



Electronic Design[®] LIBRARY

Copyright © 2024 by Endeavor Media.
All rights reserved.

A compendium of articles from *Electronic Design*

What's the Difference: SERIAL COMMUNICATIONS 101



Code	Description
DCD	Data carrier detect
DSR	Data set re
RD	Receive d
RST	Reque
TD	Tran
CTS	Cl
DTR	P
RI	



INTRODUCTION

COMMUNICATION is an inherent part of computers and a range of protocols and methodologies have been employed to facilitate this communication. Low speed serial interfaces may not be as neat as high performance PCI Express but numerous serial ports tend to be a requirement for many embedded microcontrollers to interface with digital sensors and control systems.



Bill Wong
Editor,
Senior Content
Director, Electronic
Design & MWRP

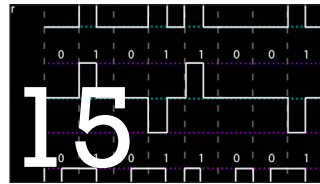
This ebook touches on the basics from baud rate to different encoding techniques that are still being used. We hope you find them useful.

CONTENTS



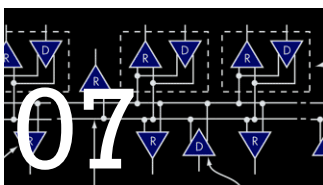
CHAPTER 1

What's The Difference Between Bit Rate And Baud Rate?



CHAPTER 4

What's the Difference Between NRZ, NRZI, and Manchester Encoding?



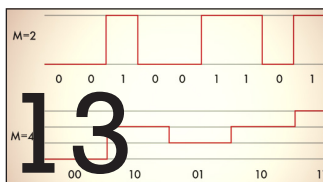
CHAPTER 2

What's The Difference Between The RS-232 And RS-485 Serial Interfaces?



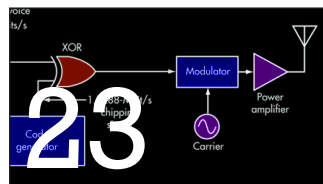
CHAPTER 5

What's The Difference Between FDD And TDD?



CHAPTER 3

What's the Difference Between NRZ and PAM?



CHAPTER 6

Fundamentals of Communications Access Technologies: FDMA, TDMA, CDMA, OFDMA, AND SDMA



Image credit: Thinkstock

CHAPTER 1:

What's The Difference Between Bit Rate And Baud Rate?

LOU FRENZEL, Technical Contributing Editor

Serial-data speed is usually stated in terms of bit rate. However, another oft-quoted measure of speed is baud rate. Though the two aren't the same, similarities exist under some circumstances. This tutorial will make the difference clear.

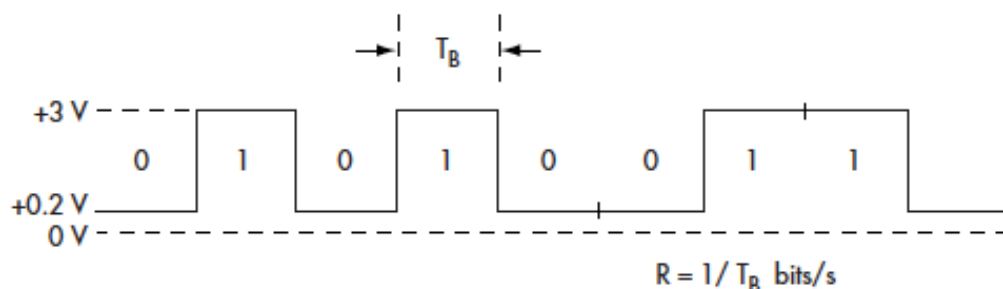
Most data communications over networks occurs via serial-data transmission. Data bits transmit one at a time over some communications channel, such as an [RS-232 cable](#) or a wireless path. **Figure 1** typifies the digital-bit pattern from a computer or some other digital circuit. This data signal is often called the baseband signal. The data switches between two voltage levels, such as +3 V for a binary 1 and +0.2 V for a binary 0. Other binary levels are also used. In the [non-return-to-zero \(NRZ\) format](#) (**Fig. 1**), the signal never goes to zero as like that of return-to-zero (RZ) formatted signals.

Bit Rate

The speed of the data is expressed in bits per second (bits/s or bps). The data rate R is a function of the duration of the bit or bit time (TB) (**Fig. 1**, again):

$$R = 1/T_B$$

Rate is also called channel capacity C. If the bit time is 10 ns, the data rate equals:



1. Non-return to zero (NRZ) is the most common binary data format. Data rate is indicated in bits per second (bits/s).

$R = 1/10 \times 10^{-9} = 100 \text{ million bits/s}$
 This is usually expressed as 100 Mbits/s.

Overhead

Bit rate is typically seen in terms of the actual data rate. Yet for most serial transmissions, the data represents part of a more complex protocol frame or packet format, which includes bits representing source address, destination address, error detection and correction codes, and other information or control bits. In the protocol frame, the data is called the "payload." Non-data bits are known as the "overhead." At times, the overhead may be substantial—up to 20% to 50% depending on the total payload bits sent over the channel.

For example, an Ethernet frame can have as many as 1542 bytes or octets, depending on the data payload. Payload can range from 42 to 1500 octets. With a maximum payload, the overhead is only $42/1542 = 0.027$, or about 2.7%. It would be even greater if the payload was anything smaller. This relationship is usually expressed as a percentage of the payload size to the maximum frame size, otherwise known as the protocol efficiency:

Protocol efficiency = payload/frame size = $1500/1542 = 0.9727$ or 97.3%

Typically, the actual line rate is stepped up by a factor influenced by the overhead to achieve an actual target net data rate. In One Gigabit Ethernet, the actual line rate is 1.25 Gbits/s to achieve a net payload throughput of 1 Gbit/s. In a 10-Gbit/s Ethernet system, gross data rate equals 10.3125 Gbits/s to achieve a true data rate of 10 Gbits/s. The net data rate also is referred to as the throughput, or payload rate, of effective data rate.

Baud Rate

The term "baud" originates from the French engineer Emile Baudot, who invented the 5-bit teletype code. Baud rate refers to the number of signal or symbol changes that occur per second. A symbol is one of several voltage, frequency, or phase changes.

NRZ binary has two symbols, one for each bit 0 or 1, that represent voltage levels. In this case, the baud or symbol rate is the same as the bit rate. However, it's possible to have more than two symbols per transmission interval, whereby each symbol represents multiple bits. With more than two symbols, data is transmitted using modulation techniques.

When the transmission medium can't handle the baseband data, modulation enters the picture. Of course, this is true of wireless. Baseband binary signals can't be transmitted directly; rather, the data is modulated on to a radio carrier for transmission. Some cable connections even use modulation to increase the data rate, which is referred to as "broadband transmission."

By using multiple symbols, multiple bits can be transmitted per symbol. For example, if the symbol rate is 4800 baud and each symbol represents two bits, that translates into an overall bit rate of 9600 bits/s. Normally the number of symbols is some power of two. If N is the number of bits per symbol, then the number of required symbols is $S = 2^N$. Thus, the gross bit rate is:

$$R = \text{baud rate} \times \log_2 S = \text{baud rate} \times 3.32 \log_{10} S$$

If the baud rate is 4800 and there are two bits per symbol, the number of symbols is $2_2 = 4$. The bit rate is:

$$R = 4800 \times 3.32 \log(4) = 4800 \times 2 = 9600 \text{ bits/s}$$

If there's only one bit per symbol, as is the case with binary NRZ, the bit and baud rates remain the same.

Multilevel Modulation

Many different modulation schemes can implement high bit rates. For example, frequency-shift keying (FSK) typically uses two different frequencies in each symbol interval to represent binary 0 and 1. Therefore, the bit rate is equal to the baud rate. However, if each symbol represents two bits, it requires the four frequencies (4FSK). In 4FSK, the bit rate is two times the baud rate.

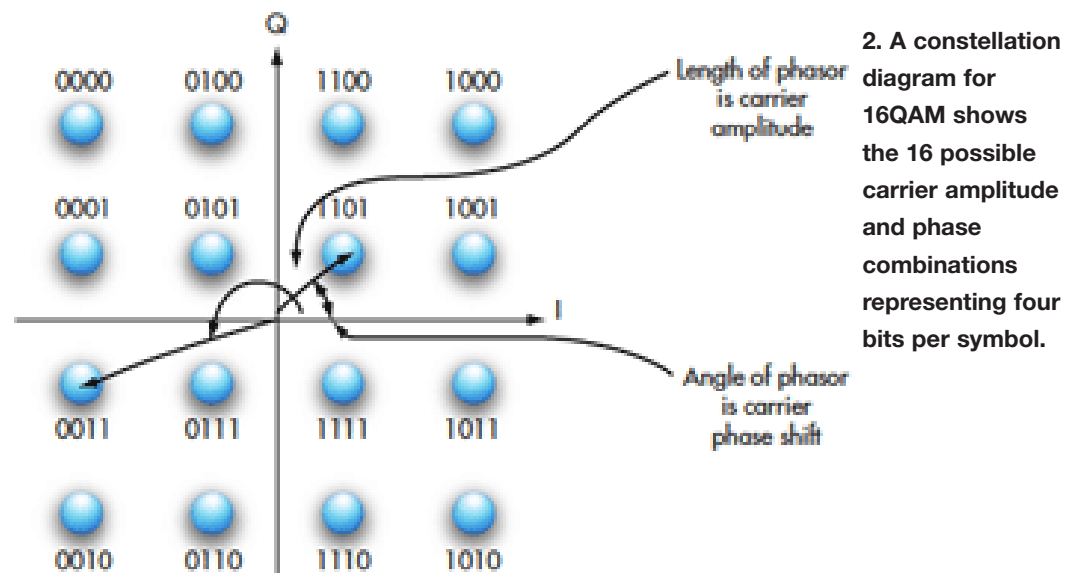
Phase-shift keying (PSK) is another popular example. When employing binary PSK, each symbol represents a 0 or 1 (see the table). A binary 0 equals 0° , while a binary 1 is 180° . With one bit per symbol, the baud and bit rates are the same. However, multiple bits per symbol can be easily implemented.

BINARY PHASE-SHIFT KEYING	
Bits	Phase shift (degrees)
00	45
01	135
11	225
10	315

For instance, in quadrature PSK there are two bits per symbol. Using this arrangement and two bits per baud, the bit rate is twice the baud rate. Other forms of PSK use more bits per baud. With three bits per baud, the modulation becomes 8PSK for eight different phase shifts representing three bits. And with 16PSK, 16 phase shifts represent the four bits per symbol.

One unique form of multilevel modulation is quadrature amplitude modulation (QAM). QAM uses a mix of different amplitude levels and phase shifts to create the symbols representing multiple bits. For example, 16QAM encodes four bits per symbol. The symbols are a mix of different amplitude levels and different phase shifts.

A constellation diagram is typically used to illustrate the amplitude and phase conditions of the carrier for each 4-bit code (Fig. 2). Each dot represents a specific carrier amplitude and phase shift. A total of 16 symbols encodes four bits per symbol, ultimately quadrupling the bit rate over the baud rate.



Why Multiple Bits Per Baud?

By transmitting more than one bit per baud, higher data rates can be transmitted in a narrower channel. Recall that the maximum possible data rate is determined by the bandwidth of the transmission channel.

Assuming a worst case of alternating 1s and 0s of data, the maximum theoretical bit rate C for a given bandwidth B is:

$$C = 2B$$

Or the bandwidth for a maximum bit rate is:

$$B = C/2$$

Transmitting a 1-Mbit/s signal requires:

$$B = 1/2 = 0.5 \text{ MHz or } 500 \text{ kHz}$$

When using multilevel modulation with multiple bits per symbol, the maximum theoretical data rate is:

$$C = 2B \log_2 N$$

Here, N is the number of symbols per symbol interval:

$$\log_2 N = 3.32 \log_{10} N$$

The bandwidth needed with a specific number of different levels for a desired speed is calculated as:

$$B = C/2 \log_2 N$$

For instance, the bandwidth needed to get a 1-Mbit/s data rate with two bits per symbol and four levels can be determined with:

$$\log_2 N = 3.32 \log_{10}(4) = 2$$

$$B = 1/2(2) = 1/4 = 0.25 \text{ MHz}$$

The number of symbols needed to get a desired data rate in a fixed bandwidth can be calculated as:

$$\log_2 N = C/2B$$

$$3.32 \log_{10} N = C/2B$$

$$\log_{10} N = C/2B = C/6.64B$$

Then:

$$N = \log^{-1}(C/6.64B)$$

Using the previous example, the number of symbols needed to transmit 1 Mbit/s in a 250-kHz channel is calculated as:

$$\log_{10} N = C/6.64B = 1/6.64(0.25) = 0.602$$

$$N = \log^{-1}(0.602) = 4 \text{ symbols}$$

These calculations assume a noise-free channel. Factoring in the noise requires the well-known Shannon-Hartley law:

$$C = B \log_2 (S/N + 1)$$

C is the channel capacity in bits per second and B is the bandwidth in hertz. S/N is the signal-to-noise power ratio.

In terms of common logarithms:

$$C = 3.32B \log_{10}(S/N + 1)$$

What is the maximum rate in a 0.25-MHz channel with a 30-dB S/N ? The 30 dB translates to a 1000 to 1 S/N . Therefore, the maximum rate is:

$$C = 3.32B \log_{10}(S/N + 1) = 3.32(.25) \log_{10}(1001) = 2.5 \text{ Mbits/s}$$

The Shannon-Hartley law doesn't specifically state that multilevel modulation must be employed to achieve that theoretical result. Using the previous procedure will reveal how

many bits per symbol are required:

$$\log_{10} N = C/6.64B = 2.5/6.64(0.25) = 1.5$$

$$N = \log^{-1}(1.5) = 32 \text{ symbols}$$

Using 32 symbols implies five bits per symbol ($2^5 = 32$).

Baud Rate Examples

Virtually all high-speed data connections use some form of broadband transmission. Wi-Fi wireless takes advantage of QPSK, 16QAM, and 64QAM in the orthogonal frequency-division multiplex (OFDM) modulation schemes. The same is true for WiMAX and Long-Term Evolution (LTE) 4G cellular technology. Cable TV and its high-speed Internet access exploit 16QAM and 64QAM to deliver analog and digital TV, while satellites use QPSK and various versions of QAM.

Land mobile radio (LMR) systems for public safety recently adopted standards for voice and data 4FSK modulation. This “narrowbanding” effort is designed to reduce the bandwidth needed from 25 kHz per channel to 12.5 kHz, and eventually 6.25 kHz. As a result, there will be more channels for additional radios without increasing the spectrum allocations.

U.S. high-definition TV employs a modulation method called eight-level vestigial sideband, or 8VSB. This method uses three bits per symbol for eight amplitude levels, which enables the transmission of 10,800 symbols/s. At 3 bits per symbol, that represents a gross bit rate of $3 \times 10,800 = 32.4$ Mbits/s. When combined with the VSB, which only transmits one full sideband and a vestige of another, high-definition video and audio can be transmitted in a 6-MHz-wide TV channel.

References

Frenzel, Louis E., *Principles of Electronic Communication Systems*, McGraw-Hill, 2008.

Gibson, Jerry D., *The Communications Handbook*, CRC Press/IEEE Press, 1997.

Sklar, Bernard, *Digital Communications, Fundamentals and Applications*, Prentice-Hall, 2001.

to view this article online,  [click here](#)

 **BACK TO TABLE OF CONTENTS**

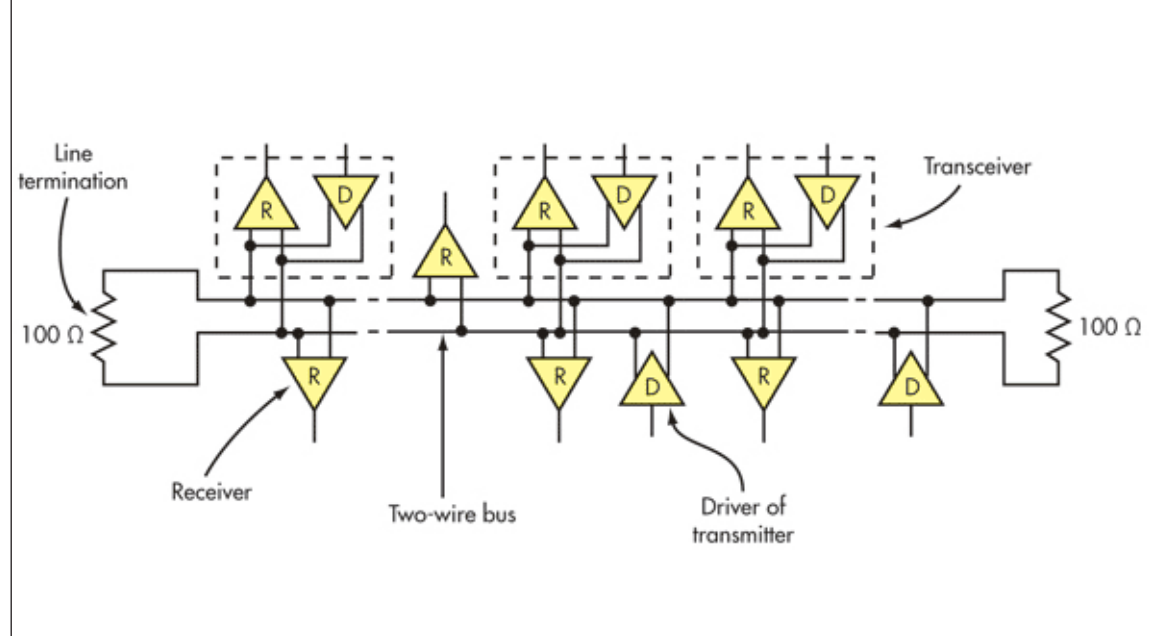


Image credit: Thinkstock

CHAPTER 2:

What's The Difference Between The RS-232 And RS-485 Serial Interfaces?

LOU FRENZEL, Technical Contributing Editor

Serial interfaces are used in most electronic equipment today. Some of the original serial interfaces like RS-232 and RS-485 are still widely used. This article summarizes these interfaces and shows where they are still used today.

Dozens of serial data interfaces are used today. Most have been developed for specific applications. A few have become universal, such as I²C, CAN, LIN, SPI, Flex, MOST, and I²S. Then there's Ethernet and USB and other higher-speed serial interfaces like FireWire, HDMI, and Thunderbolt. Two of the oldest interfaces are RS-232 and RS-485. These legacy interfaces aren't obsolete or discontinued, though. Both are still alive and well in many applications.

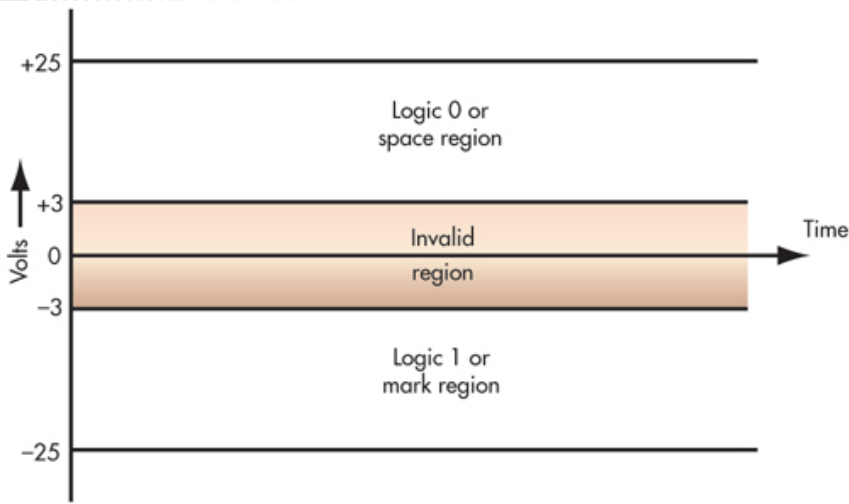
The whole purpose of a serial interface is to provide a single path for data transmission wirelessly or over a cable. Parallel buses are still used in some applications. But with high-speed data so common today, a serial interface is the only practical option for communications over any distance greater than several feet.

Serial interfaces can be used to provide standardized logic levels from transmitters to receivers, define the transmission medium and connectors, and specify timing and data rates. In some cases, they can perform serial-to-parallel and parallel-to-serial conversion or specify a basic data protocol.

The definition of logic levels, medium, and connectors is part of the physical layer (PHY) or layer 1 of the Open Systems Interconnection (OSI) networking model. Any additional functions such as data handling is part of the media access control (MAC) layer or layer 2 of the OSI model.

RS-232

One of the oldest serial interfaces is generically called RS-232. It was originally established in 1962 as a method of connecting data terminal equipment (DTE) such



1. Voltage levels define the logic 1 or mark and the logic 0 or space characters. Voltages between ± 3 V are invalid.

the same as the International Telecommunications Union – Telecommunications (ITU-T) specifications V.24 and V.28.

The standard defines a logic 1 and a voltage between -3 and -25 V and a logic 0 as a voltage level between $+3$ and $+25$ V (Fig. 1). Signal levels are commonly referred to as a mark for logic 1 and a space for logic 0. Voltages between ± 3 V are invalid, providing a huge noise margin for the interface. Noise voltages in this range are rejected. In common practice, logic 0 and 1 levels are typically as low as ± 5 V and as high as ± 12 or ± 15 V. The transmitter and receiver configurations are single-ended (not differential) with a ground reference.

The cable medium can be simple parallel wires or twisted pair. The length of the cable determines the upper data rate and generally should not exceed 50 feet. However, much longer cable lengths can be used with low data rate conditions. Today the primary goal is to use a cable with no more than 2500 pF of capacitance between wires. This limits the upper data rate to roughly 20 kbits/s. Because of the low-speed data rates used with this interface, the cable generally isn't treated as a formal transmission line. Transmission lines require matched generator and load impedances to eliminate reflections that cause data corruption.

The standard defined a 25-pin connector called the DB-25, which was designed to carry a variety of control lines as well as the serial data transmit and receive lines. Such connectors are rarely used today. Instead, a nine-pin connector called the DE-9 was defined, and today it is the de facto standard (Fig. 2).

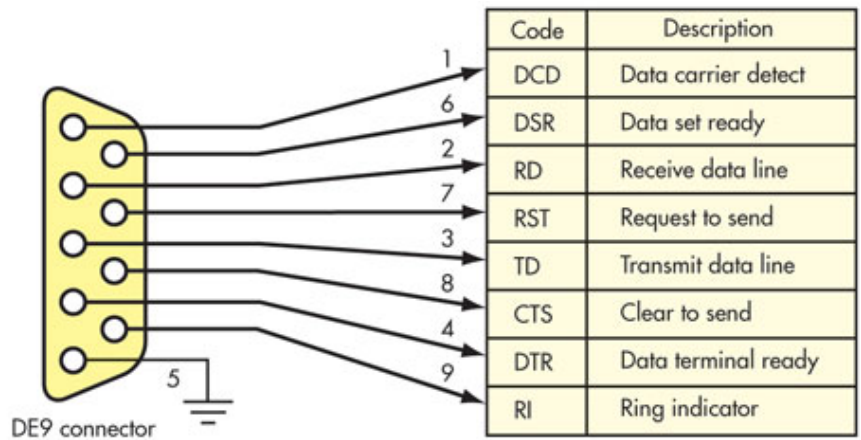
Originally, data rates for electromechanical equipment were very slow. A minimum rate was typically 75 bits/s, but rates of 150 and 300 bits/s were common. Today, data rates are defined by the protocol used with the interface and can range as high as 115.2 kbits/s. Typical data rates are 1200, 2400, 4800, 9600, 19,200, 38,400, and 115,200 bits/s. The data rate is limited by the maximum allowed slew rate of 30 V/ μ s (volts per microsecond). For short low-capacitance cables, data rates can be as high as several megabits per second with the appropriate drivers.

Many RS-232 connections are one-way or simplex. However, using the special signaling and control voltages available, two-way or half-duplex operation is possible. The two connected devices alternate transmitting and receiving operations.

as electromechanical teletypewriters to data communications equipment (DCE). Over the years its use has included connections to video terminals, computers, and modems. The first personal computers included an RS-232 called a serial port for connection to a printer or other peripheral device. Today, it is still widely used in embedded computer development systems, scientific instruments, and all sorts of industrial control equipment.

The official name of the standard is Electronic Industries Association/Telecommunications Industries Association EIA/TIA-232-F. The letter F designates the most recent standard modification and update. The standard is essentially

2. The popular DB9 connector carries the signals shown. The numbers are the pin numbers on the connector.



The control signals in the interface define the protocol for transmitting and receiving data. These signals tell the two communicating devices when they are busy, transmitting, ready, and receiving. The transmitting device is the DTE such as a computer, and the receiving device is the DCE such as a printer. The control signals used on the common nine-pin connector are:

- Data carrier detect (DCD): The DCE tells the DTE it is receiving a valid input signal.
- Data set ready (DSR): The DCE tells the DTE it is connected and ready to receive.
- Received data (RD): This is the actual signal received from the DTE.
- Request to send (RTS): This signal from the DTE tells the DCE it is ready to transmit.
- Transmit data (TD): This is the transmitted signal from the DTE.
- Clear to send (CTS): This line from the DCE tells the DTE it is ready to receive data.
- Data terminal ready (DTR): This line is from the DTE to the DCE indicating readiness to send or receive data.
- Ring indicator (RI): This line was used in older modem connections but isn't used anymore.
- Signal ground: This is the common ground connection for all signals.

Figure 3 shows the cable connections from the DTE to the DCE. Note the interconnections between the control line pins. The signals on these pins occur in response to one another in what is called a flow control or “handshaking” process.

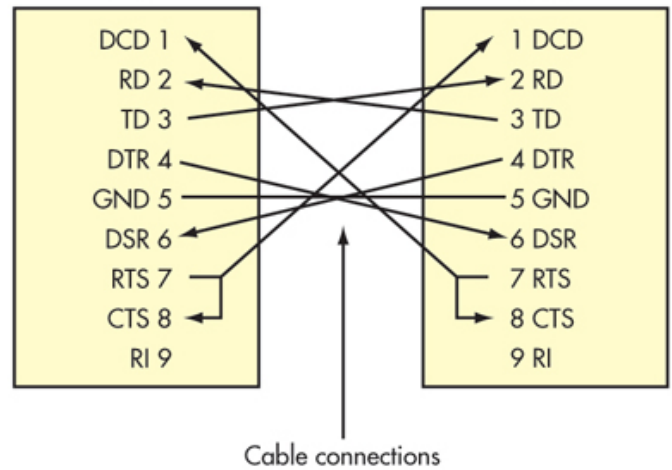
Although not formally part of the RS-232 standard, most serial devices using the interface also use what is called a universal asynchronous receiver transmitter (UART). This IC, usually separate from the line driver and receiver circuits, implements a basic communications protocol that involves transmitting up to 8 bits at a time. It performs serial-to-parallel and parallel-to-serial conversion, adding start and stop bits to signal the beginning and end of a data word, parity bit error detection, and establishment of the data rate.

The data is often ASCII characters, but any data word up to 8 bits can be transmitted (**Fig. 4**). The UART can usually be configured to handle different word sizes (5 to 8 bits), add 1, 1.5, or 2 stop bits, and include odd, even, or no parity bits. Data rates from 75 bits/s to 115.2 kbits/s are selectable.

RS-485

Also defined by the EIA/TIA standard, this interface is now called TIA-485. It defines not

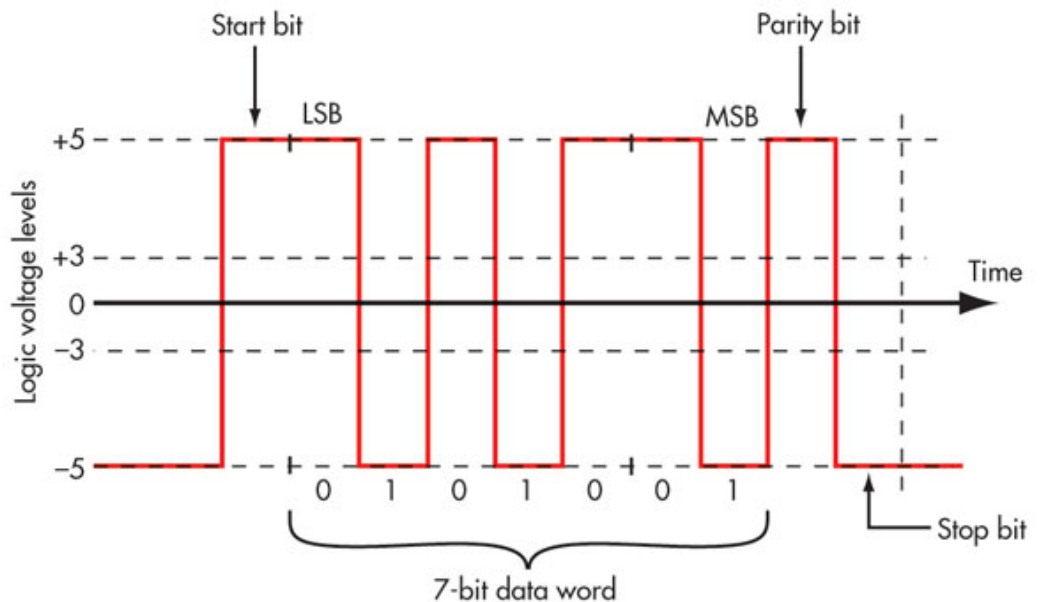
3. This is a common connection between the DTE and the DCE devices. Note the connections in the cable from one connector to the other.



only a single device-to-device interface but also a communications bus that can be used to form simple networks of multiple devices. Its configuration and specifications also extend the range and data rate beyond the RS-232 interface capabilities.

The RS-485 standard specifies differential signaling on two lines rather than single-ended with a voltage referenced to ground. A logic 1 is a level greater than -200 mV, and a logic 0 is a level greater than $+200$ mV. Typical line voltage levels from the line drivers are a minimum of ± 1.5 V to a maximum of about ± 6 V. Receiver input sensitivity is ± 200 mV. Noise in the range of ± 200 mV is essentially blocked. The differential format produces effective common-mode noise cancellation.

The standard transmission medium is twisted-pair cable of either #22 or #24 AWG solid wire. Two lines are minimum but a third reference wire can be used. Four-wire cables can also be used if full-duplex operation is desired. The cables may be shielded or unshielded, with unshielded the most common. The cable is treated as a transmission line. The



4. This is the EIA/TIA-232 signal for transmitting a 7-bit ASCII capital letter J. A start bit signals the beginning of the character. The LSB is transmitter first. An odd parity bit is included. The transmission ends with a stop bit.

nominal characteristic impedance is 100 or 120 Ω . Terminating load resistors are required to ensure a matched line condition, which prevents reflections that introduce data errors.

The standard does not define specific connectors. Various connection methods have been used, including the RS-232 DE-9 connector. Simple screw terminal connections are common in some types of industrial control equipment.

Cable length defines the upper data rate. But because of the lower logic voltage levels and the differential connection, data rates can exceed 10 Mbits/s depending on cable length. Maximum cable length is commonly defined as 1200 meters or about 4000 feet. The typical maximum data rate at 4000 feet is 100 kbits/s. A general guideline is that the product of the length of the line in meters and the data rate in bits per second should not exceed 108. A 20-meter cable, for example, would allow a maximum data rate of 5 Mbits/s.

The RS-485 interface can be used in simplex or half-duplex modes with a single-pair cable. Full-duplex or simultaneous transmit and receive operations can be implemented with a two-pair cable. A common configuration is bus network with multiple drops or connections. The standard specifies a maximum of 32 drivers (transmitters) and 32 receivers (Fig. 5). Line drivers are disconnected from the line when not transmitting. All receivers are fully connected, and the bus line is terminated in a load matching resistance.

The standard has not defined a specific communications protocol. The standard UART protocol is sometimes used. Most applications define a unique protocol.

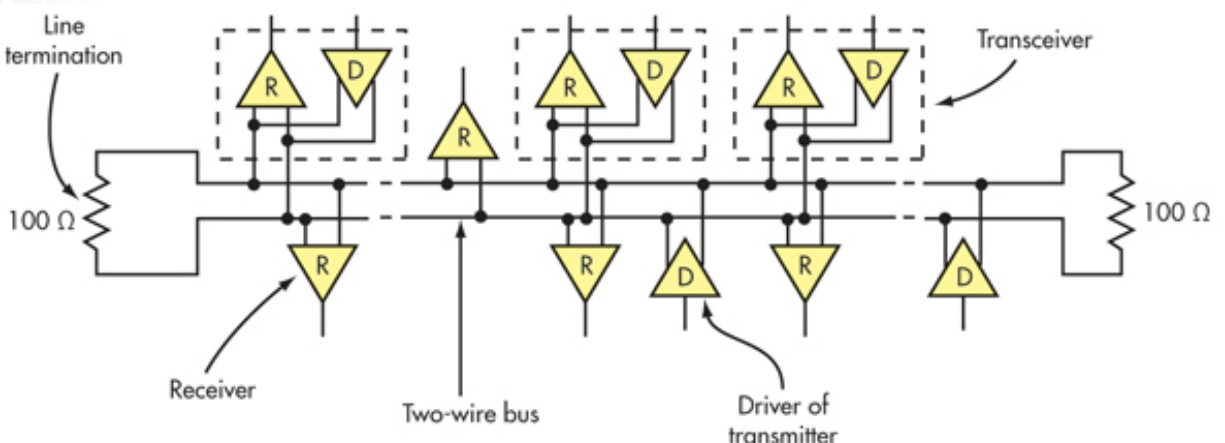
Interface Variations

Several variations of these two standards are occasionally found in practice. RS-422 is a variation of RS-485 with similar specifications but is designed only for one driver and up to 10 receivers. Logic levels range from ± 2 to ± 6 V. RS-423 is a single-ended rather than differential variation of RS-422. Otherwise, other specifications are similar to the RS-485 interface.

Applications

The TIA-232 standard is currently deployed in a wide range of low-data-rate short-range applications. It is particularly effective in equipment used in noisy environments such as factories, process control plants, and utilities sites. Cable lengths are typically less than 50 feet. Common equipment includes low-speed modems, industrial control equipment like programmable logic controllers (PLCs), computer numerically controlled (CNC) machine

5. This is a representation of a typical TIA-485 differential bus showing individual drivers (D) and receivers (R) as well as transceivers. Note the end of bus termination resistors.



tools, robots, embedded control computers, medical instruments and equipment, and embedded controller development systems.

The TIA-485 interface is also widely used in industrial applications where higher speeds and longer distances are needed. It is used in the same type of equipment as defined for the RS-232 interface plus devices like point of sale (POS) terminals, metering instruments, and large special automated machines. Networks defined by field buses like Profibus and Modbus use it as well.

Most new equipment uses the popular USB interface. However, it is often necessary or desirable to convert from one interface to another to allow different types or ages of equipment to be used together. Various converters are available to convert USB to TIA-232 or TIA-485, TIA-232 to TIA 485, or vice versa.

Summary

In general, the RS-232 is best for short-distance low-speed requirements. It is simple and low cost, and plenty of components like line drivers and receivers, UARTs, and connectors are available to build the interface. Some drivers permit data rates to 2 or 3 Mbits/s for short lines. Most interface chips include a charge pump dc-dc converter that supplies the positive and negative supply voltages allowing the IC to operate from a single 3.3- or 5-V supply.

The RS-485 is for higher speeds over longer ranges or if duplex networking capability is required. Again, many standard parts are available to create the interface. ICs have drivers and receivers that can permit more than 32 drops, and data rates can be as high as 40 Mbits/s. Some ICs also include dc-dc converters to furnish dual supply voltages (**see the table**).

to view this article online,  [click here](#)

 **BACK TO TABLE OF CONTENTS**

KEY CHARACTERISTICS OF THE RS-232 AND RS-485 SERIAL INTERFACES

Parameter	RS-232	RS-485
Line configuration	Single-ended	Differential
Mode of operation	Simplex or full duplex	Simplex or half duplex
Maximum cable length	50 feet	4000 feet
Maximum data rate*	20 kbits/s	10 Mbits/s
Typical logic levels	± 5 to ± 15 V	± 1.5 to ± 6 V
Minimum receiver input impedance	3 to 7 k Ω	12 k Ω
Receiver sensitivity	± 3 V	± 200 mV

* Maximum rate at maximum cable length

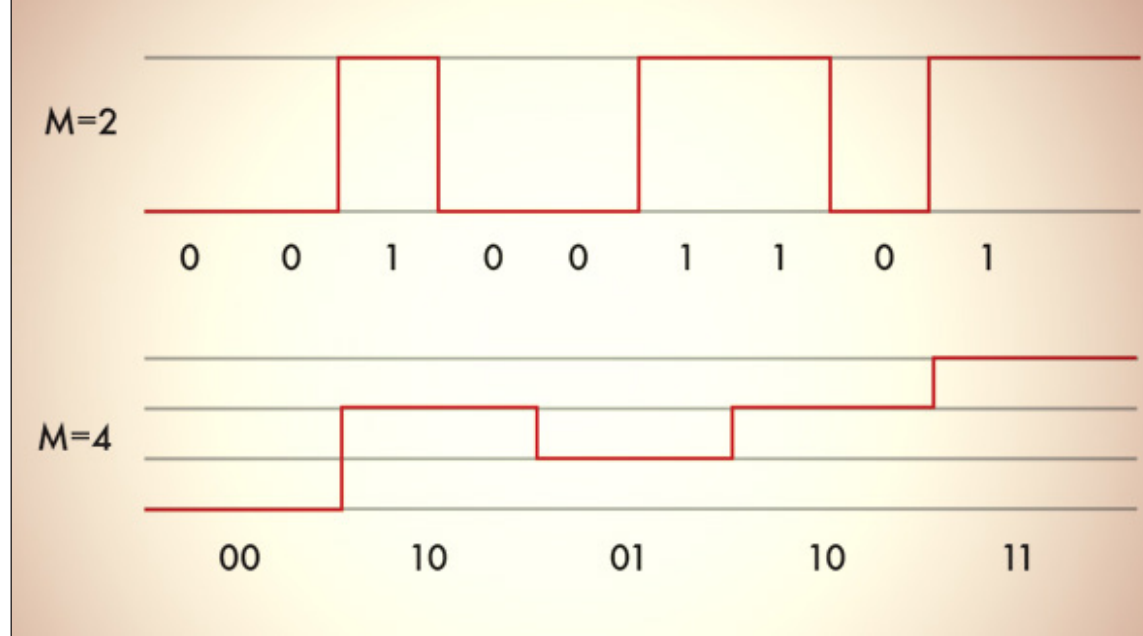


Image credit: Thinkstock

CHAPTER 3:

What's the Difference Between NRZ and PAM?

WILLIAM G. WONG, Senior Content Director, *Electronic Design and Microwaves & RF*

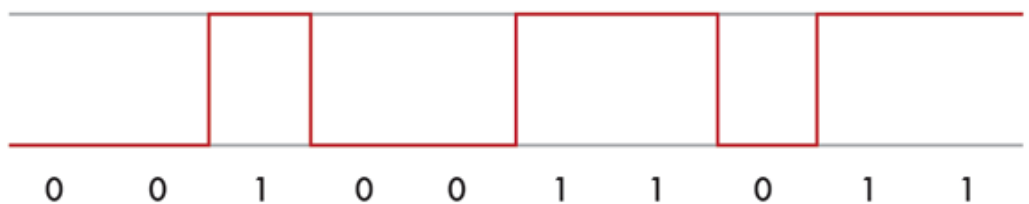
PAM-4 may be used to reach 400-Gb/s Ethernet, as well as being a good fit for other high-speed serial interfaces like Fibre Channel. So what's the difference between the usual non-return to zero (NRZ) and pulse amplitude modulation (PAM)?

PAM-4, which could be a catalyst for reaching 400-Gb/s Ethernet, will also likely find homes in high-speed serial interfaces such as FibreChannel. Pulse amplitude modulation (PAM) is a way to pack more bits into the same amount of time on a serial channel.

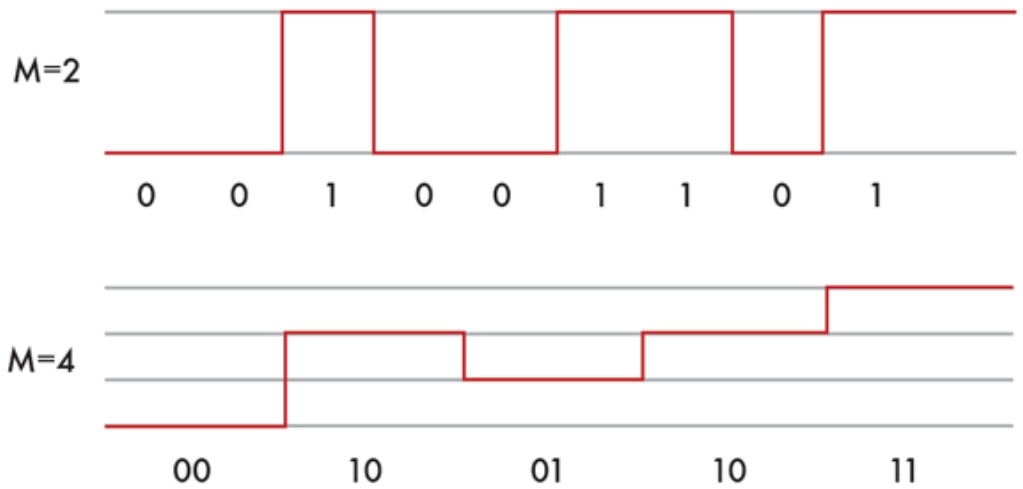
The usual mechanism used in serial communications is the binary non-return to zero (NRZ). NRZ tracks the values being sent; therefore, an idle state, where all the bits are the same value, leaves the signal at the same level during the idle time (**Fig. 1**). NRZ differs from return to zero (RZ), which transitions to the 0 level for each bit. RZ is self-clocking, but it requires more bandwidth.

An alternative to NRZ is NRZ Inverted (NRZI), in which a 1 bit is indicated by a transition from one level to another. An idle using a 1 value will result in the signal cycling each bit.

NRZ and NRZI aren't inherently self-clocking, but there are ways to address this issue. One is to use bit stuffing, which forces a transition after a set number of bits. Another approach is to only allow particular patterns that will have sufficient transitions to maintain an accurate clock at the receiver. Manchester encoding is one example of the latter. Ethernet, for instance, employs an 8B/10B Manchester code that encodes 8 bits of data



1. NRZ transmissions are easy to understand: 0 is a low level and 1 is a high level.



2. A 4-level MLS, also known as PAM-4, has four distinct levels to encode two bits of data.

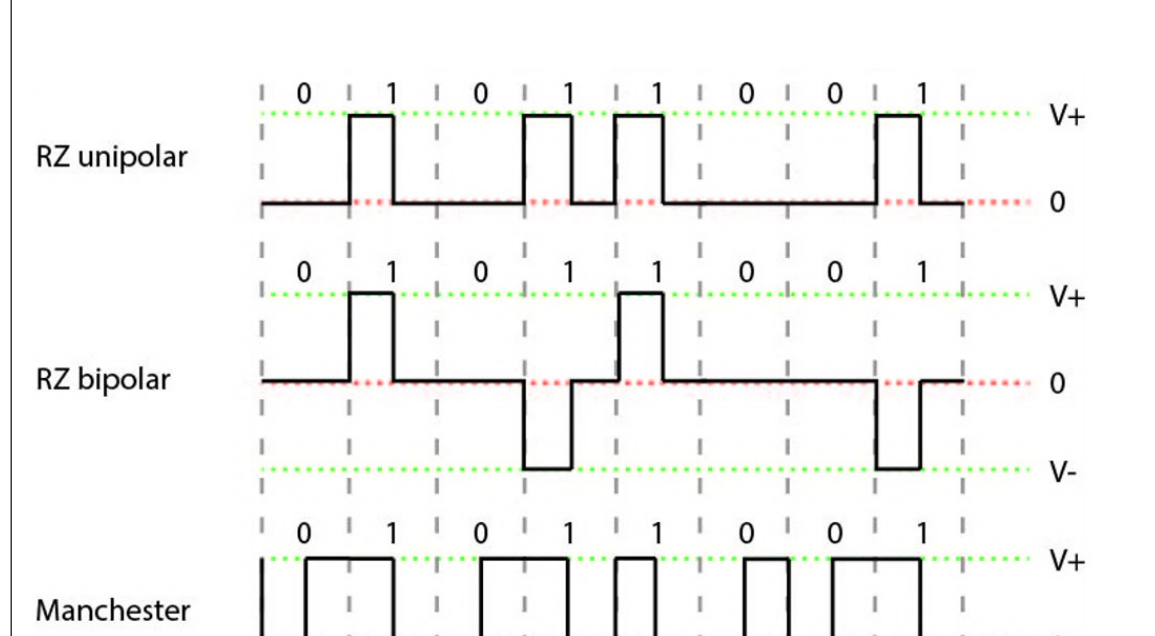
using 10-bit codes. The 10-bit codes are chosen to give the data characteristics necessary for clock recovery from the data.

Encoding more data into the same timeframe can be done using different signaling levels. Such multi-level signaling (MLS) or pulse amplitude modulation (PAM) can have multiple distinct levels. Actually, NRZ is a two-level MLS or PAM-2 system. PAM-4 (**Fig. 2**) has four distinct levels to encode two bits of data, essentially doubling the bandwidth of a connection.

Of course, there's no such thing as a free lunch. Generating or decoding more than two levels is typically more difficult, and often requires better or more complex hardware. Likewise, for high-speed signals, random and induced noise becomes a significant factor. The approach begins to pay off when there's an escalation in cabling costs. That's the hope anyway when it comes to 400 Gigabit Ethernet, which will likely employ eight PAM-4 connections running at 28 GHz.

to view this article online,  [click here](#)

 **BACK TO TABLE OF CONTENTS**



CHAPTER 4:

What's the Difference Between NRZ, NRZI, and Manchester Encoding?

WILLIAM G. WONG, Senior Content Director, *Electronic Design* and *Microwaves & RF*

NRZ, NRZI, and Manchester are popular serial encoding mechanisms. Find out how they differ from each other.

Non-return-to-zero (NRZ), non-return-to-zero-inverted (NRZI), return-to-zero (RZ), and Manchester are popular serial encoding mechanisms (see figure). Each has different characteristics that make it useful in different applications. Those dealing with UARTs will find NRZ to be the most familiar.

There is also a unipolar and bipolar version for many of these encodings. Unipolar has a transition between a zero and positive. The actual measurement can be one of many types of attributes from like voltage, current, pressure, or optical. A bipolar system has a transition between a positive and negative. Any method can employ a bipolar encoding but logically they may be the same as shown with the NRZ example.

The figure shows signals synchronized with a clock at the bottom. The types and locations of the transitions are what make the encoding mechanisms different.

NRZ has a level that matches the logical signal. In this case the clock would latch the value being transmitted at the start of each clock cycle and it would be sampled at the receiving end in the middle of the clock cycle. NRZ is commonly used with serial ports. The receiver normally has a faster clock that is synchronized with a rising or falling edge of the data transmission. This allows the determination of the center of the clock cycle to be more accurately determined (Fig. 1).

The clock highlights the different transition methods. NRZ is level sensitive related to the values being sent. RZ has a transition for a 1 value with respect to the clock. Manchester and NRZI have transitions based on the value. Manchester has a falling transition for a 1, while NRZI has a transition for a 1.

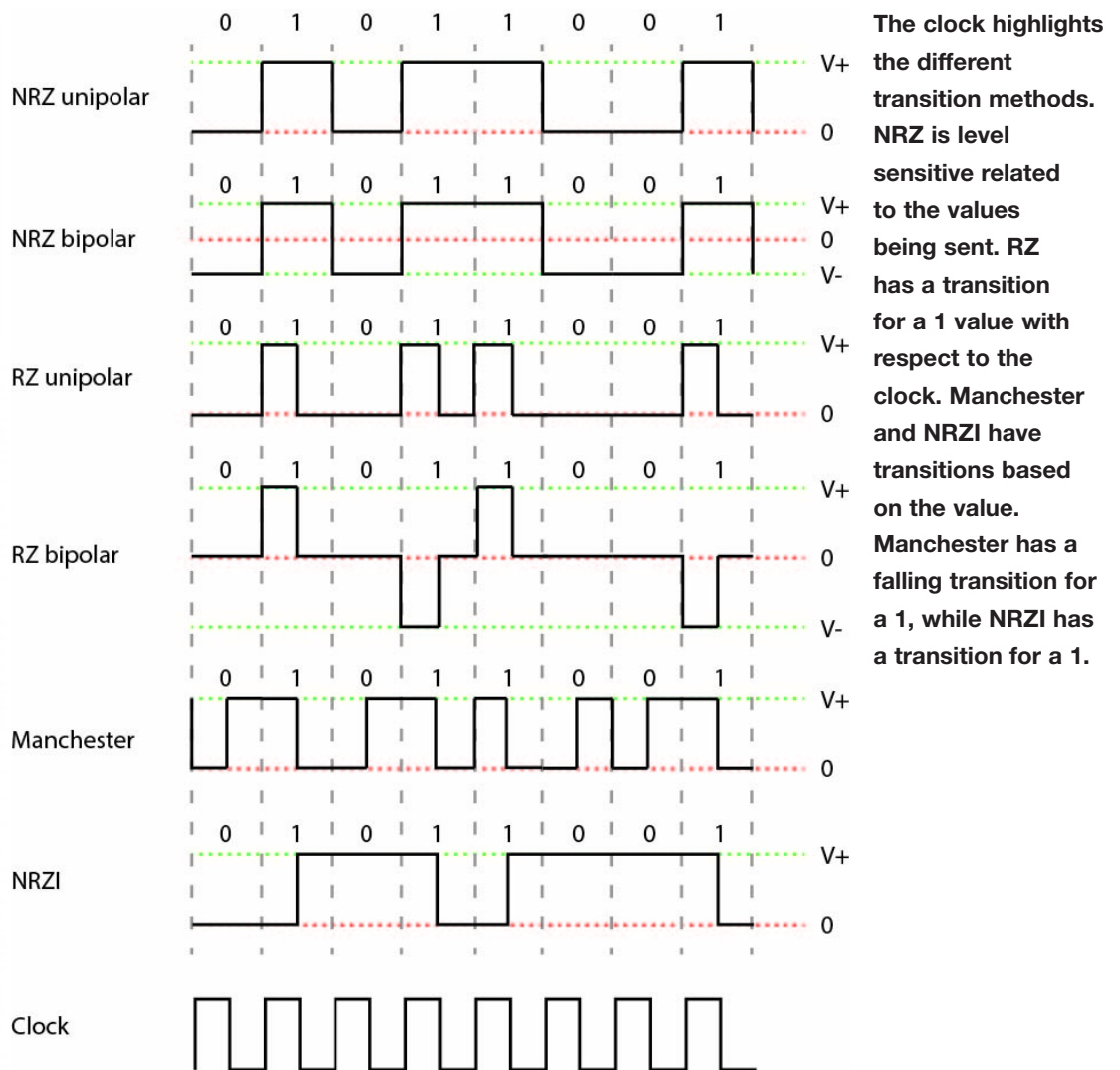
RZ uses pulses at the start of the clock cycle to indicate a 1 value. In the bipolar version,

sequential 1 values have a transition in the opposite directions. RZ will always have a transition when there are 1 values in the data.

Manchester and NRZI encodings have transitions in the middle of the clock cycle, with the type depending on the encoding. Manchester is an NRZ encoding that is exclusive-ORed with the clock. This provides at least one transition per bit. NRZI also uses a transition in the middle of the clock cycle, but this only occurs when there is a 1 value. Manchester makes clock recovery easier.


The challenge with Manchester and RZ is the frequency of the data transmission is twice that as NRZ and NRZI. The tradeoffs depend upon the mechanisms used to implement the communication. Sometimes bandwidth can be an issue.

These encoding schemes are just the starting point used for most data transmission systems. For example, universal asynchronous receiver-transmitters (UART) map an asynchronous per/byte sequence that includes a start bit and one or more stop bits. A parity bit can be included to assist in error detection. Higher level protocols often forego the parity bit and compute parity or a CRC over a data packet. This helps reduce the overhead for error detection.

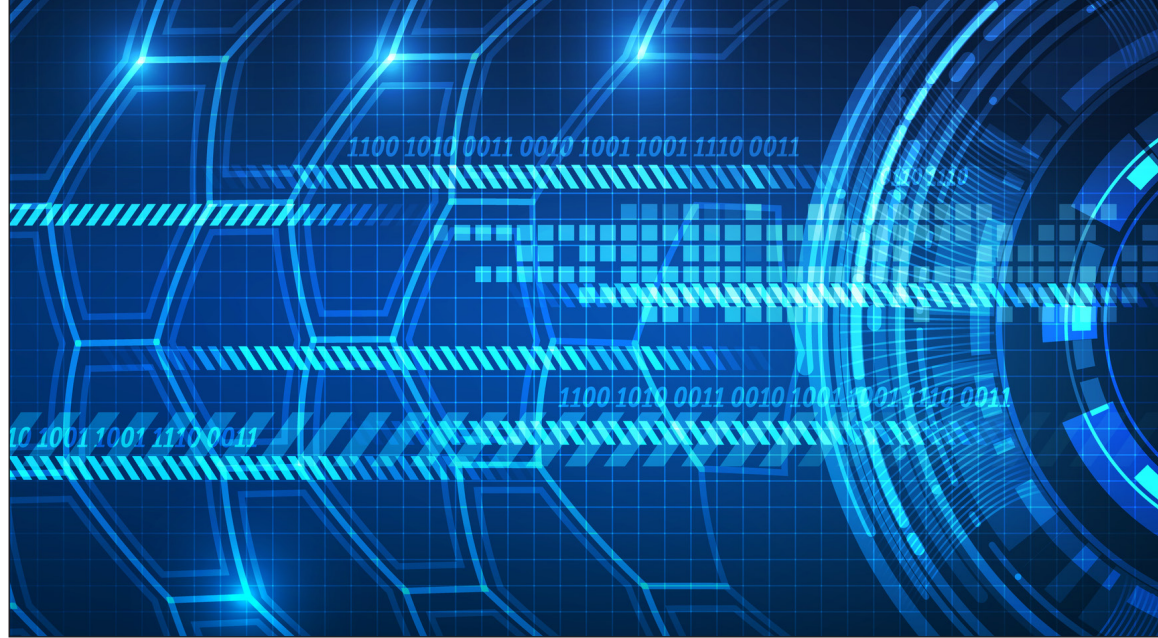


Synchronous systems typically have an idle character that can be recognized in addition to a clock recovery/synchronization scheme. Encoding schemes like Manchester encoding make this easier to implement. The synchronous support often uses encoding schemes like 4B/5B and 8B/10B. The 4B/5B uses 5 bits to encode 16 (4-bit) symbols that are unique enough that clock recovery/synchronization is possible using encodings like NRZ.

There is a bandwidth penalty with these higher level encoding schemes. The larger the collection, the lower the overhead but they require tighter timing and more complex synchronization logic. The approach scales and 64B/66B is used with 10 Gbit Ethernet.

to view this article online,  [click here](#)

 **BACK TO TABLE OF CONTENTS**



Mobile networks take advantage of two similar, but different, forms of duplexing to send and receive data quickly and efficiently.

CHAPTER 5:

What's The Difference Between FDD And TDD?

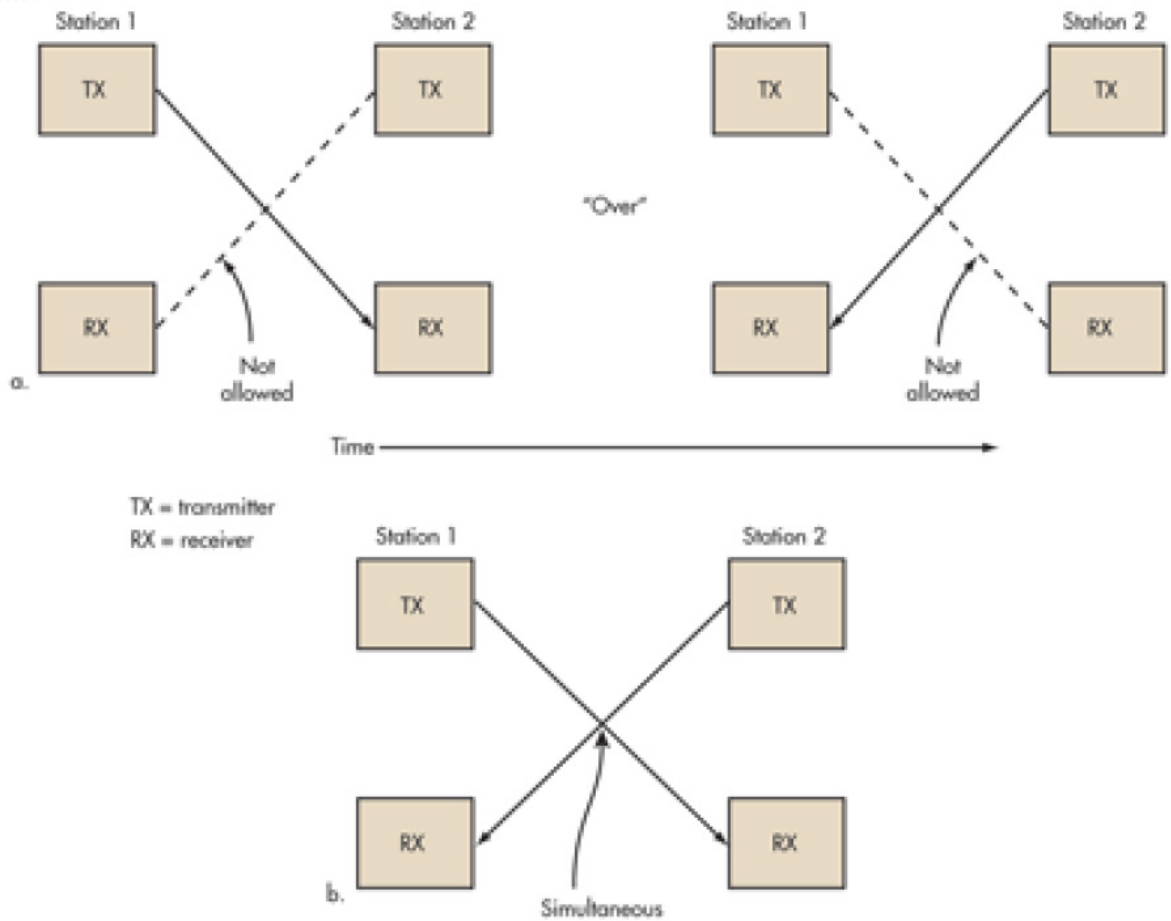
LOU FRENZEL, Technical Contributing Editor

Duplexing is the process of achieving two-way communications over a communications channel. It takes two forms: half duplex and full duplex (**Fig. 1**).

In half duplex, the two communicating parties take turns transmitting over a shared channel. Two-way radios work this way. As one party talks, the other listens.

Speaking parties often say “Over” to indicate that they’re finished and it’s time for the other party to speak. In networking, a single cable is shared as the two computers

FDD AND TDD METHODS		
Characteristic	FDD	TDD
Spectrum usage	High including guard bands	Less
Complexity	High	Low but needs accurate timing
Cost	Higher	Lower
Latency	Little or none	Depends on range, TX-RX switching times
Range	Unlimited	Shorter, depends on guard time
UL/DL symmetry	Usually 50/50	Asymmetrical as required
Dynamic bandwidth allocation	None	Can be implemented
MIMO and beamforming	More difficult	Easier



1. There are two modes of duplex operation: (a) half duplex and (b) full duplex.

communicating take turns sending and receiving data.

Full duplex refers to simultaneous two-way communications. The two communicating stations can send and receive at the same time. Landline telephones and cell phones work this way. Some forms of networking permit simultaneous transmit and receive operations to occur. This is the more desirable form of duplexing, but it is more complex and expensive than half duplexing. There are two basic forms of full duplexing: frequency division duplex (FDD) and time division duplex (TDD) (**see the table**).

A duplex system is generally used to refer to a two-way communication system such as a mobile network, a radio channel, or a computer network connection. Think of it as a system where two-way communication is possible, but not necessarily taking place at the same time. For example, a one-way street can be considered a duplex system, as it allows traffic to flow in both directions, but not at the same time.

Duplex systems are utilized in many communications networks, which allow for simultaneous communication in both directions between two connected parties, or to provide a reverse path for monitoring or adjusting equipment deployed in the field. To that end, there are two commonly used platforms for duplexing, including FDD and TDD.

FDD is a duplexing method used in cellular networks where the downlink and uplink frequencies are different. TDD is a duplexing method used in cellular networks where the downlink and uplink frequencies are the same, but the time slots for each are different.

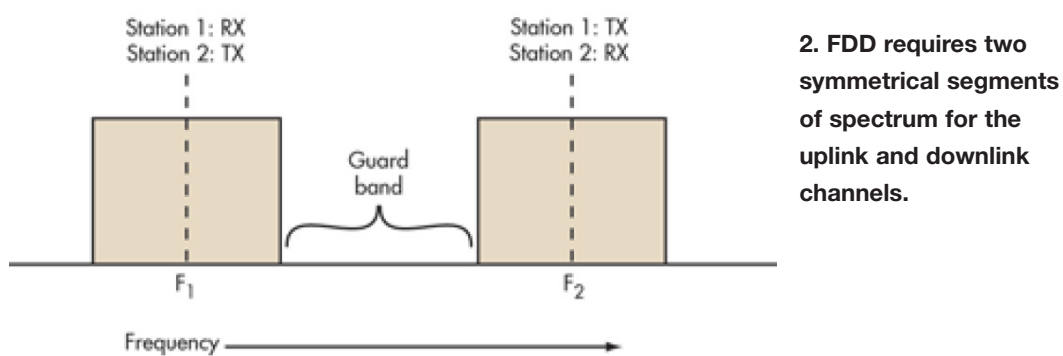
Frequency Division Duplex

FDD requires two separate communications channels. In networking, there are two cables. Full-duplex Ethernet uses two twisted pairs inside the CAT5 cable for simultaneous send and receive operations.

Wireless systems need two separate frequency bands or channels (**Fig. 2**). A sufficient amount of guard band separates the two bands so the transmitter and receiver don't interfere with one another. Good filtering or duplexers and possibly shielding are a must to ensure the transmitter does not desensitize the adjacent receiver.

In a cell phone with a transmitter and receiver operating simultaneously within such close proximity, the receiver must filter out as much of the transmitter signal as possible. The greater the spectrum separation, the more effective the filters.

FDD uses lots of frequency spectrum, though, generally at least twice the spectrum



needed by TDD. In addition, there must be adequate spectrum separation between the transmit and receive channels. These so-called guard bands aren't useable, so they're wasteful. Given the scarcity and expense of spectrum, these are real disadvantages.

However, FDD is very widely used in cellular telephone systems, such as the widely used GSM system. In some systems the 25-MHz band from 869 to 894 MHz is used as the downlink (DL) spectrum from the cell site tower to the handset, and the 25-MHz band from 824 to 849 MHz is used as the uplink (UL) spectrum from the handset to cell site.

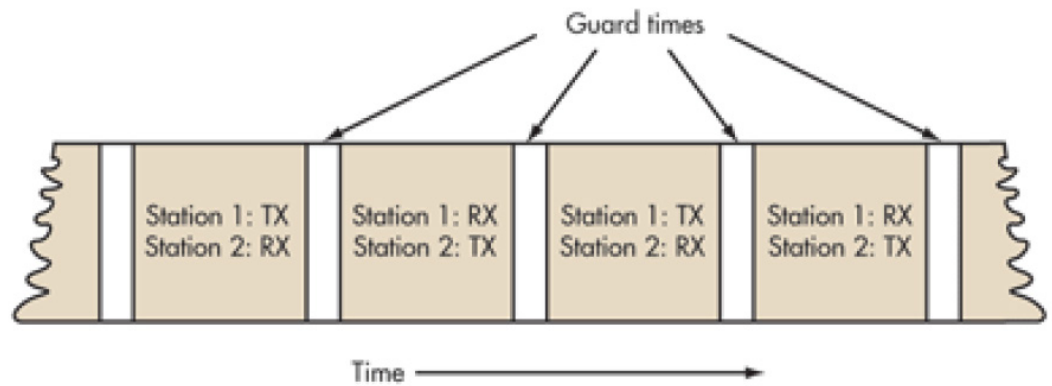
Another disadvantage with FDD is the difficulty of using special antenna techniques like multiple-input multiple-output (MIMO) and beamforming. These technologies are a core part of the new Long-Term Evolution (LTE) 4G cell phone strategies for increasing data rates. It is difficult to make antenna bandwidths broad enough to cover both sets of spectrum. More complex dynamic tuning circuitry is required.

FDD also works on a cable where transmit and receive channels are given different parts of the cable spectrum, as in cable TV systems. Again, filters are used to keep the channels separate.

Time Division Duplex

TDD uses a single frequency band for both transmit and receive. Then it shares that band by assigning alternating time slots to transmit and receive operations (**Fig. 3**). The information to be transmitted—whether it's voice, video, or computer data—is in serial binary format. Each time slot may be 1 byte long or could be a frame of multiple bytes.

Because of the high-speed nature of the data, the communicating parties cannot tell that the transmissions are intermittent. The transmissions are concurrent rather than simulta-



3. TDD alternates the transmission and reception of station data over time. Time slots may be variable in length.

neous. For digital voice converted back to analog, no one can tell it isn't full duplex.

In some TDD systems, the alternating time slots are of the same duration or have equal DL and UL times. However, the system doesn't have to be 50/50 symmetrical. The system can be asymmetrical as required.

For instance, in Internet access, download times are usually much longer than upload times so more or fewer frame time slots are assigned as needed. Some TDD formats offer dynamic bandwidth allocation where time-slot numbers or durations are changed on the fly as required.

The real advantage of TDD is that it only needs a single channel of frequency spectrum. Furthermore, no spectrum-wasteful guard bands or channel separations are needed. The downside is that successful implementation of TDD needs a very precise timing and synchronization system at both the transmitter and receiver to make sure time slots don't overlap or otherwise interfere with one another.

Timing is often synched to precise GPS-derived atomic clock standards. Guard times are also needed between time slots to prevent overlap. This time is generally equal to the send-receive turnaround time (transmit-receive switching time) and any transmission delays (latency) over the communications path.

Application Examples

Most cell-phone systems use FDD. The newer LTE and 4G systems use FDD. Cable TV systems are fully FDD.


Most wireless data transmissions are TDD. WiMAX and Wi-Fi use TDD. So does Bluetooth when piconets are deployed. ZigBee is TDD. Most digital cordless telephones use TDD. Because of the spectrum shortage and expense, TDD is also being adopted in some cellular systems, such as China's TD-SCDMA and TD-LTE systems. Other TD-LTE cellular systems are expected to be deployed where spectrum shortages occur.

Conclusion

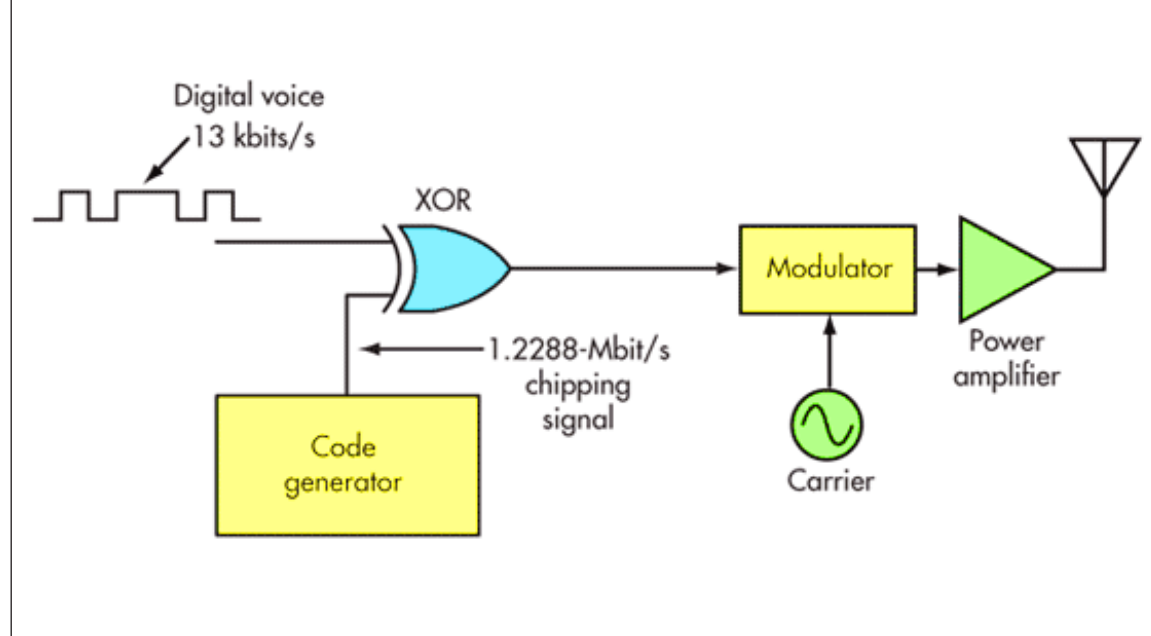
TDD appears to be the better overall choice, but FDD is far more widely implemented because of prior frequency spectrum assignments and earlier technologies. FDD will continue to dominate the cellular business for now. Yet as spectrum becomes more costly and scarce, TDD will become more widely adopted as spectrum is reallocated and repurposed.



TDD and FDD duplexing are both important methods for managing communication traffic in networks. While TDD has some advantages over FDD, such as the ability to dynamically adjust the amount of bandwidth allocated to each direction, FDD has the advantage of being more compatible with existing infrastructure. As a result, it is likely that both methods will continue to be used. That said, TDD has been proposed as a possible replacement for FDD, but it is unclear if it will be adopted on a wide scale. FDD is the more established technology and is likely to remain the dominant duplexing method in the near future.

to view this article online,  [click here](#)

 [BACK TO TABLE OF CONTENTS](#)



CHAPTER 6:

Fundamentals of Communications Access Technologies: FDMA, TDMA, CDMA, OFDMA, AND SDMA

LOU FRENZEL, Technical Contributing Editor

Can you hear me now?
Gain insight to the multiple communications access technologies, their differences and applications.

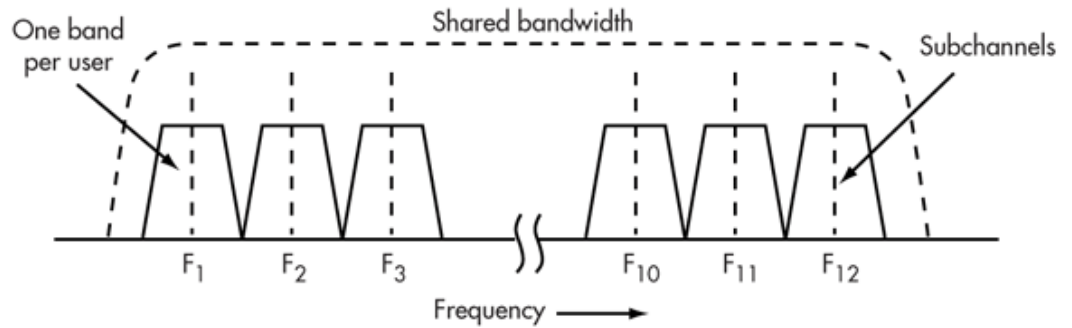
Access methods are multiplexing techniques that provide communications services to multiple users in a single-bandwidth wired or wireless medium. Communications channels, whether they're wireless spectrum segments or cable connections, are expensive. Communications services providers must engage multiple paid users over limited resources to make a profit. Access methods allow many users to share these limited channels to provide the economy of scale necessary for a successful communications business. There are five basic access or multiplexing methods: frequency division multiple access (FDMA), time division multiple access (TDMA), code division multiple access (CDMA), orthogonal frequency division multiple access (OFDMA), and spatial division multiple access (SDMA).

Each one of these takes advantage of multiplexing methods, dividing the bandwidth of the signal into different sub-bands, which are then assigned to different users in order to allow multiple users to share a single channel. Multiplexing is a communications technique that multiplexes, or combines, multiple signals into a single signal. The reverse process is called demultiplexing. For example, the voice signals of many telephone callers can be multiplexed over a single data link by using time-division multiplexing (TDM).

The multiplexing methods for FDMA, TDMA, CDMA, OFDMA, and SDMA are all generally similar. In each case, a plurality of communication channels is multiplexed together to form a single communication channel. The communication channels are then divided into a plurality of sub-channels, each of which is assigned to a different user.

Each user is then allocated a time slot in which to transmit data. In FDMA, the time slots are assigned to the users in a sequential fashion. In TDMA, the time slots are assigned to

1. FDMA divides the shared medium bandwidth into individual channels. Subcarriers modulated by the information to be transmitted occupy each subchannel.



the users in a random fashion. In CDMA, the time slots are assigned to the users based on their code sequences. In OFDMA, the time slots are assigned to the users based on their frequency offsets. In SDMA, the time slots are assigned to the users based on their location. Users can then transmit data in their assigned time slots. The data is then demultiplexed at the receive end and sent to the appropriate user.

Frequency Division Multiple Access (FDMA)

FDMA is the process of dividing one channel or bandwidth into multiple individual bands, each for use by a single user (**Fig. 1**). Each individual band or channel is wide enough to accommodate the signal spectra of the transmissions to be propagated. The data to be transmitted is modulated on to each subcarrier, and all of them are linearly mixed together.

The best example of this is the cable television system. The medium is a single coax cable that is used to broadcast hundreds of channels of video/audio programming to homes. The coax cable has a useful bandwidth from about 4 MHz to 1 GHz. This bandwidth is divided up into 6-MHz wide channels. Initially, one TV station or channel used a single 6-MHz band. But with digital techniques, multiple TV channels may share a single band today thanks to compression and multiplexing techniques used in each channel.

This technique is also used in fiber optic communications systems. A single fiber optic cable has enormous bandwidth that can be subdivided to provide FDMA. Different data or information sources are each assigned a different light frequency for transmission. Light generally isn't referred to by frequency but by its wavelength (λ). As a result, fiber optic FDMA is called wavelength division multiple access (WDMA) or just wavelength division multiplexing (WDM).

One of the older FDMA systems is the original analog telephone system, which used a hierarchy of frequency multiplex techniques to put multiple telephone calls on single line. The analog 300-Hz to 3400-Hz voice signals were used to modulate subcarriers in 12 channels from 60 kHz to 108 kHz. Modulator/mixers created single sideband (SSB) signals, both upper and lower sidebands. These subcarriers were then further frequency multiplexed on subcarriers in the 312-kHz to 552-kHz range using the same modulation methods. At the receiving end of the system, the signals were sorted out and recovered with filters and demodulators.

Original aerospace telemetry systems used an FDMA system to accommodate multiple sensor data on a single radio channel. Early satellite systems shared individual 36-MHz bandwidth transponders in the 4-GHz to 6-GHz range with multiple voice, video, or data signals via FDMA. Today, all of these applications use TDMA digital techniques.

Time Division Multiple Access (TDMA)

TDMA is a digital technique that divides a single channel or band into time slots. Each time slot is used to transmit one byte or another digital segment of each signal in sequential serial data format. This technique works well with slow voice data signals, but it's also useful for compressed video and other high-speed data.

A good example is the widely used T1 transmission system, which has been used for years in the telecom industry. T1 lines carry up to 24 individual voice telephone calls on a single line (**Fig. 2**). Each voice signal usually covers 300 Hz to 3000 Hz and is digitized at an 8-kHz rate, which is just a bit more than the minimal Nyquist rate of two times the highest-frequency component needed to retain all the analog content.

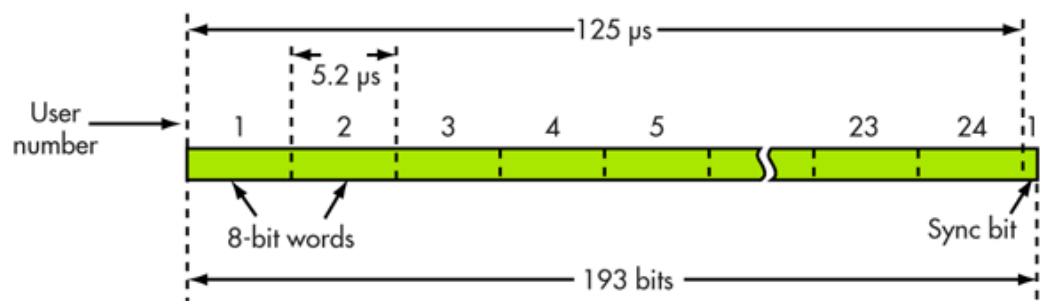
The digitized voice appears as individual serial bytes that occur at a 64-kHz rate, and 24 of these bytes are interleaved, producing one T1 frame of data. The frame occurs at a 1.536-MHz rate (24 by 64 kHz) for a total of 192 bits. A single synchronizing bit is added for timing purposes for an overall data rate of 1.544 Mbits/s. At the receiving end, the individual voice bytes are recovered at the 64-kHz rate and passed through a digital-to-analog converter (DAC) that reproduces the analog voice.

The basic GSM (Global System of Mobile Communications) cellular phone system is TDMA-based. It divides up the radio spectrum into 200-kHz bands and then uses time division techniques to put eight voice calls into one channel. **Figure 3** shows one frame of a GSM TDMA signal. The eight time slots can be voice signals or data such as texts or e-mails. The frame is transmitted at a 270-kbit/s rate using Gaussian minimum shift keying (GMSK), which is a form of frequency shift keying (FSK) modulation.

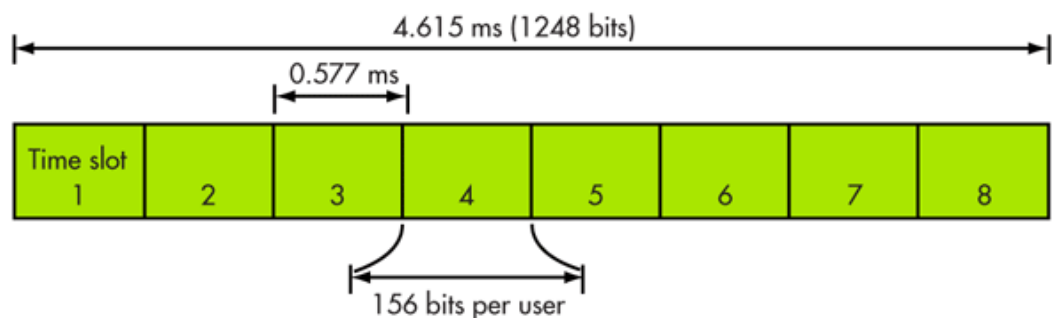
Code Cision Multiple Access (CDMA)

CDMA is another pure digital technique. It is also known as spread spectrum because it takes the digitized version of an analog signal and spreads it out over a wider bandwidth at a lower power level. This method is called direct sequence spread spectrum (DSSS) as

2. This T1 digital telephony frame illustrates TDM and TDMA. Each time slot is allocated to one user. The high data rate makes the user unaware of the lack of simultaneity.



3. This GSM digital cellular method shows how up to eight users can share a 200-kHz channel in different time slots within a frame of 1248 bits

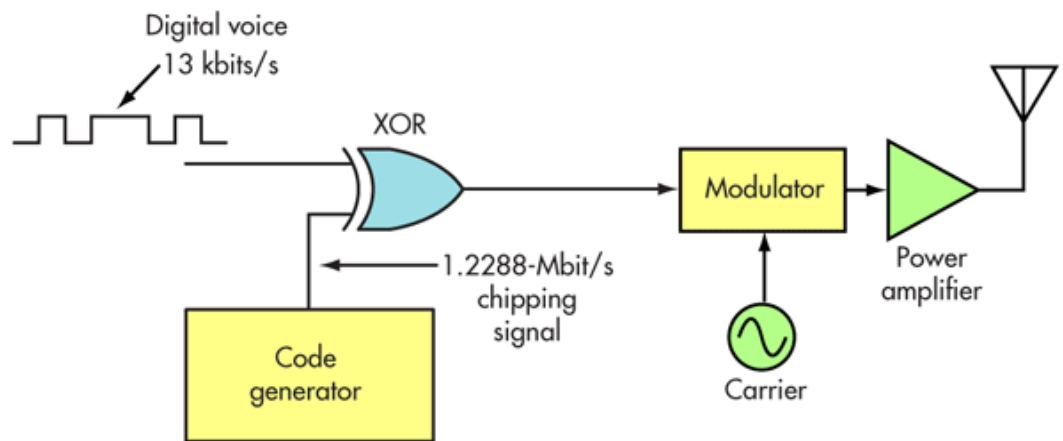


well (**Fig. 4**). The digitized and compressed voice signal in serial data form is spread by processing it in an XOR circuit along with a chipping signal at a much higher frequency. In the cdma IS-95 standard, a 1.2288-Mbit/s chipping signal spreads the digitized compressed voice at 13 kbits/s.

The chipping signal is derived from a pseudorandom code generator that assigns a unique code to each user of the channel. This code spreads the voice signal over a bandwidth of 1.25 MHz. The resulting signal is at a low power level and appears more like noise. Many such signals can occupy the same channel simultaneously. For example, using 64 unique chipping codes allows up to 64 users to occupy the same 1.25-MHz channel at the same time. At the receiver, a correlating circuit finds and identifies a specific caller's code and recovers it.

The third generation (3G) cell-phone technology called wideband CDMA (WCDMA)

4. Spread spectrum is the technique of CDMA. The compressed and digitized voice signal is processed in an XOR logic circuit along with a higher-frequency coded chipping signal. The result is that the digital voice is spread over a much wider bandwidth that can be shared with other users using different codes.



uses a similar method with compressed voice and 3.84-Mbit/s chipping codes in a 5-MHz channel to allow multiple users to share the same band.

Orthogonal Frequency Division Multiple Access (OFDMA)

OFDMA is the access technique used in Long-Term Evolution (LTE) cellular systems to accommodate multiple users in a given bandwidth. Orthogonal frequency division multiplexing (OFDM) is a modulation method that divides a channel into multiple narrow orthogonal bands that are spaced so they don't interfere with one another. Each band is divided into hundreds or even thousands of 15-kHz wide subcarriers.

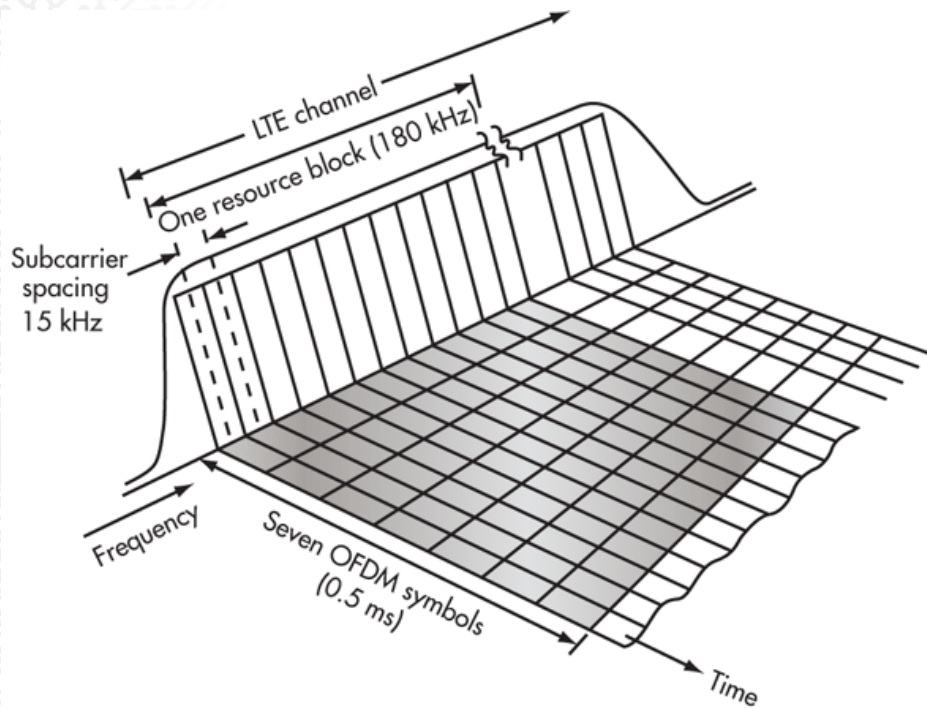
The data to be transmitted is divided into many lower-speed bit streams and modulated onto the subcarriers. Time slots within each subchannel data stream are used to package the data to be transmitted (**Fig. 5**). This technique is very spectrally efficient, so it provides very high data rates. It also is less affected by multipath propagation effects.

Spatial Division Multiple Access (SDMA)

SDMA uses physical separation methods that permit the sharing of wireless channels. For instance, a single channel may be used simultaneously if the users are spaced far enough from one another to avoid interference. Known as frequency reuse, the method is widely used in cellular radio systems. Cell sites are spaced from one another to minimize interference.

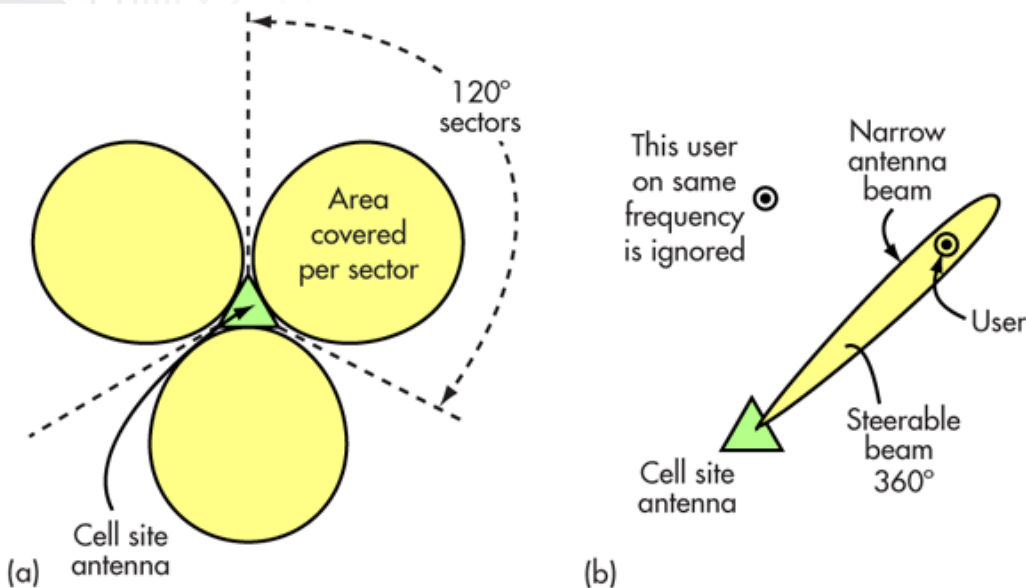
In addition to spacing, directional antennas are used to avoid interference. Most cell sites use three antennas to create 120° sectors that allow frequency sharing (Fig. 6a). New technologies like smart antennas or adaptive arrays use dynamic beamforming to shrink signals into narrow beams that can be focused on specific users, excluding all others (Fig. 6b).

One unique variation of SDMA, polarization division multiple access (PDMA), separates signals by using different polarizations of the antennas. Two different signals then can use the same frequency, one transmitting a vertically polarized signal and the other transmitting a horizontally polarized signal.



5. OFDMA assigns a group of subcarriers to each user. The subcarriers are part of the large number of subcarriers used to implement OFDM for LTE. The data may be voice, video, or something else, and it's assembled into time segments that are then transmitted over some of the assigned subcarriers.

6. SDMA separates users on shared frequencies by isolating them with directional antennas. Most cell sites have three antenna arrays to separate their coverage into isolated 120° sectors (a). Adaptive arrays use beamforming to pinpoint desired users while ignoring any others on the same frequency (b).



The signals won't interfere with one another even if they're on the same frequency because they're orthogonal and the antennas won't respond to the oppositely polarized signal. Separate vertical and horizontal receiver antennas are used to recover the two orthogonal signals. This technique is widely used in satellite systems.

Polarization is also used for multiplexing in fiber optic systems. The new 100-Gbit/s systems use dual polarization quadrature phase shift keying (DP-QPSK) to achieve high speeds on a single fiber. The high-speed data is divided into two slower data streams, one using vertical light polarization and the other horizontal light polarization. Polarization filters separate the two signals at the transmitter and receiver and merge them back into the high-speed stream.

Other Methods

A unique and widely used method of multiple access is carrier sense multiple access with collision detection (CSMA-CD). This is the classical access method used in Ethernet local-area networks (LANs). It allows multiple users of the network to access the single cable for transmission. All network nodes listen continuously. When they want to send data, they listen first and then transmit if no other signals are on the line. For instance, the transmission will be one packet or frame. Then the process repeats. If two or more transmissions occur simultaneously, a collision occurs. The network interface circuitry can detect a collision, and then the nodes will wait a random time before retransmitting.

A variation of this method is called carrier sense multiple access with collision avoidance (CSMA-CA). This method is similar to CSMA-CD. However, a special scheduling algorithm is used to determine the appropriate time to transmit over the shared channel. While the CSMA-CD technique is most used in wired networks, CSMA-CA is the preferred method in wireless networks.

Conclusion

So, what is the future of communications access technologies? There is no one-size-fits-all answer to this question, as the future of each communications access technology will depend on the specific requirements of the applications and networks that they are being used for. However, it is generally agreed that FDMA, TDMA, CDMA, OFDMA, and SDMA will all continue to play important roles in communications systems in the future. They will continue to provide higher bandwidths and capacities to meet the needs of businesses and consumers. The trend toward more wireless and mobile communications will continue as well, as these technologies offer more flexibility and increased throughput for future generations.

References

- Frenzel, Louis E., *Principles of Electronic Communication Systems*, 3rd Edition, McGraw Hill, 2008.
- Gibson, Jerry D., Editor, *The Communications Handbook*, CRC Press, 1997.
- Skylar, Bernard, *Digital Communications*, 2nd Edition, Prentice Hall, 2001.
- Tomasi, Wayne, *Advanced Electronic Communications Systems*, 4th Edition, Prentice Hall, 1998.

to view this article online,  [click here](#)

 **BACK TO TABLE OF CONTENTS**