CONTROL DATA
CORPORATION

CONTROL DATA®
6000 COMPUTER SYSTEMS

INTERCOM REFERENCE MANUAL
6000 VERSION 3

REVISION RECORD	
REVISION	**DESCRIPTION**
A	Original printing in conjunction with the release of INTERCOM Version 3.0.
(3-19-71)	Features added or changed since the previous version are indicated by a bar
	in the outside margin or by a dot next to the page number if an entire page is
	affected. Additions and changes include:
	Replacement of DIAXNOS and PEANUTS commands with ERRORS command
	Combining LOCK and UNLOCK into the single command, LOCK
	Elimination of the SETUP directive, PERMANENT
	New commands, Q, CONVERT, SAVEFL, and REDUCE
	New Permanent File manipulation commands, STORE, FETCH, and DISCARD
	New program text editor, EDITOR. The SETUP utility is retained in this
	version for compatibility
	Appendix F, EDITOR Commands, and Appendix G, 200 USER Terminal
	Performance Check, for user convenience
Publication No. 60252800	

Additional copies of this manual may be obtained from the nearest Control Data Corporation sales office.

Address comments concerning this manual to:

CONTROL DATA CORPORATION
Documentation Department
215 MOFFETT PARK DRIVE
SUNNYVALE, CALIFORNIA 94086

or use Comment Sheet in the back of this manual

CONTENTS

APPENDIXES

FIGURES

INTRODUCTION

The INTERCOM system, operating in conjunction with SCOPE provides time sharing access to a 6000 Series CONTROL DATA® Computer from a terminal at the central site or at a remote facility. The commands and directives of INTERCOM permit the terminal operator to submit a job for processing on the central computer and also to work interactively with an executing program.

Version 3.0 of INTERCOM supports a paper tape punch and reader at a Teletype terminal, and a card reader and line printer at a 200 USER terminal.

INTERCOM operates in conjunction with the SCOPE operating system to provide the above capabilities. In addition, INTERCOM has a library of utility programs which enable the user to manipulate files and edit program texts.

The INTERCOM user should be familiar with the SCOPE operating system and with the language of his source program. INTERCOM commands and directives are described in this manual.

The following 6000 series reference manuals provide further information on the CONTROL DATA 6000 series computers, the SCOPE operating system, and the compilers available for use with INTERCOM.

Computer Systems Reference Manual, Publication No. 60100000

SCOPE 3 Reference Manual, Publication No. 60305200

ALGOL Reference Manual, Publication No. 60306100

BASIC Reference Manual, Publication No. 60305000

COBOL Reference Manual, Publication No. 60253000

COMPASS Reference Manual, Publication No. 60190900

FORTRAN Reference Manual, Publication No. 60174900

FORTRAN Extended Reference Manual, Publication No. 60176600

SYSTEM DESCRIPTION 1

PURPOSE

INTERCOM directs the flow of information and data between a 6000 series computer and a number of Teletype and display terminals. In addition to the 6000 computer batch processing capabilities, terminal users are provided with the ability to:

Create programs by entering source statements at the terminal

Create, store, reference and edit files

Submit newly created programs to be run interactively through INTERCOM or submit them to a SCOPE batch queue for processing

Submit FORTRAN (Run or Extended), ALGOL, COBOL, COMPASS and BASIC programs for execution under INTERCOM control and interact with the executing programs from the terminal

Request output from jobs run in a batch queue to be directed to a remote terminal or central site printer

Enter SCOPE control cards for processing

Though the user may, with the aid of INTERCOM, create and submit programs for compilation and execution, the processing of such programs is done by the various compilers either through INTERCOM or in the batch queue. If processing is requested by an INTERCOM command or directive, the job is run through INTERCOM and the user may interact with the executing program. If the job is placed in the job stream by the BATCH command or through a remote terminal card reader, it is run in the batch queue.

TERMINALS

In the INTERCOM system any of the following terminals may be used:

Model 33 or 35 Teletype Terminal

Model 214-11, 214-12, 217-11, or 217-12 Display Terminal (BCD)

Model 217-13 or 217-14 Display Terminal (ASCII)†

200 USER Terminal

The 200 USER terminal is a 217-11, 217-12, 217-13, or 217-14 display console with a card reader and line printer. In this manual, either model Teletype terminal may be referred to as a Teletype terminal or TTY terminal; any model display terminal and the 200 USER terminal may be referred to as a display terminal or CRT terminal.

Any terminal with a compatible interface may be substituted for the specific models listed above.

†American National Standard Code for Information Interchange.

Each terminal has a keyboard for entering instructions to the central computer. As each instruction is entered, it is displayed on a teletypewriter printout at a Teletype terminal or on the screen of a display terminal.

Also, either Teletype terminal may be equipped with a paper tape punch and reader. Operation of both CRT and TTY terminals is described in section 2.

SYSTEM RESPONSE

The INTERCOM system responds to each user entry with a message intended to aid the user in choosing his next entry. A response may be diagnostic: such as, CONTROL CARD ERROR when the syntax of a command is incorrect; or the response may be informative: such as, PROGRAM TRANSFERRED TO COMPILER. The most common responses are:

COMMAND-	which asks the user to enter another command
. .	which requests the next command when the system is under control of EDITOR.
READY	which requests the next directive or command when the system is under control of SETUP

INTERCOM responds to the RETURN key (TTY) and the SEND key (CRT) as follows: when the user presses RETURN, INTERCOM sends a line feed; when he presses SEND, INTERCOM moves the entry marker to the next line.

In any case, the system response should be immediate, though response time may be affected by the type of entry or by the number of terminals logged in simultaneously (appendix E). The specific messages that INTERCOM may return following entry of an INTERCOM command or directive are given in appendix C. Appendix B contains the error messages returned by the system. In every case, corrective user action is suggested.

JOB PROCESSING

Processing user jobs is the primary function of the INTERCOM system. The INTERCOM user at a remote terminal can enter a job for interactive execution or for processing in the SCOPE batch queue.

INTERACTIVE PROCESSING

A source program may be constructed at the terminal keyboard and then entered for compilation and interactive execution under control of INTERCOM commands. Interactive execution means that the user's program can request input from the terminal and direct output to the terminal. INTERCOM commands may also be used to access a program residing on mass storage for subsequent compilation and interactive execution.

At a Teletype terminal, a program file can be punched on paper tape and subsequently entered from the terminal for compilation and interactive execution. INTERCOM commands are used to read the paper tape and control the compilation and execution.

BATCH PROCESSING

Jobs may be submitted from a remote terminal to the SCOPE input queue at the central site for batch processing by SCOPE. Batch processing means that the job is processed by SCOPE at the central site. The job must have been saved previously on a mass storage device; it is entered in the SCOPE input queue with an INTERCOM command. The output from execution can be printed on a central site line printer or directed by user request to any logged in user. This output can later be recovered or examined by that user.

At a 200 USER terminal, the INTERCOM user can enter his job as a card deck through the terminal card reader for batch execution by SCOPE. Output resulting from execution is returned to the terminal line printer unless it is directed specifically to a central site line printer or to a line printer at another terminal.

Any program file submitted for SCOPE batch processing must contain the necessary control cards as the first logical record. These control cards may include any applicable SCOPE control cards.

FILES

INTERCOM processes five types of files:

User private files

Permanent files

Input files

Output files

Common files

User private files are local files created by the individual user; they can be read, altered, or deleted only by the originator. They are dropped automatically when the user logs out.

Permanent files are mass storage files, the location and identification of which are always known to the INTERCOM system. Permanent files are protected from unauthorized access according to privacy controls specified by the creators of the files. They are also protected from destruction; they are saved, even across normal system initiation until the user releases them. If a private file is to be saved as a permanent file, it must reside on a permanent file device. To ensure that a file resides on such a device, a SCOPE control card command:

```
REQUEST,filename,*PF.
```

may be entered from the terminal before the private file is created.

Input files are mass storage files resulting from batch jobs submitted either at the remote terminal or the central site. They contain the control card images and input data to be executed through normal SCOPE processing. They are equivalent to a SCOPE system INPUT file, but should not be confused with the user's private file named INPUT.

Output files are the files remaining upon completion of batch jobs; they can be printed, punched, plotted, sent to some other output device at the central site, or sent to a remote terminal. They should not be confused with the user's private file, OUTPUT. The private file, OUTPUT, is generally in a connected state; it should not be used for a user's own information.

Common files are local files which originated as user private files but have been specified as common by the originator. The originator can be an INTERCOM user or other SCOPE user. Common files may be read, altered or deleted by all users including those at the central site.

Files are organized in the same format for SCOPE and INTERCOM.

The program text editor commands (section 4) or the SETUP utility directives (section 5) may be used to create and edit files. To examine files, the PAGE utility (section 3) may be used.

If a local file is created with the same name as an INTERCOM command, the system attempts to execute the local file instead of the command, which prevents access to the command until the file is returned.

Each user is allowed a maximum number of files. When this number is exceeded, access to additional files is blocked until he has dropped or otherwise removed the excess files from his list of private files. Private files are dropped with the SCOPE control card statements RETURN or UNLOAD. All private files are dropped automatically at LOGOUT. Permanent and common files are saved across LOGOUT and subsequent LOGIN. Permanent files are eliminated with the INTERCOM command, DISCARD, or the SCOPE control card statement, PURGE.

SECURITY AND ACCOUNTING

A terminal user may log into the INTERCOM system only if he enters a valid password defined in the INTERCOM password file. If the password is restricted, he also enters a user name specifically defined as valid for the password. If the password is unrestricted, the user name is entered, but it is not tested for validity. A restricted password is more generally used; an unrestricted password is used primarily on a short-term basis for demonstration or classroom purposes.

The password file maintained by INTERCOM prevents unauthorized access to the INTERCOM system, user private files, or SCOPE system files. It also permits accurate accounting of user activity. Each logged-in user is assigned a user identification code associated with his password entry.

Each password entry is associated with a specific field length and time limit. In addition, the password entry sets a limit to the number of private files permitted the user and specifies the access level, that is, the level of commands available to the user. INTERCOM keeps account of the central processor, peripheral processor, and terminal time, and the number of files used for each password/user name combination.

TERMINAL OPERATION 2

Teletypewriter and display terminals have somewhat different operating procedures and keyboards. In addition, methods of connecting terminals to the central computer differ; therefore, methods of initiating terminals may vary. Operating procedures and keyboards for standard teletypewriter and display terminals are described in this chapter.

The command structure and the sequence in which commands are entered are virtually the same for both teletypewriter and display terminals; therefore, the command descriptions in succeeding chapters are the same for both types of terminal.

INITIATING TERMINAL

Teletype and display terminals may be connected to the central site via DATA-PHONE data sets, or they may be connected via a dedicated line. In either case, the terminal must be turned on by the POWER ON switch on the keyboard panel to the right of the display keyboard or by the ORIG button on the Teletype console. If the teletypewriter has a LINE/OFF/LOCAL switch, it must be turned to LINE to be connected to the terminal, to LOCAL for off-line paper tape operation. The ATTENDED/UNATTENDED switch on the display keyboard must be set to ATTENDED. In addition, on the 214-11, 214-12, 217-11, and 217-12 display consoles, the LINE/BLOCK switch must be set to LINE.

DEDICATED LINE OPERATION

If a terminal is connected to a dedicated line, the user may enter commands from the remote terminal keyboard as soon as the terminal is turned on. No dialing is required.

DATA SET OPERATION

If a terminal is connected to the central site with a DATA PHONE data set, the user must establish telephone connection between the terminal and the computer.

DISPLAY TERMINAL

The following procedure links the display terminal to the computer:

1. Lift the data set receiver from its cradle, press the TALK button, and wait for the dial tone.
2. Dial the telephone number of the line to be used; the telephone will ring.
3. If a normal telephone busy signal is returned, the line is already in use.
4. If the phone is answered with a high pitched tone, press the DATA button on the phone and replace the receiver. The terminal is now connected to the computer.

TELETYPE TERMINAL

The action required to connect a Teletype terminal with the computer differs between the model 33 and model 35 teletypewriters. The following instructions pertain to the versions of each model illustrated in figure 2-1 and 2-2. If the Teletype console does not correspond to those illustrated, consult a representative of the company supplying the Teletype for the exact procedure to follow.

For both models:

1. Press the ORIG button to the right of the Teletype console.
2. Dial the number of the line to be used.
3. A busy line is indicated by a normal telephone busy signal.
4. If a high pitched, not busy signal is returned:

   For the Model 33: The terminal is connected automatically.

   For the Model 35: Press the K button at the bottom left of the console to connect the terminal keyboard.

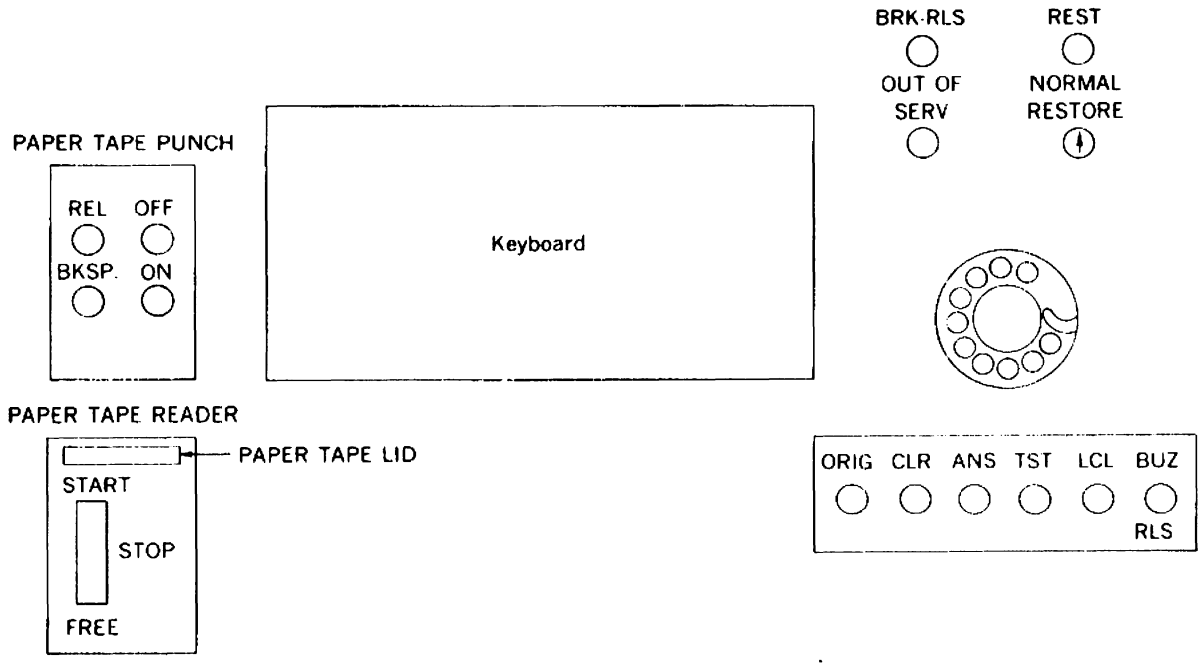


Figure 2-1. Model 33 Teletype

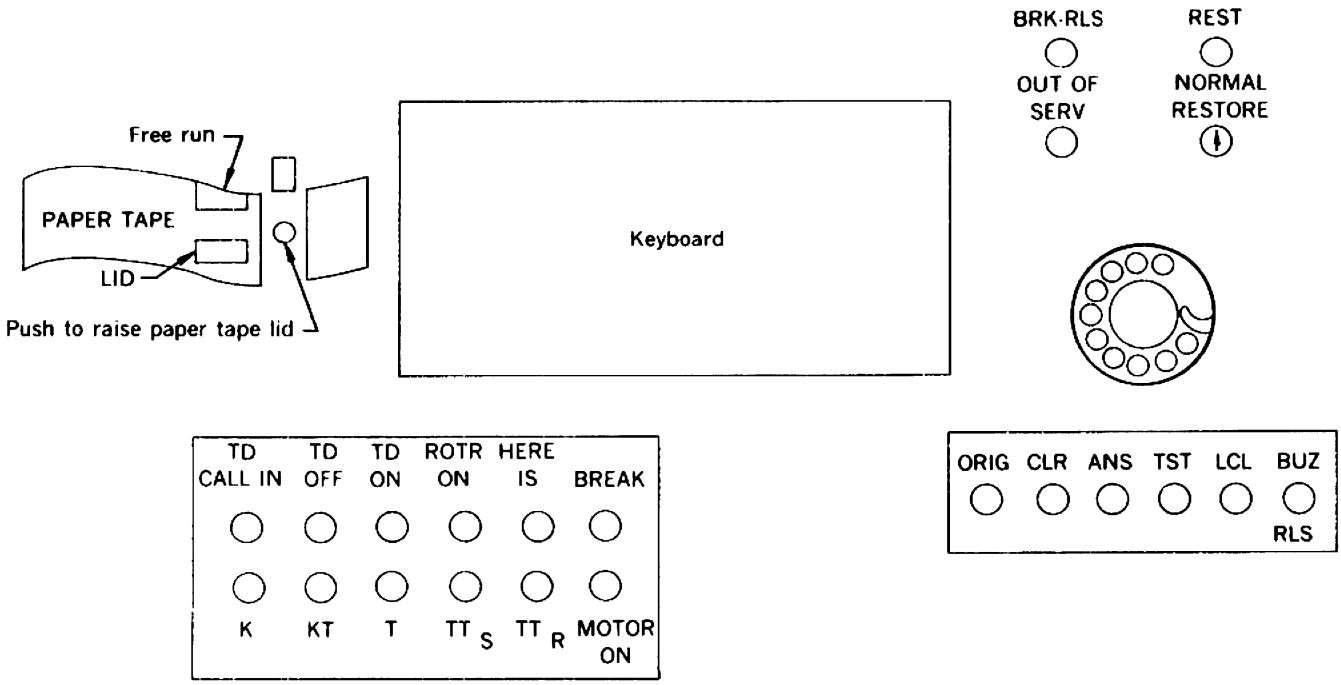


Figure 2-2. Model 35 Teletype

DISCONNECT RECOVERY

The user's terminal may become disconnected from the central computer during an INTERCOM session as the result of problems at the remote or central site or in the communications line. It is possible, in most situations, for the user to recover all his private files existing at the time of disconnect.

TERMINAL DISCONNECT

A disconnect can occur because of communication transmission failure, remote and central site hardware problems, or the inadvertent disruption of telephone or data set connections. Entry of an input character while output is being received at a Teletype can also cause a disconnect.

On a Teletype terminal, a disconnect has occurred if pressing the RETURN key does not produce a line feed response. On a CRT terminal, the console control panel ON LINE button must be flashing. If not, a disconnect has occurred.

INTERCOM allows the user approximately five minutes in which to reinitiate the communication link and log back into the system. At a terminal connected to a dedicated line, communication is resumed automatically; the user may simply log back in.

The user may attempt reinitialization of communication either from the same terminal or any other available terminal, but he must log in with the same user name and password previously entered.

If reinitiation is successful, all the user's private files are recovered. If he was under control of EDITOR, he is returned to EDITOR command mode with his edit file intact. If a program, INTERCOM or SCOPE command, or the SETUP utility was executing when the disconnect occurred, the user is returned to INTERCOM command mode and the program, command or utility must be reinitiated.

If the user is unable to reinitiate communications and log back in within the required five minutes, he is logged out automatically by INTERCOM and all private files are lost.

CENTRAL SITE SYSTEM DISCONNECT

A system failure at the central site will disconnect all terminals. If recovery of the system is successful, reconnect is automatic on dedicated line terminals; but for Teletype and 200 USER terminals, the user must reinitiate the communication link. Under these conditions, he may log back into the INTERCOM system with the same user name and password and recover all his private files.

In this case, INTERCOM retains the user's pre-disconnect file status until the system is shut down under normal conditions or until an unsuccessful system recovery is encountered. The user may attempt re-establishment of communications and log back into INTERCOM at any time during this period.

TELETYPE TERMINAL

TELETYPE ENTRY

Data is entered into the INTERCOM system through the Teletype keyboard in the form of a series of lines of indefinite length. A line is entered by typing the characters and then pressing the RETURN key. A physical line on the teletypewriter printout is limited to 72 characters; but this limitation does not affect the line entered in the INTERCOM system. The LINE FEED key may be pressed at any point in a line to return the carriage to the beginning of the next line; however, a line will not be entered into the system until RETURN is pressed. Thus, lines of more, or less, than 72 characters may be formed and entered.

When output lines are longer than 72 characters, the INTERCOM system issues a carriage return and line feed after 72 characters to prevent overprinting of lines. Some Teletype models are equipped with an automatic carriage return/line feed feature which also issues a carriage return and line feed after 72 characters; on these models, long output lines will be double spaced.

ERROR CORRECTION

Typing errors may be corrected either by deleting incorrect characters from the line, or by discarding the entire line and retyping it.

CTRL H:

Holding the CTRL key while pressing the letter H (CTRL H) causes the last valid character to be ignored, effectively backspacing the character string. Any number of backspaces may be entered in succession; however, when the beginning of the line is reached, further backspaces are ignored.

If the CDC 63-character graphic set is selected as the installation standard, an alternate method of deleting incorrect characters is available. The user may hold the SHIFT key down while pressing the letter O (SHIFT O). This causes the last valid character to be ignored, effectively backspacing the character string although the carriage moves forward. Either a back arrow or an underscore is printed depending on the teletypewriter model; no character is stored.

CTRL X:

Holding the CTRL key while pressing the letter X (CTRL X) causes the entire line of characters to be ignored so that it may be re-entered. The line is not erased from the typewritten page; but it is effectively erased, since nothing between the previous RETURN and CTRL X will be sent to INTERCOM. When CTRL X is pressed, the carriage is returned to the beginning of the next line.

INTERRUPT

Occasionally the Teletype user may want to interrupt current activity. For instance, he may want to interrupt the printing of output or terminate execution of an interactive job.

Pressing the CTRL Z, ESC or ALT MODE key, depending on the teletypewriter model, stops current activity and prepares INTERCOM to accept a directive defining the next action. Any key pressed will interrupt output to the terminal; CTRL Z, ESC, or ALT MODE are preferable because they are least likely to disconnect the terminal. If the teletypewriter is printing output from the computer, the user should wait until output stops before he enters the directive. The following directives can then be entered:

A	Terminates the job currently running (user abort). Output is discarded. The message USER ABORT is sent to the terminal followed by COMMAND-. The user may then enter another INTERCOM command. If EDITOR was in control, the user is returned to EDITOR command mode and may enter any legal command.
S	Stops output; all output currently awaiting transmission to the terminal is discarded. If more output is generated, it will be transmitted; otherwise, there is no response and input may be typed immediately.
LINE FEED RETURN	Pressing the line feed key and then the carriage return key following an interrupt will cause the interrupted activity to continue. RETURN is sufficient to resume the interrupted output, but LINE FEED is necessary to avoid overprinting. A few characters may be lost because of the interrupt.

The interrupt may not be recognized at the time it is entered and may have to be repeated.

TELETYPE KEYBOARD

The Teletype keyboard (figure 2-3) for both models 33 and 35 resembles a standard typewriter keyboard. Special characters shown on the upper portion of the keys are entered by holding the SHIFT or CTRL key down while pressing the special character key. INTERCOM requires special function keys as well as special characters. The keys used and interpreted by INTERCOM are described below; information is included as to what is stored, what action is taken, and what is printed on the Teletype listing.

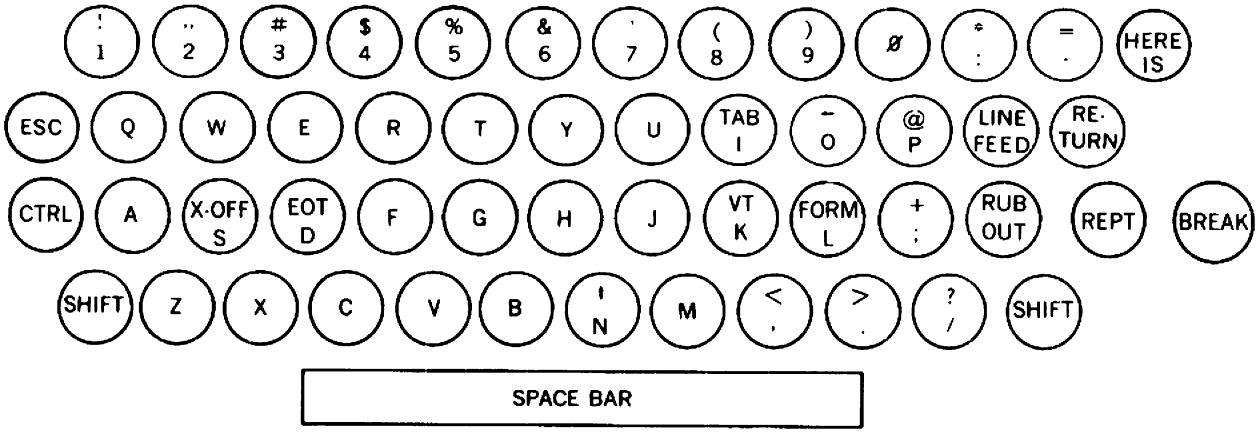


Figure 2-3. Typical Teletype Keyboard

SHIFT	SHIFT accesses the characters or functions shown on the upper portion of most Teletype keys. If pressed alone, SHIFT has no effect.
CTRL	This key is used to access the special function keys (TAB, X-OFF, EOT, etc.) and character and line delete functions. If pressed alone, it has no effect.
RETURN	The carriage return key signals, to INTERCOM, the end of a message. It also returns the Teletype printer carriage to its leftmost position; the computer returns a line feed to advance the carriage to the succeeding line. No character is stored or printed.
LINE FEED	This key spaces to the next line. The system issues a carriage return to the beginning of the new line. No end of message signal is sent. This method provides for entering lines greater than 72 characters. No character is stored or printed.
TAB (CTRL I)	TAB requests a skip to the next logical tab position as defined by the current specification when entering lines in the SETUP utility. The correct number of blanks are stored; nothing is printed.
SPACE	Pressing the space bar generates the space character. A blank is stored and printed.
CTRL H	Pressing CTRL H signals INTERCOM to erase the previous character from its input buffer. The Teletype listing is not erased. For instance, if the user types RUNY and then backspaces and types an X to replace the Y, the listing appears as RUNYX. The corrected command RUNX is entered in the buffer. No character is stored or printed.
SHIFT O (63 character set only)	If the 63-character set is the installation standard, SHIFT O may also be used to delete characters in the input buffer. For example, if the user types RUNY and then enters a SHIFT O followed by an X to replace the Y, the listing appears as RUNY←X (RUNY__X on some models). The corrected command RUNX is entered in the buffer. No character is stored.
CTRL X	Pressing CTRL and X deletes the entire message typed by the user since the previous RETURN. The message is not erased from the printout, but it is ignored by INTERCOM. The computer returns the carriage to the start of the next line. No character is stored or printed.
CTRL Z (ESC or ALT MODE)	CTRL Z, (on some models ESC or ALT MODE may also be used) is pressed to interrupt current Teletype activity. The user then enters a directive: A, S, or a line feed/carriage return to define terminal action.
Alphanumeric	The alphanumeric keys are used to input commands, data and programs. Each is stored and printed as the key is pressed.

PAPER TAPE OPERATION

If the teletypewriter is equipped with a paper tape punch and paper tape reader, paper tapes may be prepared off-line on the punch and input to the central computer with the reader. Output can be received as punched tape under control of the user's program.

The INTERCOM system recognizes many of the special function keys that appear on the Teletype keyboard, among them X-OFF and EOT. The following discussion of the paper tape punching and reading assumes that either X-OFF or EOT will be transmitted to INTERCOM from paper tape. Depending on the model or version of Teletype equipment, this may not be the case. Further, paper tape or keyboard entry of EOT on some equipment causes an immediate line disconnect.

A description of the paper tape functions available can be obtained from a representative of the Teletype equipment supplier.

PAPER TAPE PUNCH

The following procedures describe how a paper tape can be prepared on a Teletype equipped with a paper tape punch for eight-level paper tape:

OFF-LINE

1. If the terminal is connected to the computer, first LOGOUT, and then disconnect it by pressing the CLR button followed by the LCL button.
2. Model 33: Press punch ON button.

   Model 35: Press KT button.
3. Punch two or three inches of rubouts by holding REPT down while pressing RUBOUT.
4. Type the characters of an input line on the keyboard.
5. Type LINE FEED once, RETURN once, and RUBOUT at least three times to guarantee that no characters are lost.

If another line is to be input, repeat steps 4 and 5.

The last line punched on the tape should consist of the entry:

```
TAPE,OFF.
```

This should be followed by the LINE FEED, RETURN, three RUBOUTS, X-OFF (CTRL S), and several inches of RUBOUT as at the start of the tape. The EOT character (CTRL D) may be substituted for X-OFF; on a Model 35, EOT is sent automatically when the tape runs out of the reader.

If X-OFF followed by three rubouts is punched anywhere on the tape, tape reading will stop automatically at that point to allow accumulated output to be returned to the terminal. Tape reading will resume automatically when all output has been returned.

Errors made while punching paper tape can be corrected by pressing the BKSP button (on Model 33 paper tape punch panel, on Model 35 at lower right of keyboard) and then punching RUBOUT over each mispunched character. RUBOUT is used for timing only and is ignored in the text of a punched tape.

ON-LINE

Paper tape can also be punched at a logged in terminal under control of a user's program which is executing interactively. No special requests are needed in the program generating the output; if the paper tape punch is turned on, any output normally sent to the terminal is punched on the tape. The necessary line feed, carriage return, and timing characters are provided automatically; however, if X-OFF is wanted, it is added from the keyboard.

On either model, the teletypewriter is turned on and connected to the computer. If not logged in, the user should log in and then turn on the paper tape punch. On a Model 35, either the tape alone or the tape and printer can be selected to receive the output by pressing KT (keyboard and tape) or TTR (tape only) in addition to turning on the punch. On a Model 33, simply turn on the punch, the keyboard and printer will be on automatically.

A punched tape has the format shown in figure 2-4.

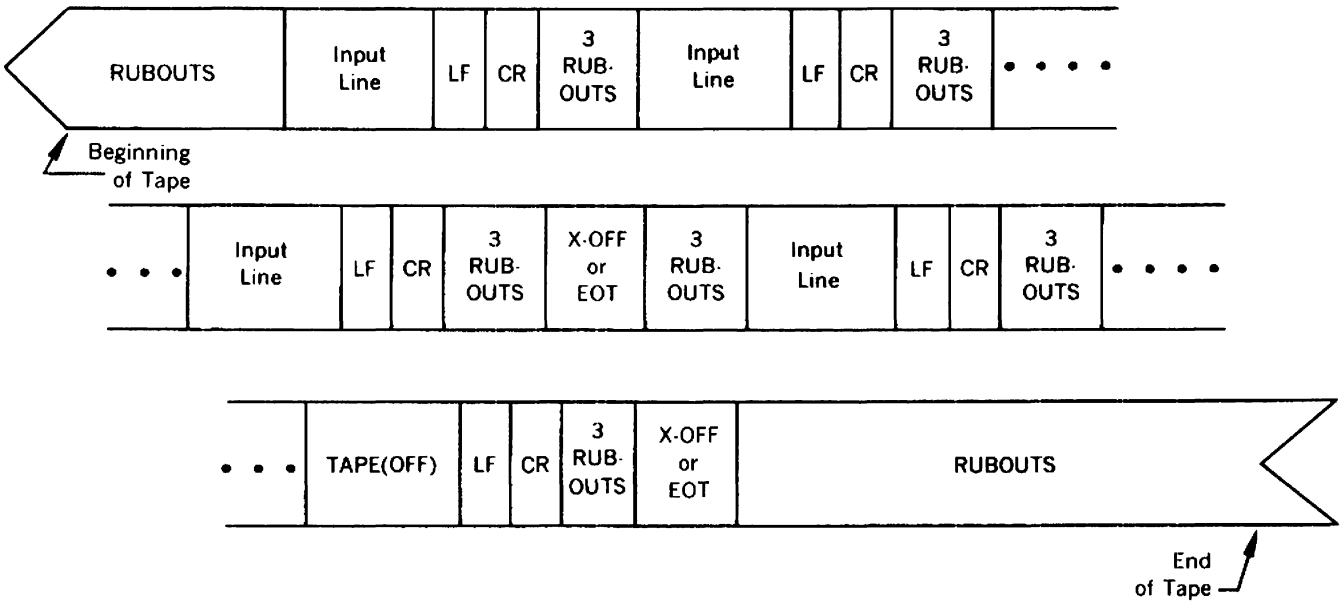


Figure 2-4. Punched Tape Format

PAPER TAPE READER

A paper tape is read using the TAPE,ON. command. The end of reading is signaled to INTERCOM with the TAPE,OFF. command unless an EOT or X-OFF character is punched on the tape.

The procedure for entering input to the central computer through the paper tape reader at a Teletype terminal is:

1. Load the paper tape in the reader: open the lid and place the tape across the cogs of the feeder wheel so the cogs fit the feeder holes punched in the tape leader, and the tape moves in the right direction (usually arrows are printed on the tape). Then snap the lid shut.

2. Model 33 only: Set reader switch to STOP position.

   Model 35 only: Activate TD CALL-IN button (left of keyboard); if forgotten, turn it on later, and then press TD ON.

3. Enter the command: TAPE,ON. from the keyboard.

4. Wait for system response from INTERCOM (COMMAND-), EDITOR (. .), or the SETUP utility (READY.).

5. Model 33 only: Set reader switch to START position; reading begins. (On some terminals, reading begins automatically.)

   Model 35 only: Tape reading begins automatically.

The tape will be read until X-OFF or EOT is encountered, or the tape runs out of the reader. Any output accumulated for the user will be returned to the terminal at this time. Tape reading will resume automatically if there is more data on the tape. If the tape runs out of the reader and TAPE,OFF. was not punched on the tape, the user types TAPE,OFF. followed by a carriage return and line feed; and then enters X-OFF or EOT to return the terminal to normal keyboard entry mode. Otherwise, tape reading will continue until the INTERCOM command TAPE,OFF. is received as part of a message.

DISPLAY TERMINAL

The display terminal is equipped with a cathode ray tube (CRT) display screen, a display controller, and a keyboard. In addition, the 224- 1 or 224-2 Card Reader and/or the 222-2 Line Printer may be included to provide a full 200 USER terminal.

DISPLAY SCREEN

The display format is an installation option. The standard formats are either 20 lines of 50 symbols per line or 13 lines of 80 symbols per line. If the screen format at a terminal differs from the standard screen for the installation, the user should enter the SCREEN command (section 3) immediately after LOGIN to notify INTERCOM of this difference. INTERCOM can then format displays to fit the terminal screen. Otherwise, the screen may not be fully utilized, or else information may be overwritten because of screen wrap-around.

DISPLAY ENTRY

Data may be entered at a display terminal in the form of one or more lines up to the number of lines that can be displayed on the screen. A line is entered by typing the desired characters and then pressing SEND. The physical line is limited by the screen size to 50 or 80 characters; the line entered in the INTERCOM system may be greater or less than the physical line. Pressing the RETURN key at any point in a line allows the user to continue entering data on the next line since nothing is transmitted to INTERCOM until the SEND key is pressed. On the 214-11, 214-12, 217-11, or 217-12 display console, a line must be transmitted when it reaches the end of the bottom line on the screen; or it will be lost. The hardware does not allow transmittal of a line that has wrapped around to the top of the screen. This restriction does not apply to the 217-13 or 217-14 display console.

Three indicators on the display screen facilitate message transmission: send indicator, entry marker, and message terminator.

SEND INDICATOR

The send indicator designates the beginning of the message to be transmitted. The use of this indicator is dependent on the type of display terminal.

On the 214-11, 214-12, 217-11, and 217-12 display terminals, the send indicator is a small block appearing on the screen to the left of the line on which the next message will start. During input from the keyboard, the send indicator is affected by two keys: SEND INDEX advances the indicator one line; CLEAR repositions the indicator to the top line. The indicator moves automatically to the next line at the completion of an output operation to the display.

On the 217-13 and 217-14 display terminals, the send indicator is a small block which may appear in any character position on the screen. All output messages to the display are terminated with a send indicator at the end of the last output line. The subsequent input operation transmits data from the send indicator to the message terminator unless:

The user presses the SEND key immediately after the send indicator; in which case, extraneous data may be transmitted

The user presses the CLEAR key before entering input; transmission consists of data from the top of the screen to the message terminator

The user presses the SEND INDEX key as part of his input, in which case, the send indicator moves to the current character position

ENTRY MARKER

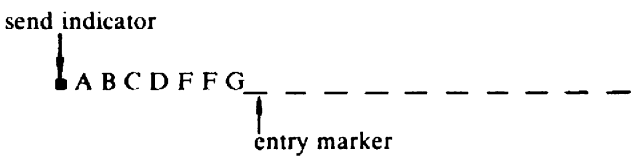
The entry marker is the first of a series of dashed underlines; it indicates the next open character position; the series extends to the right margin of the line to indicate remaining character positions. Initially, the underlines extend from left to right margins of the top line of the screen. As a symbol is entered, the underline in that position disappears, and the next underline in the series becomes the entry marker. When the end of a line is reached, the series of underlines is displayed on the next lower line. When the end of the last line is reached, the underlines reappear on the top line.

MESSAGE TERMINATOR

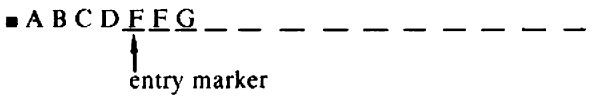
The symbol, ▲ , is used to indicate the end of a message. This symbol is entered when the SEND key is pressed during a write operation.

The following example shows how the send indicator, entry marker, and message terminator are used. Other special keys referred to in the example are explained subsequently.

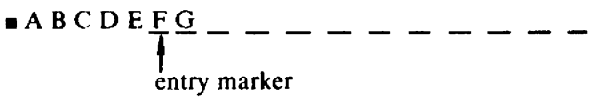
Assume the letter E has been improperly entered as F in the following string of characters.



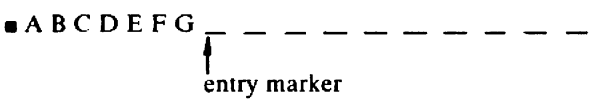
Press BKSP key three times to move entry marker under F:



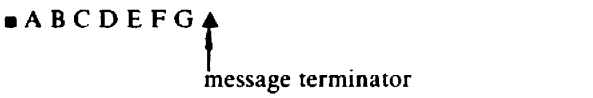
Correct F by entering E:



Enter two SKIPS to reposition the entry marker:



If SEND key is pressed:



appears on the display, and the message A B C D E F G is sent.

ERROR CORRECTION

Any errors in an input line or command format may be corrected prior to pressing the SEND key.

The entire line can be discarded by pressing LINE SKIP followed by SEND INDEX. Then a new line may be typed and sent to INTERCOM by a subsequent SEND; the line typed prior to LINE SKIP and SEND INDEX is not sent.

Individual characters may be corrected by pressing the BKSP key until the entry marker is positioned below the incorrect character (as shown in above example). When the correct character is typed, it will replace the incorrect character on the screen.

OUTPUT PAGING

If enough data is accumulated on output to fill more than screen limits allow, a single screenful is displayed; and INTERCOM waits for a directive from the user before displaying more data. On the 214-11, 214- 12, 217-11, and 217-12 keyboards, the user may continue the display by pressing the SEND key. On the 217-13 and 217-14 keyboards, he may continue the display by pressing the plus key (+) followed by the SEND key.

INTERRUPT

The user may wish to interrupt terminal activity under two different situations:

During interactive execution of his program or command

During input from the card reader or output to the line printer

He may interrupt the job display by pressing % (SHIFT 5) followed by S or A.

%S — Stops current output and discards the contents of the output buffers waiting to be sent to the terminal. Any output from subsequent execution of the job is sent to the terminal. If there is no further output, the user may enter input from the terminal keyboard.

%A — Terminates the job currently executing and discards all output from the job. The message USER ABORT is sent to the terminal followed by COMMAND-. The user may then enter an INTERCOM command. When under control of EDITOR, the user is returned to EDITOR command mode, he may enter any legal command.

In the second case, the keyboard is locked during transmission from the card reader or to the line printer. In order to halt this transmission and unlock the keyboard, the user must press the INTER key and keep it pressed down until the output stops. When output halts, the keyboard will unlock. The user should press CLEAR to clear the display screen; then he may enter a command from the keyboard.

DISPLAY KEYBOARD

The display keyboards (figure 2-5 and 2-6) resemble a standard typewriter keyboard. Some keys have a different function, and some are unique to the display keyboard. In the description of the keys used and interpreted by INTERCOM, information is included as to what is stored, what action is taken, and what is printed on the display screen.

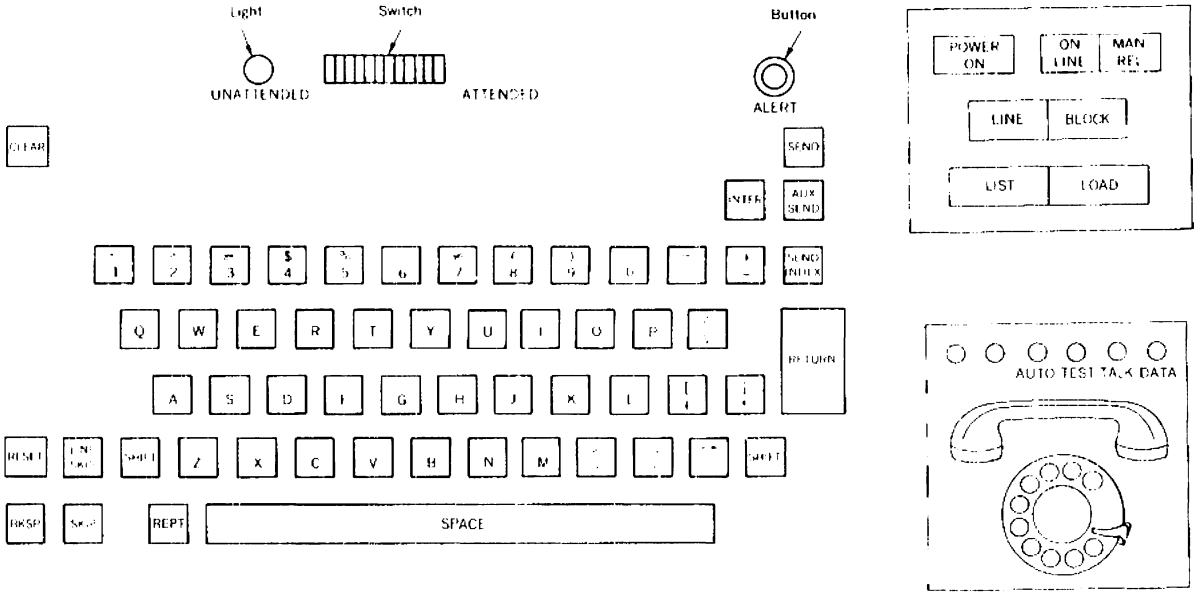


Figure 2-5. Standard 214-11, 214-12, 217-11, and 217-12 Display Keyboard (BCD)

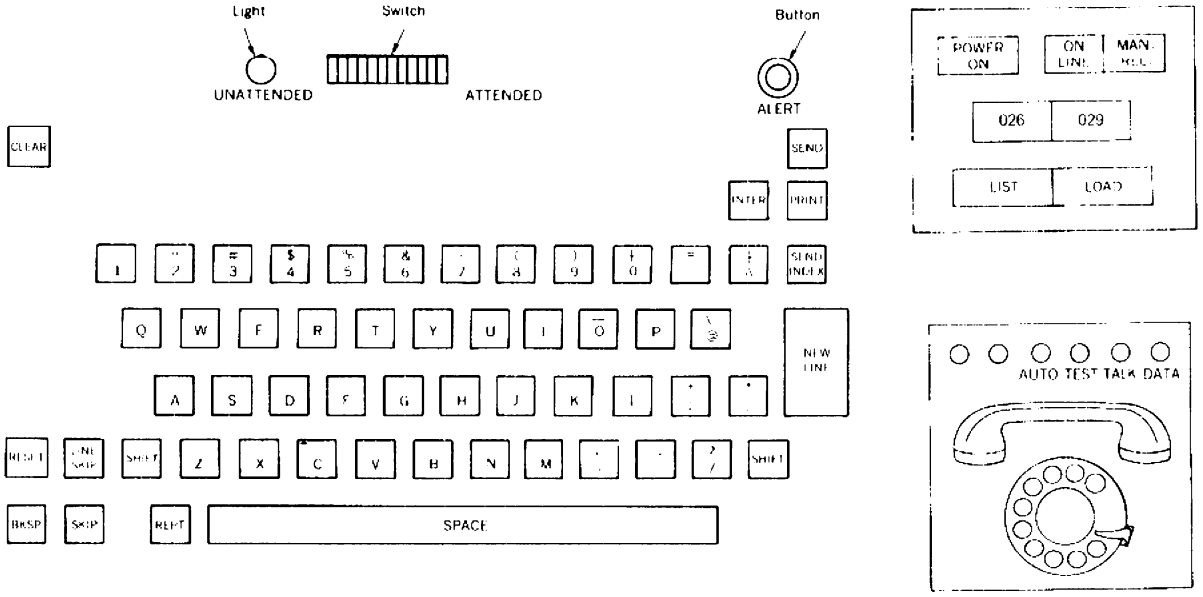


Figure 2-6. Standard 217-13, and 217-14 Display Keyboard (ASCII)

The following special keys are of particular importance to the user:

SEND — This key initiates transmission of a message from the terminal to INTERCOM. When reading or printing is not in progress, a command may be entered by pressing SEND after the command. When SEND is pressed, the symbol ▲ appears on the display at the end of the message and the information displayed between the send indicator and ▲ is transmitted. On the 217-13 and 217-14 displays, if SEND is pressed immediately after the send indicator, the ▲ symbol is not entered, and extraneous data may be transmitted.

INTER — The keyboard is locked while data is transmitted from the card reader or to the line printer. Pressing the INTER key interrupts data transmittal and releases the keyboard. The key must be pressed until transmittal stops. The keyboard unlocks when the display buffer is free; then the user may press the CLEAR key and enter a message.

% — During interactive execution, the user can interrupt and then discard the current output from his job by entering % followed by S. He can terminate the current job and discard all its output by entering % followed by A.

RETURN — The RETURN function is similar to LINE FEED on the Teletype. It returns the entry marker to the left of the next line without transmitting the message. This enables the user to control the format of his message. If RETURN is not pressed and a message longer than 50 or 80 characters is typed, the entry marker returns automatically to the right margin. When the message is transmitted, carriage return characters are ignored.

↑ or]	On the 214-11, 214-12, 217-11, and 217- 12 display terminals, one of these keys (depending on the installation character set) is used to request tabulation to the next position specified by the TAB directive of SETUP (section 5). Tabulation is not apparent on the display screen; however, spaces are entered into the character string when it is transmitted to INTERCOM. The character is displayed on the terminal screen whenever it is pressed.
! or ?	On the 217-13 and 217-14 display terminals, one of these keys (depending on the installation character set) is used to request tabulation to the next position specified by the TAB directive of SETUP (section 5). Tabulation is not apparent on the display screen; however, spaces are entered into the character string when it is transmitted to INTERCOM. The character is displayed on the terminal screen whenever it is pressed.
SKIP	The skip key moves the entry marker forward one character position each time it is pressed. It does not alter previously entered data. Nothing is stored.
BKSP	The BKSP key allows the user to backspace and replace previously entered characters. BKSP does not physically erase the characters from the display screen, nor does it cause anything to be stored. It moves the entry marker back one space each time the key is pressed, and the final position of the entry marker indicates where the next character is to be typed. Typing a character in the position indicated by the entry marker replaces any previously entered character.
SPACE	The SPACE bar generates a blank space on the screen.
CLEAR	This key clears the display screen so that the user can begin typing on the first line. It is ignored if pressed while the terminal is receiving output. When the user is typing input it will clear the screen. Nothing is stored.
REPT	This key allows repeated entry of any other key. It is pressed simultaneously with the key that is to be repeated. When pressed alone, nothing is stored.
SEND INDEX	On the 214-11, 214-12, 217-11, and 217-12 terminals, this key moves the send indicator down one line. Nothing is stored, and data currently displayed is not affected. When used in conjunction with LINE SKIP, it effectively allows the user to discard the line typed so far and restart the message on the next line.
	On the 217-13 and 217-14 terminals, this key causes any send indicator on the screen to be replaced with a blank and enters a new send indicator at the current character position.
AUX SEND	The AUX SEND key transmits data currently displayed on the screen to the line printer where it is printed. The transmitted data is whatever appears between the upper left corner and the first entry marker. AUX SEND should only be used at a terminal with a line printer.
RESET	This key moves the entry marker to the top left of the screen without clearing previously entered data. Nothing is stored.
LINE SKIP	This key moves the entry marker down one line to the next line and restores it to the left margin. Nothing is stored.

The alphanumeric keys are used to input commands, data, and programs. Each is stored and displayed as the key is pressed.

In addition to the display terminal keyboard, a panel of buttons is available at the display console (figure 2-5 and 2-6). These buttons may be used in conjunction with the card reader or line printer, as well as to initiate terminal operation.

LINE/BLOCK	This switch puts the terminal in line or block mode. This is applicable only to the 214-11, 214-12, 217-11, and 217-12 terminals. The terminal must be in line mode while using INTERCOM (see Initiating the Terminal, above).
LIST	Pressing LIST causes cards to be listed on the terminal line printer (see Listing Cards, below).
LOAD	The LOAD button starts the card reader and loads the terminal card buffer.
MAN/REL	Pressing this button unlocks the keyboard, resets the entry marker to the upper left corner of the display, temporarily interrupting any communication, and clears the terminal card buffer.
026/029	This switch allows the user to read either 026 or 029 keypunched cards. The setting must correspond with the model number of the keypunch on which cards to be read were punched.

CARD READER OPERATION

Cards may be read from the display terminal card reader with the READ command (section 7). Before entering the READ command, the card reader is initiated as follows:

1. Turn the ON/OFF switch on the card reader to ON.
2. Set the AUTO/MAN switch to AUTO.
3. Press MAN/REL button on the display console once to clear the card buffer.
4. Place the cards to be read in the right-hand feeder, face down with 12-edge toward the user.
5. Press the REG or READY switch on the reader to register the first card.
6. Press LOAD on the display console to start the card reader.

The READ command may now be entered (see section 7 for the syntax of READ). Cards will be read until an end-of-file is reached or until the hopper is empty. If the last two cards are not end-of-file cards, or an end-of-file followed by a blank card, a message is displayed:

```
READER NOT READY.
```

To continue reading, place more cards in the hopper, press LOAD to start the reader and enter the command GO.

The last card in the deck may remain in the input hopper after it has been read. To recover the card, press the READY or the REG button, depending on the card reader, and then the FEED button.

LISTING CARDS

Cards can be listed on the terminal line printer when the terminal is not connected to the central site. Start the card reader and register the cards (steps 1 through 5 of Card Reader Operation), then turn on the printer and press the LIST button on the terminal. The cards will be read and listed until the deck is exhausted. End-of-record and end-of-file cards are listed as lines of zeros. Before listing begins, the user should LOGOUT and disconnect the terminal from the central computer. Otherwise, any data output to the printer from the central site is interleaved with the card listing.

LINE PRINTER OPERATION

Before printing can start, the POWER ON/OFF switch must be turned to ON, and the START button must be pressed. The printer is ready when POWER ON and START are lit. When job output is ready, the user should enter the GO Command (see section 7) to start printing the output. Two banner pages precede the actual printed output from a file and also separate the output between consecutively printed files.

If reading is in progress when printing starts, the two operations will proceed in an interleaved manner. The line printer can be advanced manually to the top of the page by pressing the STOP button and then pressing PAGE EJECT. START must then be pressed to resume printing.

If the printer is not ready initially or becomes not ready during printing, printing stops and this message is displayed:

```
PRINTER NOT READY
```

To initiate printing, ready the printer and press START. Then enter the command GO to resume printing. If the printer stops but no message is displayed, printing should resume automatically when the printer becomes ready again.

If a parity error is encountered in the file being printed, the following message is displayed:

```
OUTPUT FILE ERROR
```

To continue printing, enter the command GO. If the parity error persists, the message reappears. In this case, the file must be repositioned or printing terminated.

Further description of printing output on the line printer is included in section 7 under the output control commands: GO, AGAIN, SUSPEND, END.

COMMAND MODE 3

INTERCOM COMMANDS

The fundamental entry in INTERCOM is the command. Once the user has logged in, and his user name and password have been validated, the system sends the terminal display:

```
COMMAND-
```

Then the user may enter any of the INTERCOM commands described in this section. When a command is entered, control transfers to the utility program which processes the command. When the requested operation has been performed, the utility program returns control to the INTERCOM command mode, and the user may enter another command. Return to command mode is signaled by the display:

```
COMMAND-
```

All user commands are entered by pressing the SEND key at the display keyboard or the RETURN key at the Teletype keyboard.

COMMAND SYNTAX

INTERCOM command structure resembles that of SCOPE control cards. Commands are terminated by a period unless the command has parameters, in which case, it is terminated by a right parenthesis or a period. If more than one parameter is specified, they must be separated by commas. The command verb and the parameters may be separated by either a comma or left parenthesis.

Most commands require a system response before the user can enter further information. Some have an abbreviated form which permits the user to enter additional information, such as file names, along with the command. Such cases are noted in the command description.

SCOPE CONTROL CARDS

Most SCOPE control cards may be entered at the terminal as commands. When the operation requested by the control card command has been performed, control returns to INTERCOM command mode and the response COMMAND- is displayed.

Permissible SCOPE control cards include COPY, COMPARE, REWIND, RETURN, CATALOG, ATTACH, PURGE, EXTEND, and so forth. REQUEST is allowed only to request a permanent file device in the form: REQUEST,filename,*PF. The format, delimiters, and interpretation of the parameters are identical to control cards entered as batch input at the central site except that permanent file control cards may be longer than 80 characters, if necessary.

The following SCOPE control cards may not be used as INTERCOM commands when the user is processing interactive jobs.

AUDIT	EDITLIB
COMMON	RELEASE

These cards are restricted either because they require operator action or because of the internal structure of INTERCOM.

Before any INTERCOM command or SCOPE control card is loaded, the system checks the user's field length and access level to determine whether he is authorized to use the particular command or control card.

Any file entered into the SCOPE batch input queue must have a control card record as the first logical record of the file. Such a control card record can be created at the terminal using the program text editor (section 4) or the SETUP utility (section 5) of INTERCOM. The control cards are typed at the terminal keyboard, card by card, until the record is complete. End-of-record cards are created by typing *EOR in EDITOR or *WEOR in SETUP.

EXAMPLES

The examples shown in this section, as well as in sections 4 and 5, result from operation of INTERCOM at a teletypewriter terminal. The format resulting from operation at a display terminal differs only in that each user response appears on a new line.

User responses in these examples are underlined; in actual operation they are not underlined.

COMMUNICATION WITH INTERCOM

As soon as the user is connected to the terminal (see Initiating Terminal, section 2), the system responds:

```
CONTROL DATA INTERCOM 3.0
DATE          mm/dd/yy
TIME          hh.mm.ss.
  '
```

The user may then log in with the LOGIN command. Once logged in, he is in INTERCOM command mode and may enter another command.

LOGIN COMMAND

The remote terminal user logs into the system by entering:

```
LOGIN.
```

The system responds:

```
ENTER USER NAME-
```

The user name may be up to ten alphanumeric characters, and must not be followed by a period. If the password is restricted, the user name must be unique and must be entered exactly the same each time; a validity check is performed. If the password is unrestricted, the user name is entered but need be unique only among users logged in with the same password; a validity check is not performed in this case.

When the user name has been entered at a TTY terminal, the system responds:

```
■■■■■■■■■■ENTER PASSWORD-
```

At a 200 USER or display terminal, the system responds:

```
ENTER PASSWORD-
```

The user then enters his password. A password is up to 10 alphanumeric characters which must not terminate with a period. The user name and password are compared to a password list maintained by INTERCOM where they comprise a unique identifier for the logged in user. On a Teletype listing, the password is overprinted on the ten-character, blocked-out line to preserve privacy. The display terminal screen is automatically cleared on acceptance of the entered password to preserve privacy.

When the user name and password are accepted, the user id (a two- character user code) and the time at which the user logged in are dislayed at the terminal. The user id is assigned permanently to the user; and unless the password list is changed, he will receive the same user id each time he logs in. INTERCOM returns the message, COMMAND-, and the user may enter any command described in this section. If he is at a display terminal, he also may enter any of the batch processing commands described in section 7

Example:

```
LØGIN.
ENTER USER NAME-USERA
■■■■■■■■■■ ENTER PASSWØRD-

03/10/71  LØGGED IN AT 14.04.02.
          WITH USER ID- AB
CØMMAND-
```

If the user's CRT screen size differs from the installation's standard screen size, he should enter the SCREEN command immediately after the LOGIN procedure is completed.

If the password file has not been created at the central site, the following message is returned after the message ENTER USER NAME-:

```
PERM FILE ERR, RETURN CODE = 12B
```

The user cannot log in until the password file is created.

TEACH COMMAND

The user can request a brief description of the basic commands and directives of the INTERCOM language and also a limited explanation of SCOPE control card commands by typing:

```
TEACH.
```

The system returns a numbered list of topics from which the user selects the one he wants described. The user may be asked to select from a further subdivision of topics, or a brief description may be output.

Teach outputs a maximum of ten lines, and then waits for user action. The user may type GO to continue the description, or END to return to command mode.

Example:

```
C0MMAND- TEACH.

                    TEACH
IF Y0U WANT T0 KN0W-
     H0W T0 USE INTERC0M                TYPE   1
     H0W T0 USE THE TTY                 TYPE   2
     H0W T0 USE THE CRT                 TYPE   3
     A SPECIFIC C0MMAND                 TYPE   4
     A SPECIFIC EDIT0R C0MMAND          TYPE   5
     A SPECIFIC SETUP DIRECTIVE         TYPE   6
     A SPECIFIC BATCH DISP0SITI0N       TYPE   7
     A REM0TE BATCH C0MMAND             TYPE   8
  1

C0MMAND M0DE AND DESCRIPTI0NS 0F C0MMANDS
   THE FUNDAMENTAL ENTRY IN INTERC0M IS A
   C0MMAND. THE SYSTEM INDICATES ITS READINESS
   WITH THE REM0TE TERMINAL DISPLAY-
              C0MMAND-
   THE USER CAN THEN TYPE THE DESIRED C0MMAND,
   F0LL0WED BY A DEPRESSI0N 0F THE SEND KEY (CRT)
   0R THE RETURN KEY (TTY), HEREAFTER SYMB0LIZED
   AS -CR-.  INTERC0M C0MMANDS ARE AS F0LL0WS-
T0 C0NTINUE TYPE  G0  , T0 END TYPE  END-END
11.04.37.ST0P
C0MMAND-
```

Suppose the user selects 4 instead of 1, the system responds:

```
 INTERC0M C0MMANDS--- TYPE C0RRESP0NDING NUMBER
 1 = L0GIN       11 = C0NNECT     20 = ST0RE
 2 = L0G0UT      12 = DISC0NT     21 = FETCH
 3 = FILES       13 = SETUP       22 = DISCARD
 4 = SITUATE     14 = EDIT0R      23 = SAVEFL
 5 = SEND        15 = C0NVERT     24 = REDUCE
 6 = PAGE        16 = BATCH       25 = MAP
 7 = ERR0RS      17 = Q           26 = SWITCH
 8 = ASSETS      18 = M           27 = SCREEN
 9 = ETL         19 = L0CK        28 = TEACH
10 = EFL

 2
```

```
LØGØUT
   WHEN THE USE ØF THE TERMINAL IS CØMPLETED,
   TYPE THE CØMMAND-
          LØGØUT. CR
   THIS CØMMAND DISASSØCIATES THE TERMINAL
   FRØM INTERCØM. THE TERMINAL DISPLAYS THE
   CP AND PP TIME USED (IN SECØNDS), CØNNECT
   TIME (IN HØURS  AND MINUTES), DATE,
   AND TIME THAT TERMINAL IS DISCØNNECTED
TØ CØNTINUE TYPE  GØ , TØ END TYPE  END-END
11.09.27.STØP
CØMMAND-
```

SCREEN COMMAND

The SCREEN command is meaningful at a display terminal only. It informs INTERCOM that the terminal where it is entered has a display screen format that differs from the installation standard.

Screen formats are either 13 lines by 80 columns or 20 lines by 50 columns. Each installation selects a format as a standard and output is displayed in the standard format. If a particular display terminal has a non-standard screen format, the terminal user may enter:

```
SCREEN.
```

The system will send display in the format corresponding to the user's screen, thereby preventing inefficient use of the terminal and possible overlap of displayed output.

If SCREEN is not entered, the standard screen format is assumed. The SCREEN command may not be entered if the user is in SETUP; he must return to command mode to enter SCREEN.

REDUCE COMMAND

On initial LOGIN, an internal system flag is set ON to indicate that the user's field length is to be reduced to the minimum required for execution. Reduction occurs after a program is loaded; field length is reduced to the program's last word address rounded upward by 100 octal. Automatic field length reduction does not occur for overlay generation or absolute overlays which load higher level overlays.

For execution of ALGOL programs, the flag must be reset to OFF because ALGOL utilizes the field length immediately following the last word address. To change the setting of the flag to the OFF condition, the user enters:

```
REDUCE,OFF.
```

Field length reduction will not occur for subsequent program execution. On completion of execution, the flag should be returned to the ON condition. The user enters:

```
REDUCE,ON.
```

Automatic field length reduction is again in effect.

Example:

The following commands may be entered from the program text editor (section 4) to compile and execute an ALGOL program:

```
..REDUCE,0FF.
..RUN,ALG0L
        .
        .
        .
(program output)
        .
        .
        .
..REDUCE,0N.
..
```

SAVEFL COMMAND

The SAVEFL command is used to set an internal flag which indicates to the system whether or not the central memory field length is to be saved after processing of each command or control card is completed. The user may save the field length for subsequent printing at the central site or remote printer, or he may save it for constructing overlays or to produce a load map.

To save central memory field length, the user enters:

```
SAVEFL,ON.
```

The field length used by all subsequent INTERCOM or SCOPE commands, or user's jobs, is saved and then restored in central memory before the next command or job is executed. However, because the field length must be swapped out after each job, response times are increased. The SAVEFL ON condition should be requested only when necessary, and the user should release the field length as soon as possible by entering:

```
SAVEFL,OFF.
```

Example:

If a user determines that one of his programs is not functioning properly, he may obtain a dump of the program area on the file OUTPUT and print this file, at the central site, with the following INTERCOM commands and the SCOPE control card, DMP. The file MYPROG contains the binary code produced from an assembly or compilation.

```
C0MMAND- SAVEFL,0N.
C0MMAND- MYPR0G.
ARITHMETIC ERR0R      M0DE = 1      ADDRESS = 013407

C0MMAND- DMP,20000.
C0MMAND- SAVEFL,0FF.
C0MMAND- BATCH.
TYPE FILE NAME- 0UTPUT
TYPE DISP0SITI0N- PRINT
TYPE FILE ID- 196A
TYPE FILE NAME- END
C0MMAND-
```

The SCOPE control card, DMP, also can be used to obtain a dump of the program field length if a user's program initiated by the EDITOR RUN command terminates abnormally. The SCOPE DMP utility automatically disconnects the file OUTPUT to prevent sending the dump directly to the terminal.

SAVEFL and DMP are not allowed while in SETUP. If the SAVEFL ON condition is in effect when the SETUP command is entered, only the field length of the SETUP utility is saved, not the field length of user programs initiated in SETUP.

MAP COMMAND

When a user logs into INTERCOM initially, a system flag is set to indicate that no load map is to be produced when a user's program is loaded. To obtain a map of his program load, the user may change the setting of the flag with the SCOPE control card, MAP, by entering:

```
MAP,ON.
```

to obtain a full load map, or

```
MAP,PART.
```

to obtain a partial load map (entry point addresses are omitted).

The file OUTPUT should be disconnected from the terminal when the map flag is in the On or Partial condition. Otherwise, the load map generated will be sent directly to the terminal.

The user should reset the map flag to the Off condition as soon as possible by entering:

```
MAP,OFF.
```

To generate a load map of his program, and to save it on the file OUTPUT for subsequent examination, the user may enter the following sequence of SCOPE control cards and INTERCOM commands. The file LGO contains the binary output from an assembly or compilation. The NOGO. control card inhibits program execution. SAVEFL,ON must be specified to save the field length between the LOAD and NOGO control card commands.

```
C0MMAND- MAP,0N.
C0MMAND- DISC0NT,0UTPUT.
C0MMAND- SAVEFL,0N.
C0MMAND- L0AD,LG0.
C0MMAND- N0G0.
C0MMAND- SAVEFL,0FF.
C0MMAND- MAP,0FF.
C0MMAND-
```

SWITCH COMMAND

When the user logs into the INTERCOM system, sense switches 1-6 are set to the Off condition. INTERCOM allows these switches to be reset from a terminal. The user enters the SCOPE control card:

```
SWITCH,n.
n = 1-6
```

Each use of the SWITCH card changes the current condition of the specified switch. For example, the first use of the card SWITCH,4. sets sense switch 4 to the On condition, the second use of SWITCH,4. sets it to Off.

The sense switches may be accessed by, and used to control, the user's executing program. The user should refer to the appropriate reference manual for further information on accessing sense switches from his program.

LOGOUT COMMAND

When the user has completed operation at the terminal, he enters the command:

```
LOGOUT.
```

When the user logs out, all his private files are released. Only permanent files and private files made common are retained between the time of a LOGOUT and any subsequent LOGIN. Common files are usually retained only for one day.

The user is disassociated from INTERCOM until a subsequent LOGIN command is entered. INTERCOM displays the date and time the user is logged out.

The LOGOUT command is not allowed when the user is under control of EDITOR.

Example:

```
C0MMAND- L0G0UT.
CP TIME            16.569
PP TIME           955.398
C0NNECT TIME       1 HRS.      17 MIN.
03/12/71  L0GGED 0UT AT 15.21.09.
```

The order of the date (month, day, year) may be changed as an installation option. The time of LOGOUT is given in hours, minutes, seconds (24-hour clock); CP/PP time is given in seconds, milliseconds. The user should disconnect his terminal from INTERCOM by turning it off, or by hanging up the data set receiver.

An automatic logout may occur under certain conditions such as INTERCOM being dropped while users are still logged in; a user leaving a terminal without logging out and another user attempting to log in; or a terminal is inadvertently disconnected, and the user is unable to re-establish communications and log back in within five minutes. In these cases, CP, PP, connect and log out time are not displayed.

COMMUNICATION BETWEEN TERMINALS

Messages can be sent to other remote terminals if the terminals are logged in and the user name is known. In addition, files may be transferred from one remote terminal to another with the INTERCOM command BATCH, if the terminals are logged in and the user id is known.

SITUATE COMMAND

A list of users currently logged in is obtained by entering:

```
SITUATE.
```

A list of user names along with the corresponding user id is displayed at the terminal; but neither the terminal location nor the password is included in the list. Any user names sharing the same password are listed first followed by other users' names if the password is restricted. Users with an unrestricted password are given only the names of users who share their password.

Example:

```
CØMMAND- SITUATE.
     USERS WITH SAME PASSWØRD
AB-USERA
     ØTHERS
CX-USERB
```

LOCK COMMAND

If the user does not want to receive messages from other terminals, he enters:

```
LOCK,ON.
```

All incoming messages from other users are locked out from this terminal. LOCK,ON does not prevent messages sent to all users, or messages sent by the central site operator, from being received at the terminal; it locks out only those messages sent specifically to the locked out user by another user.

A user who intends to execute a program with input and output data routed to his terminal, can use the LOCK command to prevent messages from being mixed with his data.

When he wants to receive messages again, he enters:

```
LOCK,OFF.
```

The terminal is unlocked automatically when the user logs out.

SEND COMMAND

To send a message, the user enters:

```
SEND.
```

The system responds:

```
TO WHOM-
```

The user may enter the name of the logged in user to whom he is sending the message. A user with an unrestricted password may send messages only to other users logged in with the same unrestricted password. A user with a restricted password may send messages to any other logged in user; he may also send a message to all logged in users by entering:

```
***
```

The message will be sent to all logged in users including those requesting no messages with the LOCK command.

In case of duplicate user names (this can occur only if the sender has a restricted password), INTERCOM selects the user to receive the message by first searching through users logged in with restricted passwords, and then through those logged in with unrestricted passwords.

A user name will not be found if the selected user is not logged in or has locked his terminal with the LOCK command. In addition, if a user with an unrestricted password attempts to send a message to a user logged in with a different password, the user name will not be found. The system returns the message: USER NAME xxxxxxxxx IS NOT ACCESSIBLE and returns the user to command mode.

If the user name is not locked out or if all logged in users were specified with ***, the system responds:

```
TYPE MESSAGE OR END
```

The user enters his message, which may be up to 72 characters long. The message is sent to the specified user, or to all logged in users if *** was specified as a user name. The system repeats its request:

```
TYPE MESSAGE OR END
```

The user can enter another message; or, if he has nothing further to send, he enters:

```
END
```

This word is recognized as a message terminator only if END are the first characters in the line followed by a carriage return (RETURN on TTY, SEND on CRT). Up to seven spaces may be inserted between END and the carriage return. This allows the word END to be embedded in a message without terminating it prematurely.

The message appears on the display screen or teletypewriter printout of the receiving terminal. The message is preceded by:

```
FROM xxxxxxxxxx -
```

which informs the user at the receiving terminal as to who is sending the message.

Examples:

```
C0MMAND- SITUATE.
     USERS WITH SAME PASSW0RD
AB-USERA
     0THERS
CX-USERB            BZ-USERC

C0MMAND- SEND.
     T0 WH0M- ***
TYPE MESSAGE 0R END-
IS ANY0NE RUNNING A BASIC PR0GRAM?
TYPE MESSAGE 0R END-
END

C0MMAND- SEND.
     T0 WH0M- USERB
TYPE MESSAGE 0R END-
CAN I RUN S0ME SAMPLE PR0GRAMS F0R Y0U?
TYPE MESSAGE 0R END-
END
```

```
CØMMAND- SEND.
     TØ WHØM- USERD
USER NAME      USERD  IS NØT ACCESSIBLE

CØMMAND-
```

COMMUNICATION WITH CENTRAL SITE

Messages may be sent from the terminal to the central site operator, and messages from the central site may be received at the terminal. Messages cannot be sent when the user is in SETUP; he should leave SETUP to enter the MESSAGE command.

MESSAGES TO CENTRAL SITE

To send a message to the central site, the terminal user enters:

M,mmm...m or **MESSAGE,mmm...m**

mmm...m is the message which may consist of any characters in the INTERCOM character set including embedded blanks. It must not exceed 58 characters. A longer message will be truncated to this maximum. The message is entered in the B display and system dayfile at the central site.

If a message is sent to the central site when another message is already being displayed there, the terminal user receives the message:

CONSOLE BUSY - TRY AGAIN LATER

After a message is displayed at the central site console, INTERCOM returns the terminal user to command mode, and the user may enter any legal command.

MESSAGES FROM CENTRAL SITE

Messages from the central site have priority over all other displays. They are displayed regardless of other activity at the remote terminal. After a central site message has been displayed, the user can resume whatever activity was interrupted by the message.

A central site message is displayed at the user's terminal as:

xx,mmm...m

where xx is the user id and mmm...m is the message.

Examples:

A user assigned the identification code AB requests information from the central site:

```
CØMMAND- M,HØW MUCH MØRE TIME CAN I HAVE?
```

The central site operator can send a reply which appears at the user's terminal:

```
AB,YØU HAVE 30 MINUTES.
```

The user then may continue his previous activity; or if execution of a command was interrupted, the system will continue execution.

FILE CONTROL

Files are used by the INTERCOM user to store and maintain information and programs, contain jobs or information for processing at the central site, hold output generated by SCOPE commands and compilers, and route information between remote terminals. Commands described in this section allow the user to obtain a list of his private and remote files, examine output files, and specify files for terminal interaction.

The BATCH command, described in a later section, may be used if files are to be renamed, submitted to the central site batch queues, recovered from the batch output queue, or transferred between private and common status.

FILES COMMAND

Entry of this command returns a list of the user's private files and any attached permanent files. The names of remote executing jobs and remote input and output files to be recovered at the user's terminal are listed also. The user enters:

```
FILES.
```

The system responds:

```
--PRIVATE FILES--
```

(list of private file names)

This list names all private files and attached permanent files belonging to the terminal user. It includes any files created by INTERCOM such as INPUT, OUTUT, SETFILE, or LGO as well as files created or attached by the user. Attached permanent files are listed by their private file names preceded by *. Files currently connected to the terminal are preceded by $.

The files INPUT and OUTPUT are used by most utilities for terminal interaction and are usually connected to the terminal. The user should avoid using these files for anything other than terminal interaction, but if necessary, they may be disconnected by the user.

Terminating an INTERCOM utility with the interrupt command may cause extraneous files to appear in the list of user's files. These are scratch files that the utility could not release because of the interrupt. Scratch files created by the ALGOL or COBOL compilers or the COMPASS assembler may also appear. All such scratch files may be released by the user.

```
--REMOTE EXECUTING JOBS--
```

(list of job names)

This list names all of the user's jobs currently being executed at the central site with output to be recovered at the terminal.

```
--REMOTE INPUT FILES--
```

(list of input file names)

This list names all remote input files associated with the user which are to be recovered at the terminal.

```
--REMOTE OUTPUT FILES--
```

(list of output files names)

This list names all remote output files associated with the user which are to be recovered at the terminal.

Example:

```
C0MMAND- FILES.
--PRIVATE FILES--
 $INPUT          $0UTPUT          *FILA            $*FILB
  LG0             F0RTX           $TEST1
--REM0TE EXECUTING J0BS--
VICS01A
--REM0TE 0UTPUT FILES--
TEST06C          5F1236R
C0MMAND-
```

RETURN/UNLOAD COMMANDS

Unwanted private files can be released with the SCOPE control card commands RETURN or UNLOAD. These commands also may be used to return attached permanent files to permanent mass storage, thereby deleting them from the user's list of private files.

The user enters:

```
RETURN,filename-1,filename-2,...,filename-n.
```

or

```
UNLOAD,filename-1,filename-2,...,filename-n.
```

The file storage space is returned to the system (unless the files are permanent) and the files are no longer available to the user.

Example:

To release the private file, BLINE:

```
C0MMAND- RETURN,BLINE.
C0MMAND-
```

STORE COMMAND

A user's private file residing on a permanent file device may be made permanent by entering:

```
STORE,filename.
```

The file name specified is the name of the private file that is to be made permanent.

The system responds:

```
ID=
```

The user then enters a permanent file user identification (1-9 alphanumeric characters) for the file.

The permanent file name will be a concatenation of the private file name and the user identification entered for the file, in that order. The retention period for the file is the default retention period defined by the installation.

To ensure that a private file resides on a permanent file device, the SCOPE control card command REQUEST,filename,*PF. should be entered before the private file is created (SAVE command, section 4). Otherwise, entry of the STORE command may return a message indicating an illegal condition. In this case, the private file may be copied to a permanent file device and then stored as a permanent file with the following control card command sequence (the private file exists as FILEA):

```
C0MMAND- REWIND,FILEA.
C0MMAND- REQUEST,FILEB,*PF.
C0MMAND- C0PY,FILEA,FILEB.
C0MMAND- ST0RE,FILEB.
ID=M00RE
C0MMAND-
```

ABBREVIATED ENTRY

```
STORE,filename,ident.
```

ident is the permanent file user identification. With this form, the system does not issue the request, ID.

Examples:

```
C0MMAND- ST0RE,L0C.
ID=VSS
C0MMAND-
```

```
C0MMAND- ST0RE,L0C,VSS.
C0MMAND-
```

After the STORE command is entered and processed, the file remains in the list of the user's private files as an attached permanent file with the name, LOC. If this file does not need to be retained as a private file, the user may detach only the private file with the SCOPE control card command, RETURN.

PRIVACY PROCEDURE

The installation may define a privacy procedure for storing permanent files and may further require that the user enter a privacy key when storing such a file:

```
STORE,filename,ident,pp.
```

pp is the privacy key (1-9 alphanumeric characters) for the file. If the privacy parameter is required but not specified in a STORE command, the system responds:

```
pp=
```

The user may then enter a privacy key.

Examples:

```
C0MMAND- ST0RE,L0C.
ID=VSS
PP=A1970
C0MMAND-
```

```
C0MMAND- ST0RE,L0C,VSS,A1970.
C0MMAND-
```

No permanent file passwords are set by the STORE command. If permanent file passwords or other permanent file parameters are desired, the SCOPE control card command, CATALOG is required.

The INTERCOM commands STORE and FETCH are intended primarily for private file manipulation. Private files made permanent with the STORE command cannot be accessed with multi-read permission by the FETCH command. However, the SCOPE control card command, CATALOG, can be used in a form which allows access, with multi-read permission, by the FETCH command, permitting several users simultaneous read-only access to the file. The following control card command:

```
CATALOG,FILEA,TESTJONES,MD=A,EX=A,CN=A,ID=JONES.
```

catalogs the private file FILEA such that it can be subsequently accessed with multi-read permission by the FETCH command in the form:

```
FETCH,TEST,JONES.
```

FETCH COMMAND

Permanent files saved with the STORE command may be accessed when the user enters:

```
FETCH,filename.
```

The file name must be that of an existing permanent file but not a private file. The system responds:

```
ID=
```

The user must enter the user identification with which the permanent file was stored.

Further, if privacy procedures are defined, the installation may require that a privacy key be specified to access a permanent file. If so, the system requests:

```
PP=
```

The user must enter the privacy key with which the permanent file was stored.

ABBREVIATED ENTRY

If a privacy key is required, the user may enter:

```
FETCH,filename,ident,pp.
```

Otherwise, he may enter:

```
FETCH,filename,ident.
```

Examples:

```
C0MMAND- FETCH,L0C.
ID=VSS
C0MMAND-
```

```
C0MMAND- FETCH,L0C,VSS.
C0MMAND-
```

A permanent file must be attached before it can be accessed by other INTERCOM commands or SCOPE control cards. An attached permanent file should not be modified directly unless a rewrite-in-place operation is performed; rather, any modifications to the attached permanent file should be made on a temporary or scratch file. When modifications are complete, the original permanent file can be eliminated with the DISCARD command and the temporary file saved as a permanent file with the STORE command. Permanent files saved with the SCOPE control card, CATALOG, should be accessed with the ATTACH card. However, if the file is cataloged in a form compatible with the STORE command, it may be accessed with the FETCH command.

DISCARD COMMAND

This command is used to delete a permanent file that was saved with the STORE command. If the file to be deleted is first attached with the FETCH command, the user may enter:

```
DISCARD,filename.
```

The file name is purged from the permanent file catalog and is no longer available to the user as either a private or a permanent file.

A file may be deleted even though it has not been attached. All parameters including file name and user identification must be specified in the DISCARD command; a privacy key may be required at installation option. Required parameters must be declared in the command entry; they will not be requested by the system.

Examples:

```
C0MMAND- FETCH,L0C,VSS,A1970.
C0MMAND- DISCARD,L0C.
C0MMAND-
```

```
C0MMAND- DISCARD,L0C,VSS,A1970.
C0MMAND-
```

CONVERT COMMAND

Files created in SETUP differ in format from files created in EDITOR. With the CONVERT command the user can copy a SETUP file to a new file in EDITOR format. The original SETUP line numbers are retained in the new file. The user enters:

```
CONVERT,filename-1,filename-2.
```

The SETUP file named by filename-1 is copied in converted format to the EDITOR file named by filename-2. The parameters must appear in the order shown above and filename-2 may not be the same name as filename-1 or any other private file.

The CONVERT command may also be used to copy a file containing a BASIC program which was not created in SETUP or EDITOR to a new file which may be accessed by EDITOR. The user enters:

```
CONVERT,filename-1,filename-2,BASIC.
```

The keyword BASIC may be abbreviated to the initial character as indicated by the underscore.

The BASIC statement numbers found at the beginning of each line are copied at the end of the line so that they may be referenced as line numbers by EDITOR. The BASIC statement numbers must be in ascending order within the original file.

Examples:

```
COMMAND- CONVERT,SFILE,EFILE.
COMMAND-
```

A SETUP file SFILE, is converted to an EDITOR file EFILE.

```
COMMAND- CONVERT,BFILE,EDFILE,BASIC.
COMMAND-
```

A file BFILE, containing a BASIC program created independently of SETUP or EDITOR, is converted to an EDITOR file EDFILE.

The CONVERT command may not be used to prepare a BASIC program for access by the SETUP utility (see TRANS directive, section 5).

PAGE COMMAND

With the PAGE command, the user may examine a private file displayed page by page at his terminal. In addition, he may: Page forward or backward a specified number of pages or lines. Tab right or left a specified number of columns to examine lines longer than the terminal line length. Search forward or backward for a specified character string and display the page containing the first occurrence of that string; the search string may be restricted to a header line, subhead line, or a column range within a line.

The results of any of the above operations may be transferred to a print file, along with user comments, for subsequent printing at either the central or remote site. The PAGE command may be used to access standard SCOPE coded files. A page on a display terminal may be 11 lines by 80 characters or 18 lines by 50 characters, depending on the screen size. If a terminal's screen differs from the installation standard, the user should enter the SCREEN command. A page on a Teletype terminal is 10 lines by 72 characters.

The maximum line length for a paged file is 150 characters; for the output print file, the maximum line length is 136 characters.

Paging of a file is requested by typing:

```
PAGE,filename-1,filename-2.
```

The first parameter names the file to be paged; if omitted, OUTPUT is assumed. The second parameter names the print file; if omitted, PRINT is assumed. The two parameters may not name the same file. The only files that may not be paged or printed are ZZZZZOU, ZZZZZIN, or ZZZZZRN which are used for scratch files or terminal input/output.

After the above command is entered, PAGE responds:

```
READY..
```

If the file to be paged is not to be repositioned to the beginning of information before paging begins, the user must enter the character N immediately following the READY.. response, before any other command.

The user can enter the character A to obtain a directory of PAGE display commands (figure 3-1). This directory describes paging and tabulation commands and how to leave the PAGE utility.

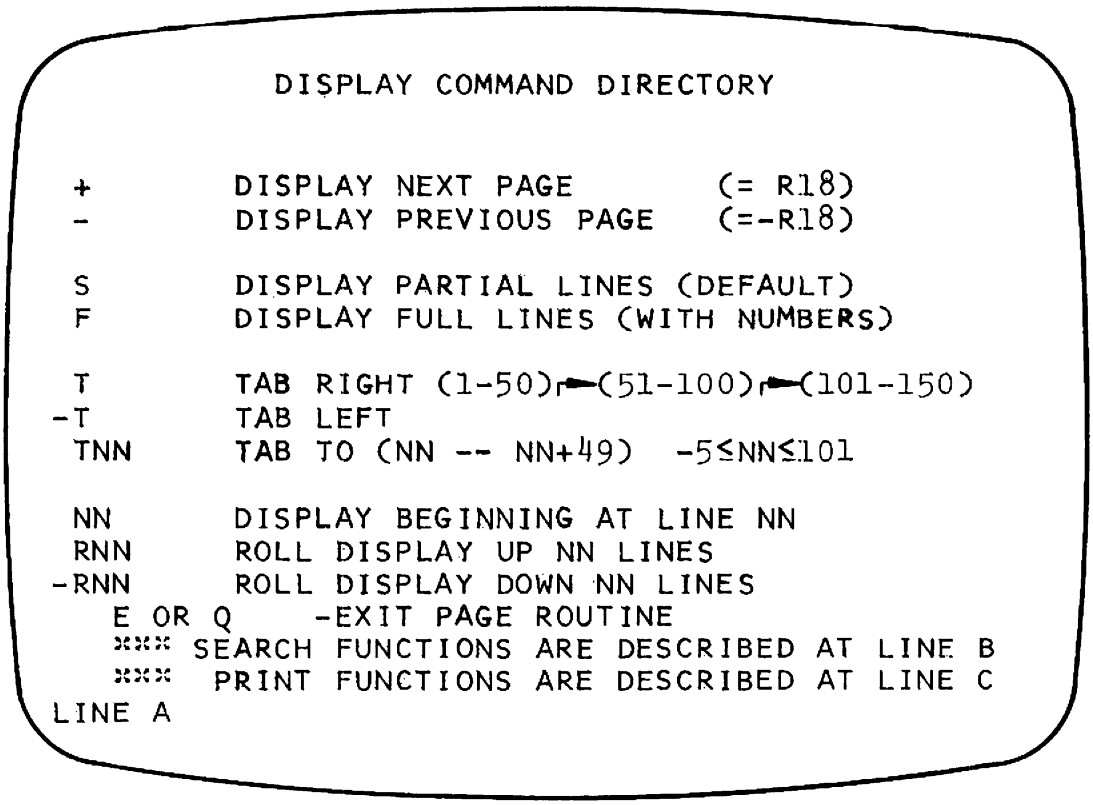


Figure 3-1. PAGE Display Command Directory (20 x 50 screen)

The PAGE utility appends a line number in character positions -5 through 0, to each line in the file. The user may enter -T5 to examine them. Partial lines are listed without PAGE line numbers and are only as long as the terminal line width. Full lines are listed with PAGE line numbers and contain the entire line. The display mode commands, F and S, and the tabulation commands, T, -T and TNN, remain in effect until specifically changed by a subsequent command.

The + and - commands are equivalent to an R (roll) command, with the size of the display page as the number of lines to roll. When full lines are being displayed, some output may not be seen if either + or - is used. In this case, the user should enter the exact PAGE line number to be displayed as the first line of the page.

When specified with the T or R command, the minus sign may be entered either before or after the alphabetic character; -T5 and T-5 are equivalent.

Figure 3-1 indicates the action of the + and - commands for a CRT terminal with a 20 x 50 screen, where + is equivalent to R18, and - is equivalent to -R18. For a CRT terminal with a 13 x 80 screen, + is equivalent to R11; - is equivalent to -R11. For TTY terminals, + is equivalent to R10; - is equivalent to -R10.

Similarly, the action indicated in Figure 3-1 for the T and -T commands is for a CRT 20 x 50 screen. For this screen, tabbing is performed from the current tab position either right or left to the next tab stop; stops are defined at character positions -5, 1, 51, and 101. The 13 x 80 screen tab stops are defined at positions -5, 1, and 71. The TTY terminal tab stops are defined at positions -5, 1, 73, and 79.

The character B can be entered to obtain a directory of search commands (figure 3-2). This directory describes character string searching capability.

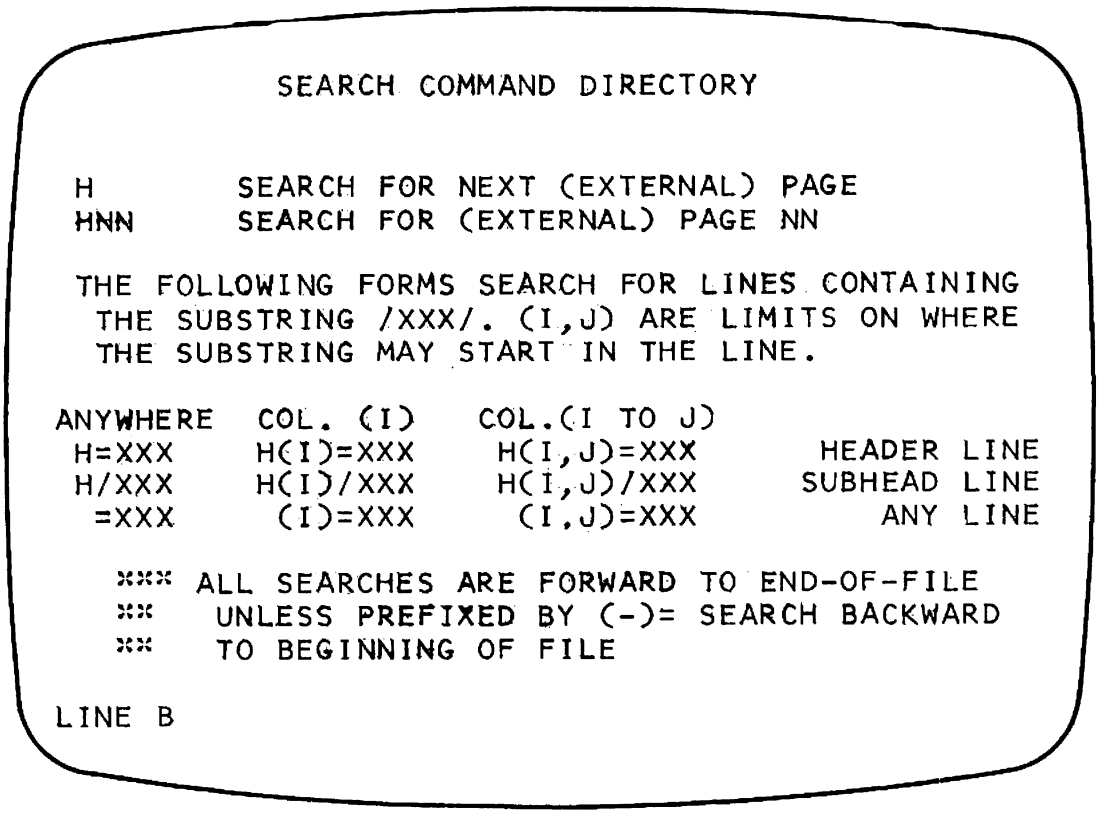


Figure 3-2. PAGE Search Command Directory (20 x 50 screen)

A search for an external page number examines only header lines from character position 122 through the end of the line. A header line contains the carriage control character 1 in column 1 to cause a page eject. A subhead line immediately follows a header line. This feature permits a search for page numbers or headings produced on the source listing by the compilers and assemblers available through INTERCOM.

A search for a character string may be made forward or backward through the file. The first line encountered which satisfies the search condition is displayed as the first line of a page of information. If search conditions are not satisfied, a message indicates that the beginning or end of the file has been reached.

Forward searches begin at the current line plus 1, backward searches at the current line minus 1. To search from the beginning of a file currently positioned in the middle or at the end , enter:

```
0,command
```

to reset the current line number to zero, performing a logical rewind.

A search substring may be up to 150 characters in length.

A directory of print utility commands (figure 3-3) can be obtained by entering the character C. These commands may be used to construct a print file.

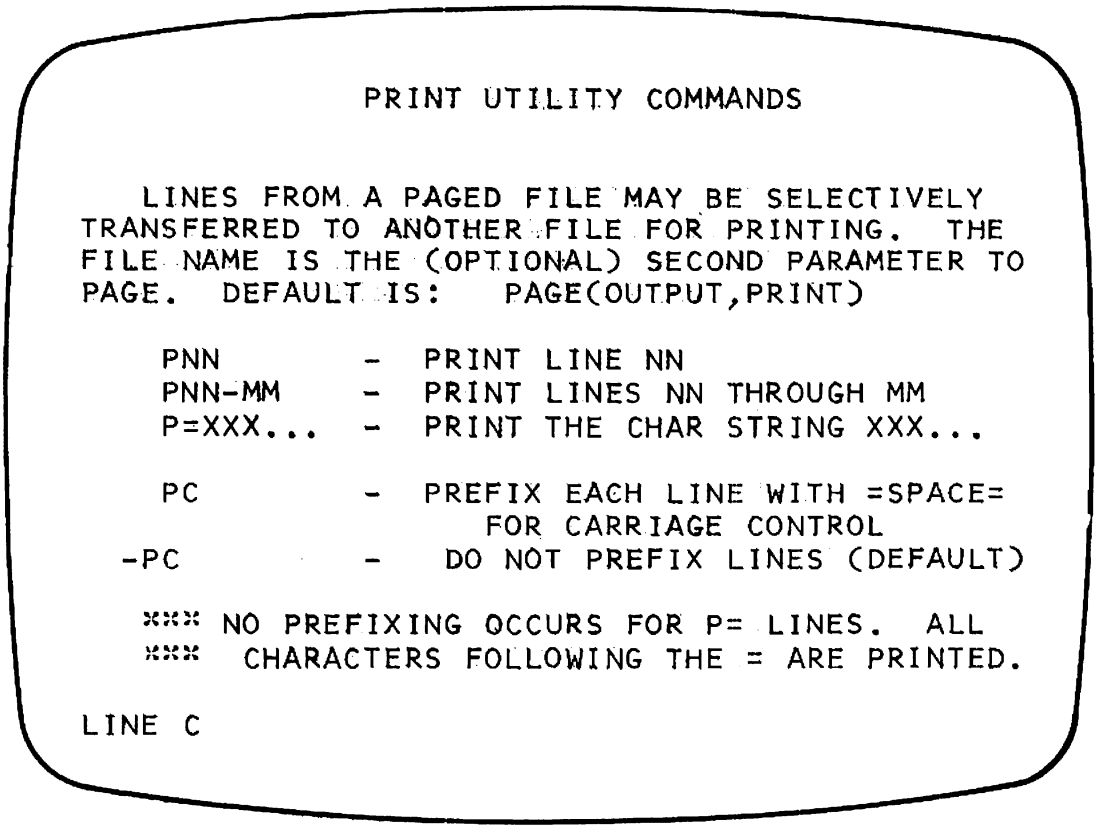


Figure 3-3. PAGE Print Utility Commands (20 x 50 screen)

Lines are transferred to the print file exactly as found on the source file (with possibly a leading blank). The PAGE line number is not included on the print file. At the end of a session with PAGE, if any print output has been generated, the print file remains positioned after all print lines but before the end-of-file. This allows more information to be placed on the same print file in subsequent PAGE sessions without having to reposition the file. No positioning is performed before writing on a print file; it is the user's responsibility to properly position any declared print file which was not previously a print file or which has been repositioned since the last PAGE session.

The prefixing commands, PC and -PC, remain in effect until changed by a subsequent prefixing command. PC should be specified to ensure single spacing of output. Since prefixing is not performed for print lines entered with the P= command, the user can control spacing on the prin: file and enter headings. Up to 136 characters may be entered with the P= command.

ENTERING PAGE COMMANDS

The last line of each displayed page is in the form:

LINE nnnnn:	**TAB = ttt**	(in partial-line mode)
	or	
LINE nnnnn:	**FULL**	(in full-line mode)

nnnnn is the PAGE line number of the first line of the displayed page and ttt is the beginning tabular character position of each line on that page. On a CRT terminal, the previous command will remain displayed as the last line on the screen.

Display, search, and print utility commands may be entered after the READY.. message or on the line immediately following the last line of displayed output (overprinting the previous command on a CRT terminal). Any number of commands may be entered in one line, separated by commas (the command line may not exceed 150 characters). Such commands will be executed consecutively, with only the final resultant page displayed. With this feature, positioning, tabbing, and searching all can be specified in one entry. Only one search or one P= command may be included, however, and it must be the last command in the sequence. Blanks in the command line, except those following the P command, are ignored.

A single command or sequence of commands is scanned and executed from the left. If an error at any point renders it impossible for PAGE to recognize the remainder of a command string and scanning terminates, the current page is displayed, and the last displayed line will contain the notation, U/xxx..., following the line number and tab information; the characters xxx... indicate the unexecuted portion of the command.

Examples:

The following command sequence spaces forward three pages to begin a search in the reverse direction for the character string XYZ (starting in any column position). Assume that this character string first occurs in line number 62 and that partial lines are being displayed:

```
+,-,+,+,+,T30,-=XYZ
```

The page will be displayed beginning at line 62 with the left margin at column position 30. The last line of the displayed page will appear:

```
LINE     62:  TAB = 30
```

A user at a Teletype terminal has recovered a file containing his job named GNR from the output queue. The file now exists as a private file, GNR0081). To examine the job's dayfile, the user enters the PAGE command and then enters a display command (=GNR,CM60000) to find the first 11 characters of his job card. The page displayed will consist of ten lines of the dayfile beginning at the line containing the job card:

```
COMMAND- PAGE,GNR0081.
READY..-T5,=GNR,CM60000

 1572: 08.11.42.GNR,CM60000,T100,P17.
 1573: 08.11.42.N613,GNR0BERTSON.                    000100
 1574: 08.11.42.
 1575: 08.11.42.ATTACH,0LDPL,INTERC0M30LDPL.
 1576: 08.11.43.CYCLE **, INTERC0M30LDPL
 1577: 08.11.43.PFN F0UND IN SD  002
 1578: 08.11.43.CYCLE 01, INTERC0M30LDPL
 1579: 08.11.43.FILE HAS BEEN ATTACHED
 1580: 08.11.43.UPDATE,Q.
 1581: 08.11.43.                                    000120

LINE  1572:  TAB = -5
```

To display more of his dayfile, the user enters a display command +:

```
+

 1582: 08.11.44.READING INPUT
 1583: 08.11.50.UPDATE COMPLETE
 1584: 08.11.51.COMPASS,I=COMPILE,S=SCPTEXT.
 1585: 08.11.51.                                          000130
 1586:
 1587: 08.12.03. MINIMUM FIELD LENGTH NEEDED = 050100
 1588: 08.12.03. ASSEMBLY COMPLETE.
 1589: 08.12.03.CP     005.406 SEC.
 1590: 08.12.03.PP     006.211 SEC.
 1591: 08.12.03.I0     001.520 SEC.

LINE   1582:   TAB = -5
```

He may now enter any of the display or search commands to further examine his file, or any print utility commands to construct a print file.

CONNECT/DISCONT COMMANDS

The user can request that specific files be designated for terminal interaction by entering the command:

```
CONNECT,filename-1,filename-2,...,filename-n.
```

Input and output will be routed to and from the terminal when the named files are subsequently written or read. The file names may be INPUT and OUTPUT, as well as any other files, including attached permanent files.

Each time a connected input file is referenced in the source program, the system waits for input from the terminal. Each time a connected output file is referenced in the source program, the output is printed or displayed at the terminal. It is not saved.

When input is expected from the user, the system waits for the user to enter it from his keyboard. For BASIC programs, the INPUT statement displays a question mark at the terminal when user input is expected. For COBOL programs, the ACCEPT FROM statement displays ENTER COBOL INPUT at the terminal. It is helpful if other programs contain a signal to the user that input is expected from the terminal. For example, the PRINT statement can be used in conjunction with a FORTRAN READ statement.

The CONNECT command need not be entered when programs are executed with the EDITOR RUN command (section 4) or with SETUP directives (section 5). In this case, the files INPUT and OUTPUT are connected automatically. If the user wants to connect any other input/output files to his terminal, he may use the CONNECT command.

A file connected to the terminal with the CONNECT command, may be disconnected with:

```
DISCONT,filename-1,filename-2,...,filename-n.
```

The specified files no longer will be connected to the terminal; they will be assigned to allocatable mass storage.

Examples:

The following FORTRAN program has been saved as a private file, AAA:

```
 90    PRØGRAM FØRT(TAPE1,TAPE2)
100 1  FØRMAT(* PLEASE ENTER A 3-DIGIT NUMBER*)
110 2  FØRMAT(I3)
115 3  FØRMAT(I4)
120 4  FØRMAT(2,1)
130    READ(1,2) I
140    IF(I.EQ.999)STØP
150    WRITE(2,3) I
160    GØ TØ 4
170    END
```

To execute with interaction, the user may enter the following commands at the terminal:

```
CØMMAND- REWIND(AAA)
CØMMAND- DISCØNT,ØUTPUT.
CØMMAND- RUN(S,,,AAA)
CØMMAND- CØNNECT,TAPE1,TAPE2.
CØMMAND- LGØ.
```

The program displays the message: PLEASE ENTER A 3-DIGIT NUMBER and waits for the user to enter the number from the terminal keyboard. When the user enters a number, the program displays the number and repeats this procedure until the user enters 999 which stops execution. The file OUTPUT is disconnected prior to compilation to prevent the source listing from being sent to the terminal by the RUN compiler.

The user may then enter:

```
CØMMAND- DISCØNT,TAPE1,TAPE2.
```

Interaction with the terminal will no longer be possible, as the input and output files TAPE1 and TAPE2 are no longer connected to the terminal.

The following BASIC program BASAMPL is stored as a private file:

```
10  PRINT "TYPE A NUMBER ";
20  INPUT X
25  IF X=0 THEN 80
30  F=1
40  FØR I=1 TØ X
50  F=F*I
55  NEXT I
60  PRINT "FACTØRIAL"X;"IS"F
70  GØ TØ 10
80  END
```

The user connects the output file BASOUT to the terminal:

```
COMMAND- CONNECT,BASOUT.
```

He then enters the following control card commands to execute the program:

```
COMMAND- REWIND(BASAMPL)
COMMAND- BASIC(I=BASAMPL,K=BASOUT)
```

The program requests a number from the terminal and stops when 0 is entered:

```
TYPE A NUMBER ?6
FACTORIAL      6  IS       720
TYPE A NUMBER ?0
COMMAND- DISCONT,BASOUT.
```

The user is returned to command mode and disconnects the file BASOUT, used for diagnostics and execution output. It is not necessary to disconnect the file OUTPUT in this case because the BASIC compiler does not write the source listing on OUTPUT.

PASSWORD FILE

A permanent user password file is created and maintained by INTERCOM. This file contains information on each password/user name as defined by the installation. This information includes the user's maximum field length and time limit, and the number of files allotted to him. If a user's files exceed this quota, he is denied access to any file creation commands until he returns or unloads his private files below the maximum.

The password file can be called or altered only with a data deck entered as a batch job at the central site card reader.

ASSETS COMMAND

The user's terminal status can be requested by entering:

```
ASSETS.
```

The system replies with a header message: ASSETS OF xx AT clock- time where xx is the user identification code, and clock-time is the current time of day. The header is followed by the number of files used, maximum number of files allotted, maximum field length, time limit, total elapsed central processor and peripheral processor time, and the EFL and ETL if entered. In addition, the user's sense switch, SAVEFL, REDUCE, LOCK, and MAP settings are given if they differ from installation default values.

Example:

```
C0MMAND- ASSETS.

ASSETS 0F BE AT   11.12.02.
FILE QU0TA          20
FILES IN USE         7
MAX FL           077700
†EFL             055000
TIME LIMIT         0500
†ETL               0100
†L0CK                0N
†SAVEFL              0N
†REDUCE             0FF
†MAP               PART
†SWITCHES 0N        136
CP TIME            .174
PP TIME          16.379
C0MMAND-
```

ETL COMMAND

The terminal user can specify a time limit for execution of each succeeding command. If not specified, INTERCOM assigns a default time limit (10 CP seconds in standard system). The specified time limit may not exceed the total time limit assigned to this user in the password file, nor may it exceed the total CP time remaining for this user for the session.

The user enters the time limit, xxxx, in octal seconds:

```
ETL,xxxx.
```

If insufficient time is specified for command execution, or if the total session time remaining is less than that required for command execution, INTERCOM returns the message: TIME LIMIT. The user may still enter certain commands to manipulate files and to exit from INTERCOM after session time is exhausted.

Most commands entered after execution of the ETL command are executed with the time limit specified by xxxx until the user resets the limit by entering the following command which restores the default time limit.

```
ETL,0.   or  ETL.
```

† These lines will be listed only if the values or settings are changed, by a user, from installation default conditions.

Example:

The user enters a program for compilation and execution with the EDITOR RUN command, but the default time limit is insufficient. He may increase his time limit with ETL and re-enter his program. The specified time limit is assigned to compilation and subsequently to execution:

```
••RUN,FTN
JØB CØMPILING
TIME LIMIT
••ETL,400.
••RUN,FTN
JØB CØMPILING
        .
        .
        .
••ETL,0.
••
```

EFL COMMAND

With this command, the terminal user may specify a field length to be allocated for execution of each succeeding program. If not specified, INTERCOM assigns a default field length (30000 octal in standard system).

The user specifies field length, yyyyyy, as the number of words in octal:

```
EFL,yyyyyy.
```

The specified field length may not be greater than the maximum field length assigned to the user in his password file. User programs executed subsequently are assigned the field length specified with EFL until the user resets the field length by entering the following command which restores the default field length.

```
EFL,0.   or   EFL.
```

Only INTERCOM or SCOPE commands whose field length requirements vary, such as compiler control cards, are affected by the EFL command.

Example:

Suppose the user requires a field length of 45,000 to load his program. He may specify this field length with EFL, and then enter his program for compilation and execution:

```
••EFL,45000.
••RUN,RUN
JØB CØMPILING
        .
        .
        .
••EFL,0.
••
```

SUBMITTING A JOB

If the user has a private file in the format of a SCOPE batch job, he may submit it to the SCOPE batch input queue with the BATCH command. This command offers other dispositions to allow the user to obtain processing at the central site. These include punching of card decks, printing of private files, and recovery of remote batch output files. In addition, BATCH provides the capability of accessing common files and renaming private files.

If a display terminal is equipped with a card reader and line printer, the user may transmit jobs directly to the SCOPE input batch queue and receive resulting output through these terminal input/output devices by using the remote batch commands (section 7).

BATCH COMMAND

The BATCH command is used to direct the disposition of a file which the terminal user has previously created and saved. Except for the PRIVATE and LOCAL dispositions, the file must be a private file accessible to the terminal user. An attached permanent file may not be specified with any disposition of the BATCH command except RENAME. No copy of a file is made, and except for LOCAL, RENAME, and PRIVATE dispositions, the file will not exist as a private file following execution of BATCH.

Files are processed one at a time with the BATCH command by typing:

```
BATCH.
```

The system responds:

```
TYPE FILE NAME-
```

The user then enters the name of the file to be processed by the central computer. Only one name may be specified. If the file is validated, the system responds:

```
TYPE DISPOSITION-
```

A number of dispositions are permissible. They describe how the user wants the named file to be handled. For instance, it can be placed in the input queue, prepared for card punching or line printing, renamed, or made a common file or a private file. A user file is processed according to the requested disposition.

For PRINT, PRINT,xxxx, PUNCH, and PUNCHB dispositions, the system responds:

```
TYPE FILE ID-
```

The user enters 1-4 alphanumeric characters which are used to create a header line for identifying the output. The header line appears in the form:

```
Ixxxxss
```

where xxxx is the user supplied file identification and ss is a sequence number assigned by SCOPE.

If the file satisfies the requirements for the specified disposition, the system requests another file for processing:

```
TYPE FILE NAME
```

If the user has another file to process, he types the file name, and upon request, the disposition for that file. If no more files are ready for batch processing, he exits from the BATCH command by typing:

```
END
```

ABBREVIATED ENTRY

```
BATCH,filename,disposition,file-id.
```

or, for the RENAME disposition:

```
BATCH,filename,RENAME,filename.
```

The file-id need be supplied only if required by the disposition. Not all parameters need be specified in the initial entry, but those specified must appear in the order indicated. Omitted parameters will be requested by the BATCH utility. If one or more parameters are specified, BATCH will terminate after processing the specified file; it will not request a new file name.

Care must be taken when entering the following form of the BATCH command:

```
BATCH,filename,PRINT,xxxx.
```

In this case, if xxxx is the user id of a logged in user or the word HERE, it is interpreted as a user id. Otherwise, it is interpreted as file-id.

BATCH COMMAND DISPOSITIONS

Disposition	Explanation
INPUT	File is placed in SCOPE batch input queue at central site to be run as a batch job. The file must already contain the required control cards as its first logical record. The JOB card determines the field length and time allotment for the job. The OUTPUT file is printed at the central site.
INPUT,xxxx	File is placed in the input queue exactly like INPUT, but the output from execution of the job is directed to the SCOPE remote output queue. xxxx must be the user id of a logged in user or the word HERE to indicate the user's own terminal. The user id will be associated with the file in the output queue. The user xxxx may have the file printed on his remote line printer, or he may make it a local (private) file by using the LOCAL disposition. To receive output on a remote printer, the line printer must be on; and he must have entered the command GO (section 7).
PRINT	File is placed in SCOPE output queue at the central site for subsequent printing on high speed line printer. A carriage control character is required as the first character of every line to be printed. Carriage control characters are supplied for output files produced by FORTRAN (RUN and Extended), BASIC and COBOL compilers. For files other than standard output files, the user must supply carriage control characters (see appendix D). A blank character can be inserted with the SCOPE control card COPYSBF.

Disposition	Explanation
PRINT,xxxx	File is placed in the remote output queue and subsequent action is selected by the user xxxx. xxxx must be the user id of a logged in user or the word HERE to indicate the user's own terminal. The user id will be associated with the file in the output queue. The user xxxx may have the file printed on his remote line printer, or he may make it a local (private) file by using the LOCAL disposition. To receive output on a remote printer, the line printer must be on and the user must have entered the command GO (seciton 7).
PUNCH	File is released for subsequent card punching at central site. Cards are punched in Hollerith code.
PUNCHB	File is released for subsequent card punching at central site. Cards are punched in binary.
PRIVATE	Applicable to common files only. File is removed from list of common files and made available to the user. He may then read, alter, or otherwise modify the file. The file can be returned to common by entering BATCH again and specifying COMMON. If not reinstated as a common file, it remains in the user's private file list and is not available to other users; it will be lost at LOGOUT if not reinstated as a common file with the COMMON disposition.
COMMON	User's private file is made a common file accessible to all other users. User may no longer modify the file, unless he has previously copied it, until file is again made private with PRIVATE disposition.
RENAME	File is to be given new name. The system responds: ENTER NEW NAME- and the user enters the name. A permanent file may not be renamed; but the private file name associated with the permanent file while it is attached can be renamed.
LOCAL	The user's file in the remote output queue is made a local (private) file. The file may have been placed in the output queue by the INPUT,xxxx or PRINT,xxxx dispositions described in this section, the READ command described in section 7, or by a DISPOSE function issued either from a terminal or at the central site. This disposition allows the user with no remote line printer (either a teletypewriter or a display terminal) to submit a batch job and recover the output file for subsequent examination with the INTERCOM command, PAGE (section 3) or with EDITOR commands (section 4).

Examples:

```
C0MMAND- BATCH.
TYPE FILE NAME- MYFILE
TYPE DISP0SITI0N- INPUT
TYPE FILE ID- ABCD
TYPE FILE NAME- END
C0MMAND-
```

User enters MYFILE in input queue at central site to run as a batch job and returns to command mode.

```
C0MMAND- BATCH.
TYPE FILE NAME- BLIST
TYPE DISP0SITI0N- PRIVATE
TYPE FILE NAME- END
C0MMAND- C0PY(BLIST,CLIST)
C0MMAND- BATCH,BLIST.
TYPE DISP0SITI0N- C0MM0N
C0MMAND-
```

User makes common file BLIST his own private file, copies it onto CLIST and then returns BLIST as common file.

```
C0MMAND- BATCH.
TYPE FILE NAME- AFILE
TYPE DISP0SITI0N- PUNCHB
TYPE FILE ID- 158E
TYPE FILE NAME- BFILE
TYPE DISP0SITI0N- INPUT,AZ
TYPE FILE ID- WXYZ
TYPE FILE NAME- CFILE
TYPE DISP0SITI0N- PRINT,HERE
TYPE FILE ID- 158E
TYPE FILE NAME- END
C0MMAND- G0
```

User punches AFILE in binary, then inserts BFILE in input queue with execution output directed to user with user id AZ logged in at another terminal. CFILE is printed at user's own terminal. GO is entered to begin printing CFILE.

The user may construct a file with EDITOR for later entry by BATCH:

```
C0MMAND- EDIT0R.                          user calls EDITOR
..F0RMAT,F0RTRAN                          EDITOR command mode established: user
..CREATE,100,10                             calls FORTRAN format specification;
  100=;PR0GRAM A(0UTPUT)                    asks to create a new file beginning in line
  110=;PRINT 1                              100 incremented by 10; and enters a
  120=1;F0RMAT(* TEST A*)                   FORTRAN program line by line.
  130=;END
  140==                                   user exits increment mode.
..ADD,10 10                               user asks to insert lines at beginning of file:
    10=JMARTIN,T200,CM55000.                SCOPE control cards are entered fol-
    20=RUN.                                 lowed by an end-of-record character
    30=*E0R                                 string.
    40==
..SAVE,CCFIL,N0SEQ                        user saves file as CCFIL with no sequencing;
..BYE                                       and exits EDITOR.
C0MMAND- BATCH,CCFIL,INPUT,JMAR.          the BATCH command is entered in INTER-
C0MMAND-                                    COM command mode to run the job just
                                            created at the central site.
```

For CREATE and ADD commands, line numbers are generated automatically in increment mode. These line numbers control the sequence of a file created in EDITOR, thus, the first record contains the control cards followed by the FORTRAN program. After the job is run, the output (including source listing, load map, the literal TEST A, and the dayfile) is printed at the central site.

Q COMMAND

This command allows the user to monitor the progress of his batch jobs by examination of SCOPE batch processing queues. The user enters:

```
Q,p.
```

where p may be one of the following:

omitted	A count of the number of jobs in the input and output queues is displayed.
I	A list of jobs in the input queue is displayed (includes job name, priority, field length, time limit, and for remote jobs, the user identification).
E	A list of all batch jobs currently executing is displayed (includes job name, priority, field length, time limit, and for remote jobs, the user identification).
O	A list of all files in the output queue is displayed (list includes job name, priority, and for remote jobs, the user identification).
A	A short list (job name only) of all entries in the input, execution, and output queues is displayed.

If a parameter other than those shown above is entered, a list of valid parameters is displayed.

EXECUTING A PROGRAM

A user can compile and execute a program directly under control of INTERCOM. He can interact with the program while it is executing if the procedures for program interaction are followed when the program is created (section 6).

The program text editor, EDITOR, or the SETUP utility may be used to edit an existing program, or to create an interactive program, line by line. When the program is complete, the user can initiate compilation and execution of the program by issuing a single EDITOR or SETUP command.

An alternate method of compiling and executing a user's program is to enter the appropriate SCOPE cards directly from the terminal once the program is created. The user should be knowledgable about the required control cards before this is attempted. An example of this method of program execution is shown in the description of the INTERCOM commands CONNECT/DISCONT.

EDITOR COMMAND

The user may enter the program text editor, EDITOR, to create and edit source programs for subsequent compilation and execution under control of INTERCOM. Specific functions of EDITOR are described in section 4. EDITOR is entered with the command:

```
EDITOR.
```

The system indicates that the user is in EDITOR command mode by displaying two consecutive periods at the terminal:

```
. .
```

The EDITOR command may not be entered under the SETUP utility.

SETUP COMMAND

In addition to EDITOR, a second editing utility, called SETUP, is available to the INTERCOM user. SETUP is included strictly for compatibility with previous versions and will not be supported in future versions of the INTERCOM system. Specific functions of SETUP are described in section 5. To enter the SETUP utility, the user enters:

```
SETUP.
```

The SETUP command may not be entered from within EDITOR.

RECOVERING PROGRAM DIAGNOSTICS

When a user's program is compiled under control of INTERCOM, the program listing is not readily available to the user for determining errors made in the program. A utility is provided to search the file OUTPUT and list the lines in error along with the corresponding diagnostics at the terminal. This utility, in most cases, is called automatically when compilation is initiated with an EDITOR or SETUP command. It may be useful, however, when control cards are used to initiate compilation or if informative diagnostics are desired.

ERRORS COMMAND

The user can request listings of compiler or assembler generated diagnostics from the file OUTPUT. The listing consists of the program or subprogram header card for each routine containing errors, followed by the lines in error and corresponding diagnostics. If a program was created in EDITOR or SETUP, the line number appears to the left of each listed line.

The user enters:

```
ERRORS,system-name.

system-name = ALGOL
              COBOL
              COMPASS
              FTN
              RUN
```

The system name may be abbreviated to the fewest number of characters to uniquely identify that system. The minimum abbreviation required for each system-name is underlined.

The ALGOL, COBOL, FTN and RUN options also list diagnostics for any COMPASS subroutines in the program. Diagnostics generated by the BASIC compiler are sent directly to the terminal as they are encountered by the compiler.

To suppress non-fatal and informative diagnostics from listings, the user enters:

```
ERRORS,system-name,SUP.
```

SUP may be abbreviated to one character, S.

Example:

```
COMMAND- ERRORS,R.

000100   PROGRAM TEST(INPUT,OUTPUT)
000180   N=10*(M+3*(I+1)
*****PN**   UNBALANCED PARENTHESIS

COMMAND-
```

All FORTRAN RUN diagnostics on the file OUTPUT are listed at the terminal.

PROGRAM TEXT EDITOR 4

GENERAL DESCRIPTION

The program text editor provides greater text editing capabilities than the SETUP utility. With EDITOR, the user can create, examine, and modify coded sequential files from a remote terminal. EDITOR can service many remote terminals simultaneously, thus offering performance advantages and improved response time.

Each user under control of EDITOR has a scratch file called the edit file. Coded sequential files to which the user has access may be copied into the edit file for subsequent editing. Any file created or modified in the edit file may be saved on mass storage or submitted directly to the INTERCOM system for compilation and execution.

The basic unit of information in EDITOR is a line of text; it may vary in length up to 510 characters. A line number is appended to each text line. The line number may be entered by the user, or generated by EDITOR. As lines are entered into the edit file, or displayed, a current line pointer is maintained by EDITOR.

Text line formats are predefined as an installation option for the ALGOL, BASIC, COBOL, COMPASS, and FORTRAN programming languages. The user may alter tabulation and margin controls, however, or define a new format.

User files may be listed, deleted, and modified. These operations may be performed on a single text line, a range of text lines (up to a complete file), or on character strings within text lines.

The two modes of operation are: command mode, established when EDITOR is called by the user, and increment mode, initiated by EDITOR when the user enters commands to create or insert text lines with line numbers automatically generated by EDITOR.

EDITOR examines each command; if it conforms to INTERCOM command or SCOPE control card format, it is executed by INTERCOM. Otherwise it is treated as an EDITOR command.

A SCOPE control card of the format:

```
command,p1,p2,...,pn)
```

where a comma is the first separator, and a right parenthesis is the terminator, cannot be distinguished from certain EDITOR commands. Consequently, when under control of EDITOR, the user should not enter SCOPE control cards in this format.

The COMMENT control card may be entered and comments may be appended to control cards; however the comment text must be terminated with a period.

All commands and text lines are entered by pressing the RETURN key on the Teletype or the SEND key on the display keyboard.

EDITOR COMMAND SYNTAX

EDITOR commands are structured as a command verb that may be followed by parameters. Some verbs require no parameters; others require at least one. A command verb with associated parameters must be entered as one line.

NOTATIONS USED

The command formats described in this section are intended to guide the programmer using EDITOR command statements. The following editorial conventions have been used.

Brackets [ ] indicate terms that may be included or omitted as required by the user.

When terms are enclosed in braces { }, only one item must be chosen; the others are to be omitted.

Ellipses . . . immediately following a statement element indicate it may be repeated at the user's option.

Special characters (table 4-1) are essential where shown.

All words shown in lower case letters represent information which the programmer is to supply. These words generally indicate the nature of the information they represent (file name, line or column number, etc).

All words printed entirely in upper case letters have preassigned meaning to EDITOR; these include command verbs and keywords.

Command verbs and keywords may be abbreviated to a unique number of characters; the minimum characters required are underlined. Additional characters may be specified up to the complete verb or keyword, but character sequence must be correct.

The command verb must appear at the beginning of the command statement. Most command parameters, however, are position independent and may appear in any order. Any exceptions are noted in the command descriptions.

All examples are shown as displayed on a Teletype. Data entered by the user is underscored.

SPECIAL CHARACTERS

Characters that have special functions in EDITOR are described in the following table. Most of them have special meaning only in EDITOR commands; they are recognized as valid data characters elsewhere. Others may be entered only for specific functions and may not be used as data characters.

TABLE 4-1. SPECIAL CHARACTERS

Character	Function
=	Used as a separator in EDITOR commands. In text replacement, it must be specified between the text strings. Within commands; it must be specified between keywords and associated variables. When text lines are entered singly, the equals sign must be specified immediately after the line number. In increment mode, when a line number other than the one displayed is to be entered, an equals sign must precede the new line number as well as follow it. An equals sign, when entered as the only character, may also be used to terminate increment mode. The equals symbol may be used as a valid data character in most other situations.
, blank	Blanks and commas are used as parameter separators: either may be used; they are equivalent. Adjacent blanks and commas are interpreted as a single separator. Both may be used as valid data characters; however, trailing blanks are truncated on input to the edit file.
interrupt characters CTRL Z ESC } TTY ALT MODE % } CRT	The INTERCOM interrupt characters for the remote terminals (section 2). An interrupt command (any interrupt character, followed by the character A) may be entered at any time when the program text editor is in use. Current action is terminated and the user is returned to EDITOR command mode. The interrupt command may be also used to exit from increment mode, to terminate a user's interactive job submitted by the RUN command, to terminate SCOPE and INTERCOM commands, and to terminate user's program calls issued while EDITOR is in use. In all cases, the user is returned to EDITOR command mode. If an interrupt command is issued while the editor is deleting, resequencing, or replacing text lines, the edit file is left in an unknown condition.
/	Used to delimit text character strings on input. The symbol must appear, when required, as the first and last characters of the string. If the slash is used as a data character within a string, it must be specified twice. For example, A/B as a text character string within a command must be entered as /A//B/.
( )	Parentheses are used to delimit column numbers in EDITOR commands. They may be used as valid data characters elsewhere.
*EOR	This character string is entered, beginning in column 1, each time an end-of-record is required in a user's file. When the file is saved on mass storage, an end-of-record is written in that position. Conversely, when a file is read, each end-of-record will be inserted in the edit file as *EOR.

ENTERING EDITOR

EDITOR COMMAND MODE

To call the program text editor, the user enters:

```
EDITOR.
```

EDITOR signals readiness to receive input by displaying two consecutive periods:

```
..
```

The user will be in EDITOR command mode and may enter EDITOR, SCOPE, or INTERCOM commands. The EDITOR command mode response will be displayed at the terminal after each command is processed. After the EDITOR command, line=text, however, the editor responds with only a line feed. In either case, another command may then be entered.

Although SCOPE and INTERCOM commands may be entered in EDITOR command mode, execution of these commands is not as efficient as when they are entered in INTERCOM command mode; response time may be affected adversely. If many SCOPE or INTERCOM commands are to be entered in succession, the user should leave the program text editor.

Messages may be sent to the central site with the MESSAGE command, and the TAPE command may be used to read paper tape while in EDITOR command mode. The remote batch processing commands (section 7) and the INTERCOM commands LOGIN, LOGOUT, and SETUP are not allowed.

INCREMENT MODE

Increment mode is initiated when the user enters either a CREATE or ADD command. This mode of operation allows the user to enter lines of text into the edit file. Line numbers are generated by EDITOR. These line numbers will be displayed at the terminal unless specifically suppressed by the user. A line number suppress parameter is provided for either command.

If the suppress parameter is specified, EDITOR generates line numbers, but only the message ENTER LINES is displayed at the terminal. The user may then enter text line by line. A new line number is generated for each text line.

If the parameter is omitted, a line number and an equals sign are displayed. The user may then enter the line of text to be associated with that line number. EDITOR generates a new line number (the previous number incremented by a defined value) and displays the new number and an equals sign.

In either case, the process continues until increment mode is terminated by either the user or EDITOR. Increment mode is not allowed under BASIC format specifications.

The editor terminates increment mode if the user is inserting new text lines between existing lines in the edit file and the value of a generated line number equals or exceeds the next existing line number. The user may terminate increment mode by entering an interrupt command or a single equals sign (table 4-1).

EXIT FROM EDITOR

BYE COMMAND

To leave EDITOR, the user enters:

```
BYE
```

To prevent inadvertent destruction of his file, the user is warned if the edit file has not been saved since it was last modified. The system displays the following message:

```
WARNING-EDIT FILE NOT SAVED
..
```

The user may either save the edit file and re-enter the BYE command, or he may simply re-enter the BYE command if he does not wish to save the file. Control returns to INTERCOM command mode, and the system displays:

```
COMMAND-
```

ALTERNATE EXIT

If the user does not care to save the contents of the edit file, he may leave EDITOR immediately by entering:

```
BYE,BYE
```

The user is returned to INTERCOM command mode, and the edit file is no longer available to him.

Examples:

To leave EDITOR after saving the edit file:

```
..BYE
C0MMAND-
```

user enters BYE
system returns to INTERCOM command mode

To leave EDITOR before saving the edit file:

```
..B
WARNING-EDIT FILE N0T SAVED
..SAVE ABC
..B
C0MMAND-
```

user enters BYE
system displays message
user enters SAVE,filename
he re-enters BYE
system returns to INTERCOM command mode

To leave EDITOR without saving the edit file:

```
..B,B
C0MMAND-
```

user enters BYE,BYE
system returns to INTERCOM command mode

EDITING

The following sections describe the EDITOR commands available to the INTERCOM user. Once the user has entered EDITOR, these commands may be used to manipulate his files. New files may be created or existing files modified. Program formatting may be automatic or declared by the user. Lines can be listed, deleted, added, searched, and replaced. Text character strings may be replaced within lines. Line numbers may be generated automatically and displayed by EDITOR or entered by the user. Line numbers within existing files may be resequenced.

FORMAT COMMAND

When EDITOR command mode is established, an installation-defined format specification is in effect. The tabular column positions, valid tabulating character, and maximum character count per input line are controlled by this specification. The FORMAT command may be used to establish other formats, either predefined or supplied by the user. Specifications established with this command remain in effect for the duration of the user's session with EDITOR, or until changed by the user. This command may also be used to obtain a list of format specifications currently in effect at the terminal.

Every line entered into the edit file is affected by the format specification, which dictates the maximum character count and columnar positions for each input line. In addition, every character entered is checked against the tabulating character. When a valid tabulating character is encountered, blanks are inserted into the data line from that point up to the next tabular position where the next data character is placed. The blank fill is an internal process, spaces do not appear on the terminal display. If a tabulating character is entered when no tabular positions exist, the tabulating character will be accepted as a valid data character.

If lines entered from the terminal exceed the specified character count, they are truncated to the maximum allowed; and a message is displayed at the terminal.

To change or list the format specifications, the user enters:

FORMAT [{ ,format-name / [,TAB=c] [,tab-1 [,tab-2 [,... [,tab-n]]]] [,CH=nnn] / ,SHOW }]

format-name — Establishes a format for data lines entered from the terminal; the name may be one of the following:

ALGOL
BASIC
COBOL
COMPASS
FORTRAN

TAB=c — Keyword; may be entered as T=, TA=, or TAB= (imbedded blanks are not allowed).

c; any valid character on the terminal keyboard (except % on a 200 USER Terminal). The character specified becomes the tab character checked on input.

tab-1,tab-2,...,tab-n — Column positions; 1-3 digits in the range 0-510. Tab column numbers must be specified in ascending sequence.

CH = nnn	Keyword; may be entered as C = or CH = (imbedded blanks are not allowed). nnn; maximum character count, 1- 3 digits in the range 1-510. Establishes the maximum character count for each input line. This count is checked also when either a SAVE or RUN command is entered. CH = 999 allows variable length lines up to 510 characters.
SHOW	Keyword; the current format specification is listed at the terminal in the form: CH = nnn TAB CHAR = c TAB COL = t1,t2,...,tn where nnn is maximum character count; c is tab character; and t is tab column position.

One, two, or all, tabulation parameters including character count, may be entered in one FORMAT command (the tab column positions are interpreted by EDITOR as one parameter). Any omitted parameter will remain unchanged from the current value.

If the FORMAT command is entered with no parameters, the tab column position is set to zero, maximum character count is set to accept variable length lines up to 510 characters (CH999), and no tab character check will be made.

FORMAT NAMES

When a format name is entered in the FORMAT command, a format specification is established at the terminal which enables the user to enter lines in the format of a specific language as listed below:

ALGOL:

Character count = 72

Tab character = ;

Tab columns = 7 10 13 16 19

BASIC:

Character count = 72

Tab character = ;

Tab columns = 0

No tab character check is made under BASIC; and increment mode is not allowed.

COBOL:

Character count = 72

Tab character = ;

Tab columns = 8 12 16 20 24

COMPASS:

Character count = 72

Tab character = ;

Tab columns = 11 18 36

FORTRAN:

Character count = 72

Tab character ;

Tab columns 7

Either FORTRAN Extended or RUN FORTRAN lines may be entered under FORTRAN format.

Examples:

To list the format specification currently in effect at the terminal:

```
••FØRM S                                    user enters FORMAT,SHOW
CH= 72  TAB CHAR=;   TAB CØL=0              system lists current specifications
••                                          EDITOR is ready for next command
```

To define a user format where the colon is the tab character, maximum character count is 50 and tab columns are 10, 20, 30, and 40:

```
••F 10,20,30,40 C=50 TAB=;                  user enters FORMAT,t1,t2,t3,t4,CH=nn,
                                            TAB=t
••                                          EDITOR is ready for next command
```

To change only the tab character in the current format specification; setting the tab character to a down arrow:

```
••FØR T=↓   user enters FORMAT,TAB=c
••          EDITOR is ready for next command
```

To establish the COBOL format specification at the terminal, and show it:

```
••F CØB                                     user enters FORMAT,COBOL
••F S                                       user enters FORMAT,SHOW
CH= 72  TAB CHAR=; TAB CØL= 8 12 16 20 24   system displays format specifications
••                                          EDITOR is ready for next command
```

CREATE COMMAND

The user enters this command to construct a new file.

CREATE [, line [, incr]][SUP]

line	The first line number to be displayed at the terminal; 1-6 digits from 1 to 999999. If omitted, system assumes installation-defined first line number.
incr	Line numbers will be incremented by this value after each text entry; 1-6 digits from 1 to 999999. If omitted, system assumes installation-defined increment value.
SUP	Keyword; suppresses display of EDITOR line numbers at terminal.

On acceptance of the CREATE command, increment mode is initiated; the first line number and an equals sign are displayed at the terminal. The user may enter a text line of 1 to 510 characters, depending on the current format specification. At least one blank character (space) must be entered to produce a blank text line.

If the suppress parameter SUP is specified, only the message ENTER LINES is returned; EDITOR line numbers are generated for each text line but do not appear at the terminal. The user may enter text, line by line, to build his file.

Increment mode remains in effect until terminated by the user with an interrupt command or entry of a single equals sign (table 4-1) which returns the user to EDITOR command mode.

While in increment mode, the user may enter a line number other than that displayed using the following form:

```
=line=text
```

line is a number of 1-6 digits, and text is a text line of 1-510 characters, depending on the current format specification. After this line is entered, increment mode resumes at the point of interrupt.

If the CREATE command is entered when the user has information in the edit file which has not been saved as a private file since it was last modified, EDITOR will ignore the command and display the message:

```
WARNING-EDIT FILE NOT SAVED
. .
```

The user may enter the SAVE command and then re-enter the CREATE command; or if he does not wish to save his edit file, he may simply re-enter the CREATE command; and it will be accepted. In the latter case, contents of the edit file are destroyed.

The CREATE command may not be entered under BASIC format.

Examples:

To create a file of two lines with a first line of 10 and an increment value of 10:

```
. .CREATE 10 10          user enters CREATE,line,incr
   10=FIRST LINE         user enters text line 1
   20=SECØND LINE        user enters text line 2
   30==                  user enters an equals sign to interrupt
                         increment mode
. .                      EDITOR is ready for next command
```

To create a file of three lines using installation-defined line and increment values; and to correct a mistake in the first line:

```
••C                                    user enters CREATE
   100=        BGIN                    user enters first line
   110==100=         BEGIN             user corrects first line
   110=        A+B(I)+A                EDITOR re-issues line number
   120=        END                     user enters third line
   130==                               user enters an equals sign to interrupt
                                       increment mode
••                                     EDITOR is ready for next command
```

To create a file of five lines with a first line of 100 and an increment value of 50 with line numbers suppressed; and to correct a mistake in the second line:

```
••CRE,SU,100,50                        user enters CREATE,SUP,line,incr
ENTER LINES                            system response
      LINE ØNE                         user enters first line
      LINE TØØ                         user enters second line
      =150=LINE TWØ                    user corrects second line
      LINE THREE                       user enters third line
      LINE FØUR                        user enters fourth line
      LINE FIVE                        user enters fifth line
      =                                user terminates increment mode
••                                     EDITOR is ready for next command
```

LINE=TEXT COMMAND

To place one line of data into the edit file, while in either EDITOR command mode or increment mode, the following form may be used:

[=] line=text

line	Line number; 1-6 digits, from 1 to 999999. In increment mode, an equals sign must precede the line number. For BASIC statement input; 1-5 digits, from 1 to 99999.
text	Line of text; 0-510 characters depending on maximum character count established by the format specification.

This command does not affect the terminal's mode of operation. The entered line may replace an existing line or insert a new line in the edit file. If no text is entered (a text line of zero length), a blank line appears at the specified line number in the edit file.

When the BASIC format has been specified for input lines, the EDITOR line number is also the BASIC statement number and part of the text. A BASIC line is entered exactly as a BASIC statement with no equals sign separating the statement number and text. The equals sign may be entered under BASIC format conditions, but it will not be stored as part of the statement.

EDITOR issues only a line feed in response to a valid line=text command; then any command may be entered.

In increment mode, the command must be entered in the form =line=text; the last line number displayed at the terminal will be displayed again to allow the next text line to be entered in proper sequence.

Examples:

To enter line 352 into the edit file; and the text "this is line 352":

```
••352=THIS IS LINE 352
```

user enters line in the form:

line = text

EDITOR responds with a line feed only

To enter a BASIC program into the edit file under BASIC format, request a listing of the edit file, and save the program as a private file named BASFIL:

```
••F,B
••200 FØR X=1 TØ 100
400 PRINT "X=";X
600 PRINT "X**2=";X**2
800 NEXT X
1000 END
LIST,A
    200=200 FØR X=1 TØ 100
    400=400 PRINT "X=";X
    600=600 PRINT "X**2=";X**2
    800=800 NEXT X
   1000=1000 END
••SAVE BASFIL
••
```

user enters FORMAT,BASIC
user enters BASIC statements line by line

user enters LIST,ALL
system lists contents of edit file

user enters SAVE,filename
EDITOR is ready for next command

To correct a line while in increment mode:

```
••CRE
    100=;PRØGRAM MINE
    200=;DIMENSIØN A(100,200)
    300==100=;PRØGRAM MINE(INPUT,ØUTPUT,TAPE1=ØUTPUT)
    300=;READ 1000,A
    400==
••
```

user enters CREATE; increment mode is initiated

user enters an equals sign to terminate increment mode

EDITOR is ready for next command

FILE STORAGE AND MANIPULATION

EDIT COMMAND

The user enters the EDIT command to load a private file into the edit file:

EDIT,filename [,SEQUENCE]

filename	Name of file to be edited; required immediately following the command verb.
SEQUENCE	Keyword; EDITOR line numbers will be assigned to each line as they are entered into the edit file. If omitted, system assumes that EDITOR line numbers already exist in the local file.

The file name may be any coded sequential file to which the user has read access, including private and attached permanent files. The file to be loaded is called the source file; it is not modified by execution of the EDIT command.

If the user enters the EDIT command with information in the edit file which has not been saved as a private file since it was last modified, EDITOR ignores the command and displays the message:

```
WARNING-EDIT FILE NOT SAVED
```

The user may then save the edit file, or if the contents of the edit file need not be retained, simply re-enter the EDIT command. In the latter case, the contents of the edit file are destroyed.

When loading a file created outside of EDITOR, the user is required to sequence the file with line numbers. When SEQUENCE is specified, EDITOR line numbers beginning with the installation-defined first line number are appended to each line of the file. The source file is not affected; line numbers appear only in the edit file. Consequently, the length of each line in the edit file is increased by six characters. Because edit file lines are restricted to a maximum of 510 characters, truncation may occur; if so, an informative message is displayed. When BASIC format specifications are in effect, the keyword SEQUENCE is not allowed. A BASIC program existing as a private file without EDITOR line numbers can be copied to a new file with EDITOR line numbers using the INTERCOM command, CONVERT (section 3).

Multi-record files may be loaded for editing, but they appear in the edit file as one record. On encountering an end-of-record in the source file, the character string *EOR is assigned a sequential line number and written in the edit file to indicate an end-of-record condition.

Example:

To load the private file named AFIL into the edit file:

```
..E AFIL          user enters EDIT,filename
..                EDITOR is ready for the next command
```

To load the private file BFIL into the edit file with line number sequencing:

```
..ED,BFIL S       user enters EDIT,filename,SEQUENCE
..                EDITOR is ready for the next command
```

SAVE COMMAND

To save the edit file as a private file, the user enters:

SAVE,filename [,NOSEQ] [,OVERWRITE]

filename	Name under which edit file is saved as a private file; required immediately following the command verb
NOSEQ	Keyword; causes EDITOR line numbers to be suppressed in private file
OVERWRITE	Keyword; causes any file of the same file name to be overwritten

The file is saved as a private file in standard SCOPE coded sequential format. The line length in the saved file is determined by the format specification currently in effect at the terminal. Lines will be blank filled or truncated accordingly. If truncation is necessary, a message is sent indicating the length of the longest line encountered; the user may change the format character count and re-enter the SAVE command. The SAVE command does not destroy the edit file.

If the file is to be retained as permanent after logout, it must reside on a permanent file device. To ensure this, the SCOPE control card command REQUEST,filename,*PF. may be entered before the SAVE command is entered. The edit file may then be saved with the SAVE command (the keyword OVERWRITE must be specified) and then made permanent with the INTERCOM command STORE (section 3).

If the user has not specified the keyword OVERWRITE, and a private file exists with the same file name, the SAVE command is ignored and an error message is displayed. An attached permanent file cannot be overwritten.

Examples:

To save the edit file under the file name FTNPRG:

```
..S FTNPRG          user enters SAVE,filename
..                  EDITOR is ready for the next command
```

To save the edit file in place of an existing private file named FTNDATA, with a line length of 400 characters and no EDITOR line numbers:

```
..F CH=400          user enters FORMAT,CH=nnn
..S,FTNDATA,0,N     user enters SAVE,filename,OVERWRITE,NOSEQ
..                  EDITOR is ready for the next command
```

LIST COMMAND

This command permits the user to list edit file lines at the terminal.

LIST [, {ALL / line-1 / LAST} [, {line-2 / LAST}]] [,/text/[,(col-1[,col-2])][,UNIT]]

ALL Keyword; all lines in the file are listed or searched for the text search string.

line-1	Line number; 1-6 digits, from 1 to 999999; first or only line to be listed or searched.
line-2	Line number; 1-6 digits, from 2 to 999999; last line to be listed or searched in a range beginning at line-1.
LAST	Keyword; if specified as first parameter, the last line in the file is displayed or searched; if specified as second parameter, the listing or search begins at line-1 and continues through the last line in the file.
/text/	Text search string; 1-20 characters delimited by slashes; file is searched for this text string (search may be restricted to a range of line and column numbers). Lines containing the text string are listed at the terminal.
col-1	Column number; 1-3 digits, from 1 to 510; first or only column number of a text string search. Must be preceded by a left parenthesis and followed by either col-2 or a right parenthesis.
col-2	Column number; 1-3 digits, 2 to 510; last column number of a text string search in a range beginning in col-1. Must be greater in value than col-1 and followed by a right parenthesis. The range of columns must be at least equal to the number of characters in the text string. Column specification is significant only if a text search string is specified. Lines are listed only if the text string occurs within the range, or if it begins in col-1 when only a single column is specified.
UNIT	Keyword; dictates the text search string appear as a unit within a line; the text string must be delimited by non-alphanumeric characters (including blank).

If the LIST command is entered with no parameters, the current line is listed (line to which the edit file pointer is set Lines which satisfy the LIST command requirements are displayed in the form:

line number = text line

Examples:

To list lines 10 through 20 of the edit file, and then the current line:

```
••L 10 20                    user enters LIST,line-1,line- 2
   10= DATA 10               system lists appropriate lines
   15= DATA 15
   20= DATA 20
••L                          user enters LIST
   20= DATA 20               system lists current line
••                           EDITOR is ready for next command
```

To list all lines in the edit file which contain the variable AX as a unit:

```
••LI,A,/AX/,U                user enters LIST,ALL,/text/.UNIT
  100=  AX=X**2              system lists all lines which satisfy command requirements
  620=  PRINT,AX
••                           EDITOR is ready for next command
```

ADD COMMAND

This command is used to insert new lines between existing lines or to add lines at the end of the edit file. The ADD command may not be used to replace or bypass existing lines.

ADD [,line [,incr]] [SUP]

line	Line number; 1-6 digits, from 1 to 999999; first line number displayed at the terminal. If omitted, system assumes last line number in edit file plus installation-defined increment value.
incr	Increment value; 1-6 digits, from 1 to 999999. If omitted, system assumes installation- defined increment value.
SUP	Keyword; suppresses display of EDITOR line numbers at terminal

Increment mode is initiated; the first line number and an equals sign are displayed at the terminal. The user may enter a line of text, 1 to 510 characters, depending on the current format specification. At least one blank character (space) must be entered to produce a blank or zero length text line.

If the suppress parameter SUP is specified, only the message ENTER LINES is returned; EDITOR line numbers are generated for each text line but do not appear at the terminal. The user may enter text, line by line, to build his file.

Increment mode remains in effect until terminated by the user with an interrupt command or entry of a single equals sign (table 4-1), or by EDITOR when an existing line number is encountered.

While in increment mode, the following form may be used to enter any line number other than that displayed:

=line=text

line is the desired line number of 1-6 digits, and text is a line of text of 1-510 characters, depending on the current format specification.

The ADD command may not be entered under BASIC format specifications.

Examples:

To add new text lines between lines 10 and 20 in the edit file and correct an error in the first line entered:

```
**A 13 3                          user enters ADD,line,incr
   13= INSET ØNE                  user enters first line
   16==13= INSERT ØNE             user corrects spelling in first line
   16= INSERT TWØ                 system redisplays line number
   19= INSERT THREE
ADD WØN'T REPLACE ØR BYPASS LINES system displays message; next line would
                                  exceed line 20 in edit file
**                                EDITOR is ready for next command
```

To add a line to the end of the edit file (last existing line number is 300, installation increment value is 10):

```
**A                               user enters ADD
  310= RETURN                     user enters text
  320==                           user enters an equals sign to terminate
                                  increment mode.
**                                EDITOR is ready for next command
```

To add lines to the end of the edit file with line numbers suppressed:

```
••ADD,S                    user enters ADD,SUP
ENTER LINES                system response
      3,RETURN  B          user enters a line,
      3,RETURN  C          and another, and
      ,END                 one more
      =                    user terminates increment mode
••                         EDITOR is ready for another command
```

DELETE COMMAND

The user enters this command to delete lines in the edit file.

DELETE, { ALL / line-1 / LAST } [, { line-2 / LAST }] [,/text/[,(col-1[,col-2])][,UNIT]][,VETO]

ALL — Keyword; all lines in the edit file are deleted or searched for the text search string.

line-1 — Line number; 1-6 digits, from 1 to 999999; first or only line to be deleted or searched.

line-2 — Line number; 1-6 digits, from 2 to 999999; last line to be deleted or searched in a range beginning at line-1.

LAST — Keyword; as first parameter, causes last line in edit file to be deleted or searched; as second parameter, causes deletion or search of lines beginning at line-1 through the last line in the file.

/text/ — Text search string; 1-20 characters delimited by slashes; file is searched for this text string (search may be restricted to range of line and column numbers). Lines containing this string are deleted from edit file.

col-1 — Column number; 1-3 digits, from 1 to 510; first or only column number of text string search. Must be preceded by a left parenthesis and followed by either col-2 or a right parenthesis.

col-2 — Column number; 1-3 digits, from 2 to 510; last column number to be searched in a range beginning at col-1. Must be greater in value than col-1 and followed by a right parenthesis. The range must be at least equal to the number of characters specified in the text search string.

Column specification is significant only if a text search string is specified. Lines are deleted only if the text string occurs within the range, or if it begins in col-1, when a single column is specified.

UNIT — Keyword; dictates that the text search string appear as a unit within a line; it must be delimited by non-alphanumeric characters (including blank).

VETO — Keyword; permits the user to approve deletions before they occur. The line to be deleted is displayed at the terminal; the user may enter Y, YE, or YES to delete, or any other character, or no character to retain the line.

The DELETE command must include at least one parameter; ALL, LAST, or a line number.

If a text search string is specified in the command, a message reports the number of deletions performed:

```
n DELETIONS
```

n is the number of lines deleted.

An interrupt command may be entered to terminate execution of a DELETE command; however, the edit file may be left with the specified lines partially deleted.

Examples:

To delete line 100 in the edit file:

```
••DELETE 100                    user enters DELETE, line- 1
••                              EDITOR is ready for next command
```

To delete from lines 200 through the last line, only if the character string AX appears in columns 7 through 72:

```
••D,/AX/,(7,72),200,L           user enters DELETE,/text/.(col-1,col-2),
                                line-1,LAST
     2 DELETIØNS                system message
••                              EDITOR is ready for next command
```

To delete all lines from line 100 through line 200:

```
••DE 100 200                    user enters DELETE,line- 1,line-2
••                              EDITOR is ready for next command
```

To delete all lines in the edit file so that a new file may be constructed:

```
••DEL AL                        user enters DELETE,ALL
••                              EDITOR is ready for next command
```

To delete with veto power, all lines in the edit file only if they contain the character C in column 1 as a unit:

```
••D,A,/C/,(1),U,V               user enters DELETE,ALL/text/,(col-1),
                                UNIT,VETO
     20=C BEGIN DØ LØØP         system displays qualifying line
N                               user elects to retain line
     50=C END SCAN              system displays qualifying line
N                               user elects to retain line
      0 DELETIØNS               system message; all qualifying lines dis-
                                played, none deleted
••                              EDITOR is ready for next command
```

RESEQ COMMAND

To resequence the line numbers in the edit file, the user enters:

RESEQ[,line[,incr]]

The command verb must be entered with at least two characters.

line	Line number at which resequencing is to begin. 1-6 digits, from 1 to 999999. If omitted, system assumes installation-defined
incr	Increment value; 1-6 digits, from 1 to 999999. If omitted, system assumes installation-defined increment value.

On acceptance of this command, EDITOR resequences all line numbers in the edit file. New line numbers are written over existing line numbers, and the current line number pointer is reset to the first line number in the file.

BASIC program files should not be resequenced with this command; if the BASIC format specification is in effect, the RESEQ command is illegal.

The user may terminate a RESEQ command with an interrupt command; however, the result may be a partially resequenced edit file which should be resequenced before further editing.

Examples:

To resequence the edit file with first line number of 100 and increment value of 100:

••RES 100 100	user enters RESEQ,line,incr
••	EDITOR is ready for next command

To resequence with installation-defined first line number and increment value:

••RE	user enters RESEQ
••	EDITOR is ready for next command

TEXT REPLACEMENT COMMAND

To replace text strings in lines of the edit file, the user enters:

/text-1/=/text-2/ [, {ALL / line-1 / LAST} [, {line-2 / LAST}]] [,(col-1[,col-2])] [,UNIT] [,VETO]

/text-1/=/text-2/	Text strings; equals sign must be specified with no spaces on either side.
/text-1/	Text search string; 1-20 characters delimited by slashes. File is searched for this string (search may be restricted to a range of line and column numbers).
/text-2/	Text replacement string; 0-20 characters delimited by slashes. Replaces text search string when conditions of the search are satisfied.

ALL	Keyword; causes a search of all lines in the edit file.
line-1	Line number; 1-6 digits, from 1 to 999999; first or only line to be searched.
line-2	Line number; 1-6 digits, from 2 to 999999; last line to be searched in a range beginning at line-1.
LAST	Keyword; as first parameter, a search is made of the last line in the file; as second parameter, a search is made beginning at line-1 through the last line in the file.
col-1	Column number; 1-3 digits, from 1 to 510; first or only column to be searched. Must be preceded by a left parenthesis and followed by either col-2 or a right parenthesis.
col-2	Column number; 1-3 digits, from 2 to 510; last column to be searched in a range beginning at col-1. Must be greater than col- 1 and followed by a right parenthesis. The range must be at least equal to the number of characters in the text search string. Replacement takes place only if the text string occurs within the column range, or if the text string begins in col-1, when a single column is specified.
UNIT	Keyword; dictates that the text search string appear as a unit within a line; the text string must be delimited by non- alphanumeric characters (including blank).
VETO	Keyword; permits the user to approve text replacement before it occurs. The changed form of the line is displayed at the terminal; the user may enter Y, Ye, or YES to accept the change, or to retain the original line he may enter any other character or no character.

If the TEXT REPLACEMENT command is entered with no parameters, the search will be performed on the line to which the current line pointer is set.

The number of replacements performed are reported in a message:

n CHANGES

where n is the number of changes made. Because more than one replacement may occur in any line, the number of changes displayed may differ from the number of lines changed.

The two text strings specified as the command verb need not contain the same number of characters; the line affected will be expanded or contracted as necessary. If the maximum character count is exceeded, the replacement occurs, and an informative message is displayed. Truncation occurs if a line exceeds 510 characters.

The text replacement string may be entered as a null string (two consecutive slashes, no imbedded blanks). This specification causes the text search string to be deleted if all search conditions are satisfied.

Examples:

To replace the variable name AX with the name BZ in the current line; AX must be a unit:

```
..<u>/AX/=/BZ/,U</u>                          user enters /text-1/=/text-2/,UNIT
      1 CHANGES                              system message
..                                           EDITOR is ready for next command
```

To replace the character string TCS, wherever it appears in the edit file, with the string TERMINAL CONTROL SYSTEM. (Two TEXT REPLACEMENT commands must be entered because the text replacement string is greater than 20 characters.) The user requests veto power:

```
..<u>/TCS/=/TERMINAL CØNTRØL SY./,A,V</u>     user enters /text-1/=/text-2/,ALL,VETO
    60=THE TERMINAL CØNTRØL SY. HAS THE      system displays changed line
                                             user accepts change
<u>YES</u>
   190=IN THE TERMINAL CØNTRØL SY. USERS MAY  system displays changed line
                                             user accepts change
<u>Y</u>
  2000=* * * TERMINAL CØNTRØL SY. ABØRT * * * system displays changed line
                                             user retains original line
<u>N</u>
      2 CHANGES                              system message
..<u>/SY./=/SYSTEM/ 60 190 U</u>              user enters /text-1/=/text-2/,line-1,line-2,UNIT
      2 CHANGES                              system message
..                                           EDITOR is ready for next command
```

To replace the character C with the character * only if C appears as a unit in column 1. All lines are searched:

```
../C/=/*/ A (1) U                            user enters /text-1/=/text-2/,ALL,(col-1),UNIT
     15 CHANGES                              system message
..                                           EDITOR is ready for next command
```

To replace the character string PROGRAM in line 2310 with a null string. (Line currently appears as 2310=END PROGRAM.) The user requests veto power:

```
..<u>/PRØGRAM/=//,2310,V</u>.                 user enters /text-1/=/text-2/,line-1,VETO
  2310=END                                   system displays changed line
<u>YES</u>                                    user accepts change
      1 CHANGES                              system message
..                                           EDITOR is ready for next command
```

COMPILATION AND EXECUTION

RUN COMMAND

With the RUN command, the user can transfer a private file or a program in the edit file to one of the available systems for assembly or compilation and, possibly, execution.

RUN,system-name [,FILE=filename] [,NOEX] [,SUP]

	Command verb must be entered as RU or RUN; system-name is required.
system-name	Compiler or assembler name; must be one of the following:

ALGOL	ALGOL compiler
BASIC	BASIC compiler
COBOL	COBOL compiler
COMPASS	COMPASS assembler
FTN	FORTRAN Extended compiler
RUN	FORTRAN RUN compiler

FILE=filename	Keyword; may be entered as F=, FI=, FIL=, or FILE= (imbedded blanks are not allowed).
	Filename; name of private file to be transferred.
	If omitted, contents of the edit file are transferred for compilation or assembly.
NOEX	Keyword; inhibits execution. If omitted, execution is performed.
SUP	Keyword; suppresses non-fatal and informative diagnostics from list displayed at the terminal (ineffective on BASIC compilations). If omitted, system displays all messages.

When compilation or assembly begins, EDITOR displays the message:

```
JOB COMPILING
```

When fatal errors are encountered during assembly or compilation, the lines containing errors (fatal and non-fatal), the EDITOR line numbers, and the diagnostics are listed at the terminal. Non-fatal diagnostics may be suppressed from the list.

If no errors are encountered, execution begins (unless explicitly inhibited). The user's field length is reduced automatically to the minimum required for execution after the program is loaded. When running ALGOL programs, the user should prevent automatic field length reduction by using the REDUCE command (section 3).

INTERACTION

During execution of BASIC, FORTRAN, ALGOL, COMPASS, or COBOL programs, the terminal user can interact with the executing program (as detailed in section 6). He can receive output at the terminal and also enter input to the executing program from the keyboard. If a Teletype terminal has a paper tape punch and reader, a program can call for input from paper tape with pauses (X-OFF) for output to the terminal. The INPUT and OUTPUT files are connected automatically to the terminal, and all I/O requests for these files are directed to the terminal. These files may be disconnected and reconnected, and other files may be connected by requests from the executing program.

The executing program will pause when input is expected. It is helpful if the program contains a signal to the terminal user that input is expected and indicates the type of input. For instance, a PRINT statement can be included just prior to a READ statement in a FORTRAN program. In COBOL programs, the message ENTER COBOL INPUT- is displayed at the terminal whenever the ACCEPT FROM statement expects input from the terminal. The BASIC INPUT statement displays a question mark at the terminal to indicate that user input is expected.

Dayfile messages generated by the program and execution diagnostics are also listed. However, for all FORTRAN programs, the file OUTPUT should be declared on the PROGRAM card to permit execution diagnostics to be listed at the terminal. Similarly, for COBOL programs, the printer file must be declared as OUTPUT to permit execution diagnostics to be listed at the terminal.

Three local user files are generated, or overwritten, when the RUN command is executed: OUTPUT, LGO, and SETFILE. For the COMPASS assembler and all compilers, except BASIC, the file OUTPUT contains the compilation listing for the user's program. This file may be printed at the central or remote site if a listing of the program is desired; or it may be examined at a terminal with the INTERCOM command, PAGE (section 3).

The file LGO contains the binary code generated by the assembler or compilers; this file may be executed later in the current terminal session without reassembly or recompilation of the program. For BASIC compilations, LGO is generated only if the user inhibits execution.

When the user's edit file is to be run, it is copied automatically to the file SETFILE, which becomes the file transferred to the compiler or assembler. The length of the lines copied to SETFILE is restricted by the format specification in effect at the terminal. All lines are blank-filled or truncated to conform to the maximum character count. If truncation occurs, a message indicating the length of the longest line encountered is displayed.

Examples:

To compile a BASIC program contained in the edit file (an error is encountered by the compiler):

```
••RU,B                          user enters RUN,system-name
JØB CØMPILING                   system message
ILLEGAL STRING IN 330           compilation error message
••                              EDITOR is ready for next command
```

To compile and execute a FORTRAN Extended program contained in the private file XPROG; the executing program requests input data, processes data, and displays results:

```
••RU,F=XPRØG,FTN                user enters RUN,FILE=filename,
                                system-name
JØB CØMPILING                   system message
ENTER 2-DIGIT NUMBER 13         program request; user enters data
THE CUBE ØF  13 IS  2197        program displays result of computation
••                              EDITOR is ready for next command
```

To assemble and execute a COMPASS program contained in the edit file (an arithmetic error terminates the job during execution):

```
..RUN C0MP                                    user enters RUN,system-name
J0B C0MPILING                                 system message
ARITHMETIC ERR0R   M0DE = 1   ADDRESS=23427   system message
..                                            EDITOR is ready for next command
```

To compile a COBOL program contained in the private file PAYROLL and then save the file LGO as a permanent file under the name PAYBIN (execution is inhibited):

```
..REQUEST,PAYBIN,*PF.          user enters
..RU F=PAYR0LL,C0B0L,N         RUN,FILE=filename,system-name,,
                               NOEX
J0B C0MPILING                  system message
..C0PY(LG0,PAYBIN)             user enters INTERCOM command
                               to rename the LGO file
..ST0RE,PAYBIN.                user enters INTERCOM command
ID=GARYR                       to create a permanent file

..                             EDITOR is ready for next command
```

STOPPING EXECUTION

At any time, the user may stop execution of a program and return to EDITOR command mode.

At a Teletype terminal, he presses the CTRL Z, ESC, or ALT MODE key followed by A; at a display terminal, he presses the % (SHIFT 5) key followed by A:

```
%A
USER ABORT
..
```

The program currently executing terminates, and the user is returned to EDITOR command mode. The contents of the edit file are not affected by a user abort.

SETUP 5

The SETUP utility program may be used to create and edit source programs and to submit these programs for compilation and execution; however, in future versions of INTERCOM, SETUP will be entirely replaced by the program text editor described in section 4.

GENERAL DESCRIPTION

SETUP is a utility program that gives the user flexibility in building and executing a program. With the SETUP utility, the user can:

Modify a program previously stored on disk

Construct a program file line by line from the terminal

Edit the program during its creation or after compilation and execution

Compile the program

Correct compilation errors and resubmit the program

Execute a correctly compiled program

Save a program file for subsequent compilation or execution

The user may enter source program lines without the restrictions of a precise card format.

All INTERCOM commands except SCREEN, MESSAGE, EDITOR, SAVEFL, and the batch processing commands are accessible to the user while in SETUP and can be entered as if he were in command mode. The SCOPE utility DMP is not allowed while in SETUP.

TEXT BUFFER

When SETUP is called by the SETUP command, the user is assigned a text buffer. He may enter into this buffer a program already stored on mass storage, or he may construct a new program in the buffer line by line. In either case, contents of the buffer can be edited from the terminal.

The standard buffer size is determined when INTERCOM is installed. Should a program be larger than the buffer size, the portion to be manipulated is called into the buffer automatically, while the rest of the program is stored on a mass storage device until needed.

The contents of the text buffer remain intact, unless altered by the user, until the user leaves SETUP. A program prepared in the text buffer can be submitted for compilation and for execution if no compilation errors occurred. If compilation produces error diagnostics, the user can correct the program and resubmit it.

A file in the buffer can be saved as one of the user's private files, with the SAVE directive. If the file is to be kept after logout, it may be saved with the SAVE directive and then made permanent with the INTERCOM command STORE.

DIRECTIVES

The operation of SETUP is controlled by directives which specify the action to be taken. Directives must often be followed by one or more options which will be requested by the system immediately after the directive is entered. If the user anticipates the request, he can enter the option following the directive and separated from it by a slash.

In addition to requests for specific directives, SETUP signals it is ready to receive input by displaying:

```
READY.
```

The user may then enter a SETUP directive or any legal INTERCOM command. The user remains in SETUP until he requests a return to command mode with BYE.

Delimiters that may be specified in SETUP directives are: space, slash, period or comma. In discussions of SETUP directives in this manual, wherever one of these delimiters is shown in a format, any of the other legal delimiters may be substituted. SETUP directives are not terminated with a period or right parenthesis. They are entered by pressing the RETURN key on the Teletype or SEND on the display keyboard.

ENTERING SETUP

The SETUP utility is entered with the command:

```
SETUP.
```

The system responds:

```
ON AT hh.mm.ss.
SYSTEM ---FORTRAN
NEW OR OLD FILE--
```

The response gives the time in hours, minutes, and seconds when SETUP was entered, the default system mode, FORTRAN; and then requests a new or old file; however, the user may enter any legal command or directive.

Example:

```
CØMMAND- SETUP.
ØN AT     14.04.38.
SYSTEM--FØRTRAN
NEW ØR ØLD FILE--
```

SYSTEM DIRECTIVE

Three system modes are available: BASIC, FORTRAN or GENERAL. In the BASIC mode, the user enters a program written in the BASIC language, or constructs his program using BASIC. In the FORTRAN mode, he enters or constructs programs in FORTRAN RUN or FORTRAN Extended. In GENERAL mode, he can use programming languages other than BASIC or FORTRAN. For instance, a COBOL program or a SCOPE control card file for subsequent execution as a batch job should be constructed and entered in the GENERAL mode.

FORTRAN is the default mode if SYSTEM is not specified. To operate in a mode other than FORTRAN, or to return to FORTRAN from another system mode, the user must issue the system directive:

```
SYSTEM
```

The system responds:

```
NEW SYSTEM-
```

Then the user specifies:

```
BASIC   or   GENERAL   or   FORTRAN
```

ABBREVIATED ENTRY

```
SYSTEM/option
```

OLD/NEW DIRECTIVE

Before files can be manipulated, an existing file (OLD) must be read into the buffer; or a new file (NEW) must be declared. To request an existing file, the user enters:

```
OLD
```

The system responds:

```
OLD FILE NAME-
```

The user enters a valid file name. After the file is read into the text buffer, the system replies:

```
READY.
```

If the file cannot be located on a mass storage device, the system requests that the user declare a new file with the specified name.

ABBREVIATED ENTRY

```
OLD/filename
```

To declare a new file, the user enters:

```
NEW
```

The system responds:

```
NEW FILE NAME-
```

The user enters a name to be associated with the file he is creating; the file name should be acceptable to the system he is using. If the file name is acceptable, the system responds as follows and the previous contents of the text buffer are lost:

```
READY.
```

ABBREVIATED ENTRY

```
NEW/filename
```

Examples:

```
C0MMAND- SETUP.
0N AT      09.18.08.
SYSTEM--F0RTRAN
NEW 0R 0LD FILE--SYSTEM
NEW SYSTEM--BASIC

READY.
NEW
NEW FILE NAME--JG0W
```

A new BASIC file called JGOW can be constructed now by the user.

```
C0MMAND- SETUP.
0N AT      00.14.53.
SYSTEM--F0RTRAN
NEW 0R 0LD FILE--0LD/F0RT

READY.
```

The user calls his private file, FORT, into the text buffer.

```
C0MMAND- SETUP.
0N AT      12.32.15.
SYSTEM--F0RTRAN
NEW 0R 0LD FILE--SYSTEM/GENERAL

READY.
NEW/CNTRL1

READY.
```

A new file can be constructed now with the name CNTRL1.

EXIT FROM SETUP

When the user has completed his job in the SETUP mode, he may return to the command mode by entering any of the following directives:

```
BYE

GOODBYE

END
```

The system displays:

```
OFF AT hh.mm.ss.
```

This response records the time the user exited from SETUP in hours, minutes and seconds.

EDITING

LINE NUMBERS

Line numbers are used for editing. Although the user can enter source program lines without format restrictions under SETUP, he must, nevertheless, precede each source line with a line number. This SETUP line number is 1-5 decimal digits; it is terminated by the first space, the first non-numeric character, or after five digits have been encountered.

As each line is entered, the system responds with a line feed on a teletypewriter or by advancing the entry marker and send indicator to the next line of a display screen. The response READY. is not returned by SETUP following input of a source line.

Each source line entered is reformatted by SETUP in accordance with the format rules of the system. The examples below show how SETUP alters lines for each system.

FORTRAN

A line for a file named FACTOR, preceded by the line number 12345, is typed:

```
12345100 FORMAT (1H1,8(5XF10.8))
```

SETUP reformats the line and enters it in the text buffer as:

1	7	73	76
100	FORMAT (1H1,8(5XF10.8))	FAC	12345

A comment line may be entered in a FORTRAN source program by specifiying the character C or * immediately following the line number. The comment, consisting of any alphanumeric character including space, may then be typed. The carriage return terminates the comment. The comment is entered in the text buffer and is treated as a FORTRAN comment line by the FORTRAN compiler. To enter a FORTRAN statement that begins with the character C, a blank should be entered between the line number and the FORTRAN statement.

A FORTRAN continuation card is indicated by a + as the first non-blank character after the SETUP line number.

BASIC

A line preceded by the line number 100 is entered into a file named FACTOR:

```
100 FOR X = 1 TO 15    or    100FOR X = 1 to 15
```

The system alters the line and enters it in the text buffer as:

1	7	73	76
100	FOR X = 1 to 15	FAC	00100

The statement number acts as the line number in BASIC.

GENERAL

If the user types the line:

```
00050REWIND(FACTOR)
```

SETUP reformats it as:

1	7 3	7 6
REWIND(FACTOR)	FAC	00050

The terminal user is concerned with the line number primarily for editing purposes. Apart from recognizing that SETUP reformats lines, he has little need to know the details of this internal conversion.

The user may resequence the line numbers of a program written in the FORTRAN or GENERAL mode with the directive RESEQ.

LINE EDITING

To insert a new line, the SETUP line number is entered followed by the line. If the line number already exists in the text buffer, the existing line is replaced by the new line. If the line number does not already exist, the line is inserted in its place in the sequence of line numbers.

A line number entered alone with no characters following it deletes the line with that number from the text buffer.

Examples:

Lines with numbers 10, 20, 30, and 40 have already been entered.

The user enters:

10	Line 10 is deleted.
25 DØ 100 K = 1,10	Insert line 25 between lines 20 and 30.
40 A = 450	Replace line 40.
50 100 CØNTINUE	New line is entered in proper position.

INTRA-LINE EDITING

To make a change within a text buffer line, the user enters a series of codes to specify the line, the type of editing and whether or not the line, as corrected, is to be displayed at the terminal. The first code entered specifies whether or not the corrected line is to be displayed at the terminal:

*	Do not display corrected line
	Display corrected line

Next, the user enters a code letter that specifies the type of line or literal to be edited:

Code	Meaning
F	FORTRAN statement number
M	SETUP line number
I	Literal
C	Last line edited

Following F or M, he enters the line number. Following I, the literal is entered within delimiters. A line number can be referenced by specifying a number plus or minus a decimal number less than 99.

For example:

`*F100+9` means edit a line 9 lines after the FORTRAN statement number 100.

Continuing on the same line, after the code and any line number or literal, the user specifies the change as indicated by the code letters:

Code	Meaning
D	Delete a literal
R	Replace a literal
I	Insert a literal

When D is entered, the next entry on the line is a 1-60 character literal enclosed within a pair of delimiters. When R is entered, the literal to be replaced follows, enclosed by a pair of delimiters, and followed by the literal to be inserted enclosed by another pair of delimiters. When I is entered, it is followed by a literal enclosed in delimiters which indicates where the insertion is to be placed; this literal is followed by the literal to be inserted, also enclosed in delimiters. These delimiters may be any typed character, but they must be a pair of the same character.

Examples:

To delete the literal, THE ARRAY NAMES, from a line with the FORTRAN statement number 100:

```
*F100D/THE ARRAY NAMES/
```

SETUP will not display the corrected line. The slashes are delimiters enclosing the literal.

To insert the literal K = 1,100 following the literal DO 200 in SETUP line number 55:

```
-M55I*D0 200*/K=1,100/
```

Asterisk and slash pairs act as delimiters.

The corrected line is displayed:

```
D0 200K=1,100
```

To replace the literal TP with the literal TO at the first occurrence of the literal GO TP in the program:

```
*I/G0TP/R/TP//T0/
```

Slashes are used as delimiters; the corrected line is not displayed.

To delete the literal *5 from the line previously edited:

```
-CDX*5X
```

X's are delimiters; the corrected line is displayed.

RESEQ DIRECTIVE

The SETUP line numbers in the text buffer can be resequenced by specifying:

```
RESEQ
```

The initial line is assigned the number 100, and successive line numbers are incremented by 10 from that point to the end of the program.

Because SETUP line numbers are also BASIC statement numbers, resequencing would destroy the flow of a BASIC program. For this reason, RESEQ is illegal in BASIC system mode. RESEQ is useful in FORTRAN or GENERAL modes.

Example:

The user inputs:

```
100 PRØGRAM A(ØUTPUT)
115 2 FØRMAT(* ABCDE*)
120 PRINT 2
```

He then resequences and lists the resequenced text buffer:

```
RESEQ

READY.
LIST

    100  PRØGRAM A(ØUTPUT)
    110  2 FØRMAT(* ABCDE*)
    120  PRINT 2

READY.
```

TRANS DIRECTIVE

This directive applies only to programs written in BASIC. It is used if a BASIC program was not created in the text buffer, but was created by other means and read into the buffer using the directive OLD. The lines have numbers, but they are not copied internally to columns 76-80. These numbers are duplicated in columns 76-80 with the directive:

```
TRANS
```

The line numbers are then available as SETUP line numbers for editing and reference. TRANS does not resequence the line numbers.

TAB DIRECTIVE

The user in GENERAL mode may control the format of his input lines with the TAB directive which allows tab positions to be preset. TAB causes internal skips to specified column positions. TAB is particularly useful in the construction of a COBOL program. If the TAB directive is used in any other mode than GENERAL, the tab character is input as a blank.

When a valid tabulation key is pressed, SETUP formats the next character entered to begin in the first tab position beyond the current column. No physical movement of the carriage is made; the line is entered into the text buffer in the format determined by the TAB directive. As many as 20 internal tab positions can be set.

The valid tabulation keys are discussed in section 2 and noted in appendix A. At a display terminal, the tab character is displayed wherever it is entered in the line. At a Teletype terminal, the |! is printed wherever it is entered in the line; nothing is printed when CTRL I is pressed.

To set one or more tabs, enter:

`TAB,n1,n2,n3,...,nj` where $1 \leq j \leq 20$

The nj are positive integers 2 to 72. Any other legal delimiter can be substituted for the comma. The first tab is set at the column number specified by n1, n2 sets the second tab and so forth up to 20 positions. Tabs need not be specified in ascending order, SETUP will resequence them automatically.

A subsequent TAB directive erases and overlays the previous TAB directive.

To clear all previous tabs, enter:

`TAB,F`

Examples:

<u>TAB/15/20/30</u>

If the line is currently positioned between columns 1 and 14, pressing the tabulation key positions the next character at column 15; if the line is positioned between columns 16 and 19, pressing the tabulation key causes the next character to be positioned at column 20; and so forth.

If the line is positioned at or after column 31, pressing the tabulation key causes the rest of the line to be ignored.

<u>TAB,8,12</u>

If the following COBOL line is entered with the tabulation key pressed where the mark ! is shown:

<u>00100!FD!FILEA</u>

the entry is positioned as follows in the text buffer:

```
        |     |1                                        |7  |7
        |8    |2                                        |3  |6
--------+-----+-----------------------------------------+---+-----
        |FD   |FILEA                                    |PRG|00100
```

DELETE DIRECTIVE

In addition to the codes which edit lines, SETUP includes a directive to delete an entire line or sequence of lines from the text buffer. The user enters:

`DELETE/n`

The number of the line to be deleted is specified by n; other legal delimiters: space, comma, or period, may be substituted for the slash.

A sequence of lines may be deleted by entering:

```
DELETE/n/m
```

The lines from n through m are deleted from the buffer; other legal delimiters may be substituted for the slashes.

A line may be deleted also by entering the line number of an existing line with no characters following it.

Examples:

To delete lines 10 through 100:

```
DELETE,10,100
```

To delete line number 345:

```
DELETE/345
```

or enter:

```
345
```

Single characters or entire lines may be corrected and/or deleted prior to entering a line or command into INTERCOM. See the Error Correction descriptions under Teletype and Display Terminals in section 2.

FILE STORAGE AND MANIPULATION

SAVE DIRECTIVE

To save the contents of the text buffer on a mass storage device, the user enters:

```
SAVE
```

The file in the text buffer is saved as a regular SCOPE file. Its name is that specified by the user in the NEW directive. If a file with the same name already exists on mass storage, contents of the text buffer replace the file on the mass storage device. A file that has been saved may be recalled into the text buffer with the directive OLD.

The SAVE directive does not create permanent files; they are created by the STORE command (section 3) or with the SCOPE control card, CATALOG. It creates user private files which are listed by the command FILES, and counted when the user enters ASSETS.

UNSAVE DIRECTIVE

A file may be removed from mass storage by entering:

```
UNSAVE
```

The system responds:

```
OLD FILE NAME--
```

The user then enters the name of the file to be removed. Contents of the text buffer are not affected by the UNSAVE directive. The directive SCRATCH is used to erase the contents of the text buffer. The deleted file will no longer appear as a user private file when the FILES command is entered. Permanent files are dropped with the SCOPE control card, PURGE or the INTERCOM command, DISCARD (section 3).

ABBREVIATED ENTRY

```
UNSAVE/filename
```

RENAME DIRECTIVE

A file in the text buffer can be renamed by entering:

```
RENAME
```

The system responds:

```
NEW FILE NAME--
```

The new name of the file is entered; it must not be the name of an attached permanent file. Further, if the new name entered is the same as an existing private file, a subsequent save operation will cause the old file to be overwritten; it will no longer be available to the user.

If the contents of the text buffer were previously saved on mass storage, only the text buffer file is renamed; the mass storage file is not affected by this directive.

ABBREVIATED ENTRY

```
RENAME/filename
```

Example:

A file named AFILE has been constructed in the text buffer:

```
SAVE
  .
  .
  .
ØLD/AFILE
```

The user saves the file and later recalls it for further manipulation. He may then rename the altered file and save it:

```
RENAME/BFILE

READY.
```

The new BFILE is stored on a mass storage device; AFILE is still stored on a mass storage device in its original form and may be removed with:

```
UNSAVE/AFILE

READY.
```

LIST DIRECTIVE

Contents of the text buffer can be listed, entirely or in part, with the list directive:

`LIST`	Lists current contents of the buffer in line number sequence.
`LIST/n`	Lists line n.
`LIST/n/m`	Lists lines n through m.

The slashes may be replaced by any of the other legal delimiters: comma, period or a space.

The list is printed as the program was entered by the user and not as SETUP reformats each line in the text buffer.

TERMINATION OF LIST

Printing of the list may be stopped by pressing:

CTRL Z or ESC	}	S	on the Teletype keyboard
%	}	S	on the display keyboard

The current output being sent to the terminal is lost. Printing will resume automatically if there is additional output to be sent to the terminal. Therefore, if the list is long, the user may have to enter the interrupt followed by S more than once.

Examples:

```
LIST
  100  PRØGRAM A(ØUTPUT)
  120  PRINT 2
  167  2  FØRMAT(* HI  INTERCØM*)
  190  END


LIST/190
  190  END
```

SCRATCH DIRECTIVE

Current contents of the buffer may be erased by entering:

```
SCRATCH
```

The system responds:

```
ENTER NEW OR OLD FILE
READY.
```

The user may then enter the NEW or OLD directive to declare a file.

Any file on mass storage corresponding to the file in the text buffer is not affected. Use UNSAVE to remove a file saved on mass storage.

COMPILATION AND EXECUTION

When the user is satisfied with the contents of the text buffer, he submits his file to the compiler for processing. Directives available to him depend on the programming language: BASIC, FORTRAN RUN, FORTRAN Extended, or COBOL.

INTERACTION

During execution of BASIC, FORTRAN, or COBOL programs, the terminal user can interact with the executing program (a detailed description is contained in section 6). He can receive output at the terminal and also input to the executing program from the keyboard. If a Teletype terminal has a paper tape punch and reader, a program can call for input from paper tape, or a program can be punched on paper tape with pauses (X-OFF) for input from the keyboard.

When a program is executing, only the files, INPUT and OUTPUT are connected to the terminal automatically. Any other files used for interaction with programs must be connected to the terminal with the CONNECT command and disconnected, when interaction is finished, with the DISCONT command (section 3).

The executing program will pause when input is expected. It is helpful if the program contains a signal to the terminal user that input is expected and indicates the type of input. For instance, a PRINT statement can be included just prior to a READ statement in a FORTRAN program. In COBOL programs, the message ENTER COBOL INPUT- is displayed at the terminal whenever the ACCEPT FROM statement expects input from the terminal. The BASIC INPUT statement displays a question mark at the terminal to indicate that user input is expected.

COMPILER CALLS

The following compiler calls are available to the INTERCOM user in the form of SETUP directives:

RUN, RUNER, and RUNX call the FORTRAN RUN compiler

FTN and FTNER call the FORTRAN Extended compiler

RUN and BASICX call the BASIC compiler

COBOLER and COBOLX call the COBOL compiler

Of these compiler calls, all but RUNX, FTN, COBOLX, and RUN when used to compile a FORTRAN program, produce an error listing automatically.

RUN DIRECTIVE

RUN is a versatile directive. The user writing a program in either BASIC or FORTRAN RUN can compile the contents of the text buffer with subsequent execution simply by entering:

```
RUN
```

The program in the text buffer is compiled by the BASIC or FORTRAN compiler depending on the current system mode. If compilation is error free, the program is executed.

If errors occur in a BASIC compilation, BASIC provides a directory of the compilation errors. A FORTRAN directory is not produced by this directive.

Examples of RUN:

The system mode is BASIC.

```
READY.
ØLD/BASTEST

READY.
RUN
PRØGRAM TRANSFERRED TØ CØMPILER
FØR WITHØUT NEXT AT 40
 09.25.45.BASIC CØMPILATIØN ERRØRS

READY.
```

The BASIC program is not executed because of an error in line 40. The user should use LIST/40 to display the line and then enter the necessary correction and call RUN again.

The system mode is FORTRAN.

```
10 PRØGRAM FILA(ØUTPUT)
20 PRINT 1
30 1 FØRMAT(* TEST FILA*)
40 END
SAVE

READY.
RUN
PRØGRAM TRANSFERRED TØ CØMPILER
 00.42.37.END           FILA
TEST FILA

READY.
```

The program file FILA is compiled and executed. The output line TEST FILA is displayed at the terminal.

BASIC COMPILE AND EXECUTE

To compile and subsequently execute a BASIC program, the user enters either RUN or:

```
BASICX
```

There is no difference between calling BASICX and RUN. When the system is BASIC, either will call the BASIC compiler.

The contents of the text buffer are compiled; and if no compilation errors occur, the compiled program is executed.

If errors are found, a complete error listing is printed at the terminal. It includes the SETUP line number where the error occurred and a description of the error. The user may correct the error and re-enter the BASICX directive.

Example:

The system is BASIC, the file is named FILEB with the NEW directive, and defined as follows:

```
10 PRINT "TYPE A NUMBER";
20 INPUT X
25 IF X=0 THEN 80
30 F=1
40 FØR I=1 TØ X
50 F=F*I
55 NEXT I
60 PRINT "FACTØRIAL"X;"IS "F
70 GØ TØ 10
80 END
SAVE

READY.
BASICX
PRØGRAM TRANSFERRED TØ CØMPILER
TYPE A NUMBER ? 3
FACTØRIAL 3 IS 6
TYPE A NUMBER ? 0

READY.
```

FORTRAN RUN COMPILE AND EXECUTE

Two directives in addition to RUN submit a program to the FORTRAN RUN compiler: RUNER and RUNX.

RUNER

The program in the text buffer will be compiled and subsequently executed if no compilation errors are found. Otherwise, an error summary is printed at the terminal. For a compilation without an error listing, the user enters:

RUNX

The program will be compiled and subsequently executed if no compilation errors are found. If there are errors, an error listing can be generated by entering the ERRORS command (section 3). RUNX should be employed only when the user is fairly certain his program is error free.

FORTRAN EXTENDED COMPILE AND EXECUTE

Two directives call the FORTRAN Extended compiler: FTNER and FTN. The directive FTNER is similar to RUNER for FORTRAN RUN. The user enters:

FTNER

It will execute the program only if no compilation errors are found. Otherwise, a complete error summary is printed automatically at the terminal.

For a compilation with no error list, and subsequent execution, the user enters:

```
FTN
```

This directive calls the FORTRAN Extended compiler. Contents of the text buffer are compiled and executed if compilation is error free. No error list is produced; however, if errors occur, the user may request an error listing with the ERRORS command (section 3).

Example:

In the following example, RUNER could be entered instead of FTNER if the FORTRAN RUN compiler were wanted.

```
CØMMAND- SETUP.
ØN AT      10.15.45.
SYSTEM--FØRTRAN
NEW ØR ØLD FILE--NEW/FØRTX

READY.
100 PRØGRAM FØRTX)INPUT,ØUTPUT)
103 1 PRINT 2
106 2 FØRMAT(* ENTER VALUE FØR A - *)
109 READ 3,A
112 3 FØRMAT(F10.5)
115 SQ-A:A
118 PRINT 3,SQ
121 GØ TØ 1
123 END
SAVE

READY.
FTNER
PRØGRAM TRANSFERRED TØ CØMPILER
10.19.32.          1  FØRTRAN ERRØRS IN  FØRTX
  PRØGRAM FØRTX
CARD NØ.  SEVERITY          DIAGNØSTIC
FØR00115        SQ-A:A
      06     FE             UNRECØGNIZED STATEMENT
10.19.55.CARD CALL LØADER FATAL ERRØR
10.20.01.NAME GREATER THAN 7 CHARACTERS
10.20.01.CARD/IMAG/ERRØRS IN FTN CØMPIL
CØNTRØL CARD ERRØR

READY.
```

If RUNER had been entered instead of FTNER, the following would have been output:

```
PRØGRAM TRANSFERRED TØ CØMPILER
10•19•32•000001  FØRTRAN ERRØRS
100 PRØGRAM FØRTX(INPUT,ØUTPUT)
115 SQ=A:A
FM*** UNRECØGNIZABLE STATEMENT (FATAL)
10•19•50•CARD CALL LØADER FATAL ERRØR
10•19•50•NAME GREATER THAN 7 CHARACTERS
10•19•50•CARD IMAG/ERRØRS IN RUN CMP•
CØNTRØL CARD ERRØR

READY•
```

User corrects line 115 and enters it in program with SAVE.

```
115 SQ=A*A
SAVE

READY•
FTNER
PRØGRAM TRANSFERRED TØ CØMPILER
ENTER VALUE FØR A - 2•5
6•25000
ENTER VALUE FØR A - 5•0
25•00000
ENTER VALUE FØR A - 9•3
86•49000
ENTER VALUE FØR A - A
```

An interrupt command followed by A stops execution and returns user to SETUP. The system responds:

```
USER ABØRT

READY•
```

COBOL COMPILE AND EXECUTE

Two directives call the COBOL compiler: COBOLER and COBOLX. The user calls the COBOL compiler to compile the contents of his text buffer with:

```
COBOLER
```

If compilation errors are found, an error summary list is produced automatically and displayed at the terminal.

A COBOL program in the user's text buffer may be compiled and subsequently executed by typing:

```
COBOLX
```

If no compilation errors occur, the program is executed. If errors prevent execution, a comment is printed at the terminal but no error list is produced. The user can request an error list by entering the ERRORS command (section 3).

Example:

```
C0MMAND- SETUP.
         .
         .
         .
SYSTEM/GENERAL/NEW/FILEC

READY.
TAB,8,12

READY.
100!IDENTIFICATI0N DIVISI0N.
110!PR0GRAM-ID. TEST TERMINAL.
200!ENVIR0NMENT DIVISI0N.
210!C0NFIGURATI0N SECTI0N.
220!S0URCE-C0MPUTER. 6600.
230!0BJECT-C0MPUTER. 6600.
240!SPECIAL-NAMES.
250!!TERMINAL IS TTY.
300!DATA DIVISI0N.
340!W0RKING-ST0RAGE SECTI0N.
350!77 DATA-IN PICTURE X(80).
360!77 DATA-0UT PICTURE X(80).
400!PR0CEDURE DIVISI0N.
410!START.
420!!ACCEPT DATA-IN FR0M TTY.
430!!M0VE DATA-IN T0 DATA-0UT.
440!!DISPLAY DATA-0UT UP0N TTY.
450!!ST0P RUN.
SAVE

READY.
C0B0LER
PR0GRAM TRANSFERRED T0 C0MPILER
  ENTER C0B0L INPUT          A B C D E F G
A B C D E F G

READY.
```

STOPPING EXECUTION

At any time, the user may stop execution of a program and return to SETUP.

At a Teletype terminal, he presses the CTRL Z, ESC, or ALT MODE key followed by A; at a display terminal, he presses the % (SHIFT 5) key followed by A:

```
%A
USER ABORT
READY.
```

The program currently executing terminates, and INTERCOM returns to SETUP. Contents of the text buffer are not destroyed by the user abort of an executing program.

Caution should be exercised, as it is possible for a program to have completed before all its output has been sent to the terminal. If this condition exists when a user abort request is entered, the SETUP utility is aborted.

If the user aborts SETUP by accident, he may retrieve his text buffer only if he had previously submitted it for compilation and/or execution. In this case the text buffer is saved as the file SETFILE and may be retrieved by entering:

```
OLD/SETFILE
```

The file may then be renamed and saved, as a user private file.

INTERACTIVE USER PROGRAMS 6

Interactive communication can take place between a user and his executing program. The program may be initiated by SETUP directives, the EDITOR RUN command, or SCOPE control cards requesting execution of the program. Terminal communication may be done through ALGOL, BASIC, COBOL, COMPASS, and FORTRAN programs.

TERMINAL INTERACTION

To facilitate interactive program execution, the user connects mass storage files to the remote terminal. Files may be connected and subsequently disconnected by CONNECT and DISCONT commands entered at the terminal, by calling connect or disconnect routines from the user's program, or with the EDITOR RUN command or SETUP compiler calls. The SETUP utility does not provide a directive to call either the ALGOL compiler or the COMPASS assembler; for interactive execution of programs in these languages, the EDITOR RUN command or SCOPE control cards must be used.

For the inexperienced user, the EDITOR RUN command is simple and easy to use. This command (as well as the SETUP compiler calls) connects the files INPUT and OUTPUT to the terminal automatically. The user, however, may connect and disconnect files from his program or connect files other than INPUT and OUTPUT by using the subroutines provided in the FORTRAN RUN, FORTRAN Extended, and ALGOL compilers. These subroutines allow the user to change the routing of information for I/O requests on mass storage files between the central site device and the remote terminal.

Once a file is connected, any input or output on that file is routed to or from the user's terminal until the file is disconnected. An output statement on a connected file will send the output to the terminal; an input statement on a connected file will await entry of input from the terminal keyboard. Because the INTERCOM system does not indicate when input is expected from the terminal, it is helpful if the user's program sends a prompting message. An output statement containing this message may be included before each input statement in the user's program. For BASIC programs, a question mark is sent automatically to the terminal for each INPUT statement. For COBOL programs, the message ENTER COBOL INPUT is sent automatically to the terminal for each ACCEPT FROM statement.

Any private file, including attached permanent files, may be connected to the terminal. If no file of the name specified exists, one is created on execution of the connect request. The only limit to the number of files that can be connected at one time, is the user's maximum file limit. Duplicate connects or disconnects are allowed but have no effect on the file, and a connect-disconnect sequence will not alter the contents or positioning of the original file. Only coded input/output requests should be issued on connected files; however, binary input/output requests will be treated the same as coded requests.

Programs compiled for interactive execution which contain connect and disconnect requests may be run also in batch jobs without recompilation. Input/output is routed to and resides on allocatable mass storage.

CARRIAGE CONTROL

Carriage control characters are required to control the spacing of data routed to the terminal. The user should supply a carriage control character as the first character of each terminal output line issued from any program (except BASIC, which supplies them automatically). If the character is omitted, results are unpredictable. The carriage control characters are described in appendix D.

CAPABILITIES

The following sections describe the capabilities of each interactive language.

ALGOL

ALGOL-60 programs and the terminal communicate through connected files using the following input/output procedures:

INPUT	OUTPUT
INREAL	OUTREAL
INARRAY	OUTARRAY
INCHARACTER	OUTCHARACTER
INLIST	OUTLIST

If the EDITOR RUN command is used, the files INPUT and OUTPUT are connected automatically; channel number 60 is equated to any of the above input procedures to receive data from the terminal, and channel number 61 is equated to an output procedure to send data to the terminal.

To use a file on a channel other than 60 or 61, the file must be assigned a channel number in a CHANNEL definition. Definitions are entered in the format of CHANNEL cards. Assuming that INPUT and OUTPUT are connected to the terminal, ALGOL will log the standard definitions and await input of CHANNEL definitions from the terminal. All channels to be utilized must be entered and acknowledged at this point. Input of CHANNEL definitions terminates with entry of CHANNEL,END or a non-recognizable definition. In the latter case, the entry is assumed to be the first data line and is backspaced over to be read subsequently by the program.

Any mass storage file may be connected or disconnected with the procedures CONNEC and DISCON. These procedures require specification of one parameter:

CONNEC(n);DISCON(n)

n is an integer value corresponding to an existing channel.

The specified channel must have a formatting area, therefore, the CHANNEL definition may not contain the A parameter. The Rt parameter should be specified, but if omitted, the system assumes an R-type channel. Output channels should be defined as paged (PPs parameter). For input channels, paging should not be specified or entered as PP0. The user should avoid connecting a file which has previously been used for mass storage input as this may cause data to be lost.

The following features of ALGOL input/output should be noted: Because a user logical record (line) may be built up character by character over several procedure calls, on output to the terminal it is necessary to use a line feed to ensure termination and transmission of an output message. Thus, a partial line on channel n will not be transmitted until OUTPUT(n,'/') or equivalent is encountered. Similarly on input (particularly for free format routines such as INREAL) not every procedure call will cause a wait for input on the terminal. If the last line entered for that channel has not been exhausted by the program, the new input will process the contents of that line before waiting for further input.

BASIC

The user at a terminal communicates with a BASIC program through the PRINT and INPUT statements designed specifically for terminal input/output. In addition, the file input and output statements, PRINT FILE and INPUT FILE, may be used for terminal input/output: the specified files must first be connected to the terminal.

BASIC displays a question mark at the terminal on encountering an INPUT statement, signalling the user that his program is waiting for terminal input. It is helpful if the user precedes his statement with a PRINT statement containing a brief description of the type of input required. For batch mode processing, BASIC programs may not contain INPUT statements; and output normally routed to the terminal is directed to the file OUTPUT for subsequent printing.

The BASIC language does not provide a method for connecting and disconnecting files from within a program. To use a file other than OUTPUT for terminal communication, the user must connect the file with the CONNECT command before program execution. Normally, the file OUTPUT is used for both terminal input and output when the EDITOR RUN command is specified.

COBOL

The Procedure Division verbs, ACCEPT and DISPLAY, provide communication between COBOL programs and the terminal user. These verbs are specified in the form:

ACCEPT identifier FROM mnemonic-name

DISPLAY identifier UPON mnemonic-name

where identifier can be an elementary or group item and mnemonic-name is equated with the terminal in the TERMINAL IS clause in the SPECIAL-NAMES paragraph of the Configuration Section in a COBOL program.

Subroutines are not provided for connecting and disconnecting files from COBOL programs.

If execution diagnostics are to be displayed at the terminal, the printer file must be declared as OUTPUT.

COMPASS

Requests for terminal input and output from a COMPASS program are made through calls to Circular Buffer I/O (CIO) either directly or through the CPC subroutine. Prior to making a CIO call, the user must have created an FET for the file to be used for terminal input/output. This file should be connected to the terminal so that the device type in the SCOPE file table specifies a remote terminal. To verify that the file is connected, the user may issue an OPEN macro to return the file's device type to the FET. The file is connected if device type is 61 octal.

To connect or disconnect a file, the user may issue CONNECT or DISCONT commands, or he may call the peripheral processor routine CON from his program. All requests from the user to the system are made through relative location 1 of the user's program (absolute location RA + 1). The call to CON follows:

In RA + 1

bits 59-42	CON in display code
40	Auto Recall bit
35-24	ZERO = connect; non-zero = disconnect
17-0	Location of parameter word X

In RA + X

bits 59-18	File name (left justified, with zero fill)
1	Set if CON detects an error, such as an illegal file name
0	Completion bit

Requests for READ and WRITE on connected files may be issued directly, or through SCOPE system macros. Input and output requests should be in coded mode. Bit 1 of the code-and-status field in the FET must be set to zero when the input/output request is made. This bit specifies the mode of the file. If a binary operation is requested, it will be treated the same as a coded request.

The circular buffer provided for terminal input should be at least as large as the longest input line expected. For a display terminal, the input line may be up to 105 words (one screenful). For a teletypewriter, the line length is not limited; the user must estimate the longest expected line length. On input, if a line is larger than circular buffer limits, an error code of 10B is returned in the code-and-status field of the FET, and the data is transferred until the circular buffer is full; remaining data is discarded.

A READSKP function may be specified which results in similar action but no error code is returned.

When coding in COMPASS, the user may have no more than one input/output request outstanding for the terminal; otherwise, data may be lost or input/output operations never may be completed. This restriction applies only to input/output requests, not to the number of files which may be connected simultaneously.

Data received from a terminal is formatted in display code character representation; The data will be left justified and blank filled as required to take up an integer number of central memory words, including the 12-bit zero byte end-of-line indicator.

Data sent to a terminal must be formatted in display code character representation. The first character position of each line is the carriage control character. The end of a line is indicated by a 12-bit zero byte in bits 0 to 11 of a central memory word.

FORTRAN

Terminal input and output from either FORTRAN RUN or FORTRAN Extended programs is accomplished by issuing formatted or NAMELIST input/output statements on connected files. If the SCOPE files INPUT and OUTPUT have been connected, FORTRAN READ, WRITE, and PRINT statements may be used for terminal communication. If the SETUP compiler call directives or the EDITOR RUN command are used, the INPUT and OUTPUT files will be connected automatically. The user should specify these files on the PROGRAM card.

The user may connect and disconnect files from his program with calls to the CONNEC and DISCON subroutines. A file can be connected to the terminal with the statement:

```
CALL CONNEC (lfn)
```

The file is disconnected with the statement:

```
CALL DISCON (lfn)
```

lfn is a file name parameter: a tape logical unit number 1 to 99, a Hollerith constant in the format nLfilename, or an integer variable containing either of the preceding forms.

Mass storage files may be connected or disconnected from a FORTRAN program any number of times, and at any time. Input files, however, must be connected while they are at a file boundary; or data may be lost.

Other FORTRAN input/output requests, such as BUFFER statements, may be used for terminal communication; the mode of transmission must be coded. A BUFFER statement may not be used for input from a terminal, because an end-of-record is expected.

On the PROGRAM card, the user must declare all files which are connected, disconnected, or referenced by READ and WRITE requests. If execution diagnostics are to be sent to the terminal, the file OUTPUT must be declared on the PROGRAM card and connected to the terminal.

REMOTE BATCH PROCESSING 7

GENERAL DESCRIPTION

A user at a 200 USER terminal or a display terminal with a card reader and line printer can submit jobs to the central computer for processing via the card reader and receive any subsequent output on the line printer. This capability is the same as that provided by the EXPORT/IMPORT system at a 200 USER terminal. It gives the remote user the same capability to submit jobs and receive output as if he were at the central site, except that he may not transmit binary data; only Hollerith data may be processed through the remote display terminal.

The commands described in this section are not legal at a teletype terminal or when a 200 USER terminal is operating under control of SETUP or EDITOR. If an attempt is made to enter batch processing commands from a teletype terminal, INTERCOM sends the message: COMMAND NOT ALLOWED FROM TTY. If a batch processing command is entered while the user is in SETUP, INTERCOM replies: ILLEGAL COMMAND WHILE IN SETUP. If a batch processing command is entered while the user is in EDITOR, a similar message is displayed.

INITIATING BATCH PROCESSING

After the user has logged in with the LOGIN command, and INTERCOM has returned the message, COMMAND, he can enter at his terminal keyboard any of the following remote batch processing commands:

IMPORT	CONTIN	OUTPUT
READ	GO	PRIOR
END	H	DROP
AGAIN	B	SUSPEND

With the exception of IMPORT, these commands are described later in this section. IMPORT is included in INTERCOM for compatibility with EXPORT/IMPORT.

PROCESSING JOBS

The remote terminal user initiates job processing by loading the card reader with cards and entering the READ command to cause INTERCOM to start reading the job. INTERCOM creates an input file and gives it the name of the job and a disposition code indicating the job came from INTERCOM.

As a complete job is read, its input file is placed in the SCOPE batch input or dependency queue. Processing is exactly the same as for input jobs submitted at the central site. A job in a dependency queue must wait for completion of a prerequisite job before it can be processed; a job in an input queue waits for sufficient resources at the central site for the job to be processed.

All output files generated by a job are given the job name and a remote disposition code associating the job with the terminal before they are placed in the output batch queue. The remote disposition code reserves the output files for INTERCOM and prevents further processing by SCOPE. Files to be printed are directed by INTERCOM to the correct terminal. Other types of remote output files are redirected by INTERCOM to be output at the central site by SCOPE.

Any job associated with the terminal can be interrupted or otherwise controlled by the terminal user. He may enter an INTERCOM command or a command to control operation of the terminal. For instance, he can request a status display of a job in process, suspend or restart printing of an output file, request that a job be dropped, and so forth. A job is considered to be associated with a terminal when it has been submitted from the terminal or is an output file produced by a job submitted from the terminal.

The user returns from remote batch processing mode to command mode upon successful completion of, or acceptance of the remote batch command. If cards are being read or output is being printed, he should press the INTER key and then the CLEAR key before entering the command.

When the remote job has been read into the SCOPE batch queue, the user may LOGOUT without losing any files associated with the job. He may then LOGIN with the same user name/password combination and retrieve the output files from his job.

BATCH PROCESSING COMMANDS

The display keyboard is used to enter commands that control input, execution, and output of jobs submitted at the terminal. If reading or printing is not in progress at the terminal, a command may be entered by typing it on the keyboard and then transmitting it with the SEND key.

When data is being read on the card reader or written on the line printer the keyboard is locked to prevent the data from being overwritten. The user may interrupt reading or writing by pressing the INTER key and waiting until the transmission in progress is completed and the keyboard unlocks. He then presses the CLEAR key and enters a command.

COMMAND SYNTAX

Unlike other INTERCOM commands, the remote batch commands may be abbreviated to their initial letter; only this abbreviation is acceptable. Optionally, the command may be terminated by a period. In any case, the command is transmitted to the INTERCOM system by pressing SEND. Pressing SEND displays ▲ on the display screen and transmits the information preceding the ▲ to INTERCOM.

When a command requires one or more parameters, the parameters are separated from the command word and from each other by a comma or blank.

If a command is not recognized or the format is incorrect, one of the following messages is displayed:

```
FORMAT ERROR    or    CONTROL CARD ERROR
```

The incorrect command remains on the screen so the user can determine the error. The corrected command or another command should be entered; normal operation of the terminal will not resume until a correct command is entered.

JOBNAME PARAMETER

Commands that reference jobs at the central site (DROP, OUTPUT and PRIOR) use a jobname parameter. Jobname is the 7-character name assigned to the job; it will reference the job as an input file, in execution, or as an output file created by the job. The user assigns his job a 1-7 character name on the job card; SCOPE assigns the last two characters of the job name as a unique identifier for each job each time it is run. If more than five characters have been

used for a job name, SCOPE replaces any characters in excess of five with the system identifier. If less than five characters were used, SCOPE fills the character position with zeros up to the last two characters. Since the last two characters are unique, they may be used as a legal abbreviation for the job name. For instance, the DROP command may be entered as:

```
DROP,jobname
```

where jobname is the complete 7-character job name, or it may be entered as:

```
DROP,jj
```

where jj are the last two characters assigned by SCOPE to the jobname.

If the designated job does not exist or is not associated with the terminal, this message is displayed:

```
FILE NOT FOUND
```

The command may then be entered with a correct jobname or another command may be entered. Some command must be entered before the terminal will resume normal operations.

With the H command the user can request a display at the terminal of his jobs in the input, dependency, or output queues. The user's jobs in execution will be displayed at the terminal when the B command is entered. (These commands are described in this section under the heading Terminal Displays.)

READING CARDS

Before jobs may be read from the terminal, the card reader must be turned on, the card buffer cleared, and the cards placed in the hopper and registered. (Complete instructions for card reader operation are given in section 2.) Then the user enters the READ command or its abbreviated form:

```
READ   or   R
```

An input file is created for the job and the cards are read into this file; the first card must be a valid job card. The input file for the job is placed in the SCOPE input or dependency queue.

More than one job can be stacked in the card reader for continuous reading with no operator intervention. Each time an end-of-file card (6/7/8/9) is encountered, the job it terminates is placed in the input or dependency queue for SCOPE processing. After the end-of-file card, the next card must be another end-of-file card or a valid job card; otherwise, reading stops. Extra end-of-file cards are ignored if no other cards separate them.

Another form of the READ command allows the user to create a private file from a card deck. He enters:

```
READ,filename   or   R,filename
```

When READ,filename is entered, all the cards in the reader are read into the named file. The filename must be a valid SCOPE file name. The first card is not treated as a job card. Transmission to the file terminates when the card reader is empty at an end-of-file card. The named file becomes a private INTERCOM file accessible to the terminal user.

If no read errors occur, card reading continues until the hopper is empty. If the last two cards read are not end-of-file (6/7/8/9) cards, or an end-of-file card followed by a blank card, INTERCOM expects more cards and will display the message:

```
READER NOT READY
```

The user may then place more cards in the reader, press the LOAD button to start the reader and enter a READ command or GO command.

If reading is initiated while printing is in process on the line printer, printing and reading will proceed in an interleaved manner.

READ ERRORS

When READ,filename is specified, no checks are made on the validity of the cards; they are read until the last card in the hopper is an end-of-file card.

When READ without a file name has been specified and the first card read (or the first card after an end-of-file card in a multiple read operation) is not a valid job card, the card reader stops and the following message is displayed:

```
JOB CARD ERROR
```

The job card must be corrected and reading restarted. Enter END,CR or E,CR before restarting the reader with a new deck.

As each card is read, it is checked to ensure that it contains only valid characters. If an illegal card punch code is detected, the input file is dropped, the read operation is suspended until the MAN/REL button is pressed, card reader is in ready state, LOAD button is pressed, and READ or CONTIN is entered from the keyboard.

The erroneous card will be among the last 24 cards read and any cards in error will be among the 12 cards displayed on the terminal screen. After corrections are made, card reading should be resumed from the first card displayed.

An uncorrectable parity error in the input file (or other mass storage device trouble) will also terminate the read operation. The input file is dropped and this message is displayed:

```
INPUT FILE ERROR
```

The user should restart the reading process from the beginning of the job.

Binary cards may not be read through the terminal card reader. If an attempt is made to read binary cards, the terminal will appear to be disconnected; to restart terminal operation, the user should press the MAN/REL button on the display console.

TERMINATING READING

Card reading can be terminated before the job is completely read by pressing the INTER key, waiting until reading stops and then typing the command:

```
END,CR   or   E,CR
```

The input file is dropped.

Reading may be restarted by following the procedures to start the card reader, clear the card buffer, register the cards and then enter the READ command. Reading will not restart with the GO command.

If card reading is not in progress, the END,CR command has no effect.

PRINTING OUTPUT

Before jobs can be printed, the terminal line printer must be turned on. (See section 2 for a complete description of printer operation.) The printer is ready when both the POWER ON and START buttons are lit.

When his job is ready to be printed, the job name and sequence number appear in the output queue in the H, O display (see H display). He then enters the GO Command and printing begins. Two banner pages precede the actual output from the file to signal the start of printing and to separate consecutively printed files. As each line is printed, it also is displayed on the screen. If a print file becomes available while input is being read, the two operations are interleaved.

The user can control the printing with carriage control characters. When the first two characters of any print line are PM, that line is not printed; but it is displayed on the screen. PM indicates printer message, and the rest of the print line may be used to request particular operator action, such as changing the print form. Printing stops until the requested action is completed, and printing is resumed by entering the GO command.

Other carriage control characters are:

1	Eject to top of page before printing
0	Double space before printing
+	Suppress space before printing
blank	Single space before printing

The user can eject paper manually to the top of the page by pressing the printer button STOP followed by PAGE EJECT. Then the START button must be pressed to ready the printer.

If the printer is not initially ready or if it leaves the ready state during printing, the following message is displayed:

```
PRINTER NOT READY
```

The condition that stopped printing should be corrected and the START button pressed to ready the printer. To continue printing, enter:

```
GO   or   G
```

If the printer stops but no message is displayed, usually it will resume printing automatically.

PRINTING ERRORS

If an uncorrectable parity error is encountered in the file being printed, this message is displayed:

```
OUTPUT FILE ERROR
```

The user should enter GO to continue printing. The system will attempt to continue printing; but if the parity error persists, the message will be displayed again each time GO is entered. Either the file must be repositioned, or printing must be terminated.

REPRINTING CURRENT OUTPUT

An output file being printed can be rewound and restarted from the beginning by entering the command:

```
AGAIN   or   A
```

Also, a print file may be backspaced, and printing restarted from some point other than the beginning of the file by entering:

```
AGAIN,n   or   A,n
```

where n specifies the amount to backspace the file. n is a positive octal integer less than or equal to 777; it indicates the number of sectors the file is to be backspaced according to the following formula:

sectors = 10 x n (values are octal)

If the beginning of the file is reached before the specified number of sectors have been backspaced, printing will restart at the beginning of the file.

AGAIN has no effect on the operation of INTERCOM if printing is not in progress.

Examples:

To halt printing and resume again from the beginning of a file:

```
AGAIN
```

To halt printing, backspace the file 30(octal) sectors, and resume printing from that point:

```
A,3
```

SUSPENDING PRINTING

Printing in progress can be halted temporarily by entering:

```
SUSPEND   or   S
```

Printing will halt immediately, but the print buffer and the current file position will be preserved unless an AGAIN command or END command is entered.

If printing is not in progress when SUSPEND is entered, no new jobs will be printed and no print buffers will be assigned, even if output for printing becomes available.

Once printing is suspended, additional SUSPEND commands have no effect.

CONTINUING PRINTING

The effect of a SUSPEND command may be negated with the CONTIN command:

```
CONTIN   or   C
```

If SUSPEND halted the printing of a file, the file will continue printing from the point where it stopped. If no file was being printed when SUSPEND was entered, CONTIN will ensure that printing is initiated as soon as a print file becomes available.

The CONTIN command has no effect if a SUSPEND command was not previously entered.

Example:

To suspend printing and then continue:

```
SUSPEND
C
```

TERMINATING PRINTING

Printing can be terminated while it is in progress by entering:

```
END,LP   or   E,LP
```

The remainder of the file being printed is bypassed and the dayfile, which is the last logical record of the print file, will be printed. To stop all printing of the file and drop the print file, a second END,LP command must be entered while the dayfile is being printed. If no printing is in progress, the END,LP command has no effect.

RESUMING OPERATIONS WITH GO

After the printer has halted for any reason except SUSPEND, or the card reader has halted and READER NOT READY is displayed, or the user has interrupted transmission for any reason and wants to return to the batch processing mode, he enters the command:

```
GO   or   G
```

This command should be entered to resume operations following:

PRINTER NOT READY message

PM print line

OUTPUT FILE ERROR message (parity error on print file)

READER NOT READY message

Message from central site operator

The terminal will continue any reading or printing previously in progress. If reading or printing had been terminated prior to the GO entry, no further transmission will occur. The GO command may be entered at any time.

TERMINAL DISPLAYS

The user may request a display of the contents of the input, dependency and output queues, or a display of the current jobs in execution at each control point. If reading or printing is in progress, the user must press the INTER button before entering the request for a display.

H DISPLAY

To display the input queue, enter:

```
H,I
```

To display the dependency queue, enter:

```
H,D
```

To display the output queue, enter:

```
H,O
```

Only files associated with the user appear in an H display. The input and output files are listed in order of descending priority (figures 7-1, 7-2); dependent files are arranged alphabetically by their dependency identifier (figure 7-3).

The screen display is updated automatically every 9 to 11 seconds.

If the commands OUTPUT, DROP or PRIOR are entered, the display will be interrupted; but it will be regenerated automatically after the command is processed. Any other command will terminate the display. Also, the display will terminate because of error messages and messages from the operator to the central site.

If the H command interrupts printing, the user should enter GO or G to resume printing after the display.

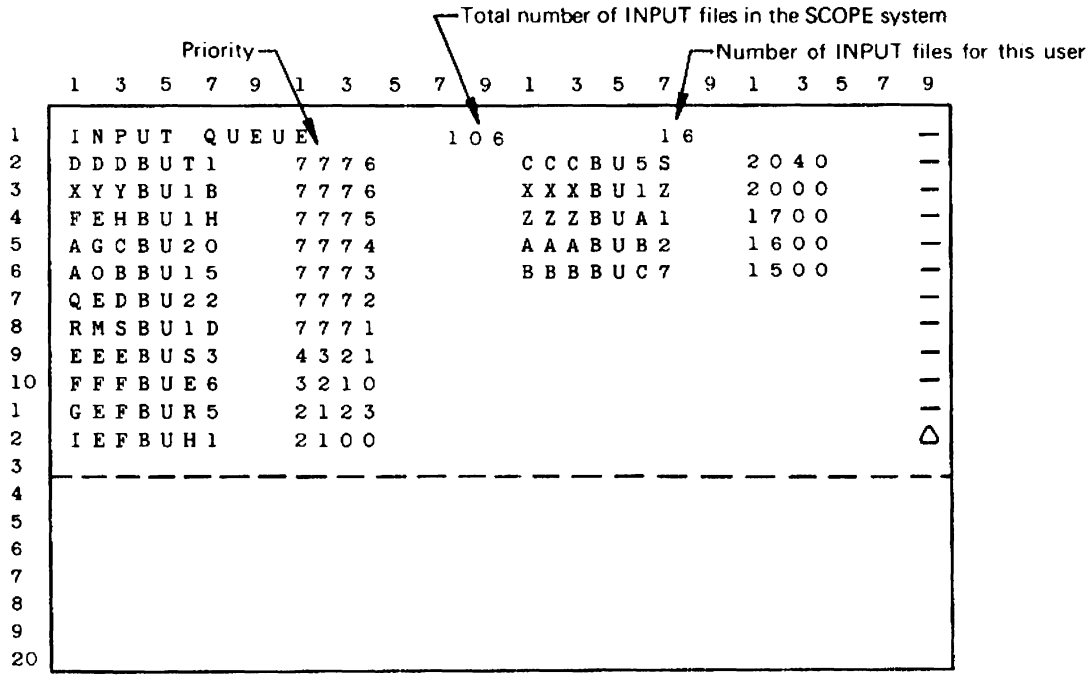
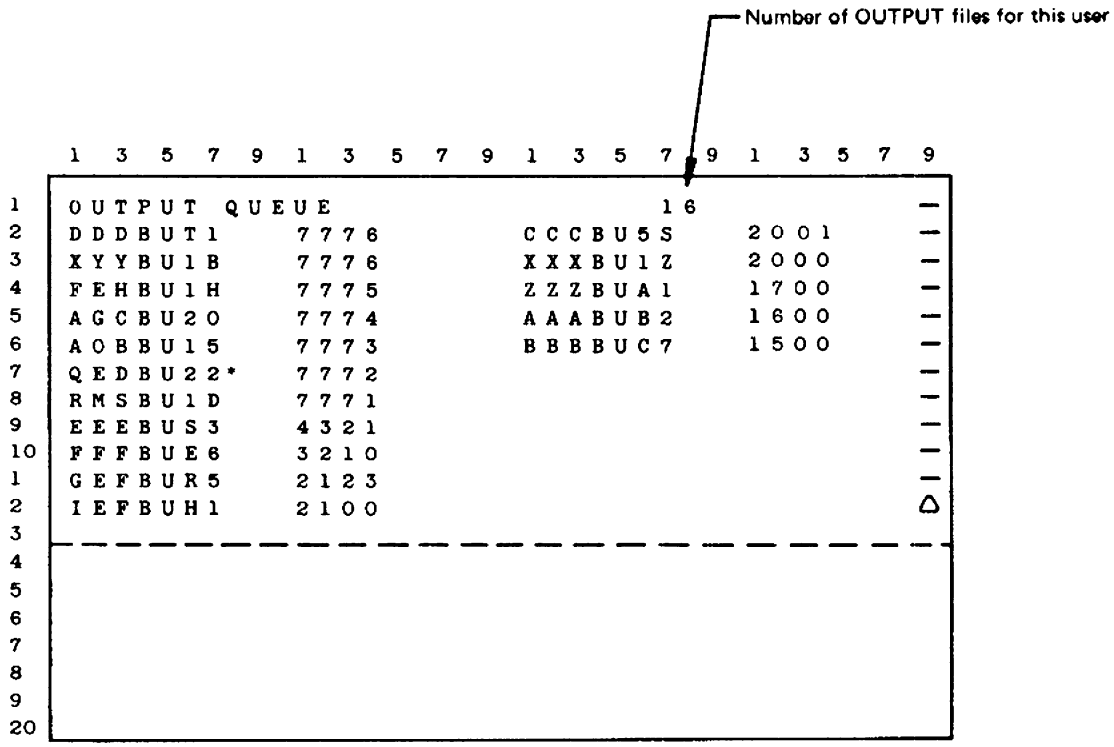


Figure 7-1. H.I Display: Jobs in Input Queue



*Denotes the file currently being printed.

Figure 7-2. H.O Display: Jobs in Output Queue

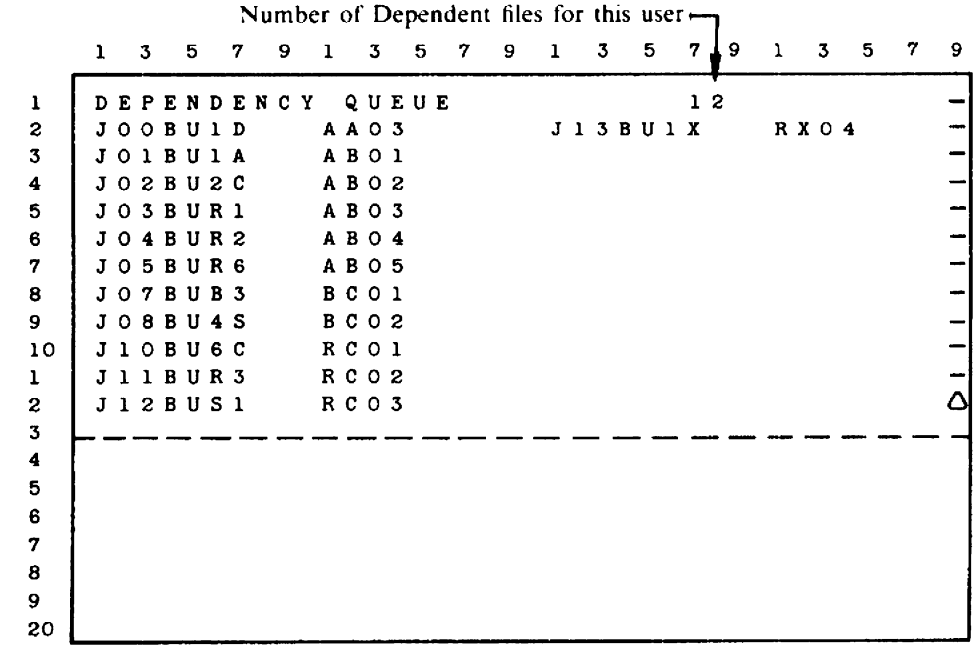


Figure 7-3. H,D Display: Jobs in Dependency Queue

B DISPLAY

The jobs associated with the user's terminal and currently in execution will be displayed in response to the entry:

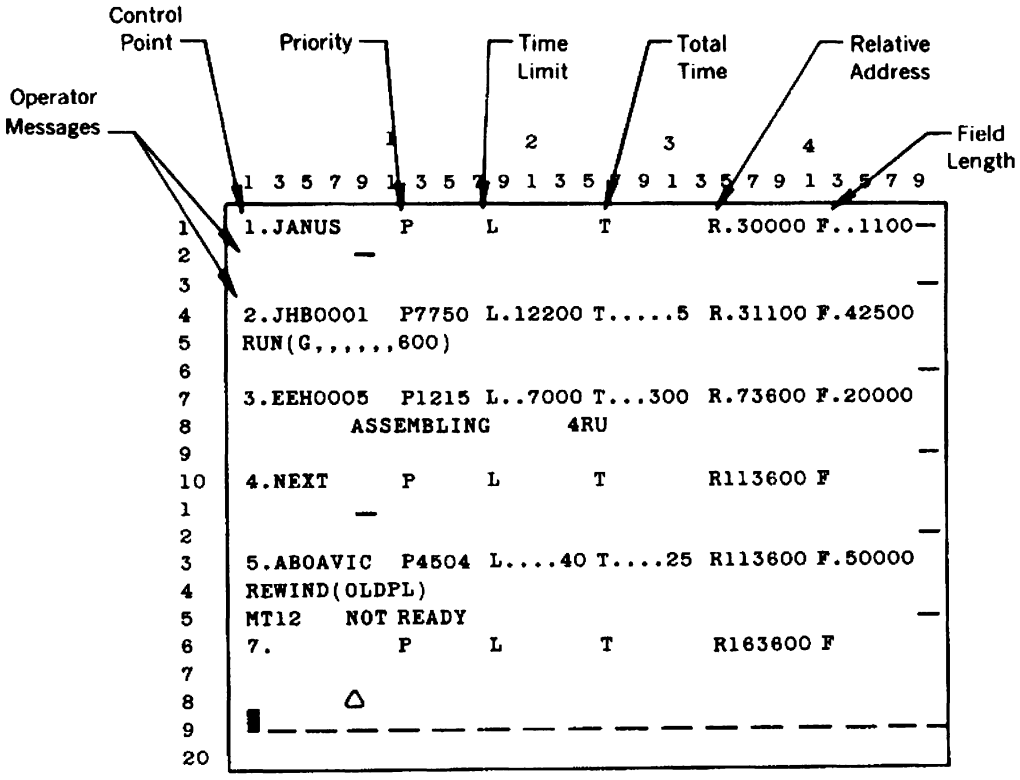
```
B
```

This display on the screen is updated approximately every 10 seconds. This interval allows the user to watch the progress of his job throughout execution simply by entering one B command.

Displayed information is similar to that shown in the central site B display; it shows the job card information for each job running at each control point. All control points are displayed except the one at which INTERCOM is running (see figure 7-4).

The commands OUTPUT, DROP or PRIOR will interrupt the display; it will be regenerated after the command is processed. Any other command entered by the user will terminate the display. The display also is terminated by error messages and messages from the user to the central site.

The display shown in figure 7-4 is on a 50 by 20 format screen. On an 80 by 13 format screen, the second and third lines at each control point would appear as a single line of 80 characters. INTERCOM is running at control point 6 and is not shown on the display in figure 7-4.



INTERCOM is running at control point 6 which is not displayed. On an 80 x 13 screen format, the second and third lines at each control point will appear as a single line of 80 characters.

Figure 7-4. B Display: Jobs in Execution

STOPPING DISPLAY WITH END

To terminate an H or B display, the user should enter:

```
END   or   E
```

Whichever display, H or B, is currently being shown will terminate. The screen is cleared and the response COMMAND- is sent. The user may now enter any legal command.

OUTPUT MANIPULATION

DIVERTING OUTPUT

Normally, the print file from a user's job is transmitted to the associated terminal for output on the terminal line printer. Output may be diverted to the central site by entering:

```
OUTPUT,jobname   or   O,jobname
```

Once this command has been entered, the input or dependent file and the output files for the job are associated with the central site; they are no longer accessible to the terminal issuing the command. All output will be processed at the central site.

Output from a job may be transferred to another terminal by entering:

```
OUTPUT,jobname,xx   or   O,jobname,xx
```

In this case xx is the user id for the logged in user receiving the job output. All files (input, dependent, and output) associated with the job will be associated with user xx and no longer accessible to the user issuing the command. The output is printed at the terminal associated with user id xx.

If a file specified in the OUTPUT command is currently printing, printing stops, and the file is rewound before it is transferred. A user may delay printing of the current file until a higher priority file is printed by specifying his own user id. The file is rewound and transferred even if it is in a suspended printing state.

As soon as the OUTPUT command is processed, the terminal returns to the batch processing mode. If an H or B display were entered previously, it will be regenerated automatically. If printing or reading were interrupted with the INTER key to enter OUTPUT, GO must be entered to resume printing or reading.

All files diverted by OUTPUT to the central site are given the fixed priority defined as an installation option.

Examples:

To transfer printing of job JOBAA05 to the central site:

```
OUTPUT,JOBAA05
```

To transfer the printing of job JOBAA06 to terminal AA:

```
O,06,AA
```

Note that the abbreviated job name (06) is sufficient to identify the job.

CHANGING OUTPUT PRIORITY

The terminal user can change the priority of the output file for any job associated with the terminal and currently in the SCOPE output queue by entering:

```
PRIOR,jobname,n   or   P,jobname,n
```

The jobname identifies the job, and n is a 1-4 digit positive octal number which defines the new priority. The number 7777 gives a file the highest priority. The 7-character jobname can be abbreviated to the last two characters.

Once the new priority is entered, it can be displayed by entering H, O.

If the specified job is currently printing, its priority cannot be changed; as it is no longer in the SCOPE output queue.

After the PRIOR command is processed, the terminal resumes normal operation in the batch processing mode.

Example:

To delay the output of the file currently printing (JOBA123) until another of his output files (JOBX135) has printed; the user may enter:

```
S
P,JOBX135,7000
OUTPUT,JOBA123,AD
C
```

DROPPING JOBS FROM EXECUTION

The terminal user may drop a job associated with his terminal while it is in the input or dependency queue or in execution by entering:

```
DROP,jobname   or   D,jobname
```

The referenced job will be dropped from execution if it is executing; if it is in an input or dependency queue awaiting execution, the input file is dropped. A job in the output queue will not be dropped and any output files already generated by a job dropped in execution will be placed in the output queue for printing. The job name is a 7-character identifier of which the last two characters are unique. The job name may be abbreviated to the last two characters.

As soon as the DROP command is processed, the terminal resumes normal operation in the batch processing mode. If an H or B display command had been entered previously, the display is regenerated automatically. An interrupted reading or printing can be resumed by typing GO or G.

Example:

To drop JOBAB77 from execution, enter either:

```
DROP,JOBAB77
DROP,77
```

INTERCOM CHARACTER SET

A

The character set options available to the INTERCOM user are listed on the following pages. Included are the CDC 63-character set, the CDC 64-character set, and the ASCII† 64-character subset along with corresponding character representations of teletypewriter and display terminal keyboards. In addition, display codes are shown.

Because of several possible character set and terminal combinations, the user should be familiar with the options selected for his installation. For example, configurations can be mismatched, such as a 63-character terminal device used in a 64-character graphic set environment.

CDC 6000 Character Set
63-Character

63-Char Graphic	TTY	200 UT (BCD)	200 UT (ASCII)	Display Code	63-Char Graphic	TTY	200 UT (BCD)	200 UT (ASCII)	Display Code
A	A	A	A	01	6	6	6	6	41
B	B	B	B	02	7	7	7	7	42
C	C	C	C	03	8	8	8	8	43
D	D	D	D	04	9	9	9	9	44
E	E	E	E	05	+	+	+	+	45
F	F	F	F	06	−	−	−	−	46
G	G	G	G	07	*	*	*	*	47
H	H	H	H	10	/	/	/	/	50
I	I	I	I	11	(	(	(	(	51
J	J	J	J	12	)	)	)	)	52
K	K	K	K	13	$	$	$	$	53
L	L	L	L	14	=	=	=	=	54
M	M	M	M	15	blank	space	space	space	55
N	N	N	N	16	,	,	,	,	56
O	O	O	O	17	.	.	.	.	57
P	P	P	P	20	≡	"	≡	#	60
Q	Q	Q	Q	21	[	[	[	'	61
R	R	R	R	22	]	]	]	!	62
S	S	S	S	23	:	:	:	:	63
T	T	T	T	24	≠	;	≠	"	64
U	U	U	U	25	→	#	→	_	65
V	V	V	V	26	∨	@	∨	]	66
W	W	W	W	27	∧	&	∧	&	67
X	X	X	X	30	↑	∧	↑	@	70
Y	Y	Y	Y	31	↓	!	↓	?	71 ①
Z	Z	Z	Z	32	<	<	<	[	72
0	0	0	0	33	>	>	>	>	73
1	1	1	1	34	≤	\	≤	<	74
2	2	2	2	35	≥	?	≥	\	75
3	3	3	3	36	¬	%	% ②	^ ③	76
4	4	4	4	37	;	;	;	;	77
5	5	5	5	40					

†American National Standard Code for Information Interchange.

CDC 6000 Character Set
64-Character

6000 Graphic	TTY	200 UT (BCD)	200 UT (ASCII)	Display Code	6000 Graphic	TTY	200 UT (BCD)	200 UT (ASCII)	Display Code
:	:	:	:	00	6	6	6	6	41
A	A	A	A	01	7	7	7	7	42
B	B	B	B	02	8	8	8	8	43
C	C	C	C	03	9	9	9	9	44
D	D	D	D	04	+	+	+	+	45
E	E	E	E	05	−	−	−	−	46
F	F	F	F	06	*	*	*	*	47
G	G	G	G	07	/	/	/	/	50
H	H	H	H	10	(	(	(	(	51
I	I	I	I	11	)	)	)	)	52
J	J	J	J	12	$	$	$	$	53
K	K	K	K	13	=	=	=	=	54
L	L	L	L	14	blank	space	space	space	55
M	M	M	M	15	,	,	,	,	56
N	N	N	N	16	.	.	.	.	57
O	O	O	O	17	≡	"	≡	#	60
P	P	P	P	20	[	[	[	'	61
Q	Q	Q	Q	21	]	]	]	!	62
R	R	R	R	22	%	%	%	%	63
S	S	S	S	23	≠	'	≠	"	64
T	T	T	T	24	→	#	→	_	65
U	U	U	U	25	∨	@	∨	]	66
V	V	V	V	26	∧	&	∧	&	67
W	W	W	W	27	↑	^	↑	@	70
X	X	X	X	30	↓	!	↓	?	71 ①
Y	Y	Y	Y	31	<	<	<	[	72
Z	Z	Z	Z	32	>	>	>	>	73
0	0	0	0	33	≤	\	≤	<	74
1	1	1	1	34	≥	?	≥	\	75
2	2	2	2	35	¬	_	% ②	^	76
3	3	3	3	36	;	;	;	;	77
4	4	4	4	37					
5	5	5	5	40					

64-Character ASCII Subset

ASCII Graphic	TTY	200 UT (BCD)	200 UT (ASCII)	Display Code	ASCII Graphic	TTY	200 UT (BCD)	200 UT (ASCII)	Display Code
:	:	:	:	00	5	5	5	5	40
A	A	A	A	01	6	6	6	6	41
B	B	B	B	02	7	7	7	7	42
C	C	C	C	03	8	8	8	8	43
D	D	D	D	04	9	9	9	9	44
E	E	E	E	05	+	+	+	+	45
F	F	F	F	06	–	–	–	–	46
G	G	G	G	07	*	*	*	*	47
H	H	H	H	10	/	/	/	/	50
I	I	I	I	11	(	(	(	(	51
J	J	J	J	12	)	)	)	)	52
K	K	K	K	13	$	$	$	$	53
L	L	L	L	14	=	=	=	=	54
M	M	M	M	15	blank	space	space	space	55
N	N	N	N	16	,	,	,	,	56
O	O	O	O	17	.	.	.	.	57
P	P	P	P	20	#	#	≡	#	60
Q	Q	Q	Q	21	'	'	[	'	61
R	R	R	R	22	!	!	]	!	62 ①
S	S	S	S	23	%	%	%	%	63
T	T	T	T	24	"	"	≠	"	64
U	U	U	U	25	_	_	→	_	65
V	V	V	V	26	]	]	∨	]	66
W	W	W	W	27	&	&	∧	&	67
X	X	X	X	30	@	@	↑	@	70
Y	Y	Y	Y	31	?	?	↓	?	71
Z	Z	Z	Z	32	[	[	<	[	72
0	0	0	0	33	>	>	>	>	73
1	1	1	1	34	<	<	≤	<	74
2	2	2	2	35	\	\	≥	\	75
3	3	3	3	36	^	^	% ②	^	76
4	4	4	4	37	;	;	;	;	77

SYSTEM ERROR MESSAGES B

The following error messages are returned by INTERCOM:

Message	Meaning
ARITHMETIC ERROR MODE = x ADDRESS = xxxxxx	Central processor error exit occurred at address indicated.
COMMAND NOT ALLOWED FROM TTY	Attempt was made to enter remote batch command from Teletype.
CONTROL CARD ERROR	Parameters entered with INTERCOM command or SCOPE control card contain invalid characters, too many characters, are separated by illegal delimiters, or are otherwise not in proper format.
CPU ABORT	Central processor program requested abnormal job termination.
ECS PARITY ERROR	Error during system storage move terminated job. Notify installation of this error.
ERR xx ON DISK I/O EDIT FILE	I/O error (code xx) detected on EDITOR edit file I/O. Re-enter EDITOR command. Notify installation of this error.
FILE QUOTA EXCEEDED	User's maximum private file limit (as specified in password file) has been exceeded. Drop, save as permanent, send as batch to central site, or otherwise eliminate excess private files before other commands can be entered.
FORMAT ERROR	Command is not recognized or has improper format.
HUNG IN AUTOMATIC RECALL	A PPU routine called with automatic recall has dropped without setting complete bit. Notify the installation of this error.
ILLEGAL COMMAND FROM EDITOR	Attempt was made to enter batch processing or other illegal command from EDITOR.

Message	Meaning
ILLEGAL COMMAND WHILE IN SETUP	Attempt was made to enter batch processing or other illegal command from SETUP.
INSUFFICIENT FL AUTHORIZED FOR REQUESTED PROGRAM	Requested program requires greater field length than that allowed user in password file.
LOADER ERROR	Program not successfully loaded.
MUJ ABORT, USER DETACHED	Multi-user job, such as EDITOR, has aborted. User is returned to command mode.
MUJ REQUEST NOT ALLOWED HERE	Request to use a multi-user job was made under illegal condition, such as from EDITOR or SETUP.
OPERATOR DROP	Central site operator dropped last command or user's program.
PLEASE LOGIN	User did not enter LOGIN as first entry.
PP CALL ERROR	Central processor program entered illegal PP call.
PP CALL WITH RECALL ERROR	PP program called by central processor program with recall, but complete bit was already set.
PPU ABORT	PP program encountered illegal request, such as illegal file name or bad permanent file request. This diagnostic also may indicate a misspelled or nonexistent command or program call.
REPEAT LINE	Because of heavy INTERCOM activity, last input line was lost. User must re-enter line.
RERUN	Remote job restarted by central site operator.
SORRY, I DO NOT UNDERSTAND	INTERCOM does not recognize this entry.
SYSTEM ERR FROM aaaaaa	System error occurred in EDITOR at address aaaaaa. Notify installation of this error.
TIME LIMIT	Command or user's program exhausted allotted central processor time. User may enter ETL command to increase time limit for subsequent commands or programs.
USER ABORT	User requested command or program to be dropped.

Message	Meaning
YOU ARE NOT AUTHORIZED TO USE THIS PROGRAM	User's access level, as specified on password file, is insufficient for this command or program.
YOU ARE OUT OF TIME	Maximum central processor time for the user's entire session with INTERCOM (as specified in password file) has been exceeded. User can perform certain cleanup functions such as saving files but then must LOGOUT.
YOU HAVE TOO MANY FILES--PLEASE RETURN SOME	User's file quota has been exceeded and last command entered is not allowed. Enter commands to reduce the number of private files below maximum.
WHAT	EDITOR did not recognize entry. Misspelled command, non-existent command, or INTERCOM or SCOPE command without a terminating period or right parenthesis was entered.

COMMAND ERROR MESSAGES C

The diagnostic messages returned by INTERCOM commands, EDITOR commands, SETUP directives, and remote batch commands are listed alphabetically in this appendix. The INTERCOM commands are listed first, followed by the EDITOR commands, the SETUP directives, and finally the remote batch processing commands available at a 200 User Terminal. The error messages returned by INTERCOM are given for each command and directive together with recommended user action.

INTERCOM COMMANDS

Command	Message	Meaning	User Action
ASSETS	None		
BATCH	INVALID DISPOSITION-PLEASE TRY AGAIN	Disposition entered is not defined.	Enter valid disposition.
	INVALID FILE NAME-PLEASE TRY AGAIN	File name does not exist or is not allowed.	Enter another file name.
	DUPLICATE FILE NAME	Attempt was made to rename file with already existing file name.	Rename file.
	THIS IS A RESERVED FILE	This file cannot be made private.	Enter another file name.
	INVALID DISPOSITION FOR PERMANENT FILE	Attached permanent file may be renamed only with BATCH command.	Copy file to nonpermanent file and re-enter BATCH command.
CONNECT	ILLEGAL FILE NAME	File name specified improperly.	Re-enter command with file name in correct format.
CONVERT	ERR-CARD NO.nnnnnnn; BLANK CARD	Blank card found in filename-1 (BASIC conversion only).	Reconstruct file or create new file in BASIC format.
	ERR-CARD NO.nnnnnnn; NO BASIC STATEMENT	Missing BASIC statement number in filename-1	Reconstruct or create new file in BASIC format.

Command	Message	Meaning	User Action
CONVERT (cont.)	ERR-CAN'T FIND FILE fffffff	File is not in list of user's files.	Re-enter command with correct name
	ERR-FILE NAME MUST BE ALPHANUM,< 8 CHAR, 1ST CHAR A-Z	File name specified improperly.	Re-enter command with file name in correct form.
	ERR-PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR-TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	ERR-fffffff ALREADY EXISTS	File already exists in list of user's files.	Discard file or try another name.
	ERR-fffffff IMPROPERLY ATTACHED FOR THIS OPERATION	File is permanent and does not have permission required for this operation.	Attach file with required passwords and try again.
	ERR-REQUIRED PARAMETERS MISSING	Specify at least two file names with CONVERT command.	Re-enter command correctly.
DISCARD	ERR-CANT FIND FILE fffffff	File not in list of user's files.	Re-enter command with correct file name.
	ERR-FILE NAME MUST BE ALPHANUM, < 8 CHARS, 1ST CHAR A-Z	File name specified improperly.	Re-enter command with file name in correct form.
	ERR-FILE NAME REQUIRED	File name must be specified.	Re-enter command with a file name.
	ERR-TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	ERR-fffffff BUSY, TRY LATER	Permanent file is currently being accessed by another user or batch job.	Try again later.
	ERR-PERM. FILE ERROR, RETURN CODE = xx	Occurred during attempt to purge permanent file.	Consult SCOPE R.M. for nature of error.
	ERR-xx MUST BE ALPHANUMERIC, 1-9 CHARS	ID or PP parameter not in proper format.	Re-enter command correctly.

Command	Message	Meaning	User Action
DISCONT	ILLEGAL FILE NAME	File name specified improperly.	Re-enter command correctly.
DMP	DMP NOT VALID WITHOUT SAVEFL,ON	SCOPE utility DMP cannot be used unless field length is saved.	Enter SAVEFL,ON and re-enter DMP utility.
EDITOR	None		
EFL	REQUEST EXCEEDS AUTHORIZATION	Requested field length greater than authorized in password file.	Try smaller field length.
	TOO MANY DIGITS	Too many digits in FL request.	Re-enter command with no more than six digits.
ERRORS	ERR-COMPILER NAME REQUIRED	Compiler name must be specified.	Re-enter command with compiler name.
	ERR-PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR-PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct form.
	ERR-TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter in correct form.
ETL	REQUEST EXCEEDS AUTHORIZATION	Time limit request exceeds that authorized in password file.	Try smaller time limit.
	TOO MANY DIGITS	Too many digits in time limit request.	Re-enter command with no more than four digits.
FETCH	ERR-CANT FIND FILE fffffff	File is not in list of user's permanent files, or ID has been entered incorrectly.	Re-enter command correctly.
	ERR-FILE NAME MUST BE ALPHANUM, < 8 CHAR, 1ST CHAR A-Z	File name specified improperly.	Re-enter command with file name in correct form.
	ERR-FILE NAME REQUIRED	File name must be specified.	Re-enter command with a file name.

Command	Message	Meaning	User Action
FETCH (cont.)	ERR-TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	ERR-fffffff BUSY, TRY LATER	Permanent file is currently being accessed by another user or a batch job.	Try again later.
	ERR-fffffff ALREADY EXISTS	User already has private file by this name.	DROP or rename private file and re-enter FETCH.
	ERR-PERM. FILE ERROR, RETURN CODE = xx	Occurred during attempt to attach permanent file.	Consult SCOPE R.M. for nature of error.
	ERR-xx MUST BE ALPHANUMERIC, 1-9 CHARS	ID or PP parameter not entered in proper format.	Re-enter command correctly.
FILES	None		
LOCK	None		
LOGIN	PERM FILE ERR, RETURN CODE = xxB	Occurred during attempt to access password file. Password file may not be loaded at central site.	Notify installation of this error.
	YOU HAVE HAD THREE TRIES–GET HELP	Incorrect user name or password entered three times.	Obtain correct user name and password and re-enter LOGIN command.
	USER NAME/PASSWORD IN USE AT ANOTHER TERMINAL	Another user is already logged in with given name and password.	Type different password/ user name.
	INVALID USER NAME OR PASSWORD	Illegal user name or password entered.	Type user name again.
	PREVIOUS USER AUTOMATICALLY LOGGED OUT	Previous user did not log out.	Enter user name.
	NO ID AVAILABLE	User attempt to LOGIN with unrestricted password for first time with this user name; all available user id's have been assigned.	Notify installation of this error.

Command	Message	Meaning	User Action
LOGIN (cont.)	UNRESTRICTED PERM FILE PROBLEMS	No information found on unrestricted password file.	Notify installation of this error.
	NO USER NAME/ PASSWORD ID AVAILABLE	No id bit map found on unrestricted password file.	Notify installation of this error.
LOGOUT	None		
MAP	None		
MESSAGE	CONSOLE BUSY– TRY AGAIN LATER	Central site operator has not cleared last user message displayed at central site.	Wait and re-enter MESSAGE
PAGE	ILLEGAL FILE NAME	File name specified improperly.	Re-enter command
REDUCE	None		
RETURN	RETURN CARD MUST HAVE PARAMETERS	Specify at least one file name with RETURN command.	Re-enter command correctly.
SAVEFL	None		
SCREEN	None		
SEND	USER NAME xxxxxxxxxx IS NOT ACCESSIBLE	User is logged out, has locked out incoming messages, or has a different password if sender has unrestricted password.	None.
	*** NOT ALLOWED FROM UNRESTRICTED USER	User has unrestricted password; may not send message to all logged in users.	None.
	PERM FILE ERR, RETURN CODE=xxB	Occurred during attempt to access password file.	Notify installation of this error.
SETUP	None		
SITUATE	PERM FILE ERR, RETURN CODE=xxB	Occurred during attempt to access password file.	Notify installation of this error.

Command	Message	Meaning	User Action
STORE	ERR-CANT FIND FILE fffffff	File is not in list of user's private files.	Re-enter command with correct file name.
	ERR-FILE NAME MUST BE ALPHANUM, < 8 CHAR, 1ST CHAR A-Z	File name is specified improperly.	Re-enter command with file name in correct form.
	ERR-FILE NAME REQUIRED	File name must be specified.	Re-enter command with a file name.
	ERR-TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	ERR-PERM FILE fffffff ALREADY EXISTS	File name already exists in list of user's permanent files.	Discard file or try another name.
	ERR-fffffff DOES NOT RESIDE ON PERM FILE DEVICE	Private file not currently on legal permanent file device.	Issue REQUEST,fname, *PF., copy fffffff to fname, then STORE, fname.
	ERR-STORE NOT DONE, xxxxxxx FULL	PF directory or catalog is currently full.	Attempt STORE when space is available.
	ERR-NO INFORMATION IN FILE	File specified contains no information.	None.
	ERR-PERM. FILE ERROR, RETURN CODE=xx	Occurred during attempt to catalog permanent file.	Consult SCOPE R.M. for nature of error.
	ERR-xx MUST BE ALPHANUMERIC, 1-9 CHARS	ID or PP parameter not entered in proper format.	Re-enter command correctly.
	ERR-HEAVY PERM. FILE ACTIVITY – TRY LATER	Heavy PF activity prevents file storage.	Try again shortly.
SWITCH	None		
TAPE	None		
UNLOAD	None		

EDITOR COMMANDS

ADD	ADD WONT REPLACE OR BYPASS LINES	ADD will not bypass or replace existing line.	Re-enter command with new line and increment values.
	ERR- INCREMENT MUST BE > 0	Increment must be greater than zero.	Re-enter command with legal increment value.
	ERR- CREATE, ADD, RESEQ, SEQ ILLEGAL UNDER BASIC FORMAT	Increment mode may not be used with BASIC	Enter BASIC lines in the form: line = text.
	ERR- LINE NO. > 999999	Next line number generated for increment mode or file sequencing exceeds maximum.	Change increment or line value and try again.
	ERR-NO INFORMATION IN EDIT FILE	User's edit file is empty.	Load or create a file and try again.
	ERR-PARAM n: ILLEGAL LINE NUMBER	Line number at parameter n outside legal range.	Re-enter command in correct form.
	ERR-PARAM n: NUMERIC PARAMETER REQUIRED	Parameter n must be numeric characters only.	Re-enter command in correct form.
	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	xxxxxxx TRUNCATED FROM LONG LINE	Line exceeds terminal's current maximum character count; first 7 characters truncated were xxxxxxx.	None required.
BYE	WARNING-EDIT FILE NOT SAVED	Command would destroy user's edit file; command was not executed	Save or delete edit file, or re-enter command to delete file automatically.

Command	Message	Meaning	User Action
CREATE	ERR- INCREMENT MUST BE > 0	Increment value cannot be zero.	Re-enter command with legal increment value.
	ERR- CREATE, ADD, RESEQ, SEQ ILLEGAL UNDER BASIC FORMAT	Increment mode cannot be used with BASIC.	Enter BASIC lines in form: lines = text.
	ERR- LINE NO. > 999999	Next line number generated for increment mode or file sequencing exceeds maximum.	Change increment or line value and try again.
	ERR- PARAM n: ILLEGAL LINE NUMBER	Line number of parameter n outside legal range.	Re-enter command in correct form.
	ERR- PARAM n: NUMERIC PARAMETER	Parameter n must be numeric characters only.	Re-enter command in correct form.
	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	WARNING- EDIT FILE NOT SAVED	Command would destroy user's edit file; command was not executed.	Save or delete edit file, or re-enter command to delete file automatically.
	xxxxxxx TRUNCATED FROM LONG LINE	Line exceeds terminal's current maximum character count, first 7 characters truncated were xxxxxxx.	None required.
DELETE	ERR- COL. OR UNIT SPECIFIED BUT NO /TEXT1/	Column number or unit parameter may be specified only with /text/ parameter.	Re-enter command in correct form.
	ERR-ILLEGAL COLUMN RANGE	Column numbers must be in ascending order and column positions must at least equal text string character count.	Re-enter command in correct form.

Command	Message	Meaning	User Action
DELETE (cont.)	ERR- ILLEGAL LINE RANGE	Line numbers must be in ascending order.	Re-enter command in correct form.
	ERR- LINE NUMBER REQUIRED	Line number required in DELETE command.	Re-enter command in correct form.
	ERR-PARAM n: COLUMN SPECIFICATION INCOMPLETE	Column numbers in parameter n not correct (col-1,col-2).	Re-enter command in correct form.
	ERR- PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct form.
	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR- TEXT STRING > 20 CHAR. NOT ALLOWED	Text strings are 20 character maximum.	Re-enter command with legal text string.
	ERR- /TEXT1/ MUST HAVE AT LEAST 1 CHAR	Text search string may not be null.	Re-enter command in legal form.
EDIT	ERR- CANT FIND FILE fffffff	File name not in list of user's files.	Re-enter command with correct file name.
	ERR- EDITOR FILE LIMIT CURRENTLY EXCEEDED, TRY LATER	EDITOR's user private file limit momentarily exceeded.	Try again in a few seconds.
	ERR- FILE NAME REQUIRED	File name must be specified.	Re-enter command with a file name.
	ERR- FILE NAME MUST BE ALPHANUM, < 8 CHAR, 1ST CHAR A-Z	File name specified improperly.	Re-enter command with file name in correct form.

Command	Message	Meaning	User Action
EDIT (cont.)	ERR- LINE NO. > 999999	Next line number generated for increment mode or file sequencing exceeds maximum.	Change increment or line value and try again.
	ERR-LINE NUMBERS OUT OF SEQUENCE	Line numbers are out of order or nonexistent.	Load file using EDIT command with sequence parameter.
	ERR- PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR- fffffff CONNECTED TO TERMINAL	File is connected to user's terminal.	Disconnect file and try again.
	ERR- fffffff IMPROPERLY ATTACHED FOR THIS OPERATION	Permanent file does not have permission required for this operation.	Attach file with required passwords and try again.
	ERR- CREATE, ADD, RESEQ, SEQ ILLEGAL UNDER BASIC FORMAT	The SEQ parameter may not be specified while in BASIC format.	Use the CONVERT utility to sequence the BASIC file for EDITOR access.
	LINES > 510 CHARS WERE TRUNCATED	Lines in file exceed 510 characters; they were truncated.	None; files with lines greater than 510 characters may not be edited.
	WARNING- EDIT FILE NOT SAVED	Command would destroy user's edit file; command was not executed.	Save or delete edit file, or re-enter command to delete file automatically.
FORMAT	ERR- CH= MUST SPECIFY COUNT < 511	Character count specified exceeds maximum.	Re-enter command with legal character count.
	ERR- PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct form.
	ERR-PARAM n: NUMERIC PARAMETER REQUIRED	Parameter n must be numeric characters only.	Re-enter command in correct form.

Command	Message	Meaning	User Action
FORMAT (cont.)	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR- TAB= MUST SPECIFY ONLY 1 CHAR	TAB= keyword may be followed by one character only.	Re-enter command in correct form. (Check that tab character parameter is followed by legal separator.)
	ERR- TABS TOO BIG OR OUT OF ORDER	Tab positions must be in ascending order and may not exceed 510.	Re-enter command in correct form.
	ERR- TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	TABS > CH VALUE CLEARED	User specified new character count; tab position parameter from previous format specification contains column numbers beyond new count. They were cleared.	None required.
LIST	ERR- COL. OR UNIT SPECIFIED BUT NO /TEXT1/	Column number or unit parameter may be specified only with /text/ parameter.	Re-enter command in correct form.
	ERR- ILLEGAL COLUMN RANGE	Column numbers must be in ascending order and column positions must at least equal text string character count.	Re-enter command in correct form.
	ERR- ILLEGAL LINE RANGE	Line numbers must be in ascending order.	Re-enter command in correct form.
	ERR-PARAM n: COLUMN SPECIFICATION INCOMPLETE	Column numbers in parameter n not specified correctly (col-1,col-2).	Re-enter command in correct form.

Command	Message	Meaning	User Action
LIST (cont.)	ERR- PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct form.
	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR-TEXT STRING > 20 CHAR. NOT ALLOWED	Text string limited to 20 characters.	Re-enter command with legal text string.
	ERR- /TEXT1/ MUST HAVE AT LEAST 1 CHAR	Text search string may not be null.	Re-enter command in legal form.
	NO SUCH LINES	No lines satisfy command requirements.	None required.
RESEQ	ERR-INCREMENT MUST BE > 0	Increment value cannot be zero.	Re-enter command with legal increment value.
	ERR-CREATE, ADD, RESEQ, SEQ ILLEGAL UNDER BASIC FORMAT	RESEQ command cannot be used with BASIC.	None required.
	ERR- LINE NO. > 999999	Next line number generated for increment mode or file sequencing exceeds maximum.	Change increment or line value and try again.
	ERR- NO INFORMATION IN EDIT FILE	User's edit file is empty.	Load or create a file and try again.
	ERR- PARAM n: ILLEGAL LINE NUMBER	Line number at parameter n outside legal range.	Re-enter command in correct form.

Command	Message	Meaning	User Action
RESEQ (cont.)	ERR- PARAM n: NUMERIC PARAMETER REQUIRED	Parameter n must be specified as numeric characters only.	Re-enter command in correct form.
	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
RUN	ERR- COMPILER NAME REQUIRED	Compiler name must be specified.	Re-enter command with compiler name.
	ERR-EDITOR FILE LIMIT CURRENTLY EXCEEDED, TRY LATER	EDITOR's user private file limit momentarily exceeded.	Try again shortly.
	ERR-NO INFORMATION IN EDIT FILE	User's edit file is empty.	Load or create a file and try again.
	ERR-PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct form.
	ERR-PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR- TOO MANY PARAMETERS	Too many parameters specified in command.	Re-enter command in correct form.
	ERR- USER FILE LIMIT EXCEEDED	Maximum private file limit exceeded.	Eliminate one or more files (RETURN command) and try again.
	ERR- fffffff CONNECTED TO TERMINAL	File is connected to user's terminal.	Disconnect file and try again.
	ERR- fffffff IMPROPERLY ATTACHED FOR THIS OPERATION	Permanent file does not have permission required for this operation.	Attach file with required passwords and try again.

Command	Message	Meaning	User Action
RUN (cont.)	LINES TRUNCATED: CH=nnn CHARS LONGEST LINE WAS mmm	Lines exceeding maximum of nnn were truncated. Longest line was mmm characters.	If truncation is not acceptable, change character count and try again.
	ERR-OVERWRITE ILLEGAL ON PERM. FILE	SETFILE already exists as attached permanent file.	Return or rename SETFILE.
SAVE	ERR-EDITOR FILE LIMIT CURRENTLY EXCEEDED, TRY LATER	EDITOR's user private file limit momentarily exceeded.	Try again shortly.
	ERR- FILE NAME MUST BE ALPHANUM, < 8 CHAR, 1ST CHAR A-Z	File name improperly specified.	Re-enter command with file name in correct form.
	ERR- FILE NAME REQUIRED	File name must be specified	Re-enter command with a file name.
	ERR- NO INFORMATION IN EDIT FILE	User's edit file is empty.	Load or create a file and try again.
	ERR- OVERWRITE ILLEGAL ON PERM. FILE	Permanent file with this name already attached.	Discard permanent file and try again, or save user file under a different name.
	ERR- PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct form.
	ERR- PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR- USER FILE LIMIT EXCEEDED	Maximum private file limit exceeded.	Eliminate one or more files (RETURN command) and try again.
	ERR- fffffff ALREADY EXISTS	File name already exists in list of user's files.	Discard file, try another name, or specify OVERWRITE.

Command	Message	Meaning	User Action
SAVE (cont.)	LINES TRUNCATED: CH=nnn CHARS, LONGEST LINE WAS mmm	Lines beyond maximum of nnn were truncated. Longest line was mmm characters.	If truncation is not acceptable, change character count and try again.
TEXT REPLACEMENT /text-1/=/text-2/	ERR- ILLEGAL COLUMN RANGE	Column numbers must be in ascending order and column positions must at least equal text string character count.	Re-enter command in correct form.
	ERR- ILLEGAL LINE RANGE	Line numbers must be in ascending order.	Re-enter command in correct form.
	ERR- PARAM n: COLUMN SPECIFICATION INCOMPLETE	Column numbers in parameter n not specified correctly (col-1,col-2).	Re-enter command in correct form.
	ERR- PARAM n: DUPLICATE PARAMETER	Parameter n appears twice in command.	Re-enter command in correct format.
	ERR- PARAM n: ILLEGAL LINE NUMBER	Line number at parameter n outside legal range.	Re-enter command in correct form.
	ERR- PARAM n: TOO MANY DIGITS	Parameter n exceeds maximum range: line numbers and increment values 1-6 digits; column numbers and tab positions 1-3 digits.	Re-enter command in correct form.
	ERR- PARAM n: /TEXT2/ REQUIRED	Text replacement string required following equals sign.	Re-enter command in correct form (a null text replacement string is specified as //).
	ERR- PARAM n: UNRECOGNIZABLE PARAMETER	Data in parameter n not recognized.	Check legal delimiters, separators, and spelling. Try again.
	ERR- TEXT STRING > 20 CHAR. NOT ALLOWED	Text strings limited to 20 characters.	Enter text strings greater than 20 characters with two or more commands.
	ERR-/TEXT1/ MUST HAVE AT LEAST 1 CHAR	Text search string may not be null.	Re-enter command in legal form.

Command	Message	Meaning	User Action
TEXT REPLACEMENT (cont.)	LINE nnnnnn > FORMAT CH COUNT	Line nnnnnn expanded beyond maximum character count currently in effect during text replacement operation; truncation does not occur.	None required.
	LINE nnnnnn TRUNCATED TO 510	Line nnnnnn expanded beyond 510 characters in text replacement operation; truncation occurred.	None required.
line=text	ERR- PARAM n: TOO MANY DIGITS	Line number is limited to 6 digits.	Re-enter command with correct line number format.
	xxxxxxx TRUNCATED FROM LONG LINE	Line exceeds terminal's current maximum character count; first 7 characters truncated were xxxxxxx.	Change format specifications or re-enter command in correct format.

SETUP DIRECTIVES

Directive	Message	Meaning	User Action
BASICX	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
BYE, END, GOODBYE	None		
COBOLER	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
COBOLX	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
DELETE	WHAT?	Line number contains invalid character; or second line precedes first.	Correct error.
FTN	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
FTNER	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
INTRA-LINE EDITING CODES	CODE UNRECOGNIZABLE	Editing code not recognized.	Correct code.
	FORMAT ERROR	Editing statement incorrectly entered.	Correct line.
	LINE IS A LITTLE TOO LONG FOR ME	FORTRAN statement (including line number) exceeds 66 characters.	Type shorter line.
	NO MATCH ON LITERAL FOUND	SETUP did not find matching literal in text buffer.	Type existing literal.
	THE LITERAL IS TOO LARGE	Literal exceeds 60 characters.	Use shorter literal.
LIST	WHAT?	Line number specification contained non-digit.	Retype line numbers.

Directive	Message	Meaning	User Action
LIST (cont.)	NO FILE.	Nothing in text buffer.	Bring file into text buffer.
NEW	FILE NAME MUST BEGIN WITH LETTER	Name entered not acceptable.	Try another name.
	FILE NAME MAY HAVE ONLY ALPHANUMERIC CHARACTERS	Name entered not acceptable.	Try another name.
	OLD FILE REQUESTED-TYPE OLD/LFN	New file specified duplicates existing private file or attached permanent file name.	UNSAVE old file and re-enter directive or enter NEW directive with a unique file name.
OLD	FILE NOT FOUND-TYPE NEW/LFN	File name specified does not exist as a private file or attached permanent file.	Enter NEW directive to create a file.
RENAME	FILE MUST BEGIN WITH A LETTER.	File name not acceptable.	Enter new name.
	FILE MAY HAVE ONLY ALPHANUMERIC CHARACTERS	File name not acceptable.	Try again.
	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
RESEQ	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
RUN	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
RUNER	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
RUNX	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
SAVE	NO FILE.	Nothing found in text buffer.	Bring file into text buffer.
SCRATCH	None		

Directive	Message	Meaning	User Action
SYSTEM	None		
TAB	None		
TRANS	NO FILE.	Nothing found in text buffer.	Bring BASIC file into text buffer.
UNSAVE	None		

REMOTE BATCH PROCESSING COMMANDS

Command	Message	Meaning	User Action
AGAIN	PRINTER NOT READY	Printer not ready initially or became not ready.	Ready the printer and enter GO.
B	None		
CONTIN	PRINTER NOT READY	Printer initially not ready or became not ready.	Ready printer and enter GO.
	OUTPUT FILE ERROR	Parity error or trouble with mass storage device.	Enter GO to resume printing.
DROP	FILE NOT FOUND	Specified jobname not found.	Correct jobname and re-enter DROP command.
END	None		
GO	PRINTER NOT READY	Printer initially not ready or became not ready.	Ready printer and enter GO.
	OUTPUT FILE ERROR	Parity error or trouble with mass storage device.	Enter GO to resume printing.
H	None		
IMPORT	None		
OUTPUT	FILE NOT FOUND	Specified jobname not found.	Correct jobname and re-enter OUTPUT command.
PRIOR	FILE NOT FOUND	Specified jobname not found.	Correct jobname and re-enter PRIORITY command.
READ	READER NOT READY	Last card not 6/7/8/9; INTERCOM expects more cards.	Put in more cards and enter GO.
	DUPLICATE FILE NAME	File name specified in READ already exists.	Re-enter command with different file name.
	END,CR OR MORE CARDS IN READER REQUIRED	Previous READ command did not encounter EOF card.	Read EOF card through or enter END,CR.

Directive	Message	Meaning	User Action
READ (cont.)	JOB CARD ERROR	First card is not valid job card.	Correct card, enter END, CR and restart reading.
	CARD READ ERROR	Illegal card punch in last 24 cards read.	Correct card and restart from beginning.
	INPUT FILE ERROR	Parity error or trouble with mass storage device.	Restart from beginning.
SUSPEND	None		

CARRIAGE CONTROL CHARACTERS D

Carriage control characters are used to control the spacing of batch output sent to a line printer and also of interactive output sent to the terminals. A carriage control character should be the first character of an output line.

BATCH OUTPUT

Output to a line printer at a 200 USER terminal is treated like output to a line printer at the central site. The carriage control characters are those used by the SCOPE operating system:

Character	Spacing
blank	single space; space to beginning of next line
0	double space; space to beginning of second line
1	page eject; skip to top of next page
+	no space; overprint line
-	single space
2	single space
all others	single space

If a character of the print line is put in the position reserved for carriage control characters, that character will not be printed and the line will be single spaced.

INTERACTIVE OUTPUT

Control characters should be specified for all terminal output generated from within user programs written in all programming languages except BASIC, which supplies the characters automatically. If omitted, the results are unpredictable.

The carriage control used depends on whether the terminal is a Teletype or display. When output ends at a Teletype terminal, the carriage remains positioned immediately after the last character sent to the terminal. The carriage control characters position the Teletype carriage before the line containing the control character. At a display terminal, the carriage is returned to the beginning of the next line so that all logical lines begin at the beginning of a physical line. The only carriage control characters are those that specify double or triple spacing, a CLEAR WRITE in which the entire screen is cleared, and a RESET WRITE in which the screen is not cleared but output starts at the top of the screen.

The following table illustrates the use of carriage control characters at Teletype and display terminals. The letters CR specify a carriage return to beginning of line, LF specify line feed to the next line.

Control Character	Spacing			
	Teletype		Display ††	
	Before Print	After Print	Before Print	After Print
+	CR†	None	None	CR,LF
blank	CR,LF		None	
0	CR,2LF		LF	
-	CR,3LF		2LF	
1	CR,10LF		CLEAR WRITE	
*	CR,10LF		RESET WRITE	
/	CR†	CR	None	
(	CR,LF		None	
)	CR,2LF		LF	
$	CR,3LF		2LF	
=	CR,10LF		CLEAR WRITE	
,	CR†	CR,LF	None	
.	CR,LF		None	
:	CR,2LF		LF	
<	CR,3LF		2LF	
>	CR,10LF		CLEAR WRITE	

†The control characters = / , will not cause an overprint following an input line entered at a Teletype terminal since the system always returns a line feed following an input line.

††Output lines on the display are always followed by a carriage return and line feed regardless of the carriage control character.

HARDWARE CONFIGURATION E

CONFIGURATION

Three configurations of hardware for INTERCOM, version 3.0, are given below: a minimum configuration, a target configuration, and a maximum configuration including the use of party lines.

Minimum Configuration:

Configuration for 6400 Computer under SCOPE 3.3 operation

6671 or 6676 Multiplexer

One Teletype or Display Terminal

Target Configuration:

Configuration for 6600 Computer under SCOPE 3.3 operation

Two 6671 and one 6676 Multiplexers

Twenty Display terminals at 2400 baud and 64 Teletype terminals.

Maximum Configuration:

The maximum number of terminals that may be connected and active depends on whether the terminals are all Teletypes or both Teletype (TTY) and Display (CRT).

Teletype and Display Terminals:

Two 6671 and one 6676 Multiplexers per assigned peripheral processor; with a maximum of three peripheral processors. On the 6671 any CRT terminal can be replaced by a TTY terminal.

Maximum of 32 CRT terminals per peripheral processor
Maximum of 64 TTY terminals per peripheral processor
} Total 288

Connected terminals (with party line):

Maximum 512 CRT terminals per peripheral processor
Maximum 64 TTY terminals per peripheral processor
} Total 1728

(A party line is available for CRT terminals only; each port can handle 16 CRT terminals and 32 ports can be connected to one PPU.)

Teletype Terminals Only:

Two 6676 data set controllers per assigned peripheral processor; with a maximum of 3 peripheral processors.

Connected terminals:

Maximum of 64 TTY terminals per 6676

Maximum of 128 TTY terminals per peripheral processor

} Total 384

These maximums exceed the capabilities of version 3.0; although the product will function, severe loss of response time will result if the maximum number of terminals are connected and active. In addition, the hardware is limited to servicing only one terminal per party line at a time. The service rate is, however, less for party lines.

The table shown below gives the maximum number of terminals for one peripheral processor of which a maximum of three may be dedicated to INTERCOM.

Configuration			
Number of 6671	Number of 6676	Number of Connected Terminals	
		CRT	TTY
1	0	16	0
0	1	0	64
1	1	16	64
2	0	32	0
0	2	0	128
2	1	32	64

Figure E-1 shows a mixed configuration with 20 terminals attached to two data set controllers, one 6671 and one 6676. Of the terminals, 15 are Teletype, and five are display. Ten Teletype terminals are connected via the 6676 to the central computer; five Teletype and five display terminals are connected via the 6671.

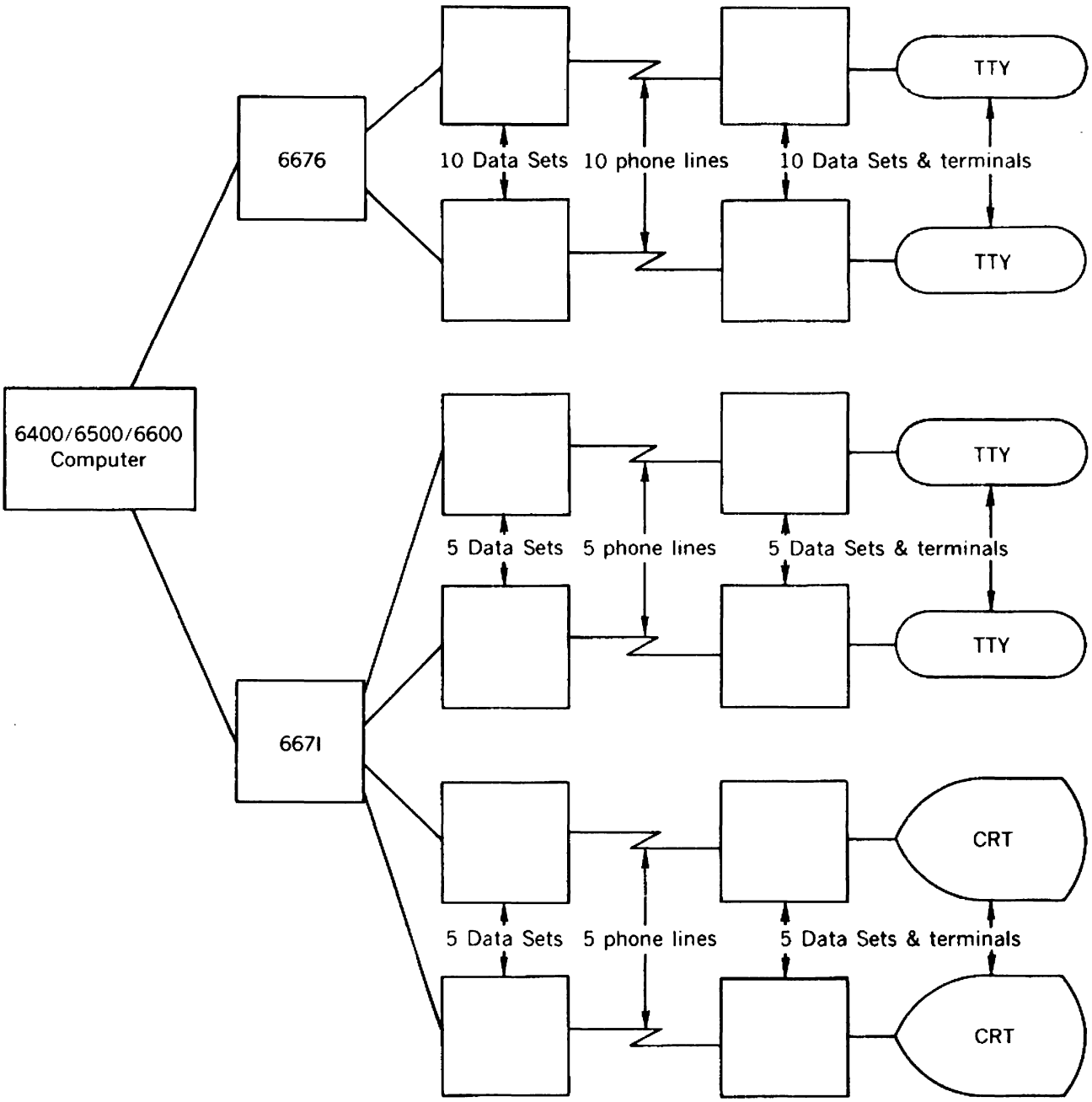


Figure E-1. Mixed Configuration with 5 Display and 15 Teletype Terminals

RESPONSE TIME

The response time is dependent upon several factors including:

Number of users on line

Size of user programs

Number of control points assigned to INTERCOM

Speed of mass storage devices used for program swapping

Type of user activity; varies from input/output or text editing to predominantly computational

Size of time slice used by INTERCOM

Use of extended core storage

The best response time for any given number and type of terminal user is achieved with a large central memory, two or three control points assigned to INTERCOM and a fast mass storage device with minimal batch competition for access.

CENTRAL SITE REQUIREMENTS

INTERCOM requires one dedicated data set controller, one data channel, and a peripheral processing unit to service the terminals. The number of control points for INTERCOM can vary from one to as many as are configured for the operating system.

A rough calculation of central memory requirements, in decimal, may be made from the following formula:

$$CM = 1280 + 16(3.5IT + 17.5BT + 19)$$

where

CM = number of central memory words required

IT = number of interactive terminals active (200 UT's or TTY's)

BT = number of batch terminals active (200 UT's)

The constant, 1280, is the number of central memory words reserved for resident programs and INTERCOM pointers and tables; the variable portion is derived from the input/output buffers and user tables which are 16 central memory words each.

EDITOR COMMANDS F

EDITOR commands are structured as command verbs that may be followed by parameters. Some verbs require no parameters; others require at least one. The format for each EDITOR command is shown below. A command and parameters must be entered into the system as one line. EDITOR command language organization and syntax conventions are described in section 4.

ADD [,line [,incr]] [SUP]

BYE

CREATE [,line [,incr]] [SUP]

DELETE, { ALL / line-1 / LAST } [, { line-2 / LAST }] [,/text/ [,(col-1 [,col-2])] [,UNIT]] [,VETO]

EDIT,filename [,SEQUENCE]

FORMAT [{ ,format-name / [,TAB=c] [,tab-1 [,tab-2 [,... [,tab-n]]]] [,CH=nnn] / ,SHOW }]

LIST [, { ALL / line-1 / LAST } [, { line-2 / LAST }]] [,/text/ [,(col-1 [,col-2])] [,UNIT]]

RESEQ [,line [,incr]]

RUN,system-name [,FILE=filename] [,NOEX] [,SUP]

SAVE,filename [,NOSEQ] [,OVERWRITE]

[=] line=text

/text-1/=/text-2/ [, { ALL / line-1 / LAST } [, { line-2 / LAST }]] [,(col-1 [,col-2])] [,UNIT] [,VETO]

200 USER TERMINAL PERFORMANCE CHECK G

When the user at a 200 User Terminal is experiencing difficulty in running a job and a hardware malfunction is suspected, he may determine or verify the problem area by performing a series of tests. The results of each test should be recorded and relayed to the hardware maintenance personnel. The tests are divided into two groups, off-line and on-line, and should be performed in the order shown.

OFF-LINE TESTS

These tests may be performed without logging into INTERCOM. Power switches on the equipment controller console, line printer, and card reader must be on.

TEST 1. CONSOLE KEYBOARD TO DISPLAY SCREEN

Prior to any keyboard entry, the display screen should be clear:

Press the CLEAR key. If no action results,

Press the console ON LINE/MAN REL switch once.

Press the CLEAR key.

If successful,

Enter data, test functions of keyboard control keys, and enter several lines of characters.

TEST 2. DISPLAY SCREEN TO LINE PRINTER

If data entered in Test 1 is properly displayed, a preliminary check of the line printer may be made by sending the screen image to the printer.

Verify that the line printer is in the ready condition (section 2).

On the 217-11 and 217-12 terminal keyboards, press the AUX SEND key.

On the 217-13 and 217-14 terminal keyboards, press the PRINT key.

The screen image should be reproduced line-by-line on the line printer.

TEST 3. CARD READER TO LINE PRINTER

To test card reader and line printer functions:

Turn the ON/OFF switch on the card reader to ON.

Set the AUTO/MAN switch to AUTO.

Press MAN/REL button on the display console once to clear the card buffer.

Place the cards to be read in the right-hand feeder, face down with 12-edge toward the user.

Press the REG or READY switch on the reader to register the first card.

Verify that the line printer is in the ready condition (section 2).

Press the display console LIST switch.

Cards should begin feeding and should be displayed on the printer, one card image to a line.

If any of the off-line tests fail to execute normally, a hardware problem exists at the remote site. There is no need to continue testing. Contact hardware maintenance personnel for diagnosis and repair.

If the card reader or line printer is inoperative, it may still be possible to use INTERCOM with display console input/outout.

ON-LINE TESTS

The following tests can be performed only if the user is logged into the INTERCOM system.

TEST 4. KEYBOARD TO CENTRAL SITE TO DISPLAY SCREEN

This test determines the status of the send/receive circuitry between the remote and central sites.

In INTERCOM command mode (section 3) enter:

```
COMMAND- ASSETS.
```

The response should be the normal display, on the screen, of the users assets. If the displayed data is erroneous, the send or receive circuits may be faulty.

TEST 5. CENTRAL SITE TO DISPLAY SCREEN

To test the operability of the receive circuit at the central site, contact the central site operator, provide him with the user identification assigned to your terminal at LOGIN, and request that he send a message to your terminal. Under normal conditions, a message should be received.

TEST 6. KEYBOARD TO CENTRAL SITE LINE PRINTER

A utility program, TESTLP, is provided to check the operability of the send circuit at the central site. In INTERCOM command mode, enter:

```
COMMAND- TESTLP.

COMMAND- DISPOSE,TESTFIL,PR.
```

Under normal conditions, the file TESTFIL should be printed at any available central site line printer. Voice communication with the central site may be necessary to confirm that the file was printed.

TEST 7. CARD READER TO CENTRAL SITE TO REMOTE LINE PRINTER

To test remote equipment operation as well as the communication lines between the central and remote site:

Load cards into card reader according to the first five steps in Test 3.

In INTERCOM command mode, enter:

```
COMMAND- READ,CARDS.

(clear screen when card reader completes reading cards)

COMMAND- COPYSBF,CARDS,FILEA.

COMMAND- DISPOSE,FILEA,PR=I(user-ident)

COMMAND- GO.

COMMAND-
```

The card images should be printed at the remote line printer, one card image per line.

TEST 8. CARD READER TO CENTRAL SITE LINE PRINTER

To further check the operability of the send circuit and card reader:

Load cards into card reader according to the first five steps in Test 3.

In INTERCOM command mode, enter:

```
COMMAND- READ,CARDS.

COMMAND- COPYSBF,CARDS,FILEB.

COMMAND- DISPOSE,FILEB,PR.

COMMAND-
```

The card images should be printed at the central site. Voice communication may be required with the central site to verify results of this test.

If the user is not successful in receiving reliable data, the final tests, 9 and 10, should indicate positively if the problem exists in the remote equipment or at the central site.

TEST 9. KEYBOARD TO USER AT ANOTHER REMOTE SITE

Contact the central site operator and obtain the names of other logged in users. Contact one of these users and request that they stand by to receive a message.

In command mode, enter:

```
COMMAND- SEND.

TO WHOM- user-name

TYPE MESSAGE OR END-

DID YOU GET THIS MESSAGE?

TYPE MESSAGE OR END-

END

COMMAND-
```

The named user should receive the message. Voice communication with the other user may be necessary to confirm that the message was received.

TEST 10. USER AT ANOTHER REMOTE SITE TO YOUR TERMINAL SCREEN

Contact another logged in user and request that he send a message to your terminal using the commands given in Test 9. Under normal conditions, a message will be received.

If all of the above tests execute normally, a software problem may exist either in the user's program or in the central site operating system.

INDEX

COMMENT SHEET

CONTROL DATA CORPORATION

TITLE: 6000 Series INTERCOM 3 Reference Manual

PUBLICATION NO. 60252800 REVISION A

Control Data Corporation solicits your comments about this manual with a view to improving its usefulness in later editions.

Applications for which you use this manual.

Do you find it adequate for your purpose?

What improvements do you recommend to better serve your purpose?

Note specific errors discovered (please include page number reference).

CUT ON THIS LINE

General comments:

FROM NAME: _____ POSITION: _____

BUSINESS ADDRESS: _____

_____

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
Documentation Department
215 Moffett Park Drive
Sunnyvale, California 94086

CUT ON THIS LINE

FOLD

FOLD

STAPLE

STAPLE