.
9. Water enters into a tank through a pipe. At the same time, a certain amount of water is output from the tank continuously. You are required to design a water level control system so that the level of water in the tank is kept constant at all times. Draw a sketch of a suitable control system. Explain what types of sensors you will be using in the design.
10. What is a paddlewheel? Explain the operation principles of a paddlewheel liquid flow meter. What are the advantages and disadvantages of this sensor?
11. Compare the vane based anemometer and the hot air anemometer. Which one would you choose to measure the air flow through a narrow pipe?
12. Explain the factors that should be considered before purchasing and using a sensor.

FURTHER READING

[Bennett, 1994] Bennett, S. Real-time Computer Control: An Introduction. Prentice Hall, Hemel Hempstead, 1994.

[D'Souza, 1988] D'Souza, A.F. Design of Control Systems. Prentice Hall, Englewood Cliffs, NJ, 1988.

[Nise, 2000] Nise, N.S. Control Systems Engineering, 3rd edn., John Wiley & Sons, Inc., New York, 2000.