

Chapter 2

DATA COMMUNICATIONS CONCEPTS

This section introduces the subject of data communications by defining some of the terminology and explaining many of the basic concepts. Data communications is a topic that relies heavily on terminology and many of these terms have their roots in the historical development of communications and computers. Without a working knowledge of these terms and concepts the material to follow will become difficult to understand.

Information and Data

Prior to any discussion relating to data communications it would seem prudent to take a step backward and consider the question of what data actually is. Often we use the terms *data* and *information* interchangeably, but are they really the same? Let's start with the word *data*.

Data

The word *data* could be defined as:

a representation of facts, concepts, or instructions in a formalised manner suitable for communication, interpretation, or processing by human beings or by automatic means.

Information

Information, on the other hand, could be defined as:

the meaning that a human being assigns to data by means of the conventions applied to those data.

In other words, information contains meaning to those who acquire or disseminate it. Data is only the means of representing information. Data is not meaningful in and of itself. It is merely a construct used for the transfer of information. Having made this distinction, you should realise that you will often find these terms used interchangeably. You will find this to be true in this document, as well as others.

Communications and Communications Systems

If information is meaningful, and data is the construct used to convey that meaning, *communications* is the action of transferring that information, in the form of data, from one location to another. A *communications system* is the mechanism used to facilitate that transfer. Mankind has developed a variety of methods to communicate information using a variety of mechanisms. Obviously, the human body has been wonderfully equipped for communication through our abilities and senses. Speech and hearing serve us well in this capacity. We also communicate visually through signals, body language and other

movement. Even our senses of smell and touch are powerful communication tools.

Unaided forms of interpersonal communication, however, are limited by how far we can see and hear. As we will see later in this section, mankind has extended the distance we can communicate using various types of technology. The most significant of these developments, that of electricity and electronics, opened the door to communications over much longer distances. Radio systems extended those distances even further, facilitating communications around the world and into space. In recent years, computer technology has enabled the evolution of *digital* communications systems, which has made a variety of types of communication accessible to virtually everyone on the planet.

Elements of a Communications System

Whether a communications system consists of two people talking, children sending messages via two cans and string, or a sophisticated satellite data link, the system will consist of five main components:

- The **source** of information.
- The **transmitter** of the data.
- The **channel**, or **medium** (string, cable, sound or radio waves) that carries the signal.
- The **receiver** of the data.
- The **destination** of the information.

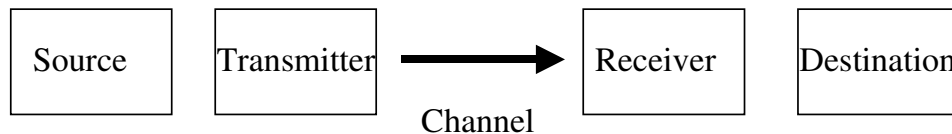


Figure 1. Block Diagram of a Communications System

Data and Signals

Communications systems can be further categorised by

- the **type of data** to be transferred
- the **type of signal** used to transfer the information

Data

The information to be transferred may be in *analogue* or *digital* form. Voice, music, video and other continuously changing signals are considered to be analogue information. Slow changing process measurements, as found in instrumentation systems, are also a type of analogue data.

Examples of information in *digital* form include computer data, on/off control signals, or other binary-based information. In recent years the trend has been toward converting most kinds of analogue data into digital form before transfer. At the receiving end of a communications link the digital signals are converted back to analogue, if necessary. This process allows standardisation of communications systems.

Signals

Signals used to convey information between communications systems may also be analogue or digital in nature. Analogue signals range across the electromagnetic spectrum. At the low end of the spectrum they may take the form of slow changing voltage or current values, such as those used in instrumentation and control applications. Older style telephone systems conveyed information using a limited band of voice frequencies. Radio systems carry information using frequencies that range from a few kilohertz to several giga-hertz. Fibre-optic systems use visible and infrared light to carry information.

Digital signals are simply discrete voltages or currents, usually binary, which represent two states. As we will see later, bipolar voltages that switch back and forth at some predetermined rate often represent these states.

You may have noticed that the distinction between data and signals blurs in some situations. For example, we have referred to voice signals as both data and signals. When the actual audible signals are converted to analogue electrical signals they become data. When this same analogue electrical signal is transferred from source to destination we call it a signal. The data may or may not undergo some type of conversion to become a signal. When data is transmitted without any conversion, such as modulation of a carrier, we often call the signal a *baseband signal*.

System Types

Communications systems, then, may take any one of the following forms:

- Analogue data / analogue signal
- Analogue data / digital signal
- Digital data / analogue signal
- Digital data / digital signals

Communications Channels

A key part of any communications system is the channel by which the information is conveyed from source to destination. The type of channel used in a given system depends on requirements such as:

- Type of data to be transferred
- Type of signal required
- Volume and speed requirements of the system (which translates into frequency bandwidth requirements)
- Distances and destinations of the transfer
- Applications of the information to be transferred
- Many other factors

Communications Channel Types

In general, communications channels can be divided into two types:

Guided Transmission Channels

Examples:

- wire pairs, twisted pairs, co-axial cable
- optical fibre
- waveguide

Unguided Transmission Channels

Examples

- radio
- infrared
- ultrasonic

Communications Tasks

The process of transferring data involves a number of steps, the complexity of which is dependent on the sophistication of the communications system. In general, the following steps are required:

- information is **coded** into a form which can be transmitted
- the message is **put on** the medium
- the receiver takes the message **off** the medium
- the receiver **decodes** it back to a useable form

For information transfer to occur both ends must agree on how these steps will be handled. Some of the issues include:

- what kind of **hardware and electrical signals** to use to transmit and receive the signals
- in what form to **encode and decode** the information for transmission
- other rules specifying **how to send and receive** the information

A partial list of communications tasks follows:

Transmission System Utilisation

This is the process of getting the most efficient use of the communications facilities. It may involve some form of multiplexing to transfer multiple information streams through a common channel.

Interfacing

This is the process of ensuring that data can travel from one stage of a communications system to the next. It may involve using the appropriate hardware or software to facilitate the transfer.

Signal Generation

Every communications system must be able to convert the data into the appropriate signals before putting it onto the channel.

Synchronisation

Timing is everything. Communications systems must be able to co-ordinate the data transfer so that the receiving end can detect the beginning and end of messages, as well as pick out each message character as it is received.

Exchange Management

Exchange management is also a process of co-ordination. In bi-directional communications the system must be capable of initiating the connection, detecting the establishment of the connection, initiating the data transfer, and other functions. This requires agreement on rules for the process, usually specified by a *protocol*.

Error Detection and Correction

Related to exchange management, error detection and correction is another convention which specifies the techniques for detecting errors in data transfer and how the problem will be rectified. Several techniques are possible and some systems may use more than one at different levels of the communications process.

Flow Control

Flow control, another protocol function, is the process of ensuring that the receiver is not overwhelmed by flood of data. Flow control may use hardware or software means to keep the sender informed when it must temporarily discontinue data transfer.

Addressing and Routing

These are tasks related to relatively more complex communications systems in which data must be received by the correct destination. Usually, information specifying the recipient is included with the message before it is sent.

Recovery

Recovery is the process of re-establishing communications, or at least resuming normal activity, after data transfer has been lost unexpectedly.

Message Formatting

Both ends of a communications link must agree on how a message is formatted, including the type of codes used.

Security

As we become more dependent on data communications in all areas of life, the desire to ensure that the information transferred is protected from interception and tampering has become increasingly important. Many systems implement some form of encryption to accomplish this task.

Network Management

Despite what some proponents of data communications may claim, communications systems still require fairly intensive maintenance, mostly in the area of configuration, monitoring, planning and modification. Network management is the ongoing process of maintain a system at optimum performance levels.

Data Coding

Written words, numbers and other recorded data are made up of characters. In a data communication system, each character is represented using digital techniques. **Digital information** is based on the binary number system which uses only two states (on/off, true/false, 1/0). Since a single binary element, or bit, can only represent two states, multiple bits must be used to represent more than two **characters**. Systems exist which use 5, 6, 7 and 8 bits to represent characters. These representations are called **binary codes**.

For transmission on data communications systems, each character (**letters, numbers, punctuation and symbols, and control codes**) is converted into a unique binary code which can be transmitted and received using the appropriate hardware and software.

When a communications link transfers **binary** codes it must convert the 1's and 0's into electrical signals that can be placed on the medium. In many cases, two DC voltage levels represent these two states. The exact values of the voltages and which state they represent depends on the rules established beforehand. In other systems, such as modems, the two states are represented by **two states of modulation of an AC signal**. Many variations in voltage, frequency and other parameters exist between different systems.

Serial and Parallel Data Transfer

Digital data can be sent and received in two main ways:

- Serial
- Parallel

Parallel Data Transfer

Parallel data transfer allows all the bits of the code to be sent at the same time using multiple conductors. This transmission method can transfer large amounts of information very quickly. Because this approach includes the cost of multiple conductors, it is typically used when the distance between sender and receiver is relatively short.

Serial Data Transfer

Serial data transfer sends each bit of a code sequentially. This is much slower but requires only two conductors (a signal conductor plus a return conductor) to carry the signals. Sometimes additional conductors are added for ancillary purposes. For example, some systems implement exchange management and flow control using signals on extra conductors in the communications channel. To minimize noise pickup many systems add a shield conductor to the signal cable.

Serial data transfer is typically used for systems where distances are longer, to minimise the costs of multiple conductors run over long distances. The main performance tradeoffs associated with serial transfer are reduced throughput and the need for additional synchronisation capabilities.

The History of Data Communications

Perhaps the earliest example of digital signals being used to transfer information over long distances dates back to the **Greek and Roman signal fires**. This was a purely binary system, signalling the occurrence or non-occurrence of predefined events. We have all heard of the use of **jungle drums** by African tribes and **smoke signals** used by North American Indians. These were also digital systems, essentially. They conveyed information by changing the length of time between beats or puffs of smoke. In each of these cases the basic requirements of a communications system are met. Senders, receivers and medium agree on methods of encoding and coding the information.

Telegraphy was the earliest method of serial data communications using electrical methods. Again, the system included senders, receivers and a medium, as well as an encoding system called the Morse code (shown on one of the next slides). The terms “mark” (signifying a logic 1) and “space” (signifying a logic 0) were coined as a result of early attempts to automate the telegraph. This system drew lines on a strip of paper moving under the armature attached to the receiving coil.

The **teletypewriter** was a later development that further automated the transmission and reception of data using electromechanical systems operated by synchronous motors.

The arrival of the **telephone**, designed for voice communication, prompted the development of techniques to transmit and receive data over this existing medium.



Figure 2. The Telephone System, originally voice only, now carries digital data

Eventually **networks** emerged, which attempted to link multiple transmitter/receivers using a common medium and allowing access to data from many sources.

The Telegraph

The telegraph was the first digital communications system. It employed DC batteries to produce current flow in an electrical loop that energised electromagnet coils at both the source and destination ends of each link. When no messages were being sent the loop was held in the “idle” state, (considered a mark, or logic 1) by a latched switch. When this switch was opened the electromagnets at both ends de-energised, alerting the receiving operator that a message would be coming in. A receiving operator could also open his switch during the reception of a message, alerting the sender that, either the line was open or the receiving operator had a priority message to send. This operation was called a *break*. During data transmission a momentary switch, called the key, was used to send pulses of current to the receiving end. Each time the current was interrupted the electromagnet coil would release an arm creating an audible click. Sending short clicks (dots) and long clicks (dashes) transferred the binary information. The encoding system was based on the Morse code.

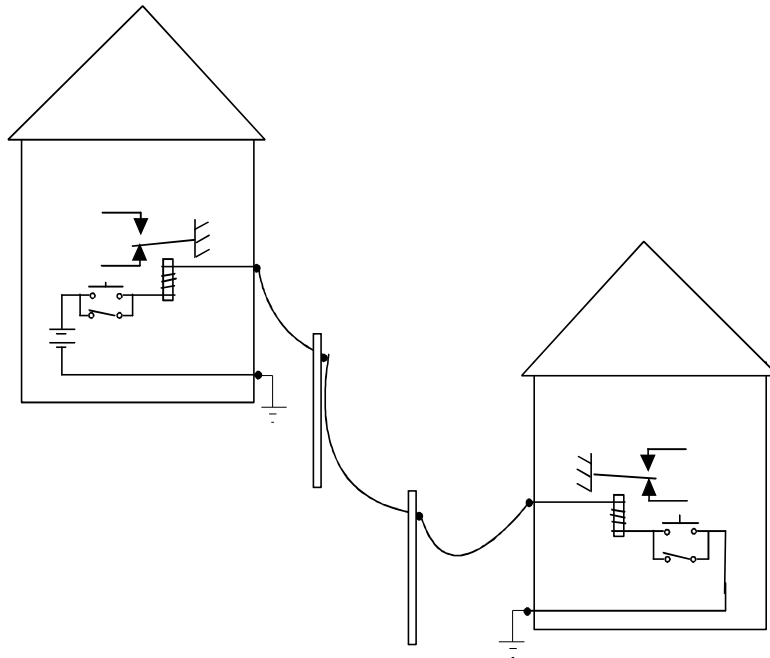


Figure 3. A Simplified Diagram of a Telegraph System

Telegraph operators became very adept at picking up the digital codes sent from the other end of the link, to the extent that they could recognize the sender by his personal keying speed and style.

The Morse Code

The telegraph employed a coding system called the Morse code. The Morse code is a **variable length code** that consists of “dots” and “dashes”, a dot being a short duration of a space and a dash being a longer duration. The Morse code uses thirty-eight different codes to represent all the upper case alphabetic characters, two punctuation symbols (comma and period) and the ten numeric characters.

The advantage of a variable length code can be seen from the table above. By using shorter codes for characters used more often in typical messages, the efficiency of the data transfer could be improved. Notice the letter “E”, a very common character, is represented by a single “dot”. A single “dash” represents the letter “T”.

Although there were several unique advantages to the Morse code it was not suitable for use with automated and computerised communications systems. These types of systems more easily implement fixed length codes where the number of bits can be predicted for each character. We will consider other types of codes that meet this requirement.

Data Communications Terminology

Data Terminal Equipment

In the early days of data communications the user interfaces to computers were called ‘terminals’. These devices consisted of a keyboard that converted these keystroke characters into binary codes. Often, these codes were transferred onto cards or paper tape by punching hole in them. The data was later loaded into the computer using an optical card or tape ‘reader’. This was satisfactory for programming computers in the same room but, as time went on, the need to transfer data over longer distances grew. Terminals (or any other device

that produced digital data) became known as **Data Terminal Equipment (DTE)** because they were a source of data.

Data Communications Equipment

Data Terminal Equipment, however, was not designed to transmit the data over long distances. Since the telephone system provided a ready-made network of communications media it was logical to transfer data via the telephone system. However, the telephone system was designed primarily to carry analogue voice signals. The result was the development of devices that could convert digital signals to analogue signals appropriate for transmission over the telephone system. These were called modems (from the terms modulator-demodulator). Modems were considered to be **Data Communications Equipment (DCE)**.

Today we consider any device that sources data to be a DTE and one that facilitates the transmission of the data to be a DCE. In recent years the term DCE has come to stand for Data Circuit-Terminating Equipment to more broadly reflect its purpose and function.

Parts of a Serial Data Communications System

We have already said that the most basic communications system includes a sender, medium and receiver. We have further considered early data communication systems using the telephone system, that included a terminal (DTE), and a modem (DCE). The diversity of modern communications systems forces us to expand some of these definitions and to look more closely at the devices and functions involved.

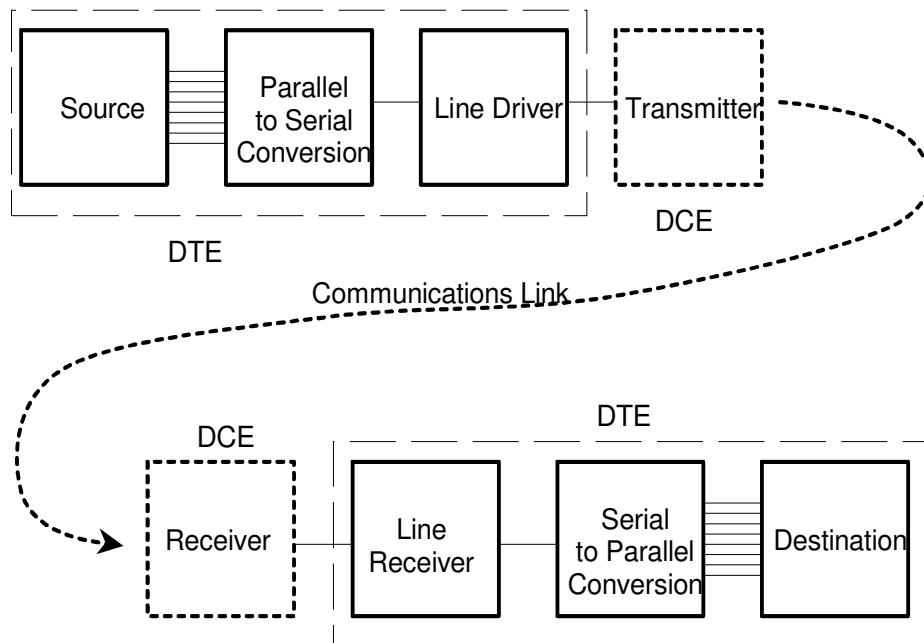


Figure 4. Parts of a Serial Digital Communications Link

The diagram above shows a system with several functional blocks, that could represent various types of equipment. For our purposes here we will consider the case of two personal computers communicating via a serial data link. (This diagram could represent any two devices communicating digitally.) The source and destination blocks represent the actual processor/memory components of the computer. The parallel to serial and serial to parallel blocks represent the Universal Asynchronous Receiver/Transmitter (UART) circuitry usually

contained inside the computer. The Line Driver/Receiver blocks represent the driver circuitry such as an RS-232 or RS-422 interface device. These first three blocks are all considered part of a DTE device.

The last block is the DCE device, and is often a modem. This block is shown in dotted lines to indicate that it is necessary for interfacing the DTE to a special medium, such as a telephone line. Today a DCE device could be a variety of devices, such as a network interface or a special line driver.

So far we have considered communications in one direction only (although it is obvious that the telegraph allowed communications in both directions). For the sake of defining categories of communications systems we will examine several different types of system structures.

Data Communications System Categories

Point to Point Communications

The first and simplest type of communications system, as we have already seen, is the **Point to Point** system. In this system communications occurs between only two points or communications nodes. Examples of interface standards that allow point to point signalling include the common RS-232 standard, as well as RS-422 and RS-423.

Industrial systems often use point to point signalling for communicating between a personal computer or hand-held programmer and programmable logic controllers (PLC), remote terminal units (RTU) or other standalone systems.

Point to point communications systems can be divided into three different types:

- Simplex
- Half-Duplex
- Full-Duplex

Simplex

Simplex communications systems convey information in one direction only. Although systems such as this are limited in usefulness in industrial situations they do exist. In general data communications a common type of simplex system is the digital personal pager. Phone numbers or other alphanumeric data are sent from a central location to the pager that displays the information on a small display screen. Since the pager has no transmitter, no acknowledgement to the sender can be made. Pagers often employ special error checking and correction techniques that can reconstruct the data in the event of an error in reception.

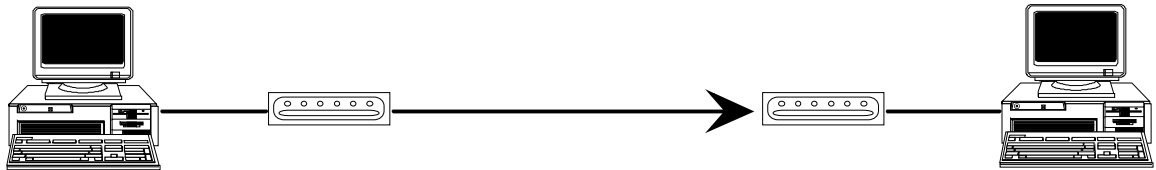


Figure 5. Simplex Communications

Half-duplex

Point to point communications systems are very common. This type of system often uses a single media channel (such as a pair of wires) to convey data. Unless special techniques are used, the channel can carry only one signal at a time. Therefore, the system must 'take turns'. First one end of the link sends a message, then the other end responds.

Inherent in a half-duplex system is the need for exchange management and flow control functions. Later, we will consider various ways that this is accomplished.

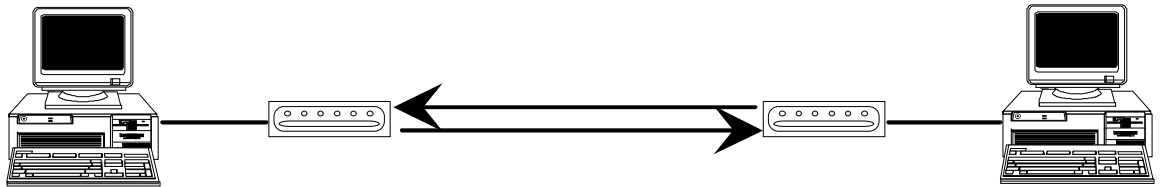


Figure 6. Half-duplex Communications

Full-duplex

A **full-duplex** system can send information in both directions simultaneously. In order to do so two communications channels must exist between the ends of the link.

The channels could consist of:

- two signal wires with a common return
- two pairs of signal wires, electrically separate
- a single pair using frequency division multiplexing (the two signals are modulated onto different carrier frequencies which do not interact)

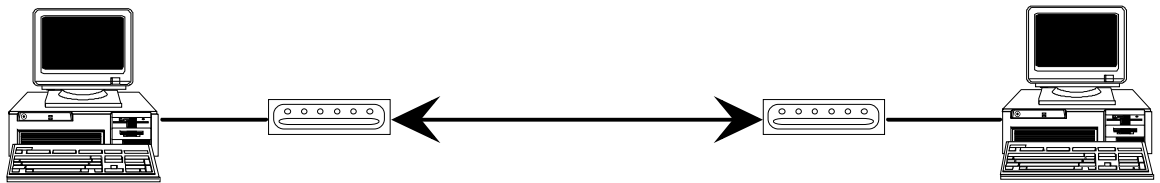


Figure 7. Full-duplex Communications

Point to Multi-point

Some communications systems are capable of sending the same data to multiple destinations. This type of system is called a **Point to Multi-point** system. In some ways this is similar to a simplex system since the information travels in one direction only. If information flow back to the source is required other techniques can be employed.

Networks

One of the most important developments in commercial and industrial data communications has been the use of **Networks** to facilitate the sharing of information and functions by multiple nodes. Networks allow more than two communications devices to communicate on a common medium, sharing resources and information. A variety of network topologies are available, each with its own features.

Data Communications Signals

As mentioned previously, the terms mark and space originate from the days of the telegraph. A moving tape was marked by the action of an electromagnet energised by the current flowing in the telegraph line. You will note from the diagram below that it is conventional to use a bipolar type of signal for digital data communications signalling. Also by convention, a mark,

or logic 1, is represented by a negative voltage, whereas a positive voltage represents a space or logic 0.

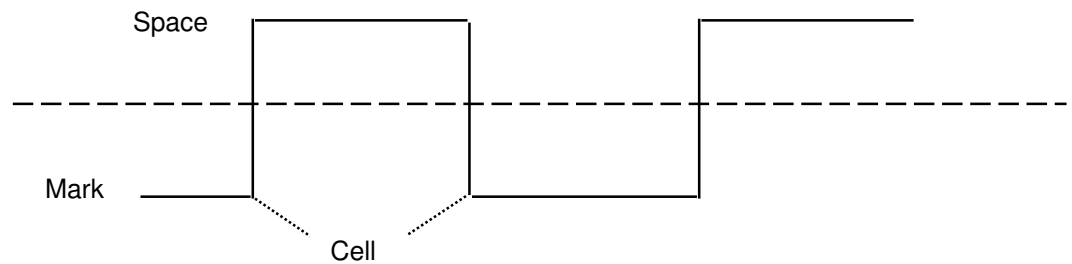


Figure 8. Terminology Used with Digital Waveforms

Signaling Speed

The speed of a data signal is dependent on the number of bits of information are transferred by the communications system. The number of bits is dependent on the length of time required to send each pulse of transmitted data. One bit time is often called a cell. The bit rate of a binary digital signal is determined by finding the reciprocal of the cell time.

Signaling rate is often specified in terms of **Baud rate**. Baud rate is considered to be the number of *cells per second*. **Bit rate** is considered to be the actual number of *data bits sent each second*.

In the case of purely digital signals Baud rate and bit rate are the same because 1 bit (a logic 0 or logic 1) is sent during each cell. However, bit rate and Baud rate can be different if more than one bit is sent during each cell time. As you will see later, multiple bits can be sent during each cell by using more sophisticated signalling techniques such as those implemented in modems.

Interface Standards and Protocols

As already mentioned one of the key issues involving communications systems is the need to **agree** on how communication will occur. For both ends of a communications link to be compatible they must agree and two main aspects of the system:

- Interface Standards
- Protocol Functions

Interface Standards

Interface standards specify the hardware and electrical aspects of the system. This includes whether the signal will be digital (DC pulses) or analogue (modulated AC), the voltage levels and signaling speed, the type of media (twisted pair, co-ax, fibre, or radio) and, if the link is a network, the topology.

Protocol Functions

Protocols are the rules which specify what kind of code will be used to represent characters, what those characters mean and are used for, how data flow will be controlled (half/full duplex) and how errors will be detected and corrected.

All communications systems must specify what their interface standards and protocols will be. In some cases some flexibility is given. For example some systems allow for different data

speeds or media to be used. When interfacing between two dissimilar systems, knowing the standards and protocols for each system makes it easier to produce a workable interface

In a later section we will consider a couple of communications models which categorise standards and protocols into several layers.

Data Communications Codes

As previously mentioned, in order to transfer the symbolic characters we use for written language and quantitative data we must convert these characters into binary codes. Binary codes are multi-bit binary numbers and can be easily converted into voltage levels or states of modulation. For effective communication we must be able to represent several different types of characters:

- alphabetic - A, a, B, b, . . .
- numeric - 0, 1, 2, 3 . . .
- symbolic - ?, !, #, % . . .
- control characters - [CR] or carriage return, [LF] or line feed, etc.....

Since there are 26 upper case and 26 lower case characters in the English alphabet, plus ten numerals and several punctuation marks, a fairly large number of unique binary codes are required. Add to that a number of non-printable control codes needed for data link control purposes and the number of codes increases further.

Older, simple systems minimised the number of characters required by using only upper case and limiting the number of punctuation and control codes. Newer systems require the use of a larger number of characters.

The number of unique binary codes available from a given length of binary number is determined by:

$$\text{number of codes} = 2^n$$

where n = the number of bits in the binary number

Depending on the rules established by the protocol, codes may be used for a variety of purposes. Communications systems use codes to transfer information in several different ways.

Alpha-numeric Characters

Sometimes textual information must be transferred to an operator interface or a database. An example of this might be an operator alert that an alarm has occurred. In this cases the data contained in the message would be alphanumeric characters represented by a binary code.

Alphanumeric characters are also sometimes used to represent a command. The software of the receiving node recognises the command and executes it. For example, some remote I/O systems use the simple string "RD" to indicate that the main measurement variable is to be read and returned to the source node.

Function Codes

Some protocols use numerical values to represent a command. For example, the Modbus protocol specifies a 'Read Coil Status' command by sending the number 01 (hex) as part of a message. (The function code must appear in the correct field of the message to be interpreted correctly.) Function codes may be sent as eight-bit binary numbers or, in some cases, each numeral is sent as a separate numerical code.

Numerical Values in Binary Format

Industrial systems must often transfer numeric values. For example, a measurement or control value. Often numeric values are transferred as eight-bit binary numbers. (Less often this kind of information is sent as a string of numeric codes.

The Baudot Code

The Baudot code is a **5-bit code** invented in 1874 by Maurice Baudot. The main advantage of the Baudot code over previous codes such as the Morse code was its **fixed length**. Electromechanical and electronic systems can more easily use a fixed length code for serial data transmission and reception because the length of each character is predictable.

The disadvantage of the Baudot code is the limited number of characters it can represent. Using 5 binary bits a total of 32 characters can be represented. As we have seen, with 26 alphabetic characters (in both upper and lower case), ten numeric characters and additional punctuation and control codes, 32 binary codes is insufficient.

Implementation of the Baudot code in Teletype applications partially solved this problem by designating two of the codes as FIGURE and LETTER shift characters. This effectively provides for two different codes: one consisting of mostly numerals and symbols and the other consisting of mostly alphabetic characters. When the FIGURE code is sent, all codes after it are interpreted according to the FIGURE code set. When the LETTER code is sent, all codes after are interpreted according to the LETTER code set. Although this system slows down the data throughput, it allows a 5-bit code to represent 62 different characters.

The Baudot Code Chart

Notice the two different columns, Letters Shift and Figures Shift, indicating the characters represented by the same code.

For example, if the code 1 1111 is sent, the mechanism of a Teletype would physically move so that subsequent codes would produce characters shown under the Letters Shift column. The code 0 0111 would produce the letter "U". The same code, sent after the Figure Shift code (1 1011), would cause the numeral "7" to be printed.

HEX	Binary	Letters Shift	Figures Shift	HEX	Binary	Letters Shift	Figures Shift
00	0 0000			00	1 0000	T	5
01	0 0001	E	3	01	1 0001	Z	+
02	0 0010	[LF]	[LF]	02	1 0010	L)
03	0 0011	A	-	03	1 0011	W	2
04	0 0100	[SPACE]	[SPACE]	04	1 0100	H	Reserved
05	0 0101	S	'	05	1 0101	Y	6
06	0 0110	I	8	06	1 0110	P	0
07	0 0111	U	7	07	1 0111	Q	1
08	0 1000	[CR]	[CR]	08	1 1000	O	9
09	0 1001	D	[WRU]	09	1 1001	B	?
0A	0 1010	R	4	0A	1 1010	G	Reserved
0B	0 1011	J	[BELL]	0B	1 1011	FIGS	FIGS
0C	0 1100	N	,	0C	1 1100	M	.
0D	0 1101	F	Reserved	0D	1 1101	X	/
0E	0 1110	C	:	0E	1 1110	V	=
0F	0 1111	K	(0F	1 1111	LTRS	LTRS

Figure 9. The Baudot Code Chart

The ASCII Code

The American Standard Code for Information Interchange is the most common code used in the western world. The details of this standard (for use in the United States) are given in the ANSI (American National Standards Institute) standard X3.4-1977. A CCITT standard, (ISA#5) specifies the ASCII code for use in the United Kingdom and other countries. The main difference between these standards is in the designation of the code for the currency symbol.

ASCII uses a 7-bit code to produce 128 unique binary patterns. Upper and lower case alphabetic characters, numerals, punctuation marks and symbols, and 34 control codes are represented. The control codes are typically used for device control and information transfer control and are not printable using conventional terminal or printer software. (When troubleshooting a communication system that uses non-printable characters special software may be needed to display these codes.)

An extended version of the ASCII code is sometimes used. This version uses 8 data bits and incorporates 128 additional unique codes for a total of 256 unique codes. Most of the additional codes are used to implement graphics characters.

The ASCII Code Chart

The code chart shown below provides a simple method of determining the binary code (or hexadecimal representation of it) for each character represented by the code. Notice that the row of binary numbers across the top of the chart show the three most significant bits (MSB) of the code, while the column of binary numbers on the left side shows the four least significant bits (LSB) of the code.

Hex		0	1	2	3	4	5	6	7
	Binary	000	001	010	011	100	101	110	111
0	0000	[NUL]	[DLE]	space	0	@	P	'	p
1	0001	[SOH]	[DC1]	!	1	A	Q	a	q
2	0010	[STX]	[DC2]	"	2	B	R	b	r
3	0011	[ETX]	[DC3]	#	3	C	S	c	s
4	0100	[EOT]	[DC4]	\$	4	D	T	d	t
5	0101	[ENQ]	[NAK]	%	5	E	U	e	u
6	0110	[ACK]	[SYN]	&	6	F	V	f	v
7	0111	[BEL]	[ETB]	'	7	G	W	g	w
8	1000	[BS]	[CAN]	(8	H	X	h	x
9	1001	[HT]	[EM])	9	I	Y	i	y
A	1010	[LF]	[SUB]	*	:	J	Z	j	z
B	1011	[VT]	[ESC]	+	;	K	[k	{
C	1100	[FF]	[FS]	,	<	L	\	l	
D	1101	[CR]	[GS]	-	=	M]	m	}
E	1110	[SO]	[RS]	.	>	N	^	n	~
F	1111	[SI]	[US]	/	?	O	_	o	DEL

Figure 10. The ASCII Code Chart

To determine the code for any given character:

- find the character on the chart
- move up the column to find the three MSB
- move across the column to find the 4 LSB

Example: Find the ASCII code for the upper case letter "D"

The 3 MSB at the top of the column are 100 (4 hex)

The 4 LSB at the left side of the row are 0100 (4 hex)

The ASCII code is: 100 0100 (44 hex)

Notice that the only difference between upper case letters and lower case letters is that the second most significant bit changes. When the SHIFT key is depressed on a keyboard, this bit is toggled to a zero to produce an upper case character. Similarly, to produce a control code (shown in the first two columns of the chart), the control key is held down while typing a letter key. The two most significant bits are changed to zeroes in the case, sending the ASCII code for a control character.

Common ASCII Control Codes

The chart above shows several of the most common control codes used in typical industrial protocols. Control codes are often sent to inform the receiving end of the link of some action to be taken or to delineate the characters following it as to their function or meaning. Sometimes this is described as "software handshaking".

As we examine protocols more closely you will become more familiar with them.

EBCDIC Code

EBCDIC is an IBM code used mostly on mainframe computers. The acronym stands for Extended Binary Coded Decimal Interchange Code.

Gray Code

Gray code is a special binary code that is usually used in position encoder applications. A position encoder produces a unique binary code for each increment of angular position of a rotating shaft. The Gray code is designed to change only one bit of the code for each increment. This provides a certain amount of inherent error detection because the receiver should never see more than one bit change per increment. If it does, there must be a problem with the encoder. Gray code is not as prevalent as it once was because stepper motors have replaced the older encoder type systems. Stepper motors allow a shaft to be positioned reliably by controlling the step sequence of the motor.

BCD Code

Binary Coded Decimal is still often used in industrial applications to interface thumb-wheel switches and numeric displays to PLCs. BCD uses four-bit binary notation for each digit of a decimal number (only ten of the possible 16 codes are used). (A somewhat obscure form of BCD represents alphabetic characters as well.)

Special Codes

As previously mentioned, in industrial systems special codes are often used for certain applications. For example, eight-bit binary is often used to represent numerical data. Sometimes numerical data is stored in 16-bit format (as in the case of PLCs). These values may be unsigned 16-bit numbers or signed 15-bit numbers (which use the most significant bit as a sign bit in a 2's complement type notation).

Binary numbers may also be used to represent function codes, commands or other messages.

Serial Data Transmission

Before moving on to more detail on how codes are transmitted in typical data communication systems take note of an important convention. Whenever ASCII, or other, characters are transmitted in a serial communication system the **least significant bit is transmitted first**.

As shown in the example below, the seven-bit code for the upper case letter D is sent from right to left.

e.g. The letter D

D = 100 0100

when transmitted:

0010001

