Design sophistication and teamwork are needed

The Remote World of D

TDigital communications have added a new dimension to the family of telecommunications. To many commercial firms, this is a relatively new and uncharted field. But some firms have begun to exploit this facet of data processing.

The difference between digital telecommunications and batch processing systems may be defined as: Digital telecommunications are random and unscheduled inputs from remote locations primarily involving message switching and data exchange, normally using assembler language; while batch processing is scheduled and usually local serial inputs of card, paper tape, or magnetic tape, primarily involved in business and scientific applications, in such higher level languages as COBOL, FORTRAN, and PL/I.

Digital telecommunications systems have a common base. Information must be transmitted from one point to another via a physical wire channel or atmospheric radio signals, using electronic or electromechanical transmitting and receiving devices. The distance can be a few yards or thousands of miles. The common base is the transmission facility (channel or circuit) over which the information is passed. The grade or identification of a telecommunications (wire) channel is based on the information-carrying capability as determined by its baud or bit speed. Channels are referred to as voice grade, subvoice, and telegraph. These telecommunications channels and their characteristics will be described later.

This article will give a brief history of message switching systems and discuss principles of functional design and implementing and testing concepts for digital communications in message switching applications. The material is presented as an overview concerning the operational design and functional aspects of digital message switching and not the engineering and system design studies. However, some technical aspects of transmission modes, code selection, hardware, and channel characteristics will be discussed.

Background of message switching

On May 24, 1844, when Samuel F. B. Morse transmitted his famous message, "What hath God wrought!" 40 miles from Washington to Baltimore, he gave birth to telecommunications. Morse's telegraph system was the forerunner of the on-line digital telecommunication systems of today.

The military, in joint ventures with commercial carrier firms and equipment vendors, has made many innovations in the field of telecommunications, particularly in digital message switching systems. Owing to the logistic and organizational (worldwide) structure, the military, by necessity, is at the far fringes of the "science" of digital communications and message switching.

The manual torn-tape Teletype method was the

Digital Switching

by William L. Harper

first generation of message switching. This system required information to be teletyped (punched) on paper tape in a 5-bit code. The code, represented by one to five holes punched through a paper tape (see Fig. 1, page 24) is known as the Baudot code and requires five bits to represent an alpha or numeric character. The bit pattern of the code can be manipulated to allow a maximum of 32 characters (2^5) to be represented. The 26 alphabetic letters and the 10 decimal figures could not be represented by the 32 characters of the Baudot code. To overcome this, some alphanumeric characters were assigned the same 5-bit code combination. A 5-bit figures-and-letters code was added to represent an upper case (FIGS) and a lower case (LTRS). By adding the two extra codes, the number of characters and special symbols that could be represented by the Baudot code was increased to 57.

After preparation of the tape or receipt of messages from a remote terminal, the operator would manually tear the message from a spool of tape affixed to the teletypewriter and hand carry it to a transmitting device for transmission over a telecommunications (telegraph) channel connected to the addressee. All in-house processing was performed manually, and switching or routing of messages was determined by a unique routing indicator (code) assigned to each addressee. Transmission speed was 60 wpm. Message switching and information exchange were cumbersome and slow. Reliability of the network rested with the communications center operator. Error checking and controls were left to his discretion. Many messages were lost or subjected to excessive delays due to human errors.

In a search for a more reliable and faster system, a semiautomatic switching network was implemented. This second-generation network used the Baudot transmission-code structure but utilized a continuous paper tape method instead of the torn-tape concept. Transmission speeds varied from 60 to 100 wpm.

The concepts of operation were primarily the same as those of the torn-tape. Some in-house processing was automated. Upon receipt of a message, the routing indicator was interpreted by the operator, and a push button corresponding to the desired addressee was depressed on the routing indicator panel. The significant difference between the torn-tape and semiautomatic network was that the latter did not require tapes to be physically handled, torn, or "walked" from a receiving position to a transmitting machine. The message could be "automatically" transmitted from the same position where it was received by pressing a button.

Moving from the torn-tape to the semiautomatic required more sophisticated hardware and a higher degree of operator training and competence. Certain control functions were added to the message format to facilitate machine processing. These control functions, placed in the message format at designated places, prevented messages from being lost, delayed, or misrouted. For the most part, the reliability of the system was shifted from the communications center operator to the operator who prepared the tape at the originating station.

The semiautomatic system was more efficient and faster than its predecessor. But the reliability of a message getting from point A to point B was not significantly improved. Interpreting message routing, detection and correction of errors, and most of the message processing were still left to the discretion of the operator.

Increased reliability

With the advent of the aerospace age, a more reliable switching network was sought, with less dependency on the human factor. In the mid-'50s, the Air Force implemented a fully automatic paper tape message switching system using the Baudot code structure with center-to-center speeds of 100 wpm and cross-office speeds (within centers) of 200 wpm. With automatic switching, routing indicators and control characters were machine interpreted, and messages were automatically routed without intervention by an operator.

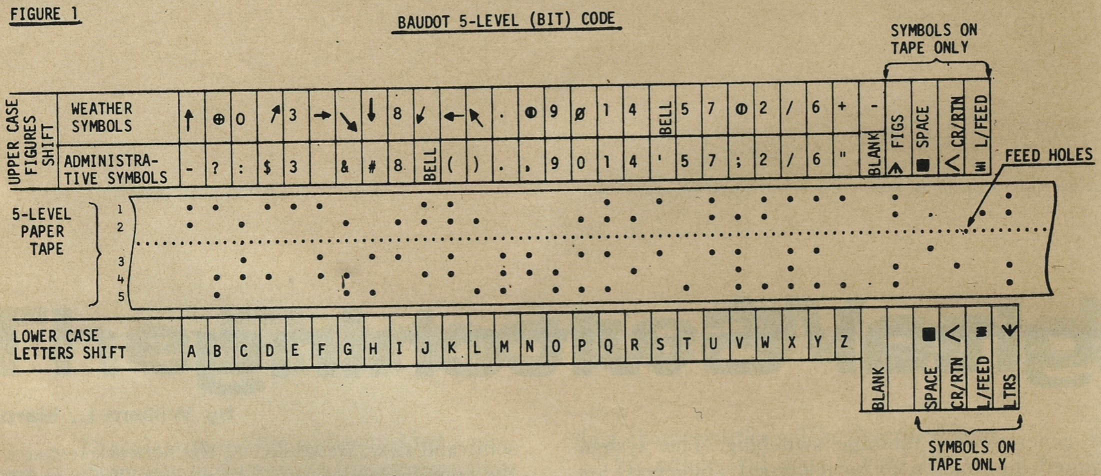


Fig. 1. Baudot 5-level (bit) code.

This third generation of message switching speeded up and facilitated message processing and information exchange. Invalid routing indicators, and garbled and incomplete messages were machine detectable. But the speed, reliability, and responsiveness were not sufficient to meet the aerospace demands for information handling. The system had two inherent weaknesses which prevented the network from being totally reliable: (1) there were no automatic controls to prevent erroneous messages from entering the network, and (2) too much discretion was left to the operator for message correction and control.

In the quest for a totally reliable system that would meet the enormous demands for information exchange, would negate entry of erroneous messages, and would be less dependent on the operator's discretion for message corrections and control, the Air Force, in 1963, with Western Union as prime contractor, implemented the world's first computerized, high-speed, digital telecommunications message switching network that performed those functions electronically, under the control of software or hard-wired programs.

The significant features of this fourth generation of message switching were:

1. Signalling speeds up to 4,800 bps, or 6,000 wpm.
2. Transmission and processing code of eight bits, seven information bits and one parity bit, for a total of 128 different code combinations to represent a data character or control function, vs. 57 for the 5-bit Baudot code (see Fig. 2).
3. Code conversions capability.
4. Automatic error detection and rejection of erroneous messages prior to entering the system.
5. Automatic recording of errors and other journal recording for historical and statistical purposes.
6. Recording and storing messages on magnetic devices vs. paper tape.
7. Electronic surveillance of messages to prevent loss or delays within the center.
8. Electronically switching messages on a priority FIFO basis.
9. Simultaneous message switching to a given number of terminals.
10. Modular hardware and software expansion capability to accommodate up to 250 terminals.

With computerized message switching, the operator acts and/or reacts to the dictates of the system. Once a message enters the system, it is under the control of an electronic (software) environment.

Functional design

Some networks may require two switching arrangements—digital message switching and a circuit switching system. Message switching involves (1) message (data) collection, (2) message storage, queueing, and retrieval, (3) message processing, which may include altering the message by deleting or adding data, code conversion, and bookkeeping functions, (4) distribution (switching) data to predetermined locations, and (5) on-line message inquiry capability, i.e., remote terminals randomly accessing a data bank in the switching center for data concerning message reruns, and historical and statistical information.

Circuit switching employs none of the above software techniques. Circuit switching is a point-to-point, circuit-to-circuit arrangement where the switching center acts as a connector of two or more remote terminals, permitting these terminals to exchange messages, and bypass the processing steps mentioned above.

Functional aspects for the operational requirements involve consideration of: (1) data source and input frequency, (2) volume, (3) distribution and data use, (4) center and terminal output requirements, (5) manual vs. automated functions, (6) growth expansion, (7) resources required, etc. The reliability of a digital communications system is in the functional architectural design and not so much the technical (software) development of the system, just as the safety, functionality, and quality of a building rest with the architect. Competent management, testing, and debugging controls should insure that the system is technically developed according to design plans. If the system's blueprints fail to specify the true operational requirements at the outset, the system will go through a costly iterative trial and error period until all functional aspects of the system are met.

A group of individuals who grew up with and understand the old system at the operator and management levels, who have an imaginative and in-depth understanding of present and future requirements, and who understand the technical (hardware/software) and economic advantages of digital communications should write the functional design specifications for the new system.

This team should not function as an *ad hoc* committee. Instead, individual members should do the data analysis and develop the functional specifications for operational features, such as:

1. Volume of data the system is to handle (analysis of terminal and user requirements).
2. Hierarchy (priority) of storage, queueing, retrieval, and switching messages.
3. Routing (switching) requirements, including alternate routes during channel outage or overload.
4. Restart and recovery procedures (for normal and emergency shutdowns).
5. Statistical and journal requirements.
6. Message throughput requirements.
7. On-line message security protection (if required).
8. Message format and control functions.
9. Message accountability during system failure.
10. Error control and correction.
11. Message storage and retrieval during system overload or channel outage.
12. Operator control vs. automatic control by computer.
13. Man - machine communications (computer printouts and operator responses).
14. Terminal control: frequency of input—free entry or scan (polled).
15. Other unique network requirements based on user/customer needs.

The team should meet periodically to review the progress of the design specifications and should consult with the user/customer during the design stages to insure network integrity. Once all requirements have been individually documented, they can be consolidated into an operational requirements document and be made available to a group of software technicians for program development. Designers must work continuously in a close relationship with programmers until each feature is independently developed and successfully tested and interfaced with all other operational features. Coordination and control are the cohesive elements throughout the writing of the design specifications and the system's software development, testing, and implementation.

The designer should have a working knowledge of the computer and transmission characteristics, as well as transmission code structures and error detection and correction concepts; otherwise, the counsel and judgment of the vendor may prevail. Unless there is a meeting of the minds, i.e., an understanding of both parties as to the data volume, operational and functional objectives, and the correct hardware system to satisfy these objectives, management may purchase or lease equipment not economically suitable for the network's requirements.

BIT POSITIONS											
			b7 →	0	0	0	0	1	1	1	1
			b6 →	0	0	1	1	0	0	1	1
			b5 →	0	1	0	1	0	1	0	1
b4	b3	b2	b1								
0	0	0	0				TEL	BLANK	K	)	Ø
0	0	0	1			#	ACK-1	UC	L	MINUS -	1
0	0	1	0				ACK-2	LC	M	+	2
0	0	1	1			OWD	REQ	LF	N	<	3
0	1	0	0				WBT	CR	O	=	4
0	1	0	1			@	REP	SP	P	>	5
0	1	1	0			%	SOM-L	A	Q	U/LINE	6
0	1	1	1			¢	ER	B	R	$	7
1	0	0	0			BELL	DM	C	S	*	8
1	0	0	1	IL		&	EOM	D	T	(	9
1	0	1	0			MC	SOLB	E	U	"	APOS '
1	0	1	1			≠	EDB	F	V	:	;
1	1	0	0			≢	EOLB	G	W	?	/
1	1	0	1			DEGREES°	RM	H	X	!	.
1	1	1	0			+0	SOM-H	I	Y	,	LOZENGE ◊
1	1	1	1		TAB	−0		J	Z	⊕	IGNORE i

FIELDATA 7-level (bit) plus 1 parity bit code is used as a standard military code in some digital communications systems. Out of a total of 128 code combinations, only 94 are used to represent either an alpha, numeric, symbol, or control function. Some code combinations may have been altered by extending or deleting some control functions or symbols to conform to system requirements.

LEGEND:

IL	Idle Line	SOLB	Start of Line Block
TEL	Teletype Assignment	EDB	End of Data Block
OWD	One Way Delete	EOLB	End of Line Block
MC	Mode Change	RM	Reject Message
ACK-1,2	Acknowledge	SOM-H	Start of Message High
REQ	Request	UC	Upper Case
WBT	Wait Before Transmitting	LC	Lower Case
REP	Repeat	LF	Line Feed
SOM-L	Start of Message Low	CR	Carriage Return
ER	Error	SP	Space
DM	Disregard Message	TAB	Tabulation
EOM	End of Message		

Fig. 2. 7-level FIELDATA (military) code.

Computer characteristics

Foremost should be (1) modular (both software and hardware) expansion flexibility, (2) internal software (processing) code compatible with the external transmission code structure, (3) interrupts and priority processing capabilities, and (4) I/O buffering and program relocatability.

Number 1 is important because a message switching system should be flexible enough to allow maximum I/O devices and communications lines to be attached without significantly disrupting operations or requiring major software modifications. Number 2 is important because this will eliminate code conversion routines, conserve core, and save I/O channel processing time in converting from one code to another. For example, when the channel is converting from an internal to an external code or vice versa, e.g., EBCDIC to ASCII, this time would be used for processing data. However, it may not be practical for the network to have a compatible code. To do so may affect the flexibility of the network by limiting the selection of terminal devices because such devices are engineered for different transmission codes.

Number 3 is significant because, when processing messages with different priorities and importance, it would be extremely difficult to process and switch messages without a complicated array of software switches. The interrupt programs for a message switching environment should be on a hierarchy-level, i.e., different types of interrupts based on data processing priorities, automatic control and bookkeeping of interrupts, and an interrupt program with the ability to interrupt and suspend the processing of another program that has interrupted and suspended the processing of a previous program. Number 4 is equally important because a message (or characters) may enter the system at random, independent of CPU control. These messages must be saved temporarily until they can be processed. Separate hardware may be used for buffering; however, some systems may require dynamic core buffering.

The computer should possess a control mechanism, either hardware or software, that will permit programs to be written without the programmer concerning himself with where the program will be located in core when it is executed. Certain hardware registers, known as base, relocation, or index registers, etc., will permit program relocation at any time in a program that is being executed: The interrupt program should control the relocation and return of the program to its previous location, or a different core location, through a series of bookkeeping functions.

Management may not have too much to say about the characteristics or compatibility of equipment or the transmission facilities because they are designed to the manufacturer's specifications rather than the user's. Volume and type of data and throughput requirements will give indications as to computer utilization and the appropriate characteristics such as speed, core size, internal software code, and core modularity.

Transmission channels

Channels are classified as simplex, half-duplex, or full-duplex. A simplex channel is a single-wire circuit, and data can be transmitted in one direction only. A half-duplex circuit is normally a two-wire circuit and can transmit data in two directions but in only one direction at a time. When speed in transmission is not a major factor, asynchronous mode and half-duplex would be desirable. A full-duplex circuit is normally a four-wire circuit and can transmit data in two directions simultaneously. These three types of channels can transmit data in a serial or parallel pattern in one of two transmission modes—asynchronous or synchronous.

To achieve an economical transmission balance, the choice between a two-wire or four-wire circuit should be measured by (1) data volume, (2) speed and turnaround requirements, and (3) data acknowledgement requirements. In a two-wire channel, transmission is halted to permit the transmission of an acknowledgment. The volume of data derived from one channel can emanate from one terminal or from several on a polled basis (each station sends when electrically called), or on a loop or multipoint circuit, either electronically called or free entry (not under CPU control).

The asynchronous transmission mode is character sensitive. In Baudot, it takes seven bits to transmit an alphanumeric character, five bits to represent the character and a "start" and "stop" bit to separate each character and to synchronize the receiving device with the transmitting device. With an 8-bit code such as ASCII or EBCDIC in asynchronous mode, it takes 11 bits to transmit a character, eight bits (including the parity but in ASCII) to represent the character and one start bit and two stop bits.

Signals are sent on a transmission line either as a mark or space. A mark represents a single bit element of a pattern that represents a character or symbol, or a single mark could represent an alpha or numeric character (see the letter E in Fig. 1). A space represents a negative information bit value.

Synchronous transmission is serial "bit stream" oriented. Data are transmitted in serial bit pattern without the synchronizing start and stop bits. The channel is clocked or sampled at regular intervals by the receiving device to record (normally in a buffer area) the information bit values.

Synchronous channels are more efficient because more data can be transmitted per unit of time than in the asynchronous mode since no transmission time is required for the start and stop bits. Most high-speed terminal devices used in digital telecommunications are equipped for synchronous operation.

Parallel transmission permits sending all information bits in a character simultaneously over separate channels. Normally, parallel transmission is used only a few feet from the computer to the communications control interface device and, from there, data is transmitted in serial pattern.

In digital telecommunications, channel speed or capacity is measured in bits per second or characters per second. Different transmission codes use a different number of bits to represent a character. In order to determine the wpm, this number of bits must be known. (Transmission codes are discussed later.) Characters per second and wpm can be determined by these two formulas:

$$\text{cps} = \frac{\text{bps}}{\text{bpc}}$$

Where—cps (characters per second)
bps (bits per second) = 2400
bpc (bits per character) = 8

$$\text{cps} = \frac{2400}{8}$$

$$\text{cps} = 300$$

Words per minute—wpm (6 characters represent a standard word in digital communications)

$$wpm = \frac{cps \times 60 \text{ seconds}}{cpw}$$

Where—cps = 300
cpw (characters per word) = 6

$$wpm = \frac{300 \times 60 \text{ seconds}}{6}$$

$$wpm = \frac{18{,}000}{6}$$

$$wpm = 3{,}000$$

Voice grade channels

A single-voice wire channel has a nominal bandwidth of 4KHz theoretically capable of transmitting 2,400 bps or 3,000 wpm. The voice channel is the basic unit for creating wider bandwidths with a higher bps capacity, permitting a greater amount of data to be transmitted over a channel. In most cases, a line cannot be effectively used at the lowest or highest ends of the frequency cycle range because the outer frequency edges act as a buffer zone to absorb normal interference.

Bandwidth greater than 3KHz for digital transmission was made possible in the late '50s and early '60s by joining computer technology with a telecommunications carrier system. The carrier system is a technique that permits several binary data streams or voice conversations over a single channel by shifting a transmitting frequency to a higher channel range within a single 4KHz circuit, and using this higher frequency band to convey data or voice communications. This technique permits the grouping or combining of these frequency channels (bands) in a special way into a single channel or circuit known as a broadband (sometimes referred to as a wideband) channel.

With modern carrier techniques in circuit disciplines, broadband channels of 8, 12, 16, 24, and 48KHz in bandwidth can be derived from conventional voice channels. These broadbands can be made more flexible by a multiplexing technique. As an example—a bandwidth of 12KHz can be subdivided into four nominal 3KHz voice channels. One channel could be used for voice communications, another for binary data transmission, the third channel for facsimile, and the fourth channel could be multiplexed to provide a number of slow-speed Baudot telegraph channels. (More about multiplexing later.) Bandwidths of 96KHz are possible on a specially conditioned four-wire circuit. Even wider bands with greater bps speed are possible with microwave or coaxial cable channels.

In digital telecommunications, an interface device known as a modem or data set, is tied to a channel to vary or change (modulate or demodulate) a signal to provide compatibility between a transmitting and receiving device and the line. A modem serves several functions: (1) it converts a carrier frequency to a lower frequency suitable for digital transmission; (2) it improves and facilitates sending data bits to obtain an optimum speed; and (3) it protects the signal from certain interferences. But the modem's primary function is to convert (modulate) binary bits, as they leave the computer or transmitting device, into a frequency signal that the transmission is engineered for, and to convert (demodulate) the signal back to its binary state suitable for the receiving machine. Modem speeds are not related to the KHz capacity of the channel. They are related to the transmitting and receiving devices, and the state of the art in modulating and demodulating methods. The effective modem speeds for noncommon carrier, e.g., Rixon Electronics, range from 45 bps (60 wpm) to 9,600 bps (12,000 wpm) for voice grade leased channels. The

BIT POSITIONS → b1				0	0	1	1
→ b2				0	1	0	1
b6	b5	b4	b3				
0	0	0	0	SOH	&	-	0
1	0	0	0	A	J	/	1
0	1	0	0	B	K	S	2
1	1	0	0	C	L	T	3
0	0	1	0	D	M	U	4
1	0	1	0	E	N	V	5
0	1	1	0	F	O	W	6
1	1	1	0	G	P	X	7
0	0	0	1	H	Q	Y	8
1	0	0	1	I	R	Z	9
0	1	0	1	STX	SP	ESC	SYN
1	1	0	1	.	$	,	#
0	0	1	1	<	*	%	@
1	0	1	1	BEL	US	ENQ	NAK
0	1	1	1	SUB	EOT	ETX	EM
1	1	1	1	ETB	DLE	HT	DEL

With BCD, 64 code combinations are possible to represent Alpha (26), Numeric (10), Symbols (11), and Control Functions (17) to include BELL (BEL) and SPACE (SP). By adding two bits to this code, it becomes 'Extended Binary Coded Decimal Interchange Code' (EBCDIC) with 256 total code combinations.

LEGEND:

SOH	Start of Heading	DLE	Data Link Escape
STX	Start of Text	ESC	Escape
BEL	BELL	ENQ	Enquiry
SUB	Substitute	ETX	End of Text
ETB	End of Transmission Block	HT	Horizontal Tabulation
SP	Space	NAK	Negative Acknowledge
US	Unit Separator	EM	End of Medium
EOT	End of Transmission	DEL	Delete
		SYN	Synchronous

Fig. 3. 6-bit Binary Coded Decimal (BCD) transcode.

Bell carrier system has modems capable of transmitting up to 230,400 bps over a leased broadband (60 voice-band lines) channel.

Subvoice and telegraph channels

The subvoice grade of channel falls between the telegraph and voice grade channels and generally has a speed range of 100 to 180 bps or approximately 120 to 220 wpm. This grade channel is serial transmission oriented and can handle any transmission code. Voice transmission is not posssible with subvoice channels.

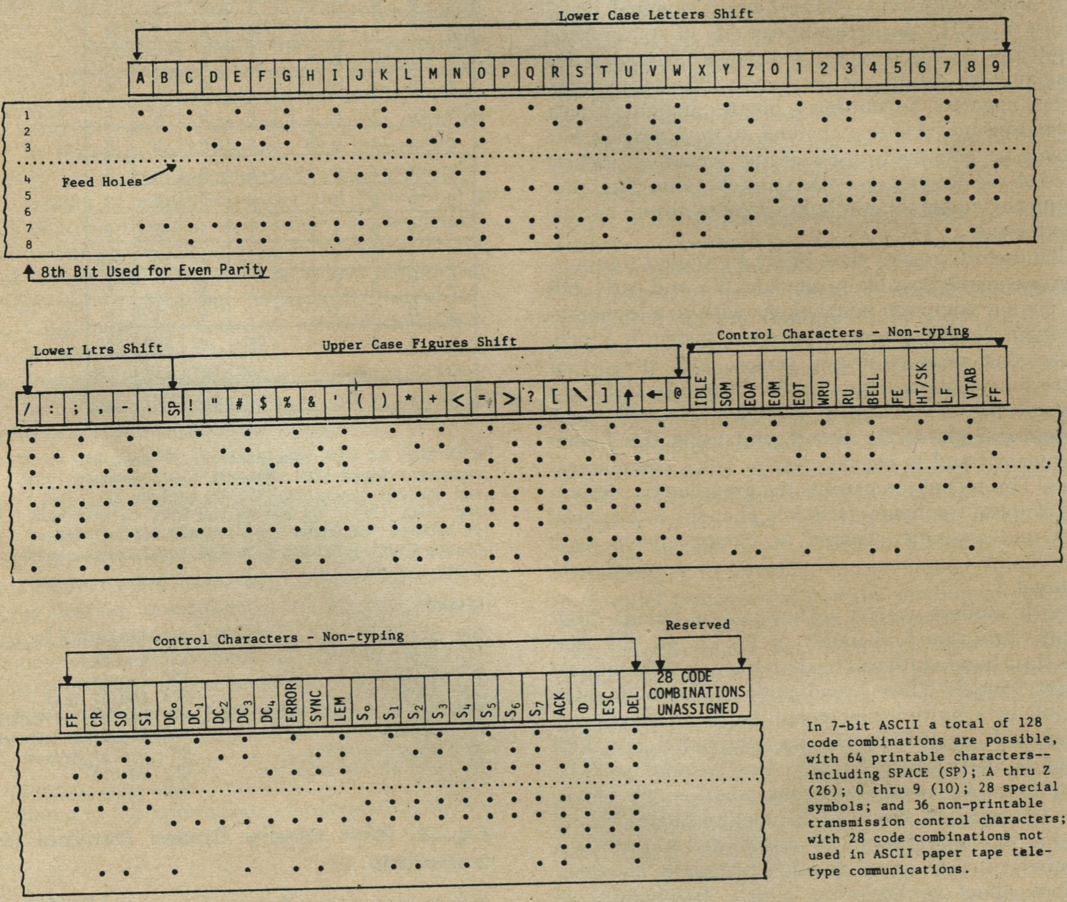
Telegraph grade channels are considered low speed. The normal range is between 45 and 75 bps or approximately 60 to 100 wpm. A carrier technique similar to the carrier system discussed in voice channels permits expanding a voice grade channel into many telegraph channels. This is known as channelizing or multiplexing.

There are two types of multiplexing—frequency division and time multiplexing. Frequency division multiplexing is used in the parallel mode. Each channel is assigned a unique frequency band or cycle. Using this principle, a voice grade channel is divided into several low-speed lines with each line assigned a specific frequency. This permits simultaneous transmission of different information on each frequency band. Time multiplexing is used in serial transmission mode. A voice channel is divided into discrete time slots with each slot transmitting different information from a different input. Several input lines supply simultaneous but different data signals to a scanning device. This device assembles and transmits a composite signal on a single line to a synchronized scanning device at the receiving end which separates the data signals and distributes them to the appropriate receiving line or device.

In binary synchronous communications (BSC), the data code is "transparent" to the line that transports it. The line serves only to transmit, at various speeds, the signals which represent the data code; the line is insensitive to the code structure.

There are several transmission codes available, but the codes favored are the 5-bit Baudot (discussed earlier), 6-bit BCD, 7-bit ASCII, and the 8-bit EBCDIC.

The Binary Coded Decimal (BCD) is an extension of the Hollerith code (used with EAM cards). BCD is a 6-bit character code and will allow 2^6 or 64 total code combinations (see Fig. 3, page 27). More



In 7-bit ASCII a total of 128 code combinations are possible, with 64 printable characters--including SPACE (SP); A thru Z (26); 0 thru 9 (10); 28 special symbols; and 36 non-printable transmission control characters; with 28 code combinations not used in ASCII paper tape teletype communications.

Legend:

IDLE	Null	SI	Shift In
SOM	Start of Message	DC_0	Device Control Reserved for Data Link Escape
EOA	End of Address	DC_1-DC_3	Device Control
EOM	End of Message	DC_4(Stop)	Device Control-Stop
EOT	End of Transmission	ERR	Error
WRU	Who are you?	S_0-S_7	Separator-Information
RU	Are you ....?	<	Less Than
BELL	Audible Signal	>	Greater Than
FE_0	Format Effector	↑	Up Arrow (Exponentiation)
HT	Horizontal Tabulation	←	Left Arrow (Implies Replaced By..)
SK	Skip (punched card)	ACK	Acknowledge
LF	Line Feed	Φ	Unassigned Control
VTAB	Vertical Tabulation	ESC	Escape
FF	Form Feed	DEL	Delete
CR	Carriage Return		
SO	Shift Out		

Fig. 4. 7-level ASCII paper tape code used with Teletype models 33 and 35.

data can be transmitted per unit of time with BCD than with ASCII or EBCDIC because BCD uses one bit less per character than ASCII and two bits less than EBCDIC. This makes BCD more efficient than either ASCII or EBCDIC because approximately 17% and 33% more data, respectively, can be transmitted. BCD has fewer control bits than the other two codes.

The USA Standard Code for Information Interchange (USASCII—more commonly, ASCII) is an 8-bit code . The code is designed to allow 2^8 or 256 code combinations. Currently, only seven bits (2^7 or 128 code functions) are being used for data and control information: 94 alphanumeric and graphic symbols and 34 control code functions (see Figs. 4 and 4A and Table 1). The eighth bit is used for transmission parity checking. ASCII was chosen by the federal government as the standard transmission and processing code. Military and other governmental agencies are favoring ASCII while many commercial firms are sticking with EBCDIC.

BIT POSITIONS →				0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1								
0	0	0	0	NUL	DLE	SP	Ø	@	P	~	p
0	0	0	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	STX	DC2	"	2	B	R	b	r
0	0	1	1	ETX	DC3	#	3	C	S	c	s
0	1	0	0	EOT	DC4	$	4	D	T	d	t
0	1	0	1	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	ACK	SYN	&	6	F	V	f	v
0	1	1	1	BEL	ETB	'	7	G	W	g	w
1	0	0	0	BS	CAN	(	8	H	X	h	x
1	0	0	1	HT	EM	)	9	I	Y	i	y
1	0	1	0	LF	SUB	*	:	J	Z	j	z
1	0	1	1	VT	ESC	+	;	K	[	k	{
1	1	0	0	FF	FS	,	<	L	\	l	¬
1	1	0	1	CR	GS	–	=	M	]	m	}
1	1	1	0	SO	RS	.	>	N	^	n	|
1	1	1	1	SI	US	/	?	O	—	o	DEL

LEGEND

NUL	Negative Value	DLE	Data Link Escape
SOH	Start of Heading	DC_1	Device Control
STX	Start of Text	DC_2	
ETX	End of Text	DC_3	
EOT	End of Transmission	DC_4	
ENQ	Enquiry (Character)	NAK	Negative Acknowledge
ACK	Acknowledge	SYN	Synchronous
BEL	Bell	ETB	End of Transmission Block
BS	Backspace	CAN	Cancel
HT	Horizontal Tabulation	EM	End of Medium
LF	Line Feed	SUB	Substitute
VT	Vertical Tabulation	ESC	Escape Character
FF	Form Feed	FS	File Separator
CR	Carriage Return	GS	Group Separator
SO	Shift Out	RS	Records Separator
SI	Shift In	US	Unit Separator
SP	Space	DEL	Delete

Fig. 4A. 7-level ASCII software code.

The Extended Binary Coded Decimal Interchange Code (EBCDIC), as the name implies, is an extension of BCD. It requires eight bits to represent a character and allows 2^8 or 256 code combination (see Fig. 5, page 30). However, unlike ASCII, all eight bits of the code are used to represent data or a control function.

Knowing the grade definitions and differences between channels and how to recognize a character by the bit code structure are of little significance. Knowing the following is important: (1) how many bits are required to represent a character; (2) what combinations of characters and symbols can be represented; (3) the channel grade and transmission mode required to transmit the code; (4) channel utilization; (5) throughput and turnaround requirements; (6) code conversion requirements; and (7) error detection and correction methods. This last item deserves further discussion.

Most message switching applications require an error detection and automatic retransmission request (ARQ) capability. The bit configuration that represents the control characters, which determine the message distribution, its priority, and security, must be protected during transmission and machine processing. Error detection and correction becomes more critical in the transmission of perishable data, i.e., meteorological data concerning severe weather warnings to command and control systems. These data must be acted upon by man or machine in real-time, and in addition to the bit configuration mentioned above, each character of the text must be protected.

The type of error detection used depends upon the system's requirements. The most economical and commonly used method is the parity bit vertical redundancy check (VRC) by character and the longitudinal redundancy check (LRC) by message block, sometimes referred to as a two-dimensional parity system. Parity may be even or odd and the check will be both a vertical bit check and a longitudinal (horizontal) check. A one-dimension parity system may be employed by using either the VRC or LRC method. However, the accuracy of the error detection would not be as valid as a combination of VRC/LRC.

The sum of the "one" bits for any single character (vertical check) or group of characters, known as a block (longitudinal horizontal check), will always be even or odd depending on which parity method is used.

To satisfy the odd or even parity arrangement in VRC, an extra bit is added to each character to make the bit configuration odd or even. The receiving device will check each character for the proper bit parity. If the check does not match the odd or even arrangement for the system, a negative check is sent to the transmitting device.

¢	Cent Sign	¬	Logical Not
.	Period, Decimal Point	-	Minus Sign, Hyphen
<	Less Than	/	Slash
(	Left Parenthesis	,	Comma
+	Plus	%	Per cent
|	Vertical Bar-Logical OR	_	Underscore
&	Ampersand	>	Greater Than
!	Exclamation Point	?	Question Mark
$	Dollar Sign	:	Colon
*	Asterisk	#	Number
)	Right Parenthesis	@	At
;	Semicolon	=	Equals
"	Quotation Mark	'	Prime, Apostrophe
∧	Exponentiation	{	Left Brace
]	Right Bracket	}	Right Brace
\	Reverse Slant	~	Tilde
[	Left Bracket		

Note: Certain of these symbols are represented by various code combinations of either Baudot, ASCII, BCD, or EBCDIC.

Table 1. Graphic symbol meanings.

In the LRC parity method, characters are grouped into horizontal columns or units known as a block, and depending upon requirements, may vary in length. Each block of data is indicated by an end of block (EOB) control character. To satisfy the LRC odd or even arrangement, a count of the number "one" bits contained in each column is inserted by the transmitting device immediately after transmitting a block of data which is usually one line of data. The receiving device compares this count with the number of "one" bits received. If they are equal, an ACK (acknowledgment) is sent and the next block is transmitted. If the count is not equal, the transmitting terminal automatically retransmits the block. Usually, after three trys without successfully retransmitting the block, an audible alarm or light, or both, alerts the transmitting operator that transmission difficulty is being experienced. The major disadvantage of a simple parity system is that corrections cannot be made.

Other error detection and correction codes are available which may be used with other transmission codes, but they are complex and more expensive than the two-dimensional VRC and LRC parity method. One such code is the cyclic redundancy check (CRC) code. This code adds check bits to the data stream by dividing the serialized bits in a block (similar to LRC) by a predetermined binary number. The remainder, being the check bits, is transmitted to the receiving device where it is compared in a similar arithmetic manner. The use of the code may monopolize as much as 20% of the bit stream for error checking and require expensive circuitry at both ends of the line.

Message switching systems developed without comprehensive testing and validation of the operational specifications would cause complete chaos when trying to implement and operate the system, and for months would cause frustrating problems, both software and operational. Although programmers may test and debug the individual features or modules of major programs until they are logically correct, when these modules are put together to do a particular processing job, it is imperative that they be tested against the actual conditions as specified in the requirements package.

The measurement of a successful test and the implementation of an on-line message switching network depend on many variables that are not present in a batch system. Errors can more easily be detected and corrected in a batch system than in an on-line system because of the nature of the two systems. In on-line systems, there are fewer stand-alone programs or subroutines. The majority of the programs/routines must interface, process and pass on, and, in some cases, alter data, and perhaps do this at different core address/locations. This makes interfacing numerous interrelated programs extremely difficult, and makes

BIT POSITIONS →b1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
→b2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
→b3	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
→b4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
b8 b7 b6 b5																
0 0 0 0	NUL	DLE	DS		SP	&	-									0
1 0 0 0	SOH	DC1	SOS				/		a	j			A	J		1
0 1 0 0	STX	DC2	FS	SYN					b	k	s		B	K	S	2
1 1 0 0	ETX	DC3							c	l	t		C	L	T	3
0 0 1 0	PF	RES	BYP	PN					d	m	u		D	M	U	4
1 0 1 0	HT	NL	LF	RS					e	n	v		E	N	V	5
0 1 1 0	LC	BS	ETB	UC					f	o	w		F	O	W	6
1 1 1 0	DEL	IL	ESC	EOT					g	p	x		G	P	X	7
0 0 0 1		CAN							h	q	y		H	Q	Y	8
1 0 0 1		EM							i	r	z		I	R	Z	9
0 1 0 1	SMM	CC	SM		¢	!		:								
1 1 0 1	VT				.	$	,	#								
0 0 1 1	FF	IFS		DC4	<	*	%	@								
1 0 1 1	CR	IGS	ENQ	NAK	(	)	—	'								
0 1 1 1	SO	IRS	ACK		+	;	>	=								
1 1 1 1	SI	IUS	BEL	SUB	|	¬	?	"								

LEGEND:

NUL	Null	PRE	Prefix	SI	Shift In
PF	Punch Off	SM	Set Mode	SMM	Start of Manual Message
HT	Horizontal Tabulation	PN	Punch On	DLE	Data Link Escape
LC	Lower Case	RS	Reader Stop	DC1	Device Control
DEL	Delete	UC	Upper Case	DC2	Device Control
RES	Restore	EOT	End of Transmission	DC3	Device Control
NL	New Line	SP	Space	DC4	Device Control
BS	Backspace	SOH	Start of Heading	NAK	Negative Acknowledge
IL	Idle	STX	Start of Text	SYN	Synchronous
CC	Cursor Control	ETX	End of Text	CAN	Cancel
DS	Digit Select	ACK	Acknowledge	EM	End of Medium
SOS	Start of Significance	BEL	Bell	SUB	Substitute
FS	Field Separator	VT	Vertical Tabulation	IGS	Information Group Separator
BYP	By Pass	FF	Form Feed	IRS	Information Record Separator
LF	Line Feed	CR	Carriage Return	IUS	Information Unit Separator
EOB/ETB	End of Block/or End of Transmission Block	SO	Shift Out	IFS	Information Field Separator
		ENQ	Enquiry		

Fig. 5. 8-bit Extended Binary Coded Decimal Interchange Code (EBCDIC).

in-depth testing a logical follow-through in a message switching environment.

Test material. Testing requirements, specifications, and procedures, along with keypunch instructions, should be developed by the architectural designers who developed the functional specifications. Equipment configuration and personnel requirements necessary for testing should also be specified. Local testing using in-house equipment may be sufficient for some systems. Others may require, in addition to local input/output, that remote terminals be used to simulate actual live conditions. This would be an ideal method for testing. However, with remote testing (unless part or all of the existing terminals can be made available for testing), additional communication lines and equipment will have to be installed to parallel the existing system. This may be an additional cost that management may not want to incur.

The test should be accomplished off-line with a sufficient quantity of predetermined and prepared testing materials. Test material consisting of dummy messages should be run through the system in sufficient quantities to test the functionality of the design specifications and to insure message protection.

Test team. Testing and implementation of the system should be a joint effort comprised of operators, programmers/analysts, and members of the functional design team or committee. A test coordinator should be appointed to direct the testing and implementation. A lead programmer(s) may be a member of the team for software consultation.

Saturation testing. During testing, every condition should be simulated with message data to specifically test each feature. Some features, such as message routing, statistical requirements, throughput, and format and control, will be fairly simple to validate. But restart/recovery, message accountability during system failure, and message storage and retrieval during system overload and channel outage will require more involved test procedures. To adequately test these features, the system should be saturated with messages of all categories and the system purposely stalled and restarted to test all aspects of these features. Without a comprehensive and in-depth shakedown, it may take many costly months to render the system reliable.

It is vital that the overload conditions be tested. This should be done by message saturation. The system should be designed so that overload occurs before the system becomes actually filled to capacity. Given a system of different message priorities, classifications, and messages with perishable data, overload should be triggered when the storage capacity has reached 80% to 90%. This 20% to 10% leeway should leave sufficient space to process high precedence and other urgent types of data.

To create an overload condition, more messages will have to enter the system than the system is capable of handling. This can be accomplished by writing test data to magnetic tape, disc, or drum, and then re-entering the data into the system simultaneously. If this is not practical, output channels and devices could be shut down, which would create the same effect. When the system reaches this predetermined figure, the overload feature will activate and low precedence messages will be written to a save area (external storage device). When the system falls below this figure, data should automatically re-enter the system and be queued for output processing. Some errors may occur only once during testing but when repeating the process that *seemed* to have caused the error, the same error condition will not repeat. Many errors will have to work themselves out over a period of time. Some errors may be caused by hardware or line conditions; such errors may be infrequent, due to random and low-volume input.

Cutover. Operator training (both terminal and switching center) should have been accomplished, and operating manuals and other documentation made available prior to final test and implementation. This is necessary to familiarize operators with the peculiarities of the computerized system. After cutover, it may be necessary to run the old system in parallel with the new system until sufficient confidence is built up with the computerized system. At this time the old system may be disbanded.

Past test observation. The validity of the system may not be proven until it has been run on-line in a dynamic message switching environment. After the system has been tested and implemented, it should be under close scrutiny for a period of time, and during this time one of the lead programmers and operations analyst should be readily available to correct any software or operational problems that may arise. A system may test out successfully in an off-line environment but when run on-line under live conditions, may experience difficulty. There are certain variables inherent in off-line testing that are not present in an on-line situation. In off-line testing, the input is calculated and controlled. For the most part, testing is done at favorable times when the processing of live messages is low and when the off-line equipment is less likely to be seized for on-line use due to equipment problems with the primary system.

Summary

Digital telecommunications and message switching principles and concepts are so complex and involved that only an overview can be stated in the few pages of this paper. The information presented here is general in nature, suggesting certain approaches to the functional aspects of the operational design and testing of a digital message switching system. ■

Mr. Harper is presently engaged in systems analysis and technical writing for the Air Force at the Communications Computer Programming Center, Tinker AFB, Oklahoma. He retired from the USAF in 1968 with 20 years' experience in telecommunications and data systems of the DOD's AUTODIN and the Air Force's Automated Weather Network.