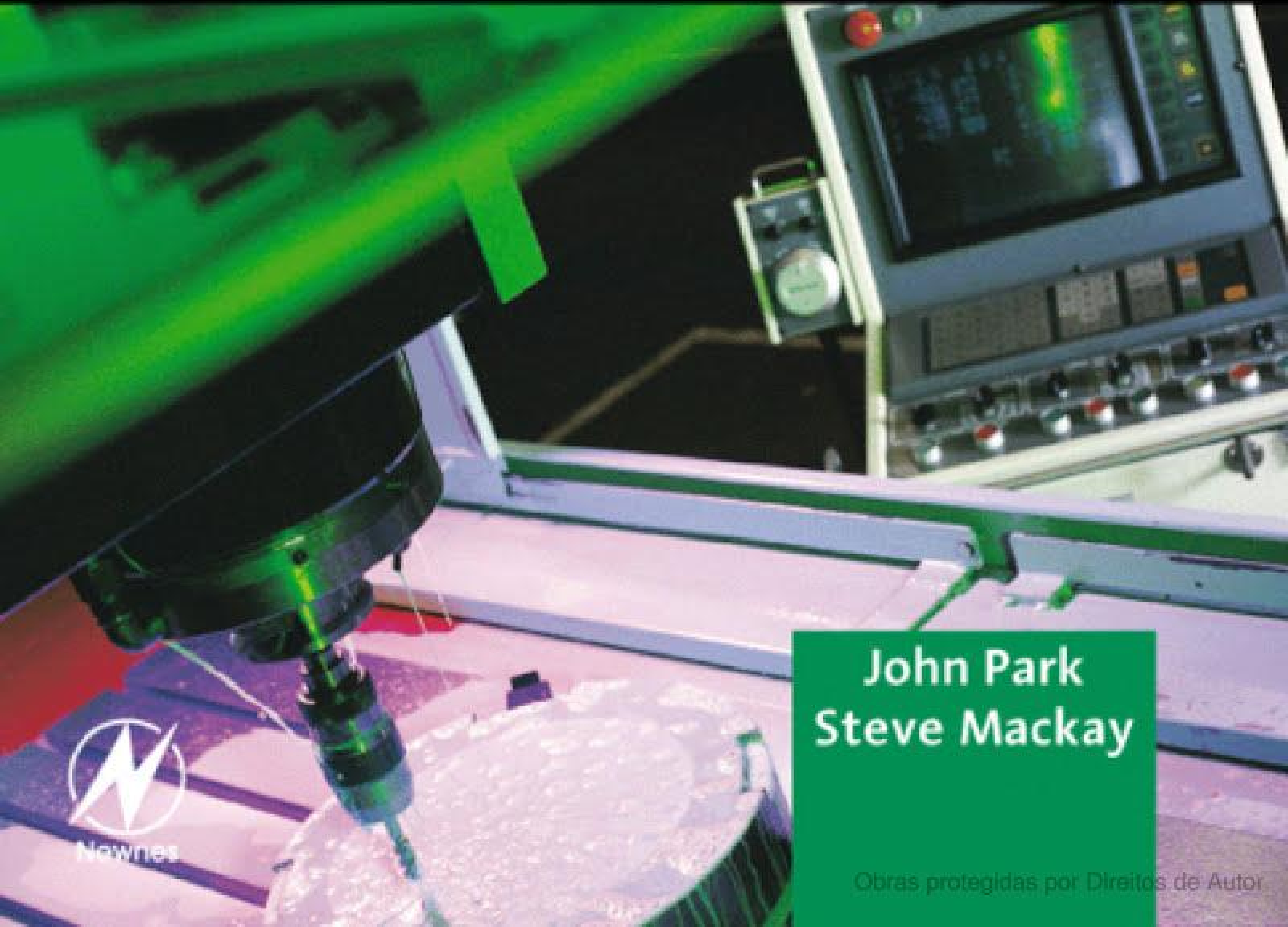
ELSEVIER

Practical
Data Acquisition for Instrumentation and Control Systems



John Park
Steve Mackay

Newnes

Practical Data Acquisition for Instrumentation and Control Systems

John Park ASD, IDC Technologies, Perth, Australia

Steve Mackay CPEng, BSc(ElecEng), BSc(Hons), MBA, IDC Technologies, Perth, Australia

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON • NEW YORK • OXFORD
PARIS • SAN DIEGO • SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Newnes is an imprint of Elsevier

Newnes

Newnes
An imprint of Elsevier
Linacre House, Jordan Hill, Oxford OX2 8DP
200 Wheeler Road, Burlington, MA 01803

First published 2003

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN 07506 57960

Typeset and Edited by Vivek Mehra, Mumbai, India
(vivekmehra@tatanova.com)

Printed and bound in Great Britain

Contents

3 Signal conditioning 36

4 The PC for real time work 67

10 The universal serial bus (USB) 271

11 Specific techniques 285

12 The PCMCIA Card 292

Appendix D Review of the Intel 8254 timer-counter chip 364

Appendix E Thermocouple tables 377

1

Introduction

In 1981, when IBM released its first personal computer or PC (as it became widely known) its open system design encouraged the development of a wide range of compatible add-on products by independent third party developers. In addition, the open system design has encouraged the proliferation of IBM compatible PCs in the market place, resulting in a rapid increase in the speed and power of the PC, as competitors vie for a market edge.

Accompanied by a significant drop in cost and a rapid expansion in software, which utilizes the increased power of the processor, the PC is now the most widely used platform for digital signal processing, image processing, data acquisition, and industrial control and communication applications. In many applications, indeed for data acquisition and process control, the PCs power and flexibility allow it to be configured in a number of ways, each with its own distinct advantages. The key to the effective use of the PC is the careful matching of the specific requirements of a particular data acquisition application to the appropriate hardware and software available.

This chapter reviews the fundamental concepts of data acquisition and control systems and the various system configurations, which make use of the PC.

1.1 Definition of data acquisition and control

Data acquisition is the process by which physical phenomena from the real world are transformed into electrical signals that are measured and converted into a digital format for processing, analysis, and storage by a computer.

In a large majority of applications, the data acquisition (DAQ) system is designed not only to acquire data, but to act on it as well. In defining DAQ systems, it is therefore useful to extend this definition to include the control aspects of the total system. Control is the process by which digital control signals from the system hardware are convened to a signal format for use by control devices such as actuators and relays. These devices then control a system or process. Where a system is referred to as a data acquisition system or DAQ system, it is possible that it includes control functions as well.

1.2 Fundamentals of data acquisition

A data acquisition and control system, built around the power and flexibility of the PC, may consist of a wide variety of diverse hardware building blocks from different equipment manufacturers. It is the task of the system integrator to bring together these individual components into a complete working system.

The basic elements of a data acquisition system, as shown in the functional diagram of Figure 1.1, are as follows:

- Sensors and transducers
- Field wiring
- Signal conditioning
- Data acquisition hardware
- PC (operating system)
- Data acquisition software

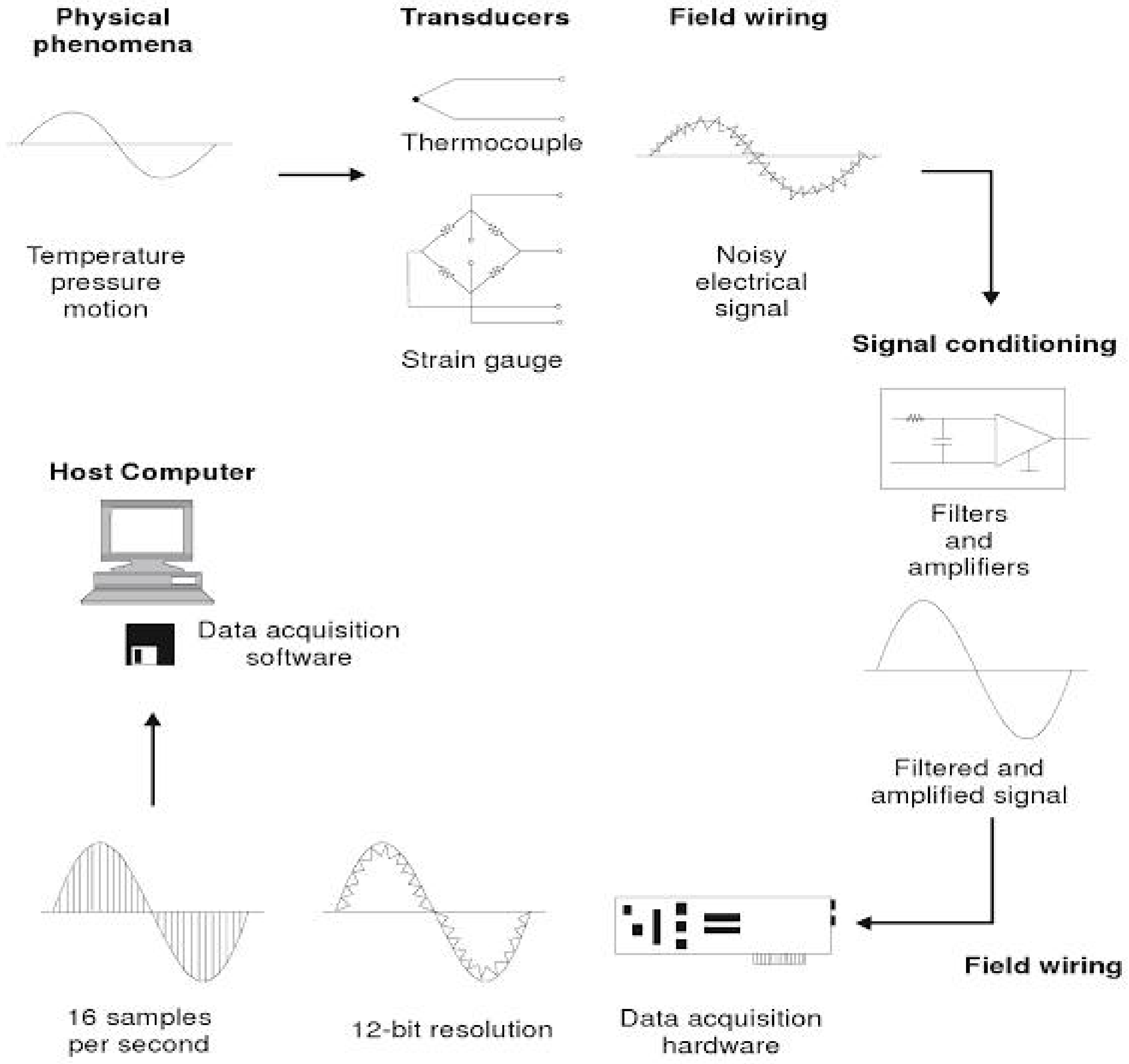


Figure 1.1
Functional diagram of a PC-based data acquisition system

Each element of the total system is important for the accurate measurement and collection of data from the process or physical phenomena being monitored, and is discussed in the following sections.

1.2.1 Transducers and sensors

Transducers and sensors provide the actual interface between the real world and the data acquisition system by converting physical phenomena into electrical signals that the signal conditioning and/or data acquisition hardware can accept.

Transducers available can perform almost any physical measurement and provide a corresponding electrical output. For example, thermocouples, resistive temperature detectors (RTDs), thermistors, and IC sensors convert temperature into an analog signal, while flow meters produce digital pulse trains whose frequency depends on the speed of flow.

Strain gauges and pressure transducers measure force and pressure respectively, while other types of transducers are available to measure linear and angular displacement, velocity and acceleration, light, chemical properties (e.g. CO concentration, pH), voltages, currents, resistances or pulses. In each case, the electrical signals produced are proportional to the physical quantity being measured according to some defined relationship.

1.2.2 Field wiring and communications cabling

Field wiring represents the physical connection from the transducers and sensors to the signal conditioning hardware and/or data acquisition hardware. When the signal conditioning and/or data acquisition hardware is remotely located from the PC, then the field wiring provides the physical link between these hardware elements and the host computer. If this physical link is an RS-232 or RS-485 communications interface, then this component of the field wiring is often referred to as communications cabling.

Since field wiring and communications cabling often physically represents the largest component of the total system, it is most susceptible to the effects of external noise, especially in harsh industrial environments. The correct earthing and shielding of field wires and communications cabling is of paramount importance in reducing the effects of noise. This passive component of the data acquisition and control system is often overlooked as an important integral component, resulting in an otherwise reliable system becoming inaccurate or unreliable due to incorrect wiring techniques.

1.2.3 Signal conditioning

Electrical signals generated by transducers often need to be converted to a form acceptable to the data acquisition hardware, particularly the A/D converter which converts the signal data to the required digital format. In addition, many transducers require some form of excitation or bridge completion for proper and accurate operation.

The principal tasks performed by signal conditioning are:

- Filtering
- Amplification
- Linearization
- Isolation
- Excitation

Filtering

In noisy environments, it is very difficult for very small signals received from sensors such as thermocouples and strain gauges (in the order of mV), to survive without the sensor data being compromised. Where the noise is of the same or greater order of magnitude than the required signal, the noise must first be filtered out. Signal conditioning equipment often contains low pass filters designed to eliminate high frequency noise that can lead to inaccurate data.

Amplification

Having filtered the required input signal, it must be amplified to increase the resolution. The maximum resolution is obtained by amplifying the input signal so that the maximum voltage swing of the input signal equals the input range of the analog-to-digital converter (ADC), contained within the data acquisition hardware.

Placing the amplifier as close to the sensor as physically possible reduces the effects of noise on the signal lines between the transducer and the data acquisition hardware.

Linearization

Many transducers, such as thermocouples, display a non-linear relationship to the physical quantity they are required to measure. The method of linearizing these input signals varies between signal conditioning products. For example, in the case of thermocouples, some products match the signal conditioning hardware to the type of thermocouple, providing hardware to amplify and linearize the signal at the same time.

A cheaper, easier, and more flexible method is provided by signal conditioning products that perform the linearization of the input signal using software.

Isolation

Signal conditioning equipment can also be used to provide isolation of transducer signals from the computer where there is a possibility that high voltage transients may occur within the system being monitored, either due to electrostatic discharge or electrical failure. Isolation protects expensive computer equipment from damage and computer operators from injury. In addition, where common-mode voltage levels are high or there is a need for extremely low common mode leakage current, as for medical applications, isolation allows measurements to be accurately and safely obtained.

Excitation

Signal conditioning products also provide excitation for some transducers. For example: strain gauges, thermistors and RTDs, require external voltage or current excitation signals.

1.2.4 Data acquisition hardware

Data acquisition and control (DAQ) hardware can be defined as that component of a complete data acquisition and control system, which performs any of the following functions:

- The input, processing and conversion to digital format, using ADCs, of analog signal data measured from a system or process – the data is then transferred to a computer for display, storage and analysis
- The input of digital signals, which contain information from a system or process

- The processing, conversion to analog format, using DACs, of digital signals from the computer – the analog control signals are used for controlling a system or process
- The output of digital control signals

Data acquisition hardware is available in many forms from many different manufacturers. Plug-in expansion bus boards, which are plugged directly into the computer's expansion bus, are a commonly utilized item of DAQ hardware. Other forms of DAQ hardware are intelligent stand-alone loggers and controllers, which can be monitored, controlled and configured from the computer via an RS-232 interface, and yet can be left to operate independently of the computer.

Another commonly used item of DAQ hardware, especially in R&D and test environments, is the remote stand-alone instrument that can be configured and controlled by the computer, via the IEEE-488 communication interface. Several of the most common DAQ system configurations are discussed in the section **Data acquisition and control system configuration p. 6**

1.2.5 Data acquisition software

Data acquisition hardware does not work without software, because it is the software running on the computer that transforms the system into a complete data acquisition, analysis, display, and control system.

Application software runs on the computer under an operating system that may be single-tasking (like DOS) or multitasking (like Windows, Unix, OS2), allowing more than one application to run simultaneously.

The application software can be a full screen interactive panel, a dedicated input/output control program, a data logger, a communications handler, or a combination of all of these.

There are three options available, with regard to the software required, to program any system hardware:

- Program the registers of the data acquisition hardware directly
- Utilize low-level driver software, usually provided with the hardware, to develop a software application for the specific tasks required
- Utilize off-the-shelf application software – this can be application software, provided with the hardware itself, which performs all the tasks required for a particular application; alternatively, third party packages such as LabVIEW and Labtech Notebook provide a graphical interface for programming the tasks required of a particular item of hardware, as well as providing tools to analyze and display the data acquired

1.2.6 Host computer

The PC used in a data acquisition system can greatly affect the speeds at which data can be continuously and accurately acquired, processed, and stored for a particular application. Where high speed data acquisition is performed with a plug-in expansion board, the throughput provided by bus architectures, such as the PCI expansion bus, is higher than that delivered by the standard ISA or EISA expansion bus of the PC.

Depending on the particular application, the microprocessor speed, hard disk access time, disk capacity and the types of data transfer available, can all have an impact on the speed at which the computer is able to continuously acquire data. All PCs, for example, are capable of programmed I/O and interrupt driven data transfers. The use of Direct Memory Access (DMA), in which dedicated hardware is used to transfer data directly into the computer's memory, greatly increases the system throughput and leaves the computer's microprocessor free for other tasks. Where DMA or interrupt driven data transfers are required, the plug-in data acquisition board must be capable of performing these types of data transfer.

In normal operation the data acquired, from a plug-in data acquisition board or other DAQ hardware (e.g. data logger), is stored directly to System Memory. Where the available system memory exceeds the amount of data to be acquired, data can be transferred to permanent storage, such as a hard disk, at any time. The speed at which the data is transferred to permanent storage does not affect the overall throughput of the data acquisition system.

Where large amounts of data need to be acquired and stored at high speed, disk-streaming can be used to continuously store data to hard disk. Disk-streaming utilizes a terminate-and-stay-resident (TSR) program to continuously transfer data acquired from a plug-in data acquisition board and temporarily held in system memory, to the hard disk. The limiting factors in the streaming process may be the hard disk access time and its storage capacity. Where the storage capacity is sufficient, the amount of contiguous (unfragmented) free hard disk space available to hold the data, may affect the system performance, since the maximum rate at which data can be streamed to the disk is reduced by the level of fragmentation.

If real-time processing of the acquired data is needed, the performance of the computer's processor is paramount. A minimum requirement for high frequency signals acquired at high sampling rates would be a 32-bit processor with its accompanying co-processor, or alternatively a dedicated plug-in processor. Low frequency signals, for which only a few samples are processed each second, would obviously not require the same level of processing power. A low-end PC would therefore be satisfactory. Clearly, the performance requirements of the host computer must be matched to the specific application. As with all aspects of a data acquisition system the choice of computer is a compromise between cost and the current and future requirements it must meet.

One final aspect of the personal computer that should be considered is the type of operating system installed. This may be single-tasking (e.g. MS-DOS) or multitasking (e.g. Windows 2000). While the multitasking nature of Windows provides many advantages for a wide range of applications, its use in data acquisition is not as clear-cut. For example, the methods employed by Windows to manage memory can provide difficulties in the use of DMA. In addition, interrupt latencies introduced by the multitasking nature of Windows can lead to problems when interrupt driven data transfers are used. Therefore, careful consideration must be given to the operating system and its performance in relation to the type of data acquisition hardware and the methods of data transfer, especially where high-speed data transfers are required.

1.3 Data acquisition and control system configuration

In many applications, and especially for data acquisition and process control, the power and flexibility of the PC, allows DAQ systems to be configured in a number of ways, each with its own distinct advantages. The key to the effective use of the PC is the careful matching of the specific requirements of a particular data acquisition application to the appropriate hardware and software available.

The choice of hardware, and the system configuration, is largely dictated by the environment in which the system will operate (e.g. an R&D laboratory, a manufacturing plant floor or a remote field location). The number of sensors and actuators required and their physical location in relation to the host computer, the type of signal conditioning required, and the harshness of the environment, are key factors.

Several of the most common system configurations are as follows:

- Computer plug-in I/O
- Distributed I/O
- Stand-alone or distributed loggers and controllers
- IEEE-488 instruments

1.3.1 Computer plug-in I/O

Plug-in I/O boards are plugged directly into the computers expansion bus, are generally compact, and also represent the fastest method of acquiring data to the computers memory and/or changing outputs. Along with these advantages, plug-in boards often represent the lowest cost alternative for a complete data acquisition and control system and are therefore a commonly utilized item of DAQ hardware.

As shown in Figure 1.2, examples of plug-in I/O boards are, multiple analog input A/D boards, multiple analog output D/A boards, digital I/O boards, counter/timer boards, specialized controller boards (such as stepper/servo motor controllers) or specialized instrumentation boards (such as digital oscilloscopes).

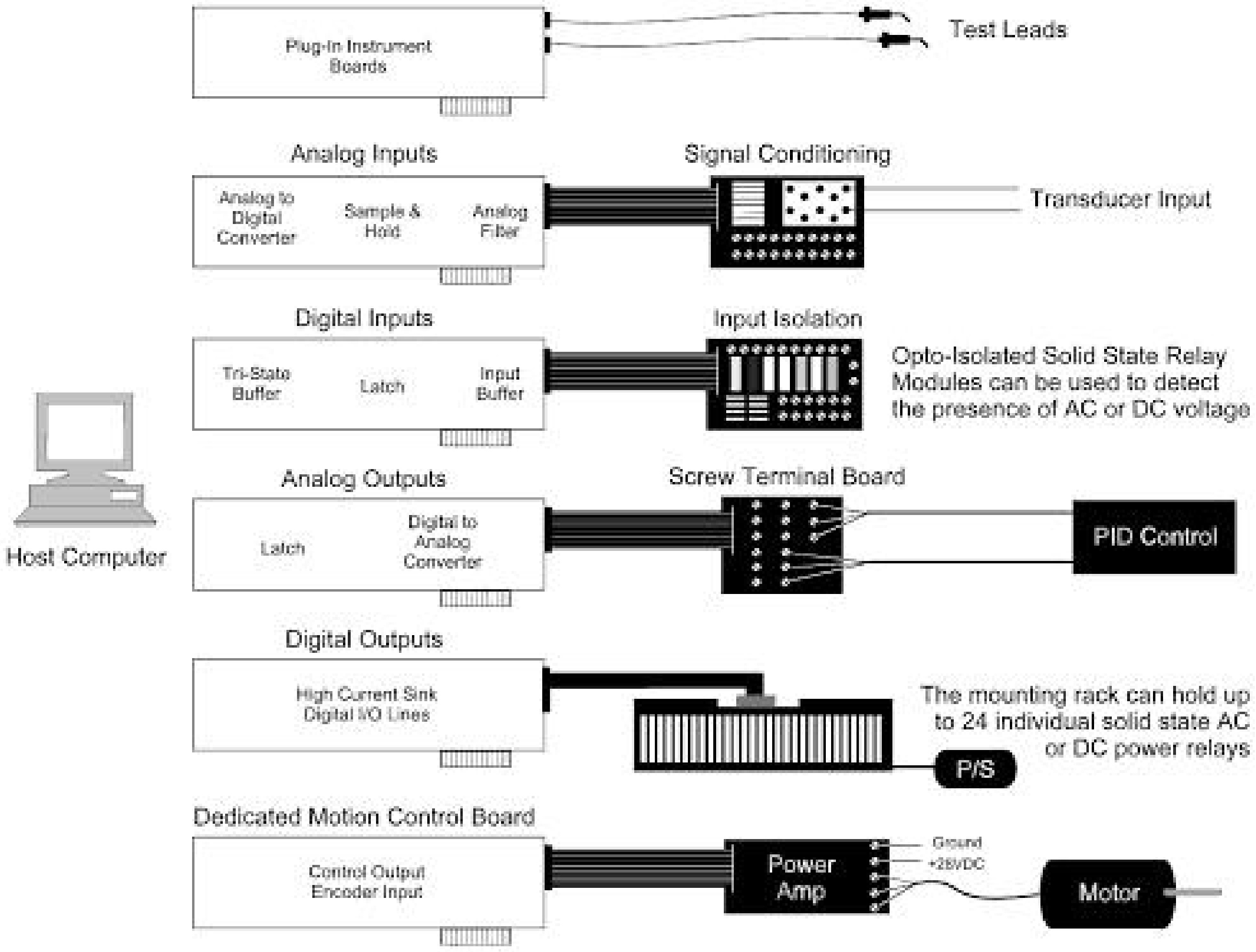


Figure 1.2
Example of computer plug-in I/O boards

Multi-function DAQ boards, containing A/D converters (ADCs), D/A converters (DACs), digital I/O ports and counter timer circuitry, perform all the functions of the equivalent individual specialized boards. Depending on the number of analog inputs/outputs and digital inputs/outputs required for a particular application, multi-function boards represent the most cost effective and flexible solution for DAQ systems.

Plug-in expansion boards are commonly used in applications where the computer is close to the sensors being measured or the actuators being controlled. Alternatively, they can be interfaced to remotely located transducers and actuators via signal conditioning modules known as two-wire transmitters. This system configuration is discussed in the following section on Distributed I/O.

1.3.2 Distributed I/O

Often sensors must be remotely located from the computer in which the processing and storage of the data takes place. This is especially true in industrial environments where sensors and actuators can be located in hostile environments over a wide area, possibly hundreds of meters away. In noisy environments, it is very difficult for very small signals received from sensors such as thermocouples and strain gauges (in the order of mV) to survive transmission over such long distances, especially in their raw form, without the quality of the sensor data being compromised.

An alternative to running long and possibly expensive sensor wires, is the use of distributed I/O, which is available in the form of signal conditioning modules remotely located near the sensors to which they are interfaced. One module is required for each sensor used, allowing for high levels of modularity (single point to hundreds of points per location). While this can add reasonable expense to systems with large point counts, the benefits in terms of signal quality and accuracy may be worth it.

One of the most commonly implemented forms of distributed I/O is the digital transmitter. These intelligent devices perform all required signal conditioning functions (amplification, filtering, isolation etc), contain a micro-controller and A/D converter, to perform the digital conversion of the signal within the module itself. Converted data is transmitted to the computer via an RS-232 or RS-485 communications interface. The use of RS-485 multi-drop networks, as shown in Figure 1.3, reduces the amount of cabling required, since each signal-conditioning module shares the same cable pair. Linking up to 32 modules, communicating over distances up to 10 km, is possible when using the RS-485 multi-drop network. However, since very few computers have built in support for the RS-485 standard, an RS-232 to RS-485 converter is required to allow communications between the computer and the remote modules.

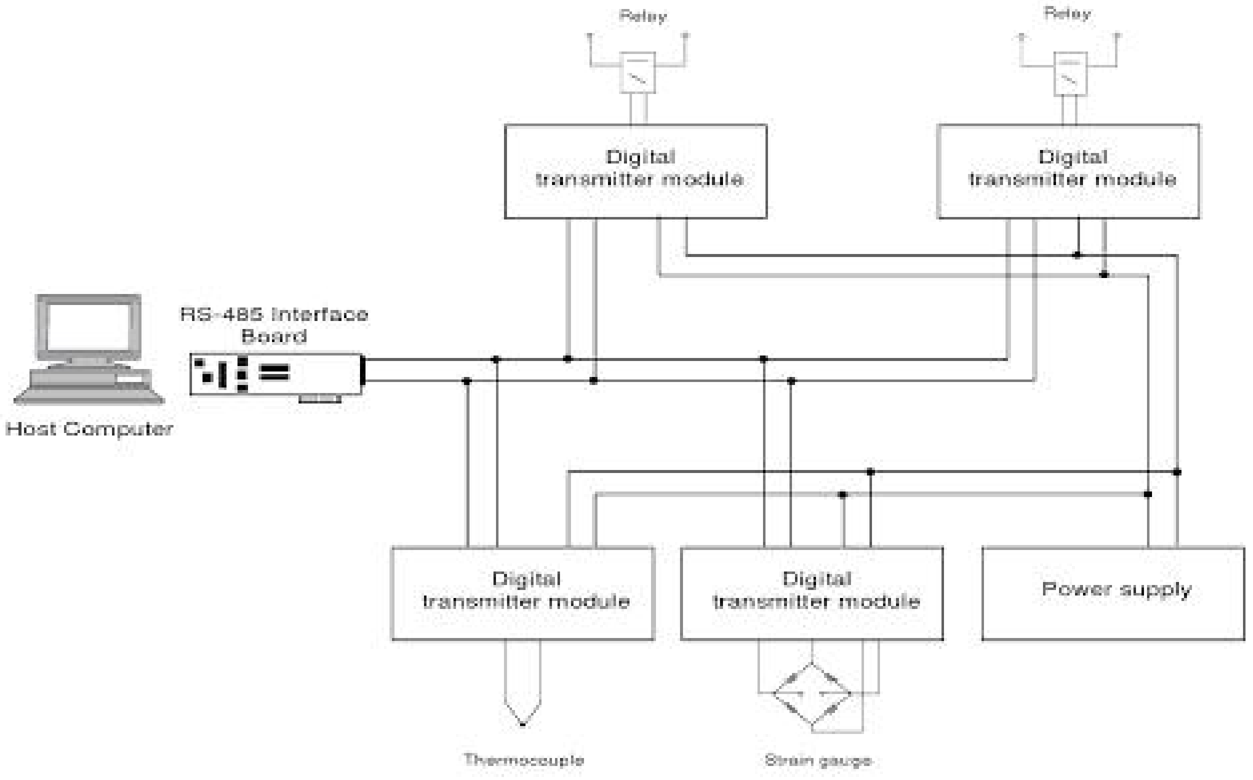


Figure 1.3
Distributed I/O – digital transmitter modules

1.3.3 Stand-alone or distributed loggers/controllers

As well as providing the benefits of intelligent signal conditioning modules, and the ability to make decisions remotely, the use of stand-alone loggers/controllers increases system reliability. This is because once programmed, the stand-alone logger can continue to operate, even when the host computer is not functional or connected. In fact, stand-alone loggers/controllers are specifically designed to operate independently of the host computer. This makes them especially useful for applications where the unit must be located in a remote or particularly hostile environment, (e.g. a remotely located weather station), or where the application does not allow continuous connection to a computer (e.g. controlling temperatures in a refrigerated truck).

Stand-alone loggers/controllers are intelligent powerful and flexible devices, easily interfaced to a wide range of transducers, as well as providing digital inputs and digital control outputs for process control.

The stand-alone logger/controller and logging data are programmed either by a serial communications interface or by using portable and reusable PCMCIA cards. The credit card size PCMCIA card is especially useful when the stand-alone logger/controller is remotely located, but requires an interface connected to the computer. This is shown in Figure 1.4.

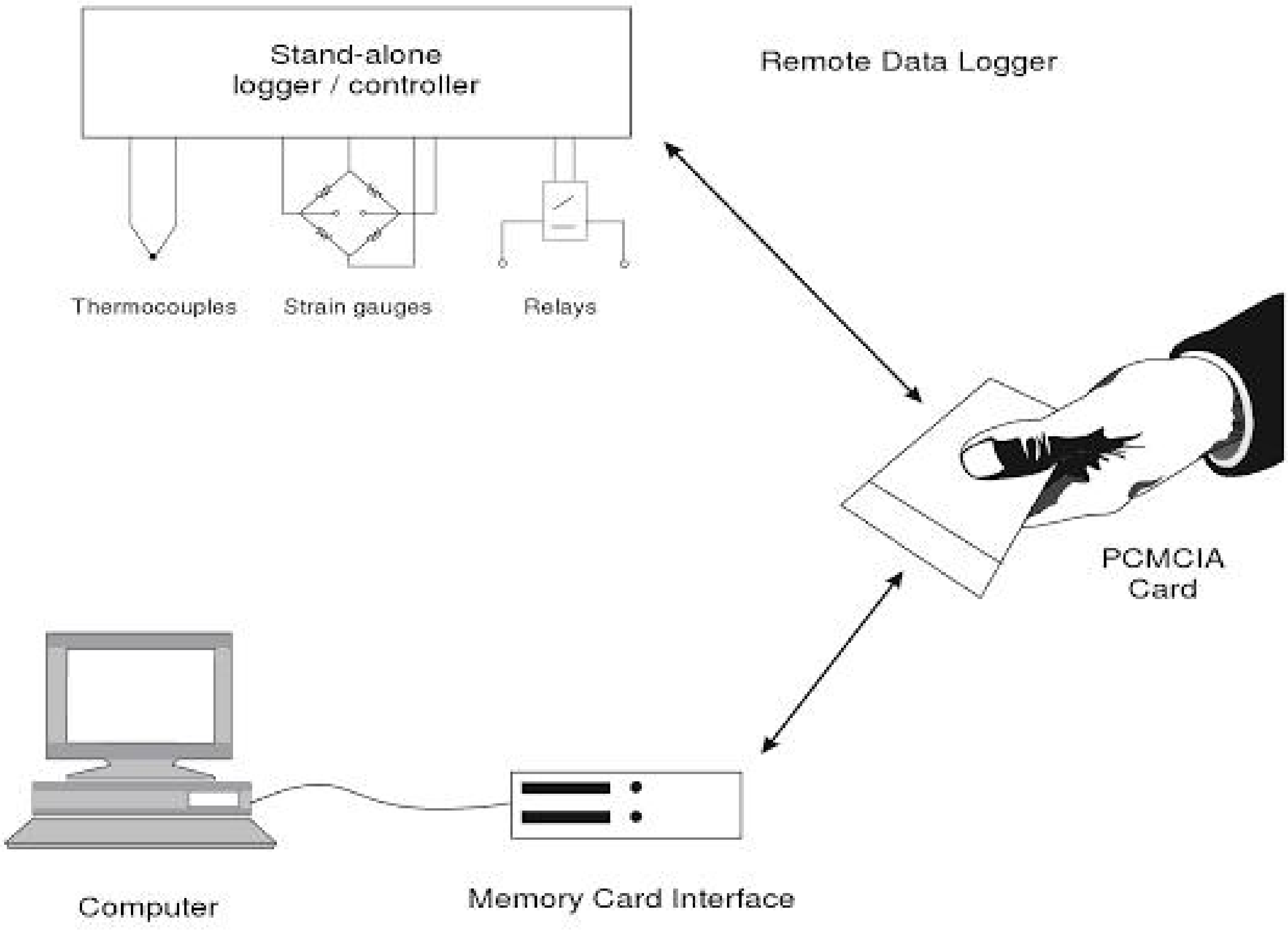


Figure 1.4
Using PCMCIA cards to program and log data from a stand-alone logger/controller

The most commonly used serial communications link for direct connection between the computer and the stand-alone logger/controller is the RS-232 serial interface. This allows programming and data logging up to distances of 50 meters, as shown in Figure 1.5. Where the stand-alone unit must be located remotely, a portable PC can be taken to the remote location or communications performed via a telephone or radio communications network using modems, as shown in Figure 1.6.

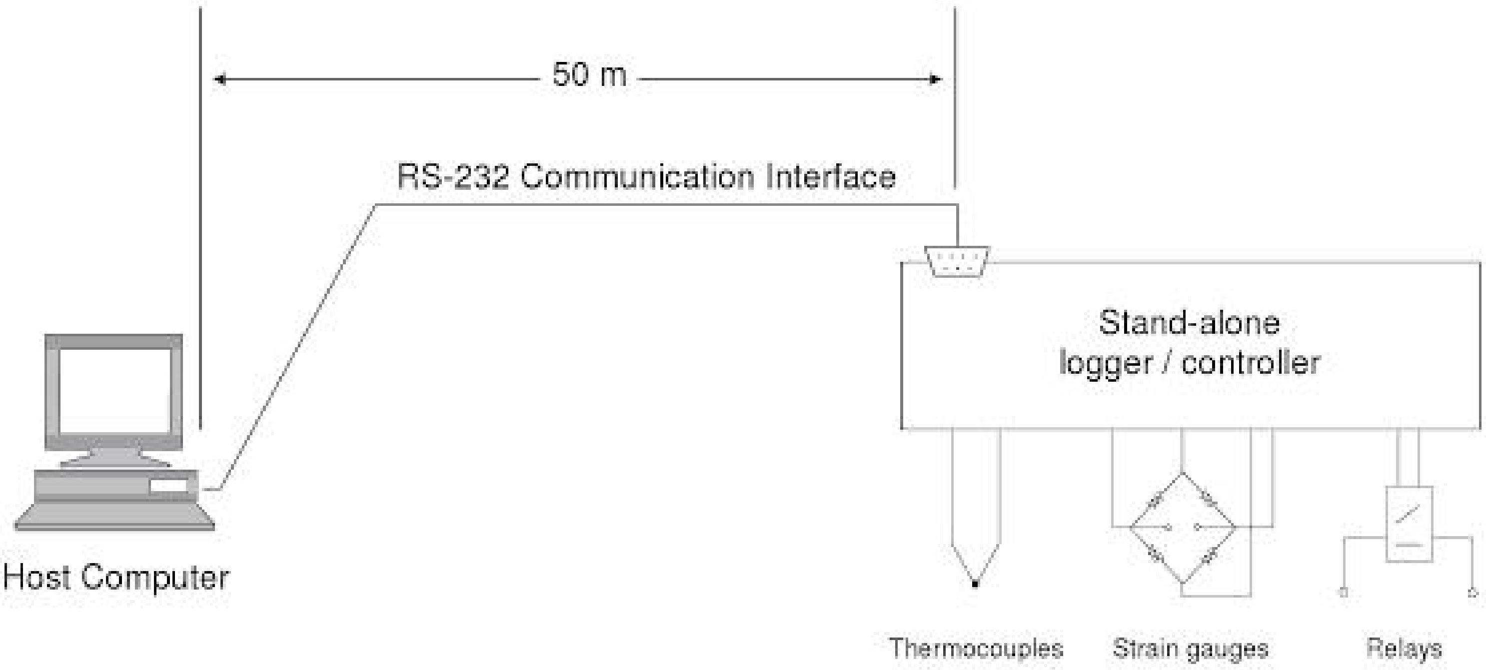


Figure 1.5
Direct connections to a stand-alone logger/controller via an RS-232 serial interface

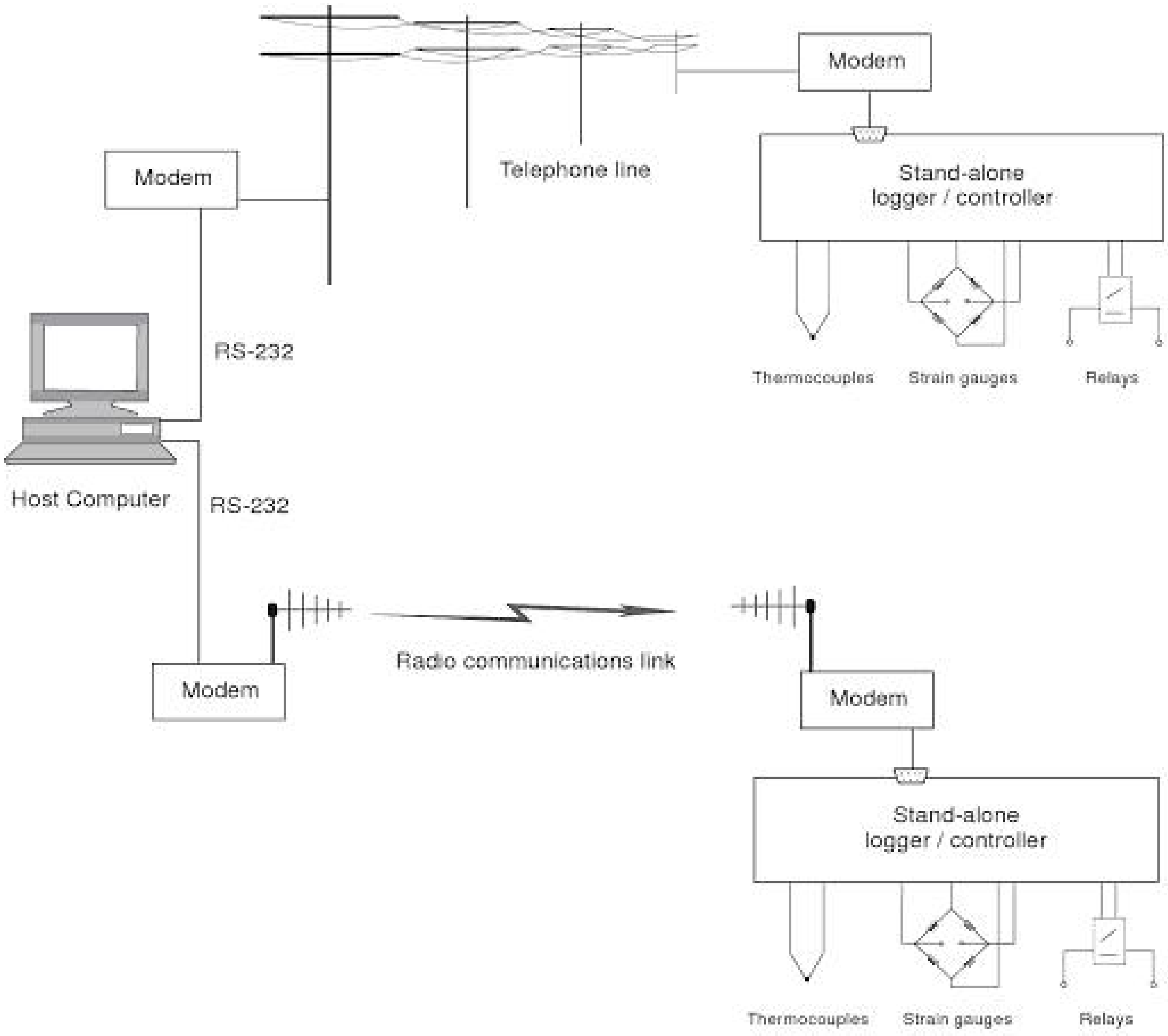


Figure 1.6
Remote connection to a stand-alone logger/controller via a telephone or radio communications network

Where an application requires more than one logger/controller, each unit is connected within an RS-485 multi-drop network. A signal unit, deemed to be the host unit, can be connected directly to the host computer via the RS-232 serial interface, as shown in Figure 1.7, thus avoiding any requirement for an RS-232 to RS-485 serial interface card.

The same methods of programming or logging data from each logger/controller are available either via the serial communications network or via using portable and reusable memory cards.

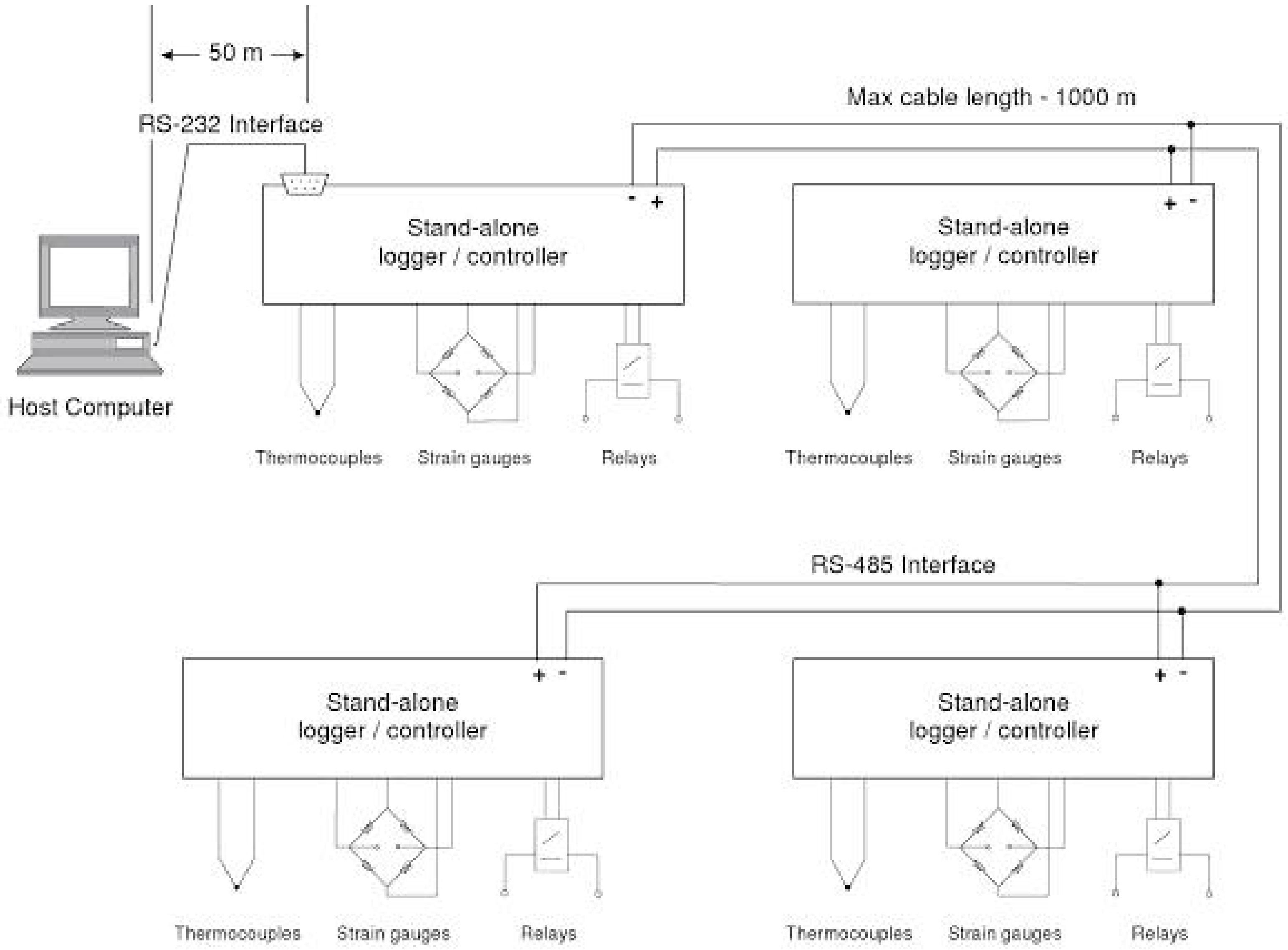


Figure 1.7
Distributed logger/controller network

1.3.4 IEEE-488 (GPIB) remote programmable instruments

The communications standard now known as GPIB (General Purpose Interface Bus), was originally developed by Hewlett-Packard in 1965 as a digital interface for interconnecting and controlling their programmable test instruments. Originally referred to as the Hewlett Packard Interface Bus (HPIB), its speed, flexibility and usefulness in connecting instruments in a laboratory environment led to its widespread acceptance, and finally to its adoption as a world standard (IEEE-488). Since then, it has undergone improvements (IEEE-488.2) and SCPI (Standard Commands for Programmable Instruments), to standardize how instruments and their controllers communicate and operate.

Evolving from the need to collect data from a number of different stand-alone instruments in a laboratory environment, the GPIB is a high-speed parallel communications interface that allows the simultaneous connection of up to 15 devices or instruments on a short common parallel data communications bus. The most common configuration requires a GPIB controller, usually a plug-in board on the computer, which addresses each device on the bus and initiates the devices that will communicate to each other. The maximum speed of communications, the maximum length of cable, and the maximum cable distance between each device on the GPIB is dependent on the speed and processing power of the GPIB controller and the type of cabling used. Typical transfer speeds are of the order of 1 Mbyte/s, while the maximum cable length at this data transfer rate is 20 m. This makes GPIB remote instruments most suited to the research laboratory or industrial test environment.

Thousands of GPIB-compatible laboratory and industrial instruments, such as data loggers and recorders, digital voltmeters and oscilloscopes are available on the market for a wide range of applications and from a wide range of manufacturers. A typical system configuration is shown in Figure 1.8.

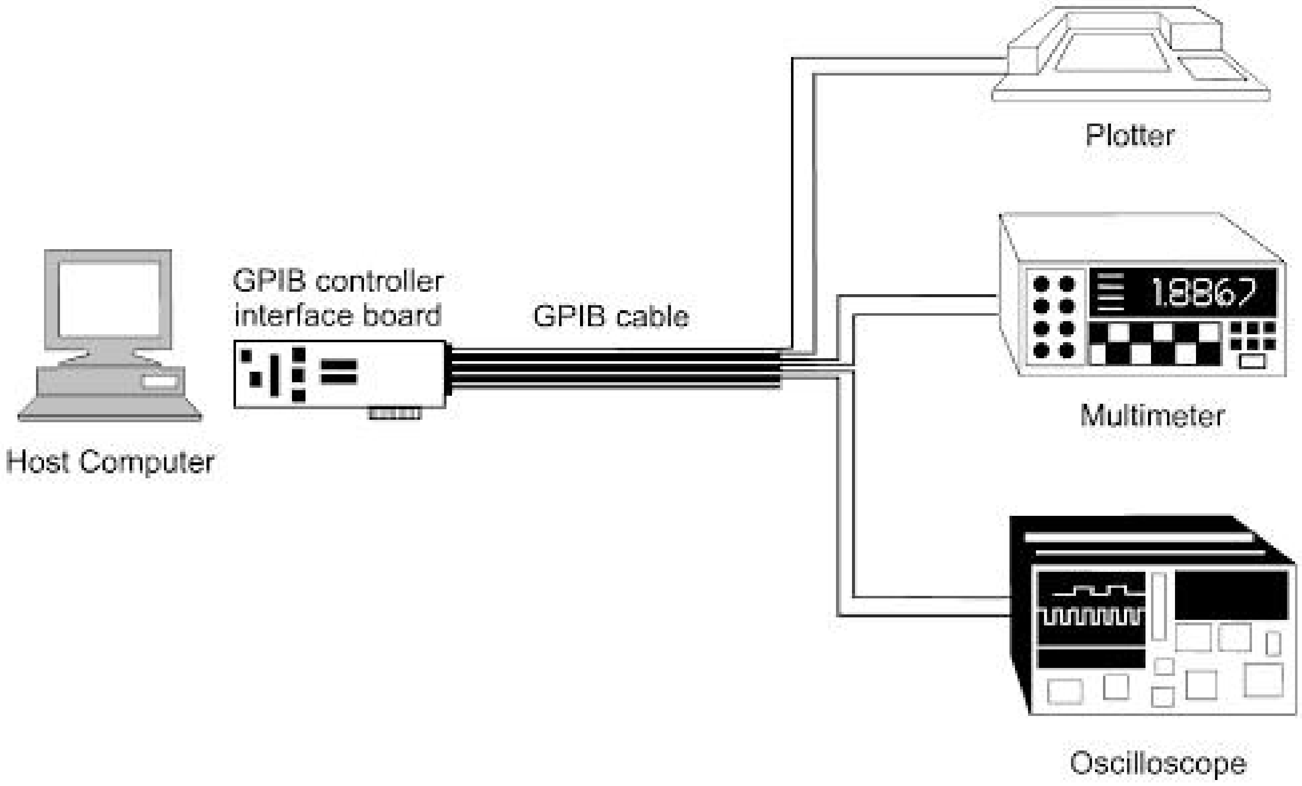


Figure 1.8
A typical GPIB system configuration

2

Analog and digital signals

2.1 Classification of signals

In the real world, physical phenomena, such as temperature and pressure, vary according to the laws of nature and exhibit properties that vary continuously in time; that is they are all analog time-varying signals.

Transducers convert physical phenomena into electrical signals such as voltage and current for signal conditioning and measurement within DAQ systems. While the voltage or current output signal from transducers has some direct relationship with the physical phenomena they are designed to measure, it is not always clear how that information is contained within the output signal. For example, in the case of a flow meter, the output is a digital pulse train whose frequency is directly proportional to the rate of flow. While the change in the flow rate of a fluid may be varying slowly with time, the output signal is a digital pulse train that may vary quickly in time, dependent on the flow rate, and not on the speed of change in the flow rate. This is shown in Figure 2.1.

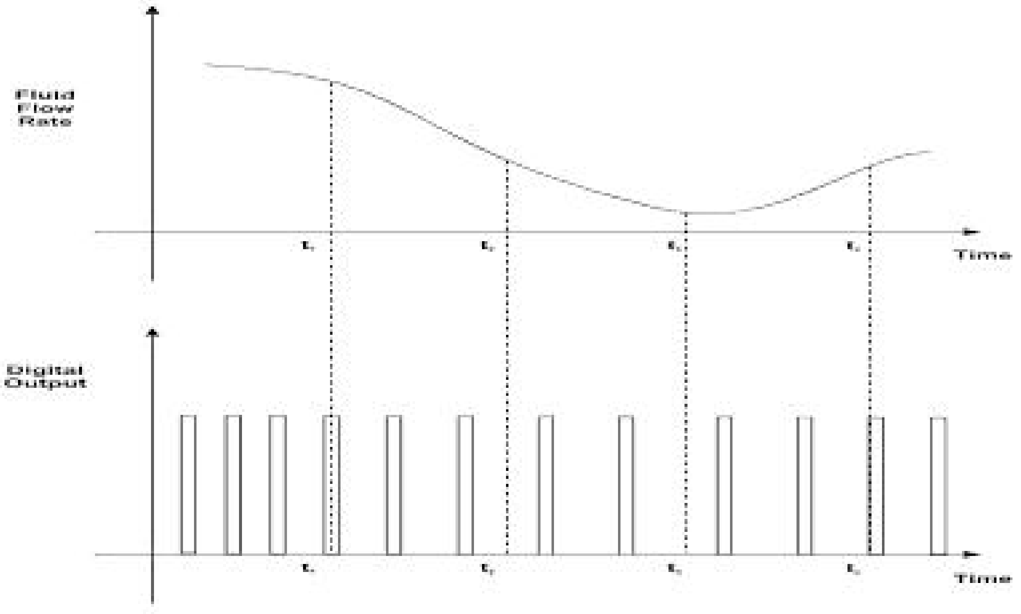


Figure 2.1
The rate of fluid flow and sign at output from a flow meter transducer

This leads us to the need for the classification of signals in DAQ systems, because it is the information contained within a signal that determines its classification, and therefore the method of signal measurement and or the type of hardware required to produce that signal. The classification of signals that may be encountered in data acquisition and control systems are defined in the sections below.

2.1.1 Digital signals binary signals

A digital, or binary, signal can have only two possible specified levels or states; an 'on' state, in which the signal is at its highest level, and an 'off' state, in which the signal is at its lowest level. This is shown in Figure 2.2.

For example, the output voltage signal of a transistor-to-transistor logic (TTL) switch can only have two states – the value in the 'on' state is 5 V, while the value in the 'off' state is 0 V. Control devices, such as relays, and indicators such as LEDs, require digital output signals like those provided on digital I/O boards.

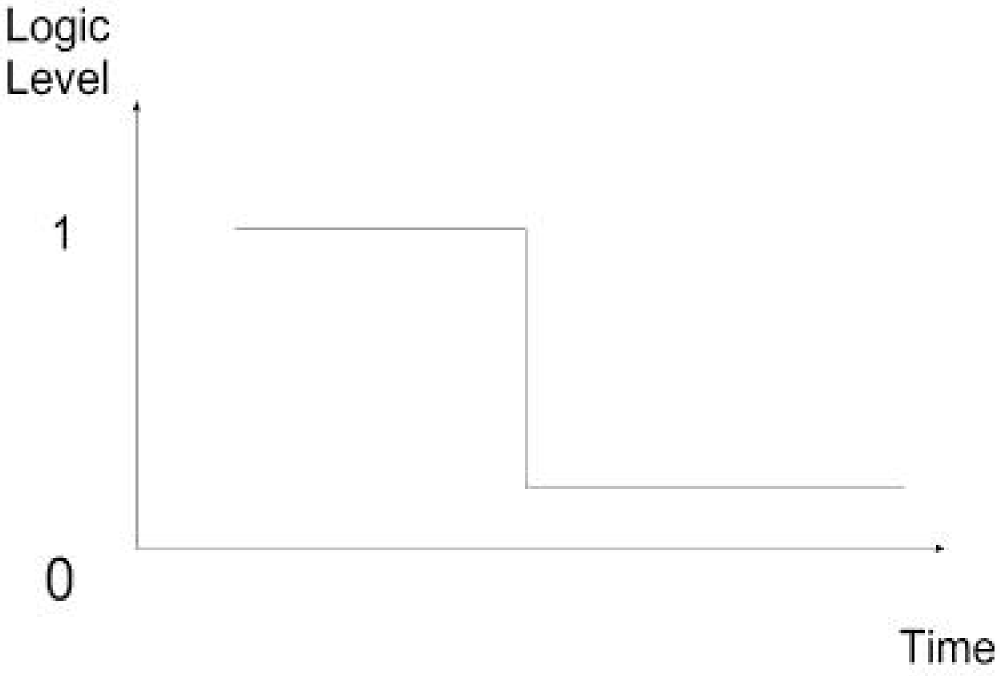


Figure 2.2
A binary digital signal

Digital pulse trains

A digital pulse train is a special type of digital signal, comprising a sequence of digital pulses as shown in Figure 2.3. Like all digital signals, a digital pulse can have only two defined levels or states. It is defined as a pulse because it remains in a non-quiescent state for a short period. A positive going pulse is one that makes a transition from its lowest logic state to its highest logic state, remains at the high logic state for a short duration, and then returns to the low logic state. A negative going pulse makes a transition from its highest logic state to the low logic state, remains there for a short duration, and then returns to the high logic state. The information conveyed in a digital pulse train is conveyed in the number of pulses that occur, the rate at which pulses occur and or the time between pulses.

The output signals from a flow meter or from an optical encoder mounted on a rotating shaft are examples of a digital pulse train. It is also possible for a DAQ system to be required to output a digital pulse train as part of the control process. A stepper motor, for example, requires a series of digital pulses to control its speed and position. While input and output digital pulse trains can be practically measured or produced using digital I/O boards, counter/timer I/O boards are more effective in performing these functions.

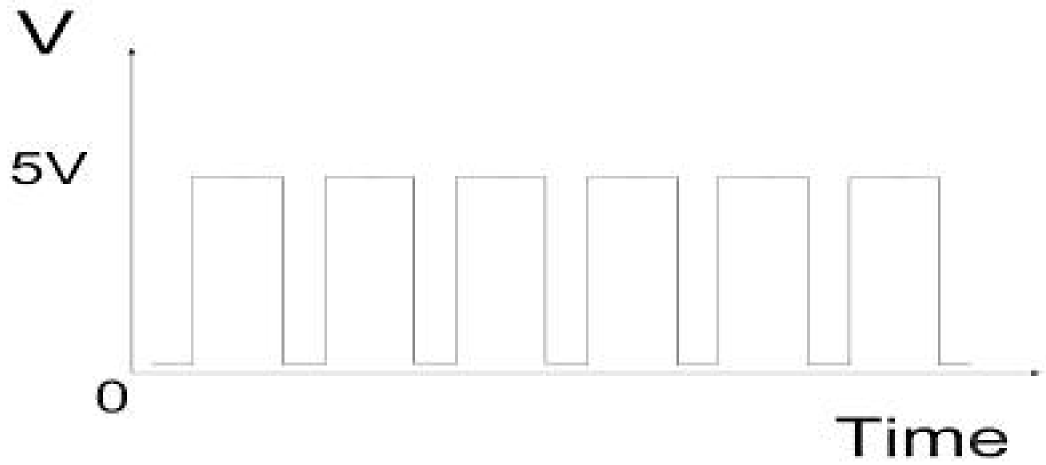


Figure 2.3
Digital pulse train signal

2.1.2 Analog signals

Analog signals contain information within the variation in the magnitude of the signal with respect to time. The relevant information contained in the signal is dependent on whether the magnitude of the analog signal is varying slowly or quickly with respect to time, or if the signal is considered in the time or frequency domains.

Analog DC signals

Analog DC signals are static or slowly varying DC signals. The information conveyed in this type of signal is contained in the level or amplitude of the signal at a given instant in time, not in how this level varies with respect to time. This is shown in Figure 2.4.

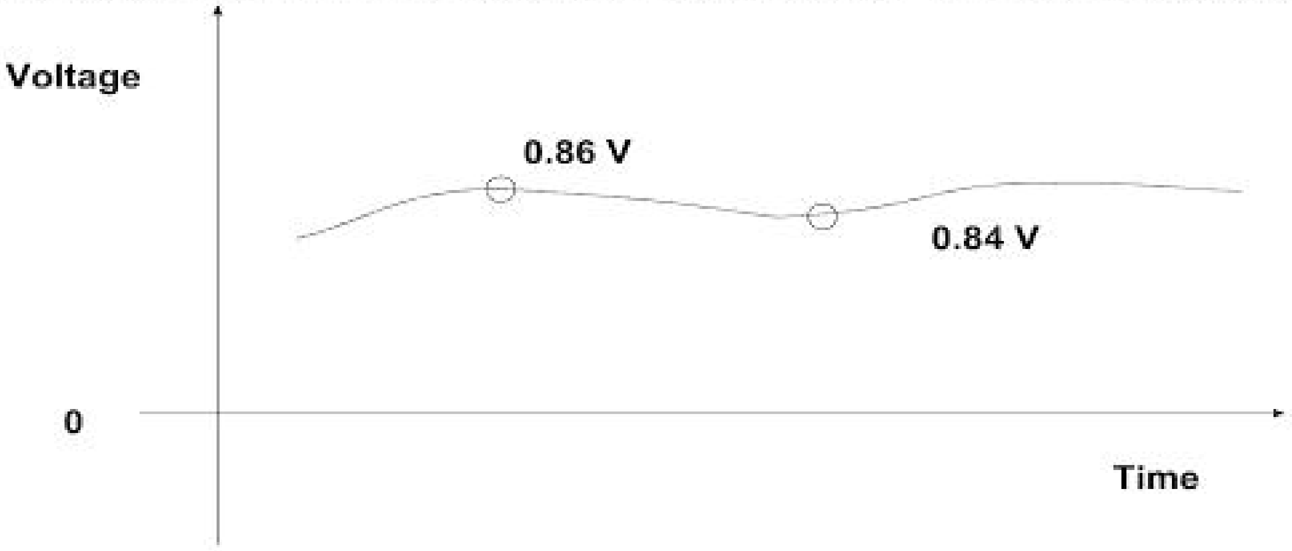


Figure 2.4
An analog DC signal

As the timing of the measurements made of slowly varying signals is not critical, the DAQ hardware would only be required to convert the signal level to a digital form for processing by the computer using an analog-to-digital converter (ADC). Low speed A/D boards would be capable of measuring this class of signal. Temperature and pressure monitoring are just two examples of slowly varying analog signals in which the DAQ system measures and returns a single value indicating the magnitude of the signal at a given instant in time. Such signals can be used as inputs to digital displays and gauges or processed to indicate a control-action (e.g. turn on a heater or open a valve) required for a particular process.

For example, control hardware like a valve actuator, requires only a slowly varying analog signal; the magnitude at a given point in time determining the control setting. DAQ hardware that could perform this task would only be required to convert the digital control setting to an analog form using a digital-to-analog converter (DAC) at the required instant in time. A low-speed general purpose D/A board could perform this function.

The most important parameters to consider for low speed A/D boards and D/A boards are the accuracy and resolution in which the slowly varying signal can be measured or output respectively.

Analog AC signals

The information conveyed in analog AC signals is contained not only in the level or amplitude of the signal at a given instant in time, but also how the amplitude varies with respect to time. The shape of the signal, its slope at a given point in time, the frequency, and location of signal peaks, can all provide information about the signal itself. An analog AC signal is shown in Figure 2.5.

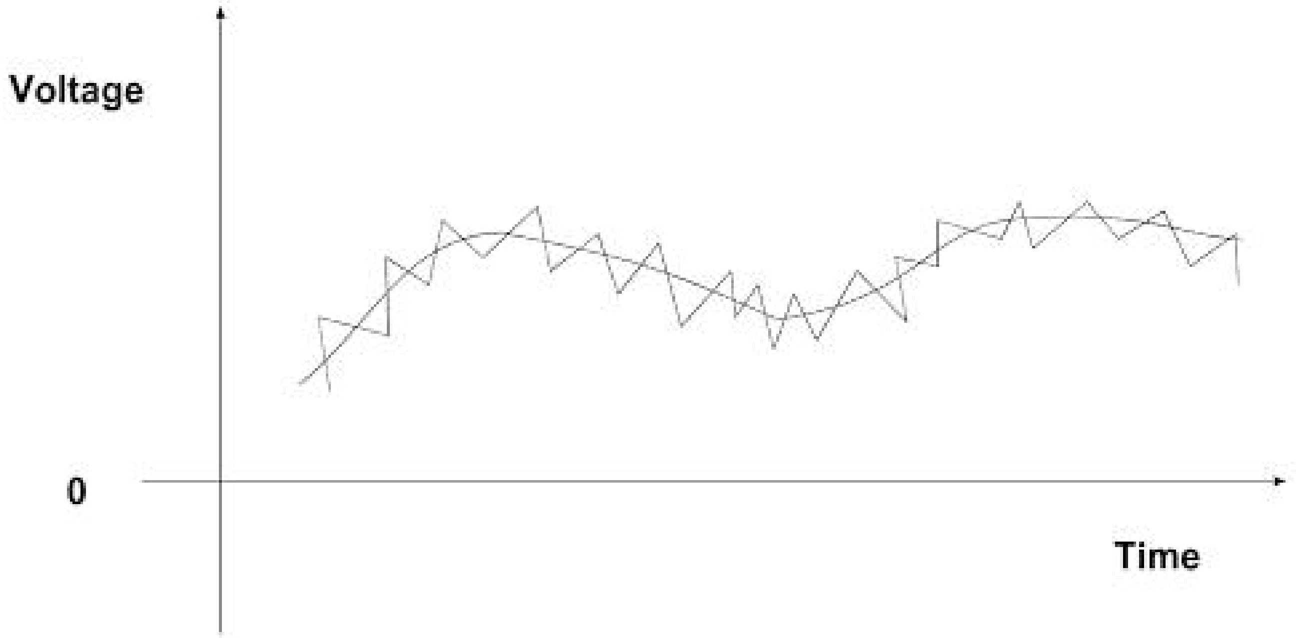


Figure 2.5
An analog AC signal

Since an analog AC signal may vary quite quickly with respect to time, the timing of measurements made of this type of signal may be critical. Hence, as well as converting the signal amplitude to a useful digital form for processing by the computer using an ADC, the DAQ hardware would be required to take the measurements close enough together to reproduce accurately the shape, and therefore the information, contained in the signal. Further to this, the information extracted from the signal may vary depending on when the measurement of the signal started and ended. DAQ hardware used to measure these signals would require an ADC, a sample clock, to time the occurrence of each A/D conversion, and a trigger to start and/or stop the measurements at the proper time, according to some external event or condition, so that the relevant portion of the signal can be obtained. A high-speed A/D board would be capable of performing these functions.

As all time varying signals can be represented by the summation of a series of sinusoidal waveforms of different magnitudes and frequencies, another useful way of extracting information is through the frequency spectrum of a signal. This indicates the magnitudes and frequencies of each of the sinusoidal components that comprise the signal rather than the time-based characteristics of the signal (i.e. shape, slope at a given point etc). This is shown in Figure 2.6.

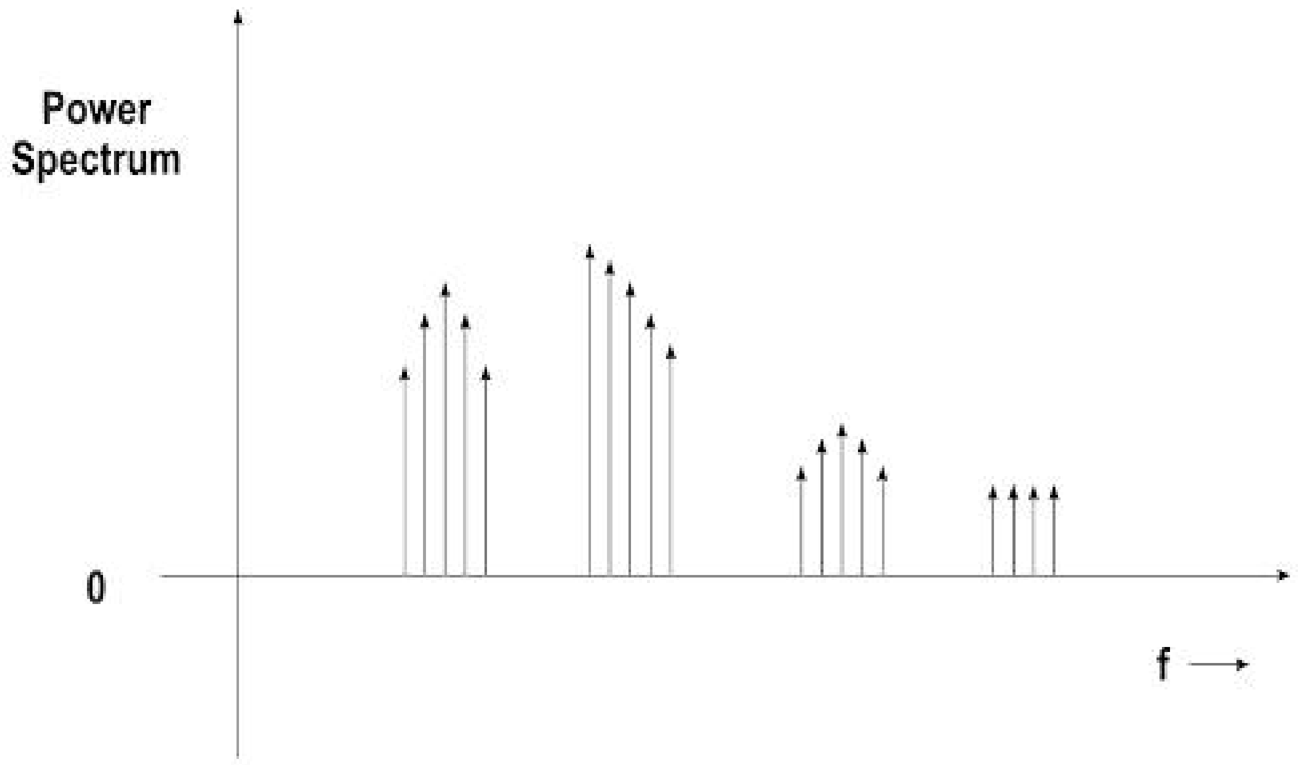


Figure 2.6
An analog AC signal in the frequency domain

Analysis in the frequency domain allows for easier detection and extraction of the wanted signal by filtering out unwanted noise components having frequencies much higher than the desired signal. The digital signal processing (DSP) required to convert the time-measured signal into frequency information and possibly perform analysis on the frequency spectrum, can be achieved with software or with special DSP hardware.

2.2 Sensors and transducers

A transducer is a device that converts one form of energy or physical quantity into another, in accordance with some defined relationship. Where a transducer is the sensing element that responds directly to the physical quantity to be measured and forms part of an instrumentation or control system, then the transducer is often referred to as a sensor.

In data acquisition systems, transducers sense physical phenomena and provide electrical signals that the system can accept. For example, thermocouples, resistive temperature detectors (RTDs), thermistors, and IC sensors convert temperature into an analog voltage signal, while flow transducers produce digital pulse trains whose frequency depends on the speed of flow.

Two defined categories of transducer exist:

- Active transducers convert non-electrical energy into an electrical output signal. They do not require external excitation to operate. Thermocouples are an example of an active transducer.
- Passive transducers change an electrical network value, such as resistance, inductance or capacitance, according to changes in the physical quantity being measured. Strain gauges (resistive change to stress) and LVDTs (inductance change to displacement) are two examples of this. To be able to detect such changes, passive devices require external excitation.

2.3 Transducer characteristics

Transducers are classified according to the physical quantity they measure (e.g. temperature, force etc).

Beyond the obvious selection of the type of transducer required to measure a particular physical quantity and any cost considerations, the characteristics that are most important in determining a transducer's applicability for a given application are as follows:

- Accuracy
- Sensitivity
- Repeatability
- Range

Accuracy

When a range of measurements is made of any process it is essential to know the accuracy of the readings and whether the same is maintained over the entire range or not. The accuracy of a transducer describes how close a measurement is to the actual value of the process variable being measured. It describes the maximum error that can be expected from a measurement taken at any point within the operating range of the transducer. Manufacturers usually provide the accuracy of a transducer as a percentage error over the operating range of the transducer, such as ± 1% between 20ºC and 120ºC, or as a rating (i.e. ± 1ºC) over the operating range of the transducer.

Sensitivity

Sensitivity is defined as the amount of change in the output signal from a transducer to a specified change in the input variable being measured. Highly sensitive devices, such as thermistors, may change resistance by as much as 5% per ºC, while devices with low sensitivity, such as thermocouples, may produce an output voltage that changes by only 5μV per ºC.

Repeatability

If two or more measurements are made of a process variable at the identical state, a transducer's repeatability indicates how close the repeated measurements will be. The ability to generate almost identical output responses to the same physical input throughout its working life is an indication of the transducer's reliability and is usually related to the cost of the transducer.

Range

A transducer is usually constructed to operate within a specified range. The range is defined as the minimum and maximum measurable values of a process variable between which the defined limits of all other specified transducer characteristics (i.e. sensitivity, accuracy etc) are met. A thermocouple, for example, could well work outside its specified operating range of 0ºC to 500ºC, however its sensitivity outside this range may be too small to produce accurate or repeatable measurements.

Several variables affect the accuracy, sensitivity, and repeatability of the measurements being made.

In the process of measuring a physical quantity, the transducer disturbs the system being monitored. As an example, a temperature measuring transducer lowers the temperature of the system being monitored, while energy is used to heat its own mass.

Transducers are responsive to unwanted noise in the same way that a record player's magnetic cartridge is sensitive to the alternating magnetic field of the mains transformer (giving rise to 'mains hum').

Some transducers are subject to excitation signals that alter their response to the input physical quantity being measured. As an example, an RTD's excitation current can result in self-heating of the device, thereby changing its resistance.

2.4 Resistance temperature detectors (RTDs)

2.4.1 Characteristics of RTDs

Resistance temperature detectors (RTDs) are temperature sensors generally made from a pure (or lightly doped) metal whose resistance increases with increasing temperature (positive resistance temperature coefficient).

Most RTD devices are either wire wound or metal film. Wire wound devices are essentially a length of wire wound on a neutral core and housed in a protective sleeve. Metal film RTDs are devices in which the resistive element is laid down on a ceramic substrate as a zig-zag metallic track a few micrometers thick. Laser trimming of the metal track precisely controls the resistance. The large reduction in size with increased resistance that this construction allows, gives a much lower thermal inertia, resulting in faster response and good sensitivity. These devices generally cost less than wire wound RTDs.

The most popular RTD is the platinum film PT100 (DIN 43760 Standard), with a nominal resistance of 100 Ω ± 0.1 Ω at 0ºC. Platinum is usually used for RTDs because of its stability over a wide temperature range (–270ºC to 650ºC) and its fairly linear resistance characteristics. Tungsten is sometimes used in very high temperature applications. High resistance (1000 Ω) nickel RTDs are also available. If the RTD element is not mechanically stressed (this also changes the resistance of a conductor), and is not contaminated by impurities, the devices are stable over a long period, reliable and accurate.

2.4.2 Linearity of RTDs

In comparison to other temperature measuring devices such as thermocouples and thermistors, the change in resistance of an RTD with respect to temperature is relatively linear over a wide temperature range, exhibiting only a very slight curve over the working temperature range. Although a more accurate relationship can be calculated using curve fitting – the Callendar-Van Dusen polynomial equations are often used – it is not usually required. Since the error introduced by approximating the relationship between resistance and temperature as linear is not significant, manufacturers commonly define the temperature coefficient of RTDs, known as alpha (α), by the expression:

$$Alpha\ (\alpha) = \frac{R_{100} - R_0}{100 \times R_0} \quad \Omega / \Omega / {}^\circ\mathrm{C}$$

Where:

R_0 = Resistance at 0ºC
R_{100} = Resistance at 100ºC

This represents the change in the resistance of the RTD from 0ºC to 100ºC, divided by the resistance at 0ºC, divided by 100ºC.

From the expression of alpha (α) it is easily derived that the resistance R_T of an RTD, at temperature T can be found from the expression:

$R_T = R_0(1 + \alpha T)$

Where:
R_0 = Resistance at 0°C

For example, a PT100 (DIN 43760 Standard), with nominal resistance of 100 Ω ± 0.1 Ω at 0°C has an alpha (α) of 0.00385 Ω / Ω / °C. Its resistance at 100°C will therefore be 138.5 Ω.

2.4.3 Measurement circuits and considerations for RTDs

Two-wire RTD measurement

Since the RTD is a passive resistive device, it requires an excitation current to produce a measurable voltage across it. Figure 2.7 shows a two-wire RTD excited by a constant current source, I_{EX} and connected to a measuring device.

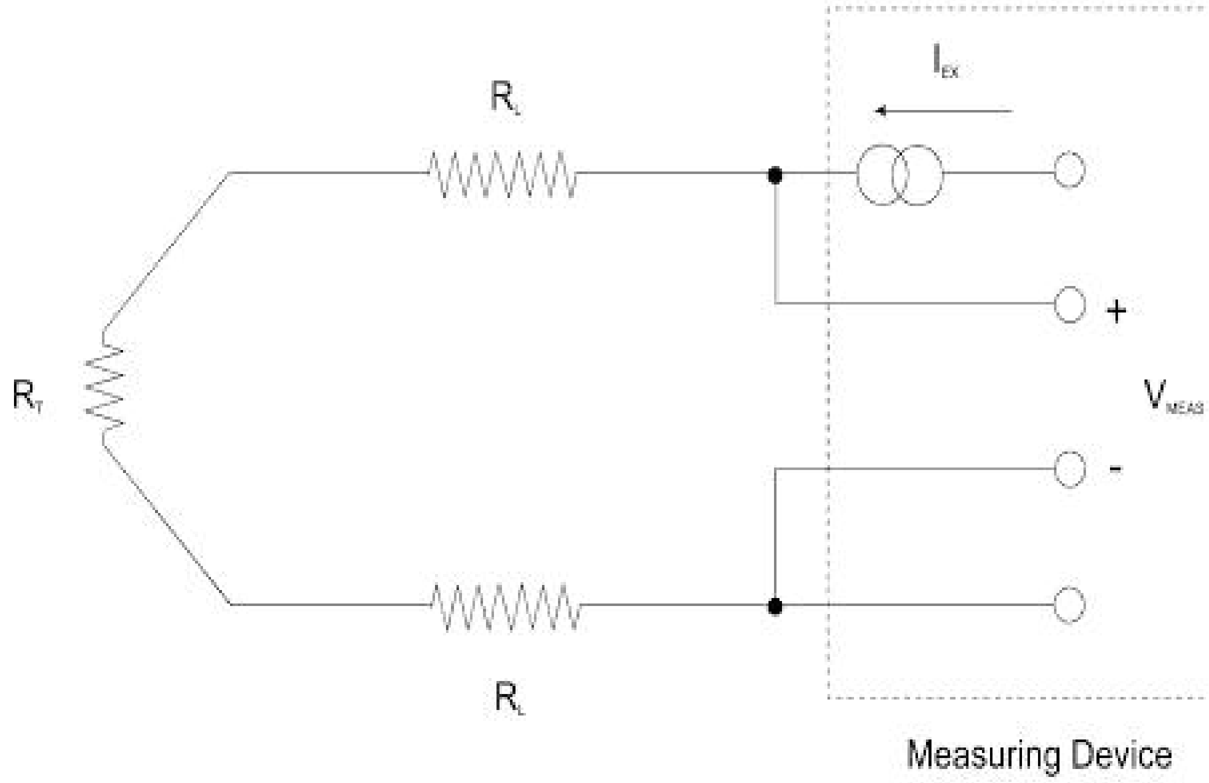


Figure 2.7
Two-wire RTD measurement

Any resistance, R_L, in the lead wires between the measuring device and the RTD will cause a voltage drop on the leads equal to ($R_L \times I_{EX}$) volts. The voltage drop on the wire leads will add to the voltage drop across the RTD, and depending on the value of the lead wire resistance compared to the resistance of the RTD, may result in a significant error in the calculated temperature.

Consider an example where the lead resistance of each wire is 0.5 Ω. For a 100 ω RTD with an alpha (α) of 0.385 Ω / °C, the lead resistance corresponds to a temperature error of 2.6°C (1 .0 Ω / 0.385 Ω / °C).

This indicates that if voltage measurements are made using the same two wires which carry the excitation current, the resistance of the RTD must be large enough, or the lead wire resistances small enough, that voltage drops due to the lead wire resistances are negligible. This is usually true where the leads are no longer than a few (<3) meters for a 100 Ω RTD.

Four-wire RTD measurement

A better method of excitation and measurement, especially when the wire lead lengths are greater than a few meters in length, is the four-wire RTD configuration shown in Figure 2.8.

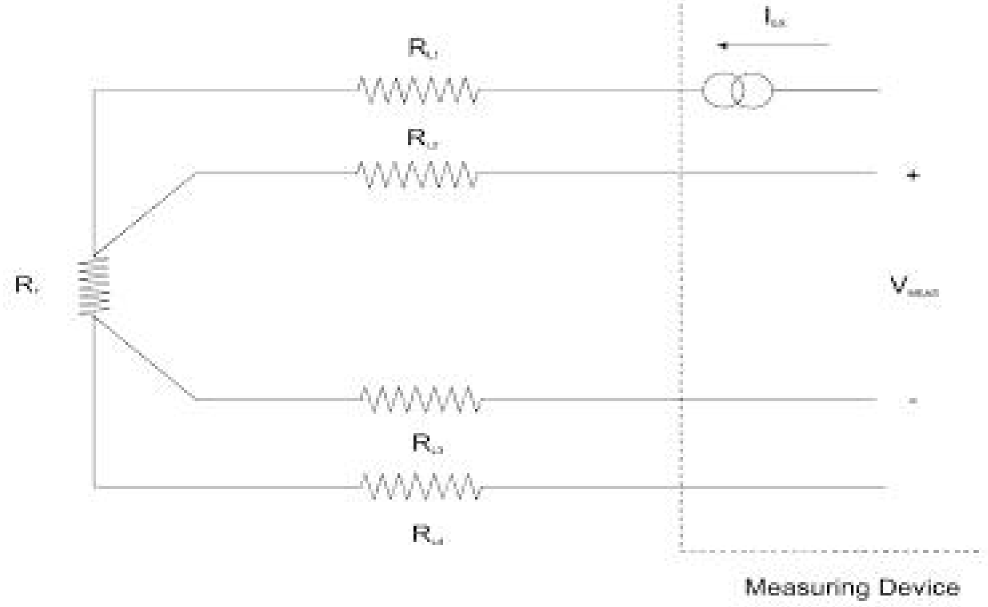


Figure 2.8
Four-wire RTD measurements

RTDs are commonly packaged with four (4) leads, two current leads to provide the excitation current for the device, and two voltage leads for measurement of the voltage developed. This configuration eliminates the voltage drops caused by excitation current through the lead resistances (R_{L1} and R_{L4}). Since negligible current flows in the voltage lead resistances, (R_{L2} and R_{L3}) only the voltage drop across the resistance R_T of the RTD is measured.

Three-wire RTD measurement

A reduction in cost is possible with the elimination of one of the wire leads. In the three-wire configuration shown in Figure 2.9, only one lead R_{L1} adds an error to the RTD voltage measured.

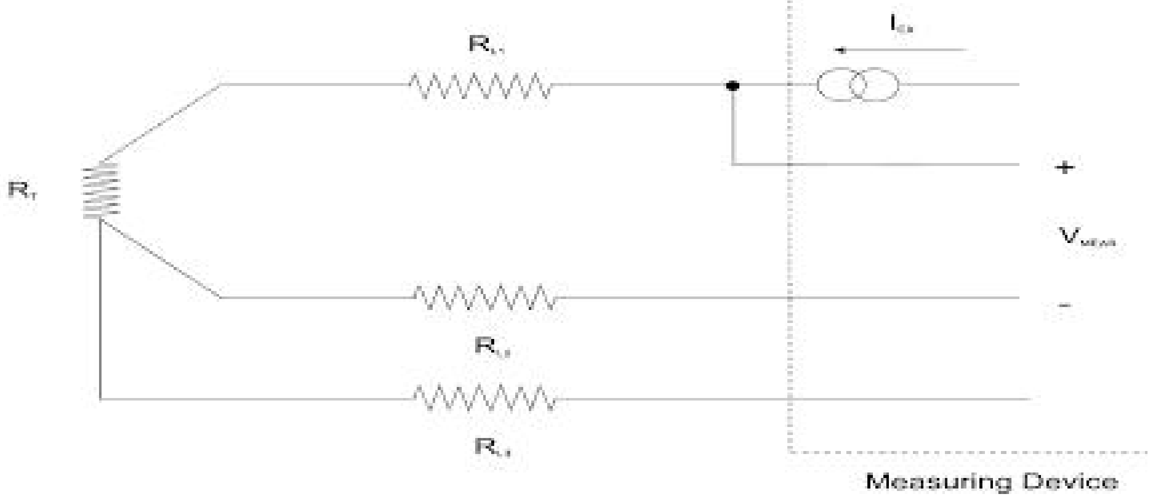


Figure 2.9
Three-wire RTD measurements

Self-heating

Another consequence of current excitation of the RTD is the possible effect that internal heating of the device may have on the accuracy of the actual temperature measurements being made. The degree of self-heating depends on the medium in which the RTD is being used, and is typically specified as the rise in temperature for each mW of power dissipated for a given medium (i.e. still air).

For a PT100 RTD device, the self-heating coefficient is 0.2ºC/mW in still air, although this will vary depending on the construction of the RTD housing and its thermal properties. With an excitation current of 0.75 mA the power to be dissipated by the device is 56 μW $[(0.75 \times 10^{-3})^2 \times 100]$ corresponding to a rise in the temperature of the device due to self-heating of 0.011ºC (56 μW × 0.2).

Inaccuracies in the temperature measurement due to self-heating problems, can be greatly reduced by:

- Minimizing the excitation power
- Exciting the RTDs only when a measurement is taken
- Calibrating out steady state errors

2.5 Thermistors

A cheap form of temperature sensing is provided by the thermistor, which is a thermally sensitive semiconductor resistor formed from the oxides of various metals. The type and composition of the semiconductor oxides used (i.e. manganese, nickel, cobalt etc) depend on the resistance value and temperature coefficient required.

More commonly used thermistor devices exhibit a negative temperature coefficient and have a high degree of sensitivity to small changes in temperature, typically 4% / ºC.

Their accuracy is typically ten times better than thermocouples but not as accurate as RTDs. Thermistors are non-linear devices and directly useful over typical temperature ranges of –80ºC up to 250ºC. With regard to this, modern microprocessor based systems (either PCs or stand-alone data loggers) can be used to relieve some of the limitations caused by non-linearities, by modeling the non-linearities with quadratic equations.

Thermistors exhibit a high resistance, typically 3 kΩ, 5 kΩ, 6 kΩ and 10 kΩ at 25ºC, although values as low as 100 Ω are available. High resistance means that the lead resistances of wires used to excite thermistors are usually negligible, requiring only two wire measurement schemes.

One of the attractions of thermistors is the wide range of shapes in the form of beads, discs, rods and probes that can be easily manufactured. Their small size means they have a fast thermal response, but can be quite fragile compared to RTDs that are more robust.

Just as excitation currents for RTDs can cause self-heating problems, this is even more the case for thermistors due to the higher device resistance values.

Self-heating problems can be greatly reduced by:

- Minimizing the excitation power
- Exciting the RTDs only when a measurement is taken
- Calibrating out steady state errors. Some authorities state that the temperature rise, in ºC, due to self-heating can be calculated by dividing the proposed internal power dissipation by 8 mW.

2.6 Thermocouples

A thermocouple is two wires of dissimilar metals that are electrically connected at one end (measurement junction) and thermally connected at the other end (the reference junction). This is shown in Figure 2.10 below.

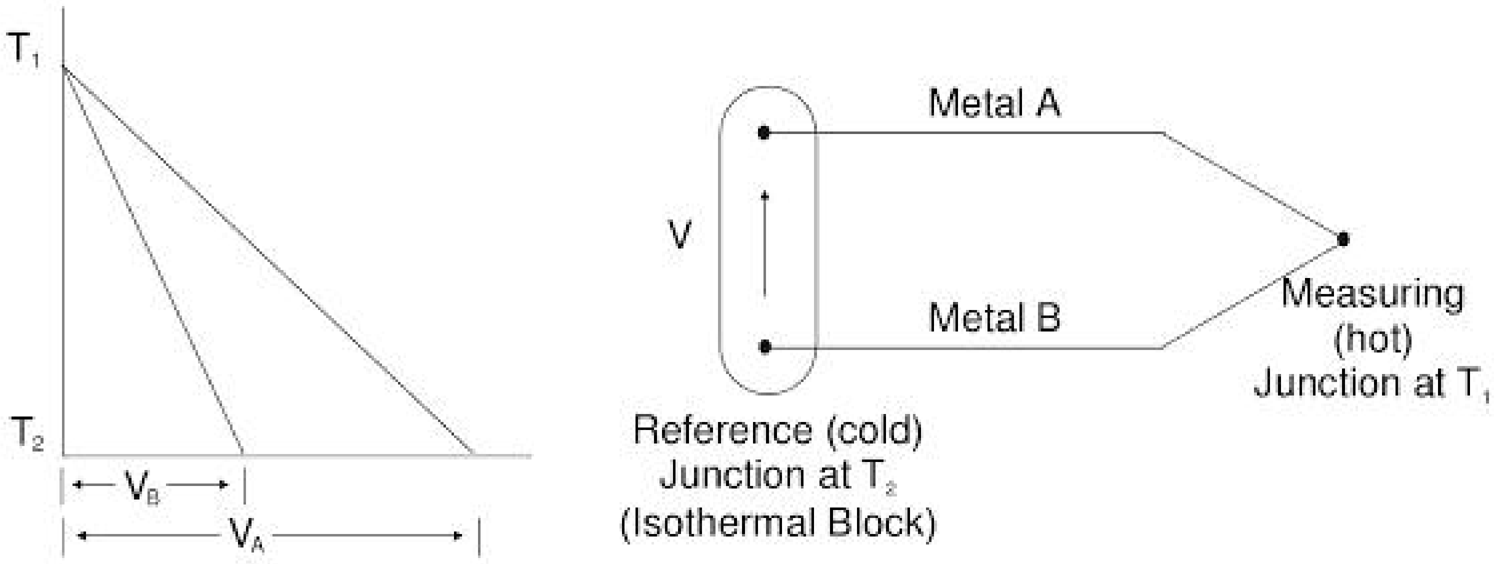


Figure 2.10
Thermocouple measurement

Its operation is based on the principle that temperature gradients in electrical conductors generate voltages in the region of the gradient.

Different conductors will generate different voltages for the same temperature gradient. Therefore, a small voltage, equal to the difference between the voltages generated by the thermal gradient in each of the wires ($V = V_A - V_B$), can then be measured at the reference junction.

Note that this voltage is produced by the temperature gradient along the wires and not by the junction itself. As long as the conductors are uniform along their lengths, then the output voltage is only affected by the temperature difference between the measurement (hot) junction and tile reference (cold) junction, and not the temperature distribution along the conductor between them.

2.6.1 Reference junction compensation

Calculations determining the temperature corresponding to a given measured voltage of a thermocouple assume that this voltage corresponds to a temperature gradient that is referenced to 0ºC. Clearly, where the reference junction is allowed to follow ambient temperature, this is not the case.

Where ambient temperature variations of the reference junction would cause significant errors in the temperature calculation from the voltage output of the thermocouple, two methods of reference junction compensation exist:

Maintain the reference junction at a constant known temperature such as an ice bath (0ºC). This is where the term 'cold junction' was originally derived.

Measure the temperature of the reference junction and add the reference junction voltage. The reference junction voltage is equal to the voltage, which would be generated by the same thermocouple if its measurement junction was at ambient temperature and its reference junction was at 0ºC.

Obviously the second option is far easier to implement and has led to the design of many cold junction compensation circuits. The necessary voltage correction can be carried out with software, hardware, or a combination of both.

Hardware compensation

Hardware compensation requires dedicated circuitry to generate a compensation voltage according to the ambient temperature of the isothermal block, and add this voltage to the voltage measured at the measuring junction. As the voltage vs temperature relationship varies between thermocouples, each thermocouple type must have a separate compensation circuit that operates over the required working range of ambient temperatures. This makes hardware compensation circuitry for thermocouples complex and expensive, and by their nature, prone to inherent errors.

Software compensation

Software compensation requires only that an additional direct reading temperature sensor, such as a thermistor or silicon sensor, be used to measure the isothermal block temperature of the reference junction. Software is then used to calculate the equivalent reference junction voltage, either by polynomial equations, or look-up tables, for the thermocouple type being used. Once calculated, this value is added to the measured output voltage from the thermocouple. The resulting voltage is converted back to a temperature, representing the true thermocouple temperature.

Note: It is not always the case that changes in the ambient temperature lead to significant errors in determining the thermocouple temperature, as shown by the example below.

Example: Consider a type S thermocouple used to measure temperatures of 1500°C within a furnace. The ambient temperature of the reference junction is 25°C ± 15°C.
Since the sensitivity of the thermocouple is 12 µV / °C at 1500°C and a change from 10°C to 40°C at the reference junction produces a change of 180 µV in the net output voltage, the equivalent change in temperature at the measuring junction is 15°C.

This represents at most a 1% error of 1500°C over the operating temperature range of the reference junction. In this case, the error introduced by changes in the reference junction temperature might be ignored.

2.6.2 Isothermal block and compensation cables

Quite often thermocouples, especially those used in industrial applications, are at a considerable distance from the measuring points and require extension leads and connectors. Conventional copper wire and connectors cannot be used for the extensions as unwanted thermocouples are created. Wire and connectors of the same material as the thermocouple must be used. The use of extension cables made of similar but less pure metals than the actual thermocouple, is an economical way of extending the thermocouple circuit.

This wire, though considerably cheaper, has a limited temperature range of typically 0°C to 100°C and must not be used where temperatures exceed this range.

Where inline connectors are used these must also be of the same material as the thermocouples. Color-coded and polarized connectors (to prevent alloy reversal) are available.

References junctions are held at the same temperature by an 'isothermal block', a physical arrangement that ensures good thermal conductivity between the ends of the thermocouple cable. It is advisable to protect the isothermal block from rapid ambient temperature changes.

2.6.3 Thermocouple linearization

In addition to requiring cold-junction compensation, thermocouples are also highly nonlinear, and thus require linearization. For example, a J type thermocouple has a thermal coefficient of 22 µV per °C at – 200°C, but 64 µV per °C at 750°C.

For most purposes, some form of software-based linearization is used. Two techniques of linearization are common:

Look-up tables: With this technique, a table of temperatures versus all possible measured voltages is stored, and the appropriate temperature is obtained via an indexing operation. This is very fast, but requires large amounts of memory. Cold-junction compensation is also difficult to handle.

Polynomial compensation: Using this technique, polynomial approximations are used to obtain temperature from voltage. The number of polynomial terms used depends on the temperature range, and the type of thermocouple. For example, type J thermocouples can be approximated to 0.1° over 0 to 760°C with a fifth-order polynomial, but an F-type thermocouple requires a ninth-order equation for only 0.5° accuracy.

For wide temperature ranges, several lower-order polynomials over narrower ranges are often used. For example, there are thermocouple board drivers that use three eighth-order polynomials for voltage-to-temperature conversions. The range of each equation is optimized for each type of thermocouple. In addition, a second-order polynomial is used to convert the cold-junction temperature to a thermocouple voltage for compensation.

The use of a second-order polynomial is only possible because the terminal block temperature varies from 0° to 70°C.

2.6.4 Thermocouple types and standards

Thermocouple standards specify the voltage vs temperature characteristics, color codes, error limits and composition of standard thermocouples. There are five standards for thermocouples in general use, namely NBS/ANSI (American), BS (British), DIN (German), JIS (Japanese), and NF (French).

Eight main types of thermocouples are general used in industry. These are divided into two main groups: base metal thermocouples (types J, K, N, E & T) and noble metal thermocouples (types R, S & B). Their composition and operating temperature range according to the NBS standard is shown in Table 2.1.

In addition, there are several high temperature tungsten-based thermocouples (types G, C & D), which allow temperature measurements between 0°C and 2320°C. As these thermocouples do not follow any official standards, manufacturers' data sheets should be consulted to ensure correct use.

Type	Positive	Negative	Temperature range □C
B	Pt, 30% Rh	Pt, 6% Rh	+300 to 1700
C	W, 5% Re	W, 26% Rh	0 to 2320
D	W, 3% Re	W, 25% Re	0 to 2320
E	Ni, 10% Cr	Cu, 45% Ni	–200 to 900
G	W	W, 26% Re	0 to 2320
J	Fe	Cu, 45% Ni	–200 to 750
K	Ni, 10% Cr	Ni, 2% Mn, 2% Al	–200 to 1250
N	Ni, 14% Cr, 1% Si	Ni, 4% Si, 0.1% Mg	–200 to 1350
R	Pt, 13% Rh	Pt	0 to 1450
S	Pt, 10% Rh	Pt	0 to 1450
T	Cu	Cu, 45% Ni	–200 to 350

Table 2.1
Thermocouple specifications (NBS Standard)

2.6.5 Thermocouple construction

In addition to thermocouple type, thermocouple style is another important factor in performance. Three basic styles are available, as illustrated in Figure 2.11(a).

The exposed, or bead, junction thermocouple has its junction exposed to air. Thermocouples with exposed junctions (Figure 2.11(b)) are generally used to measure gas temperature, and they have an extremely fast response time.

In ungrounded-junction thermocouples (Figure 2.11(c)), a conductive sheath protects the thermocouple junction. This sheath is electrically isolated from the thermocouple itself. This con-struction is particularly useful where high levels of electrical noise are present. The ungrounded junction thermocouple has the disadvantage that response time is long, typi-cally of the order of several seconds. Problems can also arise from thermal shunting, re-sulting in the junction being at a different temperature to the sheath.

In grounded-junction thermocouples, a conductive sheath also protects the thermocouple junction, and the sheath is electrically connected to the thermocouple junction. This has the advantage that response time is faster than for the ungrounded-junction type, and thermal shunting effects are minimized, while still maintaining good noise immunity. A disadvantage is the susceptibility to ground loop problems, which are particularly difficult to solve in thermocouples, due to low voltages.

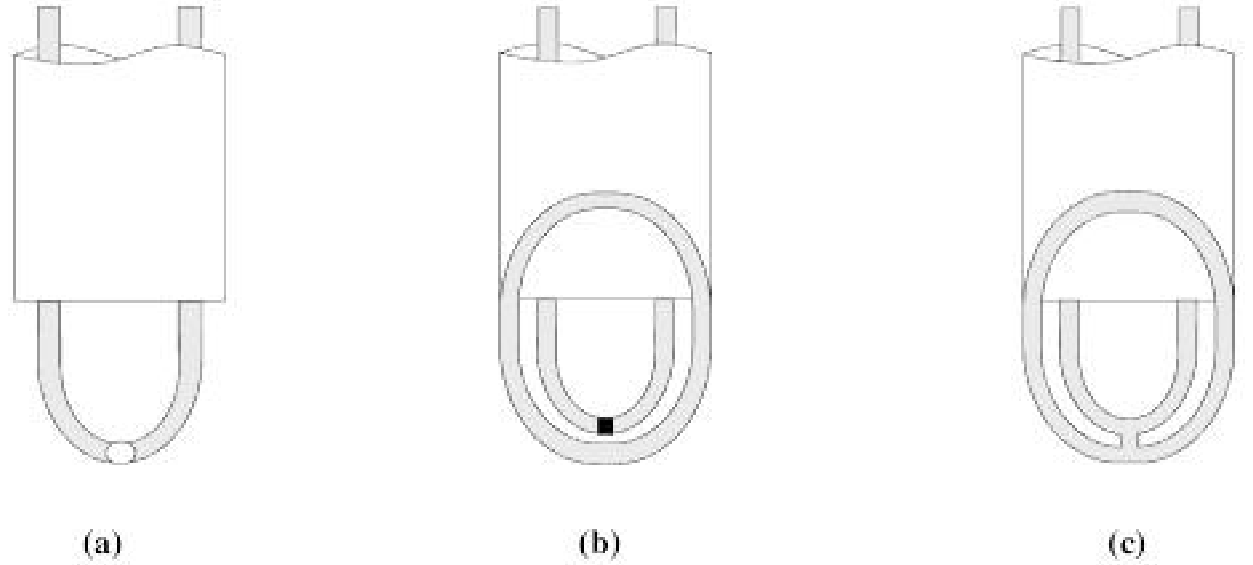

(a) (b) (c)

Figure 2.11
Thermocouple styles

2.6.6 Measurement errors

When making temperature measurements using thermocouples there are several possible sources of error, in addition to any errors that occur due to the accuracy of the measuring equipment.

These are:

- Reference junction isothermal characteristics and reference junction temperature sensor accuracy – the most significant sources of error. Temperature gradients between the temperature sensor and the terminals to which the thermocouples are connected result in errors of the magnitude of the temperature difference. Added to this is the magnitude of any inherent inaccuracies in the temperature sensor used to measure the ambient temperature.
- Induced electrical noise. Due to the low signal voltage levels from thermocouples, typically in the order of µV/°C, temperature measurements

using thermocouples are susceptible to the effects of noise. This is especially true where long thermocouple cables are used in the measurement process. The effects of noise can be reduced by amplifying the low-level thermocouple voltages as close to the source as possible, and where this is not possible, by using twisted, shielded cables.

- Quality of the thermocouple wire. Where inhomogeneities occur in the thermocouple manufacturing process, the quality of thermocouple wire and its standard voltage temperature characteristics may vary.
- Linearization errors occur because polynomials are only approximations of the true thermocouple voltage output.

2.6.7 Wiring configurations

As the voltage levels from thermocouples are very small, typically in the order of µV/°C, temperature measurements using thermocouples are susceptible to the effects of noise. Three wiring configurations are shown in the following figures:

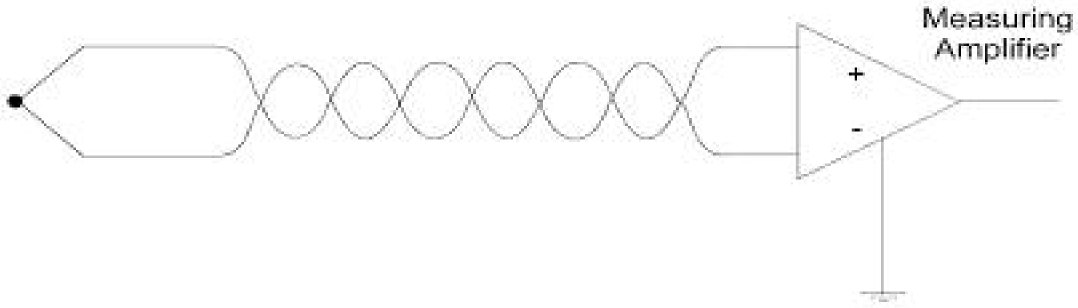


Figure 2.12
Thermocouple with no shielding

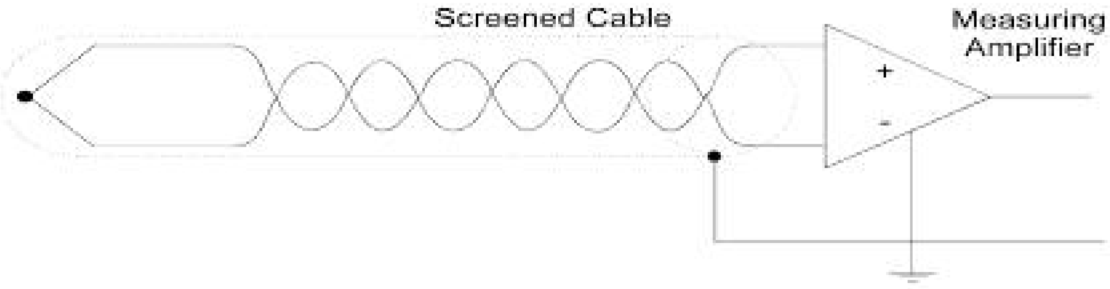


Figure 2.13
Thermocouple with thermocouple sheath and ungrounded junction

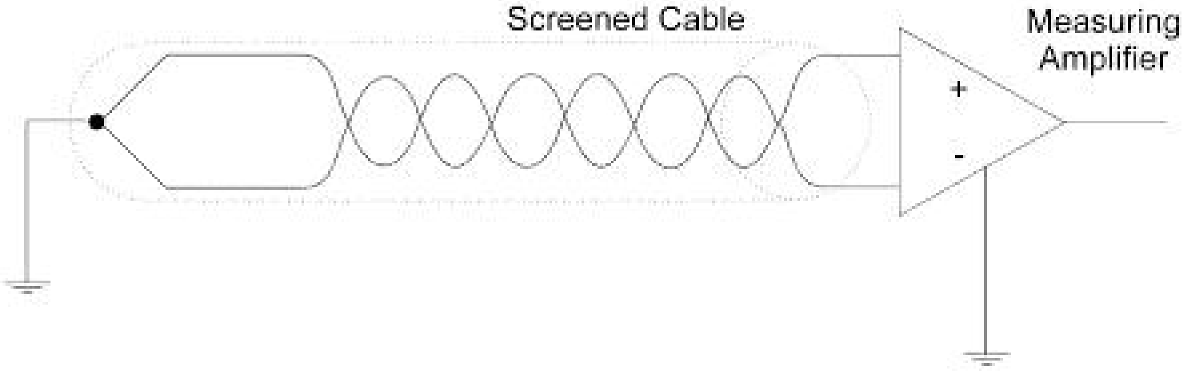


Figure 2.14
Thermocouple with thermocouple sheath and grounded junction

In addition to the wiring suggestions made above, it is important to consider isolation and over-voltage protection in the measurement circuitry, especially as a safeguard from charge buildup and other transient over-voltages on long thermocouple cables.

2.7 Strain gauges

Strain gauges are the most widely used devices for the measurement of force, or more particularly strain resulting from force. The most common type of strain gauge is the bonded resistance strain gauge, which consists of a resistive material, usually metal film a few micrometers thick, bonded to a polyester backing plate. A typical strain gauge is shown in Figure 2.15.

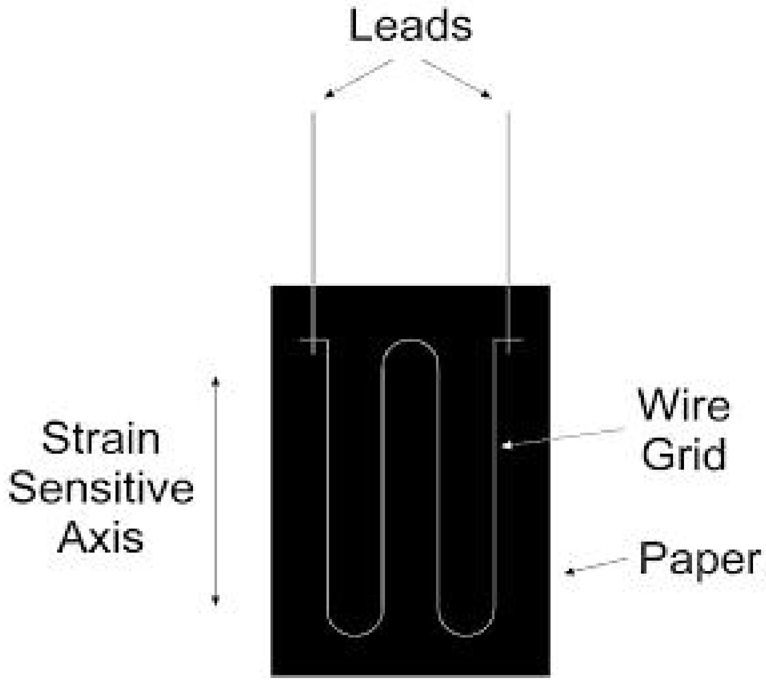


Figure 2.15
Typical bonded resistance strain gauge

The strain gauge operates on the principle that when strained, the length, cross-sectional area and resistivity of the metal film changes, thus changing the resistance of the conductor. When attached to a unit under test by an adhesive of some kind, the strain gauge experiences the same strain as the unit. The amount of strain can be measured by detecting changes in the resistance. Provided the change in length of the strain gauge is small, the relationship between resistance and strain is linear.

The ratio of the percentage change in resistance to the percentage change in length is known as the 'gauge factor' (GF) and is a measure of the sensitivity of the gauge.

$$GF = \frac{\Delta R \;/\; R_0}{\Delta L \;/\; L_0} = 1 + 2\sigma + \frac{\Delta\rho \;/\; \rho}{\Delta L \;/\; L_0}$$

Where:

R_0 = resistance in ohms
ρ = resistivity in ohms per meter
L_0 = length in meters
$\Delta R/R_0$ = fractional resistance change
σ = Poisson's ratio
$\Delta L/L_0$ = fractional change in length
$\Delta\rho/\rho$ = fractional change in resistivity

image
not
available

This can be reduced by modeling the non-linearity of the gauge in software with a suitable polynomial.

Matching of compensation resistors to the strain gauge. Where the compensation resistor in the same arm of the gauge is different by 1%, the error is 0.5%.

Measurement errors caused by accuracy; resolution of the measuring device and lead resistances

Temperature effects. The resistance of both the strain gauge and the compensation resistors vary with changes from the temperature at which a bridge is calibrated. This effect is greatly reduced by including an unstressed strain gauge in the same arm of the bridge.

Self-heating of gauges. This can be greatly reduced by energizing the bridge only while measurements are being made.

image
not
available

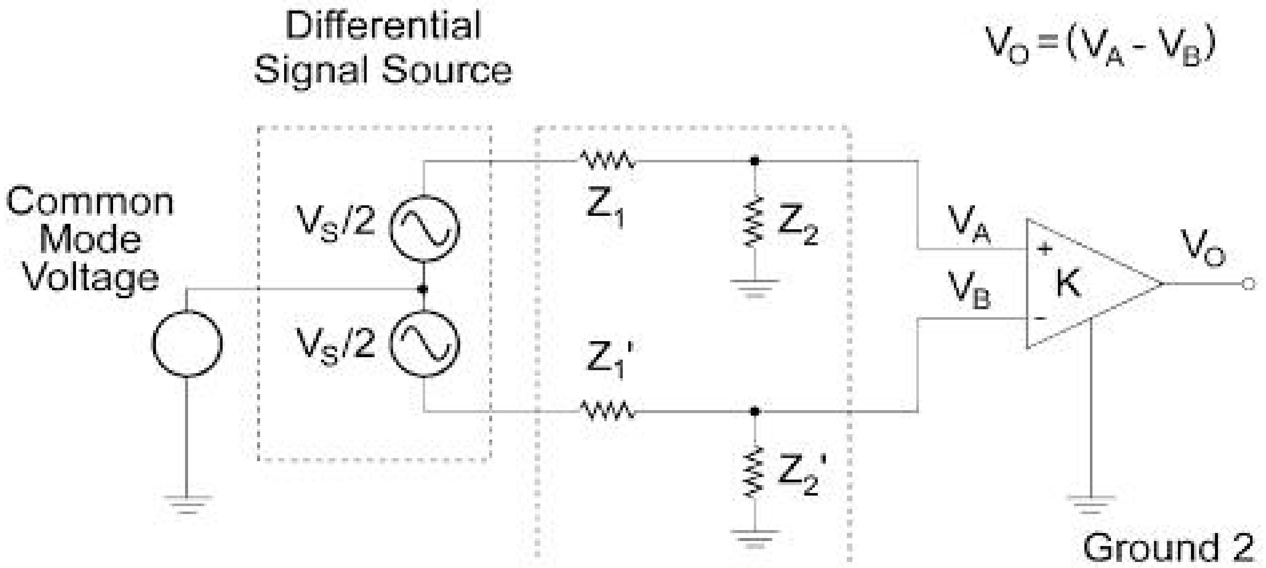


Figure 3.22
Common mode voltages

The common mode voltage V_{cm} can be calculated from the following:

$$V_{cm} = \frac{(V_A + V_B)}{2}$$

Where:

V_A = Voltage at the non-inverting terminal of the measurement system with respect to the instrumentation amplifier ground.
V_B = Voltage at the inverting terminal of the measurement system with respect to the instrumentation amplifier ground.

An example of a common mode voltage is the output from a bridge circuit, in which the small differential signal is superimposed over a much larger common mode voltage introduced by the excitation of the bridge circuit.

Common mode rejection ratio (CMRR)

Ideally, a differential amplifier would completely reject any common mode voltages present on its input signal lines and only amplify the potential difference between them. Practically, however, these devices do not totally reject common mode voltages. The common mode rejection ratio (CMRR) measures the ability of a differential input amplifier to reject signals that are common to both signal inputs.

The CMRR is defined as the ratio between the common mode signal present at the input to the amplifier and the signal produced by this voltage at the output of the amplifier, as defined by the following equation:

$$CMRR = 20\log 10(\frac{V_{cm}}{V_{out}})$$

This ratio, normally expressed in dB, can be used to calculate the output voltage error, which would occur due to a common mode voltage appearing at the input. The higher the CMRR, the better the rejection of common mode signals, and the more accurate the output due to the differential signal being measured. Typically, a CMRR of 60 dB–80 dB could be expected for a well-designed system.

Common mode input voltage limits

Practically, measurement systems also have another limitation, and this is that there is a maximum and minimum common mode input voltage allowable on each input, with

Increased noise immunity is also achieved using a balanced system, since induced noise voltages appearing on the signal wires, are equal and should be cancelled out by the differential amplifier measurement.

Bias resistors, connected between each input lead and the ground reference of the measurement system, as shown in Figure 3.28, provide a DC return path for bias currents from the inputs of the instrumentation amplifier to the reference ground.

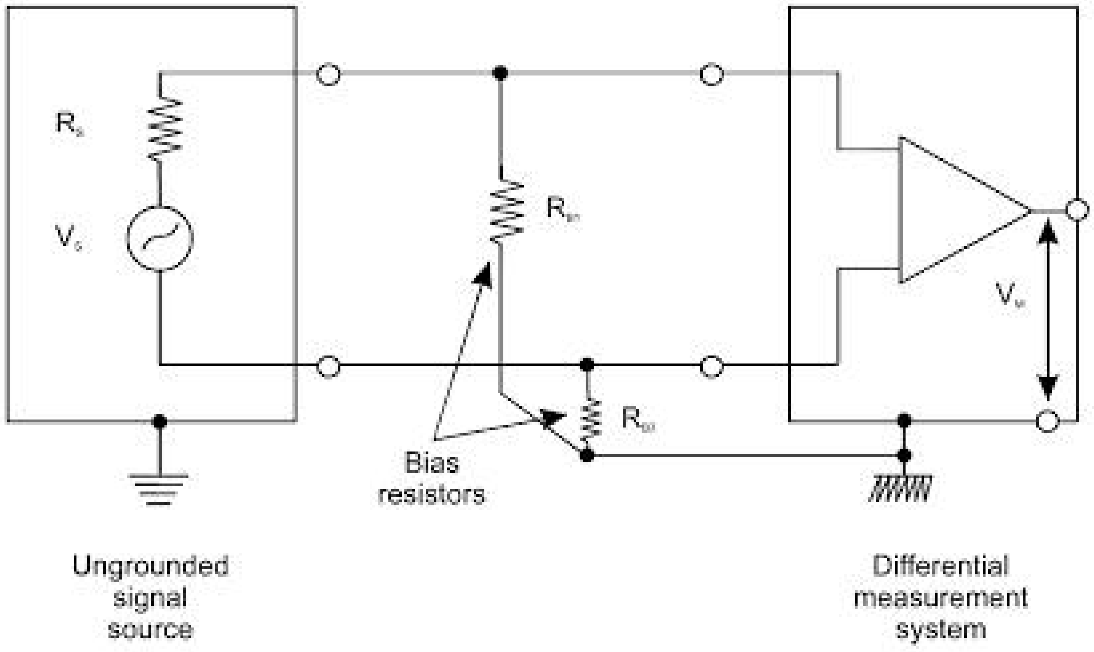


Figure 3.28
Differential measurement of an ungrounded signal source

Where the signal contains both AC and DC components (i.e. DC coupled) and the signal source has low impedance, only one bias resistor is required to be connected between the negative input and the ground reference. If the source impedance is relatively high compared to the input impedance of the instrumentation amplifier, then the imbalance caused by using a single bias resistor could lead to erroneous results. Therefore, for high source impedances both input bias resistors should be used.

For input signals, which contain no DC component (i.e. AC coupled), both bias resistors are required.

The bias resistors should be large enough to allow the source to float with respect to the measurement system ground and not to load the signal source (i.e. much greater than the source impedance), but small enough to keep the voltage at each input terminal within the input stage common mode voltage range of the measurement system. Bias resistors between 10 kΩ and 100 kΩ are typically used for low impedance sources such as thermocouples or when connecting the outputs of signal conditioning modules to data acquisition boards.

3.4.10 System isolation

To allow the measurement of signals that contain large common mode voltages, special hardware and measurement techniques are used. This typically involves isolating the measurement system from the ground reference so that signal lines, such as amplifiers, commonly used as a measurement reference, become a floating reference point.

System isolation can be carried out in the following ways:

- Using isolation transformers to reject the common mode voltage appearing on the signal lines.

This equation shows that the capacitively-coupled noise voltage is independent of the frequency of the noise source and is much greater in magnitude than in the case where the source resistance is relatively small.

Where the amplitude and the frequency of the noise source cannot be altered, the only means for reducing capacitive coupling into the signal circuit is to reduce the equivalent signal circuit resistance to ground or reduce the mutual stray capacitance. The mutual stray capacitance can be reduced by increasing the relative distance of the signal wires from the noise source, correct orientation of the conductors, or by shielding.

Magnetic field coupling

Magnetic field coupling or inductive coupling is the mechanism by which time-varying magnetic fields produced by changing currents in a noise source, link with current loops of receiving circuits. The physical representation of magnetic field coupling between a noise source and a signal circuit is shown in Figure 3.33.

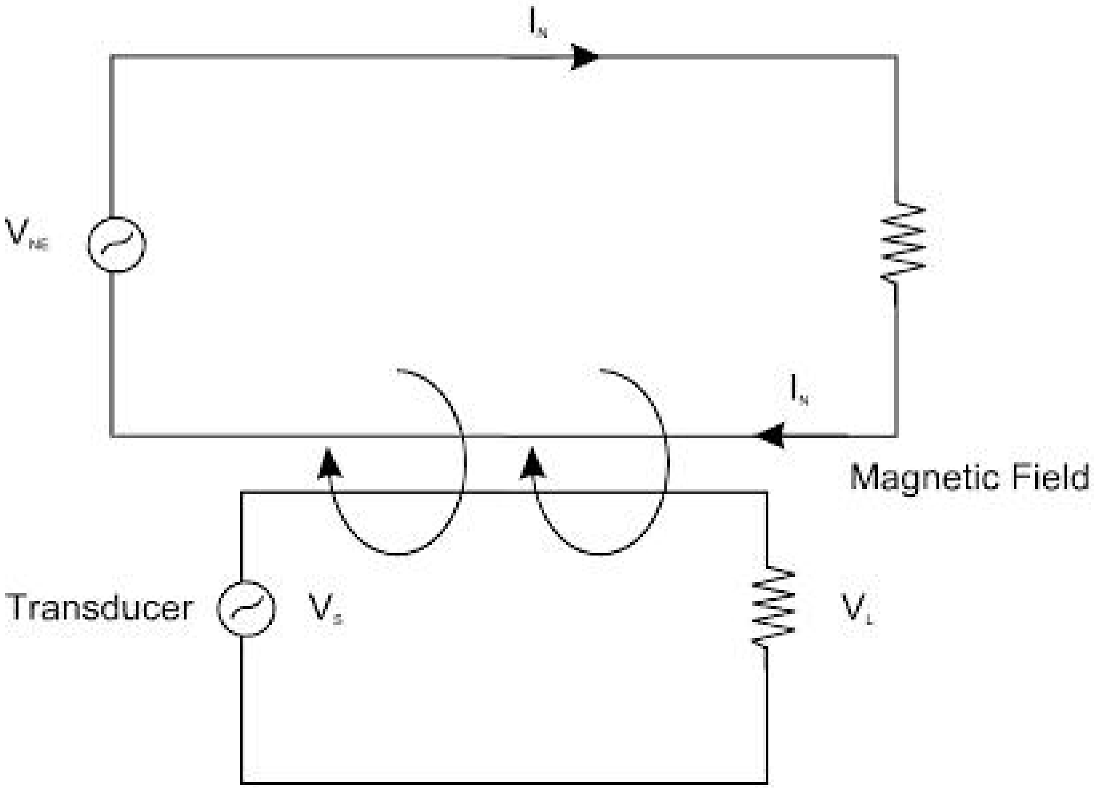


Figure 3.33
Physical representation of magnetic field coupling between a noise source and a signal circuit

Lenz's law states that the voltage, V_n induced into a closed loop signal circuit of area A is proportional to the rate of change of the magnetic field coupling the circuit loop, the flux density (B) of the magnetic field and the area of the loop. This is represented by the formula:

$$V_n = 2\,f\,BA\cos\phi\,(10^{-4})$$

Where:

f = the frequency of the sinusoidal varying flux density
B = the rms value of the flux density (gauss)
A = the area of the signal circuit loop (m^2)
ϕ = the angle between the flux density (B) and the area (A).

This equation indicates that the noise voltage can be reduced by reducing B, A, or $\cos\phi$. The flux density (B) can be reduced by increasing the distance from the source of the

of a circuit will drive a current of 2 A around the current loop if its resistance is 0.5 Ω. Where the current flow is significant, and the ground loop created by earthing of the shield has a large area, shield currents may inductively couple unequal voltages into the signal cables and be a source of interference. Where possible, shields should be earthed at one end only.

The placement of shield earths depends on the grounding of the signal source and the type of measurement system used. Figure 3.38 shows the preferred shield grounding when measuring an ungrounded signal source, using a measurement system where the signal lines are referenced to the amplifier common. It is assumed that amplifier common, although normally connected to ground may have a potential ($\Delta Vg1$) relative to ground potential. $\Delta Vg2$ represents the difference in ground potential. The circuit equivalent for this system shows that in this configuration neither of the noise voltages ($\Delta Vg1$ or $\Delta Vg2$) appears across the input terminals of the amplifier. Instead, if the shield was earthed at point B, then the noise voltage across the input terminals of the amplifier would be the voltage across the impedance of C_2 as part of the voltage divider formed with C_1.

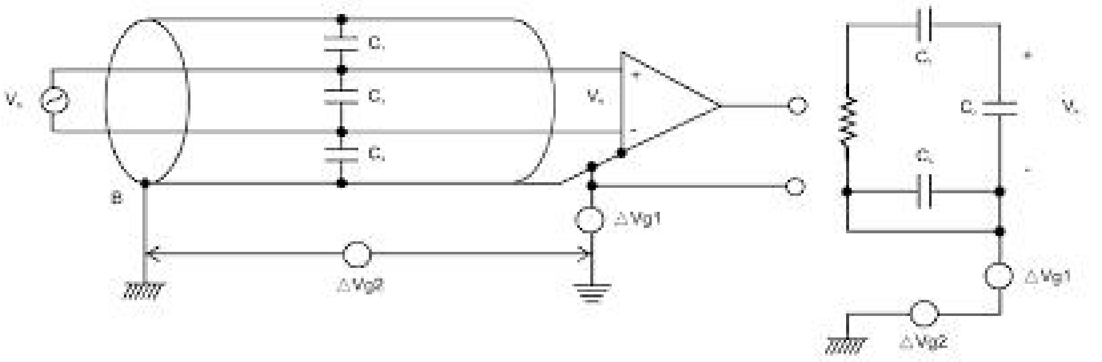


Figure 3.38
Shield grounding when measuring an ungrounded source with a grounded measurement system

When an ungrounded (differential) measurement system is used to measure a grounded source the preferred cable shielding is shown in Figure 3.39. The voltage $\Delta Vg1$ represents the potential of the source common above earth ground potential.

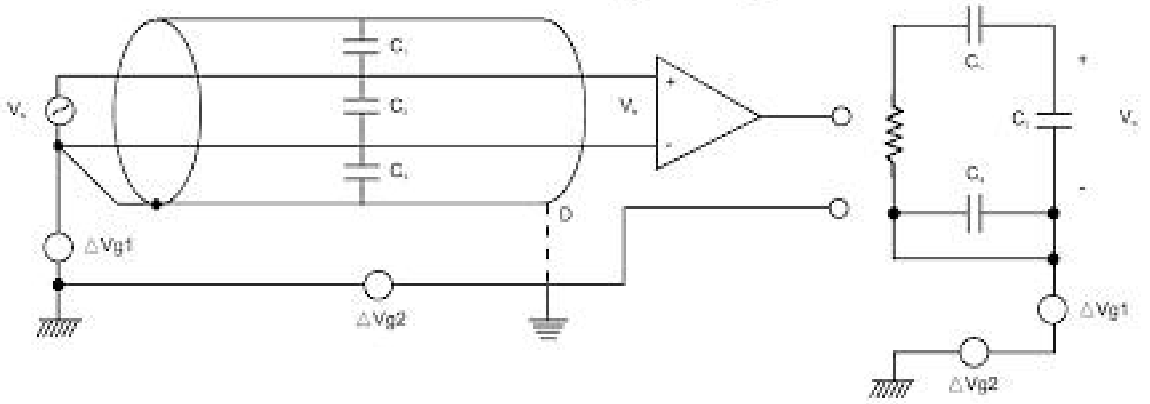


Figure 3.39
Shield grounding when measuring a grounded source with an ungrounded measurement system

The equivalent circuit for this measurement system again shows that the noise voltage appearing across the input terminals of the amplifier, is zero. If the shield was grounded at the other end of the cable at point D, then the noise voltage across the input terminals of the amplifier would be the voltage across the impedance of C_2 as part of the voltage divider formed with C_1.

Where the signal circuit is required to be grounded at both ends, the difference in ground potential and the susceptibility of the ground loop to inductive coupling deter-

4

The PC for real time work

Introduction

The key to the effective use of PCs in data acquisition and process control is the careful matching of the specific requirements of a particular application to the appropriate hardware and software available.

The personal computer consists of the following main components:

- System unit (CPU, memory, interrupt controller, DMA controller, power supply)
- I/O devices (hard disk, floppy disk, keyboard, mouse, display, COM port(s), CD
- Hardware BIOS (Basic input/output system)
- Operating system (WIN 95, 98, 2000, NT)

This chapter examines some of the important features of the PC as they relate to the data acquisition techniques studied in other sections of this course. The topics covered are:

- Operation of interrupts
- Operation of direct memory access (DMA)
- Data transfer speeds (polled I/O, interrupt, DMA, repeat inst)
- Memory (base memory, expanded memory, extended memory)
- PCI, Compact PCI, ISA bus, EISA bus
- Interfacing techniques to the PC bus
- Compact PCI

4.1 Operating systems

An operating system is the software responsible for managing the computer's resources (including hardware and software), processing commands, and controlling program execution. It provides an interface between the application software and the hardware of a particular system. Operating systems manage communications with the disk drive, display, printer, and

Graphical user interfaces offer several advantages to an application program:

- Communication between the application program and the user is in a natural symbolic form that more closely resembles the human thinking process
- Communication between the user and the computer is faster, as the user is not required to enter program names
- The learning process for programs is faster, since the communication is more natural and symbol oriented
- Programs can be more powerful. In a conventional program, intricate operations that are generally difficult for the user and often purposely omitted by the program designer, are now more feasible.

Because of these advantages, most developers believe that graphical user interfaces will be the primary interfaces used by application programs in the future. However, the graphical user interface does have some disadvantages with respect to current hardware and software:

The amount of memory and disk space needed to support the graphic environment is much larger than that needed for conventional (or text oriented) environments

Higher processing speeds are required to support the graphic environment, as larger amounts of data must be moved around, particularly from the hard disk to the display

Since hardware must be faster and contain more memory, the hardware costs are consequently higher

The amount of programming support needed for a graphic environment is more extensive.

Virtual tools

The Microsoft Windows graphical user interface, provides an ideal environment for virtual tools. In conjunction with digital and analog acquisition and control boards, virtual tools allow users to set up experiments, acquire data, and graphically display the results on the screen in real time.

Virtual tools can be developed by professional software engineers, or by less experienced users, by using packaged software. Specific software packages are available that allow the user to edit, control and logic strategies. The user can move function block icons from on-screen libraries onto the desktop and connect them into a flowchart.

Display editors allow the user to set up color graphic displays, such as instrument panels, control panels and charts. When the display panel is complete, the user simply connects function icons to it for real-time graphical output. A typical example of a virtual tool would be an oscilloscope with the CRT display and front panel controls emulated on the screen, and signal input provided by an analog to digital conversion board.

Virtual memory

Microsoft Windows allows an application program to apparently access more memory than is physically available. This virtual memory is achieved by using reserved space on the hard disk drive to emulate physical memory. Windows can also manage memory for the application programs on a dynamic basis, through the use of swap files.

4.1.3 UNIX

The UNIX operating system, originally developed by AT&T Laboratories in the late 1960s, is a powerful multi-user and multitasking operating system. UNIX is available for nearly any computer that contains sufficient real memory and a fast hard disk.

- IRQ inputs can be configured as edge-sensitive (normal) or level-sensitive via the initialization command word (ICW) register

4.2.5 Initialization required for Interrupts

Before interrupts can be handled correctly the following functions must be performed:

Initialize the 'interrupt vector table' located in the first 1024 (1 k) bytes of system memory to contain the addresses of the interrupt service routines of each of the 256 possible interrupts. Each four-byte address consists of an instruction pointer (IP) and code segment (CS) value. A large number of these are initialized by the BIOS and DOS as part of the system boot and operating system startup procedures. Initialize the 8259A PIC(s). This is largely initialized by the BIOS as part of the system boot.

Enable the system interrupt INTR by setting the interrupt enable flag bit IF in the FLAGS register.

4.2.6 I/O devices requesting interrupt service

When an I/O device asserts an interrupt request, an ordered sequence of events occurs, to direct the CPU to the interrupt service routine (ISR) that will service the specific request.

We will assume that any system and remote I/O device initialization that is required to allow the interrupt request to be handled correctly has already occurred.

The sequence of events is as follows:

- The I/O device hardware activates an interrupt request by asserting its IRQx line from 'low' to 'high'. This signal is usually 'latched' high by an interrupt request latch on the I/O device and remains high until the latch is reset and the interrupt is acknowledged (thus allowing further interrupts). These last tasks are performed by the ISR (see Interrupt service routines, p. 75).
- The interrupt controller receives this IRQx interrupt request and prioritizes it with other requests that may be coming in or pending. The interrupt controller will then send an interrupt request to the CPU on the INTR signal line under the following conditions:
    - This is the only interrupt request.
    - A lower priority interrupt is in progress.
    - Several interrupts are pending but this interrupt has the highest priority.
- If the CPU has interrupts enabled, it acknowledges the interrupt request by sending two INTA pulses to the interrupt controller. The first freezes the priority levels in the interrupt controller, while the second requests an 8-bit pointer value, called the 'interrupt type'.
- The interrupt controller places the 8-bit interrupt type onto the CPU data bus. This 'interrupt type' byte is the means by which the CPU knows where to look for the address of the interrupt service routine that will service the I/O device.

  It is used to index the 'interrupt vector table' located in the lowest 1024 (1 k) bytes of system memory. Each of the table's 256 entries (four bytes each) contains the segmented memory addresses of all the ISRs.

  The CPU multiplies the interrupt type by four (4) to get the offset to the interrupt vector table where the address of the appropriate ISR will be located. Therefore, for interrupt type S, the address of the initial service routine (ISR) to be executed will be address 20h (32nd byte).
- The CPU saves the information necessary to allow the program currently being executed to resume execution at its next instruction upon completion of the ISR.

*image
not
available*

- A two-byte address register containing the current address to which data must be written.
- A two-byte count register, which holds the number of bytes/words to be written in total.
- A two-byte count register containing the current bytes/word count of each channel.
- Control lines that allow the reading/writing of data from/to memory.

As each device only supports 16-bit addresses (limiting access to 64 kB of memory), each DMA channel has an associated 'page register' in system I/O memory to provide the added upper 4 address lines required to address the complete 20-bit (1 MB) system base address space.

4.3.2 Initialization required for DMA control

Before any DMA operation can occur, the DMA controller must be initialized.

Items requiring initialization are as follows:

- Select whether the DMA controller will read or write to memory.
- Configure the type of DMA data transfer. Four modes of DMA data transfer are available:
  - Single transfer mode
    The DRQx signal must be asserted for every byte/word transferred.
  - Block transfer mode
    A single DRQx signal DMA request initiates the transfer of an entire block of data.
  - Demand transfer mode
    Data is transferred as long as the DRQx signal DMA request is asserted and the terminal count has not been reached.
  - Cascade mode
    All DMA channels are programmed for single transfer mode.
- The total number of bytes to be transferred is loaded into the appropriate total byte/word count register. The current byte/word count register is then automatically initialized.
- The memory address to which the first data byte will be read/written is loaded into the start memory address register. The current memory address register is automatically initialized.
- The 4-bit page register corresponding to the upper four bits of the 20-bit address is written using the I/O port addresses of the PC.
- The DMA channel priorities should be set. When the PC is booted up, the ROM BIOS sets the priorities so that the lowest numbered channel has the highest priority.
- The DMA controller(s) channels that are to be used should be enabled. Channels to be enabled have the channel mask register bits cleared.

4.3.3 I/O devices requesting DMA

Assuming that the necessary DMA controller has been initialized, the standard operation, when an I/O device requests DMA data transfer on one of the channels of the DMA controller, is as follows:

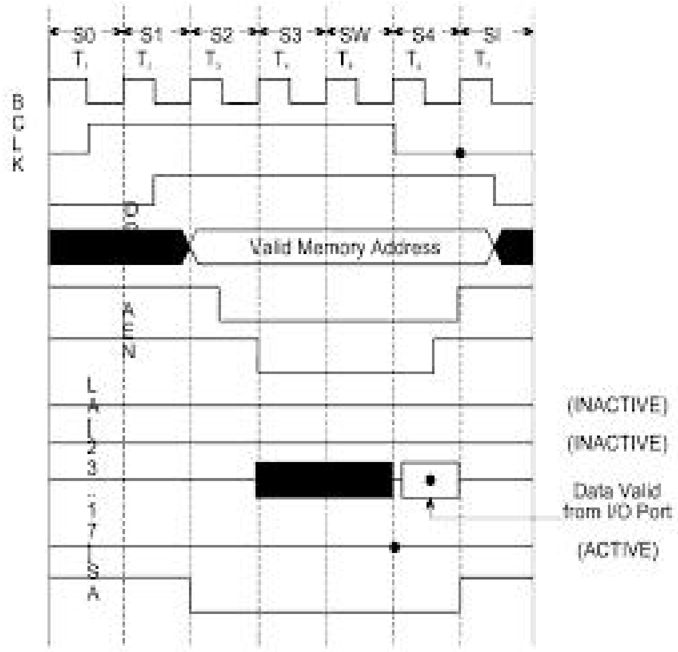


Figure 4.3
Timing chart of a memory-write / I/O-read DMA cycle

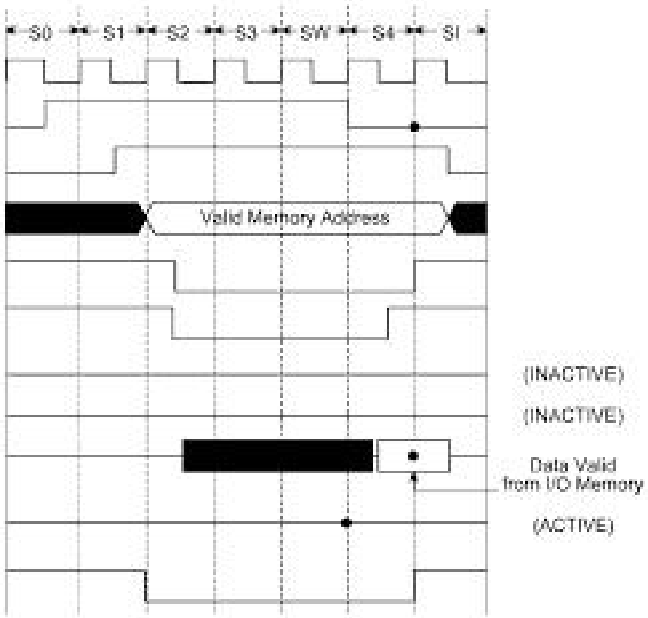


Figure 4.4
Timing chart of an I/O-write / memory-read DMA cycle

4.4 Repeat string instructions (REP INSW)

You may know that the XT used DMA to transfer data from the hard disk, while the AT does not. The AT's disk transfer rates are much higher (over 500 kB/sec for a standard AT bus interface) than the XT's. The question arises: how can the AT achieve such high data transfer rates without using DMA?

The answer is found in new instructions added to the Intel 80286 (and higher) processors. These are the repeat input and output string instructions. On the 8088/8086, repeat instructions are available for moving and processing strings in memory, but not to memory add-resses from I/O addresses (strings are just continuous sequences of bytes). Subsequent processors extended this to include I/O locations, from where a data acquisition board's samples originate.

To obtain a series of data samples from a board, the program initializes a counter register, sets up a destination memory address register and executes the repeat input string word instruction (REP INSW), giving the I/O address of the board's data register. The instruction automatically reads a sample value, stores it at the memory address pointed to by the destination address register, increments this register, decrements the counter and continues trans-

Extended 7-BCLK 8-bit memory access

The 8-bit machine cycle with additional wait states is almost exactly the same as the standard cycle shown above in Figure 4.5. CHRDY is driven and held by the expansion board somewhere during the cycle before T_C begins. The system board begins sampling CHRDY twice every machine state, beginning in the last default wait state (T_5 in Figure 4.6). Since it finds it active (low), it runs another wait state (T_6), during which it continues to sample /NOWS and CHRDY. Since CHRDY is found to be inactive at the end of T_6, the machine cycle is completed with a T_C state.

The additional wait state has the effect of lengthening the bus cycle by one bus clock, which may allow slower devices to interface to the bus. If more time is required by the I/O bus device, CHRDY may be held low until the required cycle time is completed. 2.5 μs is a recommended maximum length of time for which CHRDY may be asserted.

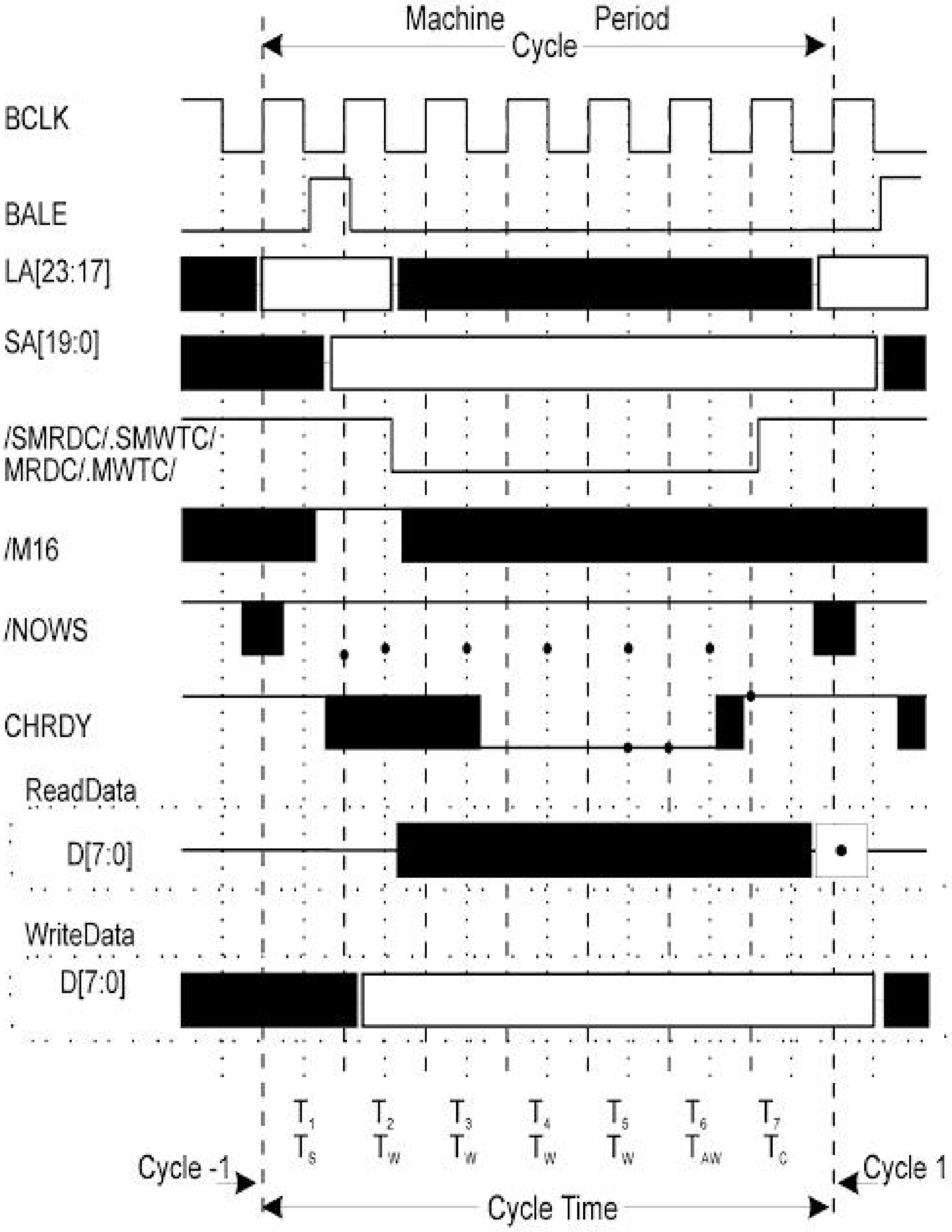


Figure 4.6
Timing chart of an extended 7-BCLK 8-bit memory access cycle

Shortened 2-BCLK 16-bit memory access

If the memory device is capable of transferring data in a shorter time than the default 3-BCLK period, it may, after asserting /M16, assert /NOWS to execute the cycle in two BCLKs.

As before, the system board finds /M16 asserted at the end of T_S, so it begins to run a standard 3-BCLK 16-bit memory access. However, halfway through T_2 (which would have been a wait state), it finds /NOWS asserted. It therefore converts the wait state into a T_C state and completes the bus cycle.

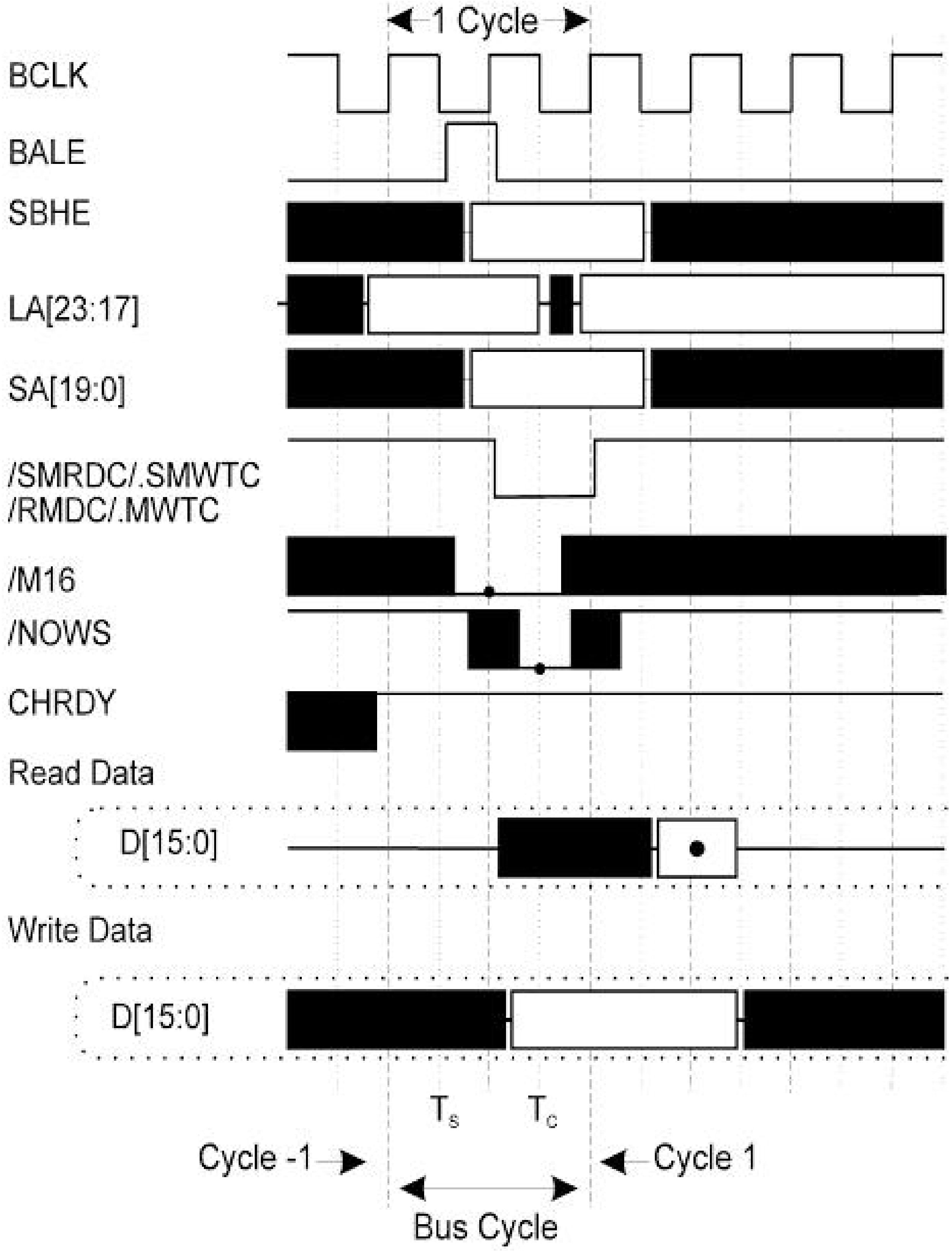


Figure 4.10
Timing chart of a shortened 2-BCLK 16-bit memory access cycle

I/O-read and -write cycles

The following three timing charts show instances of 8-bit I/O access:

- A standard 6-BCLK cycle (Figure 4.11)

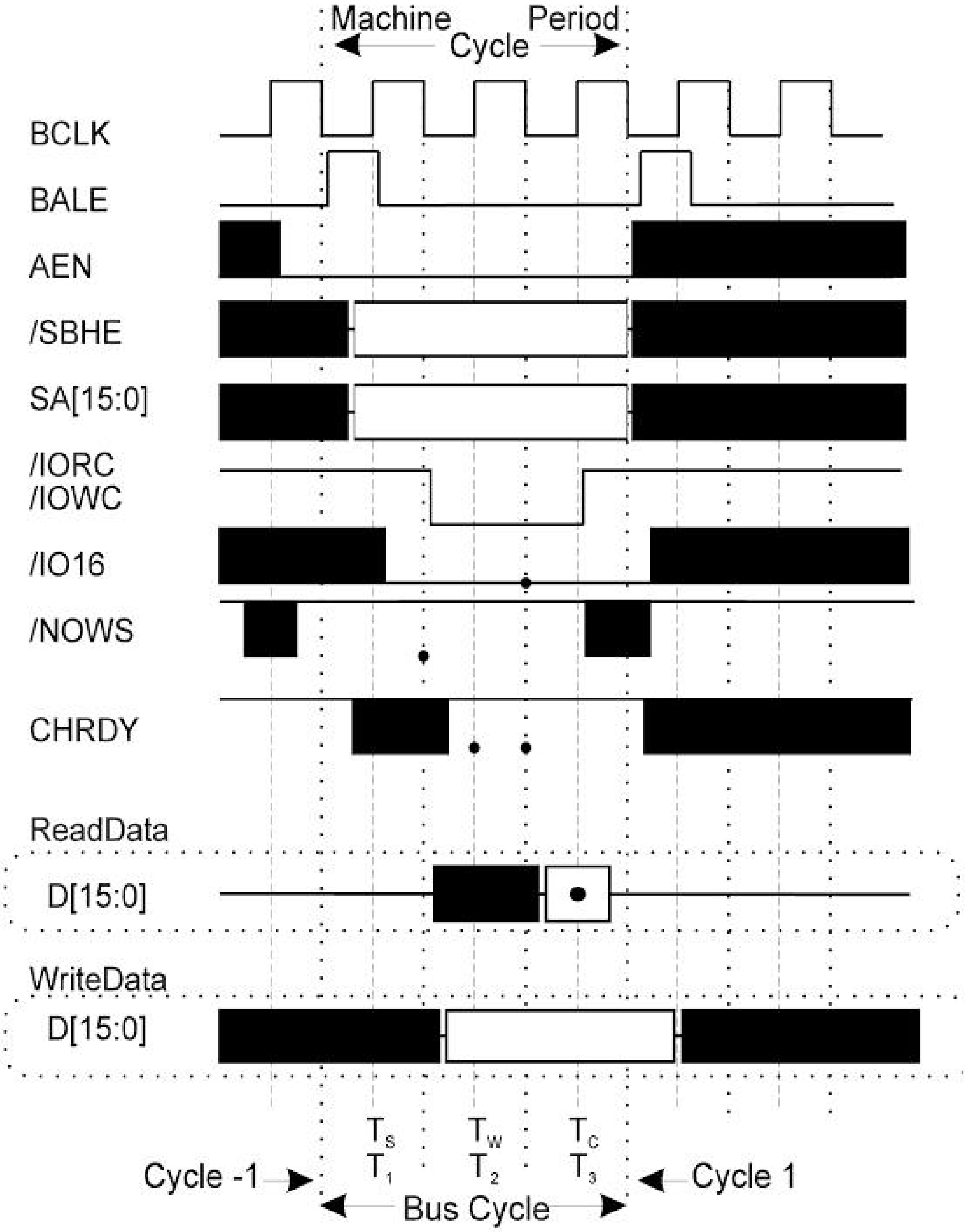


Figure 4.14
Timing chart of a standard 3-BCLK 16-bit I/O access cycle

Extended 6-BCLK 16-bit I/O access

Again, if a 16-bit I/O device cannot meet the setup and hold times of the standard 16-bit I/O cycle, it may drive CHRDY to cause the system board to insert more wait states, until CHRDY is disasserted by the expansion board.

- Support for multitasking operating systems
- Diagnostic routines

For compatibility with different types of hardware (and EMM software revisions), applications should communicate with the EMM via the assigned software interrupt. If memory is acquired, used, and released via the EMM, other programs will also be able to make use of the memory.

The advantage of EMS is that it provides additional data memory to DOS and 8088-type computers.

The limitations of EMS are:

- It requires special hardware and software drivers to operate.
- It cannot execute programs from EMS; it is used for data storage only. For example, in a DOS system with 32 kB of expanded memory and over 1 MB of base and extended memory, the maximum executable program size (without using disk overlays) is the DOS 640 kB limit, less the amount of memory used by DOS and other installed drivers.
- Much slower access time than linear base and extended memory.
- Programs cannot access EMS directly; they must go via the memory manager and this results in more complex code.

4.7.3 Extended memory (XMS)

Extended memory is the physical linear memory found above the 1 MB mark. 80286 and 80386SX processors can address up to 16 MB of base and XMS while 80386DX and 80486 processors can address up to 4 GB of this type of memory. XMS is memory addressed directly by the processor (and hence the application program) and is therefore simpler, quicker, and more efficient. Extended memory is only available as normal application memory when the processor is in protected mode; it follows that only 32-bit protected mode operating systems and extensions - such as OS/2, UNIX and MS-Windows, but not DOS – can make this memory available to programs.

4.7.4 Expansion memory hardware

All systems support a certain amount of memory on the main board. On most PCs, eight SIMM (single in-line memory module) sockets are standard. Each socket usually accepts 256 kB, 1 MB, 4 MB SIMMs and higher. Additional memory may be added on using an expansion card in the I/O channel. Some system boards provide access to the high-speed local processor bus with additional proprietary memory board slots. On 80386 systems and higher, memory may be configured dynamically in software as XMS or EMS, as is needed by applications. Main board memory has a shorter access time than memory on add-on boards; these in turn have much shorter time than memory attached to the I/O channel.

4.8 Expansion bus standards (ISA, EISA, PCI and PXI bus)

4.8.1 ISA bus

One of the primary reasons for the success of the first IBM personal computer (PC) was its 'open system' design. IBM encouraged the development of compatible add-on products by third party manufacturers by releasing details of its 8-bit expansion bus. This bus formed the basis of the industry standard architecture (ISA) bus.

DMA cycles, data is transferred directly from the I/O device to memory (or vice versa) on the data lines while the processor is disconnected from them. The DMA controller drives the control lines in this case. Bus masters may also take control of these lines.

D[15..8]

D[15..8] are the high eight bits of the 16-bit bi-directional data bus. They are similar to the lower eight data lines, D[7..0]. 8-bit wide transfers must use D[7..O]. If the currently running software requests a 16-bit transfer from an 8-bit device, the system board automatically converts it into two 8-bit cycles on D[7..O]. Adapters capable of 16-bit transfers must indicate this using /M16 or /IO16 during cycles addressed to them or the system board will convert the instruction into two 8-bit instructions. /SBHE (explained later), is asserted by the system board during 16-bit cycles.

LA[23..17]

The LA17 to LA23 (latchable address) lines form part of the latchable address bus. (The remaining lines of the latchable address bus, LA[16..2] and LA[31..24] are wired to the EISA connector and are not available in ISA systems. SA[19..0] must be used instead.)

LA16 to LA23 are unlatched and, if required for the whole bus cycle, must be latched by the addressed slave. During standard cycles, they are valid during the active time of the BALE signal (explained later) and remain valid for at least ½ BCLK period after the command signals are asserted.

During DMA or ISA bus master cycles, LA[23..17] are valid at least one BCLK before the command signals are asserted. They may be driven by an expansion board acting as a bus master. These lines may be latched with the trailing edge of BALE.

These address lines are provided in this way because they are pipelined from one cycle to the next, and to reduce address delay when they are used to decode a block of bus-attached memory.

SA[19..0]

Address lines SA0 through SA19 are used to address system bus I/O and memory devices. They form the low-order 20 bits of the 32-bit address bus. (However, only 24 of the 32 address lines are normally available in ISA systems.)

On normal cycles SA0 to SA 19 are driven onto the bus while BALE is high and they are latched by the system board on the trailing edge of BALE and are therefore valid throughout the bus command cycle.

During DMA and 16-bit ISA bus master cycles, they are driven by the DMA logic and bus master respectively. They should be valid one BCLK before the command signals and normally stay valid one BCLK after the command signals end.

With 20 address lines it is possible to address 1 MB of memory, but not all address locations are available. Base system memory, system ROMs and display memory all use addresses in this range.

The processor, using the IN and OUT instructions, addresses I/O devices with lines SA0 through SA15, while SA16 through SA19 are not used and are held inactive. Most PC I/O devices only decode the first ten address lines, (SA0 to SA9, which correspond to I/O addresses 0h to 3FFh) so care must be taken when addressing I/O devices with SA10 to SA15.

/SBHE

/SBHE (system bus high enable) is an output-only signal. When low, it indicates to the expansion board that the present cycle expects to transfer data on the high half of the D[15..0]

image
not
available

image
not
available

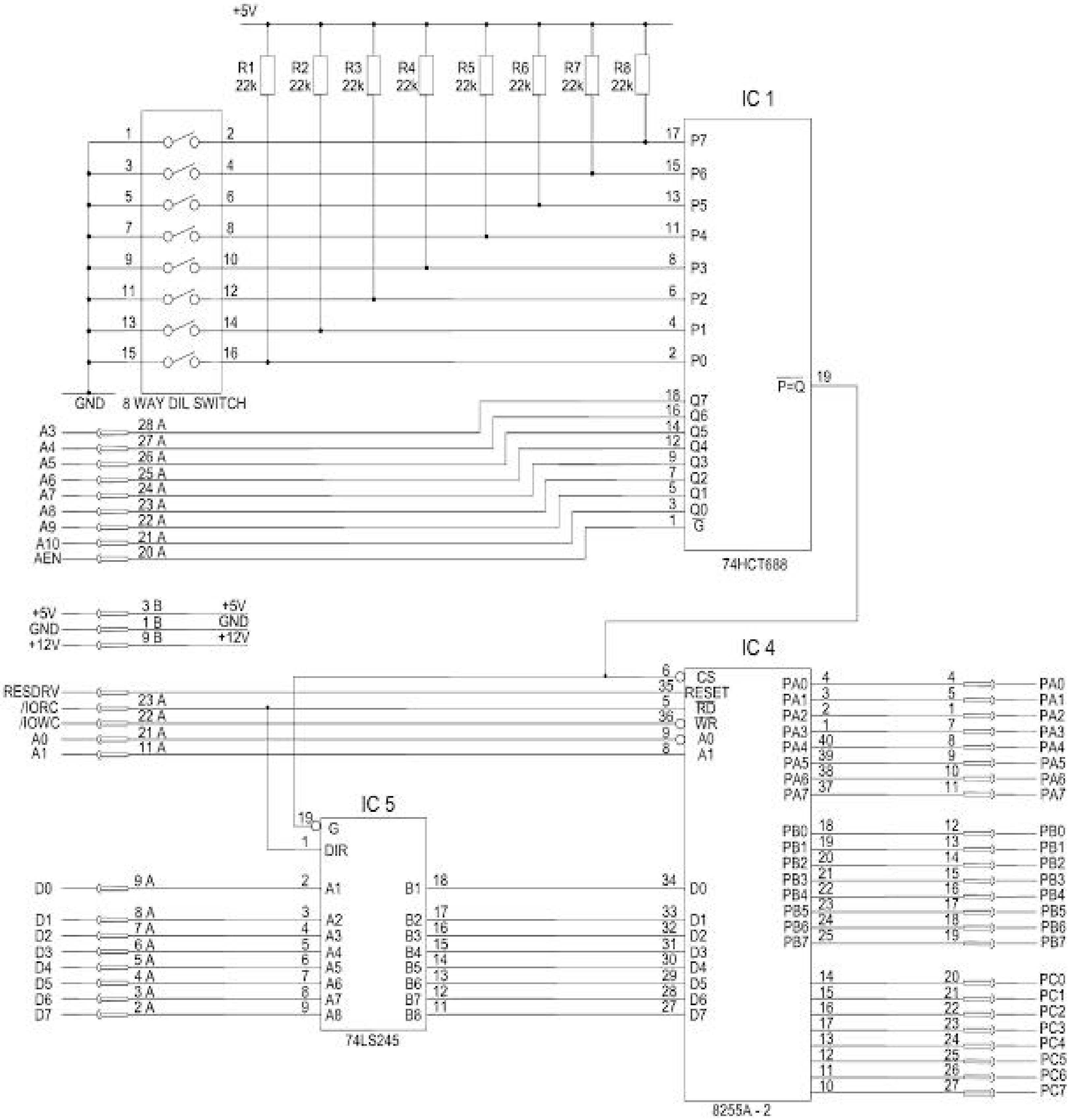


Figure 4.18
PCI structure

In 1998, the compactPCI and PXI bus were developed to combine the attributes of the PC's PCI bus with the industrial card connection system of the VME. The VME bus had been used in various forms for many years as an industrial bus rack for mounting electronic PCBs. The PCI bus is a very fast and easy way of connecting the computer to the outside world. By combining the PCI with the VME the best of both worlds was accomplished. Instead of having to open the PC and insert a PCI card, the user can just slide the card into the front of the chassis. It uses a 5-row 2 mm-pitch connector with impedance-matched pins and sockets. The integrated shielding system doubles the bus card's capacity from four to eight. The PXI version of the compactPCI has extra features that the basic compactPCI does not. These include timing and triggering functions so multiple boards can perform synchronous data acquisition, and the ability of one card to trigger another without the intervention of the system.

5

Plug-in data acquisition boards

5.1 Introduction

In recent years, the distinction between separate data acquisition systems and control systems has narrowed because an increasing number of real-life systems are designed not only to acquire data, but also to act on it. This is true of a wide range of plug-in data acquisition and control boards now available. Commonly used multi-purpose plug-in data acquisition boards, currently on the market, typically combine all aspects of data acquisition and control. This includes analog input circuitry for measuring and converting analog input voltage signals to digital format, analog output circuitry for generating analog output voltages from digital control signals, counter/timing circuitry and digital I/O interfaces. Depending on the number of analog inputs/outputs and digital inputs/outputs required for a particular application, multi-purpose boards represent the most cost effective and flexible solution for DAQ systems. Also available and widely used are plug-in boards that specialize in each of the data acquisition and control functions just mentioned.

Examples of these plug-in boards are:

- Analog input (A/D) boards
- Analog output (D/A) boards
- Digital I/O boards
- Counter/timer I/O boards

Computer plug-in data acquisition and control boards often represent the lowest cost alternative for a complete data acquisition and control system. As they interface directly to the host computer's I/O expansion bus, they are generally compact, and represent the fastest method of gathering data and/or changing outputs. These boards are most commonly used in applications where the computer is close to the sensors being measured or the actuators being controlled. Alternatively, they can be interfaced to remotely located transducers and actuators via signal conditioning modules known as two-wire transmitters.

For simplicity this chapter looks at the different aspects of data acquisition and control separately, briefly describing the main components of typical plug-in boards their uses in a

for the output of the amplifier. This means that the highest allowed settling time and the reduction in throughput caused by it, is imposed for all amplifier gain settings. More advanced A/D boards take into account the input range and amplifier gain required, thereby increasing throughput at higher signal level input ranges where lower gain settings are required.

5.2.3 Channel-gain arrays

On the original A/D boards the address of the channel to be sampled was written to the multiplexer, the gain setting sent to the programmable gain amplifier (PGA) and once the signal was settled an A/D conversion was initiated. The data was subsequently read and transferred to the PC's memory. This incurred a large software overhead. Background operation using interrupts is difficult and slower than polled I/O and accurately timed samples and higher speed data transfer methods such as DMA and repeat instructions are impossible in either case.

The use of channel-gain arrays (CGA) on many A/D boards overcomes these limitations. The channel/gain array is a programmable memory buffer on the A/D board, which contains the channel address and gain setting for each input channel to be sampled. The gain of the amplifier for a particular channel is set by the internal hardware preceding the sampling of the channel, based on the gain value read from the channel/gain array. Where a single PGA is provided for all channels, the gain required for each channel is stored in a channel/gain array. If there are individual PGAs for each input channel, the gains for each input amplifier are stored in a gain array. The gain of each remains the same until overwritten by the software. Channel-gain arrays vary in size from a few channel/gain pairs (one for each channel), to many thousands of pairs.

5.2.4 Sample and hold circuits

As shown in Figure 5.2, a sample-and-hold (S/H) device consists of an analog signal input and input buffer, an analog signal output and output buffer, a charge-storing device, usually a capacitor, and a control input that controls the switching circuitry, which in turn, connects the input to the output.

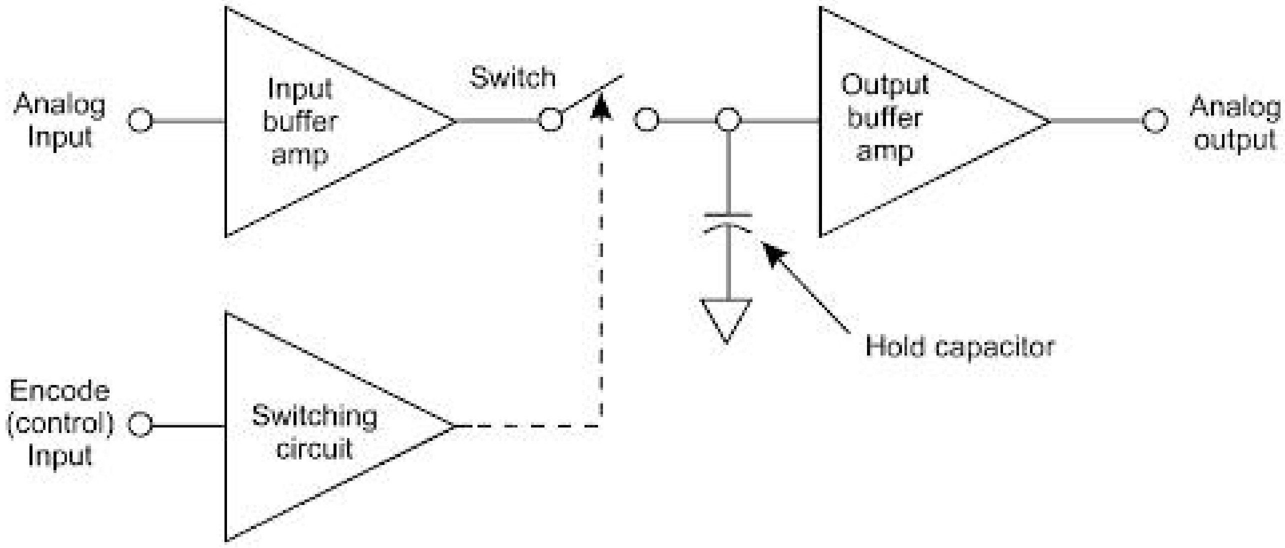


Figure 5.2
Functional diagram of a sample-and-hold device

As its name implies, a S/H has two operating states. When in sample mode, a sample command applied to the control input closes the internal switch, thereby causing the output to track the input as closely as possible. In this mode, the hold capacitor charges to the voltage

The operation of a dual slope integrating A/D converter is based on the principle that the output of an integrating amplifier to a constant voltage input is a ramp whose slope is negative and proportional to the magnitude of the input voltage.

At the start of the A/D conversion, a fixed counter is cleared to zero and the unknown analog input voltage is applied to the input of the integrating amplifier. As soon as the output of the integrating amplifier reaches zero, a fixed interval count begins. After a predetermined count period, *T*, the count is stopped. For a positive analog input voltage, the output of the integrating amplifier has reached a negative value proportional to the magnitude of input analog signal. This is shown in Figure 5.5(b). If the analog input varies during the fixed count time interval, then the output of the integrating amplifier is proportional to the average value of the input over the fixed time interval. This is especially useful for elimination of cyclical noise and/or mains hum appearing at the input.

At this point, the count register is again cleared. A negative fixed voltage reference is now applied to the input of the integrating amplifier and the count begins. When the output of the integrating amplifier again returns to zero the count is stopped. The average value of the input analog signal is equal to the ratio of the counts multiplied by the reference voltage. This is very effective in averaging and therefore eliminating cyclical noise appearing at the analog input.

Integrating A/D converters, generally include an additional and preceding phase, during which the device carries out a self-calibrating, auto-zero operation. The stability, accuracy, and speed of the clocking mechanism, the duration of the count period, and the accuracy and stability of the voltage reference, determine the accuracy of the device.

These devices are low speed, typically a few hundred hertz maximum. However, they are capable of high accuracy and resolution at low cost. For this reason they are principally used in low frequency applications, such as temperature measurement, in digital multimeters and instrumentation.

Important A/D parameters

Analog to digital conversion is essentially a ratio operation, whereby the analog input signal is compared to a reference (full-scale voltage), converted to a fraction of this value, and then represented by a digital number. In approximating an analog value, two operations are performed. Firstly the quantization or mapping of the analog input into one of several discrete ranges, and secondly the assignment of a binary code to each discrete range. Figure 5.6(a) shows the ideal transfer function of a 3-bit A/D converter with a unipolar (0 V to FSV) input. The horizontal axis represents the analog input signal as a fraction of full-scale voltage (FSV) and the vertical axis represents the digital output. An *n*-bit A/D converter has 2^n distinct output codes. While not used in practical DAQ systems, a 3-bit A/D converter represents a convenient example since it divides the analog input range into $2^3 = 8$ divisions, each division representing a binary code between 000 and 111. Figure 5.6(b) shows the ideal transfer function of a 3-bit A/D converter with a bipolar (–FSV to +FSV) input. This is equivalent to the unipolar transfer function except that it is offset by –FSV.

image
not
available

is returned as 0 × 000h, while the highest digitally coded value of 0 × FFFh (for a 12-bit ADC), represents 4.9976 V. 0 × 800h represents the mid-scale voltage of 0 V.

Conversion time

The conversion time of an A/D converter is defined as the time taken from the initiation of the conversion process to valid digital data appearing at the output. For most A/D converters, conversion time is identical to the conversion rate. Therefore, an A/D converter with a conversion time of 25 µs is able to continuously convert analog input signals at a rate of 40,000/sec. For some high-speed A/D converters, pipelining allows new conversions to be initiated before the results of prior conversions have been determined. An example of this would be an A/D converter that could perform conversions at a rate of 5 MHz (200 ns conversion time), but actually took 675 ns (1.48 MHz conversion rate) to perform each individual conversion.

Errors in A/D converters

Errors that may occur in A/D converters are defined and measured in terms of the location of the actual transition points in relation to their locations in the ideal transfer characteristic. These are discussed below.

Quantization uncertainty

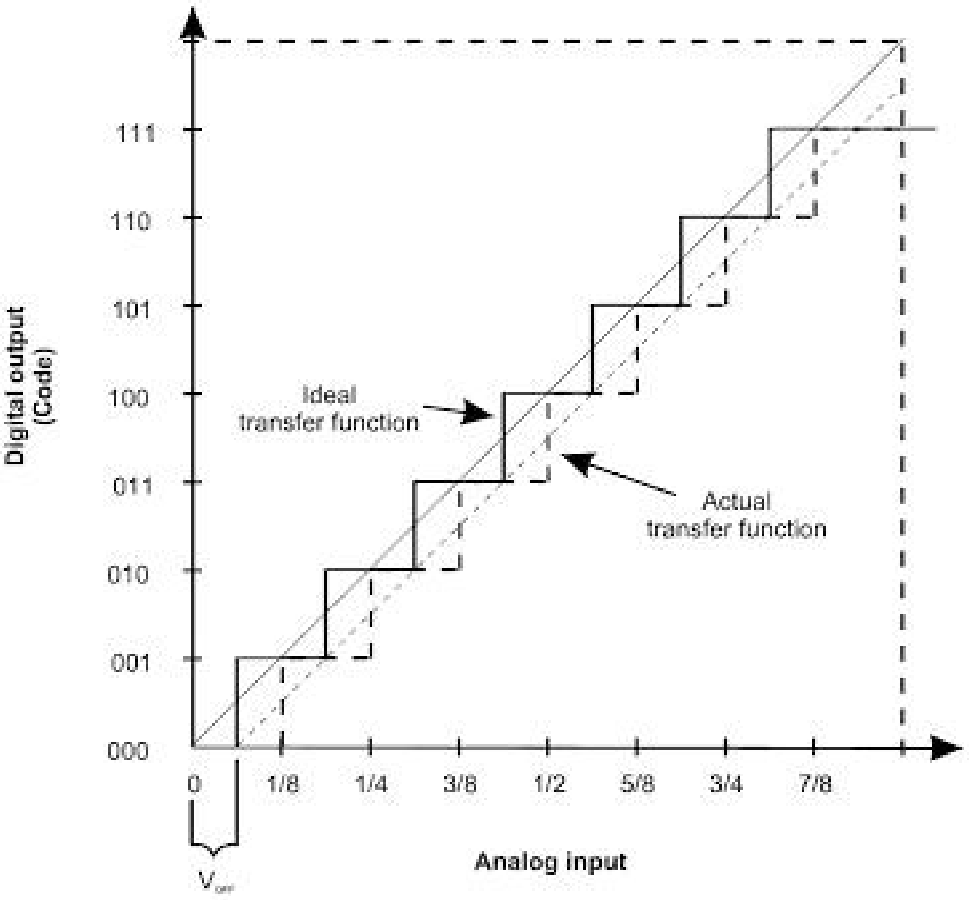
Unlike a D/A converter, where there exists a unique analog value for each digital code, each digital output code is valid over a range of analog input values. Analog inputs within a given discrete range are represented by the same digital output code, usually assigned to the nominal mid-range analog value. There is, therefore, an inherent quantization uncertainty of ± l/2 LSB (least-significant bit), in addition to any other actual conversion errors. This is shown in Figure 5 .6(b).

Unipolar offset

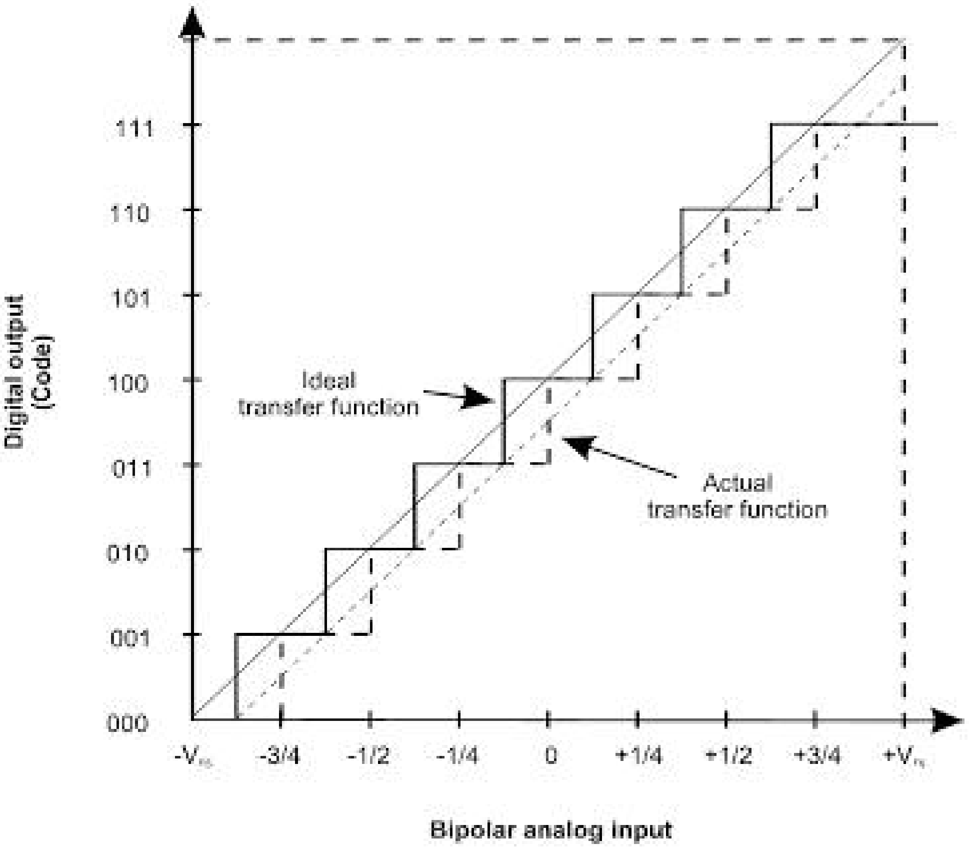
Note that in the ideal transfer function, the first transition should ideally occur 1/2 LSB above analog common. The unipolar offset is the deviation of the actual transition point from the ideal first transition point. This is shown in Figure 5.7(a).

Bipolar offset

As seen in Figure 5.7(b) the transfer function for an ideal bipolar ADC resembles the unipolar transfer function, except that it is offset by the negative full-scale voltage (–FSV). Offset adjustment of a bipolar A/D converter is set so that the first transition occurs at 1/2 LSB above –FSV, while the last transition occurs at –3/2 LSB below +FSV. Because of non-linearity, a device with perfectly calibrated end points may have an offset error at analog common. This is known as the bipolar offset error and is shown in Figure 5.7(b).



(a) Unipolar offset error

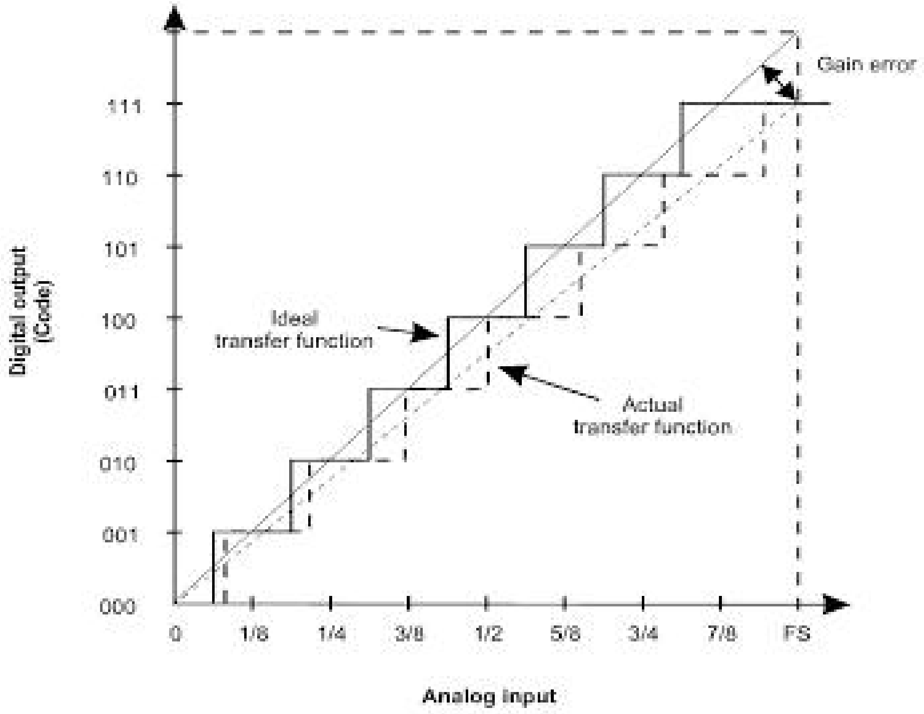


(b) Bipolar offset error

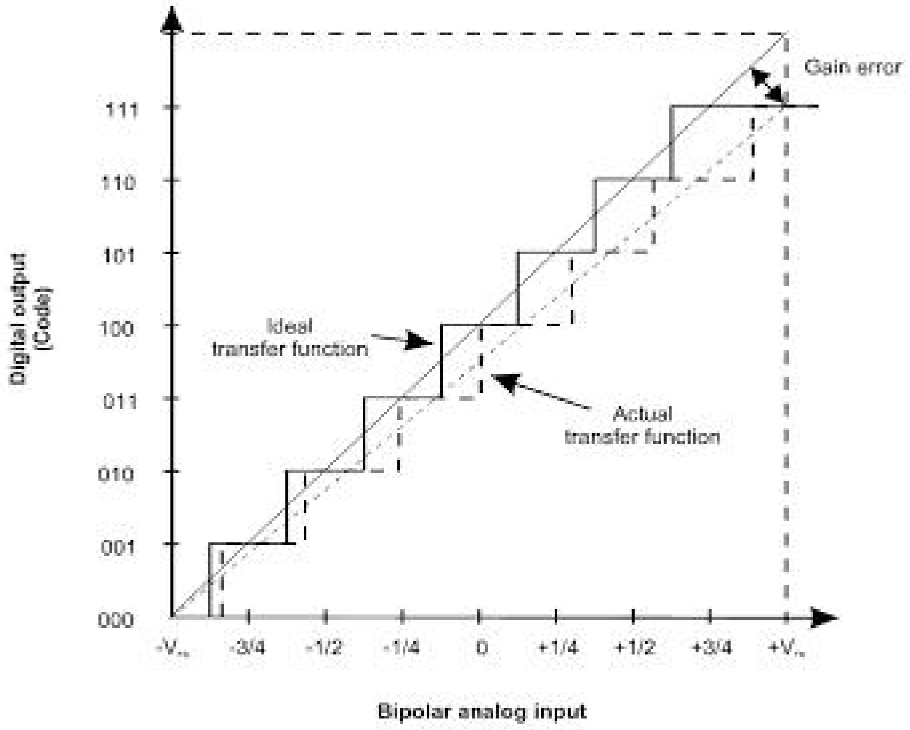
Figure 5.7
3-bit A/D converter transfer functions with offset errors

Unipolar and bipolar gain errors

The gain, or scale, factor is the number which establishes the basic conversion relationship between the analog input values and the digital output codes, e.g. 10 V full-scale. It represents the straight-line slope of the ideal transfer characteristic. The gain error is defined as the difference in full-scale values between the ideal and the actual transfer function when any offset errors are adjusted to zero. It is expressed as a percentage of the nominal full-scale value or in LSBs. Gain error affects each code in an equal ratio. Unipolar and bipolar gain errors are shown in Figure 5.8.



(a) Unipolar gain error



(b) Bipolar gain error

Figure 5.8
3-bit A/D converter transfer function with gain errors

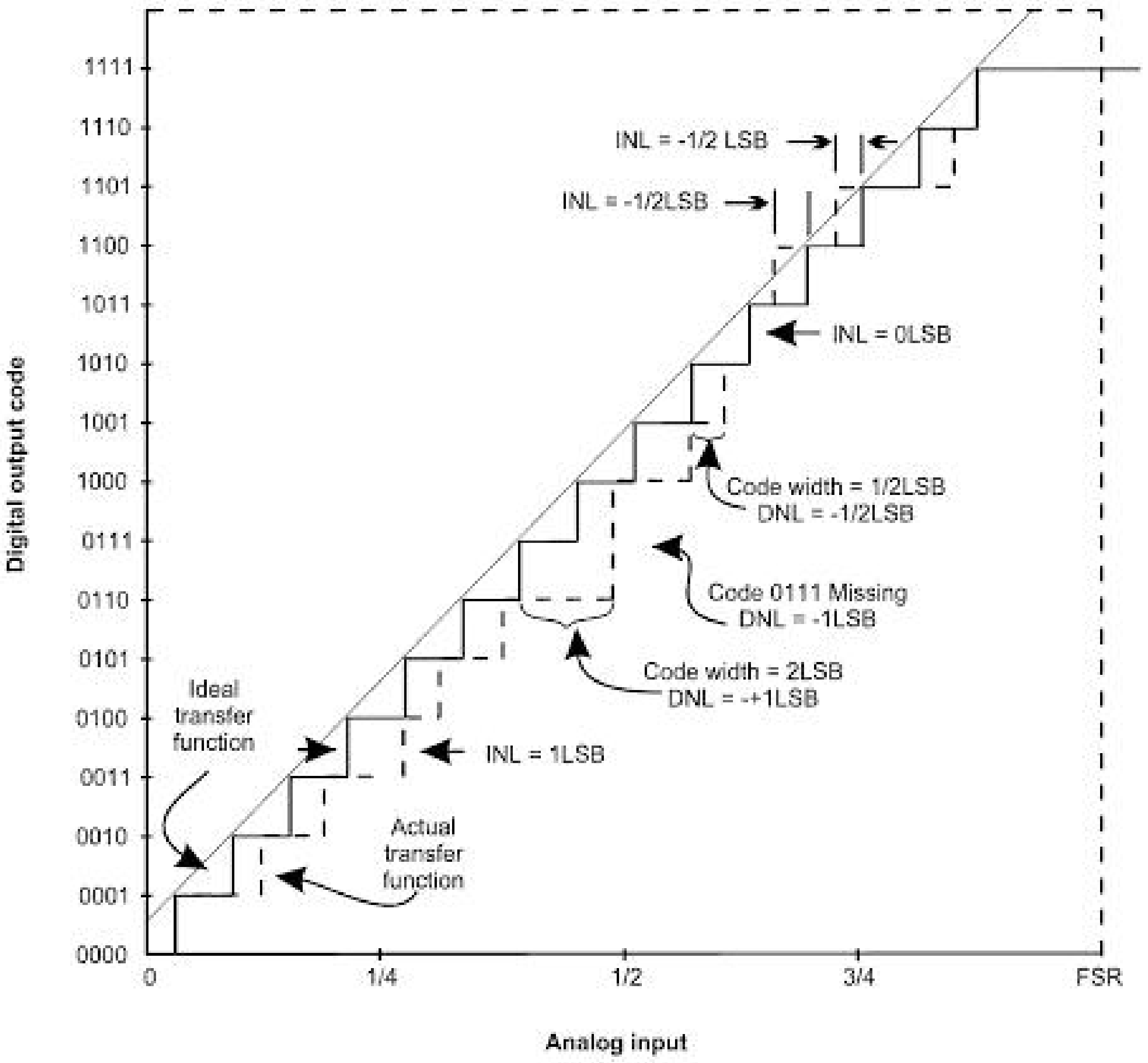
Offset and gain drift

Offset and gain errors are usually adjustable to zero with calibration, however this calibration is only valid at the temperature at which it was made.

Changes in temperature result in a non-zero offset and gain error, known as offset drift and gain drift. These values, specified in ppm/deg C, represent the ADC's sensitivity to temperature changes.

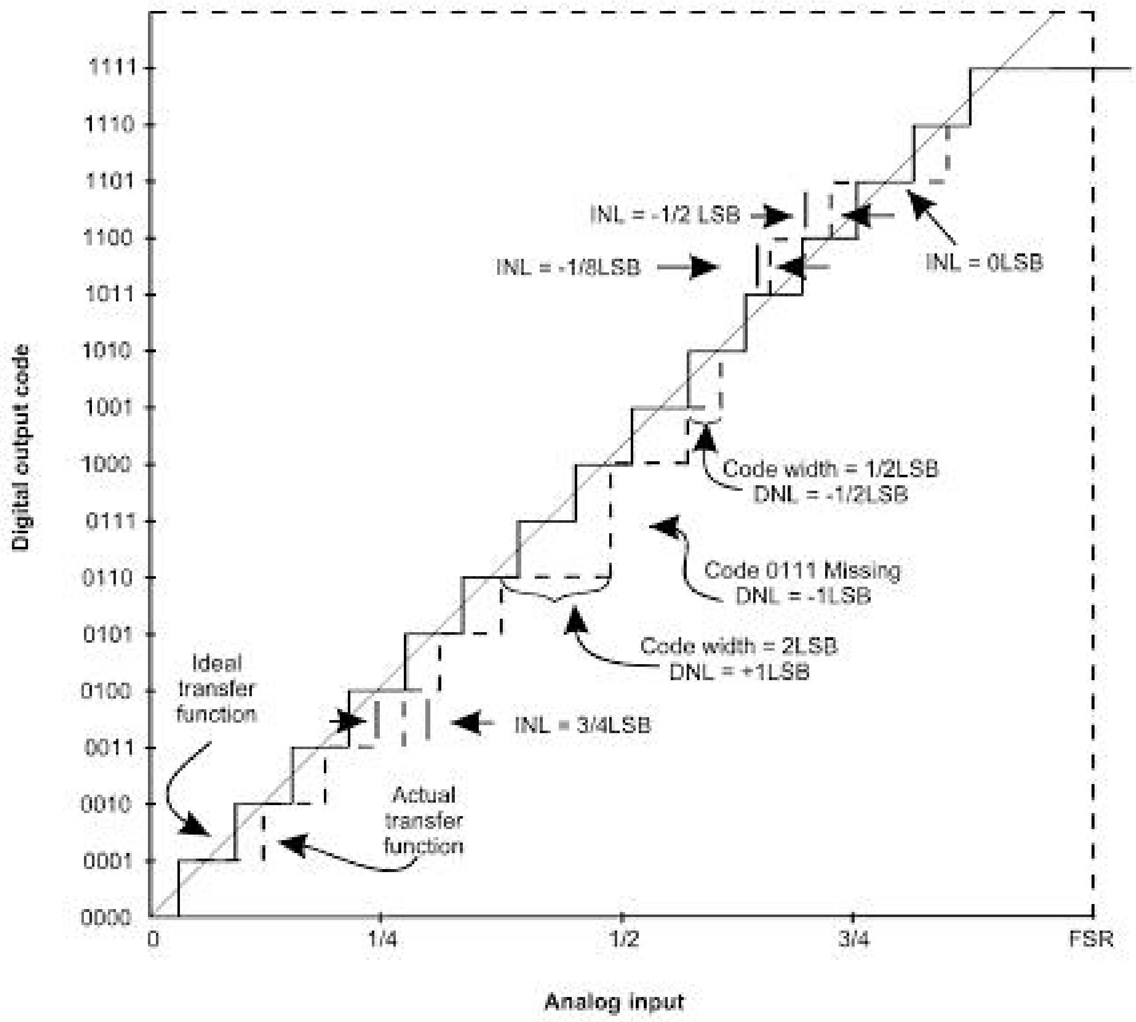
Linearity errors

With most ADCs the gain and offset specifications are not the most critical parameters that determine an A/D converter's usefulness for a particular application, since in most cases they can be calibrated out in software and/or hardware. The most important error specifications are those that are inherent in the device and cannot be eliminated. Ideally, as the analog input voltage of an A/D converter is increased, the digital codes at the output should also increase linearly. The ideal transfer function of the analog input voltage verses the digital output code would show a straight line. Deviations from the straight line are specified as non-linearities. The most important of these, (because they are errors which cannot be removed), are integral non-linearity and differential non-linearity errors. The transfer characteristics of a 4-bit A/D converter showing differential and integral linearity errors are shown in Figure 5.9(a) and Figure 5.9(b).



(a) Integral non-linearity errors specified as low-side transition

Figure 5.9(a)
Transfer function of a 4-bit A/D converter with integral non-linearity and differential non-linearity errors



(b) Integral non-linearity errors specified as center-of-code transition

Figure 5.9(b)
Transfer function of a 4-bit A/D converter with integral non-linearity and differential non-linearity errors

Integral non-linearity (INL)

This is the deviation of the actual transfer function from the ideal straight line. This ideal line may be drawn through the points where the codes begin to change (low-side transition or LST), as shown in Figure 5.9(a), or through the center of the ideal code widths (center-of-code or CC), as shown in Figure 5.9(b). Most A/D converters are specified by low-side-transition INL. Thus, the line is drawn from the point 1/2 LSB on the vertical axis at zero input to the point 3/2 LSB beyond the last transition at full-scale input. The deviation of any transition from its corresponding point on that straight line is the INL of the transition. In Figure 5.9(a), the transition to code 0100 is shifted to the right by 1 LSB, meaning that the LST of code 0100 has an INL of +l LSB. In the same figure the transition to code 1101 is shifted left by 1/2 LSB, meaning that the LST of code 1101 has an INL of –l/2 LSB.

When the ideal transfer function is drawn for center-of-code (CC) integral non-linearity specification, as shown in Figure 5.9(b), the INL of each transition may be different. Where the digital code 1101 previously had –1/2 LSB of LST INL, it now has 0 LSB of CC INL. Similarly, the code 1011 has –1/8 LSB of CC INL, where it previously had 0 LSB of LST INL.

The INL is an important figure because the accurate translation from the binary code to its equivalent voltage is then only a matter of scaling.

Differential non-linearity (DNL)

In an ideal A/D converter, the midpoints between code transitions should be 1 LSB apart. Differential non-linearity is defined as the deviation in code width from the ideal value of

A/D conversions are started by triggers; either by a software trigger (writing to an on-board register), or an external hardware trigger. Data conversions can be synchronized with external events by using external clock frequency sources and external triggers. The external trigger event is usually in the form of a digital or analog signal, and will begin the acquisition depending on the active edge if the trigger is a digital signal, or the level and slope, if the trigger is an analog signal.

In performing an analog-to-digital conversion cycle on a single input channel, the timing circuitry must ensure that the following steps are performed:

- Once the channel/gain array has been initialized, the timing circuitry increments to the next channel/gain pair. The next channel to be sampled is output to the address lines of the input multiplexer and the required gain setting is output to the programmable gain amplifier (PGA). The sample-and-hold (S/H) is put into sample mode.
- The timing circuitry must wait for the input multiplexer to settle, then for the PGA output delay time and lastly for any S/IA delay.
- The S/H is put into hold mode. The timing circuitry must wait for the duration of the aperture time of the S/H for the signal to become stable at the output of the S/H.
- A start conversion trigger is issued to the A/D converter.
- The timing circuitry waits for the end of conversion signal from the A/D converter to become active.
- The available data is then strobed from the A/D converter into a data buffer or a FIFO, from where it is usually accessible by the host computer.
- If simultaneous sampling is available on the A/D board, the timing circuitry generates the necessary sequence of strobes to the input S/H devices, so that all channels are sampled at the beginning of the sampling cycle, before the data is passed to the rest of the analog input circuitry.

Total throughput, for multiple conversions on different channels, is often increased by overlapping parts of this cycle. For example, while the A/D converter is busy converting the S/H output, the next channel/gain pair can be output to the multiplexer and PGA, so that their settling and delay times are overlapped with the A/D conversion time.

The timing circuitry may also include a block-sampling mode, which allows blocks of samples to be collected at regular intervals at the A/D board's maximum sampling rate. This is discussed in the section on *Sampling techniques,* p. 151.

5.2.8 Expansion bus interface

The bus interface provides the control circuitry and signals used to transfer data from the board to the PC's memory or for sending configuration information (e.g. channel/gain pairs) or other commands (e.g. software triggers) to the board.

It includes:

- The plug-in connector, which provides the hardware interface for connecting all control and data signals to the expansion bus, (e.g. ISA, EISA etc), of the host computer.
- The circuitry, which determines the base address of the board. This is usually a selectable DIP switch and defines the addresses of each memory and I/O location on the A/D board.

- The source and level of interrupt signals generated. Interrupt signals can be programmed to occur at the end of a single conversion or a DMA block. The configuration of the interrupt levels used is commonly selected by on-board links.
- DMA control signals and the configuration of the DMA level(s) used. The configuration of the DMA levels used is typically selected by on-board links.
- Normal I/O to and from I/O address-locations on the board.
- Wait state configuration for use in machines with high bus speeds or with non-standard timing. The number of wait states is usually configurable by on-board links.

5.3 Single ended vs differential signals

As previously demonstrated, great care must be taken in the connection, earthing and shielding of signals, received from external transducers (or similar signal sources), to signal conditioning equipment. This is especially true where the signal levels are very small and/or the signal sources are a long way from the measuring equipment. In these cases, the effects of earth loops, induced noise, and common mode voltages can introduce errors that lead to large inaccuracies in the signal measurement.

The three basic configurations for connecting input signals to signal conditioning equipment are available on plug-in A/D boards:

- Single ended
- Pseudo differential
- Differential

5.3.1 Single ended inputs

Single ended inputs are those where the signal is transmitted over a single conductor and referenced to analog ground AGND. The single conductor is connected to the HI terminal of the amplifier while the LO terminal of the amplifier is connected to AGND. This is shown in Figure 5.10.

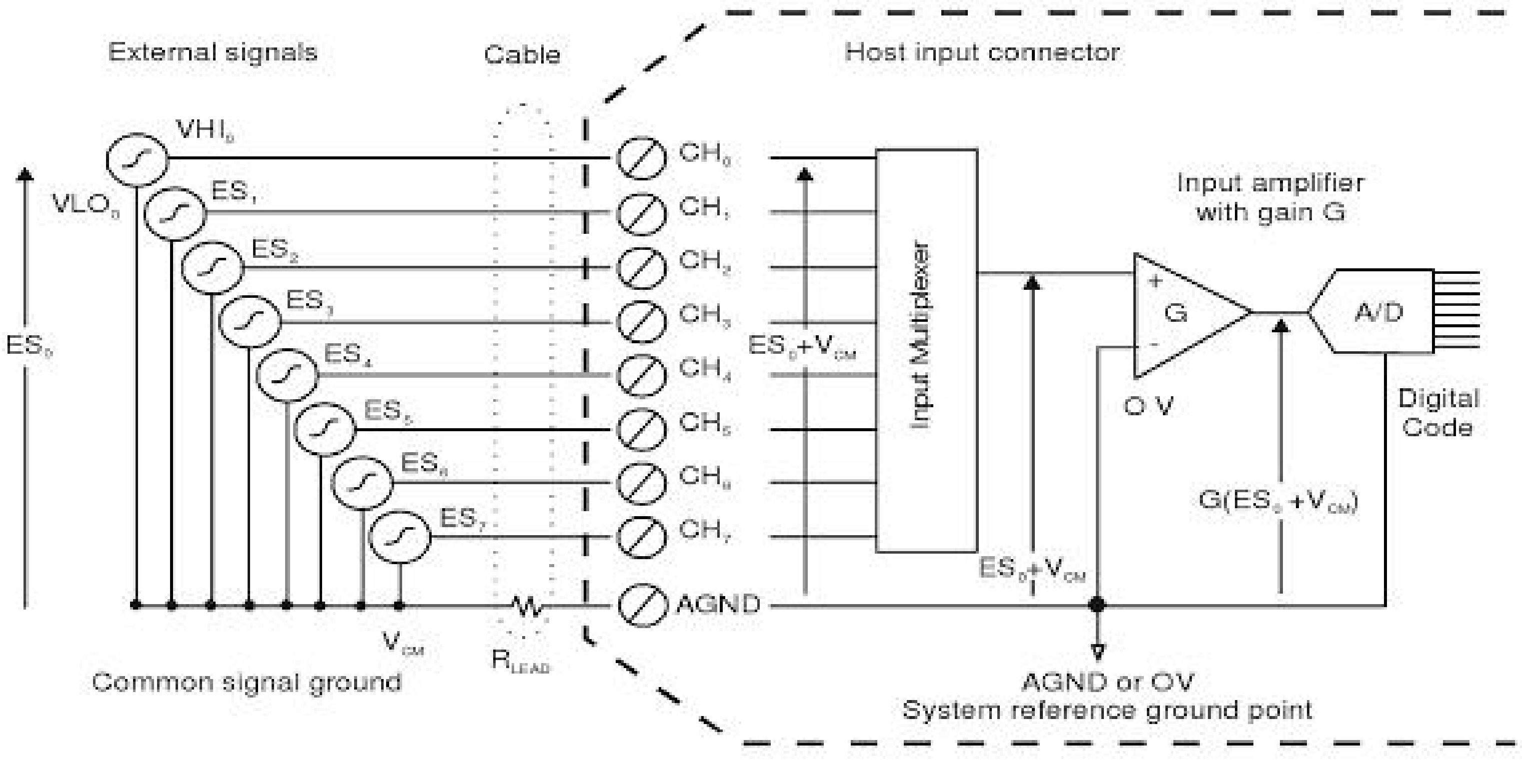


Figure 5.10
Single ended inputs

Single ended inputs usually carry high-level signals (in the order of volts), which do not require high gains (> × 5), transmitted over short distances (0.5 m). Where they are required to be transmitted over longer distances, they should be shielded, and the shield connected to AGND at the instrument end only.

While this configuration allows more inputs to be multiplexed to a single A/D converter, it should only be used where there is no practical way of bringing a remote ground or an analog ground back to the measurement point.

Because the amplifier LO terminal is connected to AGND, what is amplified is the difference between $E_{sn} + V_{cm}$ and AGND. This introduces the common mode offset voltage as an error.

Plug-in boards that do not have an amplifier (i.e. where the multiplexed input is fed straight to the A/D converter) must use the single ended input configuration.

5.3.2 Pseudo-differential configuration

The pseudo-differential input configuration is a variation of the single ended input configuration, providing some degree of common mode rejection while still allowing the maximum number of multiplexed input channels.

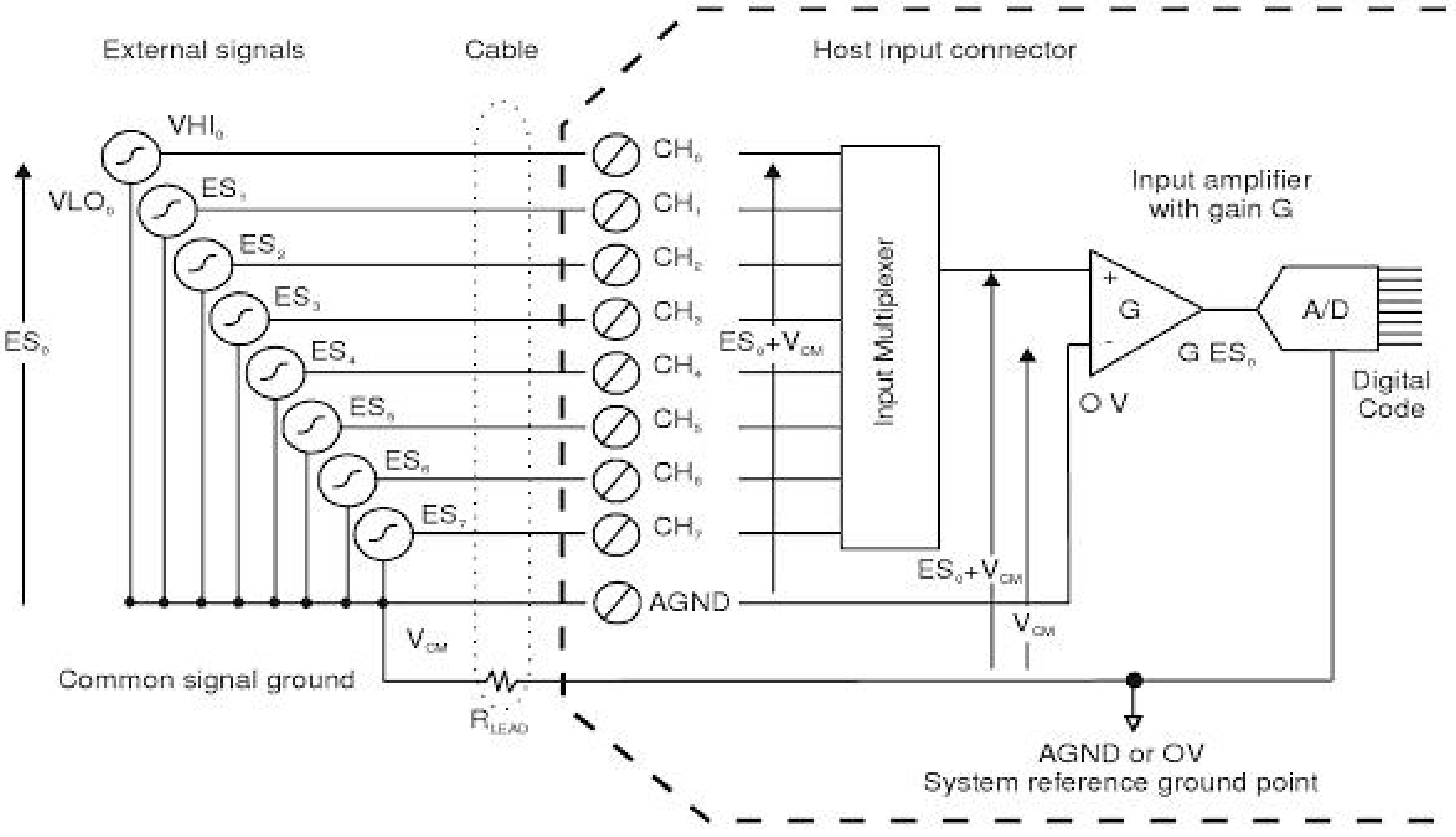


Figure 5.11
Pseudo differential inputs

In the configuration shown in Figure 5.11 the signal source LO outputs are all connected to the LO input terminal of the amplifier, while the signal source HI outputs are connected to the respective HI input for each of the channels. The LO input to the amplifier is then referenced to analog ground AGND at the signal end of the cable. This method is only possible if the LO terminal of the amplifier is brought out to the connector and the signal sources can be grounded at their signal ends. In using this configuration only the difference between the channel input $E_{sn} + V_{cm}$ and the signal ground, which has the common mode voltage on it, is amplified.

5.3.3 Differential inputs

True differential inputs, where the HI and LO outputs of the signal source are connected directly to the HI and LO terminals of the amplifier, as shown in Figure 5.12, offer the greatest noise immunity and common mode rejection.

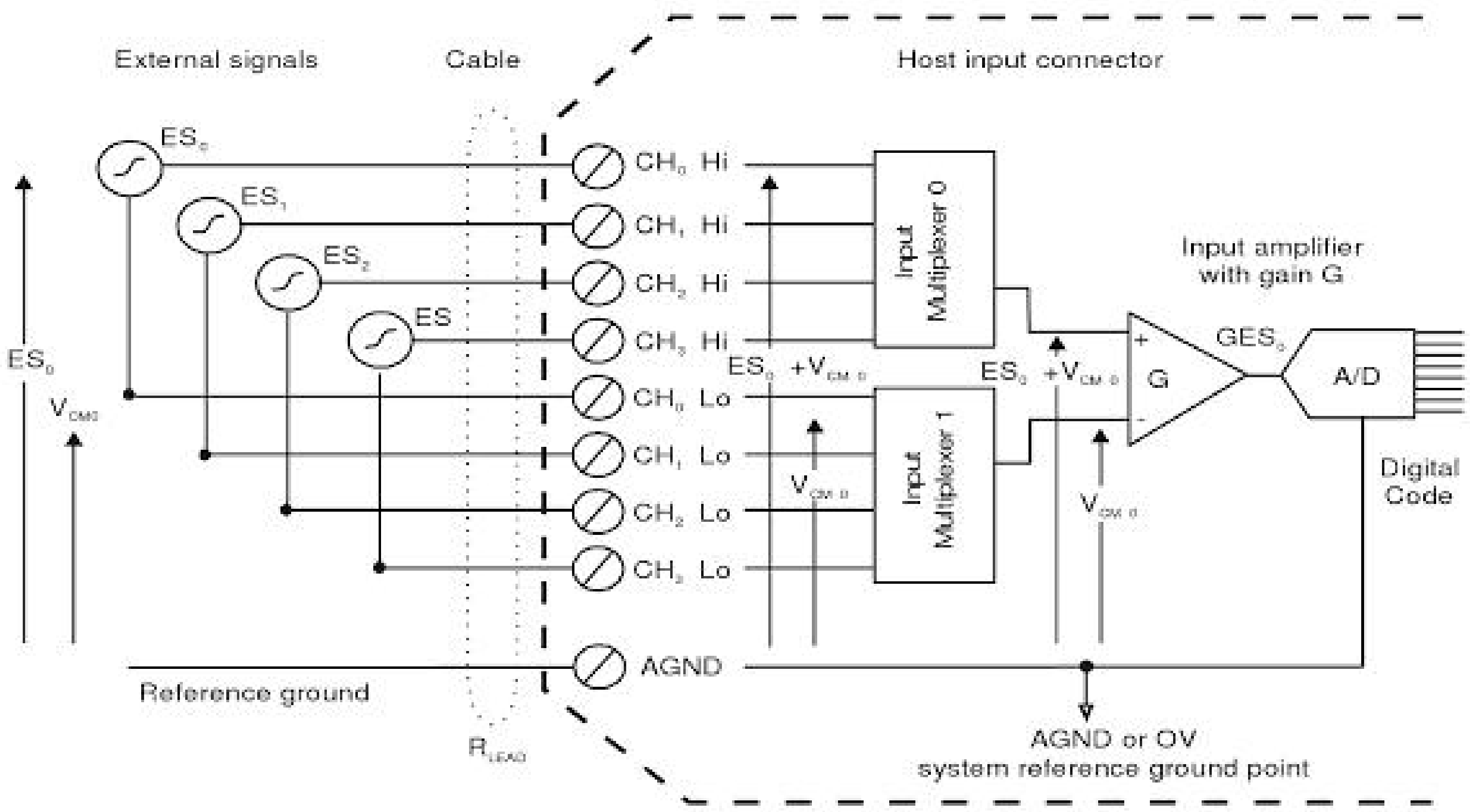


Figure 5.12
Differential input configuration

In this configuration, only the difference in the signal inputs is measured. Noise induced equally in each signal line will cancel out at the inputs of a true balanced differential amplifier, while common mode voltages (appearing at both inputs) will be rejected if the amplifier has a large CMRR. This should not preclude the added precautions of twisting differential pairs and providing earthed shields to reduce noise induced in long cables.

Differential inputs should be used:

- When measuring signals with large common mode voltages (e.g. strain gauges).
- Where several transducers with different ground points (and possibly different ground potentials) are to be measured. Connecting the Lo channel of each sensor together at a common point, as in the pseudo differential connection, can create unwanted ground currents that induce offset and noise errors at the amplifier inputs.
- When measuring signal voltages that are very small, and the signal/noise ratio is low.
- When the input transducer is physically located a large distance from the measuring device and may be susceptible to the effects of noise.

Note that for differential input configuration, two input multiplexers are needed, and for the same number of input terminals as single ended and pseudo differential inputs, half the number of input channels is available in differential mode.

Where high impedance sources are used, bias resistors may be required to return bias currents to the operational amplifier, thus preventing the floating of inputs beyond the limits of the amplifier inputs. Such bias resistors normally consist of high impedance resistors, typically between 100 KΩ and 1 MΩ, connected between the HI and LO signal lines and AGND.

5.4 Resolutions, dynamic range, and accuracy of A/D boards

5.4.1 Dynamic range

One of several considerations in determining the analog input requirements of an A/D board is the range of voltages, which each channel is required to measure. The physical parameters to be measured, the type of sensor(s) used and how they are connected, determine the input voltage ranges required.

The input range specifications quoted by board manufacturers of A/D boards refer to the minimum and maximum voltage levels that the A/D converter on the board can quantize. Typically, a selection of input ranges is provided, either unipolar (e.g. 0 to 10 V), for measuring positive voltages only, or bipolar (e.g. –10 V to +10 V), for measuring both positive and negative voltages. This allows the user to match the input signal range to that of the A/D converter, taking into account the resolution of the A/D converter and the gain required of the input amplifier.

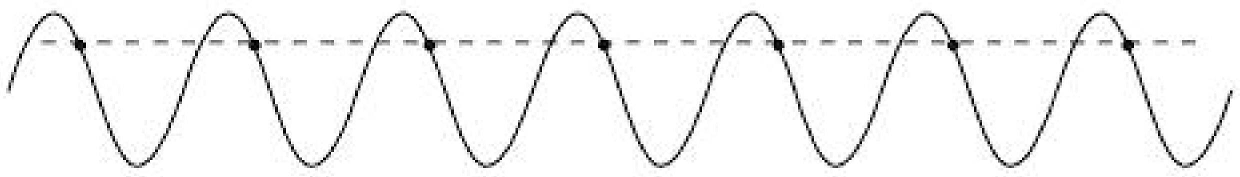
When considering the input range, it is only the dynamic range of the input signal that needs to be taken into account. For example, consider a strain gauge setup in a Wheatstone bridge configuration. The input voltage to be read has a common mode component due to the excitation of the bridge, while the small differential voltage changes, (of interest) are due to the change in strain gauge resistance. The common mode voltages do not provide any useful information and are greatly attenuated, (almost eliminated), by using differential inputs and instrument amplifiers with high CMRR. Only the small differential voltage changes are amplified and converted by the A/D converter. The amplifier gain should therefore be selected so that the maximum differential voltage change expected at the input will be amplified to cover as much of the input range of the A/D converter as possible.

As only one of the allowable range settings can be selected at any time, typically by jumpers on the board itself, care should be taken in matching the input signal requirements where more than one channel is sampled. The A/D converter input range selected must accurately measure the signal inputs from a number of channels, possibly different sensors, and therefore potentially different input voltage levels and signal ranges. The input range should therefore cover each channel's input range with as little overlap as possible, thus giving the greatest number of data points and therefore the highest resolution and accuracy.

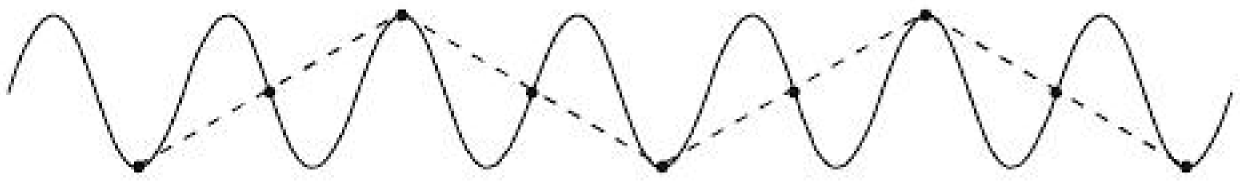
It should be noted that the input ranges specified do not necessarily refer to the maximum or minimum voltage levels that can be applied at any single input, or to the maximum allowable common mode voltage, which can be applied, to a differential input. These are specifications, more related to the input amplifier. If there are any doubts with regard to this, users should consult the board manufacturer.

5.4.2 Resolution

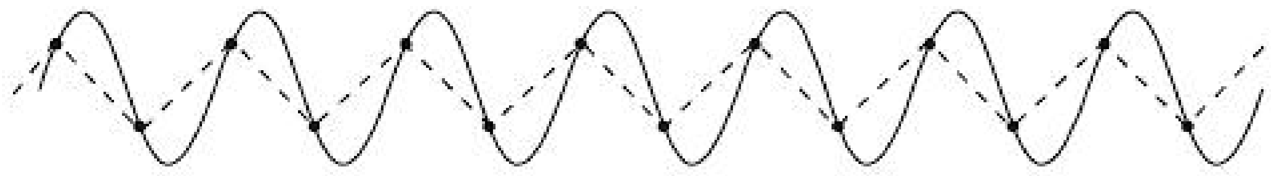
The resolution specification quoted by manufacturers of A/D boards refers to the resolution of the A/D converter used on the board. It is usually expressed by the number of bits the A/D converter uses to represent the analog input voltage (i.e. n-bit) or as a fraction of the maximum number of discrete levels, which can be used to represent the analog signal (i.e. $1/2^n$). The resolution implicitly defines the number of discrete ranges into which the full-scale voltage (FSV) input range can be divided to approximate an analog input voltage. A 12-bit



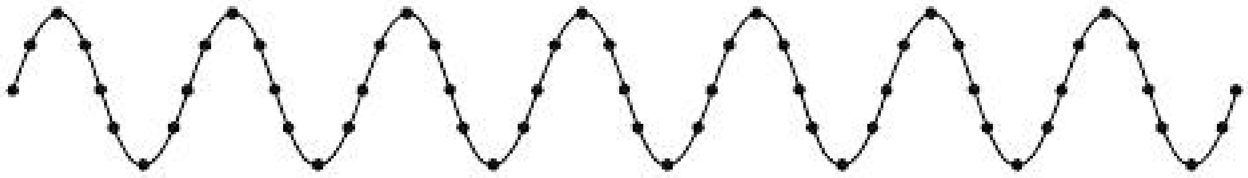
a) DC alias caused by sampling at half the Nyquist sampling rate



b) Aliasing caused by sampling at less than the Nyquist sampling rate



c)Sampling at the Nyquist sampling rate



d) Oversampled signal adequately reconstructed

Figure 5.13
Effect of sampling rate on the reconstructed input signal

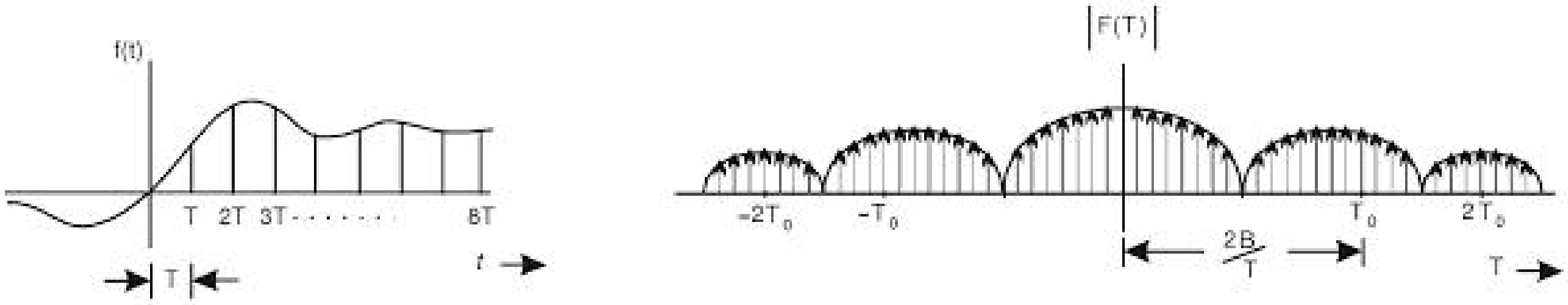
Figure 5.13 (d) shows a signal that is sampled at a frequency well above the Nyquist sampling rate. In this case, the information contained in the signal, including its shape and frequency, can be correctly reproduced. If the sampling rate is reduced to below the Nyquist sampling rate, that is, the sample points are too far apart, then the input signal is misrepresented by what appears to be a much lower frequency signal. This phenomenon is known as aliasing and is demonstrated in Figure 5.13 (b).

In Figure 5.13(a), the input signal is sampled at half the Nyquist sampling rate, which is the same frequency as the frequency of the signal itself. The reconstructed waveform appears as a DC signal. When the input signal is sampled at the Nyquist sampling rate, as shown in Figure 5.13(c), the reconstructed signal has the correct frequency but incorrectly appears as a triangular waveform. Where undersampling occurs, the frequency of the reconstructed signal appears to be much lower, lying between DC and the Nyquist frequency.

Theoretically, the effects of aliasing are more easily understood by looking at the frequency spectrum of an analog signal. Without detailing the complex mathematical descriptions and frequency analysis required, it can be shown that a time varying band-limited signal can be equally represented by its spectrum in the frequency domain. Figure 5.14(b) shows the frequency spectrum of the band-limited signal shown in Figure 5.14(a). If the time varying signal is sampled using a very narrow series of square wave pulses, as shown in Figure 5.14(c), then the frequency spectrum of the sampled waveform is the original signal with exact replicas of itself spaced about multiples of the sampling frequency. Figure 5.14(d) illustrates the frequency spectrum of a signal that is sampled at exactly twice the maximum frequency of the original signal, showing that the replicas of the original signal just touch. Oversampling the original signal, as shown in Figure 5.14(e) separates the input signal bands by a wider frequency. This is shown in Figure 5.14(f). Undersampling narrows the separation between the bands so that they fold over each other and result in aliasing, as demonstrated in Figure 5.14(g) and Figure 5.14(h). Where this occurs, the resultant signal appears as an aliased signal between DC and the Nyquist frequency, and cannot be distinguished from valid data.

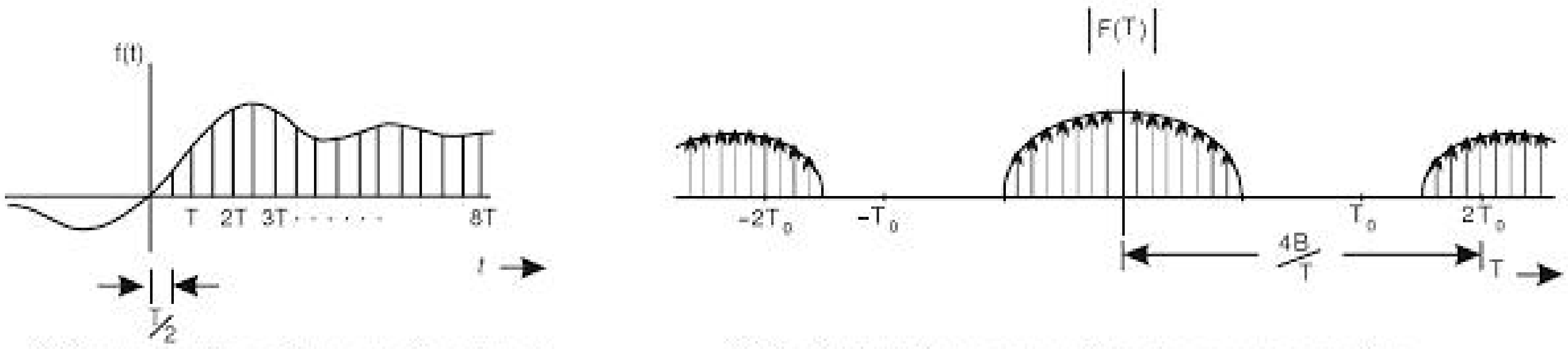
a) Time-varying band limited signal

b) Frequency Spectrum



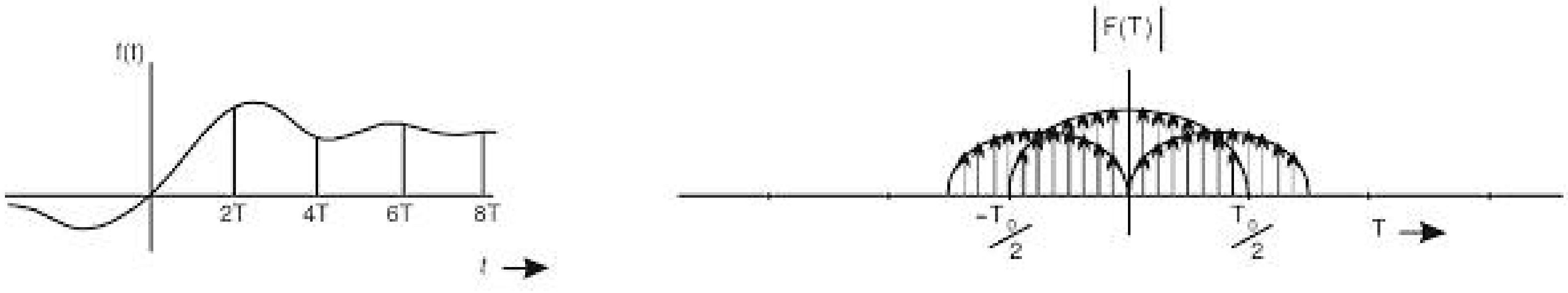
c) Signal sampled at Nyquist rate

d) Spectral replicas generated when sampling at Nyquist rate



e) Over sampling a time varying signal

f) Spectral replicas generated when over sampling

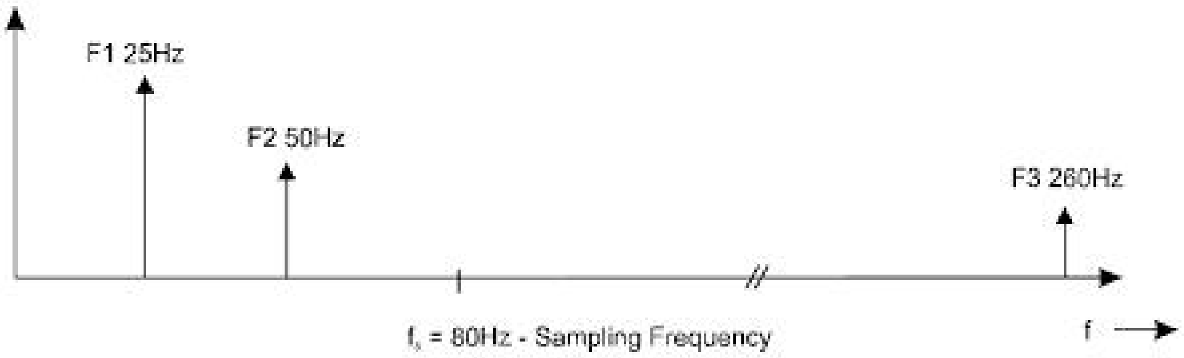


g) Under sampling a time varying signal

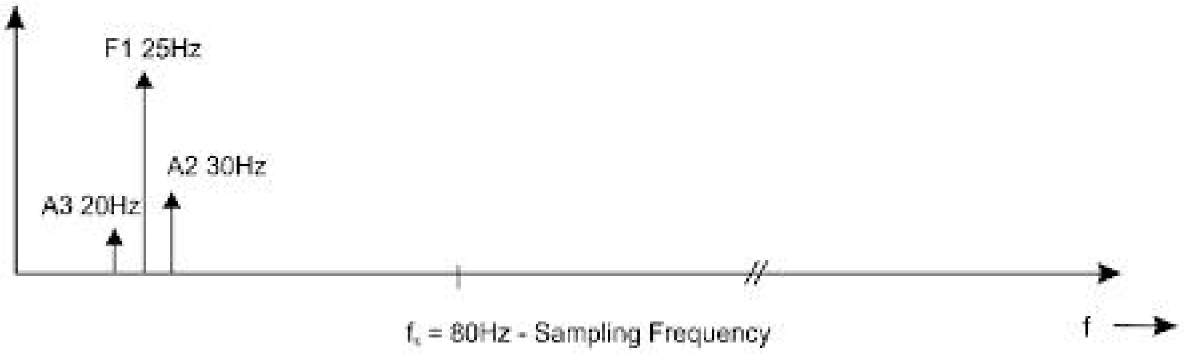
h) Spectral replicas generated when under sampling

Figure 5.14
Demonstrating the effect of aliasing in the frequency domain

Consider a band-limited signal, which contains three sinusoidal waveforms, a 25 Hz waveform representing the wanted signal, a 50 Hz signal, which is unwanted mains hum, and an unwanted high frequency noise signal at 260 Hz. Figure 5.15(a) shows the frequency spectrum of this band-limited signal.



a) Frequency spectrum of original signal



b) Frequency spectrum of original and sampled signals

Figure 5.15
Frequency spectrum of original and sampled signals

The frequency spectrum of the reconstructed signal, sampled by an A/D board at $f_s = 80$ Hz is shown in Figure 5.15(b). Frequencies below the Nyquist frequency, $f_s/2 = 40$ Hz, in the original signal spectrum, appear correctly. However, replicas of the signal frequencies above the Nyquist frequency are reproduced about multiples of the sampling frequency and therefore appear as aliases. A2 and A3 are aliases of the original signals F2 and F3 respectively. The alias frequency of any signal frequency can be simply calculated by the formula:

Alias Freq = ABS (closest integer multiple of sampling frequency – signal frequency)
Alias A2 = [80 – 50] = 30 Hz
Alias A3 = [(3)80 – 260] = 20 Hz

In this example, the resulting aliases are very close to the frequency of the signal of interest and would be very difficult to remove. Once an aliased signal has been introduced, it is almost impossible to remove it by digital filtering methods.

5.5.3 Preventing aliasing

One method of preventing aliasing is by filtering the input signal with a low pass filter with a cutoff point set to the Nyquist frequency or half the sampling rate. This type of filter is known as an *antialiasing filter*. A perfect antialiasing filter would simulate the *brick-wall* response of an ideal low pass filter, as shown in Figure 5.16, rejecting all unwanted frequency components above the Nyquist frequency. Thus, by using this filter the input signal could be sampled at twice the Nyquist rate without aliasing.

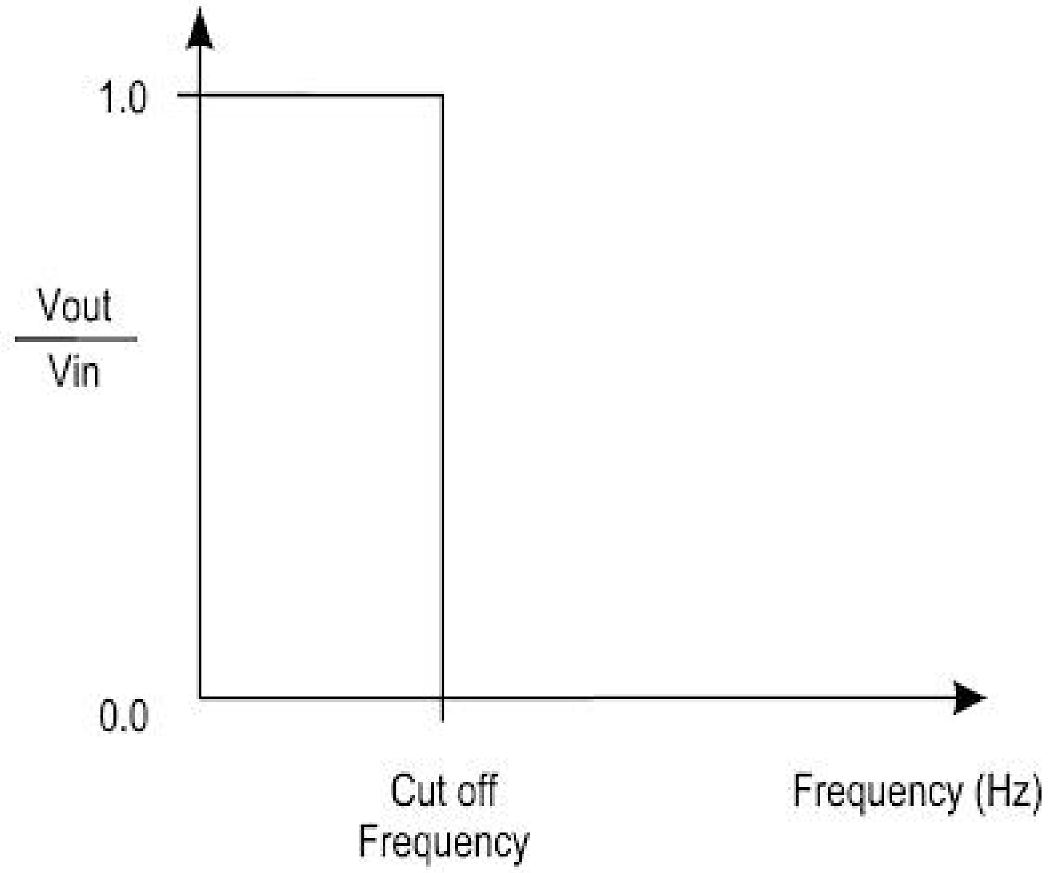


Figure 5.16
Ideal low pass filter response

Unfortunately, real filters do not simulate ideal filters, and in fact exhibit some attenuation (dB/octave) near the cutoff frequency. As shown in Figure 5.17, this roll-off may not be steep enough to totally eliminate all the higher frequency components. Although attenuated, these higher frequency components can, and will, fold down to the signal band of interest.

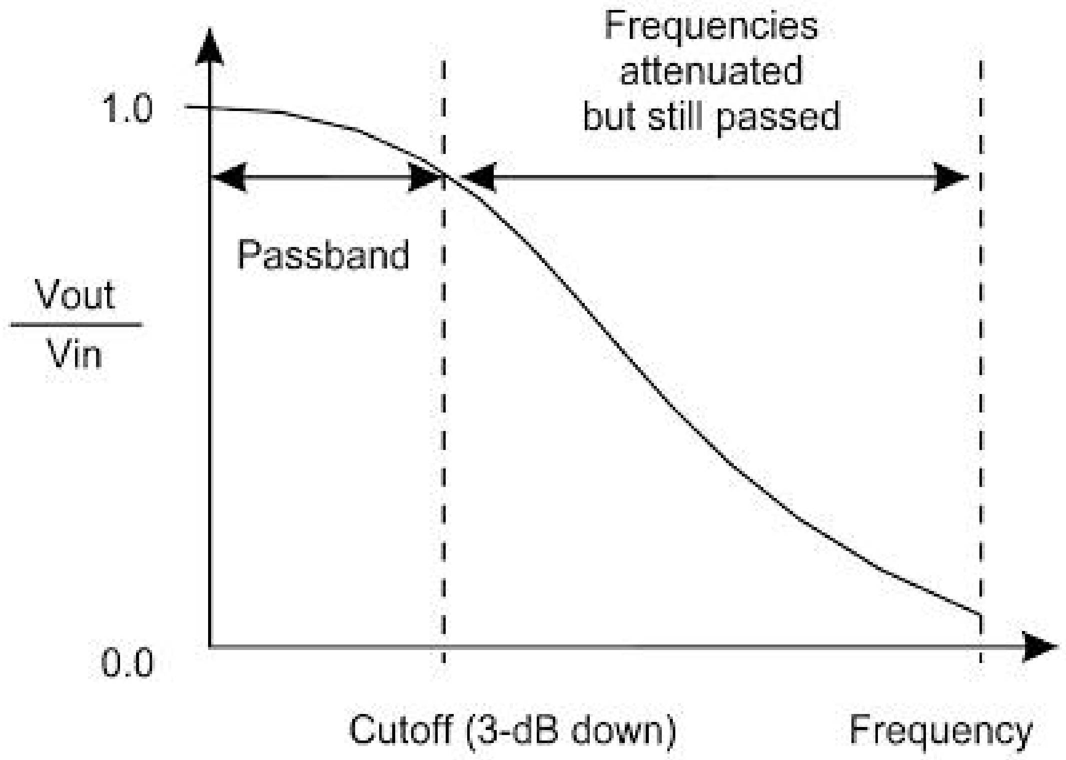


Figure 5.17
Practical low pass filter response

Therefore, to accommodate the filter cutoff frequency and roll-off, the sampling rate should be increased. Using simple passive antialiasing filters, it is recommended that the sampling rate be a minimum of about five times the cutoff frequency. Non-periodic wave-forms can be oversampled by about ten times.

High performance antialiasing filters with very steep roll off near the cutoff frequency, as shown in Figure 5.18, allow the signal to be sampled at two to three times the filter cutoff frequency.

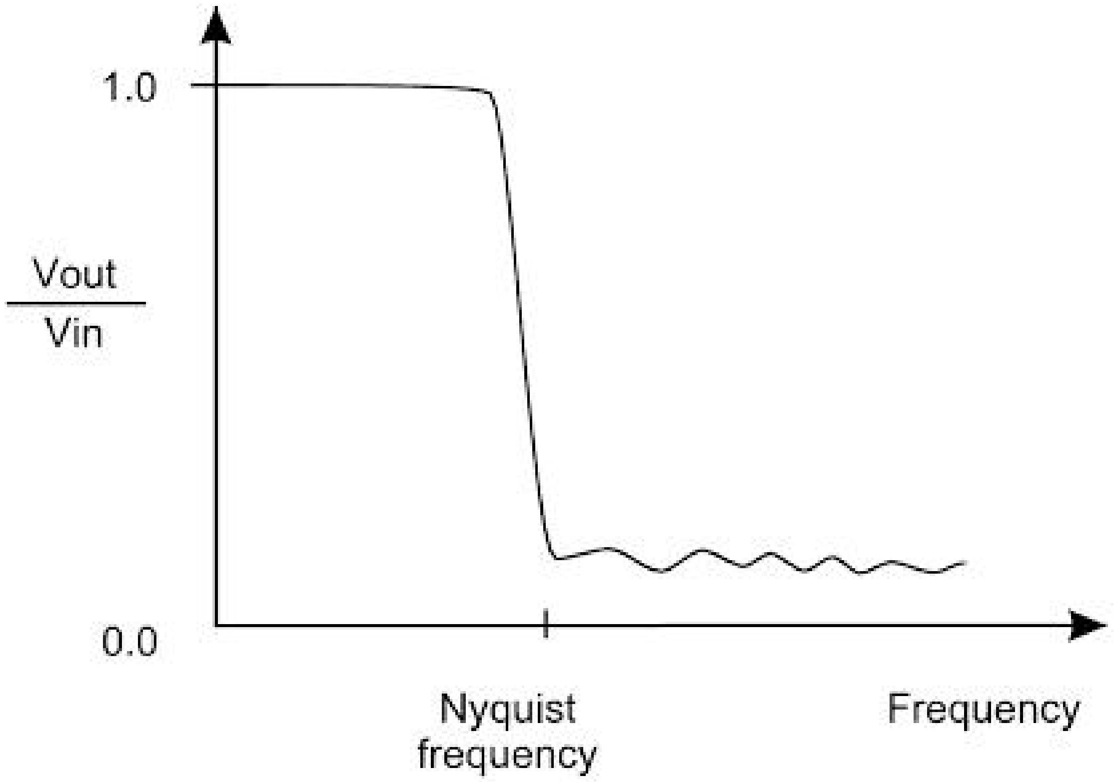


Figure 5.18
Steep roll-off antialiasing filter

5.5.4 Practical examples

A common data acquisition application is machine vibration analysis. All machines resonate at certain frequencies, both under normal operation and when driven by an external source. In this example, strain gauges were placed on the machine and the output signal sampled, digitized (yielding a time domain plot, see Figure 5.19(a) and converted into the frequency domain (for example, using FFT).

The spectrum resulting from sampling at 50 kHz is shown in Figure 5.19(b). It has two resonant frequency peaks, one around 4 kHz, and another slightly above 5 kHz. The machine vibration analyst knows that the 4 kHz component corresponds to the machine's rotational speed, but the 5 kHz component is a mystery. Passing the input signal through a 10 kHz cutoff antialiasing filter with subsequent resampling, yields the spectrum in Figure 5.19(c), clearly revealing the 5 kHz component to be an alias. Indeed, sampling the original signal (without the antialiasing filter) at 100 kHz yields the spectrum in Figure 5.19(d) and shows that an actual frequency component, present in the vibration signal, of 45 kHz has been aliased down to 5 kHz when sampled at 50 kHz.

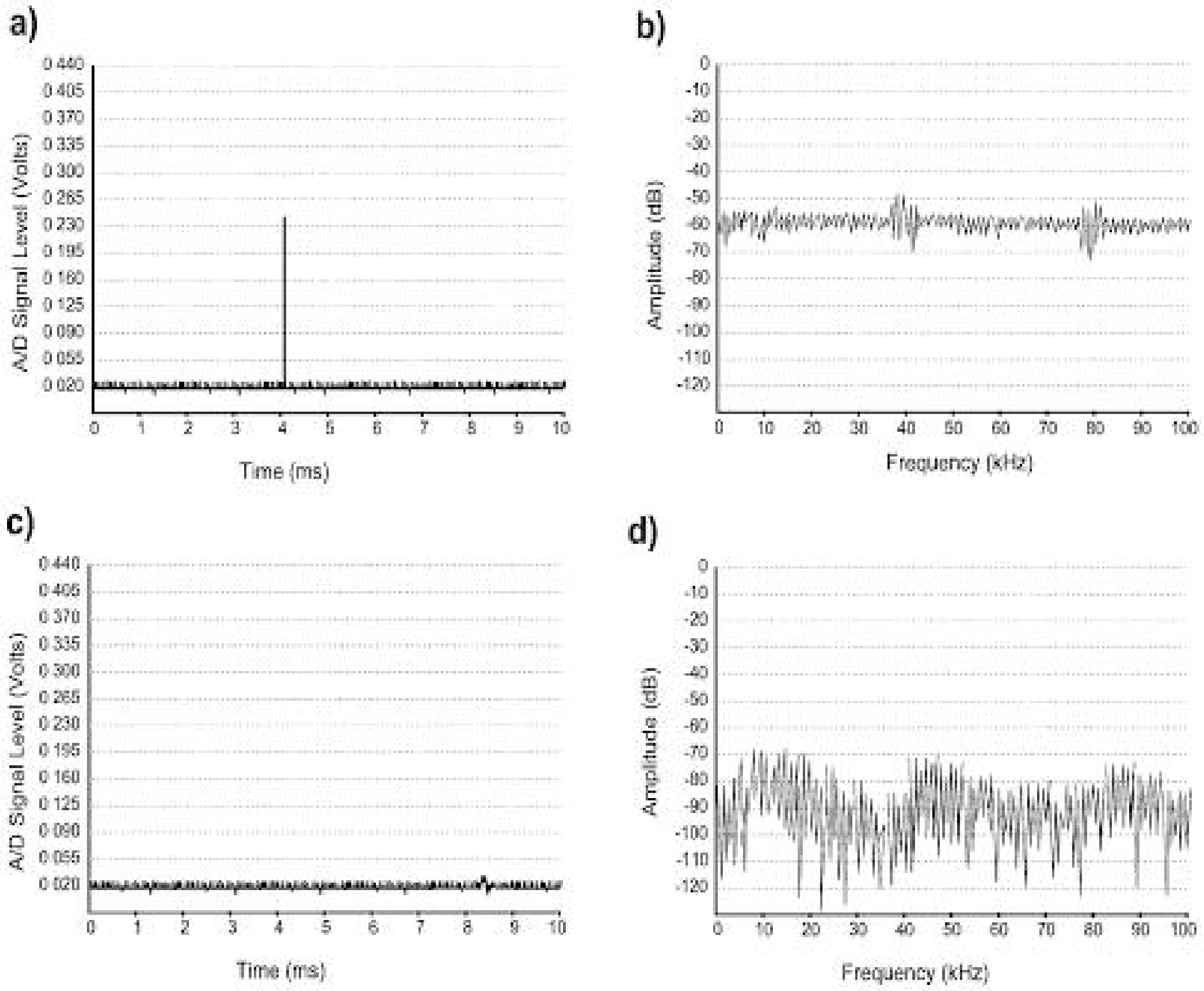


Figure 5.21
A spike causes wideband aliasing

5.6 Sampling techniques

These techniques are discussed in the following sections:

- Continuous channel scanning
- Simultaneous sampling
- Block mode operations

5.6.1 Continuous channel scanning

The method of sampling that facilitates the connecting of the required input channel to the A/D converter at a constant rate is known as continuous channel scanning. Continuous channel scanning allows channels to be sampled in a pre-determined and arbitrary order (e.g. channel 5, channel 1, channel 11), as well as at different sampling rates. An example of this would be the sampling of three channels in the following order (channel 5, channel 1, channel 11, channel 1). Channel 1 is being sampled at twice the rate as channels 5 & 11, which for an A/D board with throughput of 100 kHz represents a sampling rate of 50 kHz. Channels 5 & 11 are sampled at 25 kHz. There are two methods of continuous channel scanning, either under software control or by on-board hardware control.

Software channel scanning

Where continuous channel scanning is performed by software the address of the channel to be sampled is written to the multiplexer and the gain setting sent to the programmable gain amplifier (PGA), where one is fitted. Once the signal is settled, an A/D conversion is initiated. The data is subsequently read and transferred to the PC's memory. This incurs a large software overhead. Background operation using interrupts is difficult and slower than polled I/O and accurately timed samples and higher speed data transfer methods such as DMA and repeat instructions are impossible in either case.

Hardware channel scanning

Continuous channel scanning is implemented in hardware using channel-gain arrays (CGA). These programmable memory buffers contain a list of the channels and the gain settings required for each input channel to be sampled. When the A/D board begins sampling, input channels are sampled in the sequence loaded into the channel-gain array.

The use of on-board channel-gain arrays (CGA) overcomes many of the limitations associated with channel scanning using software and has the following advantages:

- The channel sequence information may be setup once and then sampling initiated (and repeated) with a single command. Once initiated, the sampling process is controlled by the A/D board's hardware.
- Arbitrary sample sequences may be defined.
- Within the limitations on the size of the CGA, different sampling frequencies may be specified for different channels.
- The speed of software-transfer methods such as interrupt and polled I/O is greatly increased, in many cases doubled. This is due to the fact that delays caused by the host computer transferring channel and gain information before each sample is taken, are avoided.
- Very accurate timing is achievable since the board hardware is optimized to control the individual sub-systems on the board.
- Advanced transfer methods such as DMA and repeat instructions are possible. DMA transfer is controlled directly by the hardware on the A/D board and the host computer. This is not a very flexible arrangement, since it does not allow intervention by software to change the channels being scanned once a DMA transfer has been initiated. A/D boards, which are capable of DMA but do not have channel-gain arrays may only perform DMA transfers from a single input channel, whose address and required gain is setup by software before the DMA transfer is initiated. Where channel-gain arrays are implemented, the on-board hardware will automatically change the address and gain settings during the DMA transfer. Where repeat instructions are used to transfer information, usually from an on-board FIFO, the sampling of multiple channels must continue to be performed in the background. On A/D boards, which do not have channel-gain arrays, repeat instruction transfers may only be performed on a single input channel whose address and required gain is setup by software before the repeat instruction transfer is initiated.

Practical applications

Some of the practical applications, which utilize the flexibility in the selection and throughput of individual channels, using hardware channel scanning, are detailed below.

Sampling different channels at different frequencies

When signals with different frequencies are sampled (for example, a heart rate electrocardiogram (ECG) with 300 beats/min and an electroencephalogram (EEG) with a frequency of 5 kHz), it is much more memory-efficient to sample each channel at around its

When operating in continuous scanning mode, conventional A/D conversion triggering works as follows: The sample trigger source, either from software, an on-board pacer clock, or an external clock, is programmed for a specified sample rate. Each sample trigger initiates a single A/D conversion on the next channel in the channel/gain array and every sample is evenly spaced in time.

Block mode triggering initiates an A/D conversion on all the required input channels at the maximum sampling rate of the A/D board, every time a sample trigger pulse occurs. A second counter is used to trigger the sampling of each of the channels at the maximum sampling rate. The number of samples to be taken in each block is typically stored by software in an on-board buffer, while the channel and gain for each sample in the block is read from the channel/gain array. The scan sequence is repeated at the next sample trigger pulse.

Consider an example where four channels are being sampled at a total throughput rate of 20 kHz, corresponding to a channel scan rate of 5 kHz. Figure 5.23 shows that in continuous scanning mode, the total scan time is 200 μs, with the samples evenly spaced every 50 μs. In block trigger mode, the four samples are taken in a single scan sequence at the maximum throughput of the board. Assuming the board is capable of taking samples at 200 kHz, the time between each of these four samples is 5 μs, while the total time taken for all the samples is 20 μs instead of 200 μs.

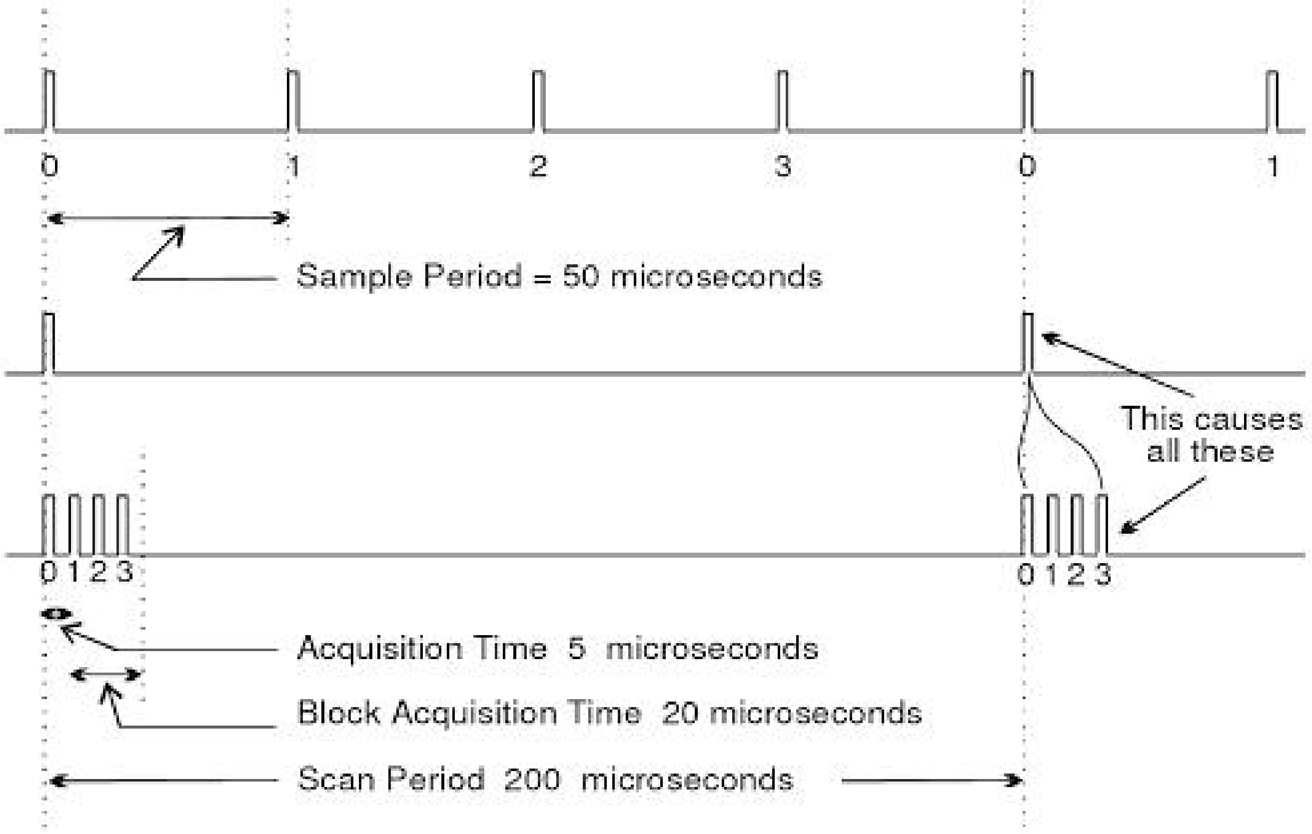


Figure 5.23
Conventional and burst trigger scanning

Where the sampling rate remains the same, that is, a sample trigger occurs every 50 μs, the throughput of the board is increased by the number of samples taken in each sample block. In this case, the throughput would be increased to 80 kHz. To maintain the total throughput rate at 20 kHz the sampling rate must be reduced by the number of channels sampled in each block. This is called the burst trigger rate and can be calculated by dividing the throughput required by the number of channels to be sampled.

$$\text{Burst trigger rate} = \frac{\text{Required total throughput}}{\text{No channels}}$$

For each burst trigger, the A/D board generates the required number of A/D conversion triggers at the maximum speed of the board. Even though the samples in a block (except the first sample) are taken at different times to the conventional triggered samples, the throughput of each channel and the time between samples on the same channel remains the same.

When using block mode triggering, data appears at the output of the A/D converter at the maximum throughput of the board. Therefore, for high-speed boards, the data transfer rate and therefore the method of transferring the data may need to be considered. For a large block count, DMA transfer, where available, will need to be used, while for small block counts, polled I/O or interrupt I/O, may be used where an on-board FIFO is utilized.

A little care must be taken when using block mode triggering with variable channel rate sampling, where some channels are sampled more often than others. It is possible that very large phase shifts are introduced because of the different times at which the two methods sample the data. Where variable channel rate sampling is used, conventional continuous channel scanning should be used.

5.7 Speed vs throughput

Throughput, or the speed at which an A/D board can acquire data, is always a consideration in data acquisition systems. There is, however, some confusion in the throughput figures quoted by board manufacturers when relating the performance of a particular board. Often these figures relate to the maximum data acquisition rate and can depend on the particular method of storing values into memory. Herein lies the real key to the throughput of data acquisition boards.

Strictly speaking, the throughput specification of an A/D board indicates the total number of analog signal input samples that can be converted to their digital equivalents per second. As it is usual for several analog input circuits to share a common A/D converter, the number of input channels in use also affects throughput. Therefore, the sample rate of each channel is the total throughput divided by the number of channels sampled.

$$\text{Maximum throughput / channel} = \frac{\text{Total throughput}}{\text{\# Channels used}}$$

For example, if you wish to sample four channels at 50 kHz each, you need an A/D board with a throughput of at least 200 kHz (four inputs X 50 KHz / input).

A/D board conversion throughput is determined by:

- Acquisition time: the time needed by the signal conditioning and acquisition circuitry (multiplexer, amplifier, filter and sample / hold), to obtain and present an accurate analog input signal to the A/D converter.
- Conversion time: the time needed to perform the actual A/D conversion and have the digital output available in a register or buffer to be transferred to memory. Here the speed (determined by the type of A/D conversion process or combination of processes) is paramount. High speed, high quality A/Ds with low drift and requiring less calibration will obviously increase the cost of a data acquisition board.

Total throughput is also determined by:

- Transfer time: the time required to transfer data to and/or from the data acquisition board to memory, where software can determine if it is to be displayed and/or transferred to a permanent storage location. The transfer rate of the data acquisition system, is the slowest of either the board throughput, the data transfer rate or, if relevant, the storage rate.

In most data acquisition systems the storage rate is not significant, since the amount of data involved is small enough to allow it all to be stored in memory, left there, displayed if necessary, and stored later. The transfer rate of the link from memory to permanent data storage is therefore not important.

Where the volume of data acquired is greater than the memory available to it, and particularly when the required throughput is very high, then alternative methods of storing the data must be found.

5.8 D/A boards

D/A boards convert digital signals from a host computer into an analog format for use by external devices such as actuators in controlling or stimulating a system or process. The principle component of all D/A boards is the digital-to-analog converter (D/A Converter).

D/A boards fall into two main categories:

- Waveform generation boards
- Analog output boards

Waveform generation D/A boards

As their name would imply waveform generation D/A boards are used in the high speed generation of analog waveforms, typically in a laboratory environment for the reproduction or simulation of noise, audio signals, power line signals and also for many other control applications.

The functional diagram of a waveform generation D/A board is shown in Figure 5.24 and comprises the following main components:

- D/A converter (DAC)
- Output amplifier/buffer
- FIFO buffer
- Timing system
- Expansion bus interface

Each of these components plays an important role in determining the speed, accuracy, and flexibility with which the D/A board can generate analog waveforms.

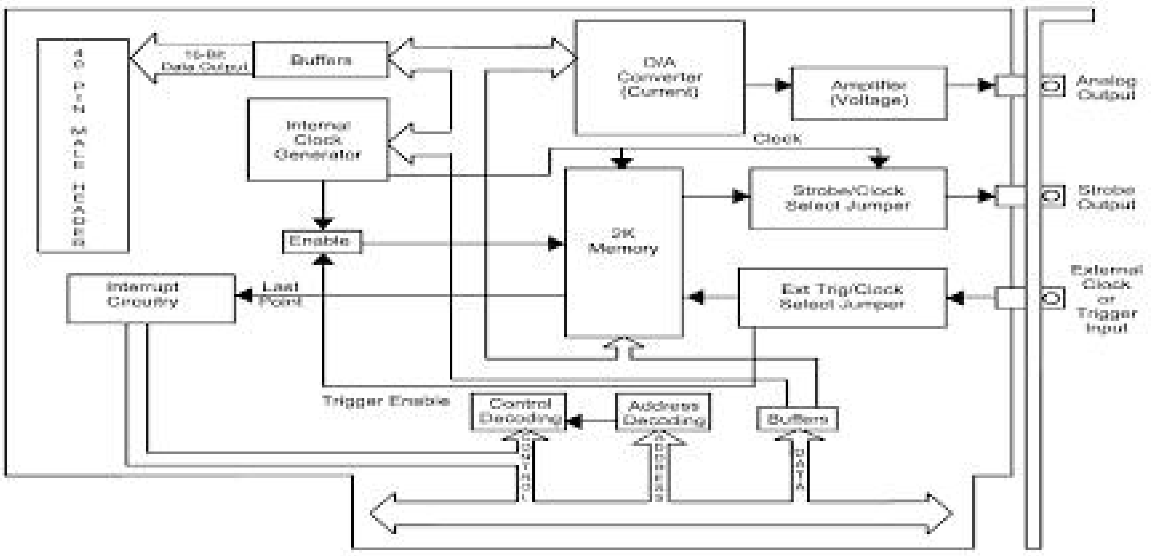


Figure 5.24
Functional diagram of a waveform generation D/A board

Analog output D/A boards

Unlike high speed, high resolution, waveform generation boards, more typical analog output D/A boards, as shown in Figure 5.25 and used for example in industrial control, are not designed for outputting precise waveforms. Instead, they maintain constant output levels unless instructed otherwise. While multi-function data acquisition boards often include two or more analog output channels, applications which require many dedicated analog outputs are most efficiently provided for by dedicated analog output boards.

Whether part of a multi-function DAC board or a dedicated analog output D/A board, the D/A conversion sub-system is straightforward in design and can be divided into two main functional components:

- D/A converter
- Output amplifier and buffer

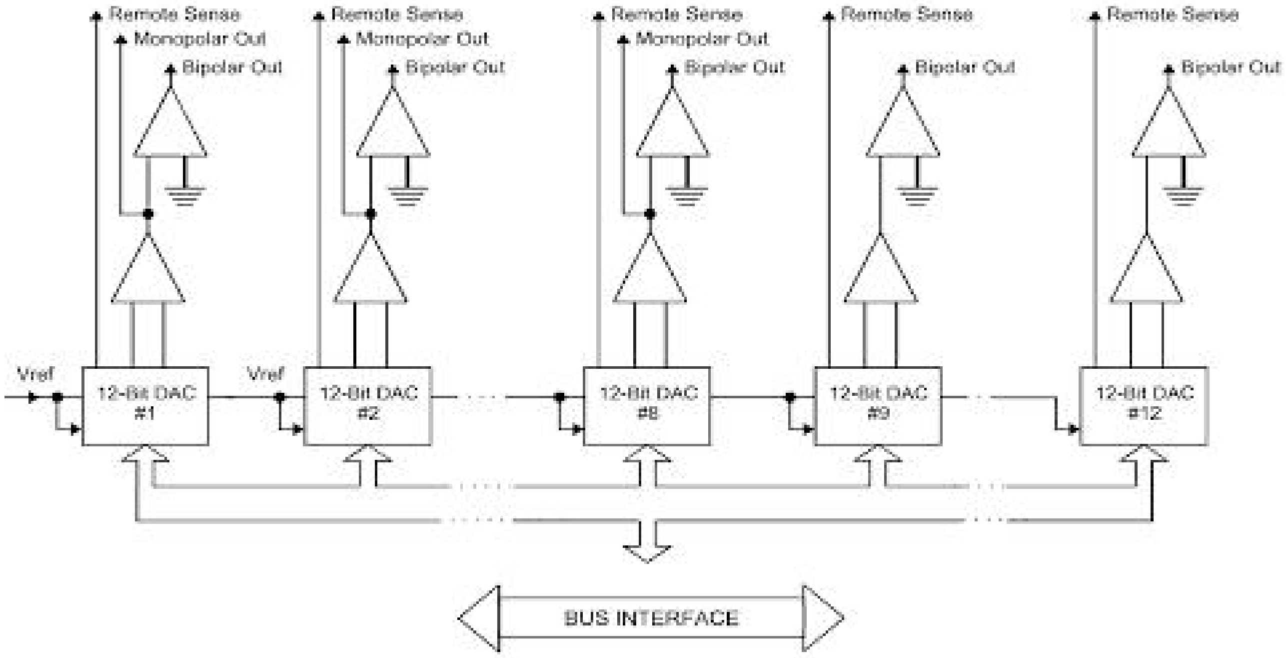


Figure 5.25
Functional diagram of an analog output D/A board

Analog output D/A boards typically have between two and sixteen dedicated output channels, each with its own D/A converter and where required output buffer/amplifier.

5.8.1 Digital to analog converters

Digital to analog converters (D/A converters or DACs) accept an n-bit parallel digital code as input and provide an analog current or voltage as output. The primary output value is a current, however, this is easily converted to a voltage using an operational amplifier. A D/A converter consists principally of a network of analog switches, controlled by the input code, and a network of precision weighted resistors. The switches control currents or voltages derived from a precise reference voltage and provide an analog output current or voltage. The output current/voltage represents the ratio of the input code to the full-scale voltage of the reference source. The main types of current output DACs and their specific important parameters are discussed in the following sections.

Weighted-current source D/A converters

The weighted-current source method of implementing a D/A converter is shown in Figure 5.26.

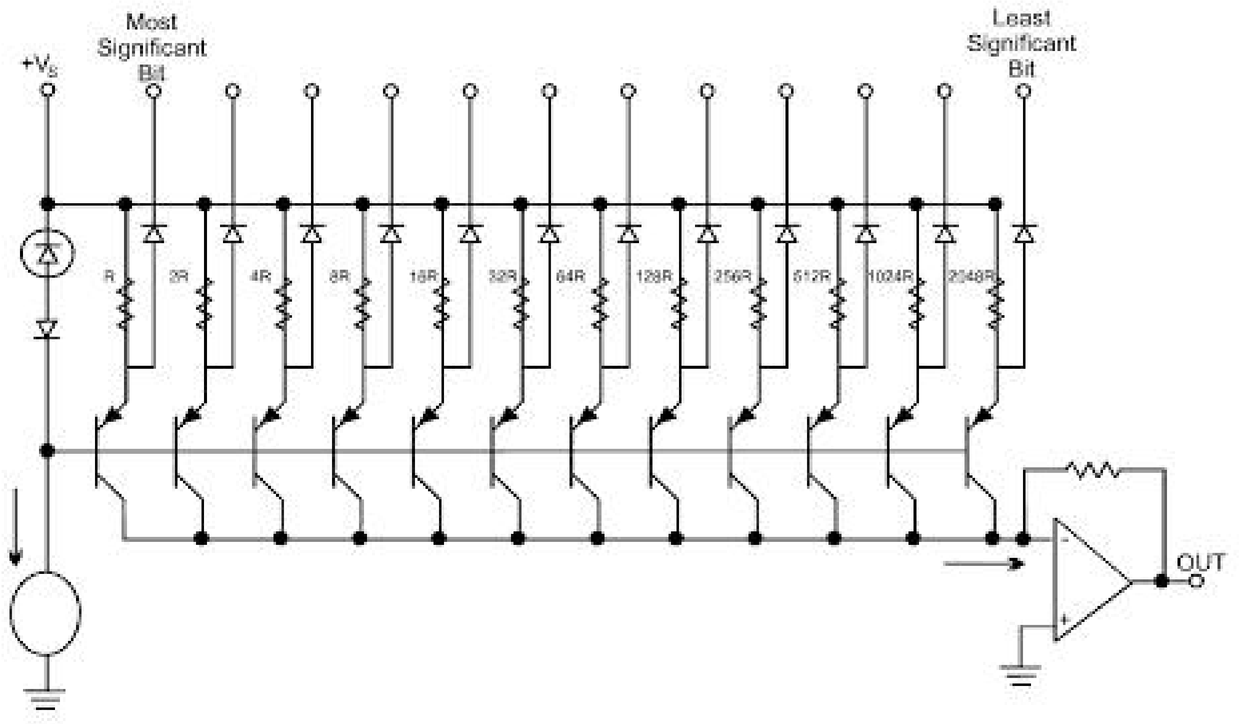


Figure 5.26
N-bit weighted-current source D/A converter

This method creates an output current, I_T, which is the summation of the weighted currents from each of the parallel transistor sources; the current contributed by each transistor set by the resistances *R*, 2*R*, 4*R*, 8*R*, etc. The selection of the currents to be summed is determined by the digital code appearing at the input. For example, if the digital voltage at the MSB is logic low, current will flow through the forward biased diode rather than through the collector of the transistor, and the transistor will remain off.

When the digital voltage at the MSB is logic high, the current flowing through the collector and emitter of the transistor is equal to V_{REF}/R. A stable reference voltage with suitable temperature compensation (base-to-emitter for each transistor) ensures that each transistor produces a constant emitter current inversely proportional to the collector resistance.

Since the output from the inverting summing amplifier is $V_0 = -I_T\, R/2$ the output voltage is directly proportional to the voltage reference according to the equation

$V_0 = V_{REF}\,(B_0 2^{-1} + B_1 2^{-2} \ldots + B_{n-1} 2^{n-1})$

Weighted codes other than straight binary can be converted by proper choice of the weighting resistors.

R-2*R* ladder D/A converters

A D/A converter which uses resistors of only two values, *R* and 2*R*, is shown in Figure 5.27.

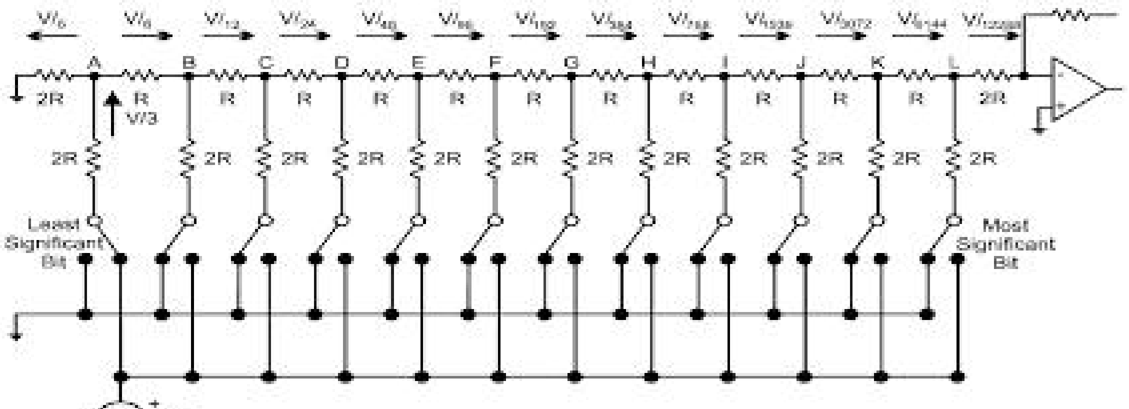


Figure 5.27
N-bit R-2R ladder D/A converter

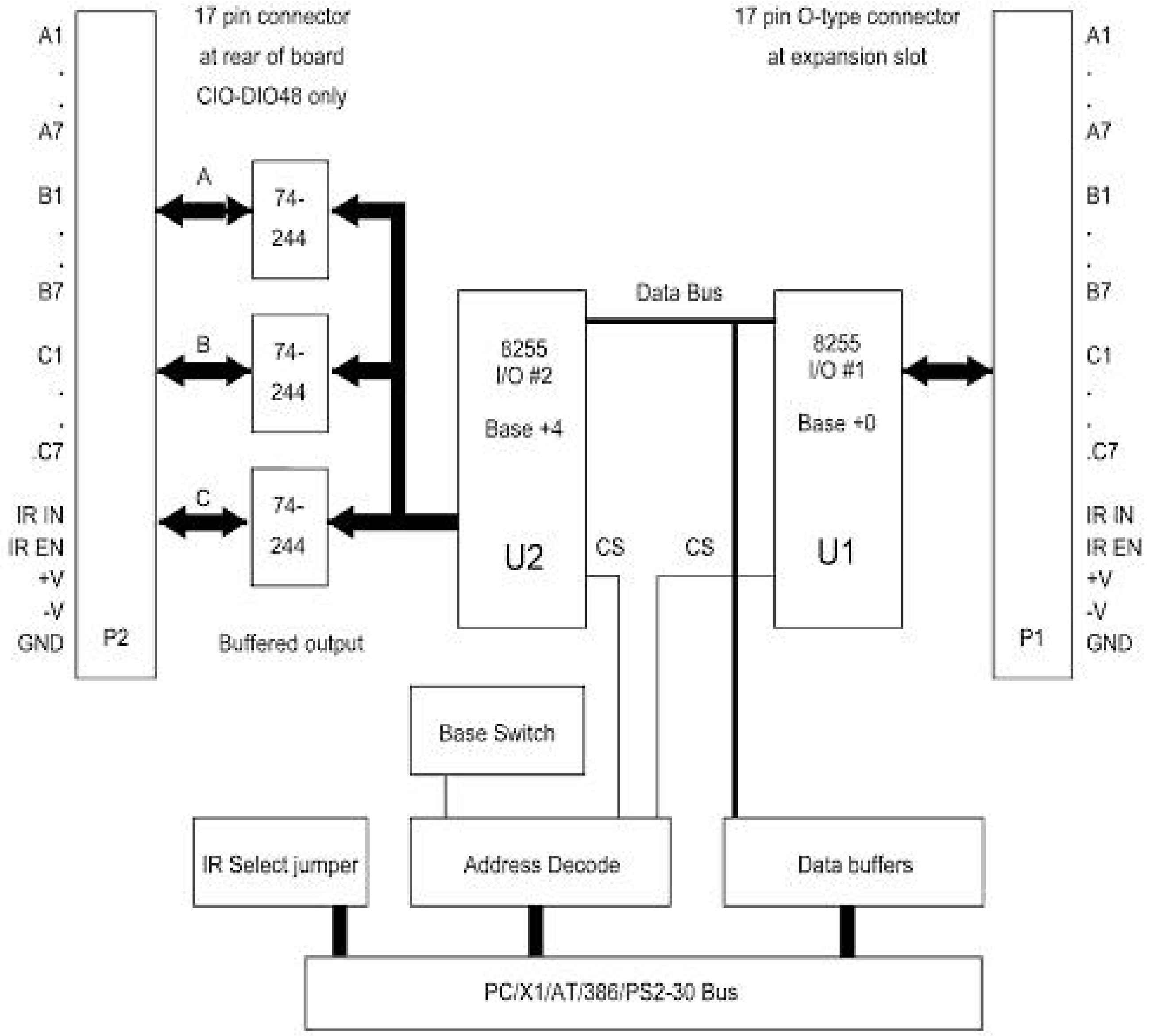


Figure 5.29
Typical digital I/O board block diagram

Figure 5.29 shows a typical digital input/output circuit. A data acquisition card or device will often combine the inputs and outputs on the same channel. The I/O channels will do either inputs or outputs but not both at the same time. The software configures the I/O channel on the card or device as either an input or output. The voltage applied to the channel, defined as an input, is usually in the form of a ground or common. This ground biases the LED of the opto-coupler on. The LED in the opto-coupler shines on the base of the transistor and turns it on. This in turn tells the input circuitry in the card that the input is 'on'.

When the channel is configured as a digital output the outgoing ground turns on the transistor. This supplies a ground to the output channel. This ground is used by the device in the field, a relay or sold state relay, to turn something on. The chips that are used to do digital output often have either diode or capacitor snubber networks on their outputs. These built in snubber networks are not intended to replace external networks that are required on long lines.

Non-latched digital I/O

Non-latched digital I/O is the mode of operation in which the state of a digital output line is updated immediately a digital value is written to the digital I/O port. In addition, for digital I/O lines configured as input lines, the current digital value present on the line when the port is read is the value that is returned. Non-latched digital I/O is the most common and simplest implementation used in digital I/O interfaces and is supported by all boards with digital I/O lines. The direction of the digital lines of a digital I/O port is conveniently set by software and can be changed as many times as required.

Latched digital I/O

For applications that require handshaking of digital data, latched digital I/O is used. In this mode of operation, an external signal determines when the data is either input to or output from the digital I/O port. The signals that are used to control the transfer of data are sometimes known as handshake lines. They are used to ensure that the digital interface is ready to input digital information appearing at the input lines, sent from a remote device or instrument, or a remote device or instrument is ready to receive data available to be sent on the output lines of a board's digital I/O interface. They could also provide digital control, in particular, to switch AC or DC power relays or alarm relays, or provide the PC tremendous power for a variety of industrial control applications.

Where digital I/O lines are used to drive panel LED displays or switch AC and DC power through relays, a high data transfer rate is not required. What is more important, however, is that the number of output lines should match the number of processes that are controlled, and that the amount of drive current required to turn the devices on/off are less than the available drive current from the output lines of the digital I/O interface.

5.10 Interfacing digital inputs/outputs

5.10.1 Switch sensing

In many applications, and particularly in industrial monitoring and control, switches form a primary interface for control actions that must be initiated by an operator. Operator controlled panel switches can be used to indicate that an action should be performed by the system. Alternatively, where switches have multiple contacts, one contact can actually perform the action required (i.e. turning on a pump), while another contact can be used to indicate that the action was actually initiated. The monitoring of abnormal system conditions can also be made easier by using limit switches to indicate that an alarm condition has been reached. In each of these cases, and in many other applications, the condition of the switch contact must be determined, requiring that the switches be interfaced and sensed by DAQ hardware.

Since switches are passive devices with no power source, they must be made to emit TTL signal levels for direct connection to a TTL compatible digital I/O interface. The open/closed position of the switch is then deduced by the TTL logic level read at the digital input. This can be carried out quite easily, as demonstrated in the two switch-sensing connections shown in Figure 5.30.

In the first switch-sensing connection, a pull-up resistor connected to one side of the switch is pulled up to the supply voltage level, which is normally available from the DAC board. The open position of the switch contact is deduced by the high logic level read at the digital input. When the switch contact is in the closed position the digital input is connected to digital ground. This configuration has higher noise immunity and has the added advantage that one terminal is connected straight to ground and can be grounded at a convenient point near the location of the switch.

In the second switch-sensing configuration, a pull-down resistor is used to present a low logic level (digital ground) at the digital input, when the switch contact is open. When the switch contact is in the closed position, the digital input is connected directly to the 5 V supply voltage. The value of the pull-up or pull-down resistor is determined by the supply voltage and the digital input current sink capability.

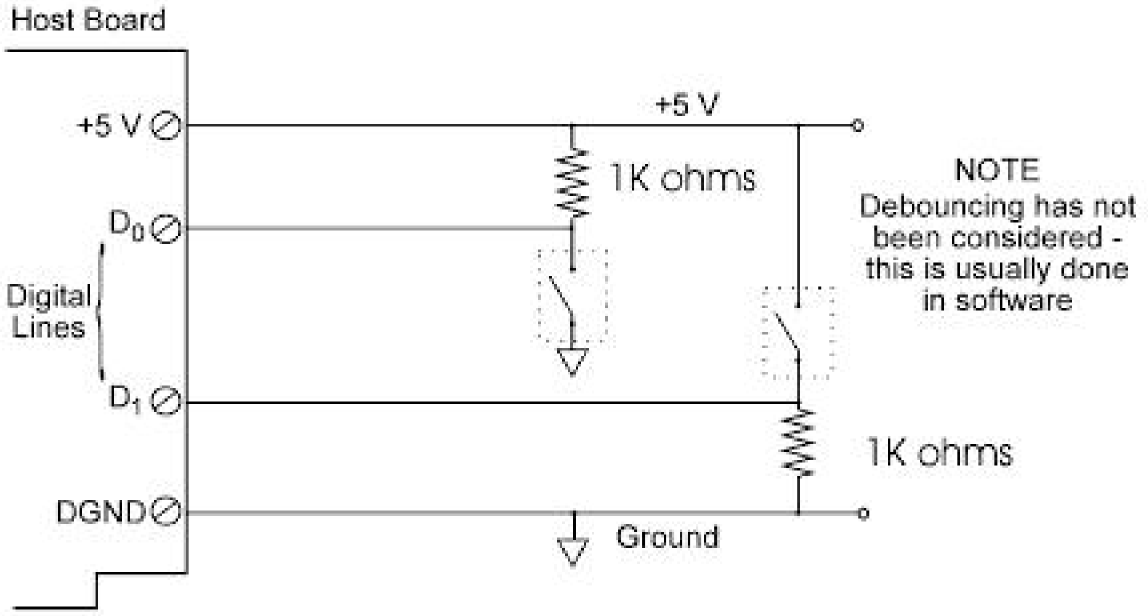


Figure 5.30
Switch position sensing circuits

Where it is likely that the signal source will be a button, switch or contact that bounces or glitches, or the signal may be a voltage higher than TTL levels, additional de-bounce and/or voltage divider circuitry is required.

5.10.2 AC/DC voltage sensing

In industrial monitoring and control, the throwing of a switch is used to begin or end an action, such as switching power to a motor or other machinery. In critical processes, the action of turning the switch is not necessarily enough to confirm that the motor has received power. This would require the sensing of the AC/DC voltages present at the motor inputs. As the AC/DC voltages involved could be quite high, any sensing circuitry directly connected to the digital I/O interface would need to provide high isolation, in addition to compatibility with the TTL digital I/O interface. A very simple AC/DC voltage sensing circuit that performs these tasks is shown in Figure 5.31.

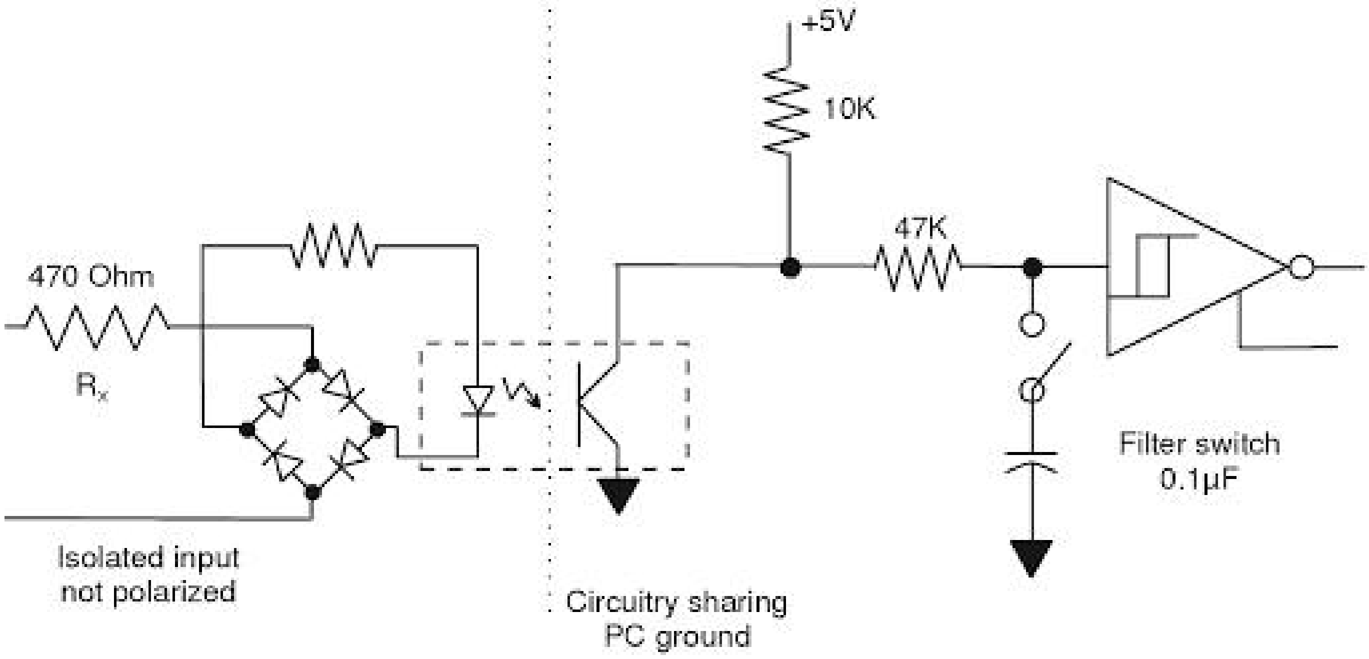


Figure 5.31
AC/DC voltage sensing circuit

There are several advantages of this low cost circuit. It is polarity insensitive and can be driven from 12 or 24 volt AC control transformers, the input voltage range can be extended

Index