

Systems

**OS TCAM
Concepts and Facilities**

IBM

Systems

OS TCAM Concepts and Facilities

**OS/MFT and OS/MVT:
Release Number 21.6**

OS/VS1: Release Number 1.0

OS/VS2: Release Number 1.0

IBM

Summary of Amendments

This revision updates the device support descriptions to include the IBM 3705 Communications Controller. The Appendix has been rewritten to include the new information.

Third Edition (January 1973)

This is a major revision of, and obsoletes GC30-2022-1, and Technical Newsletter GN30-2575, dated Sept. 30, 1972. This edition applies to Release 21.6 and to all subsequent releases of OS/MFT and OS/MVT TCAM and Release 1 and all subsequent releases of OS/VS TCAM, until otherwise indicated in new editions or Technical Newsletters.

Changes are periodically made to the information herein; before using this publication with IBM systems or equipment refer to the latest SRL Newsletter for the editions that are applicable and current.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This manual has been prepared by the IBM Systems Development Division, Publications Center, Department EO1, P.O. Box 12195, Research Triangle Park, North Carolina 27709. A form is provided at the back of this publication for readers' comments. If the form has been removed, comments may be sent to the above address. Comments become the property of IBM.

This publication is an introduction to the Telecommunications Access Method (TCAM) for persons requiring either a general knowledge of TCAM, or a basic understanding before studying more detailed documentation such as the *OS/MFT and OS/MVT TCAM Programmer's Guide*, GC30-2024 and the *OS/VS TCAM Programmer's Guide*, GC30-2034.

Readers should have a general knowledge of data processing and the IBM operating systems. Suggested readings are:

- *Introduction to IBM Data Processing Systems*, GC20-1684
- *OS Introduction*, GC28-6534
- *IBM Teleprocessing Systems Summary*, GA24-3090

A knowledge of teleprocessing is not assumed; however, most of the basic information on teleprocessing has been placed in one section (Section 2, "Teleprocessing Systems" so that readers may skip it if they wish.

This introduction to TCAM is divided into the following major sections:

- The "Introduction" briefly describes the general functions, capabilities, features, and advantages of TCAM. It also describes several message control, message switching, and message processing applications, and lists the devices supported by TCAM.
- "Teleprocessing Systems" describes teleprocessing con-

cepts and techniques, including line control, buffering, and queuing.

- "TCAM Environment" describes TCAM operating environment, line control, data sets, and buffers.
- The "Message Control Program" describes TCAM messages and message handling, illustrates message flow within a TCAM system, and how orderly message flow is maintained. Handling logical and concentrated messages is also explained.
- "Application Programs" describes the MCP/application program interface, how work units are specified, application-program buffers, and message flow. It also describes such features as network control, message retrieval, inquiry/response, coordinating TCAM and operating system checkpoints, and using QTAM application programs.
- "TCAM Service Facilities" describes the features and use of such facilities as operator control, I/O error recording, debugging aids, network reconfiguration, checkpoint/restart, and on-line test.
- The "Appendix" defines the machine and device requirements of TCAM.

In addition to the major sections, a bibliography lists related publications and publications referred to in this book, and the "Glossary" defines many teleprocessing and data processing terms used in this and other IBM publications.

Bibliography

<i>Key words</i>	<i>Title</i>	<i>Order No.</i>
Checkpoint/Restart Planning Guide	IBM System/360 Operating System Advanced Checkpoint/Restart OS/VS Checkpoint Restart	GC28-6708 GC26-3784
Data Processing Glossary	Data Processing Glossary	GC20-1699
General Information—BSC	General Information—Binary Synchronous Communications	GA27-3004

Bibliography (Continued)

<i>Key words</i>	<i>Title</i>	<i>Order No.</i>
Introduction to Data Processing	Introduction to IBM Data Processing Systems	GC20-1684
OS Introduction	OS Introduction Introduction to Virtual Storage in System/370	GC28-6534 GR20-4260
TCAM Level 4 Component Release Guide	OS TCAM Level 4 Component Release Guide	GC30-1007
TCAM PLM	OS/MFT and OS/MVT TCAM Logic OS/VS TCAM Logic	GY30-2029 SY30-2039
TCAM Programmer's Guide	OS/MFT and OS/MVT TCAM Programmer's Guide OS/VS Programmer's Guide	GC30-2024 GC30-2034
TCAM User's Guide	OS TCAM User's Guide	GC30-2025
Teleprocessing Systems Summary	IBM Teleprocessing Systems Summary	GA24-3090
TSO Command Language	IBM System/360 Operating System TSO Command Language Reference OS/VS2 TSO Command Language Reference	GC28-6732 GC28-0646
TSO Guide	IBM System/360 Operating System TSO Guide OS/VS2 TSO Guide	GC28-6698 GC28-0644

Introduction	1	The Message Control Program	17
Why Teleprocessing?	1	TCAM Messages	17
Special Characteristics of a Teleprocessing System	1	Message Handlers	18
Configuration of a Teleprocessing Network	1	Message Flow Within the System	18
Communication Lines	3	Maintaining Orderly Message Flow	21
Audio Lines	3	Message Priority and Queuing	21
Data Sets (Modems)	3	Transmission Priority	21
Advantages of a Queued Access Method	3	Handling Logical Messages	22
What TCAM Does	3	Message Concentrating	23
Features of TCAM	5	Time Sharing Option	23
Features Unique to TCAM	5	Application Programs	25
TCAM Response Time	6	The MCP/Application Program Interface	25
Applications	6	Defining Work Area and Unit	25
Message Switching	6	Specifying Application-Program Work Units	26
Data Collection	7	Message Retrieval	26
Message Processing	7	Inquiry/Response	26
Inquiry Processing	7	Coordinating TCAM and Operating System Checkpoints	27
Multiple Applications	7	TCAM Service Facilities	29
Teleprocessing Systems	9	Operator Control	29
Transmission Techniques	9	Operator Control Stations	29
Transmission Codes	9	Operator Commands from an Application Program	29
Line Control	9	Use of Alternate Destination	29
Invitation and Selection of Stations	10	Network Control	29
Polling	10	Logging	30
Addressing	11	I/O Error Recording	30
Contention	11	Additional Debugging Aids	30
Identification Exchange	11	Error-Recovery Procedures	30
Buffering	11	Mid-Batch Recovery	31
Queuing	11	TCAM Checkpoint/Restart Facility	31
TCAM Environment	13	Records for Checkpoint/Restart	31
The Operating Environment	13	Closedown	32
Terminal and Line Control	13	Restart	32
Data Sets	13	On-Line Test	32
Line Group Data Sets	14	Appendix: Configurations Supported	33
Message Queues Data Sets	14	Glossary	35
Checkpoint and Log Data Sets	14	Index	43
Buffers	15		
Program-Controlled Interruption	15		

Illustrations

<i>Figure</i>	<i>Title</i>	<i>Page</i>
1	Teleprocessing Configurations	2
2	TCAM Interfaces	4
3	An Example of Incoming Message Handler Subgroups, Using One and Multiple Message Handlers	19
4	TCAM Message Flow	20
5	Transmission Priority for Various Types of Lines and Transmission Techniques	22
16	Device Configurations Supported by TCAM	34.1

In a teleprocessing system, remote stations are linked to a computer, and to each other, by common carrier communication lines. TCAM (Telecommunication Access Method) is a generalized I/O control system that extends the techniques of data management to this teleprocessing environment. TCAM provides a Message Control Program that manages the teleprocessing network and controls the flow of data to and from remote terminals and user-written application programs.

WHY TELEPROCESSING?

A teleprocessing system can be defined as a data processing system plus communications. The communication lines are used to extend or expand a central processing system to remote locations. The basic idea of a teleprocessing system is simple: to transmit data to physically separated (remote) locations—dispersing centrally held data or collecting scattered data, or both. There are many reasons for moving data over long distances to or from a central computer: it may be more economical, faster, permit additional services, or make better use of equipment. Some typical teleprocessing applications are data collection, message switching, inquiry (such as deposit accounting or information retrieval), text editing, and remote computing. Many teleprocessing applications do not fit neatly into one of these categories, but are a combination of two or more applications.

SPECIAL CHARACTERISTICS OF TELEPROCESSING SYSTEMS

Teleprocessing has some characteristics unlike those of conventional systems. One is that message lengths can be variable. In message switching, for example, short messages containing only a few words must be processed with long messages having thousands of characters. The program must be flexible enough to accommodate such variations efficiently. This is in contrast to the more rigid record sizes found in the majority of non-teleprocessing computer systems.

Another characteristic of teleprocessing is that input is often random and unpredictable. Requirements for processing time, storage space, and lines are determined by the external operating environment and cannot be dictated by the computer. Instead, the access method must have considerable flexibility to be able to acquire resources when needed, yet not monopolize valuable computer facilities.

There are also some special problems in debugging user-written teleprocessing control programs. How can peak load conditions be simulated? How can errors associated with complex timing or queuing conditions be detected? When the system is “down” because of equipment malfunction or program errors, how are users to be informed? When the system is restarted, what data, if any, has been destroyed? Who is affected by the loss? Such questions indicate the problems associated with designing and testing teleprocessing programs.

A final teleprocessing characteristic is that communication circuits are subjected to a variety of environmental conditions, such as nearby high-voltage circuits, accidental induction between circuits (causing “crosstalk”), and electrical impulses caused by lightning strikes. These may create electrical “noise” and transmission errors. Consequently, error detection and corrections are more critical and need to be more elaborate in teleprocessing systems. Terminals must be sufficiently sophisticated to generate and check the necessary error-detection codes.

CONFIGURATION OF A TELEPROCESSING NETWORK

A teleprocessing system (or network) consists of:

- A central computer;
- Remote stations;
- Communication paths (lines or data links) that connect the remote stations to the computer (Figure 1).

The *central computer* is the central processing unit (CPU) in which TCAM is running. It also includes the transmission control unit and line adapters connecting the CPU to the communication lines.

A *remote station* is either a terminal or another computer. For example, TCAM supports the IBM System/360 and 1130 Computing Systems as remote stations, as well as several types of terminals.

A *terminal* consists of a control unit (not to be confused with a transmission control unit) and one or more input and output devices, each of which is called a *component* of that terminal. A user enters data from a keyboard or on cards, paper or magnetic tape at a station, and receives—as output from the computer—audio, printed, punched-tape, magnetic-tape, or visually displayed data. All messages between stations must pass through the computer; in addition, the computer may originate messages for the stations.

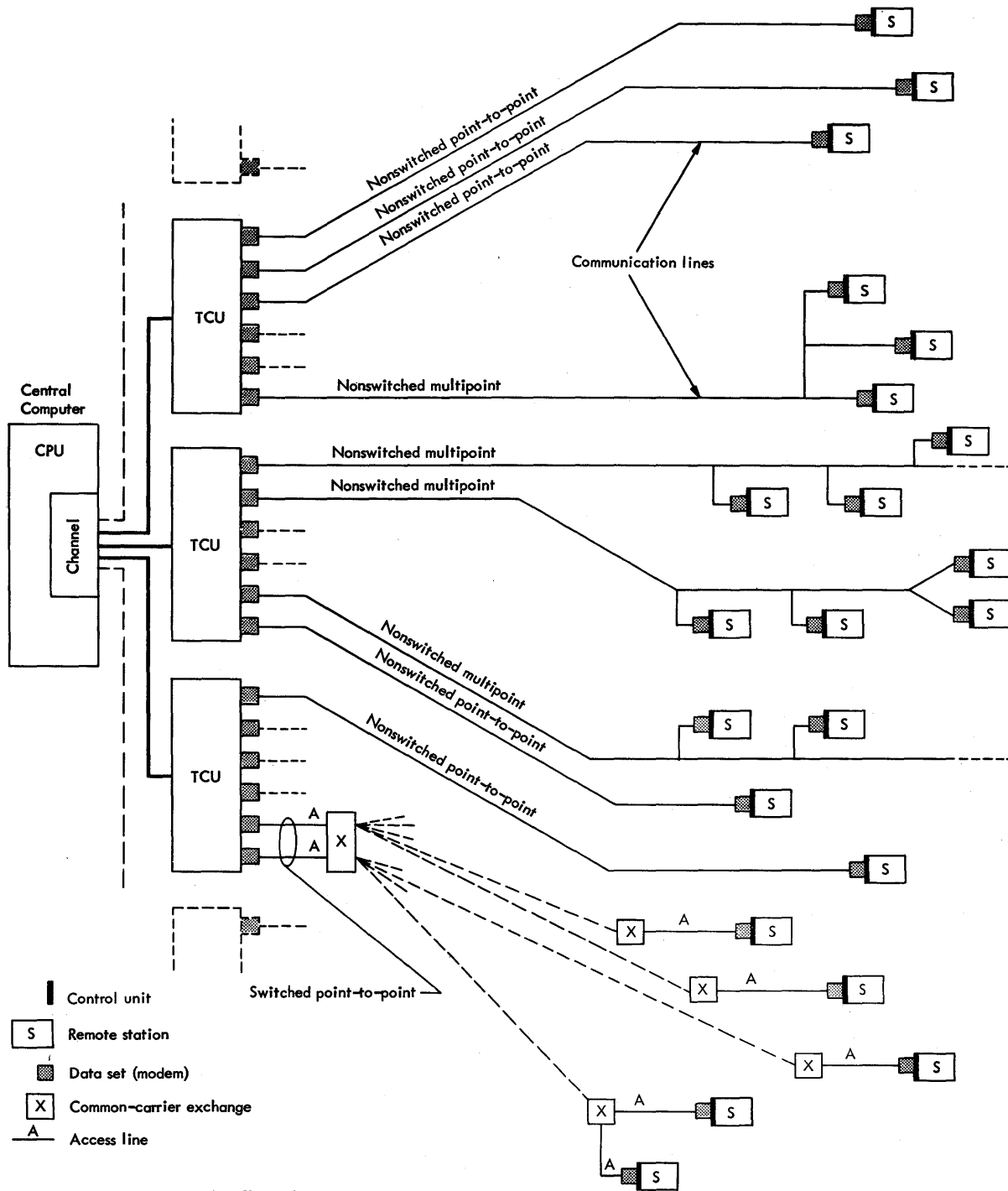


Figure 1. Teleprocessing Configurations

A remote station is connected to a computer through a transmission control unit (TCU) and common-carrier or user-owned transmission lines and facilities.

Communication Lines

A *nonswitched* communication line connects the stations on the line for either continuous or recurring periods. A nonswitched line is called *point-to-point* if it connects a single remote station to the computer, and *multi-point* if it connects more than one station.

A *switched* communication line establishes the connection between the computer and a remote station by dialing. As in a telephone network, the actual path for a given transmission is not fixed, but is automatically selected from a number of possible paths by common-carrier switching equipment. A switched line is always point-to-point, since the computer communicates with only one remote station on a line during any call.

A teleprocessing network can be either switched, non-switched, or a combination of the two (as shown in Figure 1). Dialing can be performed either at the central computer or at the remote station, and the called station can answer either manually or automatically. Not all of these options are available for all types of line configurations and remote stations. Within the limitations imposed by equipment or programming, the user chooses the options best suited to his application.

A transmission control unit usually has several access lines, each with its own telephone number, which permit simultaneous communication with several remote stations.

Audio Lines

An audio line is a switched telephone line attached to an audio response unit (ARU) such as the IBM 7770 Audio Response Unit Model 3. When audio response units are mentioned, it is assumed that they are attached to audio lines.

An *audio terminal* is a device associated with an audio response unit (ARU). The user keys in data from a terminal to the computer and receives an audible response from the ARU. The computer receives the input message and sends the audio output message to the audio terminal. Terminal control is also performed by the audio response unit.

Data Sets (Modems)

Each remote station, transmission control unit, and audio response unit can be connected to a communication line by a common-carrier *data set* (not to be confused with a *program data set*—see the second definition in the “Glossary”). Data sets are also known as modulator/demodulators or modems. A data set provides an interface between station and line, converting the binary signals of business machines to the transmission frequencies of communication equip-

ment, and vice versa. The TCAM programmer need not be concerned with data sets.

ADVANTAGES OF A QUEUED ACCESS METHOD

Teleprocessing systems vary tremendously in size and complexity. A simple inquiry system may consist of only a few lines and terminals. On the other hand, many systems are characterized by complex communication requirements, critical to carrying on the day-to-day business. As systems grow in complexity, the user looks for more flexibility, reliability, availability, and serviceability in his support system.

A queued, teleprocessing “access method,” such as TCAM, offers many advantages:

- Operators at remote stations can enter information into the system at their convenience. Queuing provides a staging area in which the information is held until the recipient—either an application program or another terminal—has been prepared to receive it.
- Information is distributed throughout the system without forcing the input to be synchronized with either an application program or other terminals.
- If queues of messages are maintained on a secondary-storage device, the teleprocessing system can be restarted after a system failure.
- Maintaining message queues on secondary storage also permits messages previously processed by the system to be retrieved.
- The asynchronous operation of the communication network and application programs increases system productivity since application programs can use prestored input.

TCAM also offers advantages over teleprocessing systems in which queuing must be provided by the programmer.

TCAM:

- relieves the programmer of the tedious work required in developing, maintaining, and expanding networks and application systems by providing a macro language.
- separates application programs from control-program functions. A TCAM interface between the Message Control Program and application program is provided through the familiar Sequential Access Method (SAM) interface.
- supplies an extensive set of service functions (for example, checkpoint/restart, operator control, and on-line terminal tests), which enhance the availability and usability of the communication system.

WHAT TCAM DOES

TCAM controls message traffic among stations, and between stations and user-written application programs (Figure 2).

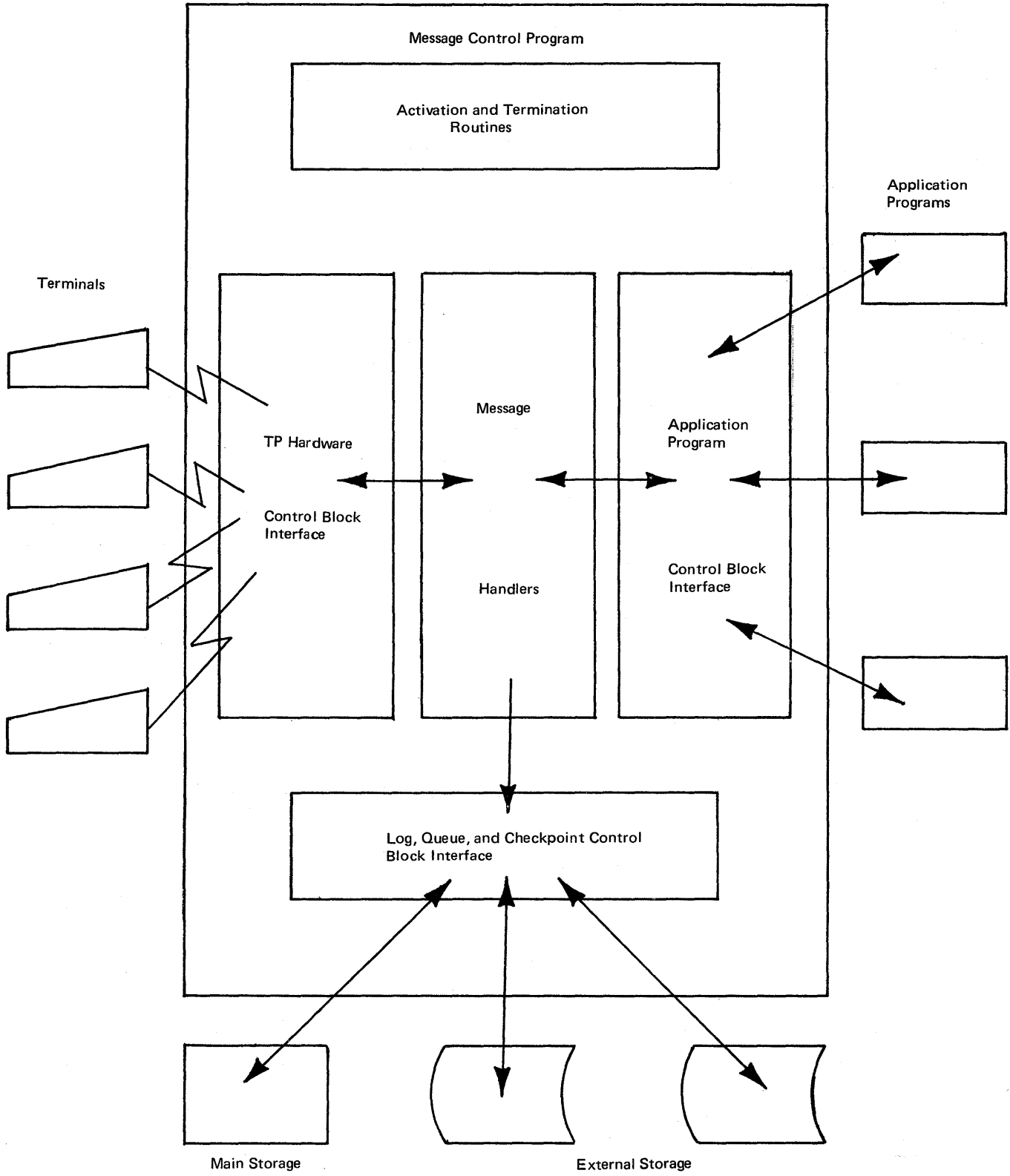


Figure 2. TCAM Interfaces

Through easy-to-use macro instructions, the user specifies his teleprocessing configuration and the main-storage areas required for message processing. These macros generate the tables and lists of control information that define the system environment.

Other macros allow the user to specify such functions as message routing, message editing, and error checking. In this way, the user tailors the system to his requirements. TCAM's high-level macro language permits the user to request functions without tedious coding and testing.

A teleprocessing control system created through the use of the TCAM macro language can:

- Establish contact and control message traffic between a central computer and local or remote stations;
- Insert and delete line-control characters automatically;
- Assign, use, and free buffers as required to hold messages during processing (dynamic buffering);
- Edit incoming and outgoing messages (for example, translate line codes to processing codes and vice versa, or insert or delete fields in messages);
- Forward messages to destination stations and application programs;
- Provide special handling for messages containing errors;
- Maintain statistical information about message traffic.

FEATURES OF TCAM

TCAM is a single system supporting multiple applications; it is not necessary to dedicate a particular terminal to a particular application. TCAM supports more types of terminals (including binary synchronous), and more types of programs and applications than any other IBM queued teleprocessing access method.

An important feature in TCAM is the separation of the teleprocessing network control performed by the Message Control Program (MCP) from the message processing performed by the application program. The two programs operate *independently* and *asynchronously*.

TCAM includes a comprehensive set of input/output, message-control, translating, error-handling, and editing routines that relieve the application programmer of the detailed programming usually required for a teleprocessing system. The application programmer can concentrate on the application program. The result is faster programming and installation. Finally, TCAM application programs need not be resident at all times, saving main storage.

The separation of message control from processing in TCAM also makes available the task-protection features of the operating system. The Message Control Program can be protected from damage because of a malfunction in an application program, and application programs can be protected from failures in each other. This task protection is

useful when installing new application programs by allowing them to run as separate tasks with their own interface to the Message Control Program.

The options provided in TCAM allow the user to trade-off several variables to design an MCP that is best suited to his application. For example, message queuing allows an access method to overcome the basic problems of a teleprocessing network: variable data rates and random inputs. When the volume of input messages exceeds the processing capacity of the teleprocessing system, the messages are held on a queue until the system can get to them.

Some other features of TCAM are:

- Recovery from line and terminal errors.
- Automatic retransmission of messages when transmission errors occur.
- Error awareness messages sent to the operator.
- Error recording by line and station to aid preventive maintenance.
- Multiple routing.
- Control program and application program checkpoints.

FEATURES UNIQUE TO TCAM

The features cited above are available with QTAM (Queued Telecommunications Access Method). The following features, however, are unique to TCAM:

- Provides a special buffering scheme that permits handling a wide range of data rates and message sizes concurrently. Buffering can be static or dynamic. Static buffering, in which sufficient buffers are assigned to a line for the duration of a transmission, requires less computer overhead; dynamic buffering, which assigns buffers as needed, is particularly suited to large or unpredictable message sizes and contributes to efficient use of main storage.
- Allows the user to take advantage of a variety of queuing options to tailor a system to an application. In addition to main-storage queuing, he can choose reusable disk queuing, in which disk storage is periodically reorganized to make storage available for reuse. The user can also select nonreusable disk queuing, or main-storage queuing with reusable or nonreusable disk backup. All types of queuing media can be used concurrently within the same system.
- Has improved efficiency through disk organization. Disk queuing can be implemented across several disk volumes instead of sequentially on one volume to optimize disk and channel performance. Multiple-arm support permits reading on one channel while seeking on another.
- Provides backup copies of messages on disks.
- Permits off-line application-program testing.

- Provides special facilities for inquiry applications. Several types of *lock mode* maintain the connection between a station and an application program for the duration of a message and one or more responses. *Conversational mode* allows a station to accept a text response to an inquiry message without first being selected.
- Brings queued, high-level teleprocessing support to binary synchronous and buffered devices, allowing TCAM to support many more *types* of terminals.
- Significantly increases speed and efficiency over QTAM.
- Expands control of individual stations. More functions can be initiated by a macro or operator.
- Permits specification of variable-size buffers for each line group, or station (output).
- Has increased maximum buffer size from 276 (for QTAM) to 65,000 bytes.
- Has five types of queuing including reuse of disk-storage space. TCAM permits all five queuing media to be used concurrently, if desired.
- Has several debugging aids.
- Can log complete messages.
- Implements the transparent line-control feature of Binary Synchronous Communications. In transparent mode, line-control can be added or deleted in the Message Control Program, relieving the application programmer of device-dependent considerations.
- Provides on-line station testing with other terminals operating normally.
- Can update control information by writing only a *data* field within a disk record.
- Restarts a channel program used in disk queuing from an “appendage” instead of issuing another Execute Channel Program command to the supervisor, reducing overhead.
- Restores most registers, making many more registers available to the user.
- Has *mid-batch recovery*: the capability to recover from permanent, text errors encountered in any block of data after the first block in a multi-block message, by restarting transmission with the block in error rather than retransmitting the entire physical message.
- Can process, on the incoming side, either user-defined logical messages or concentrated messages independently of buffer sizes and transmission lengths. On the outgoing side, can concentrate messages directed to a variety of destinations by way of a remote message deconcentrating device.
- Allows the operator to monitor and control the teleprocessing network from the computer console or a designated terminal.

- Takes, periodically, comprehensive (*environment*) checkpoints of the Message Control Program.
- Provides *incident* checkpoints to update major checkpoints.
- Allows use of a *cascade list* to accelerate message traffic. A cascade list allows a message to be sent to any one station within a group of stations. A message is queued for the station in the list having the fewest messages already queued.
- Permits simple conversion of QTAM application programs for running under TCAM.

TCAM RESPONSE TIME

Measurements show that the internal response time of TCAM is fast, and that this speed *can* be maintained at high message rates—regardless of the queuing medium selected—if sufficient storage resources are allocated.

On the other hand, response time can be jeopardized if inquiry/response data is queued on a DASD device that is forced to perform substantial seeking. This is because the DASD device contains data that is subject to large or varied amounts of delay in delivery (for example, bulk data transfer between time zones or reports whose printing is deferred until forms are mounted by the terminal operator).

To avoid delaying inquiry/response data in a system supporting multiple modes of communication, the user may wish to take advantage of certain TCAM facilities:

- Three queuing data sets—main-storage, nonreusable disk, and reusable disk—and the assignment of each *destination* in a TCAM system to the most appropriate queuing medium.
- The description of each station to TCAM as several *logical destinations*. Thus different types of output to the same station can be queued on a different data set.

APPLICATIONS

A teleprocessing system operating under TCAM can be designed for a wide variety of applications. These applications are handled or processed by the Message Control Program (MCP), the application program, or both.

Message Switching

Message switching is accomplished entirely within the MCP. Stations transmit messages to the central computer, which relays them to one or more stations.

When an incoming message is to be switched, the MCP routes the message to a queue for the destination station. If desired, information such as the date and time of receipt can be placed in the message, and the codes for both origi-

nating and destination stations verified. The MCP also can log the messages on a storage device for subsequent reference by the user with a different access method.

Message switching does not prevent a station from sending a message to an application program to be processed. TCAM places these messages on a destination queue for handling by an application program, either concurrently with the message-switching application or later.

Data Collection

Data collection also can be accomplished entirely within the MCP. Messages from stations are accumulated, queued, and processed as batch input.

The MCP can accumulate data in two ways: by placing it on destination queues for application programs, or by logging on a secondary storage device. If the first method is used, the messages can be obtained at any time by an application program. The messages are routed to destination queues as any other messages sent to an application program. The messages remain on these destination queues until an application program issues a series of GET or READ macro instructions to obtain the messages for processing.

In the second method, the MCP message-logging facility places the messages sequentially on a secondary storage device selected by the user. An access method other than TCAM must be used to retrieve them for processing.

Message Processing

A wide variety of teleprocessing applications can be processed by an application program; two of these are processing collected data and processing inquiries. If messages are collected on a destination queue, they remain there until an application program issues GET or READ instructions to obtain them for processing. The application program that processes the collected data can either:

- Operate concurrently with the collection of the data by the MCP; or
- Be loaded and initiated at a later time (for example, to process data periodically during the day).

If the data is collected, on a user-selected, secondary storage device by a logging macro, the data must be obtained for processing by an access method other than TCAM.

Inquiry Processing

In an inquiry application, the MCP receives messages, the application program processes the data contained in the messages, and the MCP sends replies to the originating stations.

The MCP receives the inquiry message from the calling station and passes it to the appropriate Message Handler within the MCP. The Message Handler determines which application program should receive the message, then routes the message to the appropriate application program.

The application program responds to the MCP, which returns the response message to the calling station by the appropriate transmission control unit. If the requested information is temporarily unavailable and the terminal is locked to the application, the application program should provide an appropriate message to let the calling station know when it can get the desired information.

An optional feature of the inquiry application is use of *lock mode*. In this mode, a station transmitting a message is held on the line by the MCP until the application program generates a response message for the inquiring station. In audio applications (which require lock mode) an application program can be used to receive and process audio inquiries by the MCP. Once a message has been processed, the application program returns an audio response message to the MCP for forwarding to the calling station.

Multiple Applications

A basic function of the TCAM MCP is to route incoming information to appropriate destinations. Since a destination can be either an application program or a station, all terminals operating under TCAM control have access to all destinations.

This section describes transmission techniques, transmission codes, line control, buffering, and queuing.

Readers familiar with teleprocessing do not need to read this section; they may skip to "TCAM Environment."

TRANSMISSION TECHNIQUES

The transmission technique is the way in which data characters are controlled on a communication line. The techniques used by computers and terminals are start-stop and binary synchronous communication (BSC); TCAM supports devices using either. BSC is used for high-speed data transmission between the central computer and a remote computer or high-speed terminal (for example, the IBM 1130 Computing System or 2780 Data Transmission Terminal). BSC establishes timing synchronization between sending and receiving equipment before the terminal sends a stream of characters. It requires more sophisticated electronics at the terminal than start-stop transmission but uses fewer control characters, thus more efficiently utilizing the communication facility.

Start-stop transmission is used for data transmission between the central computer and low-speed remote terminals (for instance, the IBM 1030 Data Collection System or 2740 Communication Terminal). Start-stop transmission gets its name from the "start" and "stop" bits, which it uses to delimit each data character.

The TCAM programmer need not be concerned with these techniques except to be aware of the control characters being used to regulate the flow of data on the communication line.

TRANSMISSION CODES

Data can be represented on a communication line by any of several transmission codes, depending on the kind of stations assigned to the line and the kind of adapters used to connect the line to the central computer. The programmer must be aware of the codes used on a line, because he must specify the translation tables to be used by TCAM.

These codes differ primarily in the number and pattern of the bits used to represent characters. A coded character may be classified as either a graphic character, representing a symbol, or a control character, controlling a terminal function.

In some codes, certain control characters are used to specify the way in which the following graphic characters are to be interpreted. Since each bit pattern can represent

more than one symbol, the number of characters that can be represented is greater than the number of distinct bit combinations.

The code most widely used in teleprocessing is EBCDIC (Extended Binary Coded Decimal Interchange Code), an eight-bit code with 256 possible combinations for text and control.

Another code important in teleprocessing is ASCII or USASCII (United States of America Standard Code for Information Interchange). This is a seven-bit code, with 128 possible characters. An eighth parity bit is often used for error checking.

Many teleprocessing systems accommodate several different codes by using translation tables for conversion, or translation, between different codes. For example, data may be received in one transmission code, converted to a code suitable for computer processing, and then converted to a third code for transmission to an output terminal.

LINE CONTROL

Most computer input/output equipment has traditionally been located near the computer, using separate lines for data and control. As communication equipment is connected to a computer by only one or two lines, conventions must be established to distinguish between the data information and the control information that must be transmitted.

Several stations may be attached to the same line. This reduces the cost of the communication facilities, but requires additional control conventions to determine which of several terminals is to send or receive data. Stations designed to operate on multipoint lines interpret characters received as control or text, according to the mode that the station is in at the time.

In a teleprocessing system, the central computer randomly receives data from remote stations during user-specified intervals. The operating procedures and control sequences used to manage a teleprocessing network are called line control (or, for BSC, data-link control). Some specific factors to be considered when designing teleprocessing line-control are:

- How contact is to be established between stations and the computer;
- How a message is to be directed to a specific station on a multipoint line;
- What happens if two stations try to send at the same time;

- What to do if a station fails to respond to a message;
- How to validate transmitted message data;
- Whether data is to be transmitted on lines attached to a message-concentrating device.

Line control can be classified by the transmission technique used for the line (start-stop or BSC). There is a set of control characters for each technique, although some of the characters are used by both. Line control also can be classified by line configuration (switched or nonswitched). In addition, individual variations arise from differences in the kinds of stations to be controlled and by the presence or absence of certain features on the stations. For example, line control may include the control characters needed to indicate a transmission error and to request retransmission, but some of the station equipment may not be capable of error checking or retransmission. All stations connected on a line must be designed to use the same type of line control and must have the same features.

In a teleprocessing system operating under TCAM, the handling of line-control characters may depend upon whether the messages are to be transmitted between similar stations (IBM 1050 to 1050, for example) or between dissimilar stations (a start-stop to a BSC station).

Messages between similar stations may have inherent line-control characters; these can be sent without changing the line control. Dissimilar stations may have different line-control characteristics; thus, the input message must have some or all line control deleted and appropriate line control inserted into the output message.

Messages between stations and application programs are treated as messages between dissimilar stations. In either case, the TCAM programmer needs to specify whether he wants line-control characters to remain, or to be removed and reinserted by TCAM.

For stations using the BSC technique, the line-control characters in a BSC message also may be transmitted in transparent or nontransparent mode, depending on the station's features. In nontransparent mode, all line-control characters are recognized and handled as for start-stop stations. (In transparent mode all characters, except for an initial line-control character designating the mode and an end-of-block signal, are handled by the station as data.)

Additional BSC information can be found in *General Information—BSC*, Order No. GA27-3004.

Errors are sometimes introduced into message data by unavoidable, transient line conditions such as cross-talk and lightning strikes. A teleprocessing system must be able to detect transmission errors and, when possible, to correct them by retransmitting the message or segment containing the error.

INVITATION AND SELECTION OF STATIONS

This section describes how the control of messages to and from stations in a teleprocessing system is accomplished by

invitation and selection. The central computer contacts a remote station either to allow the station to enter a message on the line (*invitation*), or to send the station a message (*selection*). A station is *invited* to enter a message on the line and is *selected* to have a message sent to it.

An initial contact must be established before the central computer can send a message to a station. The manner in which contact is made depends upon several variables, one being whether the station is on a switched or nonswitched line.

For switched lines, a preliminary step is also necessary before a remote station can enter a message. Although remote stations can call the computer at any time, the computer, in order to fulfill its control function, must be able to ignore incoming calls. Therefore, before a message can be sent from a remote station, the computer must condition the transmission control unit (or audio response unit) to accept incoming calls on a line. This is done by TCAM and is known as *enabling* the line. The computer's response is called *answering*. Introductory initiation and confirmation of the communication link is sometimes known as "handshaking."

If the line is not enabled, the calling station doesn't receive an answer and must try again later. If the line is enabled but is being used by another station, the calling station receives a busy signal and must try again later.

On both switched and nonswitched lines there are several methods of controlling a line. Polling and addressing are common to both types of lines. Identification exchange is restricted to switched lines.

POLLING

In *polling*, the computer periodically contacts the remote stations and invites them to enter any messages they have ready. Polling is accomplished by sending one or more polling characters. Often, two characters are used; the first to select a remote station, the second to select a specific component at that station. The polled station then indicates whether it has a message to send.

If the station is ready and has a message, it reverts to text mode and transmits the message. Other stations on the same line remain inactive. A code indicates the end of the message (EOM) and returns the station to control mode.

The computer may poll a number of remote stations and components until one is found with a ready message. If the computer does not receive either a message or status code within a set period of time, the station "times out" and is considered not ready. In TCAM, polling characters are maintained in an *invitation list* for each line.

In defining an invitation list for a nonswitched line, a user may enter station names as often as he wishes and in any order. Any station listed more than once will be polled more frequently than those listed only once.

For nonswitched lines, polling may be achieved entirely by programming or automatically by the Auto Poll feature

of the transmission control unit. A remote station cannot initiate contact on a polled, nonswitched line.

When more than one polled station is assigned to a switched line, a station calling the computer cannot uniquely identify itself; therefore, all such stations assigned to the same line must have identical polling characters.

The user may specify a polling delay (up to 255 seconds) at the end of the invitation list for a nonswitched line. When the delay expires, polling is resumed at the beginning of the list. This delay is used to minimize unproductive polling and to permit sending on the line. The polling delay may be altered by an operator command.

The user also may specify a system delay. If he does, the TCAM MCP begins a prespecified delay (up to 65,535 seconds) after current activity on all nonswitched lines has been completed. During the delay, no output is sent or stations polled. When the delay expires, line traffic is restarted. The system delay, as the polling delay, can be altered by an operator command.

ADDRESSING

Addressing is used by a computer to select remote stations for which it has messages. In a contention system (discussed below), the message is sent after the computer has successfully bid for the line. In non-contention systems, the procedure depends upon whether the lines are switched or nonswitched.

When the computer has a message to send on a nonswitched line, it sends one or more *addressing characters*. As in polling, two characters are commonly used: the first for the remote station, the second for the component. The remote station identified by these characters indicates in its response whether it is able to accept the message. If the remote station has given a positive response, the computer sends its message; if a negative response, TCAM can *hold* the message and try later.

When the computer has a message to send on a switched line, it may automatically dial the telephone number of the remote station after the line has been enabled (if needed), or it may wait for the station to dial. The function performed by the computer or station on a switched line is known as *calling*. The connection is established when either the remote station or the computer responds. The computer addresses the remote station and the message is sent.

CONTENTION

In a contention system, either the remote-station operator or the computer can initiate contact. A contention system can be set up to give either the computer or the station priority when they transmit simultaneously.

IDENTIFICATION EXCHANGE

Switched stations can identify themselves by automatically entering an *identification sequence* after a line connection

is established. TCAM compares the sequence it receives to a user-defined sequence. If there is a match, transmission begins; if not, TCAM disconnects the line.

BUFFERING

Two teleprocessing characteristics—random input and variable data rates—make the allocation of buffer storage critical in teleprocessing system design. For example, data could be lost in a teleprocessing system because of simultaneous input on multiple lines. In addition, insufficient buffers can delay responses and lower system efficiency. Two techniques to improve buffer utilization are using multiple buffers, which allows work to be performed in a buffer instead of in a work area; and, using groups of buffers (*buffer pools*) to supply buffer areas dynamically, rather than having fixed buffer assignments.

Buffer pools have an added advantage in teleprocessing systems. Because message sizes are highly variable in many applications, it would be inefficient to always assign a single buffer large enough to hold the largest expected message. Instead, each message is contained in a series of buffers allocated dynamically.

Buffer utilization would also be restricted by requiring buffers to be in contiguous storage locations. Each buffer is therefore allocated without regard for the location of previously allocated buffers, but is linked to the last and next buffer by *chaining*: that is, each buffer contains the storage address of the buffer containing the next segment of the message.

The sizes of the buffer pool and individual buffers depend on a number of factors, including the average and peak input/output data, the ratio of input volume to output volume, the data rates of the teleprocessing equipment, the amount of storage available, and the type of secondary-storage allocation.

The program must be capable of dynamically adjusting to unpredictable changes in the operating environment.

QUEUING

Queuing refers to the programming techniques used to handle transactions arriving at an unpredictable rate. Although the average rate at which messages arrive to be processed is usually less than the rate at which they are actually processed, queuing is needed to handle intermittent surges.

Like buffering, queuing is more important in a teleprocessing environment. There are several ways to handle queues in TCAM: in the sequence that messages are initiated (first in, first out: FIFO), in the order of completion (first ended, first out: FEFO); in reverse order of arrival (last in, first out: LIFO); or by priorities, depending on such factors as resources needed to process the queues, and whether the messages are input or output. The use of

FEFO in TCAM prevents delaying a short message while waiting for a longer one that started processing first.

While buffering may be accommodated in primary storage, it is more common to place queues in secondary storage, although sufficient control information for queue management is retained in primary storage. In addition, TCAM permits the user to recognize when queues are building up to critical lengths and allows him to modify normal program execution to reduce the backlog of queued items to a safer level.

Queuing in a teleprocessing environment creates some special problems not encountered with conventional computer systems. One is that a message may belong to more than one queue at a time. An example is a message for mul-

iple destinations. It would be wasteful to duplicate the entire message in the queue for each destination. Instead, only one copy of the message may need to be queued, with control entries referring to the message put in the pertinent output-queue control areas. The same message text can then be directed, at different times, to the various destinations. (This example assumes that each destination uses the same queuing medium; for instance, messages sent to each destination may be queued on reusable disk. However, if more than one medium is used, the entire message must be requeued on each medium.)

Another queuing facility necessary in many teleprocessing systems is the ability to handle high-priority messages.

THE OPERATING ENVIRONMENT

TCAM operates under control of the operating system with Multiprogramming with a Fixed Number of Tasks (OS/MFT-II) or Multiprogramming with a Variable Number of Tasks (OS/MVT), or Virtual Storage (OS/VS). OS/MVT is required with the Time Sharing Option. The Message Control Program (MCP) and application programs can be executed as separate jobs, or in the same partition after being invoked through a special system macro instruction.

The application programs operate asynchronously with the control program. The operating system supervisor controls the asynchronous operation of the various programming components of the system, according to:

- The established priorities of the various tasks;
- The completion of awaited events (such as I/O);
- The availability of resources (such as CPU time).

After being assembled and link-edited, in a library, a Message Control Program (MCP) can be initiated and executed in any partition or region. The TCAM MCP must be run with a higher priority than any TCAM application program and must be initiated before any TCAM application programs. However, TCAM's MCP does not always require the highest system priority.

If the TCAM jobs do not require the use of all available partitions (or regions), the remaining partitions can be used for other jobs. An example of such a configuration under MFT-II is: normal batch processing in three partitions, concurrent peripheral operations in a fourth partition, and an MCP performing teleprocessing functions in a fifth partition. Such a configuration is possible when the MCP can perform the required teleprocessing functions without an application program (for example, a message-switching or data-collection application).

Message processing can be done with only one application program to process all types of messages or a number of application programs to process various types of messages. Regardless of how many application programs are needed, they can be run in the same partition as the MCP, or in other available partitions or regions of the system.

TERMINAL AND LINE CONTROL

In constructing the Message Control Program, the user provides control information to identify remote stations, specify station characteristics to the system, and tell how they are to be handled by TCAM. It is not necessary for the programmer to specify the line-control scheme to be used for a

line; this information is provided implicitly at system generation and assembly times in the macro instructions for the line group and stations. The programmer must, however, have a general understanding of line-control concepts to correctly structure that portion of his program involved in message transmission, and to decide how to deal with line-control characters in his message.

In some cases, line-control characters may be left in a message. If the originating and destination stations use different line codes, the message must be translated from the line code of the originating station to EBCDIC, then to the line code of the destination station.

Contact between stations may be established in several ways, depending upon the line configuration and the stations involved. Contact is always established under the control of the central computer, which performs a number of preparatory operations.

In a TCAM system, either the computer or a station on a point-to-point contention line can bid for a line to send a message to another device. In some configurations, it is possible for both computer and station to simultaneously bid for the line.

In a contention system, invitation by TCAM gives a station an opportunity to enter data; that is, TCAM "listens" on the line for a signal from the station indicating that it has a message to send.

An alternative to the contention form of invitation is having the central computer periodically invite (poll) each station in an invitation list to enter input messages. After polling the active stations, TCAM allows the user to specify a delay to permit sending before polling is restarted at the beginning of the list. The polling interval also reduces unproductive polling on lines that are not in continual use.

For a station or application program, TCAM uses information provided by macros at assembly time and stored in control areas. The control areas used depend upon the terminal table (blocks of information about each station and application program). The size, structure, and contents of the terminal table depend upon information furnished by the user through macro instructions.

For descriptions of the *terminal entries* in the terminal table, see the *TCAM Programmer's Guide*.

DATA SETS

Seven types of data sets are used with TCAM:

- Line group
- Message queues

- Checkpoint
- Log
- Serviceability aids
- Input*
- Output*

The first two are required, the second three are optional features, and the last two used only with application programs. Except for data sets in main storage only, TCAM data sets are defined by the user by macro instructions.

Line Group Data Sets

A line group data set consists of the communication lines in a line group, up to 255 lines unless they have invitation lists. With minimum-length invitation lists, approximately 50 lines could be used. All the lines in a line group must have the same characteristics and features.

Message Queues Data Sets

Messages entered by remote stations are queued by destination: a line, a station on a line, or an application program. Because each incoming message is placed on a queue for its destination rather than being sent to the destination immediately, there can be overlap of line usage in I/O operations. A station can have several messages being entered simultaneously.

Queues for each destination and logging medium are located in one or more message queues data sets. These may reside:

- On reusable disk;
- On nonreusable disk;
- In main storage only;
- In main storage with backup on reusable disk;
- In main storage with backup on nonreusable disk.

The number of message queues data sets defined depends upon the type of queuing desired which, in turn, depends upon the application. For example, in a savings bank inquiry application short incoming messages (consisting, perhaps, of an account number, and transaction amount and code) might be sent to an application program, and long response messages returned. The TCAM user with such an application might wish to take advantage of the speed of main-storage-only queuing for his short input messages and queue on disk his long response messages, which would use up too much main storage.

Disk Queuing

Disk queuing is slower than main-storage queuing and requires disk and channel resources, but the delay is usu-

ally too small to be significant. In addition, disk queuing has certain advantages: more main storage is available to the user, messages can be held or retrieved, and checkpoint/restart can be used.

Disk queuing can be nonreusable or reusable. For a TCAM MCP that must run for an extended period with fairly heavy message traffic, the user would have to allocate more disk space if he used nonreusable queuing than if he used reusable queuing. In addition, a TCAM system using nonreusable disk queues must be closed down from time to time as the available space in the data set is exhausted. Non-reusable disk queuing is attractive for applications that have rigid retrievability requirements, that enqueue many messages for long periods of time, or that are periodically closed—as at the end of the day.

Increased disk and channel efficiency can be obtained by spreading the disk queues data set over several volumes (up to 16). Disks are organized to minimize rotational delay wherever possible. TCAM's multiple-arm support permits overlap of seek time on multiple volumes and channel operations on multiple channels.

A reusable disk can often handle the same amount of message traffic as a data set on nonreusable disk, while occupying less disk space. The data set never runs out of disk space under normal conditions. In addition, messages for an inoperative station can be sent to an alternate destination. TCAM's capability to retrieve messages that have been sent is limited with reusable disk queuing because the original message is eventually overlaid by another incoming message.

Main-Storage Queuing

Messages directed to queues in main storage will be received more rapidly than those sent to disk. Because allocation and deallocation of units for the main-storage data set is dynamic, the data set is essentially "reusable."

The user can specify a main-storage data set with backup on reusable or nonreusable disk. TCAM copies each message arriving at a main-storage queue onto disk. This type of queuing combines the advantages of both disk and main-storage queuing.

Data never has to be re-sent because main-storage units are not available, and TCAM's message-interception and message-retrieval functions can be used.

Checkpoint and Log Data Sets

The checkpoint data set consists of checkpoint records of the MCP environment maintained and stored on a direct-access storage device from which restart can be made in case of closedown or system failure.

A log data set is made up of messages or message segments placed on a secondary storage device for accounting, audit-trail, or back-up.

*See Defining Application Program Data Sets in the chapter "Application Program."

BUFFERS

Messages entering a TCAM network are read into buffers. These are main-storage areas that handle message segments between lines and queues (in storage and on disk), and between queues and application programs. (A message *segment* is the portion of a message contained in one buffer.) A buffer has two parts, an area containing control information (*prefix*) and an area containing all or part of the message. Buffers must be at least 35, but no more than 65,535 bytes.

To provide efficient buffering and best use of main storage, TCAM maintains a buffer unit pool containing buffer *units* of one size. Buffer units are the basic building blocks from which buffers are constructed. Their size (a maximum of 255 bytes) and number (a maximum of 65,535) are specified by the user.

When message traffic is in progress, a buffer unit, originally part of the unit pool, can be assigned to main-storage message queues, line group buffers, or an available-unit queue.

By linking an appropriate number of buffer units, TCAM constructs buffers as large as specified by the user. For example, if the user specifies a unit size of 60 and a buffer size of 120, TCAM links two 60-byte units for that line group.

Program-Controlled Interruption (PCI)

If program-controlled interruptions are not specified, then the number of buffers assigned initially must be sufficient to handle the entire incoming transmission to avoid losing data; no buffers are freed until either the transmission is completed or an end-of-block control character is received.

When program-controlled interruptions are specified, the user may request one of two options:

1. A relatively small number of buffers, allocated initially to handle the I/O operation, and more buffers to be allocated by PCI interrupts as needed.
2. A fixed number of buffers, sufficient to hold the entire message being sent or received, to be available before I/O operations begin.

Dynamic allocation (option #1) results in fewer buffers being tied up at any one time in an I/O operation than static buffering (option #2); however, dynamic operation requires more CPU time and may result in incoming data being lost because of the inability to replace buffers as fast as they are filled. The user can minimize this possibility by assigning more buffers to his line, making his buffers larger, or increasing the number of units in his unit pool.

The Message Control Program (MCP) is a set of routines that identify the teleprocessing network to the operating system, establish line control for various kinds of stations and connections, and control the handling and routing of messages (Figure 2). The MCP is an intermediary between the remote stations, and between remote stations and application programs. MCP routines perform input/output using information specified in MCP macro instructions.

Depending upon the requirements of the user, the TCAM MCP can perform any of the following functions:

- Enable or disable communication lines.
- Invite stations to send messages.
- Send messages to stations.
- Assign and use buffers as needed during program execution (dynamic buffering).
- Route messages to appropriate queues.
- Queue messages on a direct-access storage device or in main storage.
- Intercept or cancel messages containing errors, keep counts of errors, transmit error messages, and initiate corrective action.
- Reroute messages to alternate destinations.
- Provide message data to an application program.
- Place response messages from application programs on queues for transmission.
- Provide operator-to-system communication and control from remote stations.
- Take periodic checkpoints of the system (save status of queues and of the teleprocessing network on a direct-access storage device) for recovery in case of failure.
- Edit incoming and outgoing messages, including: translating from transmission code to EBCDIC and vice versa, inserting and deleting line-control characters, inserting date and time information in a message header, and inserting or removing user-specified data in a header.
- Log messages on secondary storage.
- Maintain a count of the number of messages received from or sent to each station.

To provide maximum efficiency, TCAM places messages in queues in main storage or on direct-access storage devices. Messages are recalled from the queues for either processing or forwarding to some destination.

The MCP itself can perform limited processing of a message. For instance, the MCP can examine the initial bytes of data in a message to obtain routing and translating information. Certain optional processing is also provided by the MCP, such as inserting the date and time of day in messages—eliminating the need for an application program to do this. The MCP can also check the input messages to determine if an error message must be sent to the input station.

An MCP is open-ended in the sense that the user can include functions not provided by TCAM by employing operating system and assembler language instructions. The user can use this information to update special counters, change various fields in main storage, and specify various functions in the message-handling section of his MCP that normally would not act on the message being processed.

With macros, the user defines data sets, and specifies equipment configurations and storage areas. These macros generate the tables and lists of control information that define the system's environment. The number and sizes of buffers are specified by the user and allocated from a single buffer unit pool as needed.

TCAM MESSAGES

Messages are sent to users and application programs. Depending upon the application, messages may consist of a header, text, or both. The header may contain control information such as originating and destination stations, and message type and number.

The header's length, format, and content depend solely on the requirements of the application and the user's preferences.

The user must specify control characters appropriate for the type of message unit. The types of units and their corresponding control characters are:

- A block is that portion of a message terminated by a line-control character—EOB, ETB, ETX, or EOT. A subblock is that portion of a BSC message terminated by an ITB line-control character.
- A segment is that portion of a message contained in a single buffer. Its size is specified by the user for each line group and application program.
- A record for an application program is, most often, that portion of a message terminated by the last data byte in the last message buffer; less often, it is a portion of a message terminated by a user-specified data character.

- A *physical* transmission is a unit of data terminated by an EOT character or, if the user specifies TCAM's conversational facility, by an ETX, ETB, or EOB character. A *logical* message is a unit of data delimited by a user-defined length or end-of-message character(s) and formed by blocking or deblocking physical transmissions. A physical transmission can contain several logical messages; or, several physical transmissions can be combined into one logical message.

TCAM, with its complete set of header-processing routines and associated macro instructions, allows the user to indicate the header-processing functions by including the appropriate macros in the MCP. In addition to the functions briefly described, many others are available, such as the detection of incorrect information in the header fields.

MESSAGE HANDLERS

Messages are processed by routines called Message Handlers, which direct all MCP—and TCAM—operations. They interpret and process control information in the headers and prepare messages for forwarding to destinations.

Each Message Handler services a particular line group, or several line groups with similar characteristics. A Message Handler is divided into two groups, one for incoming messages and one for outgoing messages, and these are subdivided into:

- An inblock subgroup, which handles incoming physical transmissions before they are divided into logical messages;
- The inheader and outheader subgroups, which handle only the headers of incoming or outgoing messages;
- The inbuffer and outbuffer subgroups, which handle all message segments;
- The inmessage and outmessage subgroups, which specify actions to be taken after the entire message has been received or sent (for example, checking for specified errors and sending an error message to the source or destination).

A group can have several subgroups of one type. However, not all of the subgroups in a group may be needed to handle a message. See Figure 3, showing some possible incoming subgroup combinations. Outgoing subgroups can be handled similarly.

A major source of Message Handler flexibility is the *option-field* capability offered by TCAM. These user-defined fields can be set and examined by TCAM to govern the execution of conditional functions, as well as to supply data to user-coded routines.

The user can design a Message Handler to accept messages with a variety of header formats and to perform different operations upon different types of messages, even when transmitted over the same line. The processing performed

by a Message Handler can be varied in two ways: by using control characters in the message header and by setting switches (one-byte fields in the option table), based on the control characters, that determine whether a given subgroup is to be executed for a message. These switches can be specified, tested, and changed to determine which of the subgroups in a Message Handler are to be executed for a particular message.

The path of a message *within* an inheader or outheader subgroup can also be controlled by the user. In addition, several Message Handler macros can be executed conditionally, depending upon a control field in the message being processed.

The Message Handler also supports user-defined logical messages and concentrated/deconcentrated messages, and mid-batch recovery. The inblock subgroup handles logical messages. Logical messages and concentrated messages are discussed later in this chapter. Mid-batch recovery is discussed in the chapter on "TCAM Service Facilities."

To supplement TCAM-provided functions, the user can code subroutines of assembler and operating system macro instructions and include them in his Message Handlers.

Message Handlers are defined by sequences of macro instructions of two types: functional macros that perform the required operations and delimiter macros that direct control to the appropriate sequence of functional macros.

MESSAGE FLOW WITHIN THE SYSTEM

TCAM message flow is shown in Figure 4.

Steps 1 and 2: The input message is prepared at a remote station and entered on a line. The message can be keyed in, or entered from a keyboard-printer, a card or tape reader, a computer, a badge reader, etc.

Step 3: The message enters the central processing unit (CPU) and is stored in a main-storage buffer assigned to that line for input. TCAM inserts chaining addresses and other control information into the buffer prefix field, which has been allowed for by TCAM and stored with the message. TCAM can also reserve space in the buffer for later insertion of the time, date, message sequence number, etc.

Step 4: The incoming message is passed, a buffer at a time, through the Message Handler's incoming group, which performs such user-selected functions as checking origin and destination, inserting the time, translating from line code to EBCDIC, and initiating an error message to be sent to the originating station. The order in which the Message Handler's incoming group scans the header fields can be varied to alter operations in the buffer.

Step 5: After processing by the incoming group of the Message Handler, the message is placed on a destination queue for either an application program or an accepting station.

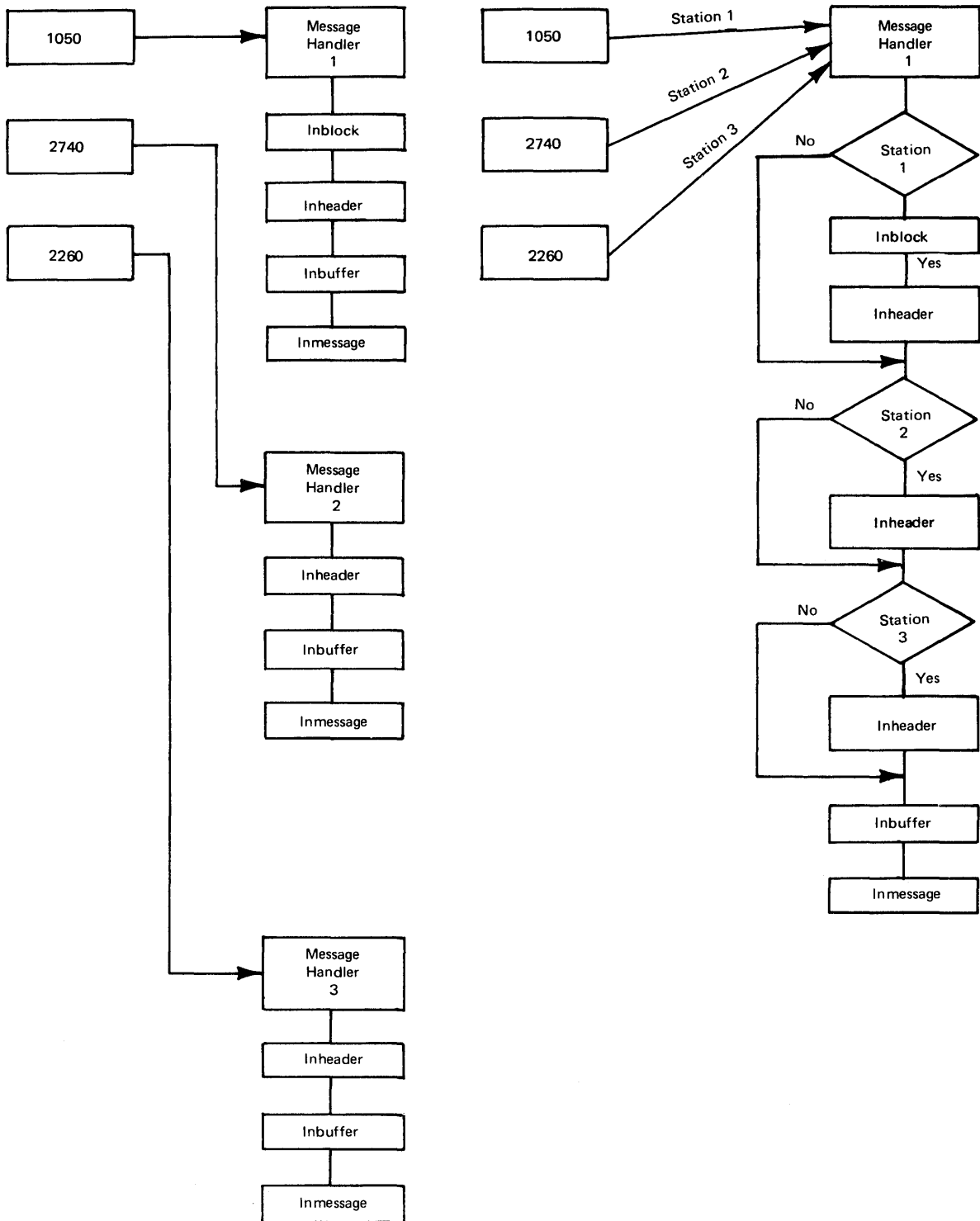


Figure 3. An Example of Incoming Message Handler Subgroups, Using One and Multiple Message Handlers

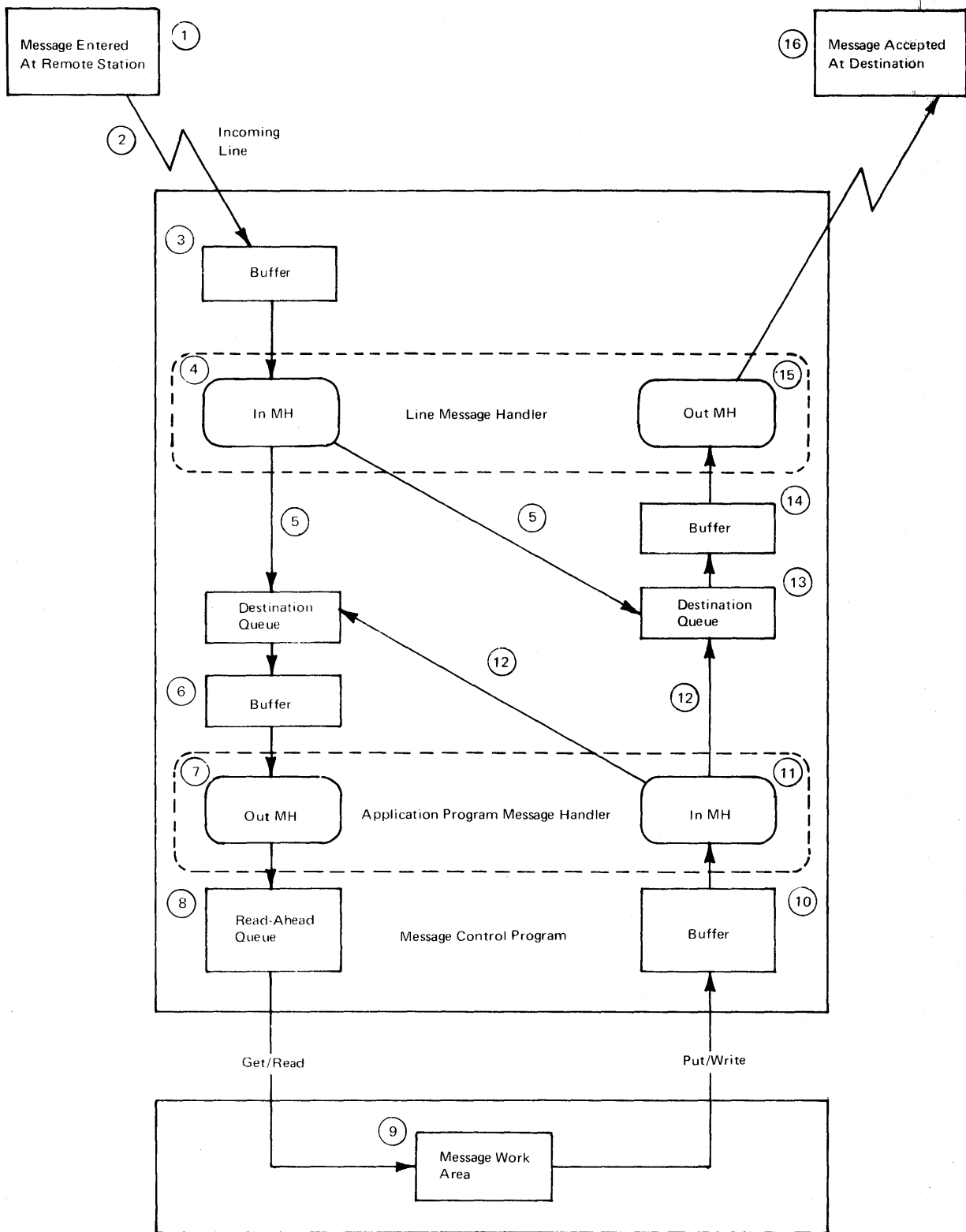


Figure 4. TCAM Message Flow

If the message requires no further processing and its destination is a station, the next operation is Step 13.

The destination queue for an application program consists of queues of message segments on a direct-access storage device or in main storage. If the destination queue is located on disk, the buffers are released; if in main storage, the buffers are used to contain the message in the main-storage queue.

All messages requiring text processing are routed to the destination queue for the application program that processes that type of message. The user controls this routing by message header and Message Handler macros.

Steps 6, 7, and 8: The message from a destination queue for an application program is placed in main storage buffers; the outgoing group of a Message Handler is placed on the *read-ahead queue*, an area of main storage from which an application program obtains work units for moving to a work area.

Step 9: TCAM passes message data to a user-specified work area in the application program. TCAM removes the prefix and the application program processes the message. The application program may return a response message to a station or another application program for further processing.

Steps 10 and 11: TCAM moves the data from the application program work area into an MCP buffer where it is handled by the incoming group of the Message Handler application program. Header or text prefixes are created, and chaining addresses and other control information inserted, as for messages coming in from a line.

The response message generated by an application program can be any length, selected by the user. After the buffer is filled, it is handled by the incoming group of the Message Handler assigned to the application program by an MCP macro.

Step 12: If further processing of the message data by another application program is required, the message is queued for that destination, and Steps 5-11 repeated. If not, TCAM places the processed message on the destination queue for the accepting station.

Step 13: The destination queue for an accepting station, like that for an application program, is part of the message queues data set and can be either in main storage or on a direct-access storage device. TCAM obtains message segments from the destination queue in first-ended, first-out (FEFO) order within user-defined priority groups.

Steps 14 and 15: The message segment is placed in a buffer, and the outgoing group of the Message Handler for the line processes the message. The Message Handler performs such user-selected functions as converting the code of the message to the transmission code for the station (if necessary), inserting the time and date in the header, logging messages, and updating message counts.

Step 16: TCAM transmits the message, minus the prefixes, to the destination station. As each buffer is trans-

mitted, its units are returned to the buffer unit pool for reuse.

MAINTAINING ORDERLY MESSAGE FLOW

Among the factors influencing the flow of messages within a TCAM system are message priority and queuing, and transmission priority.

Message Priority and Queuing

Message priority is the relative order in which messages are sent out over a line. Message priority depends upon:

- Whether queuing is by line or station (or groups of stations in a concentrated message environment);
- The relative order in which messages are received at the destination queue;
- The priorities assigned to messages.

Outgoing messages can be queued by destination line or destination station. When queued by line, messages for all stations on the line are placed on one queue. Messages are taken off the queue and sent to stations on the line in the order received (first-ended, first-out) within priority groups. Thus, for any multipoint, nonswitched line, all messages of one priority on the queue are transmitted before any messages of a lower priority, even if the destinations of the higher-priority messages are two different stations on the line.

When outgoing messages are queued by station, one queue is created for each station on a line. All messages queued for one station are sent before any queued for other stations on the same line. (The user can modify this technique for nonswitched lines by using a TCAM macro to limit the number of messages to be sent to a particular station during one contact.) Messages on a queue are sent to a station in first-ended-first-out (FEFO) order within priority groups.

For a multipoint line, the relative order in which queues of messages are transmitted is also determined on an FEFO basis. Switched stations and buffered terminals must be queued by station.

Messages to be sent to an application program are placed on a queue for that program and are removed as if they were messages queued by station; that is, they are sent to the application program on an FEFO basis within priority groups.

Transmission Priority

Transmission priority is the relative priority between sending and receiving on a line, specified by line-group by the user. Figure 5 shows the types of transmission priority available for BSC and start-stop devices on switched and nonswitched lines. There are three types of transmission priority:

Transmission Priority	Non-Switched Lines		Switched Lines	
	BSC	SS	BSC	SS
1. Send	X	X	X	X
2. Equal	X	X	X	No
3. Receive	X	X	No	No

Figure 5. Transmission Priority for Various Types of Lines and Transmission Techniques

- Computer receiving has priority over computer sending.
- Receiving and sending have equal priority.
- Sending has priority over receiving.

Transmission priority has a different meaning for each of the three configurations of stations supported by TCAM:

1. Polled stations on a nonswitched, point-to-point or multi-point line.
2. Contention stations on a nonswitched, point-to-point line.
3. Stations on a switched line.

For nonswitched polled stations, the user may specify any of the three priorities above.

TCAM provides buffered terminal support for the IBM 2740 Model 2 and the IBM 2770. For buffered terminals, either equal or send priority should be specified for polling.

For nonswitched contention stations, either send or equal priority can be specified. The IBM stations in this category are the 2740 Communication Terminal (Basic), the 2780 Data Transmission Terminal, the 2770 Data Communication System (without multipoint data-link control), and the Model 20 and 1130 computers used as terminals.

For switched stations, send priority must be specified for start-stop devices, send or equal for BSC. The relative order in which messages are sent to and received from a station on a switched line depends upon whether the station dialed the computer or vice versa.

HANDLING LOGICAL MESSAGES

Some terminals accumulate data in buffers before it is transmitted and forward no more than the content of one buffer

in a transmission. Thus, if a user's message is longer than the buffer, his message will require two or more transmissions. What the user sees as one message may be seen by TCAM as several. Alternately, some terminals permit many units of data, such as cards, to be batched and sent as a single transmission. TCAM may interpret this as one message, although to the user it may be many messages.

Logical message definition enables TCAM to process logical messages, as defined by the user, independently of the boundaries imposed by the associated physical transmissions; when several transmissions are required for a logical message, TCAM blocks the transmissions and then processes and queues the data as a single message.

When a transmission contains several logical messages, TCAM deblocks the transmission and independently processes and queues each logical message. If part of a logical message has been received from a station, output to that station is not sent until the end of the logical message has been received.

The basic concepts of the TCAM design need not be modified; messages received from terminals are still processed through a message handler and are queued in destination queues from which they are sent to an application program or another terminal. A separate subgroup, inblock, furnishes blocking and deblocking, and converts the physical transmissions from terminals into logical messages according to user specifications.

The user may specify which delimiter character or data length, or both, determine the completion of a logical message. When both are specified, the first condition met determines the end of a logical message. The user specifies the handling of any remaining portion of a physical transmission

after an end-of-message character. He can either discard this remaining portion or process it as the first buffer of another message.

MESSAGE CONCENTRATING

TCAM supports the concentrating of incoming messages into single physical messages on non-switched lines between a concentrator and a computer. When incoming messages from several terminals are sent to a remote concentrator for forwarding to the computer, TCAM extracts the messages and directs them to their appropriate incoming Message Handlers. Conversely, a Message Handler can concentrate outgoing messages into a single physical transmission for a concentrator, which can forward the various transmissions within the physical message to their appropriate destinations.

Concentrators decrease line costs by substituting one high-speed line from the concentrator for several low- or medium-speed lines, from isolated terminals to the central computer. The central computer may be programmed to act as a concentrator.

Messages can be transmitted to concentrators by using normal TCAM line-scheduling techniques. When concentrating messages, the user can have terminals share a common destination queue, or specify a separate destination queue for each attached terminal. No destination queue is required for the concentrator itself.

Once an outgoing message is removed from a destination queue and forwarded to a concentrator, the user can delay execution of the outmessage subgroup to examine incoming status information from the concentrator and its attached terminals, indicating whether the terminals are ready to receive. If the last message was received incorrectly or the printer was not operating correctly, for example, the user might want to delay transmission and take some kind of action.

The send, equal, or receive transmission priorities are handled in exactly the same way as in normal message queuing. Further, terminals attached to a concentrator can be included in the same line group with terminals not attached to a concentrator. Operator-control and application-program facilities allow the user to intercept or release messages destined for terminals attached to a concentrator simply by intercepting or releasing the concentrator.

Output to a concentrator consists of one or more queues of messages destined ultimately for terminals attached to the concentrator. TCAM removes these messages from the destination queues associated with the attached terminals and blocks them into a single, concentrated message for transmission to the concentrator.

TIME SHARING OPTION

A Time Sharing Option (TSO) provides general-purpose, time-sharing facilities in a System/360 MVT or System/370 VS2 environment. TCAM controls I/O operations for the TSO user. TSO jobs may be run in either of two environments, a TSO-only environment or a mixed TSO/TCAM environment.

In a TSO-only environment, time-sharing (or *foreground*) operations can take place concurrently with the execution of standard batch (*background*) jobs; thus, the system need not be dedicated to time sharing. While stations are logged-on only for time-sharing, they are dedicated to TSO Message Handlers.

When operating in a mixed environment, TCAM and TSO tasks concurrently share all necessary resources. The user can specify an IBM-provided Message Handler routine to relieve him of the task of designing his Message Handler routines for handling TSO messages.

Messages can be processed by application programs written by the user to suit his applications. (Some partial message processing is also done by the Message Handlers in the MCP.) TCAM passes messages and inquiries to the application programs for processing and returns messages or responses to their destinations. TCAM also provides a method of tailoring buffers to application programs.

TCAM application programs need not be concerned with the station at which a message originated, with the line-transmission code, or with previous line control, because line control is handled automatically by TCAM.

TCAM allows the user to debug his application programs in a nonteleprocessing environment using BSAM or QSAM as the access method, and a tape, card-reader, disk, or card punch for input, and a printer for output. Once debugged, many application programs can be added to TCAM, without reassembly, by changing a JCL (Job Control Language) card.

TCAM also permits the user to retrieve a message from a message-queues data set on disk after the message has been sent to its destination. In addition, TCAM has facilities for inquiry/response for start-stop devices and for coordination of MCP and application program checkpoint/restart. TCAM also provides teleprocessing-network control from an application program: examining or modifying the contents of a control block, releasing messages queued for an intercepted station, and closing down the MCP. All operator-control functions are available from application programs.

Application programs written to run with a QTAM MCP can be used with TCAM. If certain macros have been used, the QTAM application program need not even be reassembled; otherwise, most QTAM macros can be reassembled under TCAM, using a special TCAM macro as the first instruction. The modified application program will operate in most respects as under QTAM.

Application programs run asynchronously with the MCP, usually in another partition or region, but always as a separate system task or subtask. Their priority must be lower than the MCP, which must have control after a system interruption. If TCAM application programs are not in a separate region or partition, they are attached by operating system macros coded in the MCP by the user. The programs can be activated any time after the MCP is activated, and can be deactivated independently of the MCP.

Messages to be processed are placed in a destination queue by a Message Handler and routed to the application program. The messages are obtained from the destination queues and transferred to user-specified work areas, where they are processed by application programs. A response message may be returned to the MCP for transmission to a station, a list of destinations, or another application program.

In some applications, one destination queue can handle all the messages, and a single application program having a single interface with the MCP can perform the processing. If various kinds of processing are required, there are two methods of providing them:

- Each of several application programs can be provided with its own interface with the MCP and messages routed to the destination queue for the desired program.
- All messages that require processing can be routed to the same application program, where a user-written routine determines the kind of message. Messages are transferred by this routine to appropriate processing routines, or to a processing program in another partition or region.

Application programs can be "multi-thread," that is, capable of concurrently processing several transactions, using multiple interfaces.

THE MCP/APPLICATION PROGRAM INTERFACE

The TCAM MCP routes messages between an application program and other application programs or remote stations. Because an application program depends on the MCP to perform its input/output operations, it must have an interface with the MCP. TCAM provides a number of services that allow this interface to be established from an application program.

DEFINING WORK AREA AND UNIT

The amount of data transferred from the MCP to an application program by a single GET or READ macro, or from an application program to the MCP by a single PUT or WRITE macro, is called a *work unit*. The work unit is processed in an application-program *work area*.

A work unit can be an entire physical message, or a portion of a physical message, which may or may not be a record. A *physical message* is the unit of sent data terminated by an end-of-transmission, line-control character, or a unit of received data, defined by the user, that may or may not be terminated by an EOT character.

A *record* is a logical unit of data whose length is defined by macro operands and delimiting characters in the message. In TCAM, each record is transferred as part of a message, but one message can contain many records.

Just because a work unit is not an entire message does not mean that it is a record. If message processing is specified, but the entire message does not fit into the work area, TCAM provides the capability of processing a portion

of the message in the work area, then bringing in the next portion and processing it, until the entire message has been processed. The portions of the message processed in this way are not considered to be records, since message processing rather than record processing was specified; TCAM handles records and other message portions differently.

SPECIFYING APPLICATION-PROGRAM WORK UNITS

How TCAM handles work units depends upon:

- The *format* of the work unit (fixed, variable, or undefined);
- The *type* of work unit (message or record).

The user specifies the format and type of work units his application program is to process by coding operands of input and output macros for the application program. These indicate whether the work unit is a message or a record, and whether it is always the same length or may vary in length from message to message or from record to record.

Messages going to and from an application program, as those associated with stations, are placed in buffers. Buffer allocation for application programs is always dynamic. The buffers can be tailored for the application program and need not be the same size as the line buffers. The user assigns a maximum number of buffers to handle messages being transferred from MCP process queues to an application-program work area.

This technique allows overlapping of message processing in the application program and message acquisition in the MCP. If only main-storage queuing is used for process queues, it is necessary to specify only enough buffers to handle the largest work unit to be sent to the application program for optimum performance.

MESSAGE RETRIEVAL

TCAM application program macros can be used to retrieve one or more messages already on a data-set destination queue on disk (but not messages queued in a main-storage-only data set). After retrieval, messages can be processed and sent to any destination.

INQUIRY/RESPONSE

Two complementary functions, lock mode and conversational mode, can be used with start-stop, unbuffered terminals to prevent intervening traffic between an inquiry and its response. In *lock* mode, TCAM maintains a connection between a station and an application program for one or more conversational interactions. In *conversational* mode, a station is able to accept a text response to an inquiry message without having to be selected before receiving the response.

For the fastest response in an inquiry application, the station remains on the line until an application program returns the required information. While the station is in lock mode, no incoming messages are accepted from any other station on the line, and no outgoing messages other than the response message are sent to any station on the line (including the station in lock mode). Many stations on other lines may be simultaneously locked to the same application program.

There are two types of lock mode: line lock and terminal lock. In *line lock*, a physical connection is made between a station and an application program, and maintained for the duration of the conversational interaction. When a line is locked to a station, it is unavailable to other stations.

There are two types of line lock: message lock and extended lock. In *message lock*, the connection is maintained while the entire message is sent to an application program and the response message received. The first message from the application program to arrive at the destination queue for the locked station is assumed to be the response. The line is automatically freed when the response has been sent.

In *extended lock*, the same station is polled again after the response has been sent to it. If the response is positive, the next inquiry message is entered by the station. If the response is negative, the station is repolled until a positive response is received. This facility is also used to guarantee that a switched connection will be maintained until output is started for a station.

Once a station is in extended lock, all messages entered by it are assumed to be inquiry messages directed to the application program to which the station is locked. Destinations specified in the headers of messages are overridden when the station is in extended lock mode.

Message lock is used if a single inquiry will provide all the information required. For instance, an inventory application might handle inquiries requesting the quantity of a certain part in stock. Extended lock is used if a series of inquiries must be made, each requiring a response. In a credit application for example, the inquiries might ask if a person has an account, await verification, and then request the credit balance.

A Message Handler macro is used to remove a station locked to an application program from extended lock mode. It may be issued either unconditionally or conditionally.

Terminal lock establishes a logical connection between a station and an application program so that the station receives responses to its inquiries without tying up the line. Terminal lock assures the station that the next message it gets after it enters an inquiry is the response to that inquiry, but does not render the line unavailable to other stations while the inquiry is being processed. Terminal lock can be used by stations taking advantage of TCAM's buffered terminal support; line lock cannot.

Conversational mode operation is a hardware feature specified by a macro operand. When the computer receives a

message from a station using this feature, it sends a response message, rather than a positive acknowledgment, to the station. The station interprets the message as a positive acknowledgment, saving two-line-turnaround sequences.

COORDINATING TCAM AND OPERATING SYSTEM CHECKPOINTS

The TCAM checkpoints of the Message Control Program can be coordinated with operating system checkpoints of TCAM application programs by issuing macros in the ap-

plication programs. This allows the MCP and an application program to be restarted at the same point following system failure.

If the MCP fails, the application program data sets are automatically closed; after the MCP is restarted, the user can restart his application program. However, failure of the application program need not be accompanied by failure of the MCP. In some applications, the user might wish to close down his MCP following abnormal termination of an application program, so that both might be restarted from the same point.

TCAM supports telecommunication systems with some facilities specified by the user and others provided automatically by TCAM. They include operator control, network reconfiguration, message logging, I/O error recording, additional debugging aids, error recovery procedures, check-point/restart, and on-line test.

OPERATOR CONTROL

Operator control allows the user to examine or alter the status of a telecommunication network in a variety of ways. Operator commands can be entered from the system console, remote stations, and application programs.

TCAM operator commands can be used to provide any of the following functions:

- Display line or station addresses and status, option fields, error records, type of polling, and message count.
- Activate and deactivate stations.
- Activate and deactivate lines.
- Activate and deactivate invitation-list entries.
- Hold messages in a station queue.
- Release messages queued for a station.
- Alter option fields.
- Switch between Auto Poll and programmed polling.
- Change a secondary operator control station to a primary operator control station.
- Interrogate the status of a line, queue, or station.
- Initiate system delays.
- Alter the duration of the programmed polling delay or system delay.
- Start or stop TCAM debugging aids.
- Return incorrectly formatted commands.
- Provide responses in English.
- Initiate system closedown.

Operator Control Stations

In addition to the system console, various nonswitched stations can be designated as operator control stations: for example, the IBM 1050 Data Communication System, 2740 Communication Terminal, 2780 Data Transmission Terminal, AT&T 83B3 Selective Calling Station, and Western Union Plan 115A Outstation.

The user may define one primary control station and any number of secondary control stations.

A secondary operator control station can send control messages and receive responses. In addition to performing all the functions of a secondary operator control station, a primary operator control station can receive I/O error messages. If the primary control station becomes inoperative, the system console can also accept internally generated error messages, even when being used as a secondary station.

The system console is defined at system generation time and can include a card reader and other I/O devices. The system console can be specified as a primary operator control station; if it is not specified, it is secondary by default.

Responses to operator commands are placed on the destination queue for the station that entered the command and are sent through the outgoing Message Handler as normal messages. If selective execution is required, macros can be used to detect operator responses and to vary the processing path.

Operator Commands from an Application Program

An application program can enter an operator command by directing it to the application program's Message Handler. The operator command from the application program is then treated like any other operator command.

USE OF ALTERNATE DESTINATION

The TCAM user can issue a macro in either the incoming or outgoing group of a Message Handler to indicate an alternate destination when specified error conditions are detected. The alternate destination may be another station, including the originating station, or it may be an application program.

NETWORK CONTROL

TCAM provides facilities for dynamically controlling the telecommunication network through macro instructions issued in an application program or by operator commands. Facilities include stopping lines or line groups, starting or stopping a station on a line, removing a station from an invitation list, holding up transmission to a station or resuming transmission, changing the type of polling or length of the polling delay, activating or changing the system interval, inserting data in a station option field, changing primary operator control to another station, or initiating closedown of the Message Control Program.

There is protection against unauthorized use of some of these macros. In addition, TCAM allows the user to define his application program as a primary or secondary operator control station from which operator commands can be executed or error-awareness messages for the operator queued.

The user might design a special application program solely to modify the teleprocessing system if errors or other unusual conditions occur. For example, he might code macros in an inmessage subgroup handling messages coming in over a line group. These macros could test various TCAM bits and, when these bits were on, direct a special error message (or the message being handled when the error occurred) to an application program process queue. The application program could fetch error messages, analyze them, and issue operator commands (if designated an operator control station) or network-control macros to modify the system in a manner appropriate to the error. Large networks, especially, benefit from TCAM's capability to have an application program designated as the primary operator control station. These include networks in which elementary error conditions (for example, polling time-outs) could "flood" an operator control station, or networks in which "automatic" operator response to network status is desired.

LOGGING

TCAM's message-logging facility enables the user to keep a record on a sequential data set of either messages or message segments handled by a Message Control Program. Message logging can be useful in accounting, for long-term back-up, or collecting for exceptional cases. In message switching, the logging of incoming headers may supply all the information needed. In more sophisticated applications, it may be necessary to log complete messages. By determining the flow patterns of message traffic, a programmer can more efficiently allocate the resources of a telecommunication system.

Logging can also aid debugging by allowing the programmer to trace the flow of messages through a Message Control Program.

I/O ERROR RECORDING

TCAM assists error diagnosis by providing I/O error-recording for certain terminal-related I/O errors. Different types of I/O error recording can be initiated by an operator command. A *permanent* error is either an *irrecoverable* error (one for which TCAM provides no recovery procedure), or an error that TCAM has tried several times to correct without success.

Either a station or a line can be monitored. A temporary error record is made each time that a specified error occurs

and is recovered from. The user may request that TCAM check any of ten types of errors: time-out, lost data, overrun, data check, equipment check, bus out, intervention required, command reject, unit exception, or unusual leading graphic response from an IBM 2740 Model 2 terminal. Alternatively, the user can specify that a temporary-error record is to be made if *any* of the ten types of errors occurs and is recovered from (except for the 2740 graphics response). The user also selects the number of times (from 1 to 15) that a temporary-error record is to be made when the specified error occurs and is recovered from for a particular line or station.

ADDITIONAL DEBUGGING AIDS

TCAM provides, in addition to message logging, error recording, and the standard operating system dumps of the MCP partition or region, some special aids for debugging the telecommunication network and the MCP.

- A TCAM-formatted ABEND dump, provided automatically when the MCP fails, which formats control areas attached to the MCP partition or region (this is in addition to the system ABEND dump).
- A cross-reference table in main or virtual storage, which provides the user with a convenient method of locating, in a standard operating system dump, certain information associated with each open line.
- A TCAM I/O interrupt trace table, which provides, in main storage, a sequential record of the I/O interrupts occurring on a specified line.
- A dispatcher subtask trace table used to keep, in main storage, a sequential record of the subtasks activated by the TCAM dispatcher. An entry is placed in the table by the TCAM dispatcher each time a TCAM subtask is dispatched.
- A buffer trace obtained by writing buffer and status information on either magnetic tape or disk.
- A dump of the message queues data set, dynamically invoked by a separate TCAM utility, which formats the data set for immediate printing, or directs it to tape or disk for later printing.

In addition, the cross-reference table, the subtask control block trace table, and the I/O interrupt trace table can be included in a standard operating system dump.

ERROR-RECOVERY PROCEDURES

The TCAM Message Control Program includes a comprehensive set of error-recovery procedures for dealing with the various types of input/output errors that may occur in a telecommunication environment. Whenever an I/O

error interruption occurs, the TCAM error-recovery procedures take action appropriate for the type of error.

If the I/O error is recoverable, TCAM's error-recovery procedures usually retransmit the block of data in which the error occurred. Two retries are performed for start-stop stations, six for BSC stations. If these retries fail to correct the error, it is treated as an irrecoverable error.

For irrecoverable errors, TCAM sends a message to the primary operator control station, writes a permanent-error record on disk, and may terminate the connection between the computer and the station.

Mid-Batch Recovery

Mid-batch recovery is a feature of TCAM that allows the user to recover from permanent text errors in any block of data in a multiblock message except the first.

Mid-batch recovery is not provided for:

- Messages interrupted by the disconnection of a switched station.
- Messages queued on main-storage-only queues.
- Errors made in selecting a station for mid-batch recovery.

An incoming block with a permanent text error is deleted. TCAM retains the portion of the message preceding the discarded block and terminates inmessage processing. (If mid-batch recovery is not specified, TCAM cancels the entire incoming message, if specified by the user.) Otherwise, error conditions can be checked to see if they warrant cancellation.

When a permanent transmission error is detected in an outgoing multiblock message, further transmission is suspended immediately or after a specified delay (except for switched terminals). If a delay has been specified, the operator may be able to recover from a hardware error. (Examples are a stuck keyboard, a terminal out of paper or ribbon, or selection of the wrong operating mode.)

Whether or not the delay is used, transmission is resumed by repeating the message block containing the error. If data is being sent to a station attached to a message-concentrating device, the last concentrator terminal buffer (CTB) is retransmitted.

Queuing by terminal is required for mid-batch recovery.

TCAM CHECKPOINT/RESTART FACILITY

The checkpoint/restart facility allows the Message Control Program to be restarted after a closedown or system failure by using information periodically recorded on the status of each station, destination queue, terminal table entry, and invitation list in the system. Upon restart, the terminal table, option table, invitation lists (optional), and internal control blocks associated with stations and lines are re-

stored to their condition just before closedown or system failure. Outgoing message traffic to each destination resumes with the highest-priority, unsent message.

Checkpoints of the MCP can be coordinated with operating system checkpoints of TCAM application programs (described in the *Checkpoint/Restart Planning Guide*), so that the entire TCAM system can be restored to its condition at the time of closedown or failure.

Records for Checkpoint/Restart

Checkpoint records containing the information necessary to reconstruct the MCP environment upon restart are kept in a checkpoint data set on disk. Four types of records can reside in the checkpoint data set: a control record, two or more environment records, a series of incident records, and one or more checkpoint request records.

The *control record* is used internally by TCAM during restart and need not be considered by the user.

Environment records are taken automatically at certain points during the execution of the MCP to preserve the total MCP environment. Each record contains information on the status of each line, terminal, message queue, and invitation list (optional) at the time the record was taken, and the contents of the option fields for each station.

When a new checkpoint record is taken, it overlays the oldest environment record in the data set. Ordinarily TCAM uses the most recent environment record in the data set to reconstruct the MCP environment for a restart. If, however, the latest record cannot be used (perhaps because of a disk read or write error), TCAM informs the user by a console message and automatically attempts to use the next most recent record. If that record is also unusable, TCAM attempts to use the next environment record, etc.

Incident records are automatically taken to record single changes in line status, terminal status, and option fields introduced by Message Handler macros, certain application-program macros, and operator commands. Incident records are used to update the MCP environment at restart time if the user specifies that a warm restart be performed following either a quick or flush closedown, or that a continuation restart be performed following system failure (see "Restart," below).

The TCAM restart routine updates the MCP fields with the latest usable environment record, then further updates these fields with the incident records taken since the environment record was taken.

Checkpoint request records, initiated by a macro in an application program, record the status of the application program's message queues. They are used in much the same way as incident records to update the environment record during restart.

Closedown

System closedown can be initiated from either an application program or operator control station, and can be either a flush or quick closedown.

In a flush closedown, any incoming message traffic is completed, all outgoing messages are sent or "flushed" from the message queues, and an environment record is taken.

In a quick closedown, message traffic is stopped on each line as soon as any current message is completed, and the status of each queue is preserved with an environment record. Messages are sent to their destinations after restart.

Restart

Restart is here defined as any TCAM start-up other than a cold start. A cold start ignores the previous environment (for an initial start-up, there is no previous environment). The message queues are regarded as new and must have been reformatted.

There are two forms of restart that reconstruct the environment as it existed before termination. In one, the MCP environment is reestablished as it existed before system failure by using the last environment record, the incident records, and the checkpoint request records. From this point, the message queues are searched to determine the last complete message received and transmitted for each queue.

In the other form, used after either a quick or a flush closedown, the MCP environment is reconstructed as for a

continuation restart, except that there are no incident records, and the search of the message queues is minimal because an environment checkpoint was taken after message traffic was stopped.

ON-LINE TEST

On-line test is an optional TCAM facility that permits either a system console operator or a remote station user to determine whether transmission control units and remote stations are working properly by:

- Diagnosing machine errors;
- Verifying repairs;
- Verifying engineering changes;
- Checking devices periodically;
- Checking new stations brought on-line.

No programming considerations are required of the system console operator or remote station user who requests an on-line test for a device. This optional function is specified during TCAM installation. Control program also acts as an interface between TCAM and the device tests, conveys messages to the user about the tests, schedules and controls tests, and notifies the user when he makes an invalid request or must furnish more data for a test. The minimum amount of main storage for running an on-line test is 12K bytes.

Detailed user information is contained in the *TCAM User's Guide* and *TCAM Programmer's Guide*.

TCAM operates under the operating system MFT-II and MVT environments on System/360 Model 40 or above (a CPU with at least 128K of main storage); and under virtual storage (OS/VS) environments on System/370 Model 135 or above. TCAM supports the multiprocessing (IBM Model MP65) system, with the configuration-control feature, formed by two Model 65s operating as one system under one control program.

All telecommunication terminals, except the IBM 2260-2848 or the IBM 3270 in local configurations, must be attached to a transmission control unit. Terminals cannot be attached directly to a channel.

TCAM supports any combination of transmission control units on the same multiplexer channel. Up to eight control units can be attached directly to the channel. TCAM also supports the IBM 2848 Display Control, the IBM 3272 Control Unit and the IBM 3705 Communications Controller attached to either a multiplexer or selector channel.

TRANSMISSION CONTROL UNITS

IBM 2701 Data Adapter Unit

The 2701 provides an interface for the System/360 and System/370 to the teleprocessing network. With appropriate features, the 2701 can attach up to:

- Four half-duplex start-stop communications lines, or
- Four (maximum of two operating simultaneously) half-duplex binary synchronous communications lines, or
- Two half-duplex, start-stop display communications lines.

IBM 2702 Transmission Control Unit

The 2702 offers greater line handling capacities than the 2701. However, because the 2702 attaches only start-stop lines, fewer terminal types can be serviced. The 2702 can be equipped to attach up to 31 start-stop lines.

Additionally, the 2702 can be equipped for *Auto Poll* operation. Auto Poll provides automatic continuation of polling, after a negative response, without program interruption.

IBM 2703 Transmission Control Unit

The 2703 offers greater line handling capabilities than either the 2701 or the 2702, and in addition attaches both start-stop and binary synchronous communications lines. The 2703 can be equipped to attach up to 176 start-stop or up to 48 binary synchronous lines. Auto Poll is a standard feature of the 2703.

IBM 3705 Communications Controller

The 3705 is a programmable communications controller. Programs are generated by the user and reside within the 3705. Two modes of operation are available: *Emulation* mode in which the 3705 emulates 2701, 2702, or 2703 operation; and *Network Control Program* mode, which provides the function of the 2701/2702/2703 and in addition assumes a major portion of the line management and buffering responsibility.

The 3705 attaches to System/370 and, in Emulation mode, to System/360 or System/370. The 3705 communicates with a wide variety of devices in either start-stop or binary synchronous operation. The 3705 can be configured with from 16K to 240K bytes of storage, and, dependent on line speed and operating mode, with up to 352 communications lines.

Emulation mode allows the 3705 to be incorporated into existing systems with a minimum of program changes. TCAM functions, operation, and device support are the same as if the device being emulated were installed.

Network Control Program mode allows the 3705 to assume many of the TCAM functions described elsewhere in this book. The TCAM/3705 configuration can run in either an OS MFT or MVT environment. It is, however, restricted to use with the operating system in System/370 Models 145, 155, 165, and 195.

The 3705 Network Control Program conserves CPU resources by assuming the network management responsibility of the teleprocessing system. Data is presented to the CPU in a single stream with a constant format and minimum of network considerations.

By its Network Control Program, the 3705 can provide:

- Polling and addressing,
- Communication line-control character insertion and removal,
- Character checking,
- Block checking,
- Buffering,
- Message processing functions, such as date and time stamping,
- Line scheduling,
- Device-dependent, error recovery procedures.

TCAM support of the 3705 provides additional operator commands to:

- Activate and deactivate the 3705,
- Set system time into the 3705,

- Display 3705 status,
- Switch channels,
- Switch 3705s,
- Modify network configuration,
- Display selected areas of 3705 storage,
- Control 3705 program loading,
- Initiate dumping of the 3705.

TCAM when operating with the 3705 Network Control Program uses the same MCP/application program structure as is used with other TCUs; however, because network devices and options must be known to both the TCAM MCP and the 3705 Network Control Program, several TCAM macros and operands are provided to ensure compatibility.

IBM 7770 Audio Response Unit

The 7770 converts coded messages, received from the CPU, into voice messages for transmission to a terminal device. In its typical application, the 7770 receives inquires, in digital form, from terminal devices. The inquires are transferred to the CPU for processing. After processing, the CPU returns a coded reply to the 7770. The 7770 converts the reply into audio form and transmits it to the originating terminal.

The 7770 can attach to System/360 or System/370 and can be equipped to service up to 48 communications lines.

TERMINAL DEVICES

Figure 6 shows IBM terminal devices supported by TCAM. In addition to these, a variety of custom devices are available; your IBM Marketing Representative can provide information on custom devices.

TCAM also supports the following IBM systems as terminals:

- System/3,
- System/7,
- System/360,
- System/370,
- 1130 Computing System,
- 1800 Data Acquisition and Control System.

Also supported are: World Trade teleprinters using CCITT No. 2 or No. 5 code on leased point-to-point lines; and terminals using the following line-control disciplines: AT&T 83B3 or WU 115A start-stop code, over point-to-point or multipoint leased telegraph lines; AT&T CPT-TWX (33/35) start-stop code over switched lines.

Attachment of non-IBM terminals is under the provisions of the IBM Multiple Supplier Systems Policy.

Figure 6. Device Configurations Supported by TCAM (Part 1 of 2)

Station Type	Input Media	Output Media	Transmission Control Unit					Line Type	
			2701	2702	2703	3705*	7770	Switched	Non-Switched
1030 Data Collection System	Punched Cards and Badges Data Cartridge Manual Entry	Printer	✓	✓	✓				✓
1050 Data Communication System	Punched Card Paper Tape Keyboard	Printer Paper Tape Punched Card	✓	✓	✓	✓		✓	✓
1060 Data Communication System	Keyboard	Printer	✓	✓	✓				✓
2260-2848 Display Complex	Keyboard	Cathode Ray Tube	✓						✓
2265-2845 Display Complex	Keyboard	Cathode Ray Tube	✓						✓
2721 Portable Audio Terminal	Keyboard	Voice					✓	✓	
2740 Communication Terminal	Keyboard	Printer	✓	✓	✓	✓		✓	✓
2741 Communication Terminal	Keyboard	Printer	✓	✓	✓	✓		✓	✓
2760 Optical Image Unit (Attached to 2740)	Position Sensitive Probe	Projected Film Strip	✓	✓	✓			✓	✓
2770 Data Communication System	Punched Card Paper Tape Keyboard Magnetically Inscribed Documents	Punched Card Paper Tape Cathode Ray Tube Printer	✓		✓	✓		✓	✓

Figure 6, Device Configurations Supported by TCCAM (Part 2 of 2)

Station Type	Input Media	Output Media	Transmission Control Unit					Line Type	
			2701	2702	2703	3705*	7770	Switched	Non-Switched
2780 Data Transmission Terminal	Punched Card	Printer Punched Card	✓		✓	✓		✓	✓
2790 Data Communication System	Punched Cards and Badges Keyboard Rotary Switches and other Digital Devices	Guidance Display Printer	✓		✓	✓		✓	✓
3270 Information Display System	Keyboard Light Pen Magnetically Encoded ID Cards	Printer Cathode Ray Tube	✓		✓	✓			✓
3670 Brokerage Communication System	Keyboard	Printer Cathode Ray Tube	✓		✓				✓
3735 Programmable Buffered Terminal	Keyboard Magnetically Encoded ID Cards Punched Card	Printer Punched Card	✓		✓	✓		✓	✓
3741 Model 2 Data Station	Keyboard	Diskette	✓		✓			✓	✓
3747 Data Converter	Diskette	Magnetic Tape	✓		✓			✓	✓
3780 Data Communication Terminal	Punched Card	Printer	✓		✓	✓		✓	✓

* Devices shown are supported in Network Control program mode. Devices supported when in Emulation mode depend on the TCU being emulated. The 3705 is not supported by Release 1 of VS1 or VS2.

The following glossary is a listing of the communication terms used in the manual. For a complete listing of all communication terms, refer to *Data Processing Glossary*, order No. GC20-1699.

Accepting: In systems with the telecommunications access method (TCAM), the process in which a terminal obtains a message transmitted to it from the computer. Contrast with *entering*. See also *receiving, sending*.

Accepting Station: A destination station that accepts a message.

Access Line: A line which continuously connects a remote station to a switching center (exchange). A telephone number is associated with the access line.

Access Method: A technique for moving data between main storage and input/output devices.

Active Line: A line which is currently available for transmission of data.

Active Station: A station which is currently eligible for entering or accepting messages on the line.

Addressing: The means whereby the originator or control station selects the unit to which it is going to send a message.

Addressing Characters: Identifying characters sent by the computer over a line that cause a particular station (or component) to be selected to accept a message sent by the computer.

Allocate: To assign a resource for use in performing a specific task.

Answering: A procedure by which a called party completes a connection (for switched lines).

Application Program: A program written by a user that applies to his own work. Application programs run asynchronously with the Message Control Program; they are usually located in another partition or region of main storage. There may be many TCAM application programs or none, depending on the needs of the user.

ARU: See *audio response unit*.

Assemble: To prepare a machine language program from a symbolic language program by substituting absolute operation codes for symbolic operation codes and absolute or relocatable addresses for symbolic addresses.

Asynchronous Operation: Processing of messages in an application program independently from message control in the MCP; asynchronous operations in TCAM result in better utilization of the differences between handling and processing speeds. (See also *message handling and message processing*.)

Audio Inquiry: Keying or dialing data into a computer having an attached IBM 7770 Model 3 Audio Response Unit to provide an audible response.

Audio Line: A switched communication line attached to an audio response unit, such as the IBM 7770 Model 3 Audio Response Unit.

Audio Response Message: An audible response generated by an IBM 7770 Model 3 Audio Response Unit from output accepted from a computer.

Audio Response Unit: An output device, such as the IBM 7770 Audio Response Unit, that provides a spoken response to digital inquiries from a telephone or other device. The response is composed from a vocabulary of words prerecorded on a magnetic drum and can be transmitted over communication lines to the location from which the inquiry originated.

Audio Terminal: A unit of equipment associated with an audio response unit, at which keyed or dialed data is entered for transmission to the computer and which produces an audio response as output accepted from the computer.

Auto-Answer: A hardware feature that provides a transmission control unit or station with the capability to automatically respond to a call over a switched line. See also *unattended operation*.

Auto Call: A hardware feature that provides a transmission control unit or station with the capability to automatically initiate a call over a switched line.

Automatic Polling: A hardware feature of a telecommunications unit that processes a polling list, polling the terminals in order and handling negative responses to polling without interrupting the central processing unit. At the end of the list, polling is begun again automatically at the beginning of the list. Synonymous with *autopoll*. See also *polling*.

Autopoll: Same as *automatic polling*.

Available Unit Queue: In systems with the telecommunications access method (TCAM), a queue in main storage to which all buffer units are assigned initially (that is, prior to assignments to TCAM lines and application programs requiring buffers).

Batch Processing: (1)*Pertaining to the technique of executing a set of computer programs such that each is completed before the next program of the set is started. (2)*Pertaining to the sequential input of computer programs or data. (3)*Loosely, the execution of computer programs serially. (4) In systems with the time sharing option (TSO), the processing of one job step in a region, so called because jobs are submitted in a group or batch.

Bid: In the contention form of invitation or selection, an attempt by the computer or a station to seize control of a line so that it can transmit data.

Binary Synchronous Communications (BSC): Data transmission in which character synchronism is controlled by timing signals generated at the sending and receiving stations. Contrast with *start-stop transmission*.

Block: (1) A set of things, such as words, characters, or digits, handled as a unit. (2) A collection of contiguous records recorded as a unit. Blocks are separated by block gaps and each block may contain one or more records. (3) A group of bits, or n-ary digits, transmitted as a unit. An encoding procedure is generally applied to the group of bits or n-ary digits for error-control purposes. (4) A group of contiguous characters recorded as a unit.

BSC: See *binary synchronous communications*.

Buffer: (1)*A routine or storage used to compensate for a difference in rate of flow of data, or time of occurrence of events, when transmitting data from one device to another. (2) An area of storage that is temporarily reserved for use in performing an input/output operation, into which data is read or from which data is written. Synonymous with I/O area.

Buffers can also be used as work areas in TCAM. The size of TCAM buffers is designated by the user.

Buffer Allocation: The assignment of buffers by TCAM to lines or application programs to prepare for reception of message segments from stations or application programs. See also *dynamic buffer allocation* and *static buffer allocation*.

*American National Standard Definition.

Buffer Deallocation: In a sending operation, the return of the units in a buffer to the available-unit queue after the data in these units has been sent to its destination station or application program; in a receiving operation, the transfer of full buffers from the line or application program to the incoming group of the Message Handler that is to process the message segments.

Buffer Pool: (1) An area of storage in which all buffers of a problem program are kept. (2) In systems with the telecommunications access method (TCAM), a group of buffers having the same size. A buffer pool is established at initialization time in the message control program; the buffers are built in extents chained together.

Buffer Prefix: In systems with the telecommunications access method (TCAM), a control area contained within each TCAM buffer. TCAM fills the prefix area with buffer control information.

Buffer Size: The number of bytes in each buffer in a line group.

Buffer Unit: In systems with the telecommunications access method (TCAM), a block from which TCAM buffers are constructed.

Buffer Unit Pool: In a system with the telecommunications access method (TCAM), all of the buffer units for that system.

Burst Mode: A means of transferring data to or from a particular I/O device on either the multiplexer or selector channel. All channel controls are monopolized for the duration of data transfer.

***Call:** (1) To transfer control to a specified closed subroutine. (2) In communications, the action performed by the calling party, or the operations necessary in making a call, or the effective use made of a connection between two stations. (3) Synonymous with cue.

Calling: In communications, a procedure by which a first party attempts to establish a connection with a second party through a central exchange. See also *dialing*.

Cascade Entry: An entry in a terminal table associated with a cascade list.

Cascade List: A list of pointers to single, group, or process entries. A message is queued for the valid entry in the list with the fewest messages queued for it.

Catalog: (1) *An ordered compilation of item descriptions and sufficient information to afford access to the items. (2) The collection of all data set indexes that are used by the control program to locate a volume containing a specific data set. (3) To include the volume identification of a data set in the catalog. (4) Under the Disk and Tape Operating Systems (DOS and TOS), to enter a phase, module, or book into one of the system libraries.

Central Computer: The computer in which TCAM is operating as the control program for a teleprocessing network. Often referred to in this publication as *the computer*.

Central Processing Unit (CPU): The portion of a computer that interprets and executes instructions.

Channel: (1) *A path along which signals can be sent, for example, data channel, output channel. (2) *The portion of a storage medium that is accessible to a given reading or writing station, for example, track, band. (3) *In communication, a means of one-way transmission. Several channels may share common equipment. For example, in frequency multiplexing carrier systems, each channel uses a particular frequency band that is reserved for it. Contrast with *circuit*. (4) A hardware device that connects the CPU and main storage with the I/O control units. (5) As used in the tariffs, a path for electrical transmission between two or more points without common-carrier-provided terminal equipment. Synonymous with *circuit, line, link, path, facility*.

Channel Program Block (CPB): In systems with the telecommunications access method (TCAM), a control block used in the transfer of data between buffer units and message queues maintained on disk.

Checkpoint Data Set: A sequential or partitioned data set containing a collection of checkpoint entries. If a checkpoint data set is a partitioned data set, each checkpoint entry is a member.

Checkpoint Records: Records that contain the status of a job and the system at the time the records are written by the checkpoint routine. These records provide the information necessary for restarting a job without having to return to the beginning of the job. The four types of checkpoint records in TCAM are environment records, incident records, checkpoint request records, and a control record.

Checkpoint Request Record: In systems with the telecommunications access method (TCAM), a checkpoint record taken as a result of the execution of a CKREQ macro instruction in an application program; the record contains the status of a single destination queue for the application program.

Checkpoint Restart: The process of resuming a job at a checkpoint within the job step that caused abnormal termination. At the operator's discretion, the restart may be automatic or deferred, where deferring involves resubmitting the job.

Closedown: In systems with the telecommunications access method, an orderly deactivation of the message control program. See *quick closedown* and *flush closedown*.

Cold Restart: Start-up of a TCAM MCP following either a flush closedown, a quick closedown, or a system failure. A cold restart ignores the previous environment (that is, the MCP is started as if this were the initial start-up); it is the only type of restart possible when no checkpoint/restart is used.

Common Carrier: A government-regulated private company which furnishes the general public with telecommunications service facilities; for example, a telephone or telegraph company.

Communication Line: Any medium, such as a wire or a telephone circuit, that connects a remote terminal with a computer.

Component: (1) In teleprocessing, one or more input/output devices attached to a single control unit, and together making up one remote terminal. (2) A point in a communications system at which data can enter or leave; an input/output device. A component is always attached to a terminal control unit.

Concatenated Data Set: A collection of logically connected data sets. See *job library*.

Concentration: The process of combining multiple messages into a single message for transmission.

Concentrator: Any device that either combines multiple incoming messages into a single message (concentration), or extracts individual messages from data sent in a single transmission sequence (deconcentration).

Concentrator Terminal Buffer (CTB): The amount of data in individual messages transmitted either to or from a concentrator. Buffer in this instance bears no relationship to a TCAM buffer.

***Concurrent:** Pertaining to the occurrence of two or more events or activities within the same specified interval of time. Contrast with *consecutive, sequential, simultaneous*.

Contention: (1) A condition on a multidrop communication channel when two or more locations try to transmit at the same time. (2) Unregulated bidding for a line by multiple users. (3) In systems with the telecommunications access method (TCAM), any point-to-point line configuration in which the station on the line does not use polling or addressing characters.

Continuation Restart: A restart of the TCAM Message Control Program following termination of the MCP because of system failure; TCAM checkpoint/restart is used to restore the MCP environment as nearly as possible to its condition before failure.

***Control Character:** (1) A character whose occurrence in a particular context initiates, modifies, or stops a control operation, for example, a character that controls carriage return, a character that controls transmission of data over communication networks. A control character may be recorded for use in a subsequent action. It may in some circumstances have a graphic representation.

*American National Standard Definition.

Control Record: In systems with the telecommunications access method (TCAM), a record included in a checkpoint data set which keeps track of the correct environment, incident, and checkpoint request records to use for restructuring the message control program environment during restart.

Control Unit: A device that controls the reading, writing, or display of data at one or more input/output devices.

CPB: See *channel program block*.

CPU: See *central processing unit*.

***Crosstalk:** The unwanted energy transferred from one circuit, called the "disturbing" circuit, to another circuit, called the "disturbed" circuit.

CTB: See *concentrator terminal buffer*.

DASD: See *direct-access storage device*.

Data Collection: A telecommunication application in which data from several locations is accumulated at one location (in a queue or on a file) before batch processing.

Data Control Block (DCB): A control block used by access routines in storing and retrieving data.

Data Link: The communications lines, modems, and communication controls of all stations connected to the line, used in the transmission of information between two or more stations.

Data Set: (1) The major unit of data storage and retrieval in the operating system, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access. (2) A device which performs the modulation/demodulation and control functions necessary to provide compatibility between business machines and communications facilities.

Delimiter Macro Instructions: TCAM macro instructions that classify and identify sequences of functional macro instructions and direct control to the appropriate sequence of functional macro instructions.

Destination: The place to which a message is to be sent after it leaves a Message Handler: either a station, a group of stations, or an application program. One or more destinations can be specified by the user.

Destination Queue: In systems with the telecommunications access method (TCAM), a queue on which messages bound for a particular destination are placed after being processed by the incoming group of the message handler.

Destination Station: A station that accepts a message sent to it by the outgoing group of the Message Handler.

Dial: See *call*.

Dial Line: See *switched line*.

Direct Access Storage Device (DASD): A device in which the access time is effectively independent of the location of the data.

Disabling the Line: Causing the computer to condition either the transmission control unit or the audio response unit to ignore incoming calls on a switched line; the line is then available for TCAM to send queued messages to a station on that line. See *enabling the line*.

Disk: See *direct-access storage device*.

Distribution List: In systems with the telecommunications access method (TCAM), a list of terminals, each of which is to receive any message directed to the group.

Dynamic Buffer Allocation: The assignment of buffers to a line on an as-needed basis, after a message has started coming in over the line; dynamic allocation occurs following program controlled interruptions. See also *static buffer allocation*.

EBCDIC: Extended Binary Coded Decimal Interchange Code.

Enabling the Line: In systems with the telecommunications access method (TCAM), the process of conditioning the transmission control unit to accept incoming calls on a line.

End of Address: (1) One or more control characters transmitted on a line to indicate the end of non-text characters (for example, addressing characters). (2) In systems with the telecommunications access method (TCAM), a character that must be placed in a message if the system is to route that message to several destinations. The character must immediately follow the last destination code in the message header. Abbreviated EOA.

Entering: In systems with the telecommunications access method (TCAM), the process in which a terminal places on the line a message to be transmitted to the computer. Contrast with *accepting*. See also *receiving, sending*.

Environment Record: In systems with the telecommunications access method (TCAM), a record of the total telecommunications environment at a single point in time.

At restart time, an environment record is updated by the contents of incident records taken after the environment record was taken, and the updated environment record is then used to reconstruct the Message Control Program environment as it existed before MCP closedown or failure.

EOA: See *end-of-address character*.

Error Recovery Procedures (ERP): Standard procedures designed to ensure that all the routines that test particular devices provide a uniform type and quality of information.

Exchange: A unit established by a communications common carrier for the administration of communication service in a specified area which usually embraces a city, town, or village and its environs. It consists of one or more central offices together with the associated equipment used in furnishing communication service. Synonymous with *central office*.

Extended Lock Mode: A type of lock mode wherein a station is locked for the duration of several inquiry-response cycles.

FEFO (First-ended, First-out): A queuing scheme in which a message on a destination queue, whose final segment arrived first at the queue, is sent before any other message in the same priority group.

FIFO (First-in, First-out): A queuing scheme whereby equal-priority messages on the same destination queue are sent in the order that their first segments arrived at the queue.

Flush Closedown: A closedown of the TCAM Message Control Program during which incoming message traffic is suspended and queued outgoing messages are sent to their destinations ("flushed" from the message queues) before closedown is completed. See also *quick closedown*.

Functional Macro Instructions: TCAM macro instructions that perform the specific operations required for messages directed to the Message Handler; compare *delimiter macro instructions*.

Get: (1) To obtain a record from an input file. (2) The system macro instruction used in queued data access to obtain a record from an input data set. The instruction is device-independent. See also *PUT, READ, and WRITE*.

***Half Duplex:** In communications, pertaining to an alternate, one way at a time, independent transmission. Contrast with *duplex*.

Handling: Functions provided by the Message Control Program. See *message handling*.

Header: See *message header*.

Identification (ID) Characters: Characters sent by a BSC station or the central computer on a switched line to identify the sender.

Inactive Line: A communication line not currently available for transmitting data.

Inactive Station: In systems with the telecommunications access method (TCAM), a station that is currently ineligible for entering or accepting messages.

*American National Standard Definition.

Incident Record: A checkpoint record of a change in station status or in an option field since the last environment record. Incident records are used to update the information in environment records when restarting after a closedown or system failure.

Incoming Message: A message being transmitted from a station to the central computer. Compare *outgoing message*.

Input Sequence Number: A number in each message header that is checked by the Message Handler to ensure receiving messages in correct order. Its use is optional.

Inquiry and Transaction Processing: A type of teleprocessing application in which inquiries and records of transactions received from a number of remote terminals are used to interrogate or update one or more master files maintained by the central system.

Intercepted Station: In systems with the telecommunications access method (TCAM), a station to which no messages may be sent. An intercepted station can enter messages—only messages queued for the destination are stopped.

Invitation: The process in which a computer makes contact with a terminal in order to give the terminal the opportunity to transmit a message if it has one ready; forms of invitation include polling and enabling.

Invitation Delay: A period, on nonswitched polled lines, when outgoing messages are sent to stations with receiving priority. This delay is observed for all polled stations on a line when the end of the invitation list for that line is reached, whether or not the computer has any messages to send. If no invitation delay is specified, no messages can be sent. See also *system interval*.

Invitation List: In systems with the telecommunications access method (TCAM), a sequence of polling characters or identification sequences associated with the stations on line; the order in which the characters are specified determines the order in which the stations are invited to enter a message.

Job: (1) *A specified group of tasks prescribed as a unit of work for a computer. By extension, a job usually includes all necessary computer programs, linkages, files, and instructions to the operating system. (2) In System/360 operating systems, a collection of related problem programs, identified in the input stream by a JOB statement followed by one or more EXEC and DD statements. (3) In systems with the time sharing option (TSO), the processing done on behalf of one user from LOGON to LOGOFF; one terminal session.

Job Library: A set of user-identified, partitioned data sets used as the primary source of load modules for a given job.

Line: In communications, same as channel, circuit. See also *nonswitched line*, *switched line*, *point-to-point line*, and *multipoint line*.

Line Control: The scheme of operating procedures and control signals by which a telecommunications system is controlled.

Line-control Characters: Characters that regulate the transmission of data over a line—delimiting messages, checking for transmission errors, indicating whether a station has data to send or is ready to receive data, etc.

Line Group: In systems with the telecommunications access method (TCAM), a set of one or more communications lines of the same type, over which terminals with similar characteristics can communicate with the computer.

Line-group Data Set: A data set made up of all the lines in a line group, whose data is the transmitted messages.

Line Traffic: Data transmission on a line.

Link Edit: Combining and editing of program object modules into a single load module for execution in main storage.

Linkage Editor: A processing program that prepares the output of language translators for execution. It combines separately produced object or load modules; resolves symbolic cross references among them; replaces, deletes, and adds control sections, and generates overlay structures on request; and produces executable code (a load module) that is ready to be fetched into main storage.

Local Station: A station whose control unit is connected directly to a computer data channel by cable rather than by common-carrier line. Compare *remote station*.

Locate Mode: A way of providing data by pointing to its location instead of moving it.

Lock Mode: In systems with the telecommunications access method (TCAM), a mode in which the connection between a terminal and an application program is maintained for the duration of a message and its response.

Use of lock mode decreases response time. A station can be placed in lock mode for the duration of one or several inquiry/response cycles. Compare *extended lock mode*.

Log: A collection of messages or message segments placed on a secondary storage device for accounting or data collection.

Logical Message: A unit of data defined by the user by coding the SETEOM functional macro.

Logical Record: (1) *A collection of items independent of their physical environment. Portions of the same logical record may be located in different physical records. (2) A record from the standpoint of its content, function, and use rather than its physical attributes; that is, one that is defined in terms of the information it contains.

MCP: See *Message Control Program*.

Message: (1) *An arbitrary amount of information whose beginning and end are defined or implied. (2) In telecommunications, a combination of characters and symbols transmitted from one point to another on a network.

Message Control Program (MCP): A program that is used to control the sending or reception of messages to or from remote terminals.

Message Handler: In systems with the telecommunications access method (TCAM), a sequence of user-specified macro instructions that examine and process control information in message headers, and perform functions necessary to prepare message segments for forwarding to their destinations. One message handler is required for each line group having unique message handling requirements.

Message Handling: Processing of *header control information* by user-specified TCAM macro instructions. Compare *message processing*.

Message Header: The leading part of a message that contains information such as the source or destination code of the message, the message priority, and the type of message. See also *message text*.

Message Priority: The relative order in which the messages in a destination queue are transmitted to a destination. Up to 255 different priority levels can be assigned to a single destination. Priority can be specified in the message header or assigned by a macro instruction.

Message Processing: Strictly, functions, provided by a user-designed application program that operate on the text of messages; in this publication, occasionally generalized to include header processing. See also, *message handling*.

Message Queue: A queue of messages that are awaiting processing or waiting to be sent to a terminal.

Message Queues Data Set: A TCAM data set that contains one or more destination queues. Up to three message queues data sets can be specified for a TCAM Message Control Program.

Message Segment: The portion of a message contained in a single buffer.

Message Switching: A telecommunications application in which a message received by a central system from one terminal is sent to one or more other terminals.

*American National Standard Definition.

MFT-II: Multiprogramming with a *fixed* number of tasks; an operating system control program option that supervises the execution of more than one job at a time. TCAM can be operated in an MFT-II environment. See also *MVT*.

Mid-batch Recovery: The ability to recover from permanent text errors encountered in any block of data following the first block in a multiblock message.

Mixed Environment: A system in which TCAM and TSO tasks concurrently share all necessary system resources.

Modem: See *data set*.

***Move Mode:** In some variable-word-length computers, data transmission such that certain delimiters are not moved with the data. Contrast with *load mode*.

Multiplexer Channel: A channel designed to operate with a number of I/O devices simultaneously. Several I/O devices can transfer records at the same time by interleaving bytes of data.

Multipoint Line: A line or circuit connecting several stations.

MVT: Multiprogramming with a *variable* number of tasks; an operating system control program option that, like MFT-II, supervises execution of more than one job step at a time. In addition it allocates main storage dynamically to each job. TCAM MVT supports both the customer with a large job and the customer with many small jobs. See also *MFT-II*.

Non-switched Line: A connection between a remote terminal and a computer that does not have to be established by dialing.

Nontransparent mode: A form of BSC transmission in which all control characters are treated as control characters rather than as text. Compare *transparent mode*.

Online Test System (OLTS): A system that allows a user to test I/O devices concurrently with execution of programs. Tests may be run to diagnose I/O errors, verify repairs, and engineering changes, or to periodically check devices.

Operator Command: A statement to the control program, issued via a console device, or control terminal, which causes the control program to provide requested information, alter normal operations, initiate new operations, or terminate existing operations.

Operator Control Station: In systems with the telecommunications access method (TCAM), any station that is eligible to enter operator commands.

Option Field: A storage area containing data relating to a particular station, component, line, or application program used by certain Message Handler or user routines.

OS/VS: Operating system/virtual storage. A compatible extension of the System/360 Operating System that supports relocation and the extended control facilities of System/370.

Outgoing Message: A message being transmitted *from* the computer to a destination.

Output Sequence Numbers: Numbers in a message header that determine the order in which the messages are sent to a destination. See also *input sequence number*.

Partition: (1) Under the MFT configuration of the control program, a subdivision of the dynamic area that is allocated to a job step or a system task. See also *region*.

(2) In the IBM System/360 Operating System, a data set in direct access storage can be divided into partitions, called members, each of which can contain a program or part of a program. Each partitioned data set contains a directory (or index) that the control program can use to locate a program in the library. Synonymous with program library.

PCI: See *program-controlled interruption*.

Physical Transmission: The amount of data entered on a line during an entire transmission sequence, from the first byte of data to the end-of-transmission character.

Point-to-point Line: A line that connects a single remote station to the computer; it may be either switched or nonswitched.

Polling: A technique by which each of the terminals sharing a communications line is periodically interrogated to determine whether it requires servicing.

Polling Characters: A set of characters peculiar to a terminal and the polling operation; response to these characters indicates to the computer whether the terminal has a message to enter.

Polling Delay: A user-specified delay between polling passes through an invitation list for either a line or a line group. See also *system interval*.

Polling Pass: A series of operations in which the computer invites each station, in a specified sequential order, to enter messages.

Prefix: See *buffer prefix*.

Primary Operator Control Station: In systems with the telecommunications access method (TCAM), the operator control station that can receive an error recovery procedure message, send operator commands, and receive related responses.

Primary Storage: Main storage.

Priority: See *message priority* and *transmission priority*.

Process Queue: A destination queue for an application program.

Processing: Functions provided by a user-designed application program.

Program Data Set: See *data set*.

Program-Controlled Interruption (PCI): An interruption that allows buffers to be deallocated continuously, replenishing the available unit pool.

Put: (1) To place a single data record into an output file. (2) A system macro instruction used in queued data access that places a record in an output data set. It is logically sequential and device-independent. See also *GET*, *READ*, and *WRITE*.

Queue: (1) A waiting line or list formed by items in a system waiting for service; for example, tasks to be performed or messages to be transmitted in message switching system. (2) To arrange in, or form, a queue.

***Queued Access Method:** Any access method that automatically synchronizes the transfer of data between the program using the access method and input/output devices, thereby eliminating delays for input/output operations.

Quick Closedown: A closedown of the TCAM Message Control Program in which message traffic on each line is stopped as soon as any messages currently being sent or received are transmitted.

Read: (1) To acquire or interpret data from a storage device, a data medium, or any other source.

(2) A system macro instruction that retrieves a data block from an input data set and places it in a user-designated work area in main storage. This technique is used when the sequence in which the records are to be processed is unknown, or when the user does not want some or all of the automatic functions performed by the queued access technique. See also *WRITE*, *GET*, and *PUT*.

Read-ahead Queue: An area of main storage from which an application program obtains work units, using *GET* or *READ* macro instructions, to move them to a work area.

Receiving: The process by which a computer obtains a message from a line. Contrast with *sending*. See also *accepting*, *entering*.

***Record:** A collection of related items of data, treated as a unit, for example, one line of an invoice may form a record; a complete set of such records may form a file. See *logical record*.

*American National Standard Definition.

Region: (1) Under the MVT configuration of the control program, a subdivision of the dynamic area that is allocated to a job step or a system task. (2) An *overlay region* is a continuous area of main storage in which segments can be loaded independently of paths in other regions. Only one path within a region can be in main storage at any one time.

Remote Station: (1) *Data terminal equipment for communicating with a data processing system from a location that is time, space, or electrically distant. (2) Contrast with *local station*.

Resource: Any facility of the computing system or operating system required by a job or task, and including main storage, input/output devices, the central processing unit, data sets, and control processing programs.

***Restart:** To reestablish the execution of a routine, using the data recorded at a checkpoint.

Secondary Operator Control Station: In systems with the telecommunications access method (TCAM), any operator control station that can send operator commands and receive related responses.

Segment: (1) *To divide a computer program into parts such that the program can be executed without the entire program being in internal storage at any one time. (2) *A part of a computer program as in (1). (3) In telecommunications, a portion of a message that can be contained in a buffer.

Seize: To gain control of a line for transmitting data. See also *bid*.

Selection: Addressing a terminal or a component on a selective calling circuit.

Selector Channel: A channel designed to operate with only one I/O device at a time. Once the I/O device is selected, a complete record is transferred one byte at a time.

Sending: The process by which the central computer places a message on a line for transmission to a terminal. Contrast with *receiving*. See also *accepting, entering*.

Sequence Number: See *input sequence number* and *output sequence number*.

Single Entry: An entry in the terminal table associated with a single station or station component; one such entry must be created for each station in the TCAM system not defined by a group entry.

Start-stop Transmission: Asynchronous transmission in which each group of code elements corresponding to a character signal is preceded by a start signal which serves to prepare the receiving mechanism for the reception and registration of a character, and is followed by a stop signal which serves to bring the receiving mechanism to rest in preparation for the reception of the next character.

Static Buffer Allocation: The assignment of each line buffer before transmission over that line.

Station: (1) One of the input or output points of a communications system; for example, the telephone set in the telephone system or the point where the business machine interfaces with the channel on a leased private line. (2) A computer or terminal.

Subblock: That portion of a binary synchronous (BSC) message terminated by an ITB line-control character.

Subgroup: A subdivision of either an incoming or outgoing group. An incoming group is made up of inheader, inblock, inbuffer, and inmessage subgroups; an outgoing group of outheader, outbuffer, and outmessage subgroups.

Substitute Mode: A transmittal mode used with exchange buffering on which segments are pointed to, and exchanged with, user work areas. See also *locate mode* and *move mode*.

Switched Line: A communication line in which the connection between the computer and a remote station is established by dialing. Synonymous with *dial line*.

System Shutdown: See *closedown*.

System Interval: A user-specified interval during which polling and addressing are suspended on multipoint lines to polled stations. The system interval is specified during TCAM initialization and is initiated by an operator command; it optimizes productive polling by minimizing central processing unit time and synchronizing polling. The duration of subsequent system intervals can be changed with another operator command. See also *invitation delay*.

Task: A unit of work for the central processing unit from the standpoint of the control program; therefore, the basic multiprogramming unit under the control program.

TCU: See *transmission control unit*.

Telecommunications: (1) *Pertaining to the transmission of signals over long distances, such as by telegraph, radio, or television. (2) Data transmission between a computing system and remotely located devices via a unit that performs the necessary format conversion and controls the rate of transmission.

Teleprocessing: The processing of data that is received from or sent to remote locations by way of telecommunication lines.

Terminal: (1) *A point in a system or communication network at which data can either enter or leave. (2) Any device capable of sending and receiving information over a communication channel.

Terminal Table: An ordered collection of information that identifies each line, station, component, or application program from which a message can originate or to which a message can be sent.

Termination: See *closedown*.

Text: (1) *In ASCII and communications, a sequence of characters treated as an entity if preceded and terminated by one STX and one EXT communication control character, respectively. Contrast with *heading*. (2) The control sections of an object module or load module. (3) The data portion of a telecommunications message.

Time Sharing: (1) *Pertaining to the interleaved use of the time of a device. (2) A method of using a computing system that allows a number of users to execute programs concurrently and to interact with the programs during execution.

Time Sharing Option (TSO): An option of the IBM System/360 Operating System that provides conversational time sharing from remote terminals.

***Transmission:** (1) The sending of data from one location and the receiving of data in another location, usually leaving the source data unchanged. (2) The sending of data. (3) In ASCII and communications, a series of characters, including headings and texts.

Transmission Code: A code for sending information over communications lines.

Transmission Control Unit: An input/output control unit that addresses messages to and receives messages from a number of remote terminals. Synonymous with *telecommunications control unit*.

Transmission Priority: The relative order of sending and receiving for a station, specified by line group. The three transmission priorities possible are send, equal, and receive; the exact meaning of each depends upon the line configuration and type of station. See also *message priority*.

Transmission Technique: The method used to manage the transfer of data over a communication line. See *binary synchronous communications* and *start-stop transmission*.

Transparent Mode: A mode of BSC transmission in which most characters, including normally restricted, control characters, are transmitted as data. Characters retained for control are preceded by a special DLE character.

TSO: See *Time Sharing Option*.

*American National Standard Definition.

Unit: See *buffer unit* or *work unit*.

VS: See *OS/VS*.

Warm Restart: A restart of the TCAM Message Control Program following either a quick or flush shutdown: the TCAM checkpoint/restart facility is used to restore the MCP environment as nearly as possible to its condition before failure. See *restart*.

Work Area: An area designated for an application program to handle the work units obtained by a GET or READ macro instruction.

Work Unit: The unit of data obtained by a GET or READ macro instruction for processing in an application program. The work unit may be a message, record, or segment (for QTAM-compatible application programs).

**Write:* (1) To record data in a storage device or a data medium. The recording need not be permanent, such as the writing on a cathode ray tube display device. (2) A system macro instruction used to place data in an output data set from a designated work area in main storage. See also *READ*, *GET*, and *PUT*.

*American National Standard Definition.

(Where more than one page reference is given,
the major reference is first.)

- ABEND dump 30
- access line 3
- activating application programs 25
- addressing 11
- addressing characters 11
- alternate destination 17, 29
- answering 10
- application program 25
 - activating/deactivating 25
 - as operator control station 30
 - asynchronous operation 5
 - checkpoint/restart 25
 - closing system from 32
 - entering operator commands from 29
 - intercepting/releasing concentrated messages 23
 - MCP interface 25
 - message processing 7
 - modifying based on error condition 30
 - number needed 13, 25
 - off-line testing 5, 25
 - operator commands from 29
 - providing data for 17
 - running as separate task 5
 - used as alternate destination 29
- applications 6
- ARU (audio response unit) 3
- ASCII 9
- audio line 3
- audio response unit (ARU) 3
- audio terminal 3
- Auto Poll 10
 - through operator control 29
- background operations (TSO) 23
- bidding 13
- binary synchronous communication (BSC) 9
- block 17
- blocking 18, 22
- BSC (*see* binary synchronous communication) 9
- buffer 15
 - application program 25, 26
 - dynamic allocation 14
 - application program 26
 - specifying effect of PCI 15
 - variable sizes 6
- buffer pool 11
- buffer size 6
- buffer trace 30
- buffer unit pool 15
- buffered terminal, priority 22
- buffering 5, 11
- bus out recording 30
- calling 11
- canceling erroneous messages 17
- checkpoint 17, 27
- checkpoint data set 14, 31
- checkpoint request record 31
- checkpoint/restart 31
- closedown 32
 - initiating 29
 - restart following 31
- cold start 32
- command reject, recording 30
- communication line 3
 - enabling and disabling 17
- component 1
- concentrated message 23
- contention 11, 13
- control area, dump 30
- control block
 - examining or modifying 25
- control character 9
- control information, generating 5
- control record 31
- conversational mode 6, 18, 26
- cross-reference table 30
- CTB (Glossary)
 - retransmitting erroneous block 31
- data character, control 9
- data check, recording 30
- data collection 7
- data-link control 9
- data set 3, 13
 - application program
 - closing due to MCP failure 27
 - checkpoint 14
 - line group 14
 - log 14
 - message queues 14
- date/time insertion 17
- deactivating application program 25
- deblocking 18, 22
- debugging aids 30
 - logging messages 30
 - through operator control 29
- delay
 - polling interval 13
- delimiter character for logical message 22
- delimiter macro 18
- destination queue
 - application program 25
 - responses to operator commands 29
- devices supported by TCAM 33
- disk
 - back-up copies of messages 5
 - checkpoint data set 31
 - message queues data set 14
 - organization 5
 - permanent error record on 30
 - queuing 14
- dispatcher subtask tract table 30
- dump 30
- dynamic buffer allocation 5, 14-17
- EBCDIC 9
- editing 17

- enabling the line 10
- end-of-message character 18
 - for logical messages 22
- end of transmission character 25
- environment record 31
- EOB 17, 18
- EOT 17, 18, 25
- equal priority 22
 - concentrated message 23
- equipment check, recording 30
- error
 - count 17
 - line 10
 - permanent 30
 - recording, I/O 30
 - displaying 29
 - recovery procedures 30
 - text 30
- error checking, specifying 5
- error message, transmitting 17
- ETB 17, 18
- ETX 17, 18
- extended lock 26

- flush closedown 32
- foreground (TSO) 23
- formatted ABEND dump 30
- functional macro 18

- graphic character 9
- group, incoming and outgoing 18

- identification exchange 11
- inblock subgroup 18, 19, 22
- inbuffer subgroup 18
- incident record 31
- incoming message, logical 22
- inheader subgroup 18
- inmessage subgroup 18
- inquiry applications 6
- inquiry processing 7
- inquiry/response 6, 25, 26
- intercept function 17
 - application program 25
 - concentrated messages 23
 - through operator control 29
- interrupt, I/O trace table 30
- intervention required, recording 30
- invitation 10, 17
- invitation list 10
 - entries, activating/deactivating 29
 - removing station from 29
 - status 31
- I/O error, recovery 30
- I/O error recording 30
- I/O interrupt trace table 30
- irrecoverable error, TCAM procedures for 30
- ITB 17

- leading graphic response, recording 30
- line 3
 - activating/deactivating 29
 - cross-reference table 30
 - enabling and disabling 17
 - error-recording for 30
 - specifying number of temporary error records 30
 - starting/stopping 29
- line control 9, 13
 - design factors 9
- line address, displaying 29
- line control character
 - end-of-transmission 25
 - inserting and deleting 17
 - specifying whether to be removed 10, 13
- line lock 26
- line group, starting/stopping 29
- line group data set 14
- line status 31
 - determining 29
 - displaying 29
- lock mode 6, 7, 26
- log data set 14
- logging 17, 30
- logical message 18, 22
 - in block subgroup 18
- lost data, recording 30

- machine and device requirements 33
- macro instructions
 - controlling message traffic 5
 - protection against unauthorized use 30
 - types used in Message Handler 18
 - use 17
- main storage
 - message queues data set 14
 - queuing 5, 14
 - with disk backup
- MCP (*see* Message Control Program)
- message 17
 - backup copy on disk 5
 - concentration 23
 - for output 23
 - retransmitting erroneous block 31
 - logical 22
 - maintaining count 17
 - multiblock error recovery 30
 - processing by application program 25
 - units 17
- Message Control Program (MCP) 17
 - application program interface 25
 - asynchronous operation 5
 - closing 29
 - data collection 6
 - designing 5
 - dump 30
 - execution 13
 - logging messages handled by 30
 - message switching 6
 - processing messages 17
 - reconstructing for restart 31
 - restarting 31
- message count, displaying 29
- message editing 17
 - specifying 5
- Message Handler 18
 - concentrating messages 23
 - TSO 23
- message flow 18, 21
 - application program 25
 - determining for MCP 30
 - example 22
 - subgroups, example 20
- message lock 26
- message logging 30
- message priority 21
 - effect on message flow 21
- message processing 7
- message queue status 31
- message queues data set 14
 - dump 30
- message retrieval 25, 26
 - limitation 14

- message routing 17
 - specifying 5
- message segment 15
- message switching 6
- message traffic, control 3
- MFT-II 33
- mid-batch recovery 30
- mixed environment (TSO) 23
- multiblock message, error recovery 30
- multiple applications 7
- multiple-arm support 5, 14
- multiple buffers 11
- multiplexer channel 33
- multipoint line 3
- multiprocessing 33
- MVT 33

- network control 29
 - from application program 25
- nonreusable disk queuing 5
- nonswitched contention station, priority 22
- nonswitched line 3
- nonswitched station, priority 22
- nontransparent mode 10

- on-line test 32
- operator commands
 - entering 29
 - from application program 29
 - incorrectly formatted 29
 - responses to 29
- operator control 29
 - altering delays 11
 - changing primary station 29
 - from application program 25
 - intercepting/releasing concentrated messages 23
- operator control station 29
 - closing system from 32
- option field 18
 - altering 29
 - displaying 29
 - inserting data in 29
- option field status 31
- OS checkpoint 27
- outbuffer subgroup 18
- outgoing message, concentrating 23, 23
- outheader subgroup 18
- outmessage subgroup 18
- overflow, recording 30

- partition 13
- PCI (program-controlled interruption) 15
- permanent error 30
- permanent-error record 30
- permanent text error 30
- physical message 25
- physical transmission 18
 - converting to logical messages 22
- point-to-point line 3
- polling 13
 - changing delay 29
 - changing type 29
 - displaying type used 29
- polling characters 10
- polling interval 13
 - altering 13
- prefix 15
- primary operator control station 29
 - changing to secondary 29
- priority
 - concentrated message 23
 - TCAM application programs 25
 - transmission 21, 22
- program-controlled interruption (PCI) 15
- programmed polling 29
- programmed-polling delay, altering 29

- QTAM application, running with TCAM 25
- queued access method, advantages 3
- queuing 17, 21
 - by line 21
 - by station 21
 - disk 14
 - effect on message flow 21
 - options 5
 - priority for concentrated messages 23
 - requirements for mid-batch recovery 31
 - status, determining 29
- quick closedown 32

- receive priority 21
 - concentrated message 23
- record 25, 17
 - checkpoint/restart 31
- recoverable error
 - record 30
 - TCAM procedures for 30
- region 13
- register availability 6
- releasing intercepted messages 29
- response message, queuing 17
- restart 31, 32
- reusable disk queuing 5

- secondary operator control station 29
 - changing to primary 29
- segment 15, 17
- selection 10
- selector channel 33
- send priority 74, 21
 - concentrated message 23
- service facilities 29
- start-stop transmission 9
- start-up 32
- static buffering 5
- station
 - activating/deactivating 29
 - entering operator commands from 29
 - error recording for 30
 - invitation and selection 10
 - remote 1
 - removing from invitation list 29
 - specifying number of temporary error records 30
 - starting/stopping 29
 - testing 32
 - used as alternative destination 29
- station address, displaying 29
- station queue, intercepting/releasing messages 29
- station status, determining 29
- status, displaying for line or station 29
- subblock 17
- subgroup 18
 - in block 18, 22
 - example 20
- subtask trace 30
- switch 18
- switched station, priority 22
- system closedown 32
 - through operator control 29
- system console
 - as operator control station 29

- system console
 - entering operator commands from 29
- system delay 11
 - initializing/altering 29
- system environment, defining 5
- system failure, restart following 31
- system interval, activating and changing 29

- tables, generating 5
- task protection 5
- TCAM 1
 - advantages of queuing 3
 - checkpoint 27
 - environment 13
 - features 5
 - functions 3
 - interfaces 2
 - messages 17
 - service facilities 29
- telecommunication network, examining or altering status 29
- teleprocessing configurations 19, 1
- teleprocessing system 1, 9
 - special characteristics 1
 - configuration 19, 1
- temporary error record 30
- terminal 1
 - devices 34
 - recording I/O errors 30
- terminal and line control 13
- terminal lock 26

- terminal status 31
- terminal table 13
- text error, recovery from 30
- text mode 10
- time-out 10
 - recording 30
- Time Sharing Option (TSO) 23
- trace tables 30
- translation 17
- transmission, controlling message traffic to/from a station 29
- transmission codes 9
- transmission control unit 3, 33
 - testing 32
- transmission priority 21, 22
 - effect on message flow 21
- transmission techniques 9
- transparent mode 10
- TSO (Time Sharing Option) 23
- TSO-only environment 23

- unit 15
 - unit exception, recording 30
 - unit pool 15
- USASCII 9

- work area 21
- work unit 21
 - specifying 22

OS TCAM
Concepts and Facilities
Order No. GC30-2022-2

**READER'S
COMMENT
FORM**

Your views about this publication may help improve its usefulness; this form will be sent to the author's department for appropriate action. Using this form to request system assistance or additional publications will delay response, however. For more direct handling of such request, please contact your IBM representative or the IBM Branch Office serving your locality.

Possible topics for comment are:

Clarity Accuracy Completeness Organization Index Figures Examples Legibility

Cut or Fold Along Line

What is your occupation? — — — — —

Number of latest Technical Newsletter (if any) concerning this publication: — — — — —

Please indicate in the space below if you wish a reply.

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

Cut or Fold Along Line

Your comments, please . . .

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Fold

Fold

First Class
Permit 40
Armonk
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.



Postage will be paid by:

International Business Machines Corporation
Dept. E01
P. O. Box 12195
Research Triangle Park
North Carolina 27709

Fold

Fold

IBM Corporation and Facilities (File No. 2300/23/0-50) Printed in U.S.A. GC30-2022-2



International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)



International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)