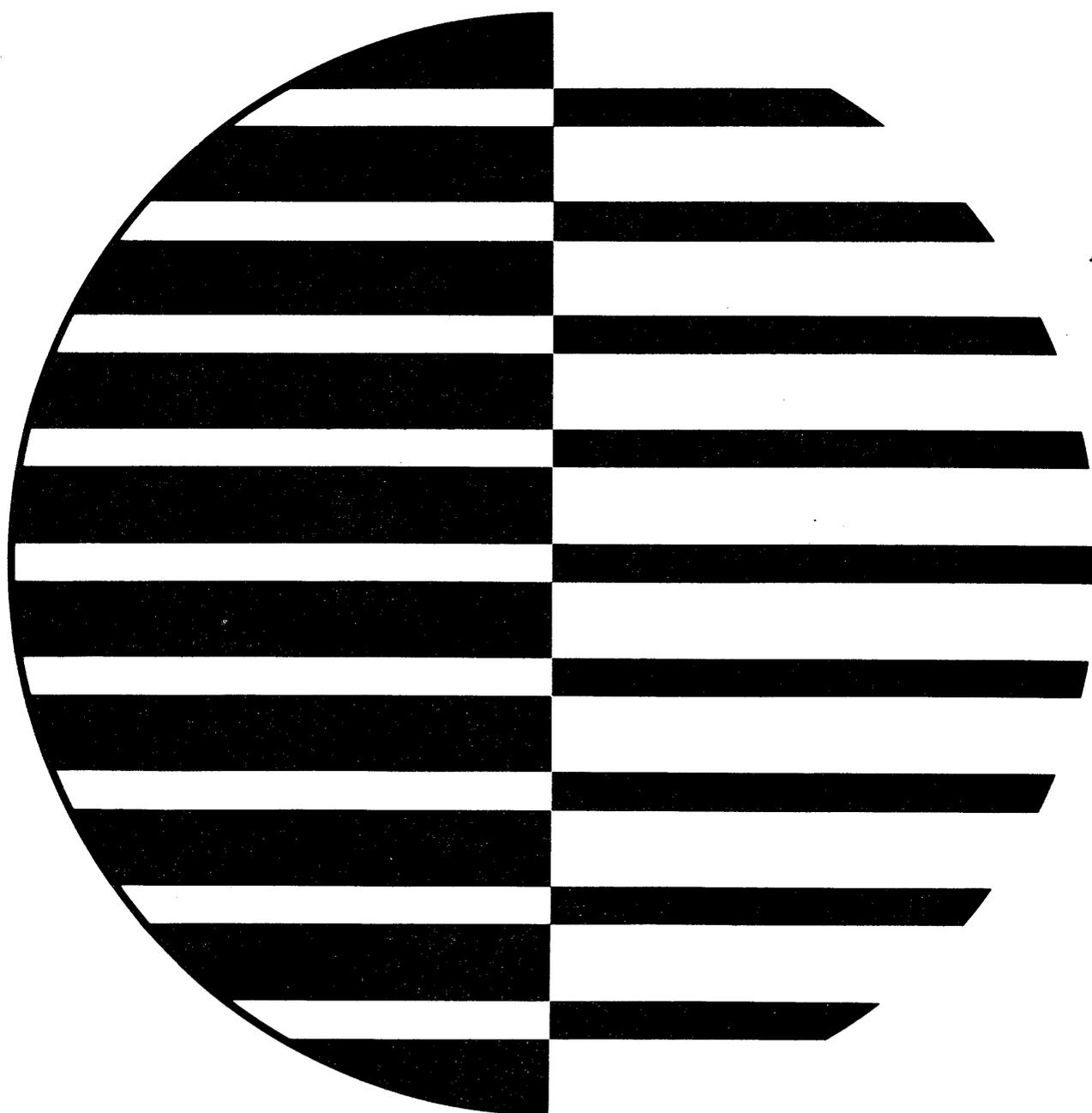


**CONTROL DATA<sup>®</sup>**

**6000 SERIES COMPUTER SYSTEMS**

**INTERCOM I Reference Manual**





# CONTENTS

INTRODUCTION	v	4 SETUP	4-1
1 SYSTEM DESCRIPTION	1-1	Text Buffer	4-1
Terminals	1-1	Directives	4-2
Time Sharing	1-1	Entering SETUP	4-1
Input/Output	1-1	SYSTEM Directive	4-2
Job Submission	1-1	OLD/NEW Directive	4-3
Files	1-2	Exit from SETUP	4-4
2 TERMINAL OPERATION	2-1	Editing	4-5
Initiating Terminal	2-1	Line Numbers	4-5
Data Set Operation	2-1	RESEQ Directive	4-6
Dedicated Line	2-3	TRANS Directive	4-7
Operation	2-3	TAB Directive	4-7
Teletype Terminal	2-3	Line Editing	4-8
Teletype Keyboard	2-3	Intra-line Editing	4-8
Display Terminal	2-5	DELETE Directive	4-10
Display Screen	2-5	Character Deletion	4-10
Display Keyboard	2-7	File Storage and	
3 COMMAND MODE	3-1	Manipulation	4-12
INTERCOM I Commands	3-1	SAVE Directive	4-12
Abbreviated Forms	3-1	UNSAVE Directive	4-12
Communication with		RENAME Directive	4-12
INTERCOM I	3-1	LIST/S Directive	4-13
LOGIN	3-2	SCRATCH Directive	4-14
LOGOUT	3-3	PERMANENT Directive	4-14
TEACH	3-2	Compilation and	
Communication between		Execution	4-16
Terminals	3-5	Compile Only	4-16
SITUATE	3-5	BASIC Compile/	
LOCK/UNLOCK	3-5	Execute	4-16
SEND	3-5	FORTRAN RUN	
File Control	3-7	Compile/Execute	4-16
FILES	3-7	FORTRAN EXTENDED	
PAGE	3-7	Compile/Execute	4-17
DROP	3-8	COBOL Compile/	
CONNECT	3-9	Execute	4-17
DISCONT	3-10	KILL Directive	4-19
Password File	3-11	APPENDIX A HARDWARE	
Parameter Cards	3-12	CONFIGURATION	A-1
ASSETS	3-13	APPENDIX B ERROR MESSAGES	B-1
ETL/EFL	3-14	APPENDIX C COMMAND MESSAGES	C-1
Submitting Job	3-15	APPENDIX D SETUP DIRECTIVES	D-1
BATCH	3-15	APPENDIX E CODE CONVERSION	
SETUP	3-17	TABLE	E-1
Diagnostic Messages	3-18		

## FIGURES

2-1	Model 33 Teletype	2-2	2-4	Display Standard Keyboard	2-7
2-2	Model 35 Teletype	2-2			
2-3	Teletype Standard Keyboard	2-4	A-1	Typical INTERCOM I Configuration with 7 Terminals	A-2

## INTRODUCTION

---

The INTERCOM I 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 I permit the terminal operator to submit a job for processing on the central computer and also to work interactively with an executing program.

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

The user of INTERCOM I should be familiar with the SCOPE operating system and with the language of his source program. INTERCOM I commands and directives are described in this manual. The system responses are self explanatory and act as a guide to the terminal operator. In addition, a command is available to request a short course in INTERCOM I commands at the terminal.



# SYSTEM DESCRIPTION

1

---

## TERMINALS

A terminal in the INTERCOM I system is one of the following:

Model 33 Teletype Terminal

Model 35 Teletype Terminal

Model 217-2 CRT Display Terminal

Each terminal has a keyboard for entering instructions to the central computer. As each instruction is entered, it is displayed on a teletypewriter at the Teletype terminals or on the screen of a display terminal. Operation of the terminals is described in section 2.

## TIME SHARING

Effectively, a time sharing system gives each user sole occupancy of the central computer. Actually, programs for several users may be running serially, in turn, for short periods of time. Each program uses the computer only for the time it is active and within the bounds of a specified time slice. A time slice is the maximum time allocated to each program during which it may operate without interruption. Program swapping allows programs to be read into or written from peripheral storage areas, so that separate programs can share the same area in central memory at different times. Thus, several terminal users may have access to the central computer at the same time.

## INPUT/OUTPUT

The terminals do not have offline input/output equipment. Cards may be punched or read and output printed on central site peripheral equipment under control of commands issued from the terminals.

Input and output for an executing program may be routed via the terminal, so that the user can influence the program through his input to the program.

## JOB SUBMISSION

A job can be constructed at the terminal, compiled, and then executed using the SETUP command; or an existing program file can be submitted to the SCOPE batch processor with the BATCH command. SCOPE control cards can be entered directly as commands; or a control card file can be created, saved, and later referenced using SETUP. All SCOPE control cards except REQUEST, EDITLIB, COMMON, and the console command DIS may be entered as commands.

## FILES

INTERCOM I has four types of file:

User private files	System files
Common files	Permanent files

User private files are local files; they are created by the individual user and can be read, altered, or deleted only by the originator.

Common files are local files which originated as user private files but have been specified as common files by the originator. That is, they may be read, altered or deleted by all other users of INTERCOM I.

System files exist in the SCOPE library. They may be loaded by any user with an adequate access level, but they may not be modified or deleted.

Permanent files are mass storage files, the location and identification of which are always known to the INTERCOM I 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 deadstart, until the user decides to release them.

Files are created and altered by the INTERCOM I user under control of SETUP. Existing files can be dropped by an INTERCOM command or under SETUP control. Disposition of local files (private, common) is specified under control of the BATCH command.

INTERCOM I files are files used by SCOPE. Limitations on file organization are the same for SCOPE and INTERCOM I.

---

Teletype and display terminals have somewhat different operating procedures and keyboards. In addition, terminals connected to the central computer via data sets are initiated differently from terminals connected via dedicated lines. Operating procedures and keyboards for standard Teletype 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 Teletype 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. The LINE/BLOCK switch on the display console must be set to LINE, and the ATTENDED/UNATTENDED switch on the display keyboard must be set to ATTENDED.

### DATA SET OPERATION

If a terminal is connected to the central site with a 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-PHONE data set receiver from its cradle and press the TALK button.
2. Dial the telephone number of the line to be used; the telephone will ring.
3. If the phone is answered with a high-pitched tone, press the DATA button on the phone and replace the receiver. The terminal is connected to the computer.
4. A normal telephone busy signal indicates that a line is already in use.

#### Teletype Terminal

The action required to connect a Teletype terminal with the computer differs slightly between the model 33 and model 35 teletypewriters (figures 2-1 and 2-2).

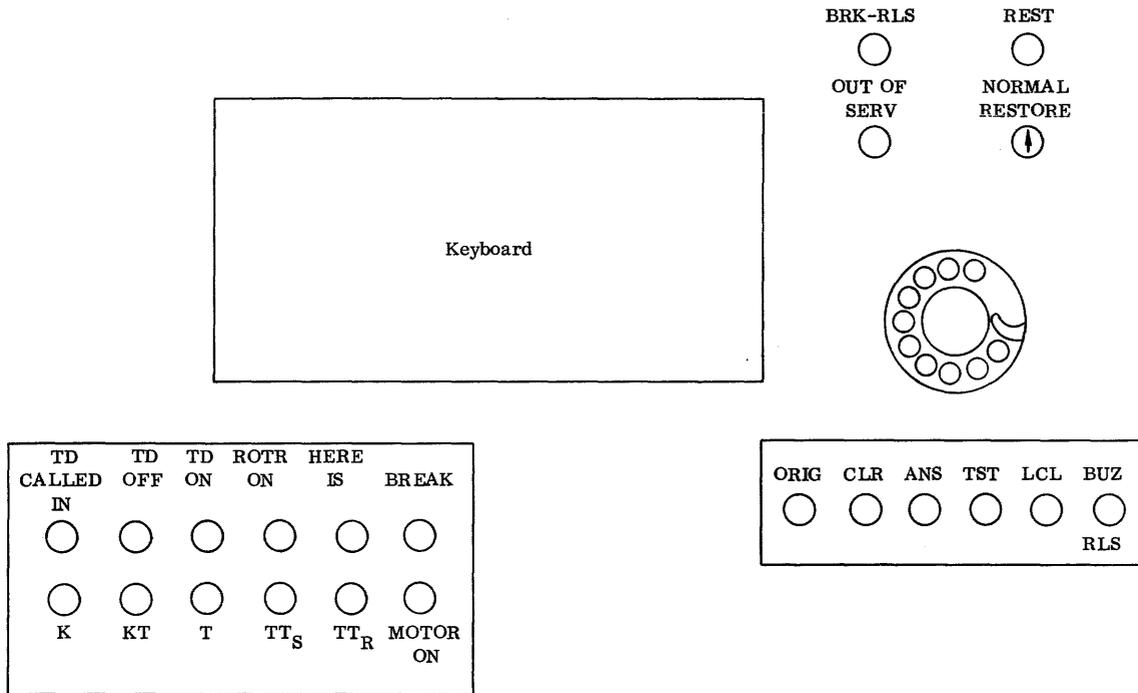


Figure 2-1. Model 33 Teletype

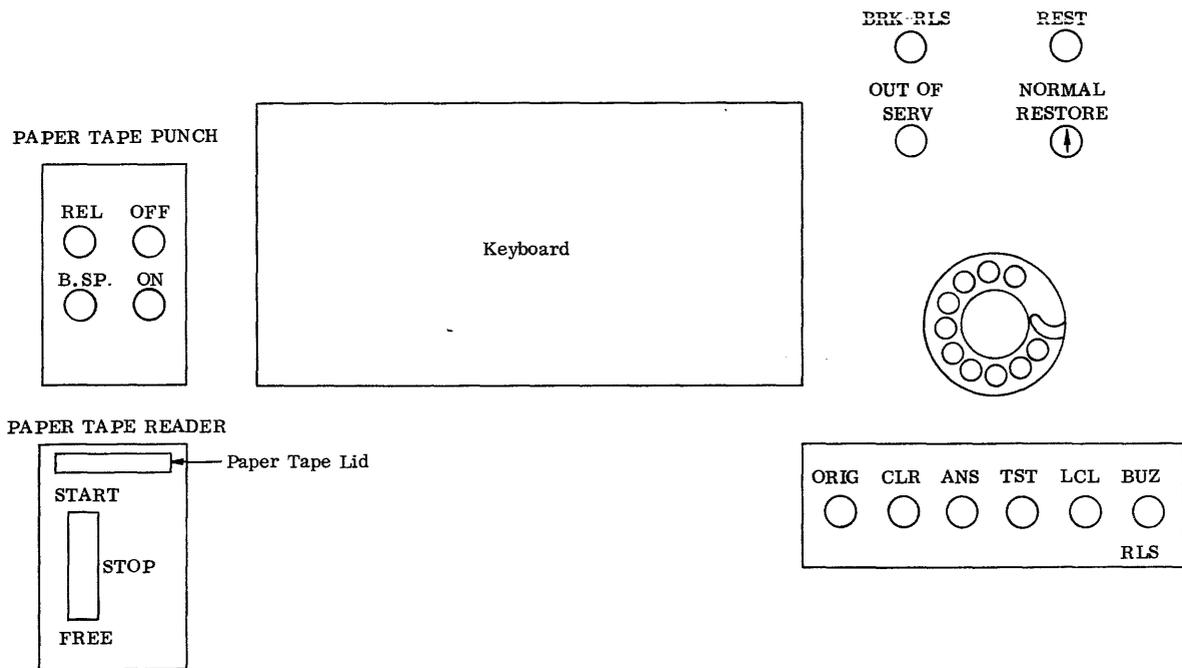


Figure 2-2. Model 35 Teletype

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. If a high-pitched, not busy signal is returned:

For the Model 33: Press the TALK button to connect the terminal.

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

4. A busy line is indicated by a normal telephone busy signal.

If the Teletype console does not correspond to either figure 2-1 or 2-2, consult a representative of the company supplying the Teletype for the exact procedure to follow.

### **DEDICATED LINE OPERATION**

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

### **TELETYPE TERMINAL**

Teletype communication with INTERCOM I takes place through a Teletype model 33 or model 35. Private line Teletypes are installed with the EIA RS232 Data Set Coupler interface.

When a Teletype key is pressed a single character is transmitted to the central sit on the communication lines. The maximum rate of transmission is 10 characters per second.

### **TELETYPE KEYBOARD**

Model 33 or model 35 Teletype keyboards (figure 2-3) resemble a standard typewriter keyboard. Some keys, however, have a different function, and some are new. The keys used and interpreted by INTERCOM I are described below; information is included as to what is stored, what action is taken, and what is printed on the Teletype listing.

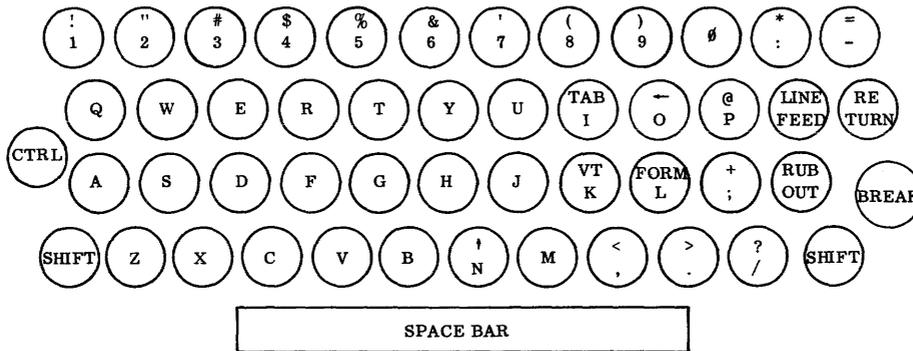


Figure 2-3. Teletype Standard Keyboard

- SHIFT This key accesses the characters or functions shown on the upper portion of the Teletype keys (except TAB accessed by CTRL). If pressed alone, SHIFT is ignored and nothing is stored or printed.
- CTRL This key is used to access the TAB function. If pressed alone, it is ignored and nothing is stored or printed.
- RETURN The carriage return key signals the end of a command or an input record. It also returns the Teletype printer carriage to its leftmost position. No character is stored or printed.
- LINE FEED This key spaces to the next line and returns the carriage to the start of that line. No character is stored or printed.
- TAB (CTRL and TAB/I) TAB requests a skip to the next logical tab position as defined by the current TAB. The correct number of blanks are stored but none is printed. One ↑ is printed to indicate the tab request. If no tab has been specified, a blank is stored.
- SPACE Pressing the space bar generates the space character recognized as a delimiter between command parameters. A blank is stored and printed.
- ← (SHIFT and /O) ← is the backspace character which signals INTERCOM I to erase the previous character from its text 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 RUNY ← X. The corrected command RUNX is entered in the text buffer. No character is stored; a ← is printed on the Teletype listing.

#  
(SHIFT and #/3)

# interrupts execution of a program while the terminal is in SETUP mode. It returns the terminal to command mode.

BREAK

This key is used to recover from a temporary disconnect in the line. When a line is disconnected, the break light comes on. Pressing the BREAK key turns off the light. If the connection has been re-established, the light remains off and normal operation can continue. If the light comes back on, the disconnect condition is still present. Data is not transmitted while the line is disconnected. Nothing is stored or printed.

Alphanumeric

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

## DISPLAY TERMINAL

The display terminal is equipped with a cathode ray tube (CRT) display screen, a display controller, and a keyboard. The display terminal transmits entire records to the central site in contrast to the Teletype which transmits single characters.

### DISPLAY SCREEN

The display format is an installation option. The standard formats are 20 lines of 50 symbols per line or 13 lines of 80 symbols per line.

Two indicators appear automatically on the display screen to facilitate message transmission: the line indicator and the entry marker.

#### Line Indicator

A small block appears on the screen to the left of the line on which the next message will start. During read operations, the line indicator is affected by three keys on the display keyboard: SEND INDEX advances the indicator one line, CLEAR and RESET reposition the line indicator to the top line. The line indicator moves automatically to the next line at the completion of a write operation.

#### Entry Marker

An 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

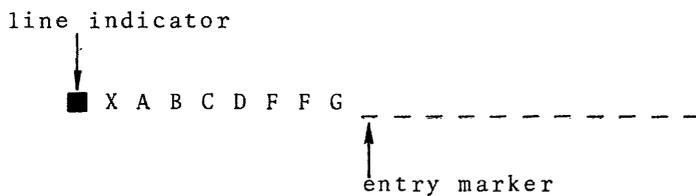
in the series becomes the entry marker. When the end of a line is reached, the series of underlines is displayed on the next lower line. When the end of the last line is reached, the underlines reappear on the top line. Thus the line indicator and the entry marker enclose the message to be sent.

Message Terminator

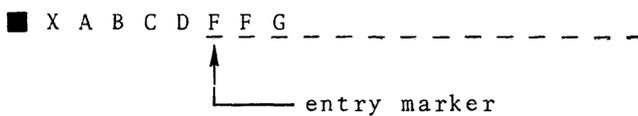
The delta symbol,  $\Delta$ , is used to indicate the end of a message. This symbol is entered when the SEND key is pressed during a write operation, and the entry marker moves one position to the right of the  $\Delta$ .

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

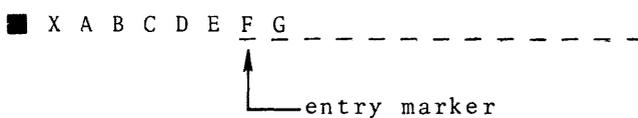
Assume the letter E in the following string of characters has been improperly entered as F.



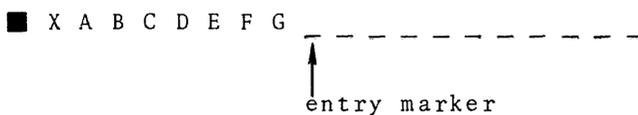
Press BKSP key three times to move entry marker under F:



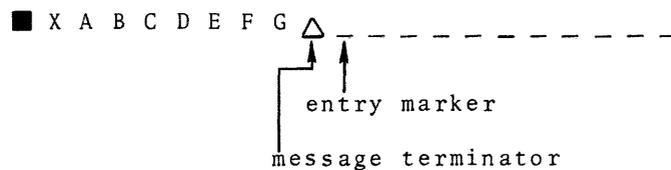
Correct F by entering E:



Enter two SKIPS to reposition the entry marker:



If SEND key is pressed:



appears on the display, and the message ABCDEFG is sent.

## DISPLAY KEYBOARD

The display keyboard (figure 2-4) resembles a standard typewriter keyboard. Some keys have a different function, and some are unique to the display keyboard. The keys used and interpreted by INTERCOM I are described below; information is included as to what is stored, what action is taken, and what is printed on the display screen.

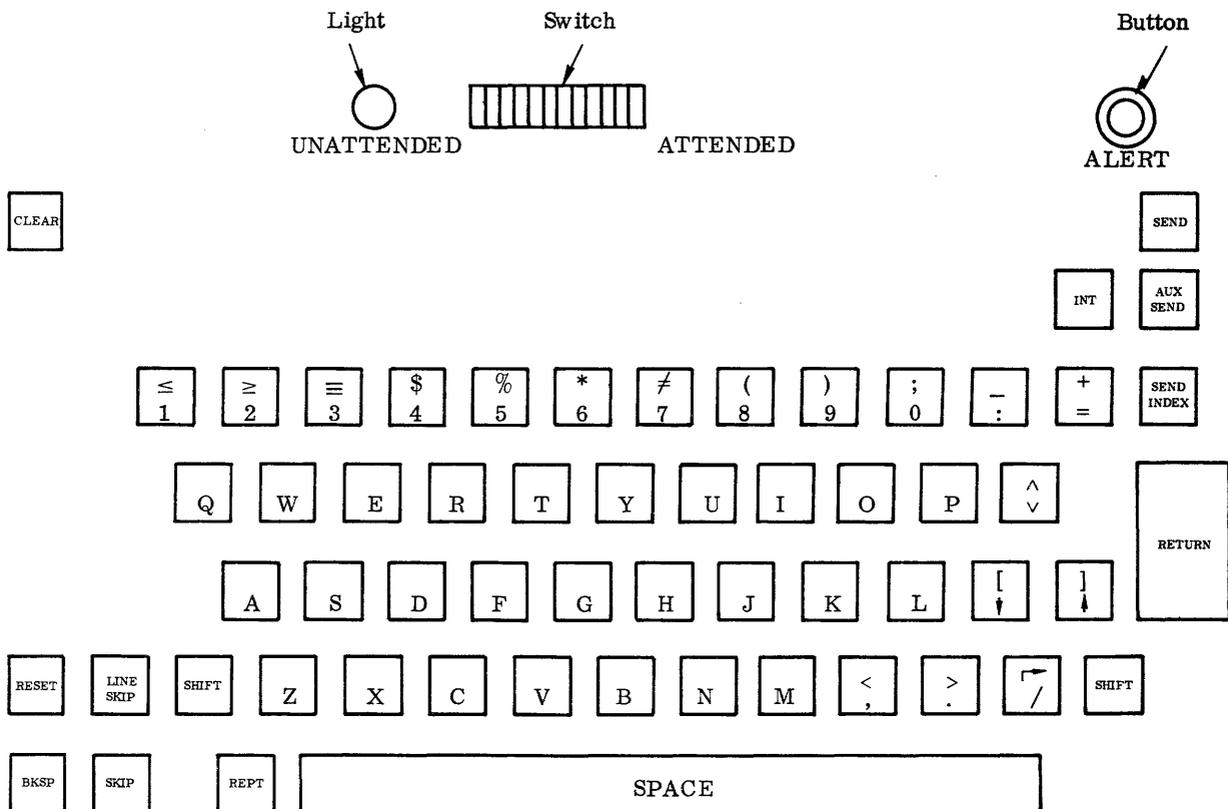


Figure 2-4. Display Standard Keyboard

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

SEND This key initiates transmission of a message from the terminal to INTERCOM I. It follows a command or an input record. When SEND is pressed, the symbol  $\Delta$  appears on the display at the end of the message; and the information between the line indicator and  $\Delta$  is transmitted. SEND is a signal to INTERCOM. Nothing is stored.

RETURN The carriage return key advances the entry marker one line so that typing can continue on the next line. Unlike the RETURN on the Teletype keyboard, it does not signal the end of a command. Nothing is stored.

↓ This key requests tabulation to the next TAB position specified in the FORMAT. Tabulation is not apparent on the display screen, and no input/output occurs until the SEND key is pressed to transmit the line. However, spaces are entered into the character string when it is transmitted to INTERCOM I. ↓ is displayed at the terminal, but if no tab was specified, a blank is stored.

SPACE The SPACE bar generates a blank space on the screen which is recognized as a delimiter by SETUP.

CLEAR This key clears the display screen so that the user can begin typing on the first line. It must not be pressed while the terminal is receiving output or when the user is entering input. Nothing is stored.

BKSP With this key, the user can backspace and overwrite 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 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 overwrites any previously entered characters.

REPT This repeat 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.

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.

SEND INDEX This key moves the line indicator down one line. Nothing is stored.

RESET This key moves the line indicator and 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.

%  
(SHIFT and %/5) The % key followed by the SEND key is pressed to interrupt execution of a user's program. It causes execution to terminate and the terminal is returned to command mode.

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



---

## INTERCOM I COMMANDS

All user commands are terminated with a period and entered by pressing the SEND key at the display terminal keyboard or the RETURN key at a Teletype terminal keyboard. In this manual, the symbol  $\Delta$  is used wherever the user should press the SEND or RETURN key.

A command may be any of the INTERCOM I commands described in this section, or any SCOPE control card format except REQUEST, EDITLIB, COMMON or the console command DIS. An INTERCOM I command calls for control by an INTERCOM I utility program; entering a SCOPE control card format as a command places the system under control of SCOPE. The utility program called by a command remains in control until the requested operation has been performed. It then returns control to INTERCOM I command mode, and the user may enter another command. Return to command mode is signalled by the system response:

COMMAND-

## ABBREVIATED FORMS

Some commands require a system response before the user can enter further information. Others allow an abbreviated form where additional information, such as file names, can be entered along with the command. Such cases are noted when the command is explained.

## COMMUNICATION WITH INTERCOM I

To enter the INTERCOM I system, the user logs in with the LOGIN command. He is then in command mode and may enter another command. The user exits from the INTERCOM I system with the command LOGOUT. He may request the system to review the operation of INTERCOM I with the TEACH command.

## LOGIN

The remote terminal user logs into the system by entering:

```
LOGIN.Δ
```

The system responds:

```
TYPE VALID USER NAME-
```

The user name may be up to ten alphanumeric characters, and must not be followed by a period. When it has been entered, the system responds:

```
TYPE VALID PASSWORD-
```

The user then enters his password. A password is up to 10 alphanumeric characters which must not contain or terminate with a period. The user name and password are entered in a password list maintained by INTERCOM I where they comprise a unique identifier for the logged in user. On a Teletype listing, the password is overprinted; so it will remain private.

Example:

```
COMMAND - LOGIN.Δ
```

```
TYPE VALID USER NAME - JOHNDOE Δ
```

```
TYPE VALID PASSWORD - XK9371Δ
```

```
12/09/69 LOGGED IN AT 00.34.15
```

```
COMMAND-
```

## LOGOUT

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

```
LOGOUT.Δ
```

INTERCOM responds:

```
YOU NO LONGER OWN ANY PRIVATE FILES.
```

When the user logs out his private files are released. Only permanent files are retained between the time of a LOGOUT and any subsequent LOGIN.

The terminal is then disassociated from INTERCOM I until a subsequent LOGIN command is entered.

INTERCOM I then displays the date and time the terminal was disconnected.

Example:

```
LOGOUT.Δ
YOU NO LONGER OWN ANY PRIVATE FILES.
CP TIME      .055
PP TIME      3.233
12/09/69 OFF AT *00.25.16.
```

The date is in the form month, day, year; the time in hours, minutes, seconds.

## TEACH

A command can be entered to request a lesson in the basic commands and directives available to the INTERCOM I user. The user types:

```
TEACH.Δ
```

The system displays a series of multiple choice questions on all aspects of INTERCOM I. After each question the system waits for a user response, and then responds with CORRECT, or a reply indicating which answer is correct.

Example:

The answers given below are wrong except in the case of question 2.

```
COMMAND-TEACH.Δ
```

```
THIS IS A PROGRAM TO TEACH SCOPE/INTERCOM COM-
MANDS AND UTILITIES. ANSWER THE QUESTIONS BY
TYPING THE APPROPRIATE LETTER.
```

1. IN WHAT ORDER MUST YOU PROCEED TO GAIN  
ACCESS TO SCOPE/INTERCOM
  - A. TYPE LOGIN, TYPE PASSWORD, TYPE USERNAME
  - B. TYPE LOGIN, TYPE USERNAME, TYPE TERMINAL SITE,  
TYPE PASSWORD
  - C. TYPE USERNAME, TYPE PASSWORD
  - D. TYPE LOGIN, TYPE USERNAME, TYPE PASSWORD

```
A Δ
YOU GOOFED, THE ANSWER IS D
```

2. A SCOPE/INTERCOM COMMAND
  - A. IS FOLLOWED BY A PERIOD.
  - B. IS ENTERED AFTER COMMAND - IS DISPLAYED  
ON THE TERMINAL.
  - C. CANNOT BE ENTERED WHILE UNDER SETUP.
  - D. A AND B
  - E. ALL OF THE ABOVE

```
D Δ
CORRECT
```

3. THE COMMAND SITUATE
- A. DISPLAYS THE USER NAMES IN THE USERNAME TABLE.
  - B. DISPLAYS THE USERNAME/PASSWORD COMBINATIONS OF THOSE CURRENTLY LOGGED IN.
  - C. DISPLAYS THE USERNAMES OF THOSE CURRENTLY LOGGED IN.
  - D. DISPLAYS THE PASSWORDS OF THOSE CURRENTLY LOGGED IN.

E Δ

SORRY ABOUT THAT, THE ANSWER IS C

4. MESSAGES CAN BE SENT TO OTHER USERS
- A. ONLY IF BOTH ARE UNDER SETUP.
  - B. BY TYPING THE COMMAND SEND, GIVING THE USER NAME OF THE RECEIVER, AND TYPING THE MESSAGE.
  - C. ONLY IF BOTH SENDER AND RECEIVER ARE LOGGED IN.
  - D. A AND B
  - E. B AND C

W Δ

SORRY, THE ANSWER IS E

5. A FILE CAN BE COMMONED
- A. UNDER THE BATCH COMMAND WITH COMMON AS THE DISPOSITION.
  - B. UNDER THE SETUP DIRECTIVE COMMON
  - C. UNDER THE SCOPE/INTERCOM COMMAND COMFILE.
  - D. NONE OF THESE.

R Δ

YOU GOOFED, THE ANSWER IS A

The user presses the # key to stop the TEACH utility and return to command mode.

#

USER ABORT  
COMMAND-

## COMMUNICATION BETWEEN TERMINALS

Users can communicate with other terminals if the terminals are logged in and the user name is known.

### SITUATE

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

```
SITUATE.Δ
```

A list of user names is displayed at the terminal; but neither the terminal location nor the password is included in the list.

### LOCK/UNLOCK

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

```
LOCK.Δ
```

When a user intends to execute a program with input and output data routed through his terminal, he would use LOCK to prevent messages being mixed with his data.

When he wants to receive messages again, he enters:

```
UNLOCK.Δ
```

Messages sent to him by other users will be received at his terminal. Messages sent while a user is locked out are not saved.

### SEND

To send a message, the user enters:

```
SEND.Δ
```

The system responds:

```
TO WHOM-
```

The user may enter the name of the user to whom he is sending the message, or if he enters:

```
***Δ
```

The message will be sent to all logged in users including those requesting no messages with the LOCK command. If a specified user has locked out all messages, the system sends the message:

```
THIS USER IS LOCKED OUT
```

The system requests TO WHOM again.

After the user responds to this system request, the system sends the request:

TYPE MSG OR END-

The user enters the message terminated with the word END. If he wants to wait until a later time, he simply enters ENDΔ. END is recognized as a terminator only if it is the first three characters and the fourth character is a Δ. Thus, the word END may be embedded in a message without terminating it prematurely.

Examples:

1. SITUATE.Δ  
JOHN-BROWN TSU-GRPA CDC-JGM  
SEND.Δ  
TO WHOM-  
\*\*\*Δ Message is sent to all logged in terminals.  
TYPE MSG OR END-  
I AM TAKING FILA FROM LIST OF COMMON FILES AND MAKING IT  
MY PRIVATE FILE Δ  
SEND ANY OBJECTIONS TO JOHN-BROWNΔ  
ENDΔ
2. SEND.Δ  
TO WHOM-  
JOHN-BROWNΔ  
THIS USER IS LOCKED OUT  
TO WHOM-  
TYPE MSG OR END-  
ENDΔ

## FILE CONTROL

Private files may be listed or dropped with INTERCOM I commands. In addition, designated files may be routed between the terminal and the central computer.

To create and edit files, the user must enter SETUP and use the SETUP directives (section 4). A file may be made private or common, with the BATCH command.

## FILES

With the FILES command the user requests a list of his private files:

```
FILES.Δ
```

The system response is:

```
-PRIVATE FILES-
```

```
(list of private file names)
```

This list names all private files including permanent files available for use and modification by the terminal user. Private files are associated with the name/password combination entered by the user in his LOGIN command. Permanent files are preceded by \*.

## PAGE

The PAGE command is used only at the CRT terminal. It reads the specified file and displays a full screen of data at a time. The screen size is selected as an installation option. It may be either 13 lines by 80 characters or 20 lines by 50 characters.

Display of a page of a file is requested by typing:

```
PAGE (filename)Δ
```

The file name specified must be an existing file to which the user has access. The next page may be requested by pressing the plus (+) or minus (-) key on the display keyboard.

```
+ advances the file one page
```

```
- turns back the file one page
```

Any character other than + or - terminates the page routine.

A file may be advanced page by page until an end of file is reached, when the system issues the message:

```
AN EOF FOUND ON FILE
```

The PAGE command may not be used to examine the contents of the SETUP text buffer prior to execution. The LIST directive of SETUP should be used for this purpose. Contents of the SETUP text buffer can be paged, however, after initial execution is completed by entering: PAGE.Δ

No file name need be specified; the current text buffer file is paged. The procedure described above for private files is used for advancing, turning back, or terminating the paging process.

Examples:

Given a screen size of 13 by 80:

PAGE (FILEA)Δ First 13 lines are displayed for a screen.

+ Δ The next 13 lines are requested;  
+ Δ and the next.

- Δ User looks at previous 13 lines then  
+ Δ pages ahead again  
+ Δ and again.

X Δ The paging process is terminated.  
COMMAND-

## DROP

When a user is in command mode, he can remove a private file from the disk with the DROP command. He enters:

DROP.Δ

The response is:

TYPE FILE NAME OR END-

The user enters one or more file names followed by END . File names must be in the list of private files returned in response to the FILES command. There is no limit to the number of file names that may be specified. The list must be terminated with the word END. The user is discouraged from using END as a file name. A file with the name END will not be dropped; it will only terminate the list at the point it appears.

Abbreviated Form:

```
DROP (filename1,filename2,...,filenamen,END)Δ
```

Example:

```
FILES.Δ           Requests a list of user's private files
-PRIVATE FILES-   System lists file names
  MYFILE          MASTER1   FILEA      *PFIL
```

Then the user may drop one or more of the listed files:

```
DROP(FILEA,MYFILE,END)Δ
```

and request a new list:

```
FILES.Δ
-PRIVATE FILES-
MASTER1      *PFIL           List containing the name of remaining
                             private files.
```

## CONNECT

The user can request that specific files be routed to the remote terminal each time they are read or written by entering the command:

```
CONNECT(filename1,filename2,...,filenamen)Δ
```

The file names specify files to be routed to the terminal. They may name the SCOPE system files, INPUT, OUTPUT, as well as user specified files.

When the INTERCOM user is working with FORTRAN (RUN or EXTENDED) source programs, the system files INPUT and OUTPUT are routed automatically to the terminal. He need not specify the CONNECT command. The system will proceed as if CONNECT had been entered.

If the file is INPUT or is a file being input, each time the file is referenced in the source program, the system waits for input from the terminal. Each time an output file named in the CONNECT command is referenced in the source program, the output is printed or displayed at the terminal. It is not saved.

Since input/output to the terminal is buffered, connecting the terminal does not affect the processing time.

## DISCONT

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

```
DISCONT(filename1,filename2,...,filenamen)Δ
```

The specified files will no longer be routed to the terminal; rather they will be routed to and reside on allocatable mass storage.

To ascertain if a response is required and also the type of response, PRINT should be used with the INPUT command of BASIC or the READ command of FORTRAN.

Examples:

1. CONNECT(INPUT,OUTPUT)Δ

```
100 FORMAT (*1 HOW DO YOU DO,*/ * PLEASE ENTER A THREE-DIGIT NUMBER,  
          */ * ANY NON-NUMERIC WILL TERMINATE - */)
```

```
101 FORMAT (3A1)
```

```
.  
. .
```

```
200 PRINT 100  
    READ 101
```

```
(users enters:  
 123Δ)
```

This command will cause system to wait for input from terminal. Printout or display resulting from the PRINT command indicates the input expected.

```
.  
. .
```

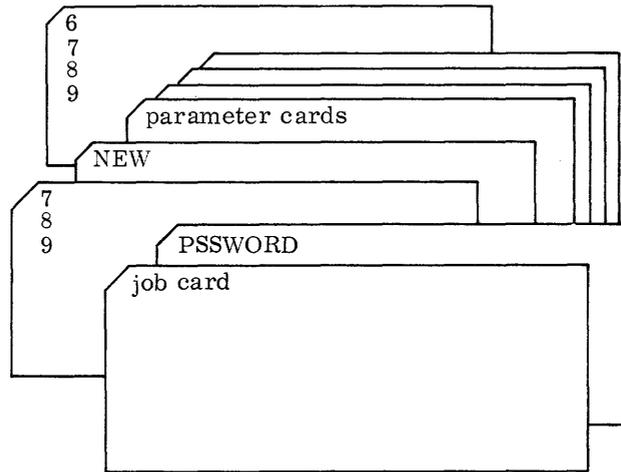
```
DISCONT(INPUT,OUTPUT)Δ These files will no longer be routed to  
and from terminal.
```

2. 100 PRINT 1  
 200 1 FORMAT (\*ENTER A, B, AND C\*)  
 300 READ 2, A, B, C,  
 400 2 FORMAT (3(F5,3,2X))

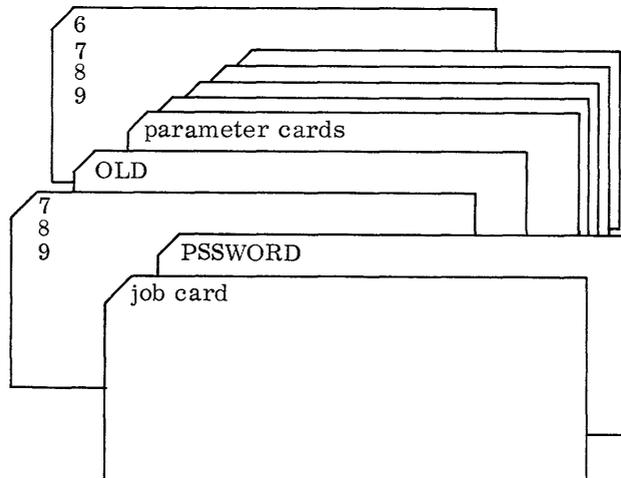
## PASSWORD FILE

A permanent user password file is created and maintained by INTERCOM I. This file is a requirement for using INTERCOM I. It may contain up to 64 entries, each defined by a unique user name/password combination. The file includes the user's maximum field length and time limit, the number of files allotted to him, and it keeps account of the processor time elapsed.

The password file can be called or altered only with a data deck entered as a batch job at the central site card reader. The following control cards are required to enter a new file:



The control cards to modify an existing password file:



## PARAMETER CARDS

One parameter card is required for each entry to be added to or deleted from the file.

To add an entry:

```
1 | 4
---|-----
ADD | U=username,P=password,F=length,T=time,A=level,N=nofiles
```

The user name and password parameters are required, the others are optional. If omitted, default values are supplied by INTERCOM I. Parameters may be specified in any order, but they must be separated by delimiters.

To delete an entry:

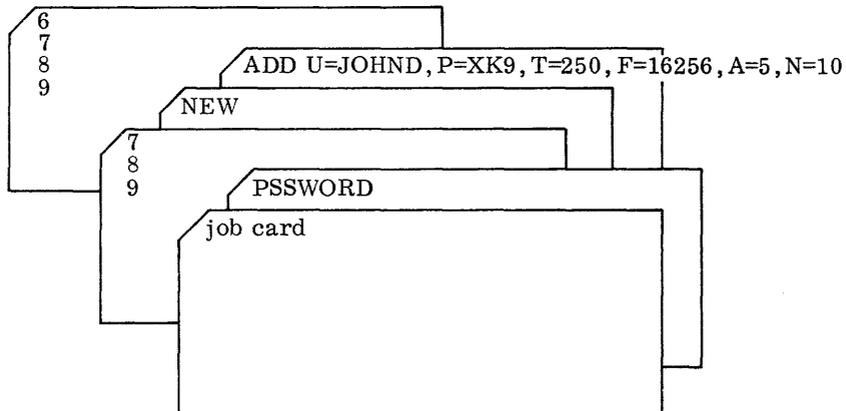
```
1 | 4
---|-----
DEL | U=username,P=password
```

The entry identified by the user name/password combination is deleted from the permanent password file.

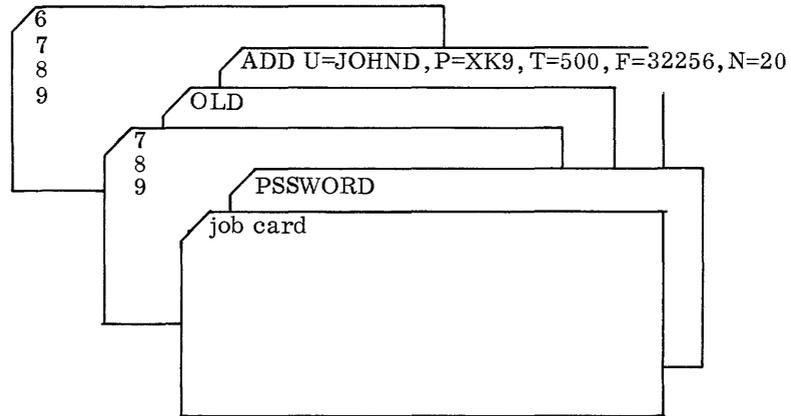
An existing password file can be modified by using an ADD card with a user name/password combination identical to one already existing in the password file; other parameters specify new values for the entry. The original entry for the user name/password combination is replaced by the new entry.

Examples:

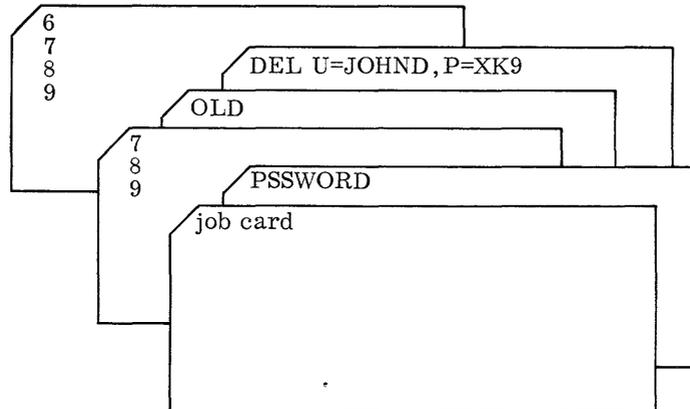
1. An entry is added to the password file:



2. The time limit, field length, and number of files of an existing entry are altered:



3. An entry is deleted from the password file:



### ASSETS

The user's status can be requested by entering:

ASSETS.Δ

The system replies with the following information:

Number of files used

Maximum number of files allotted

Maximum allowed field length

Maximum allowed time limit

Total elapsed central processor time  
Total elapsed peripheral processor time  
Current time of day

Example:

```
COMMAND- ASSETS.Δ  
FILE QUOTA      20  
FILES IN USE    1  
FIELD LENGTH 060000  
CP TIME        .024  
PP TIME        2.465  
TIME LIMIT     320  
CLOCK = *00.31.10.  
COMMAND-
```

### ETL/EFL

The user at the terminal can change his time limit and field length with two INTERCOM I commands.

He enters the new time limit, xxx, in octal seconds:

```
ETL,xxx.Δ
```

The user enters the new field length, yyyyy, as the number of words in octal:

```
EFL,yyyyy.Δ
```

These two commands affect the field length and time limit specified in the user's password file entry.

Example:

A user has the standard default values of 50000 words for field length and 500 seconds for time limit. To decrease these values, he enters:

```
ETL,350.Δ
```

```
EFL,40000.Δ
```

## SUBMITTING A JOB

To submit a job to the SCOPE batch queue, the user enters the BATCH command followed by options necessary to specify the program file and its disposition. Otherwise, he uses the SETUP command to enter his program directly to the INTERCOM I control point without the restrictions imposed by precise formatting.

### BATCH

Files are submitted one at a time to the SCOPE batch processing queue with the BATCH command. This command is entered by typing:

```
BATCH.Δ
```

The system responds:

```
TYPE FILE NAME-
```

The user then enters the file name of the file he wants placed in the batch queue. Only one name may be specified. It may name a private or common file, a system or permanent file; but the file must already exist on disk or the user cannot proceed. If the file is validated, the system responds:

```
TYPE DISPOSITION-
```

A number of dispositions are permissible. They describe what the user wants done with the named file: it can be placed in the input queue, prepared for card punching or line printing, renamed, made a common file or a private file.

When the user has entered the disposition, his file is placed in the batch queue and processed according to the requested disposition.

After the disposition has been entered, the system requests another file for batch 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 he has no more files for batch processing, he exits from the BATCH command with one of the following entries:

```
SCRAMΔ or ENDΔ
```

He is thereupon returned to command mode.

Abbreviated form:

```
BATCH(filename,disposition)Δ
```

## BATCH Command Dispositions

The following dispositions may be entered with BATCH:

<u>Disposition</u>	<u>Explanation</u>
INPUT△	Program file is placed in input queue at central site. The program must already contain the required control cards. The JOB card determines the field length and time allotment for the program.
PRINT△	File is placed in output queue at central site for subsequent printing on high speed line printer. The necessary carriage control characters must be the first character of every line to be printed.  Carriage control characters are supplied by FORTRAN (RUN and EXTENDED), BASIC and COBOL compilers. For other languages, the user, must supply carriage control characters. A blank character can be inserted with the SCOPE control card COPYSBF, (see SCOPE Reference Manual).
PUNCH△	File is released for subsequent punching on card punch at central site. Cards are punched in Hollerith code.
PUNCHB△	File is released for subsequent punching on card punch at central site. Cards are punched in binary.
PRIVATE△	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 is returned to common by specifying the COMMON disposition. If not reinstated as a common file, it remains in the user's private file list.
COMMON△	File is made accessible to all other users. User may no longer modify the file, unless he specifies PRIVATE, making it available for alteration as his own private file.
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.

Examples:

1. BATCH (MYFILE,INPUT)Δ  
TYPE FILE NAME -  
ENDΔ  
COMMAND -  
MYFILE is entered in input queue.  
User returns to command mode.
2. BATCH (BLIST,PRIVATE)Δ  
TYPE FILE NAME -  
SCRAM Δ  
COMMAND -  
BLIST is made user's private file.
3. BATCH (BLIST,RENAME)Δ  
ENTER NEW NAME -  
BUGLIST Δ  
TYPE FILE NAME -  
ENDΔ  
COMMAND -  
BLIST is renamed.
4. BATCH (AFILE,PUNCH)Δ  
TYPE FILE NAME -  
BFILE Δ  
TYPE DISPOSITION -  
INPUT Δ  
TYPE FILE NAME -  
CFILE Δ  
TYPE DISPOSITION -  
PRINT Δ  
TYPE FILE NAME -  
ENDΔ  
COMMAND -  
AFILE is punched on cards. System requests next BATCH file.  
BFILE is entered in input queue and next BATCH file is requested.  
CFILE is placed in output queue for subsequent printing.  
User returns to command mode.

## SETUP

The SETUP command introduces a utility program which permits the user to create and edit source programs and then enter them for compilation and execution at the INTERCOM I control point. The SETUP utility is entered with the command:

```
SETUP.Δ
```

Many sub-commands are available once the user has entered control of the SETUP utility. These sub-commands are called directives to distinguish them from INTERCOM I commands. The specific function of SETUP and its directives is described in section 4.

## DIAGNOSTIC MESSAGES

If the user is entering programs in the BASIC language (using BASIC directive of SETUP), diagnostics are sent directly to the terminal. Otherwise, he may request a diagnostic listing if his program terminates because of fatal errors.

A fatal error during execution is signaled by this message at the terminal:

```
FATAL ERROR xxxx
```

The error number in the xxxx field corresponds to an error condition listed in the execution diagnostics for the program.

If the program was compiled by the FORTRAN RUN compiler (RUNX directive of SETUP), the error listing is requested by entering:

```
PEANUTS.Δ
```

If the program was written in FORTRAN EXTENDED (FTN directive of SETUP) or COBOL (COBOLX directive of SETUP), an error listing can be requested with:

```
DIAGNOS.Δ
```

The listing produced by PEANUTS or DIAGNOS is an error diagnostic summary. This summary contains the header card of each program or subprogram containing an error, followed by the number of the line in which errors have occurred. Each erroneous statement is printed followed by a message such as:

```
****IDnn IMPROPERLY NESTED DO LOOPS (FATAL)
```

ID is followed by a two-letter code corresponding to the code in the list of diagnostics provided with the FORTRAN, FORTRAN EXTENDED, or COBOL reference manuals. Each message indicates whether the error is fatal or non-fatal. Execution terminates only if a fatal error is encountered. Therefore, correction of non-fatal errors is not imperative.

---

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

- Enter a program previously stored on disk
- Construct a program file line by line from the terminal
- 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. The SETUP utility alters the lines according to the format of the system under which it is running.

## 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 disk, 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 at any time.

The standard buffer size is determined when INTERCOM I is installed. Should a program be larger than the buffer size, the portion to be manipulated can be called into the buffer, while the rest of the program is stored on disk 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 there are no compilation errors. If compilation produces error diagnostics, the user can correct the program and resubmit it because the contents of the buffer are not destroyed.

A file in the buffer can be stored on disk for later use with the SAVE directive. Private and common files are saved even if the user logs out, but only permanent files are saved across deadstart.

## DIRECTIVES

The operation of SETUP is controlled by directives which specify the action the user wants to take. Directives are often followed by one or more options which will be requested by a system response 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.

Delimiters that may be specified in SETUP directives are: space, slash, period or comma. In this manual, wherever one of these delimiters is shown in a format, any of the other legal delimiters may be substituted.

## ENTERING SETUP

The SETUP utility is entered with the command:

```
SETUP.Δ
```

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.

## SYSTEM DIRECTIVE

Three system modes are available: BASIC, FORTRAN or GENERAL. In the BASIC mode, the user enters programs 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; or he can enter SCOPE control cards. COBOL is entered in the mode specified as GENERAL.

FORTRAN is the default mode if SYSTEM is not specified. To operate in a mode other than FORTRAN, the user must issue the SYSTEM directive:

```
SYSTEMΔ
```

The system responds:

```
NEW SYSTEM-
```

Then the user specifies either:

```
BASICΔ or GENERALΔ
```

Abbreviated Entry:

```
SYSTEM/optionΔ
```

## 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 created.

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 disk, the system replies:

NO FILE

Abbreviated Entry:

OLD/filename△

To construct 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 must be acceptable to the system he is using. If the file name is acceptable, the system responds:

READY.

The previous contents of the text buffer are lost.

If the file name is unacceptable because a file with the same name already exists on disk, the system response is:

DUPLICATE FILE NAME-

The user may reenter the NEW directive with another file name.

Examples:

```
1.  SETUP.△  
    ON AT *10.31.10  
    SYSTEM---FORTRAN  
    NEW OR OLD FILE-
```

NEW△

NEW FILE NAME-

MYFILE△

READY.

A new FORTRAN file called MYFILE can be constructed now by the user.

2. SETUP.△

ON AT \*05.15.50  
SYSTEM --- FORTRAN  
NEW OR OLD FILE-

SYSTEM△

NEW SYSTEM-

BASIC△  
OLD△

OLD FILE NAME-

BFILE△

NO FILE

The user requested a BASIC file previously stored on disk, but the system cannot find the file. User reenters the OLD directive with the correct file name:

OLD/B-FILE△

System responds:

READY.

3. SETUP.△

ON AT \*12.32.15  
SYSTEM---FORTRAN  
NEW OR OLD FILE-

SYSTEM/GENERAL△  
NEW/CNTRL1△

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 blank space, the first non-numeric character, or after five sequential digits have been encountered.

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))Δ

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

3	7	7	7
100	FORMAT (1H1,8(5XF10.8))	FAC	12345

#### BASIC

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

1
100 FOR X = 1 TO 15 Δ

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

1	7	7	7
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)Δ
```

SETUP reformats it as:

1	7	7
REWIND(FACTOR)	3	6
	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. A user writing in BASIC may not resequence line numbers but he may use the directive TRANS to transfer them to columns 73-76 where they are available for editing.

## **RESEQ DIRECTIVE**

The SETUP line numbers of a FORTRAN program or a SETUP file in the GENERAL mode can be resequenced simply by specifying:

```
RESEQ Δ
```

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

Because SETUP line numbers are also BASIC statement numbers, this directive does not apply to programs written in BASIC. Resequencing would destroy the flow of a BASIC program.

Example:

The user has input:

```
100 PROGRAM A(OUTPUT)Δ
120 PRINT 2Δ
115 2 FORMAT (*ABCDE*)Δ
```

He then resequences and lists the resequenced text buffer:

```
RESEQ Δ
LIST Δ
100 PROGRAM A(OUTPUT)
115 2 FORMAT (*ABCDE*)
120 PRINT 2
READY.
```

## TRANS DIRECTIVE

This directive applies only to programs written in BASIC. When a BASIC program was not created in the text buffer, but was read into the buffer using the directive OLD, the lines have numbers; however, they are not copied internally to columns 76-80. These numbers may be duplicated in columns 76-80 using the directive:

```
TRANS Δ
```

The line numbers are then available as SETUP line numbers for editing and reference.

## TAB DIRECTIVE

To some extent, the user may determine the format of his line with the TAB directive which provides the ability to preset tab positions for lines to be entered. TAB causes internal skips to specified column positions. This directive is a convenience to users in the GENERAL mode particularly.

When the TAB key on the Teletype keyboard, or the ↓ key on the display keyboard is pressed, SETUP formats the next character entered to begin in the first tab position greater than the current column. No physical movement is made; the line is entered into the text buffer in the format determined by the TAB directive. As many as 20 tabs can be set.

To set one or more tabs, enter:

```
TAB,n1,n2,n3,...,n20Δ
```

The  $n_i$  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  $n_1$ ,  $n_2$  sets the second tab and so forth up to 20 positions. Tabs must be specified in ascending order.

Any TAB directive erases and overlays the previous TAB directive.

To clear all previous tabs, enter:

```
TAB,FΔ
```

Examples:

1. TAB/15/20/30 Δ

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

2. TAB,8,12 Δ

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

00100†FD†FILEA Δ

the entry is positioned as follows in the text buffer:

	1				7	7
8	2				3	6
FD	FILEA				PRG	00100

### LINE EDITING

To insert a new line, the 5-digit line number is entered followed by the line to be entered. 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 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 00010, 00020, 00030, 00040 have already been entered.

The user enters:

00010 Δ	Line 00010 is deleted.
00025 DO 100 K = 1,10 Δ	Insert line 00025 between lines 00020, and 00030.
00040 A = 450 Δ	Replace line 00040.
00050100 CONTINUE Δ	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

Following the hyphen or asterisk, the user designates the line or literal to be edited with one of the following code letters:

<u>Code</u>	<u>Meaning</u>
F	FORTRAN statement number
M	SETUP line number
I	Literal
C	Last line edited

Following F or M he enters the statement or line number; following I, the literal is entered within delimiters. A line or statement number can be referenced by specifying a number plus or minus a decimal increment or decrement of 1 - 99.

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:

<u>Code</u>	<u>Meaning</u>
D	Delete a literal
I	Insert a literal
R	Replace a literal

When D or I 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. These delimiters may be any typed character, but pairs must be the same character.

Examples:

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

```
*F100D/THE ARRAY NAMES/Δ
```

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

2. To insert the literal I=1, 100 following the literal DO 200 in the fifth line after SETUP line number 50:

```
-M50+5I*DO 200*I/K=1,100/Δ
```

Asterisk and slash pairs act as delimiters.

The corrected line is displayed:

```
DO 200 K = 1,100
```

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

```
*I/GO TP/ R /TP//TO/Δ
```

Slashes are delimiters; the corrected line is not displayed.

4. To delete the literal \*5 from the line previously edited:

-CDX\*5X Δ

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

## 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/Δ

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/mΔ

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:

1. To delete lines 0010 through 0100:

DELETE,0010,0100 Δ

2. To delete line number 345:

DELETE /345/Δ

or enter:

345 Δ

## CHARACTER DELETION

Before a line is entered in the text buffer, it may be corrected using the left arrow on the Teletype or the BKSP key on the display terminal.

The left arrow (←) on the teletypewriter keyboard will delete the character just entered, and the correct character may be entered.

Example:

100 PRE ← OGA ← RAM ← ~~TEX~~ ← ST Δ

This line is entered in the text buffer as:

100 PROGRAM TEST

The BKSP key on the display keyboard will delete the preceding character each time it is pressed. The correct characters may then be entered. The character does not disappear on the display until it is overwritten, but each time BKSP is pressed the entry marker moves back one space, and a character entered at the new entry marker position overwrites any previously entered characters.

Example:

```
100 PREGEAMTEXT-----  
                    ↑  
                    entry marker
```

Press BKSP until the entry marker is below the first character to be corrected:

```
100 PREGEAMTEXT-----  
    ↑  
    └── entry marker
```

Enter the correct character, and then SKIP to move the entry marker to the next character to be corrected:

```
100 PROGEAMTEXT-----  
    ↑  
    └── entry marker
```

Enter the correct character, press SKIP 6 times, make the last correction, press SKIP twice more and then press SEND to enter the correct line in the text buffer:

```
100 PROGRAM TEXTΔ-----  
                    ↑  
                    entry marker
```

## FILE STORAGE AND MANIPULATION

### SAVE DIRECTIVE

To save the contents of the text buffer on disk, the user enters:

SAVE Δ

The file in the text buffer is saved as a regular SCOPE file, which is not affected by the user's LOGOUT from or subsequent LOGIN to the INTERCOM I system. Its name is that specified by the user in the OLD or NEW directive. If a file with the same name already exists on disk, contents of the text buffer replace the file on disk.

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 PERMANENT directive. It creates user private file which are listed by the command FILES.Δ, and counted when the user enters ASSETS.Δ.

### UNSAVE DIRECTIVE

A file may be removed from disk storage by entering:

UNSAVE Δ

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.

Abbreviated Entry:

UNSAVE/filename Δ

### RENAME DIRECTIVE

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

RENAME Δ

The system responds:

NEW FILE NAME-

The new name of the file is entered. If the previous name of the file in the text buffer duplicates a file name on disk, renaming the text buffer does not affect the file on disk in any way.

Abbreviated Entry:

RENAME/filenameΔ

Example:

1. A file named A-FILE has been constructed in the text buffer.

```
SAVEΔ
.
.
.
OLD/A-FILEΔ
```

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

```
RENAME/B-FILEΔ
SAVEΔ
```

The new B-FILE is stored on disk; A-FILE is still stored on disk in its original form and may be removed from disk with:

```
UNSAVE/A-FILEΔ
```

## LIST DIRECTIVE

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

LISTΔ Lists current contents of the buffer in line number sequence.

LIST/n/mΔ Lists lines n through m of the buffer.

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

Printing of the list may be terminated by pressing the S key followed by SEND:

```
SΔ
```

As soon as printing terminates, the user may input a command or data. He remains in SETUP.

Example:

```
LISTΔ
100 PROGRAM A(OUTPUT)
120 PRINT 2
167 2 FORMAT (* HI INTERCOM *)
190 END
```

## SCRATCH DIRECTIVE

Current contents of the buffer may be erased by entering:

SCRATCHΔ

Any file on disk corresponding to the file in the text buffer is not affected.

## PERMANENT DIRECTIVE

A permanent file is not only saved across LOGOUT and LOGIN, it also is protected from unauthorized access and from accidental destruction during normal system operation, including deadstart.

The user makes a SETUP file permanent by entering:

PERMANENTΔ

The system responds with a series of four questions. As the first is answered by the user, the next is displayed, and so forth, until all questions have been answered.

<u>System Query</u>	<u>User Response</u>
WHAT IS THE FILE NAME-	SETUP name of the file
WHAT IS THE PERMANENT FILE NAME-	Name which will identify the file after it has been made permanent. This name may be the same as the SETUP file name, or it may be different.
WHAT IS THE CYCLE NUMBER-	Either the word, NONE, or a decimal number 1-63. The cycle number identifies the file when more than one file is cataloged under a single permanent file name. If none is specified, the file is assumed to be the only file cataloged under the specified permanent file name.
ARE THERE ANY OTHER PARAMETERS-	Either the word, NO, or the parameters entered separately, followed by the word END also entered separately.  Parameters include such information as the retention period for the file, the name of the file creator, and sub-directory number used in cataloging a new cycle.

When these questions have been answered, the system has sufficient information to make a permanent SCOPE file of user's file.

Examples:

1. To enter a SETUP file as a permanent file in the SCOPE system with no special handling:

PERMANENT Δ

WHAT IS THE FILE NAME-

MYFILE Δ

WHAT IS THE PERMANENT FILE NAME-

MYFILE Δ

WHAT IS THE CYCLE NUMBER-

NONE Δ

ARE THERE ANY OTHER PARAMETERS-

NO Δ

2. To enter a file as a permanent file, which is one of several files cataloged under the permanent file name, and specify the originator and retention period:

PERMANENT Δ

WHAT IS THE FILE NAME-

A-FILE Δ

WHAT IS THE PERMANENT FILE NAME-

P-JM Δ

WHAT IS THE CYCLE NUMBER-

10 Δ

ARE THERE ANY OTHER PARAMETERS-

RD = 14 Δ

SD = 10 Δ

ID = JMARTIN Δ

END Δ

Retention period is 14 days  
Sub-directory number is 10.

## 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.

### COMPILE ONLY

RUN Directive:

A terminal user writing in either BASIC or FORTRAN may compile the contents of the text buffer with no subsequent execution by entering the directive:

RUNΔ

The user's program will be compiled by the BASIC or FORTRAN compiler depending on the current system mode.

### BASIC COMPILE AND EXECUTE

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

BASICXΔ

The contents of the text buffer are compiled; and if no compilation errors occurred, 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 reenter the BASICX directive.

### FORTRAN RUN COMPILE AND EXECUTE

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

RUNERΔ

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 slightly faster compilation without an error listing, the user enters:

RUNXΔ

The program will be compiled and subsequently executed regardless of compilation errors. If the program terminates because of errors, an error listing can be generated by entering the PEANUTS command (section 3). RUNX should be employed only when the user is fairly certain he has an error free program.

## **FORTRAN EXTENDED COMPILE AND EXECUTE**

Similar to RUNER for FORTRAN RUN, FTNER will compile a program written in FORTRAN EXTENDED.

FTNERΔ

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 faster compilation and subsequent execution, the user enters:

FTNΔ

This directive calls the FORTRAN EXTENDED compiler. Contents of the text buffer are compiled and executed regardless of compilation errors. No error list is produced; however, if execution terminates because of errors, the user may request an error listing with the DIAGNOS command (section 3).

## **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:

COBOLERAΔ

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:

COBOLXAΔ

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 DIAGNOS command (section 3).

Example:

The user has created and saved the following FORTRAN program:

```
100 PROGRAM A(INPUT,OUTPUT)
120 PRINT 2
131 23 CONTINUE
132 READ 3, A
133 3 FORMAT( F10.5)
136 AS-A:A
139 PRINT 3, AS
167 2 FORMAT(* THIS IS CORRECT*)
180 GO TO 23
190 END
```

Note the error in line 136;  
it should be AS=A\*A.

He then enters the directive to run and execute with an error summary:

```
RUNERΔ
```

The system responds:

```
PROGRAM TRANSFERRED TO COMPILER
COMPILING A
000001 FORTRAN ERRORS
100 PROGRAM A(INPUT,OUTPUT)
136 AS-A:A
FM** UNRECOGNIZABLE STATEMENT (FATAL)
```

Compiler notes error and names  
program, line in error and,  
type of error.

```
READY.
```

System expects user response.

The user corrects the line and enters the corrected line in his program with SAVE:

```
136 AS=A*AΔ
SAVEΔ
```

```
READY.
```

And he re-enters the program for compilation and execution:

```
RUNERΔ
```

The system responds:

```
PROGRAM TRANSFERRED TO COMPILER
COMPILING A
```

```
THIS IS CORRECT
```

This is the printout from line 120

The system waits and the user enters the value of A according to the format F10.5:

```
2.Δ
```

The system then outputs the value of AS:

```
4.00000
```

If the user enters a new value for A:

3.Δ

The system responds with a new value for AS:

9.00000

### **KILL DIRECTIVE**

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

If he is operating at a Teletype terminal he presses the # key.

At a display terminal, he presses %Δ

The program currently executing terminates and INTERCOM returns to command mode. Contents of the text buffer are lost, but they may be retrieved by entering the following command:

SETUP./OLD/SETFILE.Δ

This command re-enters the user in SETUP and calls his file back into the text buffer. The file will be the same as it was prior to compilation and execution.



# HARDWARE CONFIGURATION

A

---

## Minimum Configuration:

Configuration for 6400 Computer under SCOPE 3 operation  
6671 or 6676 Multiplexer  
One Teletype or Display Terminal

## Maximum Configuration:

Configuration for 6600 Computer under SCOPE 3 operation  
Two 6671 or one 6671 and one 6676 Multiplexers  
16 Terminals

A typical configuration with seven terminals, one 6671 and one 6676 multiplexer is shown in figure A-1. Six terminals are Teletype, one is a display terminal. Four Teletype terminals are connected via the 6676 multiplexer to the central computer; two Teletype terminals and one display terminal are connected via the 6671 multiplexer.

## 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 predominantly computational, input/output or text editing
- Size of time slice used by INTERCOM
- Use of extended core storage

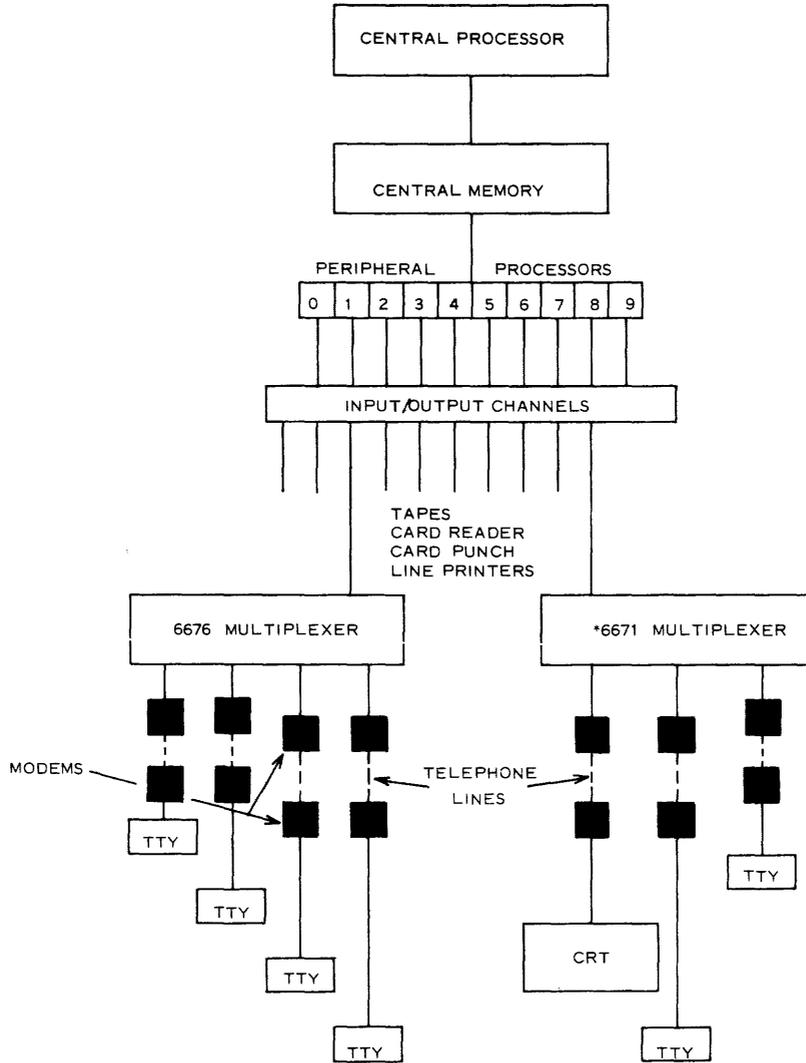
The best response time for any given number and type of terminal user is achieved with a large central memory, multiple control points assigned to INTERCOM and a fast mass storage device with no batch competition for access.

INTERCOM requires one dedicated multiplexer, 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 seven.

Central memory requirements vary with the buffer size specified by a user. They may be calculated with the following formula:

$$1000 + 8T + 2LT$$

where T = number of terminals and L = buffer size.



TTY Teletype Terminal

CRT Display Terminal

Figure A-1. Typical INTERCOM I Configuration with 7 Terminals

## ERROR MESSAGES

**B**

Message	Meaning
ARITHMETIC ERROR	Central processor error exit occurred.
CONTROL CARD ERROR	Illegal control card or incorrect command entered.
CPU ABORT	Central processor program requested that job be terminated abnormally.
DISCONNECT	Terminal disconnected briefly. Last command should be re-entered.
LINE TOO LONG	Line exceeds 80 characters.
LOADER ERROR	Program not successfully loaded.
PP CALL ERROR	Illegal peripheral processor call entered by central processor program.
PPU ABORT	Peripheral processor encountered illegal request, such as illegal file name or request to write outside field length of job.
ROLLOUT RECALL ERROR	Operator tried to run too many programs from one main program.
SORRY, I DO NOT UNDERSTAND	SCOPE/INTERCOM does not recognize this entry.
THE MYSTERY GUEST SHALL PLEASE LOGIN	Operator did not enter LOGIN as first entry.
TIME LIMIT	Job used all central processor time allotted to it. Job will terminate if an attempt is made to use CPU without increasing time limit.
USER ABORT	User requested that job be dropped.
YOU ARE NOT AUTHORIZED TO USE THIS PROGRAM	User is not allowed access to this file.
ECS PARITY ERROR	ECS parity error during system storage move terminated job.
FIELD LENGTH REQUEST EXCEEDS AUTHORIZATION	More field length requested than authorized.
HUNG IN AUTOMATIC RECALL	Periodic recall on file with no I/O request outstanding.

Message	Meaning
JOB CARD ERROR	Error in job card terminated job.
OPERATOR DROP	Operator typed M.DROP, job terminated.
OPERATOR KILL	Operator typed M.KILL, job terminated.
PP CALL WITH RECALL ERROR	PP program called with recall, but MIR found complete bit in FET set.
RERUN	Job started over again, operator typed M.RERUN.
TIME REQUEST EXCEEDS AUTHORIZATION	More time requested than authorized in user table.
YOU ARE OUT OF TIME	User exceeded time limit specified in user table.
YOU HAVE TOO MANY FILES - PLEASE DROP SOME	File limit specified in user table has been reached.

# COMMAND MESSAGES

C

Command	Message	Meaning	User Action
ASSETS.	COMMAND---	Success - current terminal status was printed	Enter any legal command.
BATCH.	TYPE FILE NAME	System ready to accept existing file name from user.	Enter existing file name.
	TYPE DISPOSITION	System ready to accept disposition name from user.	Type one : COMMON            PUNCH INPUT             PUNCHB PRINT             RENAME PRIVATE
	INVALID DISPOSITION PLEASE TRY AGAIN	Disposition entered not allowed.	Try again.
	INVALID FILE NAME - PLEASE TRY AGAIN	File name does not exist or is not allowed.	Enter another file name.
	DUPLICATE FILE NAME		Rename file.
	ENTER NEW NAME	Follow RENAME disposition - System ready to accept new name.	Enter new file name.
	THIS IS A RESERVED FILE	This file cannot be made private.	Enter another file name.
	COMMAND ---	Success - file manipulation completed.	Enter any legal command.
CONNECT.	COMMAND---	Success - files routed to terminal.	Enter any legal command.
DIAGNOS.			See Execution Diagnostics in pertinent reference manual.
DISCONT.	COMMAND---	Success - files routed to mass storage.	Enter any legal command.

Command	Message	Meaning	User Action
DROP.	TYPE FILE NAME	System ready to accept existing file name or return to command mode.	Type existing file name or END.
	COMMAND---	Specified files dropped.	Enter any legal command.
EFL.	COMMAND--	Success - field length altered.	Enter any legal command.
ETL.	COMMAND--	Success - time limit altered.	Enter any legal command.
FILES.	--PRIVATE FILES--	Following are user's private files.	None.
	COMMAND---	All available files listed.	Enter any legal command.
LOGIN.	mm/dd/yy ON AT *xx.xx.xx	Time and date user connected to INTERCOM.	None.
	TYPE VALID USER NAME	System ready to accept user name.	Type user name.
	YOU HAVE HAD THREE TRIES - GET HELP	Incorrect user name entered three times.	Obtain correct user name and log in again.
	TYPE PASSWORD---	System ready to accept password.	Type password.
	USER ALREADY LOGGED IN	Another user is already logged in under name given.	Type different password.
	COMMAND---	System ready to accept legal command.	Enter any legal command.

Command	Message	Meaning	User Action
LOGOUT.	YOU NO LONGER OWN ANY PRIVATE FILES	System will drop user's private files before logging user out.	None.
	mm/dd/yy OFF AT *xx.xx.xx	Time and date user is disconnected from INTERCOM.	None.
PEANUTS.			See Execution Diagnostics in FORTRAN Reference Manual.
SEND.	TO WHOM---	System ready to accept user name.	Type user name or ***.
	TYPE MSG OR ERROR	System ready to accept message or return to command mode.	Enter message or END
	COMMAND---	Message sent.	Enter legal command.
SETUP.	ON AT *xx.xx.xx	Informative message: time SETUP is entered.	None.
	SYSTEM-FORTRAN.	Default system assumed.	Specify new system, if desired.
	NEW OR OLD FILE.	System ready to receive SETUP directives. INTERCOM command also can be entered.	Specify directive or command.
SITUATE.	COMMAND---	Currently logged in user names printed.	Enter any legal command.



# SETUP DIRECTIVES

D

Directive	Message	Meaning	User Action
BASICX	PROGRAM TRANSFERRED TO COMPILER	Program is compiling and will be executed if error free.	None.
BYE or GOODBYE or END	OFF AT *xx.xx.xx	Last informative message upon normal exit of SETUP and return to command mode.	None.
COBOLR	PROGRAM TRANSFERRED TO COMPILER	Program is compiling and will be executed if error free. Error summary will be produced.	None.
COBOLX	PROGRAM TRANSFERRED TO COMPILER	Program is compiling and will be executed if error free.	None.
DELETE or DELETE /x or DELETE /x/y	READY.	Success - lines deleted.	Enter legal directive or command.
	WHAT?	Line number contains invalid character; or second line precedes first.	Correct error.
FTN	PROGRAM TRANSFERRED TO COMPILER	Program is compiling and will be executed if error free.	None.
FTNER	PROGRAM TRANSFERRED TO COMPILER	Program is compiling and will be executed if error free. Error summary will be produced.	None.

Directive	Message	Meaning	User Action
LIST or LIST/x or LIST/x/y	READY.	Success - lines listed.	Enter legal directive or command.
	WHAT?	Line number specification contained non-digit.	Type line numbers again.
	NO FILE.	Nothing in text buffer.	Bring file into text buffer.
NEW	NEW FILE NAME---	System ready to accept new file name.	Type new file name.
	READY.	System ready to accept new file input into text buffer.	Type input.
	FILE NAME MUST BEGIN WITH LETTER	Name entered was not acceptable	Try another name.
	FILE NAME MAY HAVE ONLY ALPHANUMERIC CHARACTERS	Name entered was not acceptable	Try another name.
OLD	OLD FILE NAME---	System ready to accept existing file name from user.	Type existing file name.
	NO FILE.	Did not find file by name.	Type existing file name.
	READY.	File read into text buffer.	Begin file manipulation.

Directive	Message	Meaning	User Action
PERMANENT	WHAT IS THE FILE NAME	System ready to accept file name.	Type file to be made permanent.
	WHAT IS THE PERMANENT FILE NAME	System ready to accept new file name.	Type new file name.
	WHAT IS THE CYCLE NUMBER	System ready to accept cycle number.	Type cycle number or NONE.
	ARE THERE ANY OTHER PARAMETERS	System ready to accept parameters.	Type NO or parameters separated by carriage returns and terminated with END.
	CATALOGED IN SDXXX, ETC.	System completed permanent function.	Enter legal directive or command.
	FUNCTION UNSUCCESSFUL	System unable to comply with request.	Repeat process correctly.
RENAME	NEW FILE NAME---	System ready to accept new file name.	Type new file name.
RENAME	FILE MUST BEGIN WITH A LETTER.		
	FILE MAY HAVE ONLY ALPHANUMERIC CHARACTERS.	File name not acceptable	Try again.
	READY.	Success - file renamed.	Enter legal directive or command.
RESEQ	READY.	Success - text re-sequenced.	Enter legal directive or command.
RUN	PROGRAM TRANSFERRED TO COMPILER.	Program is compiling and will be executed if error free.	None.
	READY.	Success - program executed.	Enter legal directive or command.

Directive	Message	Meaning	User Action
RUNER	PROGRAM TRANSFERRED TO COMPILER.	Program is compiling and will be executed if error free.	None.
	READY.	Success - program executed.	Enter legal directive or command.
RUNX	PROGRAM TRANSFERRED TO COMPILER.	Program is compiling and will execute if error free.	None.
	READY.	Success - program executed.	Enter legal directive or command.
SAVE	READY.	Success - file saved.	Enter legal directive or command.
SCRATCH	READY.	Success - text buffer is cleared.	Enter legal directive or command.
SYSTEM	NEW SYSTEM---	The system is ready to accept a system name.	Type new system name.
	READY.	New system name has been accepted.	Enter legal directive or command.
TAB	READY.	Success - tab stops have been entered.	Enter legal directive or command.
TRANS	READY.	Success - the BASIC line numbers have been duplicated in columns 76-80.	Enter legal directive or command.
UNSAVE	OLD FILE NAME.	System ready to accept existing file name from user.	Type existing file name.
	READY.	Success - file is not saved.	Enter legal directive or command.

MESSAGES IN RESPONSE TO EDITING WITHIN A LINE

Message	Meaning	User Action
CODE UNRECOGNIZABLE		Correct code.
THE LITERAL IS TOO LARGE	Literal exceeds 60 characters.	Use shorter literal.
NO MATCH ON LITERAL FOUND	SETUP did not find Matching literal in text buffer.	Type existing literal.
FORMAT ERROR	Editing statement in- correctly entered.	Correct line.
LINE IS A LITTLE TOO LONG FOR ME	FORTRAN statement (in- cluding line number) exceeds 66 characters.	Type shorter line.



# CODE CONVERSION TABLE

E

<u>DISPLAY CODE</u>	<u>DISPLAY CHAR.</u>	<u>TTY CHAR.</u>	<u>TTY CODE</u>	<u>CRT CHAR.</u>	<u>CRT CODE</u>	<u>TTY MUX CODE</u>	<u>CRT MUX CODE</u>
00	blank	Space	040	Space	120	4101	4320
01	A	A	101	A	061	4202	4061
02	B	B	102	B	062	4204	4062
03	C	C	103	C	063	4207	4063
04	D	D	104	D	064	4210	4064
05	E	E	105	E	065	4213	4265
06	F	F	106	F	066	4152	4266
07	G	G	107	G	067	4216	4067
10	H	H	110	H	070	4220	4070
11	I	I	111	I	071	4223	4271
12	J	J	112	J	041	4225	4241
13	K	K	113	K	042	4226	4242
14	L	L	114	L	043	4231	4043
15	M	M	115	M	044	4232	4244
16	N	N	116	N	045	4234	4045
17	O	O	117	O	046	4237	4046
20	P	P	120	P	047	4240	4247
21	Q	Q	121	Q	050	4243	4250
22	R	R	122	R	051	4245	4051
23	S	S	123	S	122	4246	4122
24	T	T	124	T	123	4251	4323
25	U	U	125	U	124	4252	4124
26	V	V	126	V	125	4254	4325
27	W	W	127	W	126	4257	4326
30	X	X	130	X	127	4261	4172
31	Y	Y	131	Y	130	4262	4130
32	Z	Z	132	Z	131	4264	4331
33	0	0	060	0	112	4140	4112
34	1	1	061	1	101	4143	4301
35	2	2	062	2	102	4145	4302
36	3	3	063	3	103	4146	4103
37	4	4	064	4	104	4151	4304
40	5	5	065	5	105	4152	4105
41	6	6	066	6	106	4154	4106
42	7	7	067	7	107	4157	4307
43	8	8	070	8	110	4161	4310
44	9	9	071	9	111	4162	4111
45	+	+	053	+	060	4126	4260
46	-	-	055	-	040	4132	4040
47	*	*	052	*	054	4125	4054
50	/	/	057	/	121	4137	4121
51	(	(	050	(	134	4120	4334
52	)	)	051	)	074	4123	4274
53	\$	\$	044	\$	053	4110	4253
54	=	=	075	=	113	4173	4313
55	blank	space	040	space	120	4101	4320
56	,	,	054	,	133	4131	4133

DISPLAY CODE	DISPLAY CHAR.	TTY CHAR.	TTY CODE	CRT CHAR.	CRT CODE	TTY MUX CODE	CRT MUX CODE
57	.	.	056	.	073	4134	4073
60	"	"	042	"	136	4104	4136
61	[	[	133/50	[	117	4167	4117
62	]	]	125/51	]	132	4273	4332
63	:	:	072	:	100	4164	4100
64	≠	,	047	≠	114	4116	4114
65	→	%,Rubout	045/177		135	4102/4377	4235
66	√	/	134/57	√	052	4270	4052
67	^	&	046	^	137	4115	4337
70	↑	↑	136	↑	055	4275	4054
71	↓	TAB,LF	011/012	↓	056	4022/4024	4255
72	<	<	074	<	057	4175	4057
73	>	>	076	>	072	4170	4272
74	≤	CR	015	≤	115	4257/4033	4315
75	≥	@	100	≥	075	4201	4075
76	└─┘	←	137	%	116	4276	4316
77	;	;	073	;	077	4167	4277
76	└─┘	?	077			4176	4277
Abort			INT. %				
Backspace		!	Bksp.				
Erase		CR	Reset				
EOL		LF	Send				

# INDEX

---

## ADD

ADD User-name/Password entry to Password file 3-12

## Alphanumeric

Alphanumeric Teletype-key 2-5

## ASSETS

ASSETS Command lists User-name/Password entry status 3-13

## BASIC

SETUP Line-numbers reformatted for FORTRAN and BASIC 4-5

SYSTEM Directive specifies BASIC or GENERAL 4-2

RUN Directive BASIC or FORTRAN Compilation 4-16

BASIC Directive BASIC Compilation and Execution 4-16

BASIC Diagnostics 3-18

TRANS Directive transfers BASIC Line-numbers 4-7

## BATCH

INPUT Disposition BATCH Command specifies Input File 3-16

PRINT Disposition BATCH Command prints Output file 3-16

PUNCH Disposition BATCH Command punches Hollerith Cards 3-16

PUNCHB Disposition BATCH Command punches Binary Cards 3-16

COMMON Disposition BATCH Command specifies Common File 3-16

PRIVATE Disposition BATCH Command specifies Private Files 3-16

RENAME Disposition BATCH Command renames File 3-16

BATCH Command enters Job in SCOPE Batch Queue 3-15

Exit from BATCH Command with SCRAM or END 3-15

BATCH Command Dispositions 3-16

## Backspace

Backspace Teletype-key 2-4

Character Deletion with Backspace key Teletype 4-10

## Binary

PUNCHB Disposition BATCH Command punches Binary Cards 3-16

## BKSP

Character Deletion with BKSP key Display 4-11

BKSP Display-key 2-8

## BREAK

BREAK Teletype-key 2-5

## BYE

Exit from SETUP with BYE GOODBYE or END 4-4

## Cards

PUNCH Disposition BATCH Command punches Hollerith Cards 3-16

PUNCHB Disposition BATCH Command punches Binary Cards 3-16

## Character

Character Deletion with Backspace key Teletype 4-10

Character Deletion with BKSP key Display 4-11

Code to Character Conversion Table E-1

## CLEAR

CLEAR Display-key 2-8

COBOL  
     DIAGNOS Command lists Diagnostics: FORTRAN-EXT COBOL 3-18  
     COBOL - See GENERAL  
 COBOLER  
     COBOLER or COBOLX Directive COBOL Compilation and Execution 4-17  
 COBOLX  
     COBOLER or COBOLX Directive COBOL Compilation and Execution 4-17  
 COMMON  
     COMMON Disposition BATCH Command specifies Common File 3-16  
 CONNECT  
     CONNECT Command Connects Terminal to Input/Output Files 3-9  
 Code  
     Code to Character Conversion Table E-1  
 Comma  
     Slash Space Period Comma - See Delimiters  
 Command  
     Abbreviated form of Command 3-1  
     Logging in with LOGIN Command 3-2  
     Logging-out with LOGOUT Command 3-2  
     SITUATE Command lists Logged-in users 3-5  
     LOCK Command locks out Messages from other terminals 3-5  
     UNLOCK Command allows Messages to be received 3-5  
     SEND Command sends Messages to other terminals 3-5  
     ASSETS Command lists User-name/Password entry status 3-13  
     ETL Command Alters user's Time-Limit 3-14  
     EFL Command Alters user's Field-Length 3-14  
     BATCH Command enters Job in SCOPE Batch Queue 3-15  
     Exit from BATCH Command with SCRAM or END 3-15  
     BATCH Command Dispositions 3-16  
     SETUP Command calls SETUP Utility 3-17  
     PEANUTS Command lists Diagnostics: FORTRAN-RUN 3-18  
     DIAGNOS Command lists Diagnostics: FORTRAN-EXT COBOL 3-18  
     Command Message Summary C-i  
 Commands  
     Inter-terminal Communication Commands 3-5  
     File Control Commands 3-7  
     Submitting Job Commands 3-15  
     Diagnostic Commands 3-18  
 Command-format  
     INTERCOM I Command-format 3-1  
 Command  
     FILES Command Lists User Files 3-7  
     PAGE Command Displays Screen 3-7  
     DROP Command Drops user Files 3-8  
     CONNECT Command Connects Terminal to Input/Output Files 3-9  
     DISCONT Command Disconnects Terminal from Input/Output Files 3-10  
 Common  
     Definition Common Files 1-2  
     COMMON Disposition BATCH Command specifies Common File 3-16  
 Communication  
     Communication Between Terminals 3-5  
 Compilation  
     RUN Directive BASIC or FORTRAN Compilation 4-16  
     BASIC Directive BASIC Compilation and Execution 4-16  
     RUNER or RUNX Directive FORTRAN-RUN Compilation and Execution 4-16  
     Directive FORTRAN-EXT Compilation and Execution 4-17  
     COBOLER or COBOLX Directive COBOL Compilation and Execution 4-17  
     Sample FORTRAN-RUN Compilation and Execution 4-18

Configuration

Minimum Hardware Configuration A-1  
Maximum Hardware Configuration A-1  
Typical Hardware Configuration A-2

Connects

CONNECT Command Connects Terminal to Input/Output Files 3-9

Conversion

Code to Character Conversion Table E-1

CRT

CRT - See Display

CTRL

CTRL Teletype-key 2-4

DEL

DEL deletes User-name/Password entry from Password file 3-12

DELETE

DELETE Directive Deletes entire Line 4-10

Delete

Delete Directive or Replace Literal in Line 4-9

Deletion

Line Insertion or Deletion 4-8

Character Deletion with Backspace key Teletype 4-10

Character Deletion with BKSP key Display 4-11

Delimiters

SETUP Legal Delimiters 4-2

Intra-line Editing Delimiters 4-9

DIAGNOS

DIAGNOS Command lists Diagnostics: FORTRAN-EXT COBOL 3-18

DISCONT

DISCONT Command Disconnects Terminal from Input/Output Files 3-10

Diagnostics:

PEANUTS Command lists Diagnostics: FORTRAN-RUN 3-18

DIAGNOS Command lists Diagnostics: FORTRAN-EXT COBOL 3-18

Diagnostics

Diagnostics - See Error Messages B-1

BASIC Diagnostics 3-18

Directive

SETUP Directive Specification 4-2

SYSTEM Directive specifies BASIC or GENERAL 4-2

OLD Directive requests existing File 4-3

NEW Directive specifies NEW File 4-3

RESEQ Directive resequences FORTRAN or GENERAL Line-numbers 4-6

TRANS Directive transfers BASIC Line-numbers 4-7

TAB Directive sets Line Tabulation 4-7

Delete Directive or Replace Literal in Line 4-9

DELETE Directive Deletes entire Line 4-10

SAVE Directive saves Text-Buffer as Private File 4-12

UNSAVE Directive Removes Private File 4-12

RENAME Directive Renames Text-Buffer 4-12

LIST Directive Lists Text-Buffer 4-13

SCRATCH Directive erases Text-Buffer 4-14

PERMANENT Directive specifies Permanent File 4-14

RUN Directive BASIC or FORTRAN Compilation 4-16

BASIC Directive BASIC Compilation and Execution 4-16

RUNER or RUNX Directive FORTRAN-RUN Compilation and Execution 4-16

Directive FORTRAN-EXT Compilation and Execution 4-17

COBOLER or COBOLX Directive COBOL Compilation and Execution 4-17

KILL Directive stops Execution with Interrupt key 4-19

SETUP Directive Summary D-1

Disconnects  
     DISCONT Command Disconnects Terminal from Input/Output Files 3-10

Displays  
     PAGE Command Displays Screen 3-7

Display  
     Display Terminal Initiation 2-1  
     Display Terminal Operation 2-5  
     Display Screen format 2-5  
     Display Screen Line-indicator and Entry-marker 2-5  
     Message Terminator Display Screen 2-6  
     Display Terminal Keyboard 2-7

Display-key  
     SEND Display-key 2-8  
     RETURN Display-key 2-8  
     Tabulation Display-key 2-8  
     SPACE Display-key 2-8  
     CLEAR Display-key 2-8  
     BKSP Display-key 2-8  
     REPT Display-key 2-8  
     SKIP Display-key 2-8  
     SEND-INDEX Display-key 2-8  
     RESET Display-key 2-9  
     LINE SKIP key Display-key 2-9  
     Interrupt Display-key 2-9

DROP  
     DROP Command Drops user Files 3-8

Drops  
     DROP Command Drops user Files 3-8

Editing  
     Line Editing 4-8  
     Intra-line Editing corrects Line in Text-buffer 4-8  
     Intra-Line Editing Messages D-5  
     Editing Text-buffer using Line-numbers 4-5

EFL  
     EFL Command Alters user's Field-Length 3-14

END  
     Exit from BATCH Command with SCRAM or END 3-15  
     Exit from SETUP with BYE GOODBYE or END 4-4

Entering  
     Entering message with SEND(Display) or RETURN(Teletype) 3-1

Entry-marker  
     Display Screen Line-indicator and Entry-marker 2-5

Error  
     Error Messages B-1

ETL  
     ETL Command Alters user's Time-Limit 3-14

Execution  
     BASIC Directive BASIC Compilation and Execution 4-16  
     RUNER or RUNX Directive FORTRAN-RUN Compilation and Execution 4-16  
     Directive FORTRAN-EXT Compilation and Execution 4-17  
     COBOLER or COBOLX Directive COBOL Compilation and Execution 4-17  
     Sample FORTRAN-RUN Compilation and Execution 4-18  
     KILL Directive stops Execution with Interrupt key 4-19

Exit  
     Exit from SETUP with BYE GOODBYE or END 4-4

## FILES

FILES Command Lists User Files 3-7

### Field-Length

EFL Command Alters user's Field-length 3-14

### File

INPUT Disposition BATCH Command specifies Input File 3-16

COMMON Disposition BATCH Command specifies Common File 3-16

RENAME Disposition BATCH Command renames File 3-16

OLD Directive requests existing File 4-3

NEW Directive specifies NEW File 4-3

SAVE Directive saves Text-Buffer as Private File 4-12

UNSAVE Directive Removes Private File 4-12

PERMANENT Directive specifies Permanent File 4-14

File Storage and Manipulation' 4-12

### Files

Definition Private Files 1-2

Definition Common Files 1-2

Definition System Files 1-2

Definition Permanent Files 1-2

FILES Command Lists User Files 3-7

DROP Command Drops user Files 3-8

CONNECT Command Connects Terminal to Input/Output Files 3-9

DISCONT Command Disconnects Terminal from Input/Output Files 3-10

PRIVATE Disposition BATCH Command specifies Private Files 3-16

### FORTRAN

FORTRAN is SYSTEM default 4-2

SETUP Line-numbers reformatted for FORTRAN and BASIC 4-5

RESEQ Directive resequences FORTRAN or GENERAL Line-numbers 4-6

### FORTRAN-RUN

PEANUTS Command lists Diagnostics: FORTRAN-RUN 3-18

### FORTRAN-EXT

DIAGNOS Command lists Diagnostics: FORTRAN-EXT COBOL 3-18

### FORTRAN

RUN Directive BASIC or FORTRAN Compilation 4-16

### FORTRAN-RUN

RUNER or RUNX Directive FORTRAN-RUN Compilation and Execution 4-16

### FORTRAN-EXT

Directive FORTRAN-EXT Compilation and Execution 4-17

### FORTRAN-RUN

Sample FORTRAN-RUN Compilation and Execution 4-18

### FTN

FTNER or FTN 4-17

### FTNER

FTNER or FTN 4-17

## GENERAL

SYSTEM Directive specifies BASIC or GENERAL 4-2

SETUP Line-numbers reformatted for GENERAL 4-6

RESEQ Directive resequences FORTRAN or GENERAL Line-numbers 4-6

## GOODBYE

Exit from SETUP with BYE GOODBYE or END 4-4

## Hardware

Minimum Hardware Configuration A-1

Maximum Hardware Configuration A-1

Typical Hardware Configuration A-2

## Hollerith

PUNCH Disposition BATCH Command punches Hollerith Cards 3-16

## INPUT

INPUT Disposition BATCH Command specifies Input File 3-16

## INTERCOM

Terminals using INTERCOM I 1-1

INTERCOM I Command-format 3-1

Teaching INTERCOM I with TEACH Command 3-3

## Input

INPUT Disposition BATCH Command specifies Input File 3-16

## Input/Output

CONNECT Command Connects Terminal to Input/Output Files 3-9

DISCONT Command Disconnects Terminal from Input/Output Files 3-10

## Insertion

Line Insertion or Deletion 4-8

## Interrupt

Interrupt Teletype-key 2-5

Interrupt Display-key 2-9

KILL Directive stops Execution with Interrupt key 4-19

## Intra-Line

Intra-Line Editing Messages D-5

## Intra-line

Intra-line Editing corrects Line in Text-buffer 4-8

## Job

BATCH Command enters Job in SCOPE Batch Queue 3-15

Submitting Job Commands 3-15

## Keyboard

Teletype Terminal Keyboard 2-3

Display Terminal Keyboard 2-7

## KILL

KILL Directive stops Execution with Interrupt key 4-19

## LINE

LINE FEED Teletype-key 2-4

LINE SKIP key Display-key 2-9

## LIST

S key terminates LIST 4-13

LIST Directive Lists Text-Buffer 4-13

## Line

Delete Directive or Replace Literal in Line 4-9

DELETE Directive Deletes entire Line 4-10

Intra-line Editing corrects Line in Text-buffer 4-8

Line Editing 4-8

Line Insertion or Deletion 4-8

## Line-indicator

Display Screen Line-indicator and Entry-marker 2-5

## Line-numbers

Editing Text-buffer using Line-numbers 4-5

SETUP Line-numbers reformatted for FORTRAN and BASIC 4-5

SETUP Line-numbers reformatted for GENERAL 4-6

RESEQ Directive resequences FORTRAN or GENERAL Line-numbers 4-6

TRANS Directive transfers BASIC Line-numbers 4-7

## Lists

FILES Command Lists User Files 3-7

## Literal

Delete Directive or Replace Literal in Line 4-9

LOCK  
     LOCK Command locks out Messages from other terminals 3-5

LOGIN  
     Logging in with LOGIN Command 3-2

LOGOUT  
     Logging-out with LOGOUT Command 3-2

Logged-in  
     SITUATE Command lists Logged-in users 3-5

Logging  
     Logging in with LOGIN Command 3-2

Logging-out  
     Logging-out with LOGOUT Command 3-2

Messages  
     LOCK Command locks out Messages from other terminals 3-5  
     UNLOCK Command allows Messages to be received 3-5  
     SEND Command sends Messages to other terminals 3-5

NEW  
     NEW Directive specifies NEW File 4-3

OLD  
     OLD Directive requests existing File 4-3

Output  
     PRINT Disposition BATCH Command prints Output file 3-16

PAGE  
     PAGE Command Displays Screen 3-7

Password  
     User-name and Password formats 3-2  
     Maintaining Password file with PSSWORD card 3-11  
     ADD User-name/Password entry to Password file 3-12  
     DEL deletes User-name/Password entry from Password file 3-12  
     Altering status of User-name/Password Entry in Password file 3-13

PEANUTS  
     PEANUTS Command lists Diagnostics: FORTRAN-RUN 3-18

PERMANENT  
     PERMANENT Directive specifies Permanent File 4-14

Period  
     Slash Space Period Comma - See Delimiters

Permanent  
     Definition Permanent Files 1-2  
     PERMANENT Directive specifies Permanent File 4-14

PRINT  
     PRINT Disposition BATCH Command prints Output file 3-16

PRIVATE  
     PRIVATE Disposition BATCH Command specifies Private Files 3-16

Private  
     Definition Private Files 1-2  
     PRIVATE Disposition BATCH Command specifies Private Files 3-16  
     SAVE Directive saves Text-Buffer as Private File 4-12  
     UNSAVE Directive Removes Private File 4-12

PSSWORD  
     Maintaining Password file with PSSWORD card 3-11

PUNCH  
     PUNCH Disposition BATCH Command punches Hollerith Cards 3-16  
 PUNCHB  
     PUNCHB Disposition BATCH Command punches Binary Cards 3-16  
  
 RENAME  
     RENAME Disposition BATCH Command renames File 3-16  
     RENAME Directive Renames Text-Buffer 4-12  
 REPT  
     REPT Display-key 2-8  
 RESEQ  
     RESEQ Directive resequences FORTRAN or GENERAL Line-numbers 4-6  
 RESET  
     RESET Display-key 2-9  
 RETURN  
     RETURN Teletype-key 2-4  
     RETURN Display-key 2-8  
 RETURN(Teletype)  
     Entering message with SEND(Display) or RETURN(Teletype) 3-1  
 Replace  
     Delete Directive or Replace Literal in Line 4-9  
 Response  
     Terminal Response time A-1  
 RUNER  
     RUNER or RUNX Directive FORTRAN-RUN Compilation and Execution 4-16  
 RUNX  
     RUNER or RUNX Directive FORTRAN-RUN Compilation and Execution 4-16  
 RUN  
     RUN Directive BASIC or FORTRAN Compilation 4-16  
  
 S  
     S key terminates LIST 4-13  
 SAVE  
     SAVE Directive saves Text-Buffer as Private File 4-12  
 SCOPE  
     BATCH Command enters Job in SCOPE Batch Queue 3-15  
 SCRAM  
     Exit from BATCH Command with SCRAM or END 3-15  
 SCRATCH  
     SCRATCH Directive erases Text-Buffer 4-14  
 Screen  
     Message Terminator Display Screen 2-6  
     PAGE Command Displays Screen 3-7  
     Display Screen format 2-5  
     Display Screen Line-indicator and Entry-marker 2-5  
 SEND  
     SEND Display-key 2-8  
     SEND Command sends Messages to other terminals 3-5  
 SEND(Display)  
     Entering message with SEND(Display) or RETURN(Teletype) 3-1  
 SEND-INDEX  
     SEND-INDEX Display-key 2-8

SETUP  
     SETUP Legal Delimiters 4-2  
     SETUP Utility Features 4-1  
     Exit from SETUP with BYE GOODBYE or END 4-4  
     SETUP Command calls SETUP Utility 3-17  
     SETUP Directive Specification 4-2  
     SETUP Directive Summary D-1  
     SETUP Line-numbers reformatted for FORTRAN and BASIC 4-5  
     SETUP Line-numbers reformatted for GENERAL 4-6  
 SHIFT  
     SHIFT Teletype-key 2-4  
 SITUATE  
     SITUATE Command lists Logged-in users 3-5  
 SKIP  
     SKIP Display-key 2-8  
 Slash  
     Slash Space Period Comma - See Delimiters  
 SPACE  
     SPACE Display-key 2-8  
     SPACE Teletype-key 2-4  
 Space  
     Slash Space Period Comma - See Delimiters  
 SYSTEM  
     FORTRAN is SYSTEM default 4-2  
     SYSTEM Directive specifies BASIC or GENERAL 4-2  
 System  
     Definition System Files 1-2  
  
 TAB  
     TAB Teletype-key 2-4  
     TAB Directive sets Line Tabulation 4-7  
 Tabulation  
     Tabulation Display-key 2-8  
     TAB Directive sets Line Tabulation 4-7  
 TEACH  
     Teaching INTERCOM I with TEACH Command 3-3  
 Teaching  
     Teaching INTERCOM I with TEACH Command 3-3  
 Teletype  
     Teletype Terminal Initiation 2-1  
     Teletype Terminal Operation 2-3  
     Teletype Terminal Keyboard 2-3  
 Teletype-key  
     SHIFT Teletype-key 2-4  
     CTRL Teletype-key 2-4  
     RETURN Teletype-key 2-4  
     LINE FEED Teletype-key 2-4  
     TAB Teletype-key 2-4  
     SPACE Teletype-key 2-4  
     Backspace Teletype-key 2-4  
     Interrupt Teletype-key 2-5  
     BREAK Teletype-key 2-5  
     Alphanumeric Teletype-key 2-5  
 Terminals  
     Terminals using INTERCOM I 1-1

Terminal

- Teletype Terminal Initiation 2-1
- Display Terminal Initiation 2-1
- Teletype Terminal Operation 2-3
- Teletype Terminal Keyboard 2-3
- Display Terminal Operation 2-5
- Display Terminal Keyboard 2-7

Terminals

- Communication Between Terminals 3-5

Terminal

- Terminal Response time A-1

Terminator

- Message Terminator Display Screen 2-6

Text-Buffer

- Text-Buffer Description 4-1
- SAVE Directive saves Text-Buffer as Private File 4-12
- RENAME Directive Renames Text-Buffer 4-12
- LIST Directive Lists Text-Buffer 4-13
- SCRATCH Directive erases Text-Buffer 4-14

Text-buffer

- Editing Text-buffer using Line-numbers 4-5
- Intra-line Editing corrects Line in Text-buffer 4-8

Time-Limit

- ETL Command Alters user's Time-Limit 3-14

Time-Sharing

- Time-Sharing defined 1-1

TRANS

- TRANS Directive transfers BASIC Line-numbers 4-7

TTY

- TTY - See Teletype

UNLOCK

- UNLOCK Command allows Messages to be received 3-5

UNSAVE

- UNSAVE Directive Removes Private File 4-12

User-name

- User-name and Password formats 3-2

User-name/Password

- ADD User-name/Password entry to Password file 3-12
- DEL deletes User-name/Password entry from Password file 3-12
- Altering status of User-name/Password Entry in Password file 3-13
- ASSETS Command lists User-name/Password entry status 3-13

COMMENT SHEET



TITLE: 6000 Series Computer Systems  
INTERCOM I Reference Manual

February 1970

PUBLICATION NO. 60305300 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

FOLD

FOLD

FIRST CLASS  
PERMIT NO. 8241  
MINNEAPOLIS, MINN.

**BUSINESS REPLY MAIL**  
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

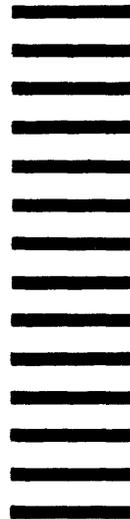
POSTAGE WILL BE PAID BY

**CONTROL DATA CORPORATION**

*Documentation Department*

**215 Moffett Park Drive**

**Sunnyvale, California 94086**



CUT ON THIS LINE

FOLD

FOLD

STAPLE

STAPLE





6000 SERIES INTERCOM I REFERENCE MANUAL



CORPORATE HEADQUARTERS, 8100 34th AVE. SO., MINNEAPOLIS, MINN, 55440  
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD